



University of
Nottingham

UK | CHINA | MALAYSIA

University of Nottingham

PhD Thesis

An Investigation of Neural Information Retrieval based Temporal Embeddings and Knowledge Extraction for Scientific Insight Generation

Student:

Ayman Salama

Supervisor:

Dr. Tomas Maul

Co-Supervisor:

Prof. Neil Crout

Student ID: 20024328

Faculty of Science and Engineering School of Computer Science

June 2024

Acknowledgments

I extend my deepest gratitude to Dr. Tomas for his unwavering support and guidance throughout the intricate journey of my doctoral studies. His combined expertise in the field and his nurturing approach have played an instrumental role in my professional development and personal growth during these years. I am immensely fortunate to have had the privilege of being mentored by such a distinguished academic. Furthermore, I am profoundly thankful to Prof. Neil, my co-supervisor, whose insightful advice and constructive feedback have been invaluable in refining my research and enhancing its quality. His academic prowess and dedication to excellence have left an indelible mark on my scholarly journey. Lastly, it is with an overflowing heart that I acknowledge the foundational role of my family. Words seem inadequate to convey the depth of my gratitude for their constant encouragement, unwavering belief, and unyielding love. Their tireless support has been my anchor, and it is to them that I owe the very essence of this accomplishment. To my beloved mother, who has been the beacon of light and hope in my life, I owe more than words can express. Her strength and wisdom have been the guiding forces behind every step I've taken. Mom, your sacrifices, and boundless efforts have shaped not only this research but also the person I've become. This achievement is as much yours as it is mine.

Publications

- Salama, A., Maul, T., Jahanshiri, E., Nizar, N. M. M. (2019) “Effective neural information retrieval in crop soil requirement classification”, 4th International conference on crop improvement. UPM.
- Salama, Ayman., Maul, T., Jahanshiri, Ebrahim. (2017). “Detecting cropping patterns of underutilised crops using online big data”, 2nd International conference on “Future Technology Conference”. IEEE.
- Salama, A., Maul, T., Jahanshiri, E., Crout, N. (2017) “Journey from text mining to neural information retrieval in agricultural data science”, ICBA 2017-4 International Conference on Big Data Applications in Agriculture. UPM.
- Nizar, N. M. M., Jahanshiri, E., Tharmandram, A. S., Salama, A., Sinin, S. S. M., Abdullah, N. J., ... & Azam-Ali, S. N. (2021). Underutilised crops database for supporting agricultural diversification. Computers and Electronics in Agriculture, 180, 105920.

Abstract

Humans today generate information at an unprecedented rate, leading to a vast accumulation of knowledge. This immense amount of data poses challenges in extracting both explicit information and implicit knowledge, predicting scientific events, and investigating scientific trends and trajectories. For decision-makers, it is particularly challenging to keep pace with this abundance to make informed decisions grounded in quantitative and up-to-date data. Current state-of-the-art methods, such as those employing Word2Vec models for semantic analysis, face several technical challenges when applied to vast and diverse datasets, including scalability issues, contextual ambiguity, difficulty in capturing implicit and explicit knowledge, and challenges with temporal embeddings. These models are also susceptible to bias and noise, and the impact of hyperparameter tuning on knowledge representation can lead to inconsistent results, further complicating their reliability for informed decision-making. While some existing research addresses the problem of knowledge evolution, they often fall short in handling large datasets, leading to scalability issues. Many studies focus primarily on changes in word meaning rather than the evolving relationships among entities, neglecting the broader context of temporal knowledge evolution. Additionally, the use of default hyperparameters in these models often overlooks their sensitivity to implicit and explicit knowledge extraction, resulting in inconsistent outcomes. Moreover, some techniques overly focus on event detection through specific text classifications, limiting their broader applicability in understanding complex knowledge dynamics over time. There is a lack of a coherent and efficient framework to represent, extract, and discover large-scale temporal knowledge to date. Therefore, this research aims to explore efficient, large-scale methodologies for extracting both explicit and implicit knowledge from extensive scientific literature and to build a big data, cloud-based knowledge evolution framework to identify scientific discoveries and their trajectories. Firstly, to build a large-scale knowledge evolution framework, it is important to study methods for efficient and accurate knowledge representation. To this end, as a case study, we investigate various embedding techniques for identifying relationships between underutilized crops and their attributes, such as global interest, vernacular and scientific names, and soil requirements. Word2Vec embeddings were analysed on extensive Wikipedia datasets, including multiple languages, and we found that our approach effectively

identified relationships between underutilized crops and their attributes. These findings were then compared with an international database of crop characteristics, which showed a 76.11% accuracy in predicting soil classifications using scientific crop names, surpassing semantic relationship extraction methods in the literature. Following this case study, we identified a suitable embedding method for accurate knowledge representation, addressing the scalability issue by leveraging a cloud-based framework capable of processing vast datasets efficiently and maintaining high predictive accuracy. To create a coherent temporal representation of knowledge and classify it effectively, it is vital to generate high-quality embeddings for these temporal vector spaces. To this end, we evaluated various hyperparameter combinations of Word2Vec (the embedding method identified for achieving the first objective) against diverse Deep Neural Network (DNN) architectures to gauge their impact on classification tasks using the Amazon Customer Review dataset. The results from this dataset, which demonstrated a generic use case for understanding the implications of hyperparameters on downstream tasks such as classification, suggest that the hyperparameter tuning method and values could be effectively applied to the process of creating the temporal vector spaces, ensuring reliable knowledge representation across varying contexts and timeframes. Finally, based on the findings from investigations on Word2Vec-based knowledge representation and its hyperparameter tuning for optimized representation for downstream tasks, we propose our Continuous Knowledge Evolution Construction method, where knowledge is represented with its temporal element. The temporal vector space is a dynamic framework that captures the evolution of knowledge over time by constructing word embeddings that reflect changes in language and concept significance across different periods. In this method, Word2Vec is used to generate embeddings for each time slice of the dataset, such as monthly intervals. These embeddings are then aligned sequentially to form a continuum, allowing us to observe and analyse how the meanings and relationships of words and concepts shift over time. This approach enables the tracking of scientific trends, the emergence of new research areas, and the fading of older topics, providing a comprehensive view of the temporal evolution of knowledge within a given field. We evaluated the proposed framework on 2.3 million publications from the arXiv dataset, spanning science, mathematics, and physics. The framework generated hundreds of millions of predictions and captured relationships between scientific entities over time. We examined thousands of scientific entities, tracking them throughout our temporal

knowledge evolution framework. Given the sheer volume of our findings, we relied on large-scale, cloud-based data analytics tools for data processing, storage, and visualization. This approach allowed us to establish 25 million temporal relationships between scientific entities, pinpoint specific scientific events, comprehend the trajectories of certain scientific trends, and discern overarching patterns in the evolution of science. The results highlighted that 76.2% of the scientific events and trends studied showed low variance, indicating a high level of predictive accuracy. To conclude, our research proposes a cloud-based temporal knowledge evolution framework to analyse the large-scale corpus of scientific literature. Through our investigation of text mining, embedding techniques, and DNNs, we elucidated implicit and explicit knowledge, uncovering millions of temporal relationships, and identifying key scientific events and trends. These results highlight the critical role of neural information retrieval of large-scale data in shaping our understanding of scientific knowledge evolution and aiding informed decision-making.

Table of Contents

Acknowledgments.....	2
Publications	3
Abstract.....	4
List of Figures	11
List of Tables	13
List of Abbreviations.....	14
Chapter 1	16
1.1. Research Background	17
1.1.1. Word Embeddings	17
1.1.2. Impact of hyperparameters on downstream tasks.....	19
1.1.3. Implicit and explicit NIR.....	19
1.1.4. Big data processing	21
1.1.5. Temporal knowledge evolution	22
1.2. Research Problem	23
1.2.1. Motivation	23
1.2.2. Problem Statement	26
1.2.3. Importance of the research.....	27
1.3. Research Objectives and Questions	28
1.3.1. The objectives of the research.....	28
1.3.2. Scope and limitations.....	31
1.3.3. Research questions	32
1.4. Research Contributions.....	34
1.5. Thesis guide.....	37
2. Chapter 2.....	38
2.1. Introduction	38
2.2. Word embeddings	39
2.2.1. An Overview of word embeddings	39
2.2.2. One hot representation.....	40
2.2.3. Distributed representation	41
2.2.4. Embedding dimensionality reduction.....	43
2.2.5. Co-occurrence matrix and dimensionality reduction	44
2.3. NIR and Embeddings.....	46
2.3.1. Word embedding architecture	47
2.3.2. Continues bag of word & Skip Gram	50
2.3.3. Semantic meaning in word embeddings.....	50
2.3.4. Embedding cosine similarity.....	51
2.3.5. Type of embedding models	52
2.3.6. Applications of word embeddings.....	60
2.3.7. Word embedding limitations	62
2.3.8. Research Trends in Word Embeddings and Transformer Models.....	63
2.3.9. Bridging the gap in word embedding for specialised knowledge extraction	64

2.3.10.	Introduction to Detecting cropping patterns	66
2.3.11.	Introduction to information retrieval in underutilised crops.....	67
2.3.12.	Introduction to effective NIR in crop soil requirement identifications	68
2.4.	Hyperparameter optimisation for word embeddings	70
2.4.1.	Introduction.....	70
2.4.2.	Importance of embeddings hyperparameter optimisation	71
2.4.3.	From context to performance: decoding word2vec hyperparameters.....	71
2.4.4.	Hyperparameter Impact on word2vec Downstream Tasks.....	73
2.4.5.	Bridging the Gap in Hyperparameter Optimization for Specialized Knowledge Extraction	75
2.5.	Temporal knowledge evolution.....	77
2.5.1.	Introduction.....	77
2.5.2.	Semantic shift.....	78
2.5.3.	Dynamic word embeddings	79
2.5.4.	Temporal word embeddings.....	80
2.5.5.	Distributional semantic shifts	81
2.5.6.	Neural latent knowledge retrieval	83
2.5.7.	Bridging the Gap in Temporal Knowledge Evolution through Dynamic Word Embeddings.....	87
2.5.8.	Conclusion	88
3.	Chapter 3.....	90
3.1.	Introduction.....	90
3.2.	Datasets	93
3.2.1.	Introduction.....	93
3.2.2.	Wikipedia multilingual corpus	94
3.2.3.	BG dataset.....	95
3.2.4.	Amazon customer review	96
3.2.5.	News articles	97
3.2.6.	ArXiv dataset	98
3.2.7.	Conclusion	99
4.	Chapter 4.....	101
4.1.	Introduction.....	101
4.2.	Detecting cropping patterns of underutilised crops using online search data 102	
4.2.1.	Introduction.....	102
4.2.2.	Method.....	103
4.2.3.	Experimental design.....	105
4.2.4.	Results and analysis	106
4.2.5.	Conclusion	110
4.3.	Journey from text mining to NIR in agriculture data science.....	113
4.3.1.	Introduction.....	113
4.3.2.	Method	114
4.3.3.	Experimental design.....	116
4.3.4.	Results and discussion	120
4.3.5.	Conclusion	121
4.4.	Effective NIR methods in crop characteristic identifications	124

4.4.1.	Introduction.....	124
4.4.2.	Method.....	125
4.4.3.	Experimental design.....	129
4.4.4.	Results and discussion.....	132
4.4.5.	Conclusion.....	136
4.5.	Chapter Conclusion	139
5.	Chapter 5.....	141
5.1.	Introduction.....	141
5.2.	Method.....	142
5.2.1.	Word embedding hyperparameters.....	142
5.2.2.	Document classifiers.....	143
5.2.3.	The evaluation methods.....	146
5.3.	Experimental design.....	146
5.4.	Results	148
5.4.1.	Comparative statistical analysis of F1 scores.....	148
5.4.2.	Visual performance evaluation of DNN architectures.....	152
5.4.3.	Impact of different word2vec hyperparameters on DNNs.....	154
5.4.4.	Word2vec epoch hyperparameter.....	154
5.4.5.	Word2vec window size hyperparameter.....	155
5.4.6.	Word2vec negative sampling hyperparameter.....	156
5.4.7.	Word2vec minimum count hyperparameter.....	157
5.4.8.	Word2vec hierarchical softmax hyperparameter.....	158
5.4.9.	Word2vec Skip-Gram vs Bag of Words architecture.....	159
5.4.10.	Word2vec dimension size architecture.....	160
5.5.	Conclusion.....	161
6.	Chapter 6.....	168
6.1.	Introduction.....	168
6.2.	Method.....	169
6.2.1.	Data pipeline and pre-processing steps	169
6.2.2.	Systematic categorization of scholarly publications in scientific literature.....	172
6.2.3.	Vector space continuum construction	174
6.2.4.	Constructing temporal sequences of preselected entities: a synthesis of chronological data	180
6.3.	Experimental Design	185
6.3.1.	arXiv DataStream: From PDFs to vector space analysis.....	185
6.3.2.	arXiv DataStream AWS solution architecture	188
6.4.	Results	191
6.4.1.	Sequence analysis of temporal entity relationships.....	191
6.4.2.	Capturing disruptive events and trends.....	196
6.4.3.	Analysis of Temporal Knowledge Evolution Performance.....	204
6.5.	Discussion	208
6.6.	Conclusion.....	209
7.	Chapter 7.....	213

8.	<i>References.....</i>	<i>222</i>
-----------	-------------------------------	-------------------

List of Figures

Figure 1 Neural Network representation of word2vec.....	47
Figure 2 Implementation of word2vec for multiple input vectors.....	49
Figure 3 Word2vec CBOW vs Skip-Ngram.....	50
Figure 4 Word2vec window size	73
Figure 5 Illustrations of the interaction between different datasets, techniques and experiments presented in this research	92
Figure 6: Architecture diagram of how to use text mining and large data to extract cropping patterns.	106
Figure 7: Global Google search queries for "BG."	107
Figure 8: Leading countries in BG Google searches.	109
Figure 9: Primary countries for BG Google searches, categorised by desktop and mobile devices.....	110
3. <i>Figure 10 Text Mining process for ecological and taxonomic data.</i>	115
<i>Figure 11 Process of data preparation and model training</i>	116
Figure 12: Text mining and NIR to extract insights about crop attributes.	119
Figure 13 Graph to the left depicts the relationship between countries and Durian. The bar chart to the right illustrates the four crops with their relationship to ten selected countries.	121
Figure 14: Text mining and NIR of 178 languages of Wikipedia to extract insights about crop attributes.	132
Figure 15: Comparison of Cosine Similarity for Soil Classification with a Selection of Ten Crops.	134
Figure 16 On the left is the Simple RNN, in the center is the CNN, and on the right is the LSTM. The upper illustrations depict the model summaries, while the lower illustrations represent the model architectures.	145
Figure 17: The process of the creation of word embedding and hyperparameters selection process and DNN architectures testing.	148
Figure 18: Graphical depiction of F1 score metrics in candlestick visualisation for RNN, CNN and LSTM	151
Figure 19: Variations in F1 scores for RNN, CNN, and LSTM with diverse word2vec hyperparameter settings.	154
Figure 20: Impact of word2vec epoch parameter on DNN architectures.	155
Figure 21: Impact of word2vec window size parameter on DNN architectures.	156
Figure 22: Impact of word2vec negative sampling parameter on DNN architectures.....	157
Figure 23: Impact of word2vec min count parameter on DNN architectures.	158
Figure 24: Impact of word2vec hierarchical softmax parameter on DNN architectures.....	159
Figure 25: Impact of word2vec Skip-Gram or Bag-of-Words parameter on DNN architectures.	160
Figure 26: Impact of word2vec dimension size parameter on DNN architectures.	161
Figure 27 Data pipeline and pre-processing steps	171
Figure 28: Discrete Knowledge Evolution Construction Method.....	176
Figure 29: Continuous Knowledge Evolution Construction Method.....	177
Figure 30: Workflow of the temporal knowledge evolution framework, which is segmented into three stages: text pre-processing, document classification, and temporal knowledge progression.	188

Figure 31: Solution architecture of the temporal knowledge evolution framework on AWS Cloud.....	190
Figure 32: EternalElevation trend function examples	193
Figure 33: PhantomFade trend function examples	194
Figure 34: PhoenixRise trend function examples	194
Figure 35: PhoenixFall trend function examples	195
Figure 36: TimelessTide trend function examples	196
Figure 37: Time-series correlation between the entity “bert” and the entities “transformer”, “language”, “image”, “cancer” and “bidirectional”	199
Figure 38: Time-series correlation between the entity “higgs” and the entities “atlas”, “collision” and “quantum”	200
Figure 39: Time-series correlation between the entity “cnn” and the entities “word”, “document” and “image”	201
Figure 40: Time-series correlation between the entity “hinton” and the entities “dnn”, “neural”, “dropout” and “acoustic”	202
Figure 41: Time-series correlation between the entity “translate” and the entities “crystal”, “transformer” and “rnn”	203
Figure 42: Time-series correlation between the entity “linguistics” and the entities “boltzmann”, “research”, “semantic” and “dnn”	204
Figure 43 Temporal Knowledge Evolution Event Prediction Performance	207

List of Tables

Table 1: BG selected names.....	96
Table 2 Example of the global knowledge database	120
Table 3: Predicted Values Compared to FAO ECOCROP Data. Bolded values highlight the top predictions that align with the actual FAO ECOCROP figures.....	136
Table 4: List of word2vec hyperparameters and their range used in the experiment.	147
Table 5: F1 score value (min, max, average, and median) for RNN, CNN, and LSTM.	149
Table 6: Example of how the vector database that store entities and its relationship to each other.....	184
Table 7 Example of data collections for the relationships among entities	205

List of Abbreviations

AI	Artificial Intelligence
Amazon EC2	Amazon Elastic Compute Cloud
Amazon ECS	Elastic Container Service
Amazon S3	Amazon Simple Storage Service
ANN	Artificial Neural Network
API	Application Programming Interface
AWS	Amazon Web Services
BERT	Bidirectional Encoder Representations from Transformers
BG	Bambara groundnut
BOW	Bag of Words
CBOW	Continuous Bag of Words
CFF	Crops For the Future
cite2vec	Citation to Vector
CNN	Convolutional Neural Networks
con2vec	Contrast set to Vector
CPU	Central Processing Unit
csv	comma-separated values
distilBERT	Distilled BERT
DNN	Deep Neural Networks
Doc2vec	Document to Vector
DWE	Dynamic Word Embeddings
ELMo	Embeddings from Language Models
FAO	The Food and Agriculture Organization
GB	Gigabyte
GloVe	Global Vectors for Word Representation
GPU	Graphics processing unit
Graph2Vec	Graph to Vector
hyperdoc2vec	Hyper Document to Vector
iACVP	identification of anti-coronavirus peptides
IDF	Inverse Document Frequency
IMDB	Internet Movie Database
IQR	InterQuartile Range

IR	Information Retrieval
IR-word2vec	Information Retrieval word2vec
LDA	Latent Dirichlet Allocation
lda2vec	LDA (Latent Dirichlet Allocation) and word2vec
LSTM	Long Short-Term Memory
ML	Machine Learning
MPNet	Masked and Permuted Pre-training for Language Understanding
NB	Naïve Bayes
NB-SVM	Naïve Bayes (NB) and Support Vector Machine (SVM)
NIR	Neural Information Retrieval
NLP	Natural Language Processing
node2vec	Node to Vector
PCA	Principal Component Analysis
RAM	Random Access Memory
RNN	Recurrent Neural Networks
RNN-LM	Recurrent Neural Network Language Model
RoBERTa	Robustly Optimized BERT Pretraining Approach
RQ	Research Question
Seq2seq	Sequence to Sequence
SVM	Support Vector Machine
t-SNE	t-Distributed Stochastic Neighbor Embedding
T5	Text-to-Text Transfer Transformer
TF	Term Frequency
TF-IDF	Term Frequency - Inverse Document Frequency
ViBERT	Vision-and-Language BERT
word2vec	Word to Vector

Chapter 1

Introduction

The domain of Natural Language Processing ([NLP](#)) is characterised by ongoing innovations that continually shape the modern computational comprehension of linguistic structures. Among these innovations, word embeddings constitute a captivating advancement in the manner through which words can be represented mathematically, and their interrelationships delineated. This research delves profoundly into the temporal intricacies inherent in word embeddings, offering valuable insights into the mechanics underpinning the representation of words as vectors over time. The purpose of the temporal elements of words or entities, allows the researchers to understand the trends and events associated with entities under study. The research further explores the ramifications posed by hyperparameters, the intricate interplay between explicit and implicit Neural Information Retrieval ([NIR](#)), as well as the multifaceted challenges encountered during the processing of extensive datasets, along with their attendant solutions. [NIR](#) refers to the application of neural network models and deep learning techniques to the task of information retrieval. Information retrieval involves searching for and retrieving relevant information from a large dataset or database in response to a user query. Traditional information retrieval methods often rely on keyword matching and hand-crafted features. In contrast, [NIR](#) leverages the power of neural networks to automatically learn representations and relationships within the data, enabling more accurate and efficient retrieval of relevant information. Studying [NIR](#) is essential due to its ability to enhance search accuracy through semantic understanding, manage complex queries effectively, and scale efficiently to large datasets. [NIR](#) systems also personalize user experiences, contribute to advancements in AI, and improve overall user satisfaction, leading to economic benefits through enhanced decision-making and productivity. Understanding the nature of information is crucial in fields like [NIR](#). Differentiating between explicit and implicit knowledge is key to processing and using data effectively, especially when predicting scientific events and trends. Explicit knowledge is clearly articulated, documented, and easy to understand, often found in written manuals and instructions. In contrast, implicit knowledge is inferred from context and experience,

even when not directly stated. It is derived from understanding broader contexts or synthesizing information from multiple sources. Recent advancements in [NIR](#) and knowledge extraction have incorporated both explicit and implicit knowledge, as well as explored the temporal evolution of knowledge entities and relationships. The Attention Knowledge Network Combining Explicit and Implicit Information (AKNEI) by Deng et al. ([2023](#)) effectively captures and describes correlations between entities in a knowledge graph, leveraging explicit relationships through knowledge graph embeddings and implicit interactions via cross-feature fusion modules. This model showed a 10% improvement in performance on datasets like movies and music, demonstrating its ability to handle both explicit and implicit knowledge. Ding et al. ([2024](#)) addressed knowledge extraction in the form of complex question answering, focusing on extracting relevant subgraphs from knowledge graphs by identifying structural dependencies, which represent implicit relationships between facts. This approach improved subgraph extraction accuracy by 15%, showing how implicit knowledge can enhance the retrieval process. These examples collectively illustrate how current research is advancing the field by integrating explicit and implicit knowledge, addressing temporal aspects of knowledge evolution, and improving the understanding and processing of complex information relationships.

This research investigates the effectiveness of applying [NIR](#) technology and various embedding techniques to extract both implicit and explicit knowledge from scientific literature. The study is partially part of a collaborative effort between the Crops For the Future ([CFF](#)) research centre and the University of Nottingham Malaysia Campus. The initial experiments concentrated on underutilised crops and the extraction of related knowledge. Finally, the research examines the practical application of word embeddings and the creation of vector spaces that span two decades of scholarly literature, thereby conducting an analysis of the evolution of temporal knowledge within the scientific literature.

1.1. Research Background

1.1.1. Word Embeddings

Word Embedding is a method to represent words as compact vectors in mathematical dimensions, unlike one-hot encoding which uses binary vectors. This enables calculations and comparison of words in mathematical terms, aiding the analysis of

linguistic relationships. Neural networks process text more effectively with these vectors, allowing for quantification and analysis of linguistic features. [Word2vec](#), introduced by Mikolov et al. ([2013a](#)), enhanced [NLP](#) by learning word embeddings that capture word relationships. It doesn't need manual annotation, relying on self-supervised learning from existing words.

[Word2vec](#) is a neural network algorithm that predicts words based on their context. It represents words as mathematical vectors within its hidden layers. The architecture involves input, output, and hidden layers. The neural network's weights, initially random, are refined using sentences to capture word context. The model's achievement lies in converting words into vectors, allowing exploration of semantic relationships. Input neurons represent words via a one-hot encoding. Output neurons correspond to surrounding words. The hidden layer captures word relationships, representing them in a vector space. Training involves refining weights to align predicted output vectors with actual ones. The hidden layer employs the distributional hypothesis, capturing word meanings through vector values. It has hundreds of dimensions. The weight matrices within the hidden layers shape the network's performance.

[Word2vec](#)'s design involves using a training window. For instance, in the Continuous Bag of Words ([CBOW](#)) model, in the sentence "The brown fox jumps on the lazy dog," the input vectors are $V[brown]$, $V[fox]$, $V[on]$, and $V[the]$, predicting the output $V[jumps]$. Word embedding encompasses a distinct and intricate challenge concerning semantic relationships. Similar words, such as "car" and "vehicle," are synonymous, representing the concept of interchangeability. On the other hand, words like "car" and "road" exhibit contextual relationships, implying interrelatedness. Both types of relationships are crucial when dealing with [NIR](#).

One of the functions that has consistently been used with the [word2vec](#) model is the similarity function which measures the similarity between word pairs. Cosine similarity calculates the cosine value of the angle separating two vectors, derived from their dot product. Its value can span from -1 to 1: a value of 1 signifies that the vectors align perfectly, 0 indicates they're at right angles, and -1 shows they're in opposing directions. The [word2vec](#) model's complex neural network architecture effectively translates words into vectors, capturing nuanced meanings through hidden layers and iterative training. However, the full extent of [word2vec](#)'s capabilities shine through in downstream tasks, where the careful adjustment of hyperparameters becomes a

pivotal factor (Rong, [2014](#)). Fine-tuning essential hyperparameters such as dimension and window size becomes indispensable, as these choices wield significant influence over the model's performance across diverse language-related tasks. By striking a delicate balance between intricacy and context and optimising other parameters like learning rate and iterations for efficient training, the model's prowess in [NLP](#) is profoundly elevated, making a substantial impact on its real-world applications.

1.1.2. Impact of hyperparameters on downstream tasks

Hyperparameter tuning holds immense significance for embedding models like [word2vec](#). These settings, chosen before training, strongly impact model performance and subsequently influence the downstream tasks for which the embeddings might be used. Proper tuning optimises the model's effectiveness across [NLP](#) tasks. Take dimension size, a crucial parameter determining the size of word vectors, for instance. It affects model complexity; being too small loses context while being too large risks overfitting. Tuning strikes the right balance. The window size is also pivotal, as it influences the context range. Smaller sizes capture immediate context and focus on interchangeability, while larger sizes broaden associations and strengthen interrelatedness. Tuning tailors the model to the specifics of the data. Learning rate, training iterations, subsampling threshold, and negative sampling parameters matter. They impact optimization, convergence, word selection, and training efficiency. Thoughtful tuning ensures effective learning. In essence, hyperparameter tuning adapts the model to data intricacies and tasks, optimising its performance. This process elevates a standard model to one excelling in [NLP](#) tasks.

1.1.3. Implicit and explicit NIR

Understanding the nature of information and knowledge is essential in various domains, including [NIR](#). The distinction between explicit and implicit information plays a crucial role in how we process and utilize data. This understanding becomes even more critical when predicting scientific events, trends, or trajectories. Explicit information is deliberately encoded for human comprehension in literature, clearly stated and easy to understand. Implicit knowledge, on the other hand, is inferred from the broader context, even if it is not directly stated by the author. This can be extrapolated from individual texts or larger bodies of work. Predicting scientific events, trends, or trajectories involves forecasting future developments in scientific research

and innovations. It means identifying potential future directions and outcomes based on current data and trends. This prediction is crucial as it helps guide research priorities, allocate resources efficiently, and anticipate future challenges and opportunities. By understanding both explicit and implicit information, researchers can better predict and navigate the evolving landscape of science and technology.

State-of-the-art methods for extracting both explicit and implicit knowledge from extensive scientific literature leverage advanced techniques across various domains. [NLP](#) techniques such as Named Entity Recognition and relation extraction are fundamental for identifying explicitly stated entities and relationships. Deep learning models, including Bidirectional Long Short-Term Memory ([LSTM](#)) and transformers, excel in capturing implicit knowledge by analysing the context surrounding the text. Word embeddings like [word2vec](#), Global Vectors for Word Representation ([GloVe](#)), and Bidirectional Encoder Representations from Transformers ([BERT](#)) are instrumental in creating vector representations of words, which help infer implicit knowledge based on semantic similarities and contextual nuances. Knowledge graphs, specifically Knowledge Graph Completion, are employed to infer unstated relationships and entities, enriching the structured knowledge base. Shardlow et al., ([2018](#)) introduced GENIA-MK annotation scheme categorizes knowledge into dimensions such as Observation, Investigation, and Analysis, facilitating a structured extraction of both explicit and implicit information. Hybrid approaches that combine rule-based and machine learning methods further enhance the accuracy of knowledge extraction, such as supervised learning systems designed to recognize new hypotheses and novel findings from the text.

Several challenges persist with these state-of-the-art methods. The complexity of implicit knowledge, which often requires nuanced understanding and inferencing, poses a significant challenge for automated systems. Scalability issues arise when processing vast amounts of scientific literature efficiently while maintaining high accuracy. Data annotation, essential for training models, is labour-intensive and demands domain-specific expertise. The dynamic nature of scientific knowledge requires continuous updates and retraining of models to keep up with new information. Additionally, interdisciplinary variations in terminology and writing styles complicate the development of universal models for knowledge extraction. For example, Rossanez et al., ([2020](#)) proposed a method named KGen, A Knowledge Graph Generator from Biomedical Scientific Literature. KGen discusses techniques for

building knowledge graphs from unstructured texts using [NLP](#) tools and machine learning models (BMC Medical Informatics and Decision Making). They also researched the challenges and advances in information extraction from scientific literature. Luo et al., ([2023](#)) introduced Unified AI Drug Discovery with Multiple Knowledge Modalities that explores leveraging structured and unstructured knowledge in AI-driven drug discovery, addressing challenges of missing modalities through sparse attention and modality masking. Nastase et al., ([2019](#)) highlighted in their paper Extracting Structured Knowledge from Scientific Publications, the use of deep learning models for polarity identification and the extraction of structured information from scientific texts.

1.1.4. Big data processing

This study leverages a dataset from ArXiv, encompassing 2.3 million scientific articles that have been authored over a span of two decades of scientific literature. In preparing this substantial volume of data, amounting to 1.7TB, a methodical sequence of operations was undertaken. These steps include the uncompressing of files, text conversion, data concatenation, text purification, stop words removal and classification. The execution of these intricate procedures demanded significant computational resources. Consequently, the data were transformed into an accessible format primed for rigorous analysis, with disaggregated files organised monthly from the years 2000 to 2020. The utilisation of advanced analytics techniques, integrated with Amazon Web Services ([AWS](#)), served to expedite the processing of the ArXiv dataset, thereby effectuating a substantial reduction in the required processing time. The employment of [AWS](#)'s cloud infrastructure was instrumental to this study, facilitating cost-effective and proficient computational resources. A diverse array of [AWS](#) tools, including Amazon Athena, AWS GLUE, Amazon Simple Storage Service ([S3](#)) Bucket, Amazon Elastic Computer Cloud ([EC2](#)) and Amazon SageMaker, was strategically deployed across various datasets and experimental designs. The methods employed in data preparation were discovered to be both cost-effective and timesaving, providing support for various aspects of the research, particularly concerning the quality of the embeddings.

1.1.5. Temporal knowledge evolution

The domain of scientific inquiry is in a state of constant flux, marked by routine innovations and technological progressions. In order to remain abreast of such dynamic alterations, it becomes imperative to analyse the evolution of knowledge within the scientific literature. In this research, we elucidate a study focused on the understanding of knowledge evolution, employing word embeddings and temporal analysis methodologies. This research necessitated the extraction of data from the scientific literature in the arXiv repository, spanning the period from 2000 to 2020. Utilising sophisticated data processing mechanisms, we managed the extensive textual datasets and subsequently employed word embeddings to construct a vector space continuum in which each month's literature is encapsulated in a distinct vector space. These vector spaces collectively compose the vector space continuum. The application of time series analysis enabled the examination of monthly trends and evolutions within these vector spaces over time. By strategically implementing these techniques and conducting a comprehensive examination of the monthly time series associated with these vector spaces, we were able to identify crucial events and derive insights into the evolution of knowledge within the field. This encompassed detecting the emergence and development of novel research subjects, as well as discerning the impact of technological breakthroughs and discoveries across the temporal spectrum. The practical applications of our study hold relevance for researchers who are pursuing an enhanced comprehension of knowledge evolution within their specialised fields. Moreover, the presented framework provides a quantified substantiation that might support researchers' intuitive insights, thereby leading to potential efficiencies in decision making.

The research did not identify any existing state-of-the-art studies addressing the temporal evolution of knowledge in the same manner or on the scale that this study does. This presents a challenge in terms of comparing the findings to existing state-of-the-art methodologies. Consequently, the focus was on the pre-identification of specific events and trends that were observed and quantified, corroborated by existing literature. These identified topics and events were tracked over time, and their characteristics were compared with those documented in existing publications that study similar phenomena or discoveries.

1.2. Research Problem

1.2.1. Motivation

In a research centre focused on the study of underutilised crops, a team of scientists has engaged in the complex task of data collection from various publications, where information pertaining to these crops may be found both implicitly and explicitly. [CFF](#) research centre in collaboration with the University of Nottingham collaborated to initiate knowledge extraction for underutilised crops. This research was partially focused on addressing the key elements of implicit and explicit knowledge extraction for those crops. Given the nature of underutilised crops as a domain characterised by limited structural datasets and dispersed data within the literature, this area serves as an excellent case study for employing [NIR](#) techniques to extract both implicit and explicit knowledge, despite the scarcity of such information in existing publications. Unlike major crops, for which data is readily accessible via international databases with dedicated institutions, the information concerning underutilised crops is scarce. The quantity of data points collected by scientists on a daily basis is restricted by their ability to sift through literature and uncover data, whether explicitly or implicitly stated. Moreover, researchers frequently make pivotal choices to steer their research towards specific crops or methodologies. These decisions can hinge on their intuition and a limited set of data points that align with their intuitions. However, even though a researcher's intuition is shaped by years of experience in the field, disparities among researchers may lead to the necessity for quantification derived from extensive datasets. Such datasets are often challenging for humans to process on a large scale or require more time than what is typically available during the decision-making process. Analysing the knowledge evolution over an extended duration is essential for comprehending the trajectory of the scientific community as reflected in the literature. Grasping this trajectory and the capacity to mathematically measure it enables informed decision-making that considers evidence, intuition, and quantified trends or irregularities in the evolution of knowledge. This, in turn, lends support to the chosen research direction.

To address this challenge, several approaches were proposed to extract the required information. Initially, big data analytics of user traffic on Google's search engine were employed to gauge global interest in underutilised crops. This led to significant discoveries of information that would otherwise have been time-consuming

to collect manually. This initial success, however, uncovered a further challenge where users use crops' local names for their search. Although databases containing local names do exist for major crops, underutilised crops often present limited data, with information sparsely distributed across various publications.

The limitations of text mining using Google's search engine are significant. Firstly, the data quality and coverage are inconsistent, with web content often lacking validation. The engine struggles with local names and synonyms, particularly for underutilised crops, leading to incomplete data retrieval. Its algorithms frequently fail to understand the context of complex scientific queries, resulting in irrelevant results. Moreover, Google does not always prioritize the most recent or pertinent publications, essential for research in fast-evolving fields. While Google Search is advanced for general use, it is not state-of-the-art for specialized scientific text mining. State-of-the-art methods utilize more sophisticated tools and techniques, such as domain-specific databases, advanced text mining algorithms like word embeddings and deep learning models, and customizable analyses for fine-tuning hyperparameters. Arora et al. ([2019](#)) highlighted one significant limitation is the anonymity and demographic representation of the data. Because the data is anonymized, it may not include searches from vulnerable groups such as the elderly or individuals in regions with low internet access, leading to representation bias. Additionally, selecting the correct search terms for topic can be challenging, potentially resulting in inaccurate data interpretation. There is also often a lack of clear conceptual models linking search terms to actual representation, making it difficult to draw definitive conclusions. Lastly, there is a tendency for studies to report positive correlations more frequently, which can skew the overall understanding of search behaviours.

To tackle these limitations, the research turned to advanced techniques for the retrieval and mining of large-scale data, focusing on [NIR](#) methods. This approach utilised sophisticated models, such as word embeddings and neural networks, to enhance the precision and comprehensiveness of data extraction. By employing these advanced techniques, this research was able to identify and retrieve valuable information about underutilised crops from extensive and diverse datasets, addressing the issues of inconsistent and sparse data. The use of underutilised crops as a case study effectively demonstrated the potential of these methods to extract both implicit and explicit knowledge from large-scale datasets. This strategy not only increased the efficiency of data collection but also uncovered previously overlooked connections and

trends, supporting more informed and evidence-based decision-making in agricultural research. The success of this [NIR](#) and mining framework highlights its broader applicability, enabling the study of knowledge evolution and the detection of emerging trends in various scientific fields.

In response, a solution utilising word embeddings was proposed to identify potential local names for each underutilised crop, with the results then forwarded to scientists for curation. The preliminary outcomes were promising, prompting the progression to a second phase focused on extracting characteristics of the crops, such as crop soil requirements. An experiment was designed and implemented to determine these requirements specifically for underutilised crops, resulting in commendable accuracy. The findings were subsequently used to populate a knowledge base, allowing scientists to curate a comprehensive table of crop characteristics. Given the ground-breaking results obtained, and with a careful experiment designed to understand the hyperparameters tuning of embeddings impact on downstream tasks, the research progressed to a final stage, concentrating on the study of knowledge evolution over extended periods. This approach facilitated the identification of technology trends, scientific discoveries, and the detection of anomalies in the literature. The experiment was designed utilising a large corpus of data over a long-time frame, expanding the scope from crop science to include disciplines such as science, mathematics, and physics. The resulting framework offers broad applicability to the wider scientific community, enabling the study of the evolution of knowledge, topics, technologies, methodologies, and event detection. This not only quantifies intuitive assessments but also significantly reduces the time typically expended on manual information extraction or evidence quantification, thereby contributing substantially to the ongoing research in the field.

To ensure the accuracy and validity of our results, This study implemented several verification steps. this study involved analysing a substantial number of words from agronomic datasets, from which we extracted attributes and characteristics for a wide variety of crops, including numerous local names and synonyms. We validated this extracted data by comparing it with an established agronomic database, achieving a high accuracy rate confirmed through a sample subset. This accuracy rate demonstrates the reliability of our text mining methods. Nonetheless, it is important to recognize that analysing this number of words is considerably less complex than

processing the vast amount of data in larger corpora, such as the English Wikipedia. Research on underutilised crops is relatively sparse, primarily due to the predominant focus of international organizations on major crops. The application of methods for extracting both implicit and explicit knowledge to underutilised crops appears to be largely unexplored in existing literature. Consequently, conducting comparisons to the current state of the art is challenging due to the absence of established benchmarks in this specific area of study.

1.2.2. Problem Statement

This research focuses on two crucial areas: the temporal evolution of knowledge through a vector space continuum and the retrieval of implicit and explicit information from academic and scientific corpora. While these methodologies exhibit considerable potential, they concurrently pose challenges, particularly concerning the establishment of systematic procedures for the extraction of explicit and implicit knowledge, as well as the identification of patterns inherent in the evolution of temporal knowledge. Although several scholars have undertaken investigations into semantic shifts and acknowledged the temporal dimension within the realm of knowledge, the systematic exploration of pattern identification for entities engaged in the temporal evolution of knowledge over an extended time interval and within an extensive compilation of scientific literature remains a scarcely explored area. The fluidity of knowledge adds an additional layer of complexity. Being an ever-changing construct, knowledge reflects alterations in comprehension, conventions, and linguistic expressions, requiring flexible strategies to ensure that the retrieval of information remains relevant and accurate. An extant challenge pertains to the absence of a comprehensive end-to-end methodology tailored for the construction of temporal vector spaces, intended to encapsulate the evolutionary trajectory of temporal knowledge, while concurrently addressing the discernment of patterns characterising temporal relationships between entities. Semantic shift is the evolution of word meanings over time, shaped by language, culture, and society, resulting in changed connotations and usages. While the attainment of a semantic shift representation is within reach, these research findings reveal an uncharted area in the detection of temporal knowledge evolution among entities. This gap also encompasses the enablement of identifying underlying

scientific trends and the recognition of temporal anomalies, all of which persist as unresolved complexities in current scientific literature.

While the methodologies for analysing the temporal evolution of knowledge and information retrieval are promising, the need for specificity in addressing large-scale text data challenges is crucial. This research zeroes in on four primary issues: the retrieval of knowledge concepts from extensive textual datasets, the mining of relationships between these concepts, the optimization of hyper-parameters in [NIR](#) systems, and the temporal evolution of knowledge. Although previous studies have explored these areas, they often exhibit significant limitations. Traditional methods for retrieving knowledge concepts from large datasets struggle with scalability and precision. Techniques for mining relationships between concepts have made progress using machine learning and natural language processing, yet they frequently lack the ability to effectively handle the dynamic and evolving nature of knowledge. This study seeks to bridge these gaps by developing a comprehensive methodology that not only retrieves and mines knowledge concepts and their relationships but also optimizes neural retrieval systems within the context of temporal knowledge evolution. By doing so, it aims to overcome the deficiencies of earlier research efforts and provide a more accurate and dynamic understanding of scientific and academic texts.

1.2.3. Importance of the research

Through exploring these interconnected challenges, this research endeavours to demystify the complexities inherent in temporal knowledge evolution and [NIR](#). The objective of the research is to establish efficient methodologies to retrieve implicit and explicit knowledge and to develop an efficient methodology to map knowledge evolution patterns to equip scholars to predict novel technologies, identify groundbreaking approaches, and pinpoint new data that may redirect and expand scholarly paths and possibilities. The specialised deployment of embeddings to trace temporal knowledge evolution highlights an area with significant potential for growth, laying a foundation for further inquiries in this domain.

In summary, this research serves as an important addition to the literature, connecting existing understanding and illuminating new directions in the exploration and applications of knowledge mining and temporal knowledge evolution. This will subsequently serve as a catalyst to accelerate the processes of knowledge retrieval, temporal event detection, and the discovery of knowledge evolution. When utilised

efficiently, it will lead to a reduction in both time and resources required by researchers for daily tasks and facilitate the quantification of their intuition in the areas of temporal event detection and knowledge evolution.

1.3. Research Objectives and Questions

1.3.1. The objectives of the research

In the vast landscape of Big Data, gleaning meaningful insights from extensive datasets, especially within the realm of scientific research, poses a significant challenge. The primary objective of this thesis is to leverage the capabilities of word embeddings to unearth both latent and temporal dimensions of knowledge within a large scientific corpus. Through this research, we begin with a focus on both implicit and explicit Information Retrieval ([IR](#)) to identify the characteristics and attributes of underutilised crops. Building upon this foundational understanding, we then transition to tracing the holistic trends and evolution of knowledge, pinpointing key scientific events, and examining overarching scientific trajectories. Using the comprehensive arXiv scientific literature dataset as our foundation enriches our exploration. As we delve into the intricate processes of data transformation and assess the efficacy of [NIR](#), and as we construct a temporal knowledge evolution vector space continuum, this thesis endeavours to transform vast volumes of data into understandable and actionable insights.

The initial focus of this research was on crop science, specifically involving the extraction of information related to local and multilingual crop names, as well as the identification of crop characteristics such as soil requirements. As we began constructing the framework for the evolution of knowledge, it became evident that a substantial amount of data was necessary. Upon obtaining a significant dataset from arXiv, the opportunity to expand the research beyond crop science arose. This expansion encompassed various fields including mathematics, physics, and broader scientific knowledge present in the analysed data. The preliminary investigations into knowledge evolution across a more extensive body of text highlighted the framework's potential to capture a broader context of scientific backgrounds. This influenced the decision to continue with this approach.

This research examines the methods of [IR](#) from a voluminous corpus of data. This process is executed by explicitly employing word embeddings as a reservoir of latent knowledge and by investigating the interchangeability and interrelatedness of various scientific entities. For instance, the study introduces an innovative methodology to detect multi-lingual local names of underutilised crops and link them with English common and scientific names. The methodology for extracting multi-lingual local names is a direct application of the interchangeability analogy of word embeddings, facilitating automatic data extraction from literature. Additionally, the research offers a ground-breaking approach to determine crop-specific characteristics such as soil requirements for underutilised crops using word embeddings and the interrelatedness analogy.

The [NIR](#) methods necessitate careful design of word embedding vector spaces, appropriately encoding latent knowledge such as crop names and soil requirements. Therefore, this research provides an in-depth analysis of the hyperparameters of word embedding models. Despite the common practice of selecting hyperparameters based on prior recommendations for specific use cases, the research demonstrates that this space can yield substantial performance variability for different downstream tasks such as [NIR](#) and document classification. Consequently, the study emphasises the significance of this area and introduces specialised techniques and hyperparameter combinations to address tasks such as interchangeability, relatedness, classification, and knowledge evolution.

The core objective of this research is to analyse and track the trajectory of temporal knowledge evolution, with an emphasis on exploring the arXiv scientific literature dataset. The study involves processing this dataset through intricate data pipeline and transformation stages, ultimately preparing 2.3 million articles for the construction of high-quality word embeddings that will facilitate downstream tasks like text classification and temporal knowledge embeddings. The task of processing 2.3 million scientific articles from arXiv requires an advanced Big Data processing pipeline. This study outlines an efficient method of data transformations utilising public cloud infrastructure, effectively converting 1.7TB of data into a format ready for the construction of word embeddings and [DNN](#) classifiers. Analysing the evolution of temporal knowledge over time necessitates the precise construction of a vector space continuum. Upon introducing a processing methodology for expansive arXiv datasets, the aim is to arrange scientific publications chronologically, thereby establishing a

representation of temporal vector spaces. This representation is carefully crafted to maintain chronological integrity across a two-decade span of scientific literature. Finally, we leverage data analytics and visualisation techniques, utilising public cloud infrastructure, to depict the trajectory of knowledge evolution, identify anomalies in scientific research, and scrutinise emerging scientific trends. The primary objectives of the research are as follows:

Core objectives:

1. Investigate the potential of [NIR](#) for deterministic data extraction. [NIR](#) uses neural network models to improve the retrieval of relevant information from large datasets, while deterministic data extraction relies on well-defined algorithms for consistent and predictable data retrieval. This objective aims to tackle the challenge of extracting both implicit and explicit knowledge, exemplified by the study of underutilised crops.
2. Examine the hyperparameters of word embedding models to enhance [IR](#) accuracy and construct the temporal vector space continuum. Achieving this objective requires a comprehensive understanding of how changes in hyperparameters affect the models used in the experiments.
3. Utilize word embeddings to reveal latent and temporal dimensions of knowledge within a vast scientific corpus by creating a vector space continuum that represents the evolution of knowledge over the past two decades. This objective addresses the challenge of comprehending knowledge progression and provides the capability to detect scientific events and understand their trends.

Supporting objectives:

4. Identify local terminologies for underutilised crops in various languages and develop methods for automatically detecting multilingual local names of crops and link these names with their English counterparts and determine crop-specific characteristics using word embeddings and interrelatedness analogies.
5. Trace the evolution of knowledge within scientific disciplines by investigating the trajectory of temporal knowledge evolution using the arXiv scientific literature dataset.
6. Examine techniques for processing a large corpus of scientific literature and training large models and utilize data analytics and visualization techniques to depict the trajectory of knowledge evolution and emerging trends.

1.3.2. Scope and limitations

This research seeks to thoroughly investigate a spectrum of relevant themes encompassing neural implicit and explicit [IR](#), hyperparameter optimization, and the temporal evolution of knowledge via the establishment of a vector space continuum. The research endeavour entails the execution of a series of empirical inquiries carefully structured to illuminate diverse downstream tasks using word embeddings. The distinct spheres under careful examination include:

- Optimising hyperparameters to facilitate automated retrieval of local crop names interchangeable with common English and scientific names.
- Designing and executing a robust [NIR](#) methodology for multilingual crop names.
- Identifying optimal hyperparameters for the relatedness analogy to enable automated retrieval of crop requirements (e.g., soil type) linked to common English and scientific names.
- Leveraging big data and text mining techniques to detect cropping patterns of underutilised crops.
- Developing a methodology for transforming big data in the context of arXiv's 2.3 million articles using public cloud infrastructure.
- Exploring the robustness of [DNN](#) against variations in word embedding hyperparameters to establish an optimal knowledge classifier.
- Creating a [DNN](#)-based knowledge classification model for investigating temporal knowledge evolution.
- Devising a comprehensive approach for studying temporal knowledge evolution through a vector space continuum.
- Establishing a large-scale data analytics platform integrated with a visualisation dashboard to illuminate temporal knowledge evolution, aiding in the identification and recognition of temporal events in the scientific literature.

The research scope outlined above encompasses intricate elements like hyperparameter optimization, multilingual crop name retrieval, big data analytics, and the evolution of temporal knowledge. However, it's vital to understand the specific boundaries shaping this study, resulting from conscious choices to focus on certain areas for depth and precision. The subsequent discourse will highlight these boundaries to provide a clear context for interpreting the methods and findings shared.

The research has language constraints, addressing only a limited set of languages when discussing multilingual crop names. Its exploration of [IR](#) and cropping patterns could be applied to scientific domains other than crop science. There's a restriction in the text mining approach for detecting cropping patterns of underutilised crops, which doesn't span to other data formats. The study's emphasis on hyperparameters selection for interchangeability and interrelatedness analogies potentially overlooks other analogy forms. The knowledge classification model, rooted in [DNN](#), is designed predominantly for examining temporal knowledge evolution, and its relevance to other domains or knowledge types may remain unprobed. Even as the study offers an in-depth look at temporal knowledge evolution, it is not possible to holistically cover all scientific domains and discoveries.

In conclusion, this research constitutes a robust and targeted exploration into the multifaceted domains of hyperparameter optimization, multilingual [IR](#), big data analytics, and temporal knowledge evolution. By delineating a specific scope and limitations, the study has strategically honed in on particular areas allowing the focus on the depth. While we acknowledge its chosen focus and inherent limitations, these constraints have been instrumental in shaping a coherent research trajectory. The insights and contributions derived from this research not only enhance the understanding of the chosen domains but also pave the way for further inquiries and advancements, fostering an enriched discourse within the scientific community.

1.3.3. Research questions

This section lists the research questions that are addressed in this thesis. These research questions have been categorised into three distinct groupings. The initial grouping of questions pertains to the exploration of [NIR](#) techniques applied to large corpora of textual scientific literature, thereby scrutinising the competence of word embeddings in addressing inquiries specific to particular domains. The second assemblage of questions delves into the examination of hyperparameters concerning word embeddings, along with their consequential effects on downstream tasks. The final cluster of queries undertakes an exploration of temporal knowledge evolution, scrutinising the potential of the temporal dimension of knowledge in conjunction with a vector space continuum.

Set 1: Neural Information Retrieval

This set of research questions addresses the first research objective to investigate the potential of [NIR](#) for deterministic data extraction which aims to tackle the challenge of extracting both implicit and explicit knowledge, exemplified by the study of underutilised crops.

- [RQ1](#): How efficacious is the utilisation of word embeddings for the extraction of domain-specific insights pertaining to interchangeability and relatedness?
- [RQ2](#): Is it possible to harness a domain-specific corpus for the purpose of training a high-calibre word embedding model capable of effectively addressing inquiries specific to the requirements of crops, such as cropping patterns, crop vernacular and scientific names and crop soil requirements? A domain-specific corpus is crucial because it provides the relevant vocabulary and context, improving the model's accuracy in addressing crop-specific inquiries and that require the utilisation of a general-purpose dataset like Wikipedia in addition to the domain specific one.

Set 2: Word Embedding Hyperparameters

This set of research questions addresses the second research objective of examining the hyperparameters of word embedding models to enhance [IR](#) accuracy and construct the temporal vector space continuum.

- [RQ3](#): How do word embedding hyperparameters affect the performance of a [DNN](#) when passed to its embedding layer on downstream tasks like classification?
- [RQ4](#): What is the effect of word embedding hyperparameters on the creation of a temporal knowledge evolution framework?

Set 3: Knowledge Evolution

This final set of research questions addresses the last research object of studying temporal knowledge evolution in the efforts of understanding scientific events and trends.

- [RQ5](#): Can temporal analysis of word embeddings through vector space continuum methodologies serve as a viable approach for the examination of knowledge evolution?
- [RQ6](#): In the temporal knowledge evolution framework, is it capable of detecting key scientific events and mapping specific scientific trends and trajectories?

1.4. Research Contributions.

The list below indicates the main contributions of this PhD.

- **Explicit Knowledge Extraction of Underutilised Crops Using Big Data and Text Mining**

To achieve the first research objective—extracting explicit knowledge of underutilised crops—the research employed advanced text mining techniques using large-scale online datasets. This methodology leveraged big data resources, such as Google search data, to identify previously unrecognized geographical regions associated with underutilised crops. By analysing search patterns and content, the study was able to detect both well-documented and newly emerging localities where these crops are cultivated, which were previously not widely recognized in the scientific literature Mohamed et al., (2017). This method provided a robust framework for the explicit extraction of crop-related information from dispersed and unstructured data, demonstrating the effectiveness of big data analytics in agricultural research.

- **Automated Identification of Multi-lingual Local Names of Crops Using Information Retrieval**

To further the first objective of extracting both implicit and explicit knowledge, the study developed an innovative information retrieval methodology that integrates word embeddings with optimized hyperparameters. This approach constructs high-quality word embeddings that effectively capture the relationships between multi-lingual local crop names and their English and scientific counterparts. The method employed a combination of embedding techniques, such as [word2vec](#), tuned to maximize accuracy in multilingual contexts. This automated system successfully linked diverse local names to standardized nomenclature, facilitating better data integration and retrieval across different languages, thereby enhancing the scope and accuracy of crop-related knowledge extraction.

- **Neural Information Retrieval (NIR) for Crop Soil Requirements**

Addressing the first research objective of implicit knowledge extraction, the study employed [NIR](#) techniques using optimized word embeddings. The embeddings were used to extract information on specific soil requirements for various underutilised crops. The methodology involved adjusting word2vec

hyperparameters to enhance the model's ability to discern relevant attributes from large text corpora. This approach, documented in the proceedings of the ICCA at UPM, Malaysia (2019), demonstrated significant improvements in the accuracy of soil classification predictions based on scientific crop names, underscoring the effectiveness of NIR in agricultural data science.

- **Optimization of Hyperparameters for Downstream Classification Tasks**

In line with the second research objective, which focuses on the impact of hyperparameter tuning on downstream tasks, this study investigated various hyperparameter combinations for word2vec embeddings. The research assessed how these combinations influenced the performance of different Deep Neural Network ([DNN](#)) architectures—specifically Recurrent Neural Networks ([RNNs](#)), Convolutional Neural Networks ([CNNs](#)), and Long Short-Term Memory ([LSTM](#)) networks—in text classification tasks. Findings revealed that [LSTM](#) networks showed superior resilience to variations in word2vec hyperparameters, suggesting their adoption for applications requiring robust classification performance. This contribution provides a refined mechanism for embedding neural networks with high-quality representations, optimizing performance across various [NLP](#) tasks.

- **Hyperparameter Tuning for Interchangeability in Word Embeddings**

As part of the second core objective, this study introduced a strategic approach to hyperparameter tuning aimed at enhancing the accuracy of interchangeability tasks within word embeddings. Interchangeability refers to the ability of distinct named entities to substitute for each other within a specific context. By carefully selecting and testing various hyperparameter configurations, the research identified optimal settings that improved the model's capability to accurately predict interchangeable entities. This advancement directly supports the explicit knowledge extraction process by refining the retrieval methods for underutilised crop data.

- **Hyperparameter Tuning for Relatedness in Word Embeddings**

Also contributing to the second research objective, the study explored hyperparameter configurations that enhance the relatedness measurements between distinct entities in word embeddings. Relatedness quantifies the degree of correlation between entities, a critical function for many downstream applications in [NLP](#). The research proposed a set of hyperparameters that

optimized the performance of relatedness tasks, facilitating more accurate implicit knowledge extraction, particularly in the context of agricultural datasets. These optimized configurations enable more effective retrieval of nuanced relationships between crop attributes.

- **Big Data Transformation Methodology for ArXiv Articles**

To meet the third research objective of studying temporal knowledge evolution, the research developed a comprehensive big data transformation methodology to handle a vast dataset of 2.3 million ArXiv articles (1.7TB). Utilizing advanced public cloud infrastructure, the study orchestrated a data processing pipeline that included multiple stages—data extraction, cleaning, transformation, and storage—culminating in a format suitable for generating word embeddings and temporal knowledge evolution models. This robust data preparation strategy was essential for creating the knowledge evolution framework, enabling the analysis of temporal patterns across diverse scientific domains.

- **Innovative Methodology for Studying Temporal Knowledge Evolution**

To address the third core objective of constructing a framework for knowledge evolution, this study introduced a novel methodology that utilizes a temporal vector space continuum. This technique transforms a chronological compilation of scholarly articles into a dynamic vector space model, where each time slice represents a different period in the dataset. By applying advanced analytics, curve fitting, and anomaly detection methods, the research facilitated the exploration of temporal developments and shifts in scientific knowledge. This methodology allows for the identification of emerging trends, evolving scientific topics, and the detection of disruptive events, providing a comprehensive tool for understanding the progression of knowledge over time.

- **Anomaly Detection in Temporal Knowledge Evolution**

This contribution focuses on detecting anomalies or ‘disruptive events’ within the temporal knowledge evolution framework. By examining significant incidents that cause substantial changes in scientific literature—such as breakthroughs, methodological shifts, or paradigm changes—the study developed a framework to identify these events through empirical analysis and comparison with well-documented historical data. This approach enables the identification of anomalies that mark significant shifts in scientific knowledge,

providing valuable insights into the dynamics of scientific progress and innovation.

- **Trend Detection in Temporal Knowledge Evolution**

Finally, the research contributes to the core objective of analysing scientific trends over time by developing a methodology for detecting long-term trends and shifts within the temporal vector space. This method identifies prolonged patterns of scientific discourse, highlighting emerging topics or the decline of formerly prominent subjects. The study applied this technique to various scientific entities, comparing their trajectories against randomly selected counterparts to validate the model's proficiency in discerning subtle trends. This contribution enhances our understanding of how scientific knowledge evolves, supporting more informed decision-making in research and policy development.

1.5. Thesis guide

This dissertation is structured into seven distinct chapters. Chapter [1](#) offers a preliminary overview of the research objectives and elucidates the primary subjects of investigation. Chapter [2](#) is dedicated to a detailed review of the literature, providing a comprehensive background to the research, and highlighting pertinent works in the field. In Chapter [3](#), the methodologies for the five experiments conducted in this research are comprehensively detailed. Chapter [4](#) articulates three experimental designs and their consequent findings, commencing with an exploration of text mining techniques for [IR](#), progressing to an analytical examination of word embeddings for the efficacious extraction of crop science data, and culminating in a detailed assessment of [NIR](#)'s applicability in crop soil requirement categorization. Chapter [5](#) dives into the intricacies of navigating the hyperparameter space of word embeddings, specifically for downstream applications such as document classification. Chapter [6](#) presents the final experiment, focusing on the temporal knowledge evolution framework, and provides an in-depth discourse on its experimental design and resultant findings. The dissertation concludes with Chapter [7](#), which encapsulates the salient conclusions and discussion on the implications of the research.

Chapter 2

Literature Review

2.1. Introduction

This chapter introduces the works related to our research, highlighting various methodologies and innovations, and identifying weaknesses and gaps. Central to our exploration are three core areas. First, we aim to identify the optimal hyperparameters employed in word embeddings tailored for specific downstream tasks, such as classification and [NIR](#). By determining these hyperparameters, we can craft optimal word embeddings, enabling effective [IR](#) from a vast corpus of data. These optimal hyperparameters ultimately pave the way for constructing temporal vector spaces to examine the evolution of knowledge over time.

The journey begins with a historical overview of word embeddings, touching upon its early foundational papers. We explore the "one-hot" representation, pivotal in understanding the origins of the challenges in word representation. As we move forward, distributed representations emerge as suitable candidates for representing entities in a corpus. Given the high dimensionality of these representations, we'll dive into embedding visualisation techniques that streamline the complex knowledge representation. We'll also touch upon the significance of the co-occurrence matrix in the realm of [NLP](#). Obtaining high quality embeddings is crucial to [NIR](#), we spotlight models such as [word2vec](#) (Mikolov et al., [2017](#)). An in-depth analysis of [word2vec](#)'s architecture will be presented, highlighting its "continuous bag of words" approach and juxtaposing it with the skip-gram architecture (Goldberg et al., [2014](#)). A salient feature of these embeddings is their ability to preserve semantic meanings in word representations. This is further elucidated with the concept of cosine similarity, showcasing how mathematical computations on vector representations can grasp the semantic connotations of words. Within the vast array of word embedding models, we'll curate and discuss a select set, evaluating their advantages and drawbacks. Proceeding further, we transition to document embedding representation, an indispensable tool for document classification, sentiment analysis, and gauging document similarity. As we explore the importance of both word and document embeddings, it's essential to dive into real-world applications of word embeddings.

We'll review how various research groups have harnessed this technology for intricate tasks, offering insights into both its potential and limitations.

Our research underscores the significance of hyperparameter optimization in word embeddings, emphasising its profound impact on the performance of downstream tasks. One tangible example of this is the potential use of [word2vec](#) in recommendation systems. Here, precise hyperparameter selection becomes paramount, ensuring the system's alignment with specific tasks, such as making accurate recommendations. The challenges of tuning [word2vec](#) for large-scale recommendation systems will be elaborated upon, detailing efforts made to overcome these hurdles.

Knowledge evolves over time and word embeddings have the ability to capture the evolution both implicitly and explicitly. Initially, literature predominantly honed in on the semantic shifts of particular entities. These shifts reveal profound insights. However, as we track the trajectory of knowledge over the years through temporal knowledge evolution, it's not just the semantics that shift that is captured but also events and trends that represent the evolution of knowledge. Knowledge expands, new ideas are birthed, while others fade into obscurity. This chapter concludes by examining methods to extract latent knowledge from expansive data sets, showcasing how researchers have unearthed insights that weren't explicitly highlighted in prior literature.

2.2. Word embeddings

2.2.1. An Overview of word embeddings

Word embeddings refer to the mathematical representation of words into a compact vector space, often represented in lower dimensions compared to one-hot encoding, which is a binary vector representation where only one element is 'hot' (set to 1), and the rest are 'cold' (set to 0). This concept is synonymous with terms like word vector or distributed word representation. This transformation facilitates the mathematical manipulation and comparison of words, allowing for the analysis of linguistic relationships and patterns across various dimensions of the vector space. When processing text with neural networks, this numerical transformation into vectors is essential, as it enables the network to operate on the textual information, thereby allowing for the quantification and analysis of linguistic features and relationships. The

foundational research they shared openly includes both the code and a model trained on 1.6 billion words. This contribution is characterised as exhibiting state-of-the-art performance in the evaluation of both syntactic and semantic word similarity. The word embedding technique is particularly remarkable for its ability to function without any manual annotation, relying instead on self-supervised Machine Learning ([ML](#)) algorithms. These algorithms are capable of learning from existing words and phrases, thereby eliminating the necessity for human interpretation. The application of word embeddings is defined by its unique capabilities, and its efficiency can be demonstrated through various examples of problem-solving across different domains.

In the following section, the foundational principles of one-hot and distributed representations will be systematically explored and elucidated. This will be followed by a presentation of data visualisation pertaining to word embeddings, coupled with an examination of the concept of dimensionality reduction. The [word2vec](#) architecture functions as a neural network and understanding its core structure is crucial for comprehending its operation. This discussion will delve into its implementation details. The subsequent section will culminate with an analysis of the [word2vec](#) similarity function and an exploration of the tuning of its hyperparameters, such as window size and dimension size.

2.2.2. One hot representation

The fundamental principle of word embedding involves the conversion of word representations into numerical values, facilitating the processing and comparison within context to extract semantic meanings and analogies from the corpus. Among various methods, the one-hot representation stands as the most rudimentary form of encapsulating words within the corpus. In this representation, each unique word in the corpus is denoted by a distinct vector. This vector is constituted by binary values $[0, 1]$, and its configuration is determined by the order of the corpus and the specific position of the word within it. When a particular word is represented, the corresponding bit in the vector is set to one, while all other bits are set to zero. One-hot encoding, a widely utilised representation in [NLP](#), has been the subject of various scholarly investigations. Yu et al. ([2022](#)) examined the effectiveness of traditional one-hot encoding in neural networks and introduced a novel approach for label representation. This new method not only captures the inherent semantic connections among different classes but also optimises them over time. As a result, the approach not only boosts the accuracy of

classification tasks but also cuts down the training duration by more than 50%, often leading to better performance on test datasets. In a distinct study, Lee ([2022](#)) introduced a novel approach to emotion recognition by constructing grayscale labels as an alternative to traditional one-hot encoding, resulting in a marked improvement in the system's performance. Furthermore, Chen et al. ([2018](#)) innovated a K-way D-dimensional discrete encoding scheme, aiming to supplant one-hot encoding. This method led to a significant reduction in the embedding layer's size, up to 98%, without compromising, and in some cases improving, the overall performance. Collectively, these studies underscore the pivotal role of one-hot encoding in [NLP](#) and reflect the ongoing research to refine and expand upon this foundational method.

2.2.3. Distributed representation

The distributed representation of a word constitutes a mathematical depiction of the word vector within the corpus. Contrasting with the one-hot representation, which primarily conveys the unique identity of the word through its index (a feature that ensures orthogonality but does not capture semantic relationships), the distributed representation offers a more nuanced portrayal. It encapsulates different values for each word in the corpus and elucidates its relationship with other words therein.

In Mikolov et al. ([2013c](#)) research, they emphasised the limitations of one-hot encoding, including its disregard for word arrangement and its incapacity to portray idiomatic expressions. It presents an approach for identifying textual phrases. A comparison between distributed representations and one-hot encoding was conducted, accentuating the efficacy and richness of the Skip-gram model. The document demonstrates the potential of elementary vector addition to yield significant outcomes, implying that a subtle level of language comprehension can be attained through fundamental mathematical manipulations on word vector depictions.

Unlike the one-hot representation, the distributed representation can be employed mathematically to extract the analogies and semantics of a word in its given context. The distributed representation of the word 'SAT' demonstrated by the numerical values within the vector may not have an immediate intuitive meaning to a human observer but represents relationships within the corpus. The vector space encompassing all the distributed representations of words in the corpus is interconnected. Mathematical operations within this vector space can reveal various

linguistic properties, such as semantic understanding, interchangeability, relatedness, and analogies, thereby enhancing the depth and complexity of textual analysis.

The key distinction between distributed representations and one-hot representations lies in their dimensionality and ability to capture the semantics of words. A one-hot representation, characterised by its high dimensionality, explicitly maps each unique word in the corpus, leading to a potentially vast co-occurrence matrix, especially in the context of a large corpus. This can result in significant computational challenges.

Conversely, distributed representations offer a reduction in dimensionality, thereby decreasing the computational cost associated with processing the associated matrices. While the initial generation of the vector space for distributed representations may be relatively computationally intensive, subsequent processing becomes significantly more efficient, even facilitating rapid online computation for production applications.

Typically, the number of dimensions in a distributed representation range from one hundred to three hundred. Adewumi et al. ([2020](#)) conducted an investigation into the optimal configuration of hyper-parameters for [word2vec](#), with a particular focus on the impact of vector dimension size. Through empirical analysis, the authors elucidated that an increase in vector dimension size beyond a specific point led to a degradation in quality or performance. In contrast to one-hot encoding, where each word is represented as a vector with a single '1' in a unique dimension and zeros everywhere else.

The underlying mathematical principle of distributed representations is to achieve a lower-dimensional space that can effectively represent the meanings of all words in the text. This approach not only enhances computational efficiency but also allows for a more nuanced understanding of word relationships within the corpus. The core of word embeddings is indeed encapsulated in this distributed representation, making it a vital component in [NLP](#) and related fields.

Distributed word representations have transformed how machine process language, capturing rich semantics. These numerical embeddings, generated by techniques like [word2vec](#), can be visualised using methods such as t-Distributed Stochastic Neighbor Embedding ([t-SNE](#)) or Principal Component Analysis ([PCA](#)). These visualisations unveil semantic relationships and contextual nuances, bridging the gap between mathematical representation and human comprehension.

Following the exploration of word embeddings, which focus on capturing semantic relationships between words by representing them in vector spaces, it's important to discuss another pivotal text representation technique: The Term Frequency-Inverse Document Frequency ([TF-IDF](#)). While word embeddings capture contextual meanings and can be fine-tuned with distributed training to handle vast datasets and complex semantic relations, [TF-IDF](#) operates on a different principle. Instead of capturing semantic relationships, [TF-IDF](#) underscores the importance of words within documents in relation to an entire corpus. This distinction between capturing word relationships versus determining word significance is crucial, as both methods cater to different aspects of text analysis. Bridging these concepts provides a comprehensive understanding of how we can represent and extract meaning from text in varied ways. [TF-IDF](#) is a foundational statistic in [IR](#) and text mining that measures a word's importance to a document within a broader corpus. By combining the term frequency ([TF](#)), which represents the occurrence rate of a term in a specific document, with the inverse document frequency ([IDF](#)), which gauges the term's rarity across all documents, [TF-IDF](#) assigns weight to terms based on their specificity to individual documents. This method aids in tasks like document summarization, text classification, and [IR](#) by emphasising terms that are most distinguishing within a document relative to a larger collection.

2.2.4. Embedding dimensionality reduction

The distributed representation of word embedding is characterised by a relatively low-dimensional vector space that encapsulates the semantic meaning of words within a given corpus. Typically, this dimensionality ranges in the hundreds rather than thousands, a scale that defies visualisation through conventional 2D or 3D diagrams. [PCA](#) has been introduced as a method to visually represent these dimensions within the vector space. In a preliminary experiment, we utilised the news articles as a smaller dataset to investigate the knowledge evolution framework. Our plan was to subsequently progress to the larger arXiv dataset. This visualisation technique aids in comprehending the clustering of words and their corresponding vectors. However, from a mathematical standpoint, the reduction in dimensionality renders it infeasible to derive exact meanings or utilise the representation for tasks such as analogy identification. Musil et al, ([2019](#)) employed [PCA](#) to visualise the structure of word embeddings for Czech words. Through this visualisation, the authors demonstrate that

[PCA](#) helps in revealing a higher degree of organisation by part of speech in decoder embeddings, enhancing the understanding of how different models encode linguistic information.

The examination of visualised word embeddings unveils latent linguistic patterns, harmonising with dimensionality reduction methods like [t-SNE](#) and [PCA](#). Significantly, these embeddings derive from the co-occurrence matrix, carefully capturing contextual relationships between words. The synergy among dimensionality reduction, visualisation, and the co-occurrence matrix enhances our ability to unveil intricate language structures, opening avenues for advanced natural language understanding and text analysis.

2.2.5. Co-occurrence matrix and dimensionality reduction

Dimensionality reduction pertains to the process of decreasing the number of variables or dimensions within a dataset, all the while conserving paramount information such as intrinsic structures, patterns, and relationships. The principal objective of dimensionality reduction resides in the streamlining of complex high-dimensional datasets, in order to mitigate challenges like the curse of dimensionality. This process concurrently seeks to enhance computational efficiency and interpretability, thereby maintaining the core essence of the original data.

A co-occurrence matrix, in the context of linguistics, is a two-dimensional array that records the frequency or occurrence of pairs of items, such as words or features, appearing together within a specific context or dataset. Each cell within the matrix represents the count of co-occurrences for a particular pair of items. The co-occurrence matrix provides each word with a distributed representation of dimensionality n , where n is the size of the dictionary. Co-occurrence matrices are employed to capture patterns, relationships, or associations between items and are utilised in fields like [NLP](#), [IR](#), and [ML](#). Hasan et al. ([2021](#)) conducted a thorough examination of [PCA](#), particularly within the domain of [ML](#). The study accentuates the proficiency of [PCA](#) in mitigating the intricacy of data, all the while maintaining indispensable information. As delineated in the paper, implementing dimensionality reduction, [PCA](#) not only curtails the spatial requisites of data but also reduces the computational time, thereby augmenting the overall computational efficacy.

Dimensionality reduction will be applied to decrease the dimensionality of the co-occurrence matrix pertaining to specified terms within a given contextual

framework. The co-occurrence matrix serves as an effective representation of word relatedness, particularly when the scale of the corpus is considerable, reaching the magnitude of "millions of words." However, a notable challenge inherent in the co-occurrence matrix lies in its dimensionality, which expands exponentially with increasing corpus size. This expansion can potentially overwhelm the computational capacity of conventional processors. Commencing with the one-hot encoded representation vector and proceeding towards the creation of a distributed representation, followed by the formulation of the co-occurrence matrix and its subsequent factorization for dimensionality reduction, we culminate in the conceptualization of the word embedding vector. This vector embodies reduced dimensionality and encapsulates the essence of co-occurrence analogies aligned with semantic meaning. [PCA](#) can be applied to reduce the dimensionality of the co-occurrence matrix while preserving the most important information. Here's how you can use [PCA](#) to achieve this:

Let's consider a co-occurrence matrix C with dimensions $(n \times n)$, where n is the number of elements being analysed.

1. Calculate the mean vector μ of the rows (or columns) of the co-occurrence matrix C . This can be done by summing up the rows and dividing by n

$$\mu = \frac{1}{n} \sum_{i=1}^n C_i$$

2. Centre the co-occurrence matrix by subtracting the mean vector from each row (or column):

$$\bar{C} = C - \mu$$

3. Calculate the covariance matrix Σ of the centred co-occurrence matrix $\bar{C}$

$$\Sigma = \frac{1}{n-1} \bar{C}^T \bar{C}$$

4. Compute the eigenvectors v_i and eigenvalues λ_i of the covariance matrix Σ :

$$\Sigma v_i = \lambda_i v_i$$

5. Sort the eigenvectors in decreasing order based on their corresponding eigenvalues.

6. Choose the top k eigenvectors to form a projection matrix W (where k is the

desired number of dimensions for reduction).

7. Project the centred co-occurrence matrix $\bar{C}$ onto the lower-dimensional space using the projection matrix W

$$Y = \bar{C} W$$

Here, Y is the transformed co-occurrence matrix with dimensions $(n \times k)$, capturing the most important relationships between elements in a reduced-dimensional space.

The synergy of the co-occurrence matrix and distributed training becomes apparent in methods like [word2vec](#), which focus on word embeddings. These approaches utilise co-occurrence statistics amidst words to construct compact, distributed representations that encapsulate semantic insights. Subsequently, these representations undergo enhancement via distributed training, tapping into parallel processing capabilities to fine-tune model parameters. This dual approach enhances the model's proficiency in capturing the nuanced semantic relationships embedded within the co-occurrence patterns present in linguistic data.

The forthcoming exploration encompasses a detailed dissection of the [word2vec](#) framework, elucidating its operational mechanics. Subsequent analysis seeks to unveil the inherent limitations embedded within the model, fostering a comprehensive understanding of its inherent scope and confines. Beyond this, the research endeavour extends its scope to encompass a diverse array of alternative word embedding methodologies, each poised to be carefully juxtaposed against the foundational [word2vec](#) paradigm.

2.3. NIR and Embeddings

Neural Information Retrieval ([NIR](#)) enhances information retrieval by utilizing neural network models like [CNNs](#), [RNNs](#), and transformers to capture complex data patterns and contextual nuances. This approach, through end-to-end learning, negates the need for manual feature engineering and adapts efficiently with fine-tuning, improving relevance and handling ambiguous queries. However, [NIR](#) faces challenges including high computational costs, large data requirements, interpretability issues, and potential biases. Despite these challenges, [NIR](#) offers significant advantages in scalability and relevance. Its applications span web search engines, question-

answering systems, recommender systems, and digital libraries, where it significantly improves the accuracy and relevance of retrieved information.

2.3.1. Word embedding architecture

This section delves into an elaborate exposition of the technical instantiation of the [word2vec](#) neural network. Essentially, the architecture of [word2vec](#) can be distilled into elementary input, output, and hidden layers, as delineated in Figure 1.

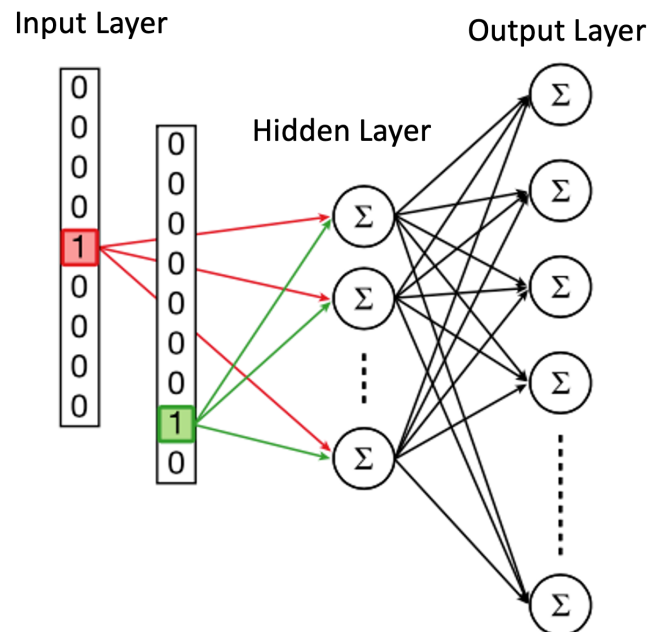


Figure 1 Neural Network representation of [word2vec](#).

Considering that [word2vec](#) functions using a neural network framework, it adheres to the basic architecture of input, hidden, and output layers. The neural network's layers in [word2vec](#) correspond to word embeddings, with the network's weights being initially set to random values. The network iteratively refines these weights by analysing sentences containing the designated word, ultimately optimising them to best encapsulate the word's context within the corpus. Typically, the embedding corresponds to the weight vectors, aligning with the hidden layer's activation patterns. The remarkable breakthrough of [word2vec](#) lies in the resultant weight vectors located within the hidden layer. These resultant vectors for various words can be subjected to mathematical operations, enabling the resolution of diverse linguistic inquiries regarding similarity, relationships, and semantic meaning. A classic illustration involves the vectors for "king," "man," "queen," and "woman." In an extensive corpus, the vectors for "king" and "man" tend to be proximate, just as those for "woman" and "queen." An elegant manifestation of [word2vec](#)'s capabilities emerges in its

mathematical interpretation of the equation " $Vector[king] + Vector[queen] - Vector[man] = ?$ " which yields the outcome " $Vector[woman]$."

The essence of [word2vec](#)'s achievement lies in its capacity to translate the words within a given corpus into mathematical vectors, a consequence of the neural network's hidden layer dynamics. This vector space facilitates quantitative exploration of semantic nuances and relational aspects among various entities present in the corpus.

Input Layer: The input layer is composed of individual neurons, each corresponding to a unique word present within the corpus. Within this layer, a one-hot representation is employed, constituting an extensive vector whose length is contingent upon the size of the corpus. This vector assumes the role of an index indicator for a specific word within the corpus. While the majority of its components are assigned a value of zero, the element corresponding to the particular word in question is assigned a value of one.

Output layer: The neurons within the input layer correspond to words, establishing a direct mapping between the representation of words and the output layer. This alignment is mirrored in both layers. The output vector embodies the anticipated outcome when presented with a specific input vector. The operational sequence commences with the initialization of the random weight values for the hidden layer, and then the input vector is applied. Following this, the output vector is computed, allowing for the quantification of error by the configuration of the resultant vector in relation to the target vector in the output layer. Subsequently, backpropagation of the gradient of the error with respect to network weights occurs throughout the network, thereby performing adjustments to the weight matrices within the hidden layers. This iterative adaptation process aims to align the output vector more closely with the target vector. Iterating this process across the entire training dataset facilitates the gradual refinement of the model's accuracy. The underlying objective is to attain the highest level of fidelity between predicted and actual output vectors.

Hidden layer: The hidden layer draws upon the foundational premise of the distributional hypothesis, as articulated by Firth, J. ([1957](#)) "You shall know a word by the company it keeps." Embedded within this hypothesis, the hidden layer assumes the role of a vector representation, encapsulating a specific word within the corpus by assigning numerical values within the range of -1 to 1. Typically characterised by a

dimensionality of several hundreds, the hidden layer's weight matrices furnish the essential parameters. By passing the input vector through the network's weight matrices, the outcome manifests as a probability distribution. This distribution delineates the likelihood of the central word in the context, which serves as the output word vector, appearing within its given context framework provided as input vectors. The process of word embedding occurs within the latent layers that are inherent to the architecture of [word2vec](#).

In practice, the [word2vec](#) implementation involves the integration of a training window. Consider the sentence "The brown fox jumps on the lazy dog." For a window size of 2 centred around the word *[jumps]*, the input layer accommodates four distinct vectors— $V[brown]$, $V[fox]$, $V[on]$, and $V[the]$ —with the anticipated output being $V[jumps]$. These parameters encompass the concept of predicting the word *[jumps]* based on its contextual neighbours, adopting a window that spans two words to the left and two words to the right. This procedure is referred to as "masking," wherein the output word is concealed, and the network is tasked with making a prediction for it. Figure 2 graphically illustrates the arrangement of input multi-vectors and the corresponding target output vector.

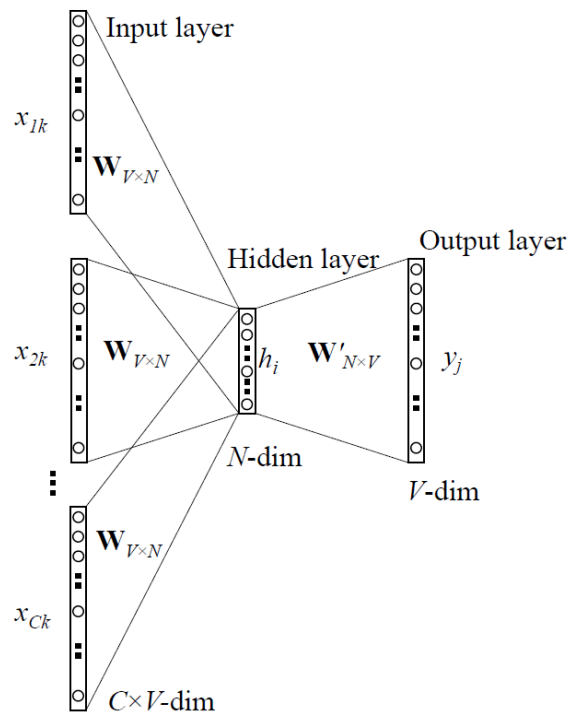


Figure 2 Implementation of [word2vec](#) for multiple input vectors.

2.3.2. Continuous bag of word & Skip Gram

[CBOW](#), a notable paradigm within word embedding techniques, operates by employing a corpus-derived set of words as input, tasking the neural network with predicting the most probable word co-occurring within the contextual vicinity of these input words. Conversely, the Continuous Skip-gram model employs a different approach, utilising a singular word vector as a foundation for predicting the words encompassing its context. It is worth noting that [CBOW](#) exhibits computational efficiency when compared to skip-gram; however, empirical evidence substantiates the enhanced accuracy of the skip-gram model in capturing intricate linguistic nuances. For a visual explanation of the distinctive architectural underpinnings characterising these training methodologies, refer to Figure 3. This portrayal of [CBOW](#) and the Continuous Skip-gram model within the figure offers a graphical elucidation of their structural disparities, thereby augmenting our understanding of their operational distinctions.

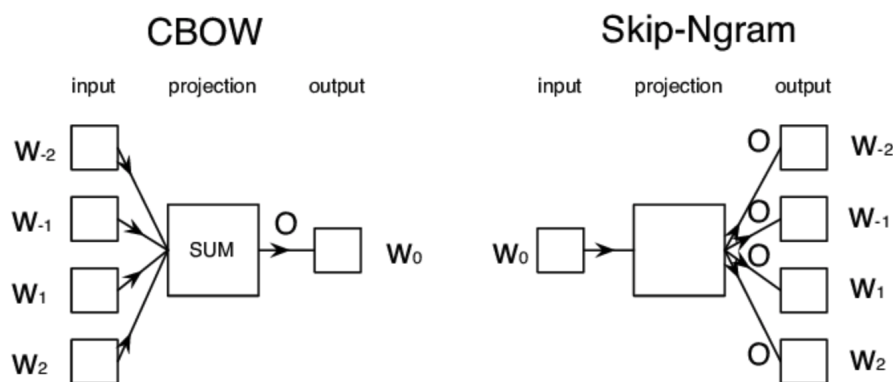


Figure 3 [Word2vec CBOW](#) vs [Skip-Ngram](#)

2.3.3. Semantic meaning in word embeddings

Building upon the preceding discourse on the architectural underpinnings of [word2vec](#) and its role within the neural network framework, a deeper exploration of word embeddings comes to the fore. This exploration takes into consideration the intricate relationships that exist among words, where terms such as "car" and "vehicle" may be deemed synonymous, yet their usage can often entail subtle differentiations. Additionally, words like "car" and "road" exhibit contextual relationships that go beyond mere synonymy, encompassing a web of interrelatedness within specific contextual confines. While these terms may not be inherently interchangeable, they co-occur in

distinct contextual settings. In the practical realm, particularly within business contexts, the necessity emerges to identify all named entities within a given context, potentially leading to the interchangeable treatment of terms like "car" and "vehicle" for specific business cases, while maintaining delineation from terms like "road." This distinction draws attention to two fundamental concepts: the concept of word interchangeability and the broader dimensions of attributes and relatedness, often manifested through the lens of "related analogies." There are similar words such as "car" and "vehicle" that can be considered synonyms. However, there are many possible differences between the two words. Related words like "car" and "road" are not synonyms but related. They are not interchangeable but still, they appear with the same context somehow. Word embeddings can still find similarities between "car" and "vehicle" and "road". In real business problems, it might be needed to get all the named entities in a specific context. So, the word "car" and "vehicle" might be the same as far as your business case is concerned. In that case, it is required that the system will differentiate between the word "road" and the word "car". This can be summarised into two concepts, word interchangeability and entity relatedness.

2.3.4. Embedding cosine similarity

One of the most renowned outcomes stemming from the [word2vec](#) methodology is the establishment of a similarity function that operates on pairs of words. To elaborate, this function quantifies the similarity between two distinct embedding vectors, each representing a different word within the context window. Various mathematical techniques can be utilised to compute the similarity between these vectors, including methods such as the Euclidean distance and the cosine similarity. Concerning cosine similarity, it is computed by taking the dot product of the vectors. A simplified conceptualization involves interpreting this process as an evaluation of the angle between the two vectors within the vector space.

Jatnika et al. ([2019](#)) published research that delves into the computation of word similarity in the English language through techniques of word representation. The utilisation of [word2vec](#) is explored for converting words into vectorized forms. The model is constructed using a corpus of 320,000 English Wikipedia articles, followed by the application of the Cosine Similarity calculation method to quantify the similarity between words. To evaluate the model, it is tested against WordSim-353, consisting of 353-word pairs, and SimLex-999, with 999-word pairs, both having human-assigned

similarity values. Pearson Correlation is employed to assess the correlation accuracy. The obtained results reveal correlations of 0.665 for WordSim-353 and 0.284 for SimLex-999 when configured with a window size of 9 and a vector dimension of 300. Srivamsi et al. (2023) explored the process of converting words within biomedical data into word embeddings. The research highlights the Milvus vector engine, an open-source software designed to efficiently and accurately manage large-scale data representations in vector spaces. This tool proves particularly valuable for handling high-dimensional biomedical data containing numerous distinct terms. By utilising the Milvus vector engine, the research focuses on transforming biomedical data into numerical vectors, facilitating applications in [ML](#) and data analysis. In the biomedical context, the adoption of word embeddings offers the potential to uncover new insights and discoveries through enhanced analysis of extensive datasets. These embeddings serve to reveal patterns and connections in biomedical literature and may even support the categorization of medical articles based on their content.

2.3.5. Type of embedding models

Multiple alternative methods for word embeddings have been developed to overcome specific limitations present in [word2vec](#). These methods span distinct categories, including word-level techniques like [GloVe](#) Pennington et al. (2014) and fastText, which capture semantic nuances, as well as Document to Vector [doc2vec](#), Hyper Document to Vector [hyperdoc2vec](#) (Han et al., 2018), Citation to Vector [cite2vec](#) (Berger et al. 2016), Contrast set to Vector [con2vec](#) (Nguyen et al. 2021) and [LDA](#) (Latent Dirichlet Allocation) to Vector [lda2vec](#) (Moody et al. 2016), which extend embeddings to documents. Additionally, graph and network embeddings such as Node to Vector [node2vec](#) (Grover et al. 2016) and Graph to Vector [graph2vec](#) (Narayanan et al. 2017) offer ways to represent structural information in graphs. It is important to note that each of these approaches has its own inherent limitations. In the subsequent section, we delve into a comprehensive exploration of several of these techniques, accompanied by a comparative analysis to highlight their respective strengths and weaknesses.

Word embedding models

Fasttext, an influential word embedding algorithm, was introduced by Joulin et al. (2016a) in Facebook AI research. In its essence, fasttext tends to exhibit slower

performance compared to [word2vec](#), even though it was implemented in C++. The experience of running the code with the Gensim library is very similar. Developed in C++, fasttext simultaneously extends its accessibility through Python. Contrary to several contentions, Facebook AI research posits that fasttext is adept at undergoing training on a billion-word corpus in under ten minutes, facilitated by a standard multicore Central Processing Unit ([CPU](#)). These assertions were elucidated within the research work authored by Joulin et al. ([2016a](#)). Overcoming a critical impediment concerning the operational speed of fasttext within memory, he proffers a remedy by applying product quantization for word embedding storage within memory. This remedy substantiates its superior performance over contemporaneous benchmarks by a substantial margin. Joulin et al. ([2016b](#)) introduce fasttext as an efficient text classification tool. Results demonstrate that fasttext not only matches the accuracy of advanced deep learning models but also excels in speed. On a standard multicore [CPU](#), it can be trained on over a billion words in less than ten minutes and can classify 500,000 sentences among 312K categories in under a minute. The underlying reason for fasttext's reduced swiftness can be attributed to its operational methodology. The concept of similarity within fasttext extends more profoundly than that within [word2vec](#). Specifically, the calculation of similarity in fasttext involves an analysis of sub words. For instance, the term "superpower" undergoes subdivision into sub word constituents such as "su," "up," "pe," and so forth. The algorithm subsequently evaluates the similarity relationships among these sub word fragments. This intricate process, central to fasttext, contributes to its relatively slower execution speed. This method's rationale is grounded in the notion that certain sub word fragments may encapsulate stronger morphological connections to the complete word. Consequently, when two words share a substantial number of sub word fragments, their semantic similarity increases, surpassing scenarios where only complete words are considered. This characteristic aligns with fasttext's pronounced proficiency in capturing word morphology. The aspect of word interchangeability, inherently valuable within linguistic contexts, gains heightened significance within fasttext. This is particularly notable due to the observed tendency for interchangeable words to share a multitude of sub word constituents (Bojanowski et al., [2017](#)). The algorithmic slowness inherent in fasttext constitutes a noteworthy drawback within its implementation. This tardiness, as previously explored, primarily stems from the supplementary computational burden incurred through sub word calculations. Notably, this computational approach affords

fasttext a distinct advantage in capturing morphological nuances, setting it apart from alternative word embedding methodologies. Nonetheless, the adoption of sub word calculations introduces a discernible susceptibility to semantic errors, potentially engendering situations where lexically distinct words, such as "night" and "knight," are erroneously linked under a semblance of close similarity.

Pennington et al. (2014) developed [GloVe](#) which constitutes an unsupervised learning methodology aimed at deriving word embeddings. Its procedural methodology diverges from the conventional approaches employed by [word2vec](#) and fasttext. In contrast to these counterparts, [GloVe](#) necessitates the comprehensive generation of the co-occurrence matrix in a single instance, a process that demands substantial memory resources for its storage. This intrinsic characteristic may result in significant memory overhead, especially when applied to extensive textual corpora such as Wikipedia or Google News. The preparatory stage involving corpus pre-processing and the subsequent loading of the co-occurrence matrix within the [GloVe](#) framework requires a high Random Access Memory ([RAM](#)) configuration. Due to the computational demands thus posed, its practical application to extensive corpora proves to be suboptimal. Nevertheless, [GloVe](#) exhibits commendable performance when applied to smaller datasets.

In the context of word and document embeddings, which are known for their robust capabilities but often entail complex mathematical operations and sensitivity to hyperparameters, WordRank, published by Ji et al. (2015), presents a notable alternative. This algorithm simplifies the process of ranking words according to semantic relationships, standing apart from traditional embedding methods. By constructing a co-occurrence matrix, WordRank offers a more direct and effective way to discern and anticipate semantic connections. This methodology acts as a conduit between the intricate nature of conventional embedding models and the requirement for a more user-friendly and noise-tolerant solution, especially for smaller corpora. The ensuing section will delve into the intricacies of the WordRank algorithm, its practical implementation, and its unique attributes in contrast to other models in the domain. The output of the WordRank algorithm consists of a list of words ranked according to their similarity to a given input word. By leveraging this ranking and integrating it with comparison features, the system can accurately predict semantic relationships between words. One of the critical limitations of WordRank, however, lies in the sensitivity of the ranking. Errors in the top-ranked words can lead to significant

deviations in the ranked list, whereas mistakes in the lower ranks have a more negligible effect on the final prediction. WordRank is available in the Gensim library, with implementations in both Python and C++. It is worth noting that WordRank's training time is substantially longer compared to some other models. Nevertheless, it has demonstrated effectiveness in dealing with small corpora, typically consisting of less than a million words. Its performance is particularly well-suited to tasks that aim to identify interchangeability and semantic relationships between words, and it exhibits a robust resistance to noise. Conceptually, WordRank shares similarities with Learning-to-Rank techniques within the field of [ML](#).

[Word2vec](#) is a widely used word embedding technique known for its efficiency and scalability, but it has several notable limitations. A primary limitation is its inability to capture sub word information, which makes it less effective in handling morphologically rich languages. This model also struggles with out-of-vocabulary words, as it cannot generate embeddings for words not seen during training. Additionally, [word2vec](#) embeddings are static and do not capture context-specific meanings of words, leading to issues with polysemy. For example, the word "bank" in "river bank" versus "financial bank" is represented identically, ignoring contextual differences. Moreover, [word2vec](#) requires large amounts of training data to produce high-quality embeddings, making it computationally expensive and resource intensive. Despite these limitations, [word2vec](#) remains a foundational technique in natural language processing due to its simplicity and effectiveness in capturing general semantic relationships between words.

Document embedding models

[Doc2vec](#) is a technique introduced in the research paper by Le et al. ([2014](#)). The research proposed the incorporation of a "tag," which can be envisioned both as a classifier and a word vector. This tag serves the purpose of representing an entire sentence, such as the example "The brown fox jumps on the lazy dog," which can be associated with the vector "cute." This vector captures the essence of the sentence by encapsulating its semantic context, and the words that hold higher contextual frequency are inclined to exhibit greater proximity to this vector. [Doc2vec](#) operates within the realm of semi-supervised learning, with [word2vec](#) constituting the self-supervised counterpart. While [word2vec](#) autonomously learns word embeddings from unlabelled data, the [doc2vec](#) approach incorporates tagged segments that require

manual classification – representing the supervised aspect of the technique. One of the prominent practical applications of the [doc2vec](#) technique lies in its utilisation for the interpretation of user-generated comments. These comments represent a valuable source of information for a diverse range of businesses. It is widely acknowledged that comment sections often harbour a degree of bias, potentially leading to misleading implications for business strategies. Nevertheless, despite this inherent bias, business proprietors frequently engage in the analysis and quantification of user comment sections. Leveraging word embeddings for the bulk processing of comment sections has demonstrated its efficacy. A notable study by Savigny et al. ([2017](#)) explored this domain, focusing on the examination of comment sections within the context of Indonesian YouTube. The study employed a composite algorithm, integrating a supervised Convolutional Neural Network ([CNN](#)) alongside [TF-IDF](#) published by Sparck et al., ([1972](#)) and [doc2vec](#) techniques. The outcomes of this study asserted a 76.2% accuracy rate, surpassing the established baseline performance. In a study conducted by Maslova et al. ([2017](#)), the integration of [doc2vec](#) with supervised [ML](#) was examined. This approach was applied across diverse languages within social media user reviews and opinions, yielding promising outcomes for automated sentiment analysis of social networks. Following this, each sentence underwent manual categorization corresponding to its appropriate category. These categories encompassed a range of possibilities, including the genre of a movie plot, the theme of a restaurant, or even the source of the sentence within the corpus. The resulting performance of this configuration achieved an accuracy of approximately 50%, surpassing that of random guessing. This configuration can be characterised as a semi-supervised algorithm, given its amalgamation of the [doc2vec](#) technique with a [ML](#) classifier employing labelled sentence data. This structural arrangement is envisaged to produce an output in the form of a ranking number or a probability. This probability signifies the degree of proximity between a given document (sentence or comment) and a specific tag (category or theme).

The [hyperdoc2vec](#) embedding model, published by Han et al. ([2018](#)), introduces four distinct concepts: content awareness, context awareness, newcomer-friendliness, and context intent awareness. The implementation of this model involves the assignment of two vector representations to each hyper-document, coupled with the modelling of citations. This is achieved through the adoption of two primary approaches. The first approach concerns citations as textual entities. In this context,

citations are treated as words and subsequently replaced with the corresponding document title. Word embedding techniques are then applied to both the words within the corpus and specialised words generated via the replacement algorithm. Consequently, this yields word representations that jointly encapsulate both context words and citations within the same vector space. The second approach revolves around context as content. Researchers often provide summaries of documents and cite relevant references in their scientific literature. In this context-driven approach, the summary or "context" is appended to the target document. Subsequently, the [doc2vec](#) technique is employed to generate document embeddings that are characterised by their awareness of both context and content.

[Hyperdoc2vec](#) generates two distinct vectors for each operation: one for the input document and another for the output document. During the training process, inputs are aligned with context words and the "input document," while outputs correspond to the "output document."

In essence, the model attempts to predict the likelihood of a target document being associated with a given input document and context words, employing an averaging mechanism. This innovative approach thus amalgamates content awareness, context awareness, and the inherent dynamics of citation relationships, culminating in a refined understanding of the intricate interplay between documents and their contextual environments.

Berger et al. ([2016](#)) developed [cite2vec](#) which constitutes a model that was designed with the objective of implementing a 2D document navigation visualisation tool. The foundational algorithm governing this navigation system is established upon the principles of word embedding. The navigational algorithm was specifically engineered to empower users to access documents based on contextual references and citations. The amalgamation of context and target cited documents results in the formation of a unified document embedding, which effectively preserves document similarity and contextual significance.

[Con2vec](#) was published by Nguyen et al. ([2021](#)) at the Dalhousie University in Canada. This study focused on the vector embedding of Wikipedia concepts and entities. The methodology adopted involved replacing anchor text with corresponding concept names within the Wikipedia corpus. The main objective of this approach was to encompass interconnected anchor concepts found in the corpus text. This effort resulted in improved accuracy within question-and-answer datasets. The underlying

principle of this model revolves around substituting Wikipedia's concept names with their respective page IDs. Subsequent to this substitution, a comprehensive word embedding procedure is applied to the entire corpus, leading to the creation of a word representation vector space. This space serves as the foundation for measuring semantic similarity between concepts and words.

Moody et al. ([2016](#)) developed the [lda2vec](#) method that represents a distinctive approach in the realm of textual analysis, existing independently of the Gensim library and maintaining its autonomous status. [Lda2vec](#) introduces a transformative aspect by amalgamating the principles of [word2vec](#) with the concept of Dirichlet-distributed latent document-level mixtures of topic vectors. This pioneering effort adds a novel dimension to the domain by formulating a model that yields sparsely populated, yet readily interpretable, document mixtures. Noteworthy for its uniqueness, this research paradigm achieves the simultaneous acquisition of word vectors and the revelation of linear interrelationships among words, presenting a multifaceted contribution to the field.

Network/Graph embedding models

Stepping beyond the confines of word and document embeddings to the intricate domain of graph and network embeddings. This progression is a response to the realisation that interconnected networks aptly depict various real-world systems. These networks consist of nodes, representing entities, intricately linked through edges that hold significance. Similar to how word and document embeddings reveal semantic intricacies in text, graph and network embeddings aim to extract the nuanced relationships between nodes within these complex networks. Yet, these embeddings aren't just theoretical constructs. They have a dual role, offering compact representations while also empowering practical applications. These tasks span predicting links, classifying nodes, and uncovering network clusters. As we shift our focus to graph and network embeddings, we inherently value the importance of preserving and leveraging structural insights present in interconnected data. This transition seamlessly aligns with the core principles that laid the foundation for word and document embedding techniques. One such pivotal advancement is embodied in [node2vec](#), as conceptualised by Grover et al. ([2016](#)). [node2vec](#) introduces an adaptive and scalable framework designed for the extraction of feature representations from networks. This innovative approach harnesses a technique

known as biased random walks, which imparts a malleable interpretation to a node's neighbourhood. Subsequent utilisation of a Skip-Gram model facilitates the generation of continuous feature representations, effectively encapsulating the intricate network relationships that define nodes. The applicability of [node2vec](#) extends across an array of downstream tasks, encompassing node classification, clustering, and the intricate task of link prediction. In a parallel vein, the Large-scale Information Network Embedding (LINE) framework, introduced by Tang et al. ([2015](#)), constitutes a response to the intricate challenge of embedding large information networks into lower-dimensional vector spaces. What sets LINE apart is its dual focus on conserving both local and global network structures, thus rendering it pertinent to a diverse spectrum of information networks. Noteworthy is the inherent efficiency of the LINE approach, adept at generating embeddings for networks that encompass a multitude of vertices and edges, encompassing millions and billions, respectively, all within a time frame of a few hours. Another model in the evolution of embedding representation learning materialises in the form of [graph2vec](#), an innovation credited to Narayanan et al. ([2017](#)). [Graph2vec](#) introduces a fresh outlook by culminating in data-centric distributed representations dedicated to entire graphs, in contrast to the narrower scope of substructures like nodes or subgraphs. Encompassing a self-supervised paradigm, [graph2vec](#)'s versatility transcends distinct tasks including graph classification and clustering. Empirical evaluations substantiate [graph2vec](#)'s dominance over substructure representation learning paradigms, concurrently establishing its competitiveness with the forefront graph kernel techniques. These techniques extend the idea of embedding from words to more complex structures like nodes, lines, graphs, paragraphs, and items, enabling more sophisticated analysis and prediction tasks in various domains.

[Word2vec](#) is often favoured over embedding techniques such as [cite2vec](#), [con2vec](#), [lda2vec](#), [graph2vec](#), and [node2vec](#) due to its straightforward implementation, computational efficiency, and robust capability to capture semantic relationships in large text corpora. Unlike [cite2vec](#) and [con2vec](#), which are designed for specific applications like citation and contrastive analysis, [word2vec](#) provides a more versatile and general-purpose approach suitable for a variety of text mining tasks. It also avoids the added complexity and computational demands of graph-based methods like [graph2vec](#) and [node2vec](#), which are tailored for network data rather than textual data. Furthermore, while [lda2vec](#) combines topic modelling with word

embeddings, it introduces additional layers of complexity that are not always necessary. [Word2vec](#) strikes a balance by offering high-quality word embeddings with efficient performance.

2.3.6. Applications of word embeddings

Having established a comprehensive understanding of both word embeddings and document embeddings, we now transition our focus towards the practical applications that these ground-breaking techniques enable. The ability to represent words and documents in semantically meaningful vector spaces has unlocked a treasure trove of possibilities across a myriad of domains. In this section, we will delve into a spectrum of applications that highlight the versatility and transformative potential of word and document embeddings. Nikfarjam et al. ([2015](#)) successfully leveraged word embeddings techniques to extract salient features pertaining to drug reactions from a social media corpus. This approach yielded a notable 82% accuracy rate, constituting a noteworthy advancement beyond the established baseline performance. Sharma et al. ([2023](#)) presented an innovative approach to [IR](#) that leverages a semantic knowledge-based retrieval strategy, with a specific emphasis on the significance of word similarity and relevance. By combining the [word2vec](#) model with Horse Herd Optimization (HHO), the study employs this amalgamation to derive feature vectors used for classification. This integration significantly bolsters the precision of the system. Notably, the proposed [IR word2vec](#) ([IR-word2vec](#)) model outperforms established methods. The results showcase advancements in both the similarity index and query result accuracy, all achieved within a rapid execution time of 1.7 seconds. Ozsoy et al. ([2016](#)) explored the use of [word2vec](#) word embeddings for a recommendation system, specifically examining Foursquare's social media check-in data. The findings exhibited promising outcomes in terms of enhancing recommendation engine efficacy through the integration of word embeddings. Notably, word embeddings methodologies, such as [word2vec](#), have also been harnessed by Google for synonym discovery and search query expansion. This application was exemplified in the SELECTCROP agricultural tool, developed by the [CFF](#) research centre Nizar et al. ([2021](#)), wherein [word2vec](#) was employed to identify synonyms for locally diverse crop names, facilitating effective search and crop suitability assessment. Furthermore, in the realm of machine translation, Mikolov et al. ([2013b](#)) introduced a pioneering approach involving the training of word embeddings on

divergent translations of identical content, subsequently demonstrating the potential for vector-based similarity extraction and efficient translation, yielding a substantial 90% precision for English-Spanish translation. Furthermore, Chelba et al. ([2013](#)) researched a substantial one-billion-word dataset that holds promise for evaluating translation and word embeddings methodologies. Shifting to question-answering systems, while various efforts have been dedicated to automating AI agents for human query response, Weston et al. ([2015](#)) pointed out the absence of an all-encompassing solution. They proposed an enhanced memory network model empowered by word embedding as a viable avenue for achieving automated question-answering functionality. Beyond these applications, [word2vec](#) proves instrumental in sentiment analysis, exemplified by the study led by Mesnil et al. ([2014](#)) wherein the Facebook AI research group investigated the Internet Movie Database ([IMDB](#)) movie review dataset. Through a combination of [ML](#) techniques, including Naïve Bayes ([NB](#)) and Support Vector Machine ([SVM](#)) ([NB-SVM](#)), Recurrent Neural Network Language Model ([RNN-LM](#)), and sentence vectors, they facilitated sentiment classification, sharing their codebase for result replication. Furthermore, the utilisation of word embeddings emerges as a promising avenue for detecting textual anomalies, enabling the extraction of novel features from diverse text sources like social media statuses and tweets. Kurata et al. ([2022](#)) developed Identification of Anti-coronavirus Peptides ([iACVP](#)), in which [word2vec](#) was applied to systematically explore [ML](#) methods for [iACVP](#)s. The Random Forest classifier with [word2vec](#) was found to be particularly effective, with the [word2vec](#) dictionary and optimal k-mer value enhancing discrimination between samples, leading to the creation of the [iACVP](#) web server. Styawati et al. ([2022](#)) conducted sentiment analysis on online transportation service applications. Lilleberg et al. ([2015](#)) develop an efficient classification method using the [word2vec](#) text embedding model and the Support Vector Machine ([SVM](#)) algorithm. [Word2vec](#), implemented with the skip-gram architecture, was used for feature extraction, transforming words into vector form, while [SVM](#) handled data classification, resulting in better accuracy. Hajibabaei et al. ([2021](#)) research focuses on addressing the complexity of node classification in extensive graph data within the [ML](#) domain. It presents a novel approach for semi-supervised node classification using the Wikipedia hyperlink dataset. This method leverages both the network structure and textual node descriptions to create a multiclass classifier capable of predicting likely categories for test nodes. The study highlights the effectiveness of integrating [word2vec](#) and directed

graphSAGE algorithms to achieve encouraging outcomes in graph multiclass classification. Additionally, the research employs [t-SNE](#) to visually represent node embeddings in a two-dimensional layout.

2.3.7. Word embedding limitations

[Word2vec](#)'s efficacy is well-established for single-word representations, yet the challenge lies in effectively encoding multi-word entities. Previous research efforts, including Facebook's InferSent published by Conneau et al. ([2017](#)) and the earlier Skip-Thoughts concept published by Kiros et al. ([2015](#)), have sought to address this without definitive success. An elementary strategy involves deriving representations for multi-word entities by computing the mathematical average of their constituent word embeddings. In the realm of document classification, [TF-IDF](#) emerges as a prominent technique, wherein word frequency is balanced by inverse document frequency. [TF-IDF](#) has enjoyed long-standing success due to its simplicity and efficacy. This approach contrasts with more complex models like [doc2vec](#) and [word2vec](#), which are challenging to debug and modify. While [TF-IDF](#) typically omits stop words to avoid confusion, [word2vec](#) retains them for vector-based calculations. The divergence between word embeddings and [TF-IDF](#) also extends to model complexity. [TF-IDF](#)'s classical baseline involves straightforward counting and calculation, enabling easier modification and control. Conversely, intricate models like [word2vec](#) and [doc2vec](#) pose challenges in debugging and altering, at times leading to obscure error sources. While not universally applicable, word embeddings introduce unprecedented possibilities. [TF-IDF](#) offers simpler execution and comprehension, whereas word embeddings are notably more intricate. Nonetheless, [word2vec](#) confronts limitations in model expansion. Incorporating additional text into an existing trained model proves challenging, lacking established techniques for seamless extensions. Such expansion necessitates revising the original corpus and retraining the model, rendering it computationally intensive. In contrast, the [word2vec](#) model excels in representing individual words' contextual nuances, but it inherently lacks the capacity to classify word sequences. Determining the similarity between two sentences demands supplementary approaches. A prevalent method involves calculating the mean vector representation of words within a sentence, though this oversimplifies intricate semantic relationships inherent in word sequences. Benedetto,

(2023) conducted an experiment with [word2vec](#) and [TF-IDF](#) methodologies in the domain of estimating question complexity. The research studied the comparative performance of these approaches and sheds light on their efficacy in the evaluation of domain-specific knowledge, thereby indicating their pertinence for tasks reliant on specialised subject understanding.

2.3.8. Research Trends in Word Embeddings and Transformer Models

Current research trends in word embeddings, particularly techniques like Word2Vec, fastText, and GloVe, focus on refining these models for improved semantic understanding and application across diverse domains. These models have proven effective in capturing intricate linguistic relationships and have been widely applied in sentiment analysis, recommendation systems, and biomedical data analysis, as demonstrated by studies such as those by Jatnika et al. (2019) and Srivamsi et al. (2023). However, the landscape of [NLP](#) has shifted significantly with the advent of transformer-based models like GPT Radford et al., (2019). It is noted that transformers-based models are a more advanced techniques such as [BERT](#) published by Devlin et al. (2018) and Distilled [BERT](#) ([DistilBERT](#)) published by Sanh et al. (2019), particularly when applied to large datasets. Transformers-based models excel across various academic domains. [DistilBERT](#) comes close to [BERT](#) in performance, and these models surpass other methods even with limited data. Transformer-based models using linguistic features perform well in reading comprehension tasks, whereas those employing frequency-based features (like [TF-IDF](#)) and word embeddings (such as [word2vec](#)) are more effective in assessing domain knowledge (Benedetto, 2023). These transformer models have outperformed traditional word embeddings in various [NLP](#) tasks due to their ability to capture context more dynamically through mechanisms like self-attention. For instance, [BERT](#)'s bidirectional approach allows it to understand the context of a word based on both preceding and following text, offering a more nuanced understanding than the unidirectional approach of [word2vec](#). Despite these advancements, transformer models have their own limitations, particularly in computational efficiency and resource requirements, which can be prohibitive for large-scale and real-time applications. Studies like those by Ethayarajh (2019) suggest that while transformers offer enhanced performance,

they also demand significantly more computational power and data to train effectively compared to traditional word embeddings.

In the context of this research, which focuses on the temporal evolution of knowledge and extracting implicit and explicit information about underutilised crops, transformer models present a promising avenue. The ability of transformers to dynamically adjust and understand context over time aligns well with the goals of tracking semantic shifts and temporal knowledge evolution. However, integrating these models with traditional embeddings to leverage their respective strengths—contextual depth from transformers and computational efficiency from word embeddings could yield a robust solution for specialized domain applications.

In summary, while traditional word embeddings continue to play a critical role in various [NLP](#) tasks, the current research trend is increasingly favouring transformer-based models for their superior contextual understanding and versatility. This research aims to bridge the gap by exploring hybrid approaches that combine the strengths of both methodologies to enhance temporal knowledge extraction and evolution, particularly in the domain of underutilised crops.

2.3.9. Bridging the gap in word embedding for specialised knowledge extraction

Despite the advancements in word embedding techniques such as [word2vec](#), [CBOW](#), Skip-Gram, fastText, and GloVe, there remains a substantial gap in their application to the specialized domain of underutilised crops. While these models have proven effective in general linguistic tasks, they fall short in extracting implicit and explicit knowledge specific to agricultural contexts. For instance, existing models are adept at handling sentences within general contexts, such as predicting "jumps" from "The brown fox ____ over the lazy dog" based on surrounding words. However, they lack the capability to delve into niche agricultural texts to identify and understand specialized terminology and context, such as the scientific and vernacular names of crops, geo-location, cropping patterns, and specific soil requirements.

The core limitation lies in the models' training on vast, heterogeneous datasets like Wikipedia or Google News, which do not encompass the depth and specificity required for agricultural applications. For example, while [word2vec](#) can effectively capture the semantic relationship between words like "car" and "vehicle," or the

contextual association between "car" and "road," it struggles with agricultural terminology that requires a more nuanced understanding. This is evident in the use of similarity functions like cosine similarity, which, while effective for general word pairs, do not adequately capture the complex relationships inherent in agricultural data.

Moreover, the embedding models' inherent focus on high-frequency words within large corpora means that rare or specialized terms, such as those related to underutilised crops, are often inadequately represented. FastText attempts to address this by considering sub word information, thereby capturing morphological nuances. However, this approach still falls short when dealing with the intricate and specific knowledge required for agricultural domains. For instance, fastText's subdivision of "superpower" into sub words like "su," "up," and "pe" does not translate effectively to agricultural terms that need a holistic understanding rather than fragmented sub word analysis.

In the biomedical context, the research by Srivamsi et al. ([2023](#)) utilizing the Milvus vector engine demonstrates the potential of word embeddings for transforming high-dimensional biomedical data into numerical vectors, facilitating data analysis and ML applications. However, similar approaches have not been sufficiently explored or adapted for agricultural data concerning underutilised crops. The challenges of accurately capturing crop characteristics, such as soil requirements, remain unaddressed by current embedding models.

This research aims to bridge this gap by leveraging word embeddings to perform both implicit and explicit knowledge extraction specifically tailored to underutilised crops. By refining embedding models to accurately identify scientific names, vernacular names, geo-locations, cropping patterns, and soil requirements, this work seeks to go beyond the limitations of existing models. This involves developing specialized corpora and embedding techniques that can encapsulate the unique linguistic and contextual characteristics of agricultural texts. Such an approach not only enhances the accuracy of knowledge extraction but also supports better agricultural practices and sustainability by providing a deeper understanding of underutilised crops, which current word embedding methodologies fail to achieve.

2.3.10. Introduction to Detecting cropping patterns

Given the pivotal role of agricultural biodiversity, underutilised species can significantly contribute to diversifying food sources and addressing climate variability (Chivenge et al., [2015](#)). The surge in biodiversity research has notably enhanced the creation of a consolidated data repository and knowledge base (Batista-Navarro et al., [2016](#)). By mapping the specific regions and sites where these crops are farmed, a global information exchange network can be swiftly established, benefiting scientists, farmers, and policymakers. Such a network can facilitate the documentation of local agricultural methodologies and value chains for these crops, promoting their broader acceptance and integration into global agricultural diversification strategies (Van et al., 2012).

The ongoing research and adoption of underutilised crops is paramount. It holds the potential to alleviate global challenges such as climate change, nutritional deficiencies, food security, and the over-reliance of developing nations on imported crops (Ebert et al., [2014](#)). Furthermore, this research aligns with the United Nations Sustainable Development Goals, specifically No Poverty, Zero Hunger, and Climate Action (Griggs et al., [2013](#)).

While comprehensive data exists for primary crops like rice, wheat, maize, and soybean, thanks to dedicated research institutions like IRRI for rice (IRRI, [2014](#)) and CIMMYT for maize (Pingali et al., [2001](#)), underutilised crops often lack such centralised research hubs. This absence results in a scarcity of organised data. A thorough investigation into underutilised crops necessitates the identification of farming regions and cultivation patterns, whether for consumption, fodder, or commercial products. Such achievement can unveil existing value chains and engaged communities.

A preliminary objective of this study was to identify the various naming conventions for the target crop. An experiment was undertaken to collate and validate all conceivable names for the crop, subsequently utilising these terms for data extraction from search engines such as Google Search engine. It's noteworthy that a single crop can possess several varieties. For instance, the apple has over 7500 varieties (Ebert et al., [2014](#)). Intriguingly, the same apple variety might have distinct names across languages or even within the same language based on geographical locations. Terms such as varieties, local names, synonyms, and landrace names can

all refer to specific crops (Villa et al., [2005](#)). While major crops like rice and maize have well-documented names, and many of their varieties have genetic sequences archived in gene banks (Sherry et al., [2001](#)), most underutilised crops lack such comprehensive documentation. Their varieties, landrace and cultivar names and their cropping patterns are not well documented. Cropping patterns denote regions where underutilised crops might be planted or where there's an increasing interest in studying or using these crops.

Addressing this gap necessitates a specialised exploration of underutilised crop nomenclatures. In this study, we utilised big data analytics to capture global interest in particular underutilised crops using global search engines such as Google. Through the development of a systematic approach to retrieve search data from the Google search engine and establishing a processing pipeline, the data is converted into insights and information beneficial for researchers. This information was subsequently integrated into the knowledge repository for underutilised crops.

Web crawling and data mining techniques, effective in discerning societal patterns (Trevisan, [2014](#)), have also found utility in the biological and agricultural domains. For instance, Van der Velde et al. ([2012](#)) employed search volume data to deduce crop-specific planting and harvesting details, integrating climate and soil data. Similarly, search engine data has been instrumental in healthcare for forecasting trends and identifying disease epicentres (Bardak et al., [2015](#); Carneiro et al., [2009](#)).

This research aims to discern potential cropping patterns of underutilised species through online search engine data analysis, with the goal of categorising data by location and temporality, thereby unveiling potential crop utilisation patterns and tracking interest in underutilised crops over time.

2.3.11. Introduction to information retrieval in underutilised crops

Agricultural data, similar to data in various fields, often lacks uniformity and centralization. Numerous global and local entities invest years in research, producing vast amounts of both unstructured and semi-structured data. Structuring and centralising this data facilitate large-scale processing and informed decision-making. However, the emphasis on data centralization has decreased due to the rapid growth of data generation. Modern advancements in both hardware and software now allow for real-time processing of existing and streaming data. The experiment discussed in

this chapter chronicles the transition from text mining methodologies to [NIR](#) within the realm of agricultural data science.

Wikipedia, being an open platform, doesn't restrict data contribution, making it vast and diverse. Traditional text mining techniques might struggle with such extensive and intricate unstructured datasets. However, recent advancements in Artificial Intelligence ([AI](#)), coupled with improvements in computational capabilities, have equipped researchers with powerful analytical and visualisation tools, previously deemed unattainable.

In this research, (Mitara et al., [2017](#)) utilised [NIR](#) to process the Wikipedia corpus and derive agricultural insights. [Word2vec](#) and Fasttext (Joulin et al., [2016a](#)), innovative neural network methodologies developed by Google Research and Facebook Research respectively, were employed for text analysis. This experiment integrated a trained and fine-tuned version of these tools with the CropBASE knowledge system, analysing the 3.4 billion-word Wikipedia corpus in relation to the knowledge base. The primary objective was to find the connections between standard English crop names and various country names. Both [word2vec](#) and Fasttext convert textual corpora into vector spaces, and the resultant vectors can gauge word relationships. The findings revealed significant associations between crop names and countries, consistent with expert knowledge. These results can serve as a preliminary search tool for researchers navigating extensive text corpora.

2.3.12. Introduction to effective NIR in crop soil requirement identifications

This research focuses on the study of agriculture and food datasets. Most of agriculture and food research centres focus on the main 25 crops which feed more than 90% of calories consumed by the human population. Global knowledge system for underutilised crops is ongoing research conducted in [CFF](#) Research Center (Gregory et al., [2019](#)). The system provides a relational database and NoSQL database to introduce crops by 4000+ variables and 44 metadata for each variable. The variables are defined in three dimensions, taxonomical, geographical and data sources (Nizar et al., [2021](#)). The system can store multiple values for the same variable based on the crop variety, reference of the value and geographical location where the data is collected from. Several methods are used to populate the database such as the integration of international databases on climate, soil and crops.

Additional extensive manual information extraction is performed from the scientific literature. Value chain data “CONNECT and ASSESSCROP”, are comprehensive experiments conducted through the value chains for crops to build a relational database to trace information about crops from germplasm, seed inventory, plot and crop management, harvesting and postharvest process, product manufacturing, nutrient sampling and testing, market and consumers ([CFF Research](#), 2019).

Transforming textual information into structured formats, like datasets, often demands substantial time and effort. Consider the task of gathering data on underutilised crops from academic sources; it can be labour-intensive. Advanced techniques such as [NIR](#) could be useful if combined with a human-in-the-loop process. In the task of search engines and social media recommendation systems (Baltrunas et al., [2011](#)), [NIR](#) is used. However, for highly sensitive and regulated scientific studies like medical, legal and agricultural sciences [NIR](#) applications are restricted due to the decline of its accuracy (Onal et al., [2018](#)).

In this experiment we opt to use word embeddings which are numerical representations of words that capture their semantic meanings in vector spaces. These representations have gained considerable attention in academic research due to their potential in various applications. Prominent companies like Google and Facebook employ word embeddings to enhance functionalities such as search and recommendation systems. Several techniques, including [word2vec](#), [doc2vec](#), [GloVe](#), [BERT](#), and [RoBERTa](#) (Liu et al., [2019](#)), have been introduced in the literature. These methods determine the meaning of a word based on its contextual words and translate this textual information into numerical vectors, enabling mathematical operations.

This study seeks to streamline the data collection process, aiming to aid the development of specialised decision support systems. As a testament to the efficacy of [NIR](#), we present an approach to discern the soil requirement for underutilised crops. Our [word2vec](#) model, trained on [72GB](#) of Wikipedia text encompassing 178 languages and 8.1 billion words, was employed to predict the soil classification needs of 2,201 crops. This prediction was subsequently validated using The Food and Agriculture Organization ([FAO](#)) ECOCROP data (Ramirez-Villegas et al., [2013](#)).

2.4. Hyperparameter optimisation for word embeddings

2.4.1. Introduction

The aim of this experiment is to develop a high-quality text classification model. This model is intended to sift through the vast collection of 2.3 million documents from ArXiv before testing the data in the Temporal Knowledge Evolution experiment. By filtering the data, we ensure that the temporal knowledge evolution vector space is provided with high quality embeddings. This curated dataset will then serve as the foundation for training the vector space. The efficacy of the filtration process will be evident in the quality of the timeline produced by the vector spaces within the temporal knowledge evolution process.

The study primarily focuses on optimising hyperparameters to produce high-quality word embeddings using the [word2vec](#) model. The experimental design involves systematically testing various hyperparameter combinations to achieve an optimal word embedding vector space. This vector space is subsequently used for classifying Amazon Customer Reviews into predefined review rankings. For the classification task, various [DNN](#) architectures are employed in conjunction with the word embeddings to achieve the highest classification accuracy. The central component of these networks is the embedding layer, which integrates the word embeddings. These embeddings are then converted into an embedding matrix suitable for the Keras embedding layer. An open-source library, Keras, provides a programmatic interface for Artificial Neural Networks ([ANNs](#)) specifically for TensorFlow library. Keras can be considered as a high-level neural networks Application Programming Interface ([API](#)) as it facilitates an easy and fast deployments for both [RNN](#) and [CNN](#), and runs seamlessly on both [CPU](#) and Graphics Processing Unit ([GPU](#)). Importantly, this layer is designated as non-trainable to ensure that Keras does not alter the weights of the embedding layer, thereby maintaining an impartial evaluation of the embedding quality. The Amazon Customer Review dataset was chosen for evaluating Deep Neural Network classification models because it contains a wide range of text data from various product categories and customer opinions. This diversity is ideal for testing and improving text classification algorithms. The dataset helps us benchmark performance, optimize hyperparameters, and ensure model generalization, which is vital for applying these techniques to agricultural datasets. Insights gained can be transferred to classify texts from sources like Wikipedia or

scientific publications on crops. Using this dataset, we build a strong foundation in text classification, supporting our goals in agricultural data science, such as detecting cropping patterns and tracking scientific knowledge evolution.

2.4.2. Importance of embeddings hyperparameter optimisation

In the realm of textual representation, [word2vec](#) emerges as a principal technique for generating vector spaces from textual datasets. Its effectiveness largely hinges on the calibration of several hyperparameters. One of the most salient among these is the dimensionality of word embeddings. Striking the right balance is essential: dimensions that are too small might omit crucial semantic details, whereas exceedingly large dimensions present computational challenges. While (Levy et al., [2014](#)) vouched for 300 dimensions, Zhang et al. ([2016](#)) proposed a modest 128 dimensions. Another pivotal hyperparameter is the window size, which establishes the contextual boundaries for each word in its vector space. Although larger contexts can encapsulate a broader semantic range, they also risk incorporating noise from unrelated terms. The learning rate, which modulates the pace of vector space adjustments during training, is another hyperparameter of note. While higher rates can accelerate model convergence, they may inadvertently cause the model to exceed its optimal configuration. In their exploration, Zhang et al. (2019) pinpointed a learning rate of 0.05 as optimal for enhancing [word2vec](#)'s performance. Other parameters, such as the number of negative samples during training, the subsampling rate, and the word inclusion frequency threshold, also bear significant weight. Specifically, Tissier et al. ([2020](#)) documented improvements in sentiment analysis using [word2vec](#) with a subsampling rate of 10^{-4} and a frequency threshold set at 20. Given these intricacies, researchers are tasked with the meticulous fine-tuning of these hyperparameters, tailored to their specific datasets and research objectives.

2.4.3. From context to performance: decoding word2vec hyperparameters

The window size parameter holds a crucial role in the framework of the [word2vec](#) algorithm. It determines the range of neighbouring words to be considered within the computational context, spanning both the left and right sides of the focal word. As depicted in Figure 4, the window size defines the extent of words taken into account

when centred on the target word "jumps." For instance, when the window size is set to a smaller value, such as "1," the likelihood increases significantly for the term "brown" to appear in proximity to the occurrence of "fox." On the other hand, modifying the window size to "2" introduces a probability of around 50% for the word "brown" to emerge adjacent to "jumps." It's worth emphasising that the presented probabilistic values are illustrative and apply within the context of a single sentence as described in the original paper of Mikolov et al. ([2013a](#)). In reality, as the analysis extends to a broader collection of sentences used to compute spatial vectors, these probabilities will undergo adjustments accordingly.

Increasing the window size, as demonstrated through an illustrative example with a value like "fifty," effectively broadens the contextual scope around the target term. This expansion contributes to a more comprehensive understanding of the attributes associated with the focal terms, allowing for the incorporation of a wider array of entities. Such an approach is particularly beneficial when word features are infrequently positioned closely to the target word. Conversely, enlarging the window size may lead to reduced interchangeability of the target term due to the incorporation of a broader context. On the other hand, a reduction in window size can enhance the interchangeability of the target term by focusing on immediate neighbouring words. It is important to note that the impact of window size on interchangeability and analogies of connectedness is more nuanced and dependent on various factors. Caselles-Dupré et al. ([2018](#)) explored the significance of fine-tuning hyperparameters when employing [word2vec](#)'s Skip-Gram with Negative Sampling (SGNS) in recommendation systems. By conducting a series of tests on four distinct datasets, the authors discovered that adjusting hyperparameters, especially the negative sampling distribution parameter, could substantially enhance performance. The study also found that the optimal value for the negative sampling distribution parameter differed across datasets and could even be negative. In one case, the performance on the 30Music dataset rose from 11.16% to 23.75% with full optimization. These findings highlight the necessity of tailoring hyperparameters to the specific data and task at hand in recommendation systems. In summary, the window size parameter significantly affects the selection of neighbouring words for analysis in the [word2vec](#) algorithm. Its manipulation offers distinct advantages depending on the specific goals and characteristics of the linguistic data being processed.

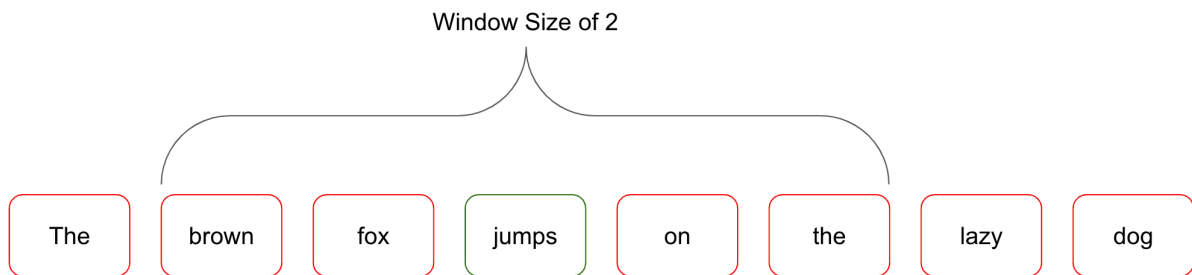


Figure 4 [Word2vec](#) window size

The dimension size parameter pertains to the number of nodes in the model's hidden layer. Typically, this parameter encompasses a range spanning from one to three hundred. As the dimension size increases, there is an associated risk of overfitting to the training data, which subsequently translates into suboptimal performance during testing. The calibration of this parameter becomes imperative when a scenario unfolds where the model exhibits heightened accuracy on the training set coupled with diminished accuracy on the testing set. This pattern serves as an indicator that the dimension size is excessively large, and mitigation of the overfitting can potentially be achieved through its reduction.

In addition to the dimension size and window size parameters, several other hyperparameters play a crucial role in the [word2vec](#) algorithm. The learning rate parameter determines the step size during gradient descent optimization. The number of training iterations influences the convergence of the model. The subsampling threshold parameter affects the probability of discarding frequent words during training. Additionally, the negative sampling parameter determines the number of negative samples drawn during training. Each of these hyperparameters contributes to shaping the performance and characteristics of the [word2vec](#) model. Proper tuning of these parameters is essential to achieve optimal outcomes in various [NLP](#) tasks.

2.4.4. Hyperparameter Impact on word2vec Downstream Tasks

The study empirically demonstrated that the optimal mix of hyperparameters is closely tied to the specific end tasks. By experimenting with various hyperparameter combinations, they aimed to generate superior vector representations suitable for cutting-edge applications like classification (Adewumi, [2020](#)). Their assertions were validated through both intrinsic and extrinsic evaluations, specifically for tasks like named entity recognition and sentiment analysis. Intriguingly, they observed that

impressive analogy scores don't always align with high F1 scores. Beyond a certain threshold, expanding the vector dimensionality diminishes its quality. Additionally, they highlighted that the sheer volume of the dataset might be less critical than its content and the excellence of the vector space representation. The F1 score is an essential metric for assessing the performance of binary classification models, as it integrates both precision and recall into a single measure. It is calculated as the harmonic mean of precision and recall, which allows for a balanced consideration of both metrics. The formula for the F1 score is expressed as: $F1\ score = 2 * (Precision * Recall) / (Precision + Recall)$. Here, precision, or positive predictive value, is the ratio of true positive predictions to the total positive predictions made by the model, defined as $Precision = True\ Positives / (True\ Positives + False\ Positives)$. Recall, also known as sensitivity or true positive rate, is the ratio of true positive predictions to the total actual positive instances, defined as $Recall = True\ Positives / (True\ Positives + False\ Negatives)$. The F1 score achieves its optimal value of 1 when both precision and recall are perfect, and its worst value of 0. This metric is particularly useful in scenarios with imbalanced class distributions, providing a more comprehensive evaluation than accuracy alone as described in the paper of (Rijsbergen, [1979](#)).

A recent study explored the optimization of hyperparameters for [word2vec](#) to produce superior word embeddings and vector representations. This investigation emphasised parameters such as the minimum word frequency, dimension of the vector, context window size, and iteration count (Yildiz, [2021](#)). The researchers presented two novel techniques that are more computationally efficient compared to traditional grid and random search methods. With a corpus of approximately 300 million words, they employed deep learning models for subsequent tasks. The objective was to categorise documents into ten distinct categories. This categorization was leveraged to assess the efficacy of the created word embeddings and vector domains. The findings indicated a 9% enhancement in classification accuracy, underscoring the significance of hyperparameter optimization in achieving top-tier word embeddings, which subsequently improves the performance of deep learning classification downstream tasks.

This research studies how to effectively utilise [word2vec](#) as a knowledge representation, which has since been applied across various fields such as recommendation systems, forecasting, and network analysis. Given the widespread

use of [word2vec](#), the research by Chamberlain et al. (2020) questions the adequacy of its default hyperparameters, especially for recommendation systems. The authors highlight the importance of hyperparameter fine-tuning, revealing that unrestricted optimization can enhance the hit rate by an average of 221% compared to default settings. However, such optimization can be computationally intensive, making it unsuitable for large-scale tasks. When constrained by runtime, the optimization still achieves a 138% average improvement in hit rate. The study also explores the feasibility of determining optimal hyperparameters from smaller data samples. By optimising hyperparameters on just 10% of the data, they still observed a 91% average improvement when applied to the complete dataset. In a practical application, when these optimised hyperparameters were implemented in Twitter's "Who To Follow" recommendation service, engagement rates increased by 15%.

Caselles-Dupré et al. (2018) studied the effects of using Skip-gram with negative sampling. Although this setting is often used in [word2vec](#) to produce superior word embeddings and vector representations for a particular dataset, the pre-configured model might not be universally effective for diverse tasks such as classification, sentiment evaluation, and recommendations. It's often essential to adjust the hyperparameters based on the dataset in question and its associated downstream activities. The research revealed that fine-tuning often overlooked hyperparameters, such as the negative sampling distribution, epoch count, subsampling metrics, and window size, can notably enhance performance in recommendation tasks. A pivotal discovery from this study is the distinct nature of hyperparameter optimization in [NLP](#) and Recommendation tasks.

2.4.5. Bridging the Gap in Hyperparameter Optimization for Specialized Knowledge Extraction

Despite significant progress in optimizing hyperparameters for word embeddings, existing literature often lacks the depth required for high-quality implicit and explicit knowledge extraction and temporal knowledge evolution, particularly in specialized domains like agriculture. While studies such as those by Levy et al. (2014) and Zhang et al. (2019) provide insights into optimizing dimensionality and learning rates for general tasks, they fall short in addressing the specific needs of extracting and evolving knowledge and the lack of specialisation like in underutilised crops. For example, the research by Tissier et al. (2020) on sentiment analysis using optimized

subsampling rates and frequency thresholds does not extend to the nuanced requirements of for specialised domain. Such as in underutilised crops where accurate identification of scientific names, vernacular names, and crop characteristics like soil requirements is crucial. These tasks demand a more sophisticated understanding of how hyperparameters influence word embeddings in capturing and evolving specialized knowledge over time. Moreover, Caselles-Dupré et al. ([2018](#)) demonstrated the significant impact of fine-tuning hyperparameters in recommendation systems, revealing substantial performance gains. However, their findings are largely confined to the realm of recommendation tasks and do not delve into the complexities of temporal knowledge evolution and as highlighted the need to optimise for specialised domain specific data such as agricultural datasets. Similarly, the work by Chamberlain et al. ([2020](#)) on optimizing hyperparameters for recommendation systems, which resulted in a dramatic improvement in hit rates, underscores the potential benefits of tailored hyperparameter tuning but does not address the specific challenges posed by any domain specific data such as agricultural contexts. Hyperparameter optimization in word embeddings often involves balancing multiple parameters such as window size, dimension size, learning rate, and negative sampling. For instance, the impact of window size on context capture, as illustrated in studies by Levy et al. ([2014](#)) and Zhang et al. ([2016](#)), is critical in understanding how contextual information is encapsulated within embeddings. Yet, this research generally overlooks how these parameters can be fine-tuned to improve the extraction of agricultural knowledge and it completely lacks the focus on the temporal dynamic characteristics of knowledge and how it changes overtime. Additionally, current approaches to hyperparameter optimization, like those proposed by Yildiz ([2021](#)) using novel techniques for more efficient hyperparameter searches, show promise in improving classification accuracy. However, these methodologies need to be adapted and expanded to consider the temporal aspects of knowledge evolution and the specific requirements of underutilised crops.

This research addresses this gap by focusing on the meticulous optimization of hyperparameters for word embeddings to enhance both implicit and explicit knowledge extraction and the temporal evolution of this knowledge. This involves developing a deeper understanding of how hyperparameters such as window size, dimensionality, learning rate, and negative sampling specifically affect the embedding quality in agricultural datasets. By tailoring these hyperparameters, this work aims to facilitate

more accurate and comprehensive knowledge representation, ensuring that the dynamic and specific needs of underutilised crops are effectively met, thereby advancing the field beyond the limitations of existing research.

2.5. Temporal knowledge evolution

2.5.1. Introduction

Scientific research is an ever-changing domain, marked by frequent technological breakthroughs and novel findings. Understanding the progression of knowledge within scientific literature is crucial to stay ahead of these shifts. This research delves into the progression of knowledge by employing word embeddings and time series analysis methodologies.

For this research, we sourced data from the arXiv repository, covering publications from 2000 to 2020. This vast textual data was refined using large scale cloud-based data processing methods. Subsequently, word embeddings were utilised to construct vector spaces for each month. Time series analysis was then applied to unveil the variations and trends in these vector spaces on a monthly basis. Through this analysis, we discerned significant events, observed the genesis and growth of fresh research themes, and assessed the influence of technological innovations and discoveries over the years.

The outcomes of our research underscore the efficacy of using temporal knowledge evolution over vector spaces to identify patterns and shifts in scientific literature. This methodology offers valuable insights for scholars aiming to comprehend knowledge evolution in their domain and for entities that financially support research or channel investments into scientific advancements.

Our study offers insights into the progression of scientific knowledge and innovation throughout time. Through the analysis of temporal vector spaces, we discerned the emergence and evolution of novel research topics. Additionally, we observed the influence of new technologies and discoveries in the discipline. This methodology can assist researchers in pinpointing significant trends and unveiling fresh research avenues in their domain. Furthermore, it can guide policymakers in their decision-making for downstream tasks.

This section delineates the methodology employed to construct a cloud-based big data pre-processing pipeline for processing the large dataset from arXiv. The process commences with the decompression of data and ends with the generation of a monthly file encompassing the scientific literature published within that month. Decompression is a critical initial step in the data pre-processing pipeline for several essential reasons. The arXiv dataset is often stored in a compressed format to save space and facilitate faster transmission, but many data

processing tools and algorithms require the data to be uncompressed for proper analysis. Direct access to raw, uncompressed files is necessary for detailed tasks such as document classification, clustering, and vector space creation, which are fundamental to the study. These tasks involve extensive manipulation of the text data, which cannot be efficiently performed on compressed files. Moreover, decompression ensures compatibility with the software tools used in the pipeline, enabling smooth execution of subsequent data processing steps and preparing the dataset in the required format and structure for complex analyses. It is imperative to undertake a document classification procedure during pre-processing. This facilitates the filtering of pertinent documents for subsequent clustering, adhering to the previously discussed methodology for document classification using the word embeddings created by the hyperparameter tuning experiment. The monthly clustering of these classified publications serves as the bedrock for the development of high-quality vector spaces. Such a task requires the alignment of vector spaces, a consequence of the creation of temporal vector spaces. When a vector space is constructed, the values of embeddings are determined based on the initial values of the hidden layer, which are typically randomised. Consequently, the embedding values of the vector space for one month may not be relative to those of other months. It's essential to align these vector spaces to ensure that the embeddings from all months are comparable, facilitating the calculation of similarity across different vector spaces. Upon the establishment of these temporal vector spaces, we embarked on time series analysis. This entailed a carefully crafted procedure to pinpoint the specific entities under investigation. To study the evolution of knowledge, two primary methodologies were adopted: event detection and trend identification pertaining to knowledge evolution.

2.5.2. Semantic shift

Semantic shift, also known as semantic change or semantic drift, embodies the linguistic phenomenon wherein the meaning of a word or phrase undergoes a transformative evolution over time due to cultural, social, historical, and technological influences. This gradual or abrupt change in meaning leads to the original semantic sense of a term diverging into new connotations. The shifts can be categorised into several types: broadening, where a word's meaning becomes more inclusive, such as "dog" expanding from a specific animal to various breeds; narrowing, which involves a more specific meaning emerging, as seen with "meat" narrowing down to animal flesh; amelioration, where a word gains a more positive connotation like "knight" progressing from servant to esteemed warrior; pejoration, imparting a negative implication, such as "knight" shifting from attendant to title; metaphorical extension,

where metaphoric usage adds new layers, like "keyboard" transitioning from a musical instrument to a typing interface; metonymy, connecting a word to something closely related, as seen with "crown" representing monarchy; and shifts in register, prompting contextual connotation changes. This phenomenon underscores language's adaptive nature to diverse communication needs within evolving cultural and contextual landscapes.

2.5.3. Dynamic word embeddings

Dynamic Word Embeddings ([DWE](#)) form a distinctive category of lexical representations crafted to encapsulate the fluid semantic nuances and affiliations of words across different temporal epochs. In contrast to their static counterparts, where words are represented by fixed vectors, dynamic word embeddings portray words as intricate functions that blend linguistic and non-linguistic contextual elements. This advanced approach enables the encapsulation of the semantic changes that words undergo over diverse temporal intervals and societal contexts. The subsequent section outlines notable scholarly contributions in the realm of Dynamic Word Embeddings. The paper by Yao et al. ([2018](#)), stands out for its creation of a unique statistical model that takes into account the time-sensitive nature of word vector representations, thereby reflecting the changing meanings of words over time. By employing a dataset from NYTimes, the authors developed a method that learns time-aware embeddings while simultaneously addressing the alignment challenge. Their experimental approach included a variety of evaluation methods for temporal word embeddings, revealing that their technique not only accurately captures word evolution but also surpasses existing methods in semantic accuracy and alignment quality. In a separate contribution, (Bamler et al. [2017](#)) designed a probabilistic model for time-stamped text data, carefully tracking the semantic paths of words by representing them within hidden trajectories in an embedding space. Additionally, Wang et al. ([2019](#)) introduced two novel post-processing methods, variance normalisation (PVN) and dynamic embedding (PDE), which were successfully integrated with well-known word embeddings techniques like [word2vec](#) and [GloVe](#), leading to improved and sophisticated word representations. In summary, this research aims to collectively explore diverse dimensions of dynamic word embeddings, including the modelling of temporally aware embeddings, the understanding of semantic evolution, and the

application of post-processing methodologies to established embedding frameworks such as [word2vec](#) and similar techniques.

Temporal and dynamic word embeddings both enhance the semantic understanding of language but serve different purposes. Temporal knowledge evolution studies how knowledge evolves, focusing on the relationships between entities and how these relationships change over time, rather than just changes in word meanings. Additionally, Temporal word embeddings are designed to capture how the meanings of words change over different time periods. This involves training embeddings on time-stamped text data and aligning them to reflect shifts in usage and meaning across various eras. They are particularly useful for historical linguistic analysis and tracking cultural changes. In contrast, dynamic word embeddings adapt to different contexts within the same period, utilizing models like [BERT](#) or GPT to generate context-specific meanings for words. This makes them effective for handling polysemy and improving performance in tasks that require nuanced language understanding, such as question answering and sentiment analysis. While temporal embeddings focus on diachronic changes and evolving knowledge relationships, dynamic embeddings address synchronic variation in language.

2.5.4. Temporal word embeddings

Temporal word embeddings or Diachronic word embeddings, constitute a specialised class of lexical representations within the domain of [NLP](#) (Kutuzov et al., [2018](#)). In contrast to conventional embeddings that centre around semantic meaning exclusively, temporal word embeddings uniquely encompass temporal attributes, effectively encapsulating the sequential occurrences of words in text over time. This aspect holds particular relevance in domains like the analysis of historical texts, exploration of linguistic trends, studies on language evolution, and the monitoring of temporal shifts in word sentiment. The methodologies for developing temporal word embeddings encompass techniques such as enriching prevailing models with temporal features, deploying dynamic embeddings that capture temporal progression by training across discrete time intervals, and leveraging contextualised models capable of implicitly grasping temporal intricacies through holistic contextual comprehension. These embeddings find utility in diverse applications such as historical linguistics, sentiment analysis, document chronology, authorship attribution, and trend analysis in extensive text datasets. Their value lies in the amalgamation of

linguistic expertise and advanced [ML](#) methodologies, which collectively offer profound insights.

Recent advancements in temporal word embeddings have continued to enhance the field's understanding and application. Gong et al. ([2020](#)) introduced a model that enriches word embeddings with temporal and spatial information, providing insights into the evolution of word meanings over time and across locations. Additionally, Liao et al. ([2021](#)) developed a dynamic contextual word embedding model that transforms pre-trained embeddings into time-specific representations using temporal neural networks, effectively capturing semantic drift. Mandt et al. ([2021](#)) proposed dynamic embeddings for temporal knowledge graphs, which can extrapolate future entities and relations, offering valuable predictions in dynamic data environments. These advancements underscore the growing sophistication of temporal word embeddings in capturing the dynamic nature of language. The work by Gong et al. ([2020](#)) highlighted that incorporating temporal and spatial dimensions into word embeddings significantly enhances the capture of socio-cultural influences on language. This model not only captures temporal changes but also adapts to location-specific variations, making it particularly useful for cultural and linguistic trend studies. Liao et al. ([2021](#)) observed that dynamic contextual embeddings are effective in modelling semantic drift, which is crucial for applications requiring an understanding of evolving word meanings, such as historical text analysis and trend detection. Meanwhile, Mandt et al. ([2021](#)) demonstrated that their method for temporal knowledge graphs allows for accurate future predictions by modelling the evolution of entities and their relationships over time. This capability is particularly advantageous in fields like predictive maintenance, recommendation systems, and dynamic content generation. These major observations illustrate the versatility and depth of temporal word embeddings in various [NLP](#) applications, reinforcing their importance in both academic research and practical implementations.

2.5.5. Distributional semantic shifts

Distributional semantic shifts encompass modifications in the significance or semantic relationships of language units across a temporal spectrum, as detected through analysis of extensive textual corpora. These shifts become apparent by contrasting the contextual arrangements of words during distinct chronological periods or situational contexts. Within the paradigm of distributional semantics, the semantic

essence of lexical items is derived from their tendencies to co-occur with other terms within textual contexts (Sommerauer et al., [2019](#)). Exploring distributional semantic shifts necessitates a comprehensive examination of alterations in interlexical relationships over time. To illustrate, take into consideration the term "mouse". In earlier times, it may have exhibited noticeable semantic proximity with "cheese" and "trap." However, in contemporary language, it could potentially display heightened collocational affinity with "computer" and "click." The origins of distributional semantic shifts are multifaceted, encompassing alterations in linguistic usage, cultural transitions, technological advancements, and sociocultural inclinations. Their scholarly investigation frequently adopts techniques rooted in computational linguistics and [NLP](#). These methods leverage the comparative analysis of word embeddings, compact vector representations signifying linguistic elements, across divergent timeframes or contexts. Quantifying vector concurrence or divergence facilitates the identification and elucidation of semantic nuances or modifications. The implications of distributional semantic shifts resonate across multiple domains, including linguistics, historical linguistics, sociolinguistics, and the digital humanities. By decoding how language mirrors societal dynamics and the adaptability of lexical connotations, these shifts illuminate the intricate evolutionary path of language over temporal spans.

Recent research has significantly advanced the understanding and management of distributional semantic shifts. One notable study introduces an adaptive prompting strategy, named SemPrompt, designed to tackle the challenges of continual learning environments where semantic changes are frequent and unpredictable. This method strikes a balance between universal and task-specific prompts by employing a two-level semantic grouping process. The macroscopic semantic assignment level ensures broad semantic categorization, while the microscopic refinement level adjusts these categories to maintain prompt relevance over time. This adaptive approach effectively manages prompts, ensuring they adapt to new tasks and evolving contexts, thereby enhancing the handling of semantic shifts in dynamic scenarios (Gupta et al., [2023](#)). Another important contribution comes from a study that presents a clustering-based vector monitoring algorithm for measuring distributional shifts in text data. By employing language model-based embeddings and k-means clustering, this method identifies regions of high data density and monitors shifts by comparing the relative frequencies of these clusters between baseline and new data. The approach utilizes metrics such as Jensen-Shannon divergence to

quantify these shifts, offering a precise measure of semantic changes over time. This technique is particularly effective for maintaining the performance and reliability of natural language processing systems in dynamic environments (Kim et al. [2023](#)).

The link between distributional semantic shifts and dynamic embedding lies in their mutual focus on the evolution of word meanings over time. While distributional semantic shifts highlight changes in the contextual relationships and meanings of words across different periods, temporal word embeddings explicitly model these changes by incorporating temporal dimensions into the embeddings themselves. Both approaches leverage extensive textual corpora to capture the dynamic nature of language, utilizing comparative analyses of word contexts over distinct timeframes. Essentially, temporal word embeddings provide a concrete methodology to quantify and visualize distributional semantic shifts, thereby offering a more structured and detailed representation of how language evolves temporally.

2.5.6. Neural latent knowledge retrieval

Apart from our utilisation of word embeddings and neural networks for explicit information extraction, there is also dedicated research focused on extracting implicit information. Researchers frequently delve into multiple publications and may arrive at conclusions that were not explicitly stated in the literature. This process significantly contributes to the accumulation of human knowledge and the discovery of novel concepts or methodologies. A research team at The University of California, with the collaboration of Lawrence Berkeley Lab and Google USA, published a paper in the journal *Nature* to demonstrate that high-quality word embeddings can capture latent knowledge in materials scientific literature (Tshitoyan et al., [2019](#)). The research team gathered 3.3 million academic abstracts from over 1,000 journals specialising in materials-related research, spanning the years 1922 to 2018. The total vocabulary size is approximately 500,000 words. The researchers used a skip-gram variation of [word2vec](#) with 200-dimensional embeddings of that target word. They also employed alternative algorithms like [GloVe](#) for a comparative analysis. Target words, including 'LiCoO₂' and 'LiMn₂O₄' (utilising one-hot encoding), served as inputs for a neural network. A singular linear hidden layer with 200 neurons (word embeddings) was utilised, resulting in analogous hidden layer weights post-training. The research adhered to specific operational standards, particularly during the phases of data collection and processing. Initially, the text was screened to exclude foreign

languages, utilising techniques like text searches and regular expressions. Articles that had metadata tags such as 'Announcement', 'BookReview', and 'News', among others, were discarded. Additionally, any abstracts that had titles with specific keywords like 'Commentary', 'Conference', or 'Correction' to name a few, were also excluded from the study. Any leading or trailing copyright details present in the abstracts were removed, as were introductory words like 'Abstract:'. To further streamline the data, binary classifiers were deployed to pinpoint abstracts that focused on inorganic materials. The tool ChemDataExtractor was then utilised to process 1.5 million abstracts. Regular expressions helped in recognizing valid chemical formulae, with the example given being that NiFe and Fe₅₀Ni₅₀ are equivalent. All text was transformed to lowercase and had its accents removed, with the sole exception being chemical formulae. Lastly, to ensure uniformity, all numerical values in the text were replaced with a distinct token. The original corpus used by the researchers contains 3.3 million abstracts. The researchers decided only to use the abstracts that can be considered relevant to the topic. Accordingly, they designed and implemented their simple classifier. They trained a binary classifier (Logistic regression) based on labelled abstracts 'relevant' or "not relevant". Annotations were manually carried out on 1,094 abstracts chosen at random. Out of these, 588 were marked as 'relevant' while 494 were deemed 'not relevant'. These labelled abstracts were employed to develop a Logistic Regression classifier where each document was characterised using a [TF-IDF](#) vector. The team used the [word2vec](#) functionality in Gensim, introducing a few tweaks. They observed that the skip-gram method combined with negative sampling loss ($n = 15$) yielded the most optimal results. The vocabulary was made up of words that appeared more than five times. All standardised chemical formulas were incorporated, regardless of how often they were mentioned. Keywords, such as "thermoelectric," are represented in a manner analogous to material formulas like "Bi₂Te₃." This resemblance in representation is exemplified by the high cosine similarity between the embedding of thermo-electric materials and the term "thermoelectric." This similarity stems from the fact that the text corpus contains documented behaviour of this material as thermoelectric. Interestingly, several materials with high cosine similarities to "thermoelectric" have never been explicitly mentioned alongside this term or any other definitive indicators of thermoelectric materials in the abstracts. This observation suggests the potential discovery of new thermoelectric materials. The researchers performed a deep statistical analysis of the

resultant word embeddings. They sourced data from 18 distinct 'historical' text corpora, focusing solely on abstracts penned between 2001 and 2018. This data enabled them to forecast the top 50 thermoelectrics likely to be referenced in upcoming literature. The revelations were indeed remarkable. For instance, CuGaTe₂, now recognized as one of the leading thermoelectrics of the modern day, would have been earmarked as a top-five contender a full four years prior to its official publication in 2012. Similarly, compounds like ReS₂ and CdIn₂Te₄ only gained recognition in scholarly circles as potent thermoelectrics around 8–9 years after they would've initially emerged in the algorithm's top-five roster. In contrast, substances like HgZnTe and SmInO₃, laden with either costly elements like Sm and In or hazardous ones like Hg, remain unexplored. A standout aspect of this study lies in the model's interpretability. For instance, three materials, including CsAgGa₂Se₄ from the top-five predictions, share ties with pivotal context words associated with 'thermoelectric', such as 'chalcogenide' and 'band gap'. This showcases that word embeddings can transcend mere compositional or structural parallels. They hold the promise to unveil concealed insights, often eluding human scientists.

Recent research has explored innovative methods for neural latent knowledge retrieval. One paper, "Discovering Latent Knowledge in Language Models Without Supervision" Long et al. ([2022](#)), introduces an unsupervised framework to align model predictions with reality by leveraging the consistency structure of truth, using various models to uncover hidden insights in datasets. Another study, "Retrieval-Augmented Neural Response Generation Using Logical Reasoning and Relevance Scoring" Walker et al. ([2023](#)), enhances dialogue systems through a combination of neural sequence labelling, probabilistic logic programming, and commonsense reasoning to extract and link relevant knowledge from a knowledge base, thus generating contextually accurate responses. These advancements demonstrate how latent knowledge can be effectively retrieved and utilised, similar to previous work using word embeddings in materials science literature.

Munková et al. ([2014](#)) conducted a study to investigate the impact of stop word removal on data analysis outcomes. They found that the removal of stop words did not significantly improve the quantity or quality of extracted rules from corpora. Most of the rules (comprising 64% of the total) were extracted from corpora, suggesting that stop word removal has no statistically significant effect on the quality or quantity of the extracted rules (Baradad et al., [2015](#)). In contrast, Bardada et al. (2015) explored the

influence of stop word removal on the semantic clustering of Twitter data. They experimented with different stop word lists: a standard list, a list derived from the most frequent words in the corpus, and a list generated from the least frequent words in the corpus. Surprisingly, the choice of stop word list significantly affected the results of semantic clustering. These findings led the authors to emphasise the importance of careful selection of the stop word list when conducting semantic clustering of Twitter data.

The research drew inspiration from the principles of latent knowledge retrieval, particularly in data processing and the nuances of temporal element construction. This methodology, as demonstrated by Tshitoyan et al. ([2019](#)) in their study on capturing latent knowledge in materials scientific literature, provided valuable insights. Elements from their approach, including sophisticated data processing techniques and the detailed incorporation of temporal dynamics, were utilised to inform the experimental design. The nuances of the temporal element, such as the longitudinal analysis of data spanning multiple decades and the prediction of future trends based on historical data, were especially influential. However, it is crucial to note that the experimental design was not replicated or copied. Instead, the findings from their study were used as a foundation to develop and structure independent experiments, tailored to address specific research questions. Additionally, while Tshitoyan's research provided valuable insights, it was limited in the sense that it did not construct a comprehensive temporal framework to study the evolution of knowledge. Their approach relied on clustering specific periods to predict chemical compounds as candidates, without addressing how knowledge evolves over time through the construction of temporal knowledge. This aspect will be elaborated upon in this research, which aims to fill this gap by constructing a detailed temporal framework to study the evolution of knowledge.

2.5.7. Bridging the Gap in Temporal Knowledge Evolution through Dynamic Word Embeddings

Despite the significant advancements in understanding semantic shifts and developing dynamic word embeddings, existing research has largely overlooked the temporal evolution of knowledge, especially on a large scale. Semantic shift studies, such as those by Yao et al. ([2018](#)) and Bamler et al. ([2017](#)), have primarily focused on capturing changes in word meanings over time through dynamic embeddings. While these approaches successfully reflect the changing meanings of words in general contexts, they do not study the knowledge evolution and how entities change its relationships over time. For instance, while Yao et al. ([2018](#)) employed a statistical model to capture time-aware embeddings using a dataset from NYTimes, and Bamler et al. ([2017](#)) designed a probabilistic model for time-stamped text data, these models primarily address semantic changes in a broad sense without focusing on the temporal evolution of large scale knowledge. Similarly, the work of Wang et al. ([2019](#)) on variance normalization and dynamic embedding post-processing techniques enhances word representation but does not extend to tracking the evolution of specialized knowledge over time. Furthermore, research into temporal word embeddings, such as that by Kutuzov et al. ([2018](#)), highlights the importance of capturing temporal attributes to understand historical texts and linguistic trends. However, these studies tend to concentrate on general linguistic applications, like sentiment analysis and document chronology, rather than on the temporal element of the knowledge evolution itself. The examination of distributional semantic shifts, as discussed by Sommerauer et al. ([2019](#)), provides valuable insights into changes in word relationships over time, yet it does not address how these shifts impact the understanding of specialized knowledge domains. Neural latent knowledge retrieval research, exemplified by Tshitoyan et al. ([2019](#)), demonstrates the potential of high-quality word embeddings to uncover latent knowledge in materials science literature. This study showcases the ability to predict future trends in materials research by analysing historical data. However, such methodologies have not been applied in a specific domain of chemical compounds and focus only on knowledge discovery and didn't address how the knowledge evolve over time and how the temporal elements of knowledge can be mapped.

This research addresses this gap by focusing on the temporal element of knowledge and its evolution over time, specifically for large scale data set like ArXiv dataset. By using word embeddings to map the temporal knowledge evolution, this work aims to capture how knowledge develop and evolve over time. This novel approach not only enhances our understanding of knowledge shifts in specialized domains but also provides a dynamic tool for tracking and predicting future trends in scientific practices. This contribution is pivotal as it introduces a temporal dimension to knowledge extraction and evolution, which has been largely neglected in existing literature, thereby offering a new avenue for research and application in large scale knowledge dataset.

2.5.8. Conclusion

In this chapter, we explore different word embeddings techniques and their applications in knowledge representation. We discussed various models, including [word2vec](#), [GloVe](#), fasttext, among others. Tracing back to the roots of knowledge representation, we explored the evolution from one-hot representations and co-occurrence matrices to advanced word embeddings and distributed representations. Beyond word embeddings, it was essential to explore document embeddings, which allow networks to represent larger chunks of text, be it a sentence or an entire document. This is vital for numerous applications, such as document classification, sentence similarity, question answering, and sentiment analysis. With word embeddings as the central concept, extracting information from textual sources becomes feasible. A prominent theme in the literature is the consistent and effective use of word embeddings by researchers to extract information. This is attributed to the capability of these embeddings to encode semantic relationships within their values. A deep understanding of the inner workings of [word2vec](#), especially its training via bag-of-words or skip-gram architectures, sheds light on the influence of hyperparameters on the quality of the resulting embeddings. The sliding window technique used in creating these embeddings equips the network with a rich semantic representation. This representation accentuates connections between entities based on relatedness or interchangeability. These two principles form the bedrock of information extraction from corpora. For instance, the principle of interchangeability can reveal alternative names for crops across languages or even scientific crop names. On the other hand, the relatedness principle can aid in determining soil

requirements for specific crops, drawing upon the connections between crop names and soil types in literature. Experimental observations underscore the significance of the hyperparameters in word embedding models. For example, the window size in [word2vec](#) is pivotal in crafting embeddings vital for tasks such as [IR](#) or document classification. A review of the literature affirms the crucial role of hyperparameters in generating superior word embeddings that, in turn, influence the overall accuracy of downstream tasks.

The study of semantic drift—how word meanings change over time—is a prominent topic in [NLP](#) and linguistics. A key challenge for researchers is aligning word embeddings from different time periods to track these changes. The semantic shift helps in not only understanding language evolution but also in revealing insights about broader societal and cultural shifts. The practical importance of recognizing semantic changes spans various applications, from improving search engines to constructing knowledge graphs. As language evolves due to various factors, the way we perceive and share information also transforms, making the study of semantic shifts crucial in both academic and practical contexts. A noteworthy study by Berkeley and Google labs tackled another significant topic of latent knowledge extraction. By incorporating a temporal dimension, the researchers could discern latent knowledge, enabling the extraction of implicit information not overtly present in the textual sources. This mirrors the human ability to infer and formulate novel ideas. While neural networks capture information by training on vast corpora, enabling them to encode knowledge from various sources, this amplifies their potential to reveal implicit knowledge.

This chapter sets the stage for the experiments to be discussed in the subsequent chapters. We'll delve into the overarching methodology crafted for this study, closely inspecting each individual contribution and the corresponding experimental procedures.

Chapter 3

Datasets

3.1. Introduction

In this chapter, we will delineate the dataset used in our experiments for this study. The primary objective of this research is to uncover effective methods for knowledge extraction from a large corpus of data. Such knowledge can either be explicit, reflecting the direct intentions of the authors, or implicit, discernible only through a comprehensive examination of a broader set of literature. Moreover, we aim to explore the evolution of this knowledge over time, which enhances our capacity to identify scientific trends and events that can guide research direction. The initial focus revolved around extracting information about crops. A pivotal area of research in crop science involved developing a knowledge base for underutilised crops, employing both manual and automated techniques. Primarily, we examined how to find cropping patterns using big data analytics. By analysing vast amounts of online search activities performed by the general public, we sought to grasp global interests in underutilised crops and the consequent cropping patterns.

Financial constraints often limit research institutions, which in turn affects data collection processes that require significant human forces. Researchers frequently need extensive periods to study literature, extract valuable information, and update the knowledge base. Hence, the urgency for AI-assisted methods became paramount. This influenced the research into techniques for efficiently extracting data from scientific literature. Consequently, we explored word embeddings to identify scientific and local names of underutilised crops in various languages—a notably difficult task, especially for underutilised crops, given that their associated information is often dispersed and not systematically organised. This research highlighted the limitations of traditional text mining methods when faced with vast unstructured datasets like Wikipedia. Transitioning to [NIR](#) techniques, specifically [word2vec](#) and Fasttext, proved more adept at analysing such extensive corpora. Despite initial challenges, updated training on newer Wikipedia editions enhanced the accuracy and vocabulary of the models. These advancements underscore the importance of embracing modern methodologies to extract meaningful insights from large-scale data.

The exploration into efficient [NIR](#) emphasised the crucial role of hyperparameters and their effects on tasks like identifying interchangeable entities, identifying relationships between entities, document categorization, and particularly, the generation of high-quality embeddings for tracking knowledge evolution over time. A segment of this research is dedicated to examining the importance of word embedding hyperparameters and their substantial influence on tasks, such as document categorization.

Scientific committees often rely on both qualitative and quantitative data to inform their funding decisions. Gathering this data can be daunting in itself. Yet, quantifying scientific trends and predicting major events in the field presents even more formidable challenges. These predictions can be influenced by individual biases and are frequently based on a scientist's intuition. The final experiment in this study delves into methods to quantify temporal knowledge evolution, aiming to provide a robust foundation for studying scientific events and predicting trends. This not only supplements or refines the intuition of scientists but also offers pivotal data points that can guide decision-making in terms of research directions.

The preliminary results of research prompted to broaden the research scope beyond underutilised crops to encompass wider scientific fields like physics and computer science. The earlier experiments in this research utilised the Wikipedia dataset to extract both implicit and explicit knowledge related to crop science. Transitioning from a general dataset like Wikipedia (Holman, [2008](#)) to the specialised arXiv dataset, which focuses on scientific literature, demonstrated significant value due to the superior quality of curated articles in comparison to Wikipedia. The arXiv dataset encompasses information from a broad spectrum of science, mathematics, and physics. Conducting extensive experiments on temporal knowledge evolution to analyse scientific events, chart scientific trends, and forecast their trajectories has shown to offer advantages in understanding the wider scientific landscape.

In understanding the multifaceted nature of knowledge, it is essential to distinguish between its explicit and implicit forms, as well as to explore how it evolves over time. Knowledge encompasses information, understanding, and skills acquired through experience or education, blending both factual data and complex insights. Explicit knowledge is straightforward, easily documented, and communicated, typically found in written documents, manuals, and databases. In contrast, implicit knowledge

is more elusive, residing within individuals' minds as intuitions, insights, and experiences that are not readily articulated. This research investigates these definitions and methods to study them and examines the dynamic process of knowledge evolution, which refers to the way knowledge changes and develops over time. By integrating new data and perspectives, understanding shifts, new insights emerge, and existing information is re-interpreted, highlighting the importance of tracking and predicting future trends in specialized domains.

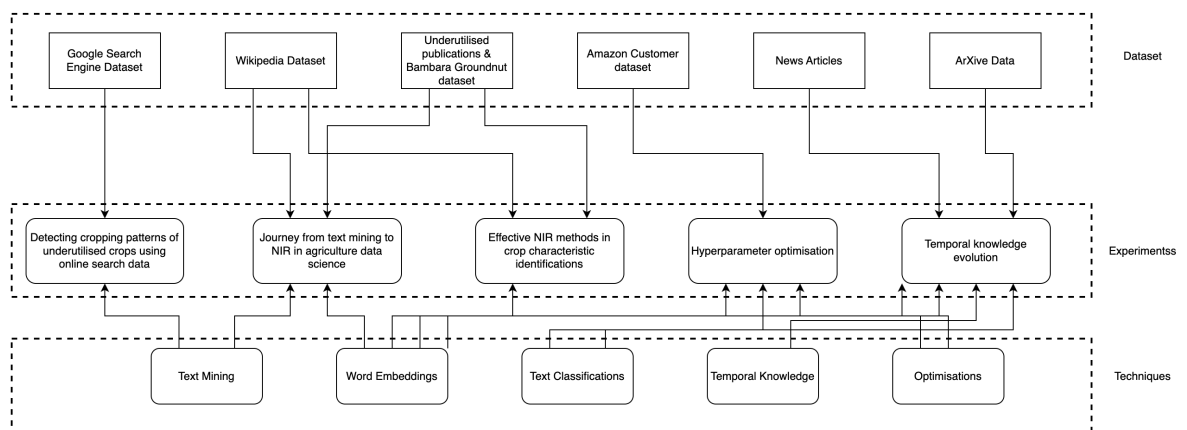


Figure 5 Illustrations of the interaction between different datasets, techniques and experiments presented in this research

This research leverages diverse datasets and sophisticated techniques to explore agricultural data science, focusing on underutilised crops. Figure 5 illustrates the interaction between different datasets, techniques and experiments presented in this research. The utilised datasets include the Google Search Engine Dataset, Wikipedia Dataset, Underutilised Publications & [BG](#) Dataset, Amazon Customer Dataset, News Articles, and arXiv Data, each serving a specific purpose. The experiments are categorized into five main areas: detecting cropping patterns using online search data, exploring the efficacy of [NIR](#) methods in text mining, accurately identifying crop characteristics through high-quality word embeddings, conducting hyperparameter optimization for document classification, and examining the temporal evolution of scientific knowledge using the arXiv dataset. Techniques employed include text mining, word embeddings, text classification, temporal knowledge evolution, and optimization. Through these methods, this research aims to bridge knowledge gaps and provide valuable insights into the evolving landscape of agricultural data science and beyond. The subsequent sections will delve deeper into the methodologies and

findings of each experiment, highlighting the significance of our contributions to the field.

3.2. Datasets

3.2.1. Introduction

In this section, the datasets used for different experiments are discussed. The multilingual and English Wikipedia datasets form the basis for the [IR](#) experiments. A specialised collection of literature on underutilised crops was also curated for this purpose. The arXiv dataset played a role in the study of temporal knowledge evolution. In addition to these large datasets, smaller ones like the Amazon User Reviews dataset and news datasets were employed in initial experiments before shifting focus to the broader arXiv dataset.

To prepare textual datasets for embedding layers, several preprocessing techniques are employed to ensure the data is in an optimal format for model training. Initially, tokenization splits text into individual words or tokens. This is followed by converting all characters to lowercase to maintain consistency. Stop words, which are common words like “and” or “the,” are removed to focus on more meaningful content. Stemming and lemmatization reduce words to their root forms, aiding in treating different forms of a word as a single entity. Punctuation and numeric characters are removed to simplify the text, and extra whitespace is cleared to clean up the dataset. Ensuring a consistent encoding format, such as UTF-8, prevents errors related to special characters. Spell check and correction help standardize words affected by typos, and special characters are handled appropriately, especially in data from sources like social media. These preprocessing steps transform textual data into a standardized and simplified form, enhancing the performance of embedding models by providing clean and consistent data for learning accurate word representations. The preprocessing techniques can vary based on the unique characteristics of each dataset. While common preprocessing techniques provide a foundational approach, each dataset demands tailored preprocessing to address its unique requirements and ensure the most effective preparation for embedding layers.

3.2.2. Wikipedia multilingual corpus

Wikipedia, the world's most extensive online encyclopaedia, has evolved into a treasure trove for researchers and the general populace. Established in 2001, it has since burgeoned into a vast knowledge reservoir, with articles spanning numerous languages and subjects. The structured nature of Wikipedia further amplifies its appeal. Articles are categorised, interlinked with related topics, and supplemented with a wealth of metadata, references, and external links. Such a structured layout simplifies data extraction, analysis, and interpretation. Researchers have tapped into Wikipedia's dataset for diverse purposes. For instance, in the realm of [NLP](#), it has been a cornerstone for training language models, semantic analysis, and algorithm development for text summarization. Social scientists have delved into Wikipedia to decode collaborative behaviour, information dissemination patterns, and inherent biases in crowd-sourced content. Its applications also span [ML](#), knowledge graph construction, and [IR](#). Given the expansive potential of the Wikipedia dataset, it's unsurprising that a plethora of academic papers have emerged, discussing its multifaceted applications and dimensions. One such notable research (Burns et al., [2023](#)), introduces the Wikipedia Webpage (WikiWeb2M) suite, a comprehensive dataset retaining the entirety of images, text, and structural data within a page. The dataset's versatility is evident as it can be harnessed for tasks ranging from page description generation to contextual image captioning. Another seminal work by Wang et al. ([2021](#)), unveils a dataset that pairs Wikipedia articles with a corresponding knowledge graph, aiming to bolster research in conditional text generation and graph representation learning. The dataset, named WikiGraphs, synergizes each Wikipedia article from the renowned WikiText-103 benchmark with a subgraph extracted from the Freebase knowledge graph. The research not only introduces the dataset but also showcases baseline results for a spectrum of tasks employing this dataset.

In this study, we employed the Wikipedia dataset, encompassing all 178 languages available during the experimentation phase, to execute [NIR](#). The primary objective was to identify both the scientific and local names in different languages of underutilised crops. We also explored the effectiveness of identifying the relationship between countries and crops. The findings were then cross-referenced with a pre-gathered list of crop names and geo-cropping pattern from the [CFF](#) dataset,

specifically focusing on the Bambara groundnut ([BG](#)). The subsequent section delves into the [BG](#) dataset, providing illustrative examples.

3.2.3. BG dataset

The [CFF](#) Research Center is a specialised research institution focused on underexplored crops. One of its primary subjects of study is the [BG](#) (Mayes et al., [2012](#)). To foster collaboration and knowledge-sharing, [CFF](#) initiated the BamYield network, connecting various research entities working on the [BG](#). This study employs the [BG](#) as a case to demonstrate the use of the Google AdWords tool to collect global public interest in discerning cropping trends by analysing global public interest. This is achieved by examining search engine queries related to different names of the [BG](#). Data from the [CFF](#) BamYield initiative will be instrumental in corroborating the findings of this study.

The preliminary phase of the research entailed the identification of various names for the [BG](#), including its scientific nomenclature, synonyms, varieties, landraces, cultivars, and vernacular names across languages. This task engaged multiple experts, drew from academic literature, and integrated information from both local and global databases. Notably, the databases consulted in this study include (USDA-GRIN NRCS, [2010](#)), (GBIF, [2016](#)), (Caracciolo et al., [2013](#)), (South Africa, [2016](#)) and [CFF](#)'s CropBASE (Jahanshiri et al., [2015](#)). It was observed that the initial compilation of [BG](#) names, sourced from global databases and literature, contained several inaccuracies. These errors can be attributed to the historically limited attention given to such crops. A carefully designed manual review was undertaken to refine and authenticate this list. The finalised names for the [BG](#) are presented in Table 1. The manual review was performed by the expert team within [CFF](#) who are responsible for the CropBASE database.

Table 1: [BG](#) selected names

Bambara groundnut names	
<i>Group 1</i>	<i>Group 2</i>
Bambara groundnut	kacang bogor
Bambara Bean	ถั่วหนัง
Bambara-bean	バンバラマメ
Bambara Erdnuss	Feijão jugo
Bambara-Erdnuss	班巴拉花生
Bambarra groundnut	Congo goober [6]
Guisante bambarra	Erderbse
Madagascar groundnut	jinguba-de-Cabambe
Jugo bean	비그나서브테라네아
vigna podzemní	Pisello di terra
Voandzeia subterranean	Pois arachide
Vigna subterranean	Voandzou
Glycine subterranean	kacang poi
Maní de bambarra	бамбарский земляной орех

3.2.4. Amazon customer review

In this research, the Amazon Customer Review dataset was utilised to train and assess the efficacy of several [DNN](#) classification models. These classifiers were fed with high-quality embeddings derived from the hyperparameter tuning experiments using Gensim's [word2vec](#). Amazon provides a distinctive dataset known as Amazon Customer Reviews (also referred to as Product Reviews). This dataset has been collected over a span of two decades, starting from 1995. It encompasses reviews from millions of Amazon customers, culminating in hundreds of millions of individual reviews. These reviews encapsulate customer feedback regarding their experiences with the products and their transactions on the amazon.com platform. The Amazon Customer Reviews dataset has been frequently employed in academic research, especially in the domain of [NLP](#). Moreover, it has been a valuable resource in the fields of [IR](#) and [ML](#). The dataset comprises over 130 million reviews and is hosted on the Amazon [S3](#) bucket in the US East Region. It is accessible in TSV format, with each review represented on a separate line. Tabs serve as delimiters in the file, and neither quotes nor escape characters are used.

Recent research has delved into the classification of Amazon customer reviews using advanced [ML](#) techniques. Alotaibi et al. ([2023](#)) introduced a novel framework that employs a bag-of-features approach for feature extraction, combined with a hybrid deep neural network for classification. This model, when tested on Amazon and AliExpress product reviews, achieved a remarkable 91.5% accuracy, outperforming many contemporary models in various metrics. On a related note,

Tripathy et al. ([2023](#)) developed a hybrid method that integrates an enhanced genetic algorithm with analysis of variance (ANOVA) for feature selection in sentiment analysis. Their approach, when applied to the Amazon Review dataset, yielded an accuracy of 78.60% and reduced the feature set significantly, demonstrating its efficiency compared to other algorithms. Both studies underscore the potential of [ML](#) in refining the classification of online customer reviews, offering valuable insights for e-commerce platforms and manufacturers.

In this research, the Amazon Customer Review dataset was employed for hyperparameter tuning with the objective of achieving high-quality word embeddings. Through this investigation, we examined various DNN methodologies to understand the influence of distinct hyperparameters on the efficacy of the word embeddings. Furthermore, we assessed how this efficacy translated to the precision of downstream tasks, such as classification.

The Amazon Customer Review dataset was selected to evaluate Deep Neural Network classification models due to its extensive and diverse text data, covering various product categories and customer opinions. This richness makes it ideal for testing and refining text classification algorithms. The connection to our main research objectives lies in the need to develop robust text analysis techniques. Evaluating models on the Amazon dataset allows us to benchmark performance, gain insights into hyperparameter optimization, and ensure model generalization, which are crucial for applying these techniques to agricultural datasets. Successful techniques from these experiments can then be transferred to classify and analyse texts from sources like Wikipedia or scientific publications about crops. By using this dataset, we lay a strong foundation for understanding text classification models, which is essential for tasks such as detecting cropping patterns and tracking the evolution of scientific knowledge, ultimately supporting our primary research goals in agricultural data science.

3.2.5. News articles

The HuffPost dataset is a comprehensive collection of approximately 210,000 news headlines that span from 2012 to 2022 (Misra, [2022](#)). It stands out as one of the most extensive news datasets, making it a pivotal resource for a range of computational linguistic studies. It's worth noting that after the initial collection of this dataset in 2018,

HuffPost discontinued its broad archiving of news articles. This shift means that while there are about 200,000 headlines from the period between 2012 and May 2018, the subsequent years up to 2022 only contribute around 10,000 headlines.

Regarding the dataset's content, each entry offers a category indicating where the article was published, the headline or main title of the news piece, a list of the article's authors, a direct link to the original article, a brief description summarising the news piece, and the publication date of the article. The dataset is diverse, covering 42 different news categories. To provide a glimpse into its composition, the three most populated categories are politics with 35,602 articles, wellness with 17,945 articles, and entertainment having 17,362 articles.

Khosa et al. (2023) explored news category prediction using transformers models like [BERT](#), Robustly Optimized [BERT](#) Pretraining Approach ([RoBERTa](#)), Masked and Permuted Pre-training for Language Understanding ([MPNet](#)) (Song et al., 2020), and Text-to-Text Transfer Transformer ([T5](#)) (Raffel et al., 2020). Using two news datasets from Kaggle, the study aimed to determine the effectiveness of sentence embeddings when paired with Softmax and Random Forest classifiers. Results highlighted the efficiency of [MPNet](#) versions v1 and v3 with Random Forest, achieving a high F1 score of 97.7%. On the other hand, a paper by (Kotiyal et al., 2023) addressed the challenge of fake news on social media. They introduced a novel approach to detect fake news in English and Hindi posts. By creating a dataset through web scraping and [NLP](#) techniques, they trained a deep learning model that utilised features like Embeddings from Language Models ([ELMo](#)) embeddings and [TF-IDF](#) to classify news into various categories. The study compared [ML](#) and deep learning models for classification using two datasets with over 45,000 articles.

In this study, the news categories dataset was utilised for an initial exploration of temporal knowledge evolution. This preliminary test laid the groundwork for evaluating the effectiveness of temporal knowledge evolution and the creation of temporal vector spaces that will be discussed in detail later. The promising outcomes from this initial test paved the way for a more extensive temporal knowledge evolution study based on the arXiv dataset.

3.2.6. ArXiv dataset

Since its inception in 1991, the arXiv online manuscript preprint archive has served as an important source of literature for researchers across various fields, including

computer science, mathematics, physics, and numerous interdisciplinary areas (Ginsparg et al., [2011](#)). As of the time of writing, the repository boasts over 2.3 million articles, offering a rich multigraph dataset that encompasses full-text articles, article metadata, and internal co-citations (Saier et al., [2023](#)).

The arXiv dataset has been employed in a multitude of studies. For instance, researchers have utilised the topology of the co-authorship graph to delve into link prediction, tested a hierarchically structured network model with the authorship graph, trained and evaluated an automatic text labelling system using the category labels of arXiv documents, and explored the efficacy of "paragraph vectors" in capturing document similarity using a subset of the full texts available on arXiv (Okamura, [2022](#)). This work discusses the integration of arXiv into the research ecosystem and its crucial role in transforming the dynamics of scientific research. The authors quantitatively demonstrate how arXiv has helped accelerate and shape the research community.

Clement et al. ([2019](#)) examined the citation data of arXiv, providing an analysis of the long-term citation trends across various disciplines. Their findings underscore the significant role of arXiv preprints in the research community. The authors developed a pipeline to standardise and streamline access to arXiv's data, which was subsequently used to analyse a citation graph and a corpus of full-text research articles. They also presented the extraction of a co-citation network from the arXiv dataset. In conclusion, they highlighted the instrumental value of the arXiv dataset in advancing research in structured relational modelling.

In this chapter, we will delve into the intricate use of the arXiv dataset for studying temporal knowledge evolution. We will provide an in-depth look at the pre-processing of this expansive dataset, leveraging cloud computing capabilities. This includes processing textual data, removing stop words, conducting lemmatization, performing stemming, and eliminating special characters, among other steps. These pre-processing measures are vital in preparing the data for the word embedding neural network. This network produces multidimensional vector spaces, serving as the foundation for the temporal knowledge evolution experiment.

3.2.7. Conclusion

This chapter has demonstrated the essential role of various datasets in enhancing our comprehension and application of computational and data-driven methods. The

combination of multilingual and English Wikipedia datasets was pivotal for effective [NIR](#) experiments, aiding in the identification and analysis of underutilised crops. Additionally, the curated collection of specialized literature and large datasets, including arXiv and Amazon User Reviews, provided a robust foundation for exploring temporal knowledge evolution and improving text classification techniques.

Crucial preprocessing steps, adapted to the unique characteristics of each dataset, optimized the data for embedding models, transforming textual data into a consistent and simplified format. This preprocessing significantly enhanced model performance and accuracy. The investigation of the Wikipedia multilingual corpus, the [BG](#) dataset, Amazon customer reviews, and news articles highlighted the broad applicability and versatility of these datasets across different research areas.

The use of the Wikipedia dataset was instrumental in identifying both scientific and local names of underutilised crops, while the [BG](#) dataset enabled the analysis of cropping trends based on global public interest. The Amazon Customer Review dataset was crucial for hyperparameter tuning and evaluating deep neural network classification models, showcasing the importance of high-quality word embeddings. The initial exploration of temporal knowledge evolution using news articles provided a foundation for a more detailed study using the arXiv dataset, emphasizing its significance in scientific research.

In conclusion, the datasets discussed in this chapter, along with tailored preprocessing techniques and advanced machine learning methods, have greatly contributed to the advancement of this research. These findings highlight the value of utilizing diverse and comprehensive datasets to deepen our understanding of complex phenomena, thereby supporting the primary research goals in agricultural data science and beyond.

Chapter 4

Information Retrieval

4.1. Introduction

Agriculture, as the foundation for food security and a vital component of the global economy, is undergoing unprecedented challenges and transformation. Major crops have historically been the mainstay of human dietary needs; however, underutilised crops present a promising solution to diversifying food sources and buffering against the challenges of climate variability. An overarching problem, though, is the limited knowledge about these underutilised crops, especially their cultivation patterns and ideal conditions for growth. Our global food system's resilience hinges on addressing this knowledge gap. At the crossroads of technology and agriculture, this research delves deep into harnessing computational advancements to illuminate dark areas of agronomic knowledge.

The journey started with the extensive collection of data on crop ecology, taxonomy, and other relevant variables. This data, amounting to thousands of semi-structured documents, is a testament to the vastness of existing research. However, its dispersion and non-standardized nature pose challenges. Traditional text mining approaches, while proficient in semi-structured data, are not always effective with complex, unstructured data sources like Wikipedia. Therefore, the advancement of computational algorithms, such as [ANN](#) and methods like [word2vec](#), has been of paramount importance. These modern techniques, while intricate, offer a dynamic and efficient approach to data processing by creating interconnecting vector spaces for words, facilitating deep semantic understanding.

Crowdsourced data, particularly from platforms like search engines and social media, holds tremendous potential for insights. Salama et al., ([2017](#)) analysed the search engine data for predicting [BG](#) cultivation patterns and has already showcased the efficacy of such approaches. Though labour-intensive, these methods hold the promise of shedding light on global crop cultivation and interest trends. Nevertheless, the dynamic realm of Big Data comes with its challenges. Data acquisition can be cumbersome, often compounded by the intricacies of methodologies needed for processing vast datasets.

This research pivots towards a more holistic and comprehensive approach. By employing [NIR](#) techniques and incorporating models such as [word2vec](#), we focus on extracting intricate details from a colossal dataset from Wikipedia. We initiated our research on the English Wikipedia corpus to detect cropping patterns and gauge worldwide interest in underutilised crops. We then expanded our investigation to Wikipedia in 178 languages to integrate the information from various regions and correlate it with local crop names. Such a methodology is not only aimed at discerning details like crop soil requirements and crop vernacular and scientific names of underutilised crops but also provides a powerful tool for research in vast text datasets. Through rigorous comparisons with established databases and continuous optimization, our experiments underscore the integral role of modern computational techniques in advancing our understanding of global agricultural trends.

In summary, by melding crowdsourced data, [ML](#), and advanced neural network techniques, this research embarks on a quest to harness the power of technology in the service of agriculture, shedding light on underutilised crops, and their potential role in fortifying our global food systems.

This chapter presents three critical experiments at the intersection of agriculture and computational methods. The first study investigates the use of online search data to identify cultivation patterns of lesser-known crops. Next, we discuss the shift from traditional text mining to [NIR](#) in the context of agricultural data analysis. Lastly, we examine the application of [NIR](#) techniques in classifying crop characteristics. Together, these experiments highlight the growing integration of technology in agricultural research.

4.2. Detecting cropping patterns of underutilised crops using online search data

4.2.1. Introduction

Underutilised crops have the potential to diversify food sources and address climate variability challenges. However, a significant barrier to their broader acceptance is the limited knowledge about the regions where they are cultivated. These crops are often grown in marginal areas via subsistence farming. Currently, there is an absence of a comprehensive global database or an effective method to obtain information on the cultivation patterns of these underutilised crops. This research presents a novel

approach to identifying patterns by examining online search engine data, with the objective of assessing the global temporal interest in these crops. As a demonstration, we gathered keywords related to [BG](#) from both local and international sources. Utilising the Google AdWords platform and 30 specific terms related to [BG](#) as listed in Table 1, we gathered search data over a two-year period at the city-specific level. Preliminary evaluations conducted via a software prototype unveiled trends in [BG](#) search behaviours, providing valuable insights into probable cultivation regions and attributes of the value chain. To validate our findings, we compared our data with known networks of [BG](#) cultivators and researchers (Mayes et al., [2012](#)). The outcomes not only aligned with established regions but also highlighted new potential areas for exploration. This suggests that our method can offer valuable insights into cultivation trends and research interests globally.

4.2.2. Method

Daily, search engines worldwide process billions of queries. Recent interdisciplinary research has leveraged this search engine data to derive insightful information and indicators (Polykalas et al., [2013](#)). These keyword search records, representing collective user data, are typically presented in diverse formats by search engines. The primary goal of this research is to identify efficient techniques for extracting both implicit and explicit information from extensive datasets. One task within this research is to determine the global cropping pattern of underutilised crops. The study began by employing traditional methods to gather this information from Google search engine data. While the experiment succeeded in pinpointing the cropping pattern, it highlighted the significant manual effort and the use of specialised software that becomes challenging to repurpose if the original dataset format alters.

Google handles more than 3.5 billion queries daily (Stats, [2023](#)). Considering the vast user base of Google and its associated search platforms, the data from these searches can highlight societal interest in specific topics (Bardak et al., [2015](#); Carneiro et al., [2009](#); Van et al., 2012). The Google AdWords platform aids businesses in tracking their performance. Within AdWords, there are multiple analytical tools. For instance, the Keyword Planner (Geddes et al., [2014](#)) assists businesses in selecting relevant keywords. A notable feature of this tool is its ability to analyse current keyword trends. Users input specific words or phrases, and the Keyword Planner provides data on the frequency of these search terms within specified regions and

timeframes. This section elaborates on the procedures and settings employed within the Google Keyword Planner tool to retrieve search request counts for specific keywords.

Input: The Google Keyword Planner accepts keywords either as plain text or in a comma-separated values ([CSV](#)) format. For this study, a [CSV](#) list containing 40 [BG](#) names, as previously mentioned, was uploaded.

Configuration:

- **Desired Output:** Volume data.
- **Search Engine Selection:** Google search engine along with its affiliated platforms.
- **Geographical Scope:** Global.
- **Time Frame:** July 2014 to June 2016.

The timeframe of the extracted data was constrained by the limitations of the extraction tool (using the free version). This experiment serves as foundational work for retrieving domain-specific information and paves the way for subsequent experiments. There was no need to rerun the initial experiment since later experiments utilised more advanced and efficient techniques, such as word embeddings and temporal knowledge evolution.

Procedure: The tool necessitates manual operation for each individual country. It generates a list detailing search request count, segmented by country and month. Additionally, the tool categorises results based on the device utilised for the search, such as desktop, mobile, or tablet.

In this study, the Google Trends tool was examined, which offers insights into specific search keyword trends (Choi et al., [2012](#)). Numerous research initiatives across different fields have begun utilising Google Trends to explore societal interactions in their studies (Trevisan, [2014](#)). However, a limitation of Google Trends is its representation of trends in percentage form rather than absolute numbers. Specifically, Google Trends designates the peak search result within a given time or location as 100%, with subsequent results shown as percentages relative to this peak. Thus, it does not present the exact number of search queries. Additionally, Google Trends sets a minimum limit for displaying search trends; if search counts fall below this limit, the data remains undisclosed. Given that searches for underutilised crops often yield relatively low counts, as will be discussed in the results section, Google

Trends is not the primary tool for this study. Nevertheless, it served to corroborate Google keyword findings and to identify additional keywords suggested by the tool.

4.2.3. Experimental design

In our experiment, we harnessed a vast volume of search engine data, specially from Google, which processes over 3.5 billion queries daily, to understand societal interest in specific topics. Utilising tools within the Google AdWords platform, primarily the Keyword Planner, we inputted a set of keywords, specifically a [CSV](#) list containing 30 [BG](#) names, to analyse their search frequency in designated regions and timeframes. [BG](#) is an underutilised crop, and its vernacular names can be found in various publications. The study began with a manual review where scientists in [CFF](#) responsible for CropBase examined the literature on [BG](#) to identify all its scientific and vernacular names. This resulted in a list of 30 [BG](#) names. This experiment relied on the fact that the Google search engine compensated for the typos using its own AI correction mechanism. These names were subsequently input into the Google Adwords tool to obtain Google search data related to the various names of [BG](#). This tool provided information on global monthly search interest for [BG](#), broken down by country, city, and device. To ensure the accuracy of the data, we cross-referenced the results with Google Trends, which offers an indication of global search interest relative to a chosen baseline, such as the scientific name of [BG](#). This step aided in refining the data by identifying and eliminating outliers. Should the data not meet our validation criteria, it enters a 'human-in-the-loop' process for manual verification, where it is either approved or rejected. The refined list of search results for [BG](#) is then subjected to text mining tools to derive cropping patterns from the provided [CSV](#) files. The text mining tools are custom built tools using different programming languages such as python, PHP and shell scripts on Linux. Following the extraction, another review step ensures the output format meets criteria and passes basic checks. The basic checks in our experiment include verifying the accuracy of the initial [CSV](#) list of 30 [BG](#) names, ensuring they are correctly spelled and formatted. We then confirm that the search data from Google AdWords includes the correct search frequency information for each name, covering the designated regions and timeframes. This data is cross-referenced with Google Trends to validate search interest and identify any outliers, which are then eliminated. Data that does not meet validation criteria undergoes a 'human-in-the-loop' process for manual verification. The validated data is then processed using text mining

tools to derive cropping patterns, with a final review to ensure the output meets specified criteria and passes basic integrity checks. If the extracted data passes this step, it moves to its final storage destination. Otherwise, it re-enters the 'human-in-the-loop' process for manual validation, after which it is either approved for storage or rejected. The final dataset undergoes processing using a visualisation dashboard, and the cropping patterns of [BG](#) are displayed on a global map using Google Maps charts. The Figure 6 provides a visual representation of the data extraction and processing steps, as well as the subsequent creation of a visualisation dashboard and [BG](#) global cropping patterns maps.

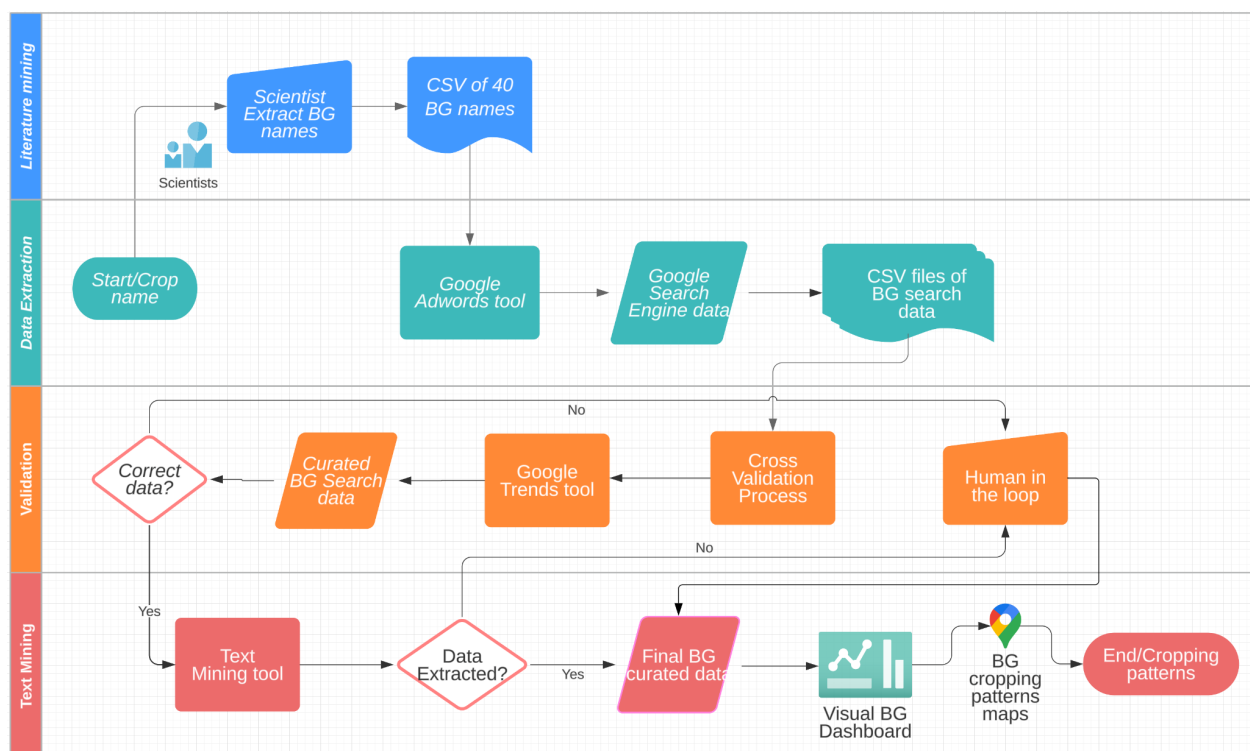


Figure 6: Architecture diagram of how to use text mining and large data to extract cropping patterns.

4.2.4. Results and analysis

Utilising a dataset of 230 [CSV](#) files, we analysed search request counts related to [BG](#) across various countries, devices, and months, processed using Linux Centos shell scripting. Our geographical analysis, visualised through Google GeoChart, highlighted the average search counts from July 2014 to June 2016, with countries such as Indonesia, Thailand, and the United States showing significant interest. Local terminologies for [BG](#) varied, with "kacang bogor" predominant in Indonesia, reflecting its commercial importance. Temporal examination revealed fluctuating search trends,

notably increased interest in Indonesia from 2015 to 2016. Device-based analysis indicated mobile searches were prevalent in commercialised regions, while desktop searches dominated in [BG](#) research-intensive areas like the United States and South Africa. This study offers insights into global [BG](#) trends from geographical, temporal, and device-based perspectives, yielding 120K numerical data points, such as optimal crop temperatures, and 30K textual data like taxonomical crop names.

We developed a web-based geographical interface utilising Google GeoChart to visually represent the data. Figure 7 depicts the average number of search queries associated with [BG](#) names, segmented by country, for the period extending from July 2014 to June 2016. The leading countries in terms of search volume are as follows: Indonesia with 1,110 queries, Thailand with 470, the United States with 400, both South Africa and Malaysia with 320 each, Nigeria with 260, and the United Kingdom with 220.

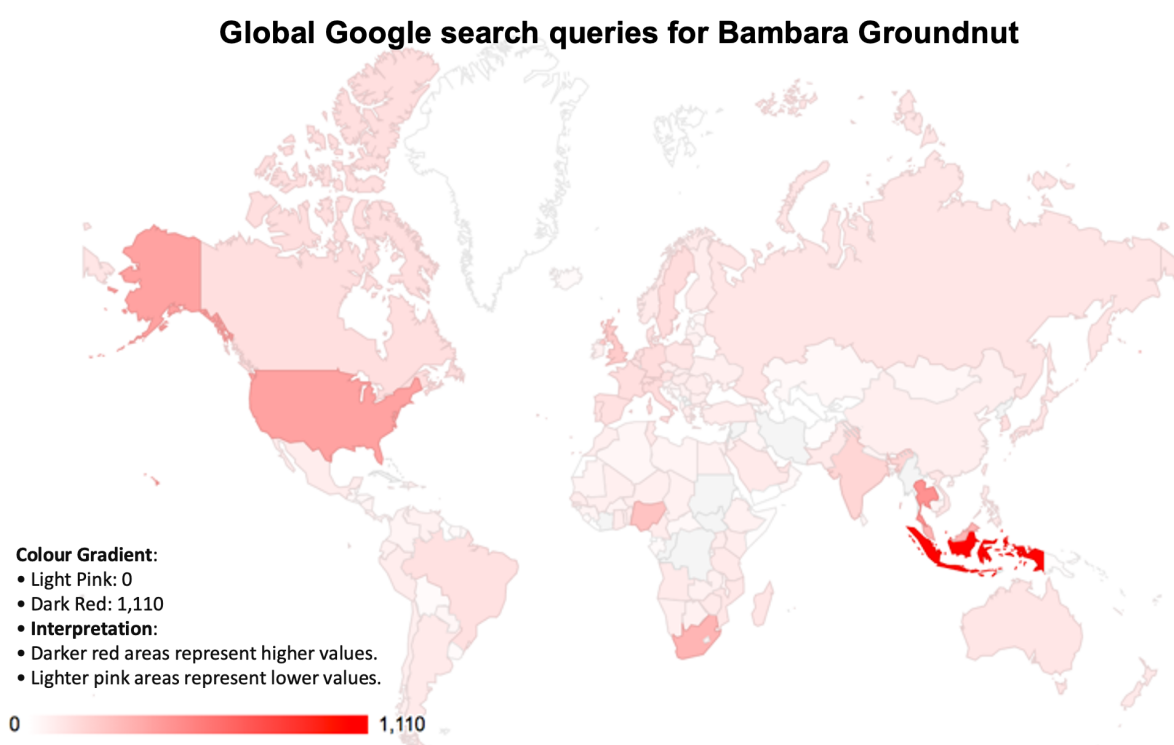


Figure 7: Global Google search queries for "[BG](#)."

In Indonesia, the [BG](#) has been commercialised and is readily available for purchase in markets, as noted by (Plahar et al., [2009](#)). Locally, it is referred to as "kacang bogor." Among the various names for the [BG](#), "kacang bogor" dominates online searches in Indonesia, accounting for 91% of all related queries. Thailand follows Indonesia in this

regard, with the most frequently searched term being "ถั่วเหลือง." In the United States, one might anticipate that the primary search term would be "Vigna subterranea," the scientific designation for the [BG](#), given the country's research institutions. However, the terms "Bambara groundnut" and "Bambarra groundnut" (with an additional "r") are the most prevalent. Preliminary investigations indicate that this pattern might be due to African Americans' efforts to re-establish ties with crops of ancestral significance, such as the [BG](#). In countries like the United Kingdom and South Africa, the term "Bambara groundnut" emerges as the predominant search query. Conversely, in Malaysia, the term of choice is the native name, "Kacang poi."

The metric "Requests Data Time," generated by the Google Keywords tool, aids in conducting temporal analyses and visualising search trends over defined periods. As illustrated in Figure 8, there's a noticeable trend in search queries covering a two-year period from July 2014 to June 2016. The blue line on the graph signifies the search trend specific to Indonesia. It's noteworthy that two distinct peaks emerge in March for both 2015 and 2016. We consulted experts from [CFF](#) team regarding [BG](#) and they confirmed that these peaks align with the crop's annual harvest season. The Indonesian data indicates a significant rise in interest from 2015 to 2016, implying a growing public interest in [BG](#) and related products. In Thailand, similar patterns were observed around November 2015 and October 2016, though it might correspond to the Thai harvest season for [BG](#).

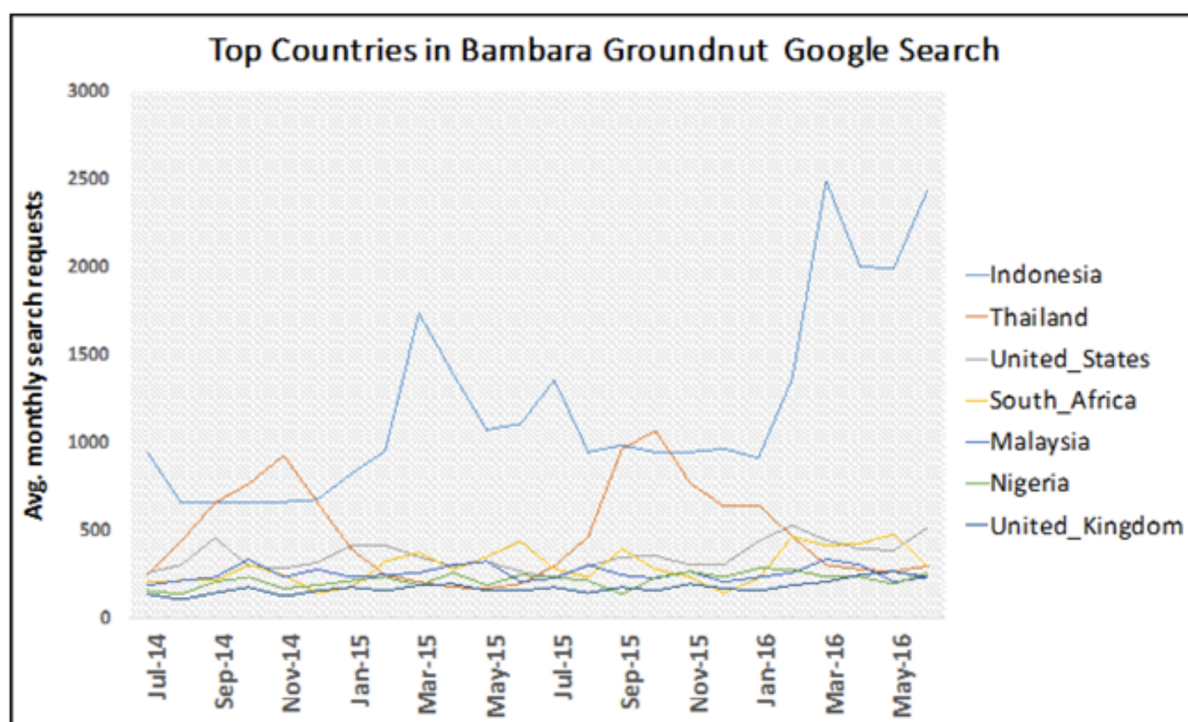


Figure 8: Leading countries in [BG](#) Google searches.

The third metric generated by the tool relates to the devices utilised to initiate search queries. The Figure 9 showcases the average monthly search requests, categorised by device type: Desktop and Mobile. It is postulated that a higher frequency of desktop-based searches could suggest activity from research institutions, while an increase in mobile and tablet searches might indicate a broader interest from the general public and farmers. The data indicates that countries like Indonesia, Thailand, and Malaysia predominantly use mobile devices for their search queries, suggesting significant engagement from the general public. This observation is consistent with the fact that in these countries, [BG](#) is a key commercial crop and is foundational to several products. On the other hand, the United States and South Africa demonstrate a higher propensity for desktop-based searches, which might indicate a focus more aligned with research and industrial sectors (Khan et al., [2021](#)) (Tan et al., [2020](#)) (Mayes et al., [2019](#)). In countries where [BG](#) is a crucial commercial crop, its versatility and nutritional benefits have led to its integration into various products. [BG](#) flour, rich in protein and fibre, serves as a gluten-free alternative for baking bread, cakes, and cookies, catering to those with gluten intolerance or celiac disease (Khan et al., [2021](#)). Additionally, roasted [BG](#) is a popular, nutritious snack, offering a healthier option compared to many conventional snacks (Tan et al., [2020](#)). The legume is also

processed into plant-based milk, which is lactose-free and packed with essential nutrients like calcium, iron, and protein (Mayes et al., 2019). Furthermore, BG is utilised in animal feed to enhance the nutritional quality for livestock, and it is processed into protein supplements for athletes and health-conscious consumers (Khan et al., 2021). Traditional dishes incorporating BG, such as soups, stews, and porridges, not only hold cultural significance but also provide balanced nutrition (Tan et al., 2020). These diverse applications underscore BG's foundational role in both the food industry and agricultural sectors of these countries.

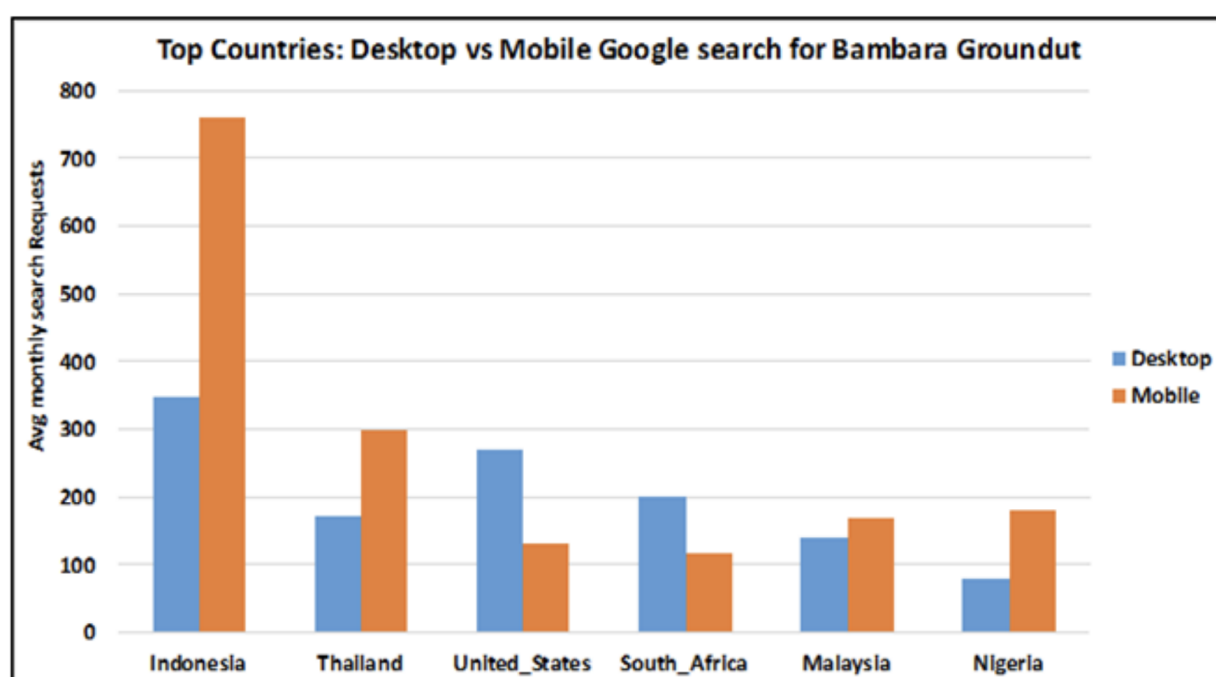


Figure 9: Primary countries for BG Google searches, categorised by desktop and mobile devices.

4.2.5. Conclusion

The significance of underutilised crops in addressing global challenges like climate change, food security, and nutritional deficiencies cannot be overstated. While primary crops have been extensively researched and documented, underutilised crops often lack such centralised hubs of information, leading to gaps in organised data. This study aimed to bridge this gap by leveraging the vast data generated by search engines, specifically Google, to discern potential cropping patterns of underutilised species. The cropping patterns indicate areas where underutilised crops are potentially cultivated. These patterns also highlight regions with potential interest in such crops, which can

help identify potential markets. By analysing search engine data, we sought to categorise information based on location and temporality, offering insights into crop utilisation patterns and tracking interest in these crops over time. For example, as a result of this experiment, a graph will demonstrate the temporal global interest in one the underutilised crop which will indicate the potential temporal cropping pattern. The findings from this research not only contribute to the knowledge repository for underutilised crops but also underscore the potential of big data analytics in agricultural research. As we move forward, it is imperative to continue such interdisciplinary approaches, combining traditional agricultural research with modern data analytics, to promote the broader acceptance and integration of underutilised crops into global agricultural strategies.

The research into identifying global cropping patterns of underutilised crops using Google search data highlights several technical challenges and innovations. Traditional methods required significant manual effort and specialized software, making them labour-intensive and not easily adaptable to different datasets. The Google Keyword Planner tool provided valuable insights through keyword analysis and data categorization by region and time frame, but it also necessitated manual operation and data segmentation, adding complexity. Google Trends offered useful trend analysis, but its relative data representation and minimum display limits posed challenges, particularly for niche topics with low search volumes. To address these limitations, the study suggests leveraging machine learning techniques like word embeddings and [NIR](#). These advanced methods can automate information extraction, enhance scalability, and provide deeper insights by capturing semantic relationships between terms. Integrating these modern techniques with traditional approaches represents a promising interdisciplinary strategy, enhancing the ability to analyse and utilize big data in agricultural research to tackle global issues such as climate change, food security, and nutritional deficiencies.

In the experiment, while the method used was effective in discerning the cropping trends of underutilised crops, the manual effort and specialised text mining pipeline were not scalable or adaptable for other datasets or similar tasks. Techniques rooted in ML, like word embedding and [NIR](#), might serve as better alternatives for both direct and indirect information extraction. The forthcoming experiment will dive into the specifics of retrieving data using word embeddings, focusing on the Wikipedia dataset, especially details pertaining to local and scientific names of underutilised crops and its

relationship to countries. The experiment paves the way for the journey from text mining to [NIR](#).

In conclusion, this experiment has explored a novel approach to assess global temporal interest in underutilised crops, using [BG](#) as a case study. The study utilised online search engine data to uncover patterns in [BG](#) search behaviours, shedding light on probable cultivation regions and characteristics of the value chain. The results have provided valuable insights into the cultivation trends and research interests related to [BG](#) on a global scale. The analysis revealed that [BG](#) gained significant interest in countries such as Indonesia, Thailand, the United States, South Africa, Malaysia, Nigeria, and the United Kingdom. Local terminologies for [BG](#) varied, reflecting its diverse presence across different regions. Temporal analysis indicated fluctuations in search trends, with notable spikes aligning with the crop's annual harvest season in some regions. Moreover, the study examined the devices used for search queries, highlighting variations between mobile and desktop searches. This distinction may indicate different levels of engagement from the general public, farmers, and research institutions across various countries.

In summary, this research contributes to our understanding of the global dynamics surrounding underutilised crops, showcasing the potential of using online search data as a valuable tool to identify cultivation patterns and research interests. Further exploration and application of this approach could aid in promoting the sustainable cultivation of underutilised crops and addressing food security and climate challenges in the future. While our employment of big data techniques and text mining operations has provided valuable insights into cultivation patterns and global interest in underutilised crops, it's essential to recognize its limitations. The reliance on search engine data, although revealing, might not capture the entire spectrum of interest or the nuances in the cultivation of underutilised crops. This method largely depends on the visibility and accessibility of online resources, potentially neglecting regions with limited internet penetration or those using alternative sources of information. Additionally, while search data can hint at patterns of interest, it doesn't provide in-depth scientific insights or detailed local cultivation practices. Hence, the progression to [ML](#), and specifically the exploration of [NIR](#) in scientific literature and Wikipedia, becomes imperative. Implementing [ML](#) techniques like word embedding will allow us to map the semantic distribution of knowledge more comprehensively. Through this,

we aim to fill the knowledge gaps left by our initial approach, systematically extracting richer information and thereby enhancing our understanding of underutilised crops.

4.3. Journey from text mining to [NIR](#) in agriculture data science

4.3.1. Introduction

The progression of science has led to an accumulation of vast amounts of research data. This data, often dispersed across various structured or semi-structured formats, presents a challenge. Our journey began with 2,355 pieces of semi-structured data on crop ecology and taxonomy obtained from [FAO](#) ECOCROP database (Ramirez-Villegas et al., [2013](#)). The manual collection and documentation of this data consumed hundreds of hours. While humans can interpret this data with ease, manual standardisation would be time-consuming and prone to errors. Hence, the need for computational algorithms becomes evident. As we scale from thousands to millions of documents, there's a requisite shift in computational methodology. This vast corpus of data holds the potential for extracting reliable information. Early studies, such as that by Sebastiani, ([2002](#)), highlighted the potential of [ML](#) in text classification. Techniques like the [CNN](#) have shown promise in this domain, as evidenced in (Kim, [2014](#)). [Word2vec](#), for instance, has been effectively utilised for ontology concept extraction from data corpuses (Wohlgemant et al., [2016](#)). Similarly, Swoboda et al. ([2016](#)) demonstrated [word2vec](#)'s capability in extending seed taxonomy. The evolution of ANN), coupled with enhanced computational power, has paved the way for groundbreaking experiments (Goodfellow et al., [2016](#)). Training neural networks with vast data sets, previously deemed impossible, is now feasible. The core idea revolves around representing each word as a numerical vector, which, when processed against other words and documents, creates a vector space. This space facilitates myriad analyses by measuring similarity between words using methods such as Cosine similarity, Jaccard similarity, Euclidean distance, and Manhattan distance. All of these methods reflect on their relationships and provide valuable insights for various text analysis tasks. Crowdsourced data analysis holds immense potential. Institutions, governments, and industries invest heavily in understanding market dynamics and human behaviour. The proliferation of social media, the internet, and open-source projects has led to the phenomenon known as Big Data. Navigating this data can yield insights that save time, effort, and resources. However, challenges persist, primarily

in data acquisition and the methodologies, software and hardware needed for data processing. While various algorithms exist for these tasks, research in this domain is ongoing.

In the study by Salama et al., (2017) analysed over 300,000 Google search events were analysed to predict [BG](#) cropping patterns, but challenges were faced due to digital biases and the limitations of text mining semi-structured data. To address these issues, we expanded their approach in subsequent research by incorporating alternative crowdsourced datasets and developing sophisticated tools for large dataset analysis, including [ML](#) techniques. They emphasized the use of [NIR](#) methods, particularly [word2vec](#), to analyse a 3.4 billion-word Wikipedia dataset, revealing significant connections between crop terminologies and countries. This advanced approach, integrated with the CropBASE knowledge system, demonstrated the effectiveness and flexibility of [NIR](#) methods in providing a more comprehensive understanding of underutilised crops, overcoming limitations of traditional text mining.

4.3.2. Method

In the pursuit of deeper insights and improved knowledge extraction from vast data corpora, our research embarked on evaluating the integration of two prominent methodologies: text mining and [NIR](#). As vast and varied datasets are becoming increasingly prevalent, especially in disciplines such as taxonomy and ecology, understanding and efficiently extracting relevant information becomes paramount. Herein, we present an overview of our methodological journey, detailing the processes, challenges, and results encountered across both methodologies. The first approach, text mining, enabled us to design a deterministic algorithm that expertly navigated through a corpus of 1.2 million words from EcoCrop dataset (Ramirez-Villegas et al., 2013), pulling out pertinent data points and categorising them for easier accessibility such as scientific names, vernacular names and other crop characteristics. On the other hand, using [NIR](#) based on the [word2vec](#) model, is aimed at capturing semantic nuances by training on vast datasets like the English Wikipedia corpus. This method's aspiration was to grasp the intrinsic relationships between common crop names and a myriad of concepts, a challenge further compounded by evolving lexicons and database versions.

Text mining

Initially, we employed text mining to develop an algorithm for deterministic data extraction from a modest corpus, encompassing "1.2 million words from EcoCrop dataset (Ramirez-Villegas et al., [2013](#))" from taxonomy and ecology documents. The primary objective was to design rudimentary data collection tools to format and centralise all documents. Due to the data volume and human involvement in data collection, we encountered various data deformities and special characters. The algorithm's subsequent task was to clean the data, removing special characters and ASCII encoding to avert potential crashes. The data was categorised into three types: textual paragraphs, tabulated data sheets, and specific text within tables. We formulated two extraction algorithms: a PHP-based one for numerical data tables and a Linux shell script for extracting data from tabular or paragraph formats. The processed data was then structured and stored in a relational knowledge-based system called CropBase (Jahanshiri et al., 2015; Jahanshiri et al., [2014](#)). *Figure 10* illustrates the text mining process for data extraction from documents. The database used in this system is a MySQL database. We used a combination of scripts with PHP and Python along with Linux Shell Scripting to automatically collect the data and clean it. The structured dataset is stored in a form of a relational database.

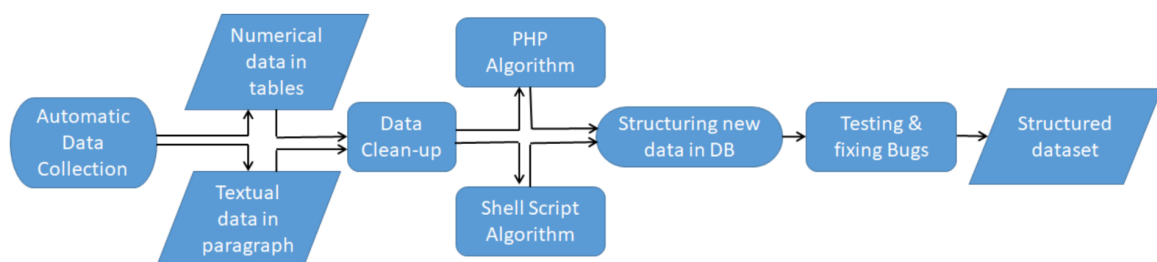


Figure 10 Text Mining process for ecological and taxonomic data.

[NIR](#) with Wikipedia

The alternative approach involved training [word2vec](#) on the English Wikipedia corpus and subsequently gauging the semantic proximity between common English crop names and various concepts. To validate the accuracy of the extracted data, it was compared against the Ecocrop database (Ramirez-Villegas et al., [2013](#)) which contains crop attributes and characteristics. The initial experiment utilised a pre-

trained model from English Wikipedia 2015. However, this model presented challenges: 80% of the target vocabulary was absent from the 2015 Wikipedia dictionary. Additionally, the model was case-sensitive, failing to reconcile variations in word capitalization. These issues, among others, compromised vocabulary measurement accuracy. Despite these challenges, the results retained semantic relevance and facilitated substantial data analysis. To enhance accuracy and expand the vocabulary dictionary, we trained a [word2vec](#) model on the English Wikipedia 2017 edition. The training, implemented in Python, spanned approximately 10 hours on a specialised Google Cloud Machine with 70 [CPU](#) Cores and 128 Gigabytes ([GB](#)) RAM, operating on Debian OS. The English Wikipedia corpus, as of October 2017, stood at 14[GB](#) in compressed XML format. *Figure 11* delineates the training and model utilisation steps. The model, post-training on the 2017 Wikipedia edition, exhibited a 30% vocabulary dictionary expansion compared to the 2015 pre-trained model.

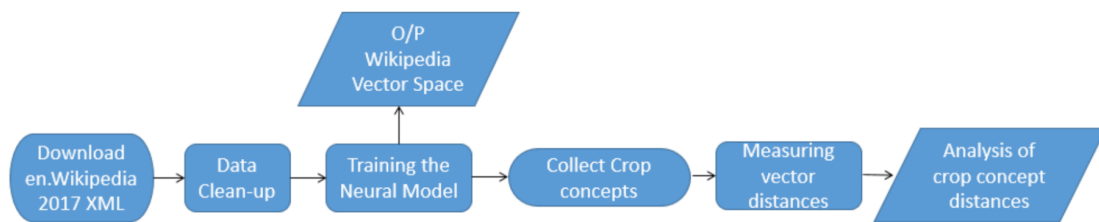


Figure 11 Process of data preparation and model training

4.3.3. Experimental design

In this experiment, our primary aim is to extract explicit and implicit knowledge for underutilised crops, focusing on taxonomy and ecology documents and online resources. We adopted two main approaches: text mining and [NIR](#). Initially, we engaged in text mining, extracting and cleaning data from a corpus of 1.2 million words, and subsequently categorising it into various formats. This structured data was stored in CropBase, a relational knowledge system. Parallely, [NIR](#) was implemented through the [word2vec](#) model trained on the English Wikipedia corpus 2017 edition, aimed to ascertain semantic relationships between common English crop names and diverse crop attributes. Despite initial challenges with the Wikipedia pre-trained model, the 2017-trained model demonstrated a 30% enhancement in vocabulary coverage. *Figure 12* illustrates the text mining and [NIR](#) process to extract insights about crop attributes.

The text mining procedure begins with manual data collection by researchers. Data is sourced from online platforms, published works, tables, [CSV](#) files, and online databases such as EcoCrop dataset (Ramirez-Villegas et al., [2013](#)), focusing on crop attributes. Once this data is combined and cleaned, it totals 1.2 million words EcoCrop dataset (Ramirez-Villegas et al., [2013](#)). The data is then classified into three types: tabular data, publications, and semi-structured text, which is systematically formatted text from which explicit knowledge can be extracted. Two distinct data processing methods were developed. The first addresses the tabular data, ensuring its cleanliness, curation, and readiness for relational database upload. An intermediate governance step verifies the data's quality, and if issues arise, manual curation is initiated. The second method, based on shell scripting, manages text data. This involves cleaning the text, processing characters, and extracting specific crop knowledge for database readiness. Non-standardized textual data is retained in a refined corpus for the [NIR](#) pipeline. Like the first method, a validation process is in place, with manual intervention as needed.

The [NIR](#) process leverages word embedding techniques to produce quality embeddings that extract both implicit and explicit data from the existing corpus. This starts with acquiring English Wikipedia files in XML format, the only format accessible during this study. Python was utilised for data transformation from XML to text. Subsequent [NLP](#) tasks were employed, such as tokenization, stemming, lemmatization, and various data cleaning techniques. Once prepared, the data undergoes [word2vec](#) training preparation. Prior to training, the data undergoes a validation and curation check. If issues are detected, manual intervention is required. Otherwise, it progresses to the final repository for training. Training preparation involves hyperparameter selection. The training dataset merges the Wikipedia English corpus with the cleaned corpus from the text mining process. Once embeddings are generated, a validation step commences, comparing specific database entities with the relevant embeddings and evaluating their similarity. The derived relationships are compared with data from the text mining process that is stored in the relational database to assess word embedding effectiveness. This information is visualised, as presented in this chapter's results. The data is then directed to a database for researcher access and future studies. Notably, within the crop knowledge base, there's a distinction between human-gathered information and machine-extracted data, crucial for maintaining data governance and integrity.

In our experiment which focused on extracting knowledge about underutilised crops, we employed text mining and [NIR](#) techniques to process a corpus of 1.2 million words from various online platforms and databases. The cleaned and categorized data was stored in CropBase, a relational knowledge system hosted on a MySQL database in the cloud. To ensure data quality, an intermediate governance step was implemented to verify cleanliness, curation, and database readiness, with manual curation employed as needed. For the [NIR](#) process, we utilised the 2017 English Wikipedia corpus to train the [word2vec](#) model, enhancing vocabulary coverage by 30%. The data underwent thorough validation and curation checks, with manual intervention if issues were detected. The embeddings generated were compared with entities from the Wikipedia corpus and the CropBase database to evaluate similarity and effectiveness. This structured approach allowed us to distinguish between human-gathered and machine-extracted data, maintaining robust data governance and integrity throughout the process.

In this experiment, the goal is to extract both explicit and implicit knowledge about underutilised crops by focusing on taxonomy and ecology documents and online resources. The process involves two main approaches: text mining and [NIR](#). Initially, researchers manually collected data from various sources, including online platforms, publications, tables, [CSV](#) files, and databases, compiling a corpus of 1.2 million words. This data was cleaned and categorized into tabular data, publications, and semi-structured text, then stored in CropBase, a relational knowledge system. For text data, shell scripting was used to clean, process characters, and extract specific crop knowledge, while tabular data was curated for database readiness. Both data types underwent validation and manual intervention to ensure quality. In parallel, the [NIR](#) approach employed word embedding techniques using the [word2vec](#) model, trained on the 2017 English Wikipedia corpus, to uncover semantic relationships between crop names and attributes. The Wikipedia corpus was converted from XML to text, followed by [NLP](#) tasks like tokenization, stemming, lemmatization, and cleaning. This processed data, combined with the cleaned text mining corpus, was used to train the [word2vec](#) model. The resulting embeddings were validated by comparing them with database entities and visualizing the relationships. Notably, the model demonstrated a 30% enhancement in vocabulary coverage, meaning it recognized 30% more relevant words compared to the pre-trained model. This comprehensive approach ensures that the knowledge base maintains data governance and integrity,

distinguishing between human-gathered and machine-extracted information, ultimately facilitating advanced research on underutilised crops.

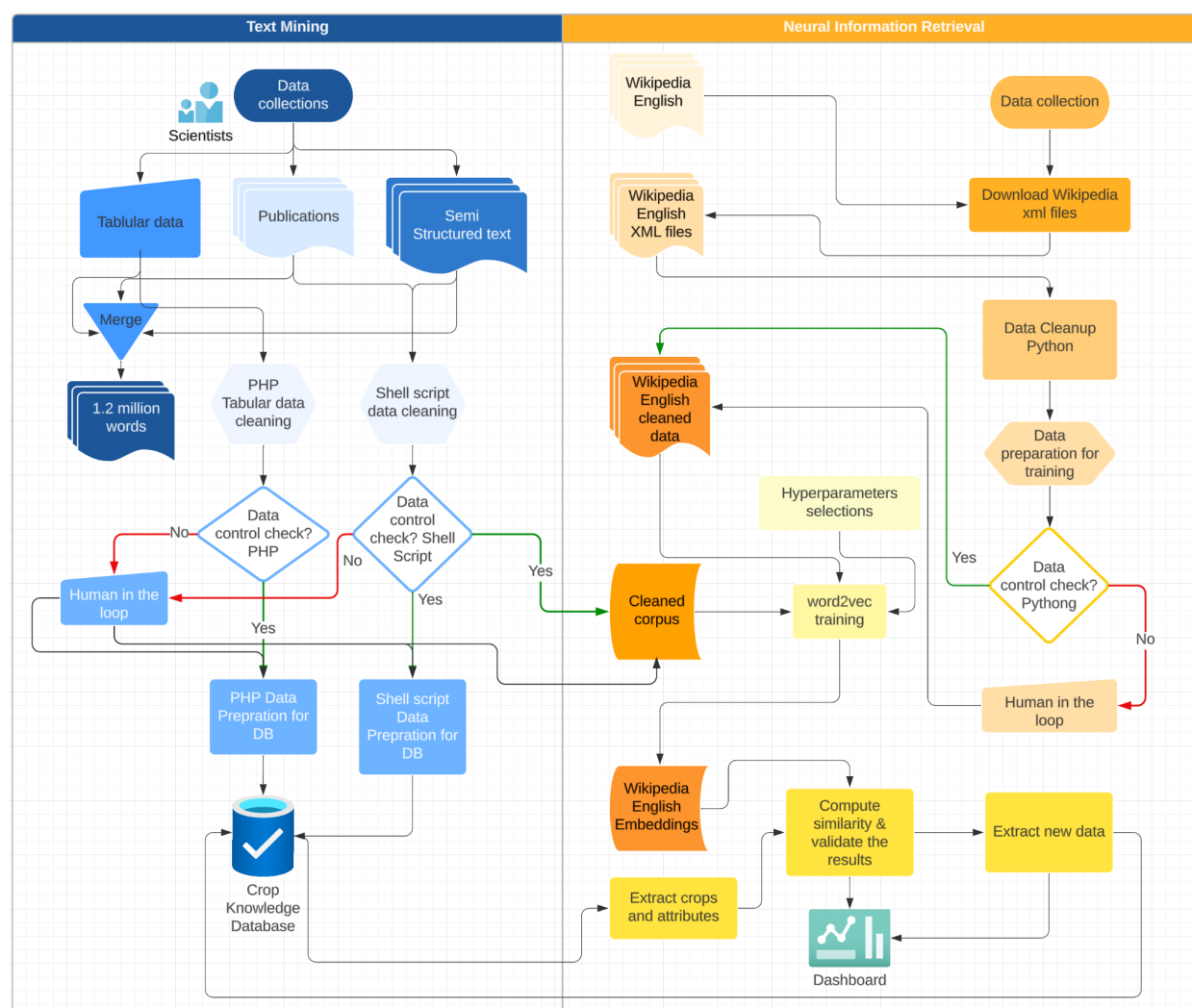


Figure 12: Text mining and [NIR](#) to extract insights about crop attributes.

The purpose of text mining in this process is to take the input from unstructured data for example, textual data that include quantifiable data explicitly but in a natural language. The extracted data will be stored in a database to be used for validation and comparison in a later stage as indicated in the diagram. Table 2 shows a sample of how the database of the global knowledge database for the crop looks like. The crop database is a dynamic system built on a relational data model, continually being enhanced to improve its functionality. It consists of two main components: the physical database, hosted on the cloud using MySQL technology, and the user interface developed with SQL Maestro Technology, available on the CropBASE website. The system offers varying levels of access; open access allows users to view, filter, sort,

and bulk download data, while restricted access is designated for administrative, curation, and data management activities. Data is presented in primary and secondary sections across different tabs for easy access and updates. Primary data includes crucial information on crops, such as taxonomy, images, uses, nutritional data, adaptation requirements, and market information. Secondary data comprises additional details that enhance the primary data's usability, such as accuracy flags, location details, online datasets, AI-generated data, and metadata including sources and contributors. This structured approach ensures comprehensive data management and accessibility for users.

Table 2 Example of the global knowledge database

☐		Actions	Crop ID	Name	Scientific Name	Family	Intraspecific Category	Synonyms	Crop Type	Notes	Metadata ID
☐			10001002783	Nectarine	Prunus persica var. nucipersica	Rosaceae	Variety (varietas)	Prunus persica var. nectarina	Permanent	NULL	8409
☐			10001002780	Bergamot	Citrus bergamia	Rutaceae	Species	Citrus aurantium ssp. bergamia	Permanent	NULL	8394
☐			10001000952	Swede	Brassica napus subsp. rapifera	Brassicaceae	Subspecies	Brassica napus var. napobrassica, Brassica napobrassica	Temporary/Permanent	NULL	8418

The extracted information from the input corpus included scientific names, vernacular names, synonyms, and cropping patterns specific to the locations where crops are cultivated. The methodology involves using text mining and training a [word2vec](#) model on the input data to generate embeddings representing this knowledge. A similarity function is then used to identify relationships among crop scientific names, vernacular names, and geo-locations. The model's input comprises textual data from Wikipedia and domain-specific literature, while the output consists of the generated embeddings. The similarity function applied to these embeddings will produce data to be stored in a database. This output will be validated against existing data collected from the EcoCrop database.

4.3.4. Results and discussion

The [word2vec](#) model was trained using data sourced from the 2017 edition of the English Wikipedia. Once trained, it generates a vector space that corresponds to the lexicon found within this English Wikipedia dataset. Utilising this model, relationships were gauged between 91 English crop names and several thousand distinct concepts, including country names, vernacular crop names, and more. The resulting output is a scalar value that varies between -1 and $+1$; with $+1$ signifying a direct relationship and -1 indicating an inverse relationship. An inverse relationship in cosine similarity

for word embeddings means that two words are conceptually opposite, indicated by a cosine similarity value close to -1, such as “hot” and “cold” or “love” and “hate.”. Upon examining a selection of these results manually, it was determined that employing data visualisation techniques would be most efficacious for illustrating the outcomes of this extensive data analysis. Consequently, three distinct visualisation methodologies were employed to represent the data.

The used visualisation is a geographical map, elucidating the interrelations between countries and the English names of crops, depicted in Figure 13, crafted using the Google GeoChart library. This map distinctly showcases the relationship of the crop, durian, with various global nations. A green colour gradient is utilised, with intensifying shades representing strengthening relationships. It is observable that the durian crop exhibits substantial ties with Southeast Asian and equatorial nations, congruent with widespread understanding of the crop's provenance. Nevertheless, few countries are excluded from the visualisation map because the list of countries within the used software tool for visualisation does not precisely match the list of countries from the Google search engine tool. The final visualisation, presented in Figure 13, is a bar chart delineating the relationships between four specific crops and the top ten countries associated with each.

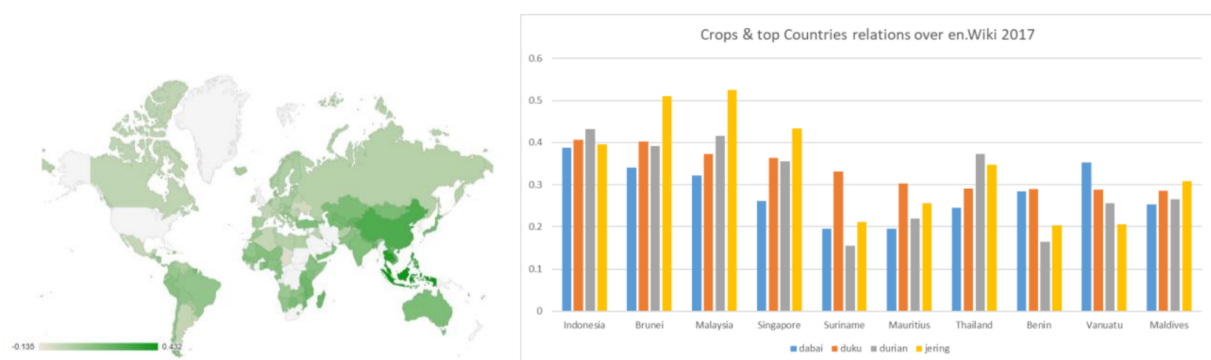


Figure 13 Graph to the left depicts the relationship between countries and Durian. The bar chart to the right illustrates the four crops with their relationship to ten selected countries.

4.3.5. Conclusion

The realm of agricultural data science has witnessed a significant evolution in methodologies for data processing and analysis. Traditional text mining techniques,

while effective for deterministic data extraction from semi-structured datasets, face challenges when confronted with vast and intricate unstructured corpora like the English Wikipedia corpus. Our research journey transitioned from these conventional methods to the more advanced [NIR](#) techniques, leveraging the capabilities of [word2vec](#). These neural network methodologies, when fine-tuned and integrated with the CropBASE knowledge system, demonstrated their prowess in analysing the extensive 3.4 billion-word English Wikipedia corpus. Not only did they identify meaningful associations between crop names and countries, but they also provided a robust tool for researchers navigating large text corpora. The challenges faced, such as vocabulary limitations, were addressed by training on updated Wikipedia editions, showcasing the adaptability and potential of [NIR](#) in the ever-evolving landscape of data science. As data continues to grow exponentially, the importance of such advanced methodologies in extracting valuable insights cannot be overstated.

In our research, we embarked on a comprehensive evaluation of text mining and [NIR](#) methodologies to extract deeper insights from extensive data corpora, particularly in taxonomy and ecology. Utilizing text mining, we developed a deterministic algorithm that efficiently navigated a 1.2-million-word corpus, addressing significant challenges such as data deformities and special characters. This process involved creating PHP-based tools for numerical data extraction and Linux shell scripts for textual data, eventually storing the cleaned and categorized data in the CropBase knowledge system. Concurrently, our [NIR](#) approach involved training a [word2vec](#) model on the English Wikipedia corpus to capture semantic nuances between crop names and related concepts. The initial use of a pre-trained model from the 2015 Wikipedia dictionary revealed substantial challenges, such as an 80% vocabulary absence and case sensitivity issues. To overcome these limitations, we trained a new [word2vec](#) model on the 2017 Wikipedia edition, utilizing a specialized Google Cloud Machine with significant computational resources. This effort resulted in a 30% vocabulary expansion and improved semantic analysis accuracy. Our findings highlight the transition from traditional text mining techniques, effective for deterministic extraction from semi-structured datasets, to advanced neural network methods capable of handling large, unstructured corpora. The adaptability and continuous refinement of these methodologies are crucial in the ever-evolving field of data science, particularly for agricultural data processing and analysis.

This research undertook a thorough evaluation of text mining and [NIR](#) techniques to extract valuable insights from large data corpora. Through text mining, we created a deterministic algorithm capable of efficiently navigating a 1.2-million-word corpus, addressing significant challenges like data deformities and special characters. This involved developing PHP-based tools for extracting numerical data and Linux shell scripts for textual data, with the cleaned and categorized information stored in a relational knowledge-based system. Simultaneously, our [NIR](#) approach trained a [word2vec](#) model on the English Wikipedia corpus to capture semantic nuances. Initially, using a pre-trained model from the 2015 Wikipedia dictionary presented issues such as substantial vocabulary gaps and case sensitivity problems. To mitigate these, we trained a new [word2vec](#) model on the 2017 Wikipedia edition, using a high-performance Google Cloud Machine. This resulted in a 30% vocabulary expansion and improved semantic analysis accuracy. Our study highlights the evolution from traditional text mining methods, suitable for deterministic extraction from semi-structured data, to sophisticated neural network approaches that handle large, unstructured corpora effectively. The continuous improvement and adaptability of these techniques are vital in the field of data science for efficient information retrieval.

The advancement of data science, especially in understanding crop ecology and taxonomy, has been remarkable. Starting with a manual collection of over 2,355 data pieces, we have seen the potential of computational algorithms, as highlighted by various researchers, in transforming this raw data into meaningful insights. The use of the [word2vec](#) model on the large Wikipedia dataset has proven particularly useful in identifying associations between crop terminologies and global dynamics. The various visualisations provided a clearer and more intuitive understanding of these complex relationships. The experiment results show promise of accurate word embedding techniques that are able to capture the relationship between crops and countries as demonstrated in the experiment result section. In this experiment, the results were compared to data from selected regions known for cultivating [BG](#), which are well-documented by the [CFF](#) Research Centre (Gregory et al., [2019](#)). This comparison yielded new insights previously unrecognized in regions like the United States regarding their relationship to [BG](#). This area has subsequently been investigated further by researchers. Information on underutilised crops is a scarce resource due to their limited popularity and usage. The aim of this research is to extract valuable information and compare it to the limited knowledge available from [CFF](#). In the

subsequent experiment, the scope will be broadened to include both scientific and vernacular names. This will facilitate evaluation using the [FAO ECOCROP](#) database (Ramirez-Villegas et al., [2013](#)), which contains extensive information useful for assessing the results. The information extracted in this experiment comprises both implicit and explicit knowledge. Explicit knowledge pertains to the documented locations where the crop grows, while implicit knowledge can be inferred by comparing the characteristics of the crop to the environmental characteristics of the locations, as captured by embeddings.

As the field of data science continues to grow, we anticipate even more refined methods to emerge, enhancing our grasp on intricate topics like crop ecology and taxonomy. This research serves as both a reflection of our current capabilities and a precursor to future advancements in the domain. The aim of this study was to discern global interest in underutilised crops by analysing the association between crop and country names. A prominent limitation arose from the reliance on the English language. Since underutilised crops aren't universally recognized, their information might be published in local languages or linked with vernacular names rather than their scientific counterparts, which could substantially alter the study's outcomes. To address this, we broadened our approach to include 178 languages available on Wikipedia, enabling the embeddings to recognize associations with vernacular names. Furthermore, we extended our investigation to encompass additional crop characteristics like soil requirements, which might not be explicitly cited in existing literature. This additional measure highlighted the ability of embeddings to retrieve implicit knowledge, demonstrating their proficiency in capturing data that can be indirectly deduced from literature.

4.4. Effective [NIR](#) methods in crop characteristic identifications

4.4.1. Introduction

This study delves into the examination of agriculture and crop data. Notably, the primary crops provide the majority of the calorie intake for humans. An ongoing project at the [CFE](#) Research Centre has introduced a robust system that covers numerous variables and metadata details for each variable concerning underutilised crops. These variables span taxonomical, geographical, and data source dimensions. The system's unique feature allows for recording multiple values for a singular variable,

depending on crop type, reference, and data collection site. Data integration occurs through various means, from international databases integrated to manual data extraction from scientific sources. The challenge comes in the transformation process to convert written content into structured data. For instance, collecting data on underutilised crops from academic papers requires significant effort. [NIR](#) techniques might offer efficiency, especially when paired with human supervision. However, [NIR](#)'s accuracy dips when used in unsupervised [ML](#), due to precision concerns. Nonetheless, in domains like web searches and social media, [NIR](#)'s application in recommendation systems has been successful, leading to satisfactory user experiences.

In this experiment, we investigated neural-based information retrieval techniques for agronomic research using the [word2vec](#) approach to word representations. Our goal was to discern soil requirements and both the vernacular and scientific names of underutilised crops. To this end, we assembled a comprehensive dataset from Wikipedia spanning 178 languages, totalling 72[GB](#) in size. By applying a self-supervised neural network to this data, we uncovered notable insights about soil texture preferences for various crops. The experiment leveraged a multi-lingual Wikipedia corpus from 178 languages to create word embeddings using the [word2vec](#) model, identifying relationships between entities like countries, vernacular names, and soil compositions. Text data was extracted from XML files, processed through [NLP](#) techniques, and validated both automatically and manually to ensure high-quality embeddings. These embeddings were used to predict and validate crop soil requirements, which were then integrated into a relational database for further research. The next section will provide a detailed description of the experiment design. We then compared our trained neural model results with the [FAO](#) ECOCROP database (Ramirez-Villegas et al., [2013](#)) to measure the performance of the model and measure its ability to extract crop characteristics which will be validated against a crop knowledge database. This study underscores the pivotal role of hyperparameter fine-tuning in optimising word representations.

4.4.2. Method

This methodology investigates our approach of leveraging [word2vec](#) on Wikipedia's large corpus spanning 178 languages. The overarching goal was to foster a more holistic comprehension of crop-related intricacies and draw connections between

crops and various related facets, such as regional vernacular names, scientific names, soil types, and associated countries. This section elucidates our multifaceted strategy, from data extraction and consolidation to the intricacies of our training procedure, and the application of our trained model to unearth patterns within the realm of crop cultivation.

Word2vec experiment with agriculture data

In our previous research, we undertook two experiments to delve into the geographical consumption patterns of underutilised crops. The initial experiment harnessed data from the Google search engine, specifically from the Google AdWords tool, spanning two years. This data encompassed global users' searches for specific crops, incorporating both scientific and vernacular names. By employing text mining and Google Maps, we analysed and visualised 300,000 search events, aiming to discern global crop trends through time series analysis. We integrated data from international databases to compile an exhaustive list of 30 names for a selected crop as indicated in Table 1. However, processing Google AdWords data proved labour-intensive, introducing potential human errors and necessitating numerous adjustments to ensure accuracy.

Our subsequent experiment broadened the scope, transitioning from merely search engine data to encompassing the entirety of the English Wikipedia corpus. This expanded dataset offered a novel perspective, shedding light on the common knowledge surrounding these crops. The second experiment, utilising [NIR](#), generated a vector space from the English Wikipedia corpus. This shifted our focus from merely vernacular knowledge to a holistic understanding of crop-related topics. Relying solely on the English Wikipedia corpus, however, posed a limitation. Given that many underutilised crops are primarily cultivated and used by indigenous communities removed from mainstream agriculture, much of this knowledge is likely documented in languages other than English.

Consequently, the primary aim of our study was to broaden our knowledge base to include non-English languages. To this end, we explored the Wikipedia corpus across 178 languages, creating an expansive vector space that encapsulated all these languages. Vectors then are retrained on a comprehensive Wikipedia corpus encompassing 178 languages to extract novel insights into underutilised crops. Utilizing the [word2vec](#) model on this extensive multi-lingual dataset enables the

identification of relationships between diverse entities. This approach significantly enhances the understanding of soil requirements for various crops, facilitating more accurate predictions and bridging the gap between theoretical knowledge and practical applications.

Training word2vec on Wikipedia all languages

Our training procedure was executed in two distinct phases. Initially, we extracted the textual content from XML files across all languages. Subsequently, we integrated the processed textual content from these languages to formulate a vector space model, which would serve as the foundation for our subsequent analyses.

The methodology, though straightforward, proved to be efficacious:

1. We commenced by extracting the textual content from the Wikipedia corpus spanning 178 languages.
2. Utilising Linux shell scripting, we merged all the resultant text files, culminating in a consolidated file size of 72GB.
3. The subsequent challenge entailed training the [word2vec](#) model on this substantial 72GB dataset.
4. Our computational infrastructure comprised a Dell PowerEdge R730, equipped with a 16-core [CPU](#) and 128GB RAM, operating on Centos 6.5 Final.
5. The training process reached completion in a duration of 29 hours and 13 minutes.

Creating a test set from text mining on ECOCROP

The [word2vec](#) training across 178 Wikipedia languages was executed using a self-supervised approach with a two-layer neural network. In this context, "self-supervised" denotes that the model generates output labels directly from the raw text, such as central words within a context window, eliminating the need for manual labelling—a process that would be exceedingly resource-intensive for this scope. The resultant vector space, derived from the [word2vec](#) training across all Wikipedia languages, offers the capability to ascertain relationships between diverse entities. In our study, we leveraged this vector space to unveil connections between crops and various related entities, including countries, vernacular names and soil compositions. The Algorithm 1 delineates the procedure to process the data, subsequently generating a test set, similarity index, and ranking accuracy for evaluation.

We focused on three primary soil textures: clay, loam, and sand. These terms were selected due to their prevalent use in characterising crop soil texture prerequisites. The process involved iterating over 2,201 crops, considering both their scientific and vernacular English denominations. The pre-trained [word2vec](#) model was then employed to predict similarity. These predictions were processed by the [word2vec](#) model trained on all 178 languages, yielding an array of prediction values ranging between 0 and 1. We juxtaposed these values with data sourced from [FAO ECOCROP](#) (Ramirez-Villegas et al., [2013](#)), which categorises optimal soil textures into heavy, medium, and light levels. We interpreted "heavy" as soils predominantly composed of clay, "medium" as those with a loam or silty texture, and "light" as those rich in sand, as per the Food and Agriculture Organization's guidelines.

For instance, consider the prediction for *Abelmoschus moschatus*:

[*"Clay:"* 0.10494453, *"Loam"* 0.13737687, *"Sand:"* 0.00425547]

This prediction elucidates the affinity between the crop's name (either scientific or vernacular English) and the soil types. The relational value is most insightful when assessed comparatively. In the provided example, the "Loam" soil exhibits the highest cosine similarity with the crop, surpassing the values for "Sand" and "Clay". This suggests that the "Loam" soil category is likely the most suitable for *Abelmoschus moschatus*. It is imperative to emphasise that the absolute value of cosine similarity is not inherently informative when considered in isolation. The cosine similarity serves to quantify the proximity between two entities. For a more nuanced understanding, one might assert that entity X exhibits a closer similarity to entity Y than it does to entity Z. In culmination, we devised an algorithm to autonomously validate predictions against the ECOCROP dataset (Ramirez-Villegas et al., [2013](#)).

Algorithm 1 Extracting training and testing set.

```
1: procedure EXTRACTNAMESANDCLASSES
2:   Initialize TextMiningTool
3:   Initialize RegularExpressionTool
4:   Open EcocropData from “aymansalama/text-mining-ecocrop” on
      Github
5:   for each line in EcocropData do
6:     ScientificNames  $\leftarrow$  Use TextMiningTool to extract
7:     EnglishCommonNames  $\leftarrow$  Use TextMiningTool to extract
8:     SoilClasses  $\leftarrow$  Use RegularExpressionTool to extract
9:     Store ScientificNames, EnglishCommonNames, SoilClasses in Rela-
      tionalDatabase
10:  end for
11:  Close EcocropData
12:  Print “Extraction Complete”
13: end procedure
14: procedure CREATETESTINGSETS
15:   for each Soil Class name do
16:     Create a line like (English common name, Soil class)
17:     Create a line like (Scientific name, Soil class)
18:   end for
19: end procedure
20: procedure CALCULATESIMILARITY
21:   Load the vector space of word2vec for all Wikipedia languages to mem-
      ory
22:   for each line in the two sets (Crop name: [English common name, Sci-
      entific names], Soil class) do
23:     Measure the similarity between the two entities (Crop name, Soil
      Class)
24:   end for
25:   Sort all similarities by the highest score
26:   Convert the distance into ranks, e.g., [Abelmoschus moschatus, “Clay:”
      2nd, “Loam” 1st, “Sand:” 3rd]
27:   Validate the predicted class against the recorded soil class from Ecocrop
      data
28: end procedure
```

Algorithm 1 Extracting training and testing set.

4.4.3. Experimental design

This experiment dives deeper by incorporating the Wikipedia corpus across a large corpus of 178 languages. The primary methodology includes the extraction of textual content from XML files of all these languages, leading to the formation of an expansive vector space model—a crucial foundation for subsequent in-depth analyses. Employing the [word2vec](#) training model on this multi-lingual dataset has bestowed the

capability to discern relationships between an array of entities, ranging from countries and vernacular names to soil compositions. A significant emphasis has been placed on understanding the soil texture prerequisites for different underutilised crops, considering both their scientific and vernacular English names. The goal is to predict crop soil requirements and autonomously validate these predictions against an established dataset, thereby bridging the gap between theoretical predictions and practical applications.

The design of this experiment integrates methodologies akin to those of the prior experiment. However, the underlying procedures and the coding architecture have been entirely revamped to cater to the 178 languages and to align with the recent Wikipedia file format modifications. Additionally, advancements in pre-processing methods were incorporated, and contemporary libraries were utilised, contributing to the enhanced efficiency of both the scripts and the resultant model. The process of text mining initiates hands-on data compilation by researchers. Information is obtained from online platforms, including online databases such as ECOCROP (Ramirez-Villegas et al., [2013](#)) that emphasise crop characteristics, tables, and [CSV](#) records. After the integration and refinement of this data, the accumulated data is then segmented into two categories: tabulated records and semi-structured content. Two separate procedures for data processing were instituted. The first concentrates on tabulated records, ensuring its precision, management, and readiness for integration into a relational database. An intermediary review stage confirms data accuracy. If discrepancies are identified, manual management is undertaken. The second procedure, which relies on shell scripting, supervises the text data, focusing on text refinement, character handling, and the extraction of particular crop knowledge to make it suitable for database integration. Any non-conforming text data is preserved in a polished collection for the subsequent [NIR](#) workflow. Similar to the first approach, a validation mechanism is established, allowing for manual corrections when necessary.

The experiment leverages a range of contemporary libraries and processes to handle the extensive multilingual dataset from Wikipedia. Libraries such as NLTK for natural language processing tasks like tokenization, stemming, and lemmatization, Gensim for training the [word2vec](#) model, Pandas for managing tabulated records, BeautifulSoup for extracting text from XML files, and NumPy for numerical computations were utilised. The text refinement process involves cleaning the text by

removing noise and irrelevant data through tokenization and lemmatization. Character handling ensures proper interpretation and conversion of special characters and diacritics across the 178 languages. Shell scripting automates data processing tasks, focusing on text refinement, character management, and extraction of crop-related information, ensuring the data meets the required format for MySQL database integration. A validation mechanism verifies the processed data, allowing for manual corrections to maintain accuracy and reliability before further analysis or database integration. These combined methodologies enhance the data quality and facilitate accurate predictions and analyses of crop soil requirements.

The [NIR](#) methodology utilises word embedding mechanisms to generate high-quality embeddings, extracting both implicit and explicit knowledge from the pre-existing collection. This begins by procuring Wikipedia files in XML format, spanning across 178 languages available during the study. The transformation from XML to textual format was achieved using Python. Successive [NLP](#) operations were applied, including tokenization, stemming, lemmatization, and assorted techniques for data refinement. Once the data is primed, it transitions to the [word2vec](#) training phase. Before training commences, data is subjected to a verification and curation examination. If anomalies are identified, manual corrections are mandated; otherwise, it advances to the concluding repository ready for training. The training stage includes selecting the appropriate hyperparameters. The training dataset integrates the Wikipedia corpus in 178 languages with the purified data from the initial text mining procedure. Once embeddings are formulated, a validation phase begins, contrasting specific database elements with their corresponding embeddings and gauging their congruence. The established associations are juxtaposed with data from the initial text mining phase that's archived in the relational database to determine the efficacy of word embeddings. The insights gleaned are graphically represented in this chapter's outcomes. Ultimately, data is channelled into a database, available for researcher consultation and subsequent investigations. Similar to the previous experiment, a clear separation exists between manually accrued details and data extracted through [ML](#), a pivotal aspect for preserving data governance and credibility. Figure 14 illustrates the process of text mining and [NIR](#) of 178 languages of Wikipedia to extract insights about crop attributes. To accurately visualize the relationships and embeddings in high-dimensional word vectors, various dimensionality reduction techniques were employed. Among these, [PCA](#) was utilised to effectively reduce the dimensionality of

the word embeddings, enabling them to be visualized and analysed. This reduction process preserves the essential patterns and structures within the data, allowing for a clearer interpretation of the relationships between different word vectors. Other advanced techniques, such as [t-SNE](#) or Uniform Manifold Approximation and Projection, can also be used to enhance the visualization and analysis of these embeddings, providing deeper insights into the semantic and syntactic properties of the words.

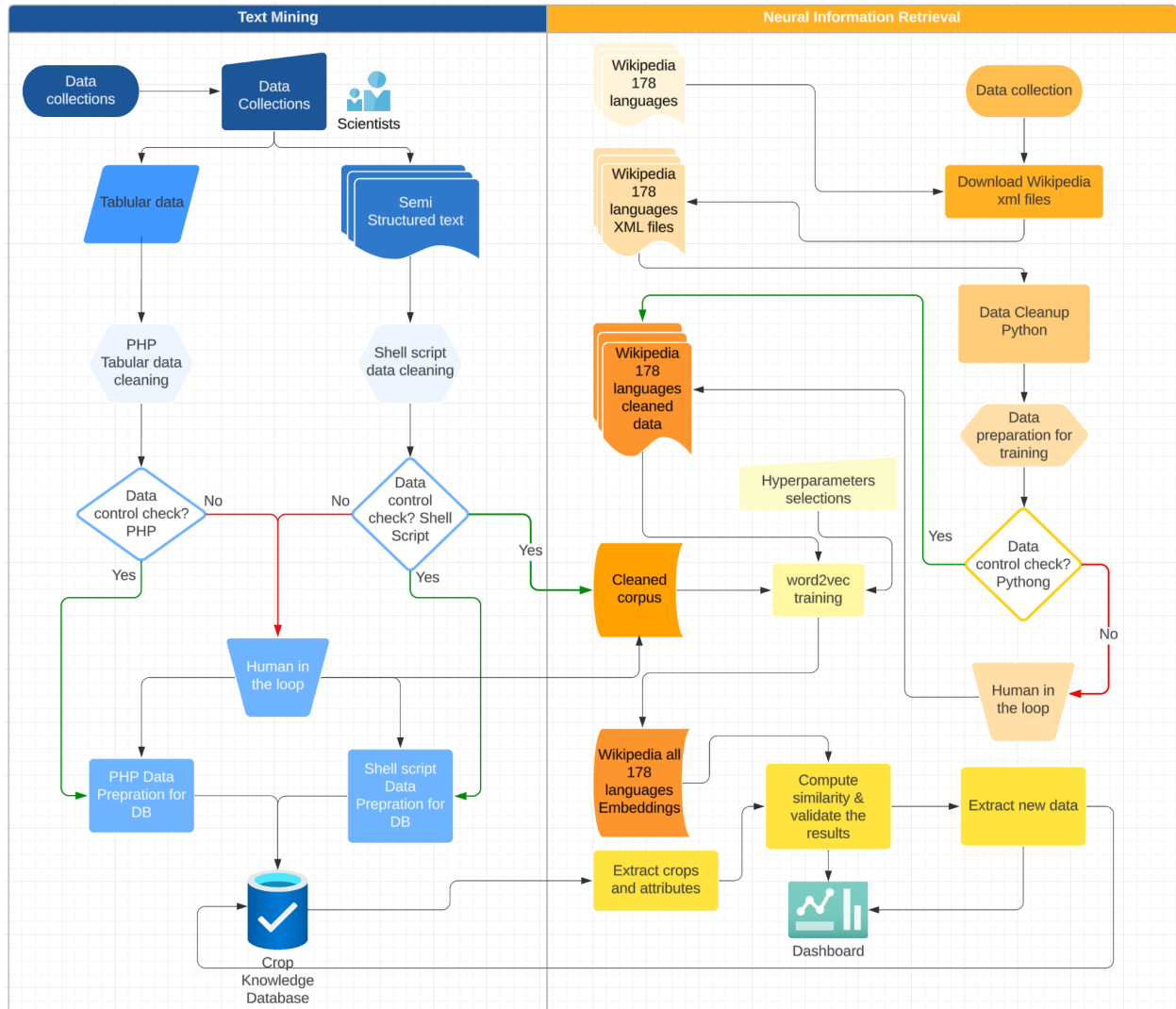


Figure 14: Text mining and [NIR](#) of 178 languages of Wikipedia to extract insights about crop attributes.

4.4.4. Results and discussion

This study seeks to assess the capability of [NIR](#) to acquire quantitative information suitable for various applications, including extraction and categorization tasks. Such an approach has the potential to minimise the human intervention required in the data-gathering phase. In evaluating the prediction capabilities of the trained model, we

compared its outcomes with the ECOCROP database (Ramirez-Villegas et al., [2013](#)) from [FAO](#). The achieved accuracy rate stood at 76.11%. These findings imply the effective utility of [NIR](#) in data acquisition. However, it's crucial to emphasise that [NIR](#)'s intended function isn't to supplant human involvement in data gathering. Instead, it serves as an auxiliary tool aimed at streamlining the data collection process. This model boasts the capability to process vast numbers of documents, subsequently offering a visual representation of quantitative data analytics. Such insights can guide and refine the direction of data gathering activities.

Traditional text mining methods, which employ regular expressions to extract knowledge from semi-structured data, are typically deterministic. In this study, we analysed 691K words from agronomic datasets (as described earlier e.g., [BG](#) dataset), successfully extracting attributes and characteristics about 2.2K crops such as 13K local names and 1.7K synonyms. The extracted data are compared against ECOCROP database (Ramirez-Villegas et al., [2013](#)). These text mining methods achieved a 97% accuracy rate, verified against a 20% sample subset. Yet, analysing 691K words is relatively straightforward compared to the challenge of processing 3.4 billion words, as seen in the English Wikipedia corpus. The feasibility of the text mining on the 691K-word dataset was due in part to its semi-structured nature, although inconsistencies in format were observed. The semi-structured dataset consists of textual data saved as text. However, the authors arranged it using a particular sequence and distinct separators, facilitating the extraction of specific values through text mining. Text mining became essential to transform this semi-structured data into coherent knowledge.

In the experiment, the primary variables under assessment were the crop identifier and the classification of soil. A crop's identifier might be articulated either through its scientific or vernacular names. The findings underscore that crop identifiers expressed in scientific names hold more robust data representation than their vernacular counterparts. This observation aligns with the prevalent understanding that scientific terminologies are deployed with greater precision in academic literature. An assessment of the accuracy rate for soil classification, when reliant on vernacular crop names, stood at 49.97%. This is notably inferior to the accuracy levels achieved when utilising scientific terminologies. Figure 15 offers a snapshot, showcasing ten selected scientific crop names compared against soil classifications. These comparisons leverage cosine similarity metrics, where the value indicates an accurate match in soil

classification. The model was trained using the [word2vec](#) algorithm on a multilingual dataset derived from Wikipedia XML files across a diverse range of languages. Input data encompassed textual content from Wikipedia and the ECOCROP database, with the objective of predicting crop soil requirements and validating these predictions against established data sources. The training process involved various natural language processing techniques such as tokenization and lemmatization, with carefully selected hyperparameters to enhance performance. Evaluation of the results involved comparing the model's predictions with the ECOCROP database, revealing significant accuracy. Moreover, soil classification accuracy was higher when using scientific crop names compared to vernacular names, assessed through cosine similarity metrics.

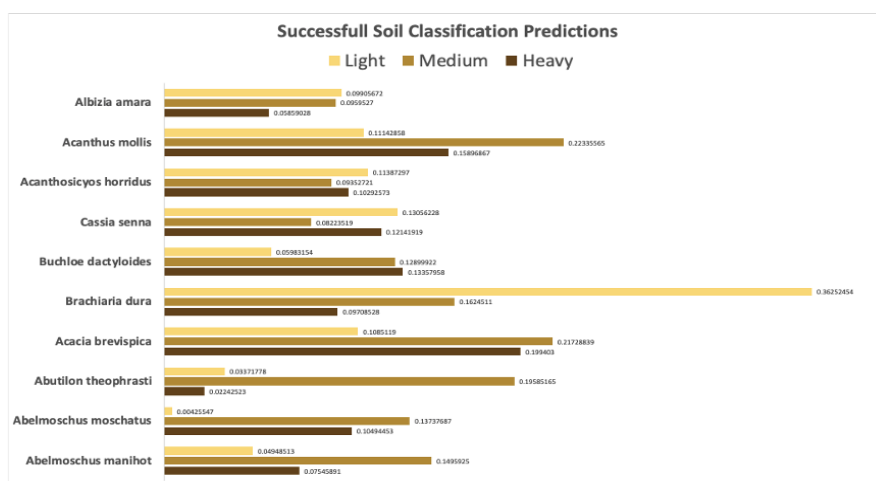


Figure 15: Comparison of Cosine Similarity for Soil Classification with a Selection of Ten Crops.

The initial crops dataset that we prepared from our sources encompasses 2021 distinct crop identifiers. Utilising the [word2vec](#) training model, 1997 scientific names for these crops were successfully extracted from the total of 2021. It's plausible that the crop names absent from the model's vocabulary can be attributed to their infrequent appearance within the reference corpus. The predictive model adeptly identified the appropriate soil category for 1520 crops but fell short with 477 crops. This translates to a success ratio of 76.11%. Table 3 presents a side-by-side comparison: soil classification predictions are delineated on the left, while data from the [FAO](#) ECOCROP (Ramirez-Villegas et al., [2013](#)) is displayed on the right. For clarity, only a subset of ten crop examples is portrayed, illustrating the model's accurate predictions compared against the ECOCROP database. We compared the

results of the trained model with the [FAO](#) ECOCROP data (Ramirez-Villegas et al., [2013](#)) to validate the accuracy of the predicted soil texture preferences for various underutilised crops. For instance, the analysis of *Abelmoschus moschatus* emphasised the importance of using relative cosine similarity values to determine optimal soil types. We explored various mathematical techniques to match crop scientific and vernacular English names with soil types within the vector space of word embeddings. Our findings revealed an accuracy of 76.11% when matching soil classifications with crop scientific names. In contrast, using vernacular English names yielded an accuracy of 49.97%. For instance, the predictions for *Abelmoschus manihot* and *Abelmoschus moschatus* aligned perfectly with the [FAO](#) ECOCROP data, both correctly identifying medium soil texture. Similarly, *Abutilon theophrasti* was accurately matched to medium soil texture, confirming the reliability of our method. However, there were some discrepancies, such as the prediction for *Brachiaria dura*, which did not match the [FAO](#) ECOCROP data, indicating light soil texture instead. The results were compared to EcoCrop database to validate and confirm the findings. EcoCrop database is the benchmark baseline that the researchers are using in Crops For The Future Research Centre. For instance, the predictions for *Abelmoschus manihot* and *Abelmoschus moschatus* aligned perfectly with the [FAO](#) ECOCROP data, both correctly identifying medium soil texture. Similarly, *Abutilon theophrasti* was accurately matched to medium soil texture, confirming the reliability of our method. However, there were some discrepancies, such as the prediction for *Brachiaria dura*, which did not match the [FAO](#) ECOCROP data, indicating light soil texture instead. These results underscore the efficacy of [NIR](#) for knowledge extraction and highlight the superior accuracy achieved using scientific names over vernacular names in this context. In summary, our research demonstrates the potential of [NIR](#) in agricultural studies. This approach offers a promising avenue for efficiently predicting crop soil requirements, underscoring the value of word embeddings in academic research. Our forthcoming experiment will primarily concentrate on strategies to derive high-quality word embeddings for downstream tasks by carefully crafting hyperparameter tuning experiments.

Table 3: Predicted Values Compared to [FAO ECOCROP](#) Data. Bolded values highlight the top predictions that align with the actual [FAO ECOCROP](#) figures.

Scientific name	Prediction			FAO ECOCROP Soil Texture		
	Clay	Loam	Sand	Heavy	Medium	Light
<i>Abelmoschus manihot</i>	0.075	0.149	0.049	0	1	0
<i>Abelmoschus moschatus</i>	0.104	0.137	0.004	0	1	0
<i>Abutilon theophrasti</i>	0.022	0.195	0.033	0	1	0
<i>Acacia brevispica</i>	0.199	0.217	0.108	0	1	0
<i>Brachiaria dura</i>	0.097	0.162	0.362	0	0	1
<i>Buchloe dactyloides</i>	0.133	0.128	0.059	1	0	0
<i>Cassia senna</i>	0.121	0.082	0.130	0	0	1
<i>Acanthosicyos horridus</i>	0.102	0.093	0.113	0	0	1
<i>Acanthus mollis</i>	0.158	0.223	0.111	0	1	0

4.4.5. Conclusion

This study explored the use of word embeddings, specifically the [word2vec](#) model, to predict soil requirements and to identify the vernacular and scientific names of underutilised crops. Using data from Wikipedia across 178 languages, we created a 72GB dataset and processed it through self-supervised word embeddings. "The embedding, created with all 178 languages, effectively captured the relationship between crop vernacular names and scientific names in different languages, unlike the embeddings created solely with English Wikipedia. The results highlighted the superior accuracy of predictions using crop scientific names compared to their common English names.

The research tackled significant challenges and introduced several innovations in leveraging [word2vec](#) for agricultural data analysis. One major challenge was the labour-intensive and error-prone process of handling Google AdWords data, which required extensive manual adjustments to ensure accuracy. The initial reliance on the English Wikipedia corpus also limited the scope of the study, as crucial information about underutilised crops is often found in non-English languages. Additionally, training the [word2vec](#) model on a substantial 72GB dataset demanded considerable computational power, utilizing a Dell PowerEdge R730 server. There was also a notable disparity in prediction accuracy when using scientific names compared to vernacular names, with scientific names proving more reliable. To overcome these challenges, the research implemented several innovations. It expanded the dataset to

include Wikipedia articles in 178 languages, creating a more inclusive vector space model. The methodology involved extracting and merging text from these languages and training the [word2vec](#) model on this large corpus, effectively capturing relationships between crop-related entities. The use of a self-supervised learning approach with a two-layer neural network eliminated the need for manual labelling, enhancing efficiency. An algorithm was developed to validate model predictions against the FAO ECOCROP dataset, ensuring the accuracy of soil texture predictions. Furthermore, hyperparameter optimization through random search improved the quality of word embeddings, which were then integrated into various deep neural network classifiers. This research highlights the potential of utilizing multilingual word embeddings and advanced machine learning techniques in agricultural studies. The innovations in data integration, model training, and hyperparameter optimization addressed significant challenges, enhancing the accuracy and applicability of the findings.

In this study, we assessed the potential of [NIR](#) in the realm of data acquisition. A [word2vec](#) model was trained on the exhaustive Wikipedia corpus, spanning 178 languages. Leveraging this trained model, predictions were made concerning soil requirements for various crops, which were subsequently validated against the ECOCROP (Ramirez-Villegas et al., [2013](#)) soil classification database for 2021 crops, resulting in an accuracy rate of 76.11%. It is imperative to underscore the significance of terminology selection for optimising prediction accuracy. For example, adopting common English names in lieu of scientific crop names considerably diminishes the prediction's accuracy. These outcomes suggest that [NIR](#) could serve as a powerful tool for automated data collection, particularly in the creation of structured scientific databases. Understanding the advantages of utilising high-quality word embeddings is essential for the accuracy of [NIR](#) tasks, which aim to extract both implicit and explicit knowledge from literature. While these embeddings significantly aid in extracting information beyond just explicit crop characteristics, there is still a pressing need for more detailed research to enhance performance and achieve greater accuracy. This chapter underscored the significance of hyperparameter selection manifested in the experimental designs. Guided by recommendations from existing literature, we chose hyperparameters tailored specifically for tasks involving interchangeability and relatedness. We also conducted an experiment to confirm our selection of word embedding hyperparameters. This helped extract both vernacular and scientific

names of underutilised crops, as well as comprehending the relationship between crops and global interest. This includes crop attributes extraction like soil requirements for different underutilised crops. It's imperative that we deepen our understanding of how the hyperparameters of word embeddings influence varied downstream tasks. These include tasks like relatedness, interchangeability, and their effect on document classification when integrated into the embedding layers of different [DNN](#) architectures. Such insights are vital for our experiment on the temporal knowledge evolution framework. This is essential to pre-process data before the creation of the continuum vector space, the foundational element of our framework.

Comparing the results obtained in these results by Jatnika et al. ([2019](#)). The methodologies of this study and those employed by Jatnika et al. exhibit several key differences. This research utilised a multilingual dataset derived from Wikipedia XML files and the ECOCROP database, specifically targeting crop soil requirements. In contrast, Jatnika et al. used a large corpus of English Wikipedia articles to address general semantic similarity tasks. Regarding model and techniques, this study trained [word2vec](#) on a specialized multilingual agricultural dataset, emphasizing the prediction of crop soil requirements. Techniques such as tokenization, lemmatization, and hyperparameter tuning were implemented to optimize performance. Jatnika et al., however, applied [word2vec](#) using [CBOW](#) and Skip-Gram architectures, exploring various configurations of window sizes and vector dimensions to identify the best semantic similarity measures.

This research achieved a 76.11% accuracy in predicting soil classifications when using scientific crop names, significantly outperforming the 49.97% accuracy obtained with vernacular names. Conversely, Jatnika et al. reported Pearson correlation values of 66.5% for WordSim-353 and 28.4% for SimLex-999, indicating moderate to weak correlation in word similarity tasks.

This study demonstrates several advantages over the work by Jatnika et al. Firstly, the domain-specific focus on agriculture significantly enhances the practical utility and relevance of the findings. By directly comparing model predictions with the ECOCROP database, this research offers targeted insights applicable to real-world agricultural practices. In contrast, Jatnika et al.'s research addresses general semantic similarity, which, while broader, is less tailored to specific applications. Secondly, the high accuracy (76.11%) achieved in the practical task of soil classification for crops highlights the effectiveness of the model in real-world scenarios. Although Jatnika et

al. achieved a good Pearson correlation in word similarity tasks, their model's applications are more generalized and not as focused on specific predictive tasks as this research.

4.5. Chapter Conclusion

In conclusion, across multiple experiments, we have ventured into the multifaceted realm of utilising data science, big data, and [ML](#) techniques to deepen our understanding of global interest in underutilised crops and enhance our insights into crop ecology, taxonomy, and related data acquisition. By tapping into online search engine data, we unveiled patterns in [BG](#) search behaviours that informed probable cultivation regions and characteristics of its value chain, particularly spotlighting interest in countries like Indonesia, Thailand, the U.S., among others. This study underscored the potential of search data in mapping cultivation patterns and deciphering research interests on a global scale. Elevating our perspective to a broader scope, we now turn our attention to [ML](#) and [NIR](#) applications, we managed to extract cultivation patterns and global interest in these underutilised crops. Our subsequent experiments showcased the promise of [ML](#), with the employment of word embedding techniques, particularly [word2vec](#), to map the semantic distribution of knowledge. Leveraging a voluminous Wikipedia dataset, we harnessed the [word2vec](#) model to identify associations between crop terminologies and global dynamics, offering a clear visualisation of these intricate relationships. The reliability of word embedding techniques in capturing relationships between crops and countries was evident, and the ensuing experiment expanded the scope to analyse all available languages within Wikipedia. This approach enriched our understanding of crop-specific characteristics, including soil requirements, and enabled us to pinpoint both vernacular and scientific names for various crops. Furthermore, our exploration of [NIR](#) as a tool for data acquisition showed significant promise. By training a [word2vec](#) model on a comprehensive Wikipedia corpus spanning 178 languages, we predicted soil requirements for multiple crops with a good accuracy rate. However, the significance of precise terminology selection was emphasised, given its impact on prediction accuracy. The potential of [NIR](#) in automated data collection and the creation of structured scientific databases became evident, signalling future advancements in this

direction. The experiments detailed in this chapter demonstrate how text mining techniques and word embeddings can effectively unveil the potential for implicit and explicit knowledge extraction. The explicit information regarding crop scientific and vernacular names, as well as cropping patterns and cultivation locations, is readily available in the literature. The experiments utilizing embedding techniques successfully extracted this knowledge, which was then compared to international crop databases to quantify the accuracy of the methods. Conversely, the soil requirements for underutilised crops are often not explicitly mentioned in the literature. These requirements are implicitly conveyed through the use of different synonyms or by referencing soil characteristics without explicitly listing the soil type. The experiments involving word embeddings across Wikipedia data and scientific literature for agriculture demonstrate the ability of these embeddings to capture such implicit information. The extracted information was then compared with international databases, as indicated in the results presented earlier. In essence, the convergence of our experiments highlights the transformative power of data science in understanding and promoting the sustainable cultivation of underutilised crops. As we look to the horizon, we envision an era where refined computational algorithms and enhanced [ML](#) models will further advance our grasp on crop ecology, taxonomy, and the broader challenges of food security and climate resilience.

Chapter 5

Embedding hyperparameters

5.1. Introduction

In textual representation, [word2vec](#) stands as a critical method for creating vector spaces from text datasets, with its success dependent on precise hyperparameter tuning. Notably, the dimensionality of word embeddings plays a pivotal role, as it can affect both semantic detail and computational feasibility. Likewise, the window size determines the semantic context, with potential implications for noise from unrelated terms. Further, the learning rate influences the speed and quality of model training. With other hyperparameters like subsampling rate and word frequency threshold also impacting performance, it's evident that researchers must thoroughly adjust these factors according to specific needs.

In the previous chapters we explored effective methods for extracting explicit data from scientific literature. Initially, we employed text mining and utilised search engine data to identify cropping patterns for underutilised crops. We then applied word embedding techniques, training them on the Wikipedia English corpus, to investigate global interest in these underutilised crops. Our research expanded to include training embeddings across multiple languages, allowing us to gather details about these crops, such as cropping patterns, vernacular and scientific names, and specific crop characteristics such as crop soil requirements. Through these experiments, we discerned the significant impact of hyperparameters on the quality of the generated embeddings. This quality directly influences downstream tasks like [NIR](#), classification, and even the temporal knowledge evolution which will be discussed later in the thesis.

The central focus is on optimising hyperparameters for the [word2vec](#) model to yield optimal word embeddings. These embeddings will then be tested using various [DNN](#) architectures, including [RNN](#), [CNN](#), and [LSTM](#), to classify Amazon reviews. Within these architectures, the embedding layer stands as a cornerstone, integrating the word embeddings. To ensure an unbiased assessment of the embedding quality, this layer's weights are kept static during training.

5.2. Method

5.2.1. Word embedding hyperparameters

In our study, we employed the widely recognized open-source Python library, Gensim, to formulate mathematical representations of textual content. Such representations typically encapsulate semantic nuances. Gensim is designed with a focus on computational efficiency and provides a range of unsupervised neural network algorithms, including [word2vec](#), Fasttext, and Latent Dirichlet Allocation. These algorithms primarily construct a statistical matrix capturing the co-occurrence relationships among words within a given corpus. Scholars and practitioners frequently utilise Gensim to discern similarities between words or entire documents. Additionally, it serves as a valuable tool for subsequent tasks like document classification and sentiment analysis. For our experiments, we adopted Gensim version 4.0.1. Subsequent sections will detail our approach to training and optimising the hyperparameters of the [word2vec](#) model.

The term "word2vec_epochs" denotes the number of iterations the model undergoes over the corpus. Typically, a higher epoch count results in word embeddings that more faithfully mirror the corpus. In the context of training algorithms, "word2vec_sg" plays a pivotal role: a value of 1 signifies the use of skip-gram, while 0 indicates [CBOW](#). The window size, on the other hand, relates to the model's sliding window on the corpus. A broader window captures an expansive context, emphasising word relatedness, whereas a narrower one homes in on word interchangeability, with 5 being the conventional window size. The vector size is instrumental in determining word vector dimensionality. A larger dimension delves into the intricate nuances of each word, while a smaller one offers a more overarching view. For visualisation, techniques like dimensionality reduction can be invaluable, compressing dimensions to a more visual-friendly 2 or 3, apt for 2D or 3D diagrams. The "word2vec_hs" parameter sheds light on whether hierarchical SoftMax is employed for training. Hierarchical SoftMax will be used for model training if set to 1. Furthermore, when the "word2vec_negative" parameter exceeds 0, it activates negative sampling, defining the number of "noise words" to be included. This method allows the model to modify a select set of weights during each training instance. To conclude, the "min_count" parameter ensures words with frequencies below a certain threshold are excluded from the model.

Given the vast number of hyperparameter combinations, exhaustive testing would be computationally intensive. Hence, a random search was used on 1000 combinations to identify optimal and suboptimal hyperparameter sets. While there are various hyperparameter optimization strategies like grid search and manual search, grid search is resource intensive. In this experiment, various hyperparameters were explored as suggested by previous research. We set ranges for Negative Sampling, Skip-Gram, [CBOW](#), and Hierarchical Softmax based on standard guidelines. Dimension Size was evaluated between 1 and 500 due to concerns about knowledge representation and computational challenges. Window Size ranged from 1 to 50 to balance term interchangeability and relatedness. We limited Epochs to 50 to balance quality and computational efficiency. Min Count was assessed up to 50 to ensure a diverse vocabulary.

Random search is a recognized method for hyperparameter tuning in [ML](#), as evidenced by several academic studies. Elgeldawi et al. ([2021](#)) conducted a comparative analysis of hyperparameter tuning techniques, highlighting the utility of random search among other methods, though the highest accuracy was achieved by the Support Vector Machine Classifier (SVC) using Bayesian Optimization with a score of 95.6208%. Rijdsdijk et al. ([2021](#)) emphasised the prevalent use of random search in the side-channel community, but their primary focus was on the potential of reinforcement learning for hyperparameter optimization. Mantovani et al. ([2015](#)) specifically investigated the efficacy of random search for tuning Support Vector Machines, finding that its predictive performance was equivalent to other techniques like meta-heuristics and Grid Search, but with a notable reduction in computational cost. Lastly, Sumathi et al. ([2020](#)) discussed the importance of hyperparameter tuning, with their primary focus on the grid search method for Random Forest classifiers.

5.2.2. Document classifiers

In the subsequent phase of our experiment, we delve into the classification task. This process involves harnessing the vector space derived from word embeddings as an initial input. We then embark on training a [DNN](#) classifier tailored for a specific classification task. Although the resulting accuracy is intrinsically linked to the [DNN](#) models, our overarching objective is to uncover any potential variations in accuracy that might emerge from alterations to the foundational word embedding layers.

Our ambition in this segment of the research is to identify the most efficient hyperparameter combination that will result in the most efficient text classifiers, placing a spotlight on the role of word embedding layers. For our classification architecture, we've opted for the [DNN](#). The vector space of word embeddings is seamlessly integrated as the first layer in a Keras model, functioning as an embedding layer. Our exploration led us to experiment with diverse network architectures, encompassing the likes of Simple [RNN](#), [CNN](#), and [LSTM](#). [RNN](#) (Williams et al., [1989](#)) are designed for sequential data processing, suitable for applications such as time series forecasting and linguistic tasks. Their architecture allows for the modelling of temporal patterns in data. Nonetheless, training [RNNs](#) on extended sequences can be challenging due to issues like the vanishing and exploding gradient. [CNN](#) (LeCun et al., [1998](#)) are predominantly used in visual data tasks, such as image recognition. They have the capability to learn spatial feature hierarchies from images, enhancing their classification performance. The shared-weight design of [CNNs](#) also promotes computational efficiency. However, their training often necessitates a significant volume of labelled data and can be computationally demanding, especially with deeper network structures. [LSTM](#) networks (Hochreiter et al., [1997](#)) are a variant of [RNNs](#) tailored to retain information over longer sequences. They are adept at recognizing long-term data dependencies, which is beneficial for tasks involving extended data sequences. While [LSTMs](#) alleviate some of the challenges faced by traditional [RNNs](#), such as the vanishing gradient problem, their more intricate design can lead to extended training durations and increased computational needs. The Sequence-to-Sequence ([Seq2Seq](#)) architecture, primarily developed for tasks such as machine translation, can be repurposed for text classification by transforming the input sequence into a condensed vector and subsequently decoding this vector into a specific category label (Sutskever et al., [2014](#)).

To carefully assess the performance of our models, we leaned on a spectrum of evaluation metrics. These included traditional measures such as accuracy, precision, recall, and the F1 score, all of which were applied to both the test and validation datasets. A subsequent illustrative Figure 16 offers a granular breakdown of the [DNN](#)'s layered structure and inherent parameters.

The initial layer, known as the Embedding Layer, is a manifestation of the vector space word embeddings crafted by [word2vec](#), as alluded to in our preliminary

discussions. Following this, we have the spatial dropout layer, a pivotal component that supports the independence among feature maps. Subsequent to this is a layer that can oscillate between being a simple [RNN](#), [CNN](#), or [LSTM](#), contingent on the specific model blueprint. The batch normalisation layer is the fourth layer. By normalising inputs for each mini batch, it paves the way for a more anchored learning trajectory, reducing the number of training epochs required for efficient networks. The dropout layer succeeds this, characterised by its strategy of sporadically dropping neurons during the training cycle. The global max pooling layer then compresses the entirety of the feature map into a singular, consolidated value. This is somewhat reminiscent of adjusting the pool_size to mirror the dimensions of the input feature map. Concluding the sequence is the dense layer, which sets the stage for the terminal output layer.

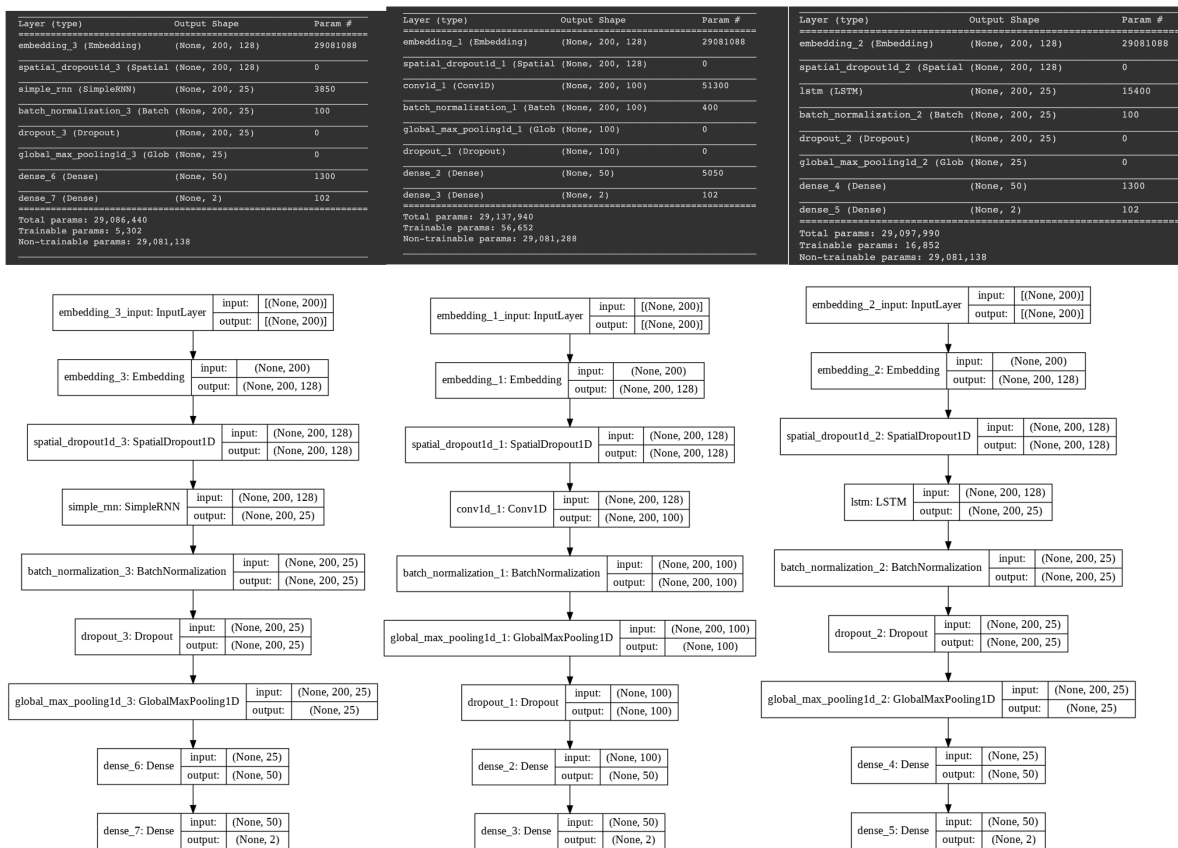


Figure 16 On the left is the Simple [RNN](#), in the center is the [CNN](#), and on the right is the [LSTM](#). The upper illustrations depict the model summaries, while the lower illustrations represent the model architectures.

5.2.3. The evaluation methods

In our experiment, we embarked on an exploration of various hyperparameters tailored for [word2vec](#). During each iteration, the resulting vector space word embedding was transformed into an embedding matrix, which was then integrated into three distinct neural networks: [RNN](#), [CNN](#), and [LSTM](#). Various accuracy metrics were carefully evaluated throughout each iteration. F1 score was utilised as a primary measure to gauge the efficacy of the models. Between precision and recall, the F1 score strikes a balance between them. Precision denotes the ratio of correctly predicted positive results to the total predicted positives, showcasing the reliability of positive predictions. Conversely, recall captures the ratio of correctly predicted positive outcomes to all actual positive cases, highlighting the model's capability to identify all pertinent cases. By amalgamating these metrics, the F1 score provides a holistic evaluation of the model, factoring in both false positives and false negatives.

5.3. Experimental design

This experiment aims to analyse the influence of [word2vec](#) hyperparameters on downstream tasks such as document classification using various [DNN](#) architectures. The objective is to generate high quality embeddings suitable for applications involving interchangeability, relatedness, and the evolution of temporal knowledge. In this segment, we dive into the specifics of the experimental methodology and its procedures. The initial step involves preparing data from the Amazon Customer Reviews Dataset. This involves various [NLP](#) tasks like tokenization, stemming, lemmatization, stop word elimination, lowercasing, punctuation and numeric removal, whitespace clearance, encoding and decoding, and spell check correction. For post-processing, the data is saved in two different formats. The first is textual data to be used for the embedding layer, while the second is a [CSV](#) format containing the reviews and their categories for different [DNN](#) models.

The second phase focuses on generating hyperparameter combinations for word embeddings. Initially, we manually set the hyperparameter ranges as specified in Table 4. Following that, we randomly generate [word2vec](#) hyperparameter combinations for each word embedding training iteration.

Table 4: List of [word2vec](#) hyperparameters and their range used in the experiment.

HYPERPARAMETER	VALUE RANGE
DIMENSION SIZE	[10:500]
WINDOW SIZE	[1:50]
NUMBER OF EPOCHS	[1:50]
NEGATIVE SAMPLING	[5:20]
SKIP-GRAM	[0-1]
HIERARCHICAL SOFTMAX	[0-1]
MINIMUM COUNT	[1:50]

The third phase is dedicated to word embedding training. This involves a loop that traverses all randomly generated [word2vec](#) hyperparameter combinations. Each cycle trains [word2vec](#) on the generated textual dataset from the first phase using the specified hyperparameter combination, resulting in an embedding. Once all iterations conclude, there's a process that integrates all embeddings to form a comprehensive list, representing the Amazon Customer Review dataset word embedding using various [word2vec](#) hyperparameter combinations.

In the fourth and final stage, we loop through the list of embeddings. For each cycle, different [DNN](#) architectures namely [RNN](#), [CNN](#), and [LSTM](#) are trained using the respective embeddings. Upon completion of training, testing, and validation for all embedding combinations, the model artefacts, inclusive of accuracy metrics, are collated and sent to a visualisation tool for graphical representation as shown in the graphs in this chapter. Figure 17 illustrates the four processes and how they interact with each other.

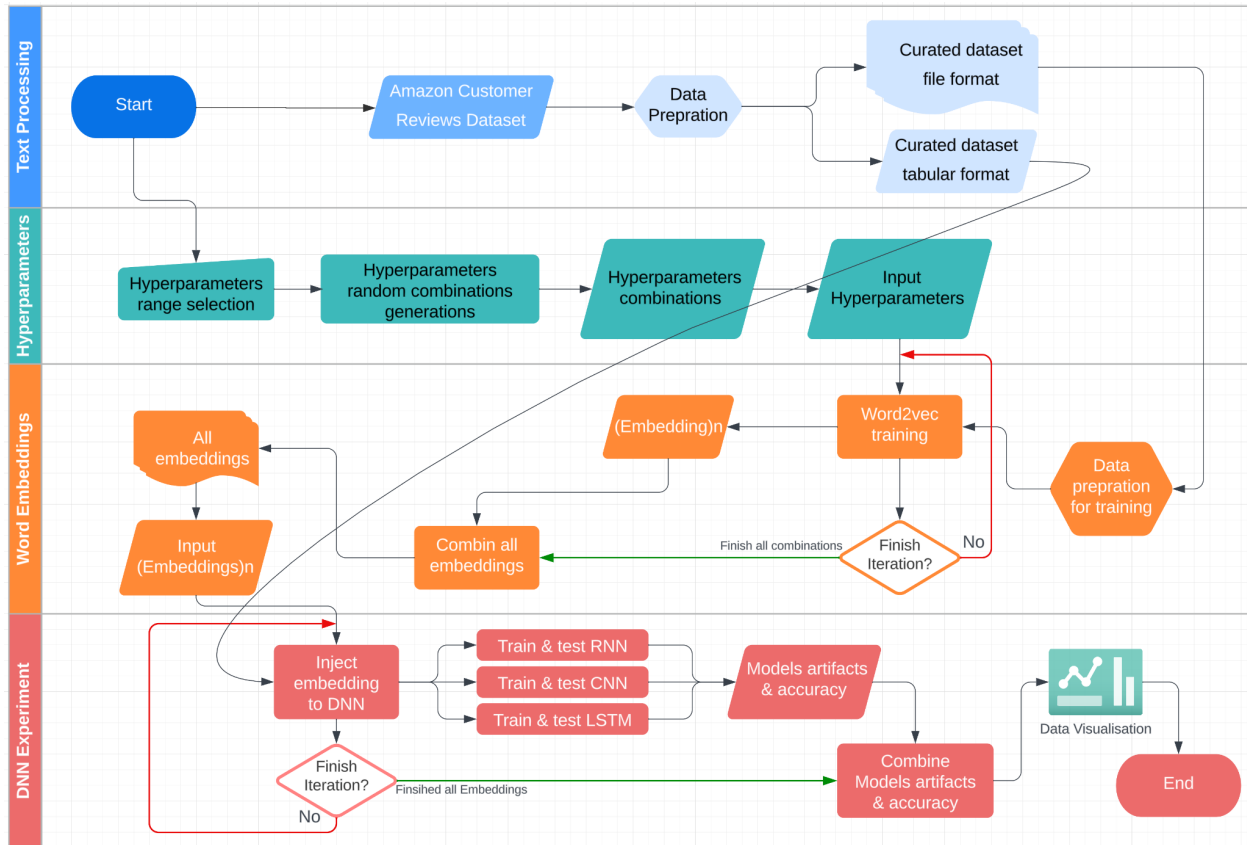


Figure 17: The process of the creation of word embedding and hyperparameters selection process and [DNN](#) architectures testing.

5.4. Results

The primary objective of this study is to pinpoint the optimal set of hyperparameters for the [word2vec](#) model. In this section, we will examine the effects of various [word2vec](#) hyperparameters on the classification performance of [DNNs](#) such as [RNN](#), [CNN](#), and [LSTM](#). As outlined in the methodology, we will test multiple randomly chosen [word2vec](#) hyperparameters. These hyperparameters are then applied to the embedding layer of the [DNNs](#). Subsequently, the [DNNs](#) are trained for the classification task and their performance is evaluated. The influence of each [word2vec](#) hyperparameter combination is further detailed in the following sections.

5.4.1. Comparative statistical analysis of F1 scores

This section evaluates the collective influence of all the hyperparameters on selected [DNN](#) architectures, specifically [RNN](#), [CNN](#), and [LSTM](#). In the experiment, 210 test runs were conducted using various [word2vec](#) hyperparameter combinations. The resulting word embeddings were integrated into the embedding layer of [DNN](#)

architectures, specifically [RNN](#), [CNN](#), and [LSTM](#). These models were subsequently trained on the Amazon Customer Reviews dataset for classification purposes. The outcomes were then gathered and analysed as follows.

Table 5 presents the range (minimum, maximum), as well as the average of F1 Scores for the test dataset across various [DNN](#) architectures, namely Simple [RNN](#), [CNN](#), and [LSTM](#). Among the networks tested, [CNN](#) recorded the peak F1 score at 92.5%. Meanwhile, [LSTM](#) displayed remarkable consistency with a high median F1 score of 91.9%, although [CNN](#) had a slightly higher average F1 score of 91.05%. This suggests that the [LSTM](#) architecture exhibits greater stability in response to hyperparameter variations in [word2vec](#)-generated embedding layers. On the contrary, with an F1 score at the lower end of 83.2%, the Simple [RNN](#) was the least effective. It's a widely accepted fact that Simple [RNNs](#) are generally outperformed by [LSTMs](#) in terms of accuracy, which aligns with the pre-experimental expectations.

Table 5: F1 score value (min, max, average, and median) for [RNN](#), [CNN](#), and [LSTM](#).

	MIN	MAX	AVERAGE
RNN	0.831	0.832	0.8315
CNN	0.896	0.925	0.9105
LSTM	0.891	0.919	0.905

To display the results visually, we chose the candlestick graph to display the overall performance of each [DNN](#) architecture. A candlestick chart is a visual tool originally designed for tracking the movement of financial instruments over time. Each candlestick represents data for a specific time frame and displays the opening, closing, highest, and lowest values. In scientific research, these charts have been repurposed to illustrate data fluctuations, trends, and anomalies in experimental findings. Furthermore, they can be tailored to represent statistical distributions, with the candlestick's body showcasing the central data range and the wicks extending to the outer data points. This offers scientists a detailed snapshot of data behaviour, aiding in more informed analyses and interpretations.

In order to provide a comprehensive summary of the F1 scores for each model, various statistical measures were employed. The 'Minimum' value in each model's column was derived using the formula MIN (range), which identifies the smallest F1 score in the dataset for that specific model. Conversely, the 'Maximum' value was computed with the MAX (range) function, highlighting the peak performance achieved by the model across iterations. The '1st Quartile' and '3rd Quartile' values represent

the 25th and 75th percentiles, respectively. These were calculated using the formula `QUARTILE.EXC (range, k)`, where `k` can be 1 or 3 for the first or third quartile. Essentially, for the 1st Quartile, 25% of the data points are below this value, and for the 3rd Quartile, 75% of the data points are below this value. The 'Median', also known as the 50th percentile, was computed using the `MEDIAN (range)` function, providing a central value that divides the data into two equal halves, with half the scores above it and half below. These statistics collectively provide a detailed perspective on the model's performance distribution, ensuring both the breadth and depth of understanding.

Figure 18 presents the F1 score summary statistics obtained from multiple iterations of three distinct [DNN](#) models: [RNN](#), [CNN](#), and [LSTM](#). These statistics are computed over various hyperparameter combinations of [word2vec](#). After generating the output from the [word2vec](#) embedding layer, it is passed to the embedding layer of each [DNN](#). Subsequently, training and testing are performed on the dataset for classification purposes. These statistics are ideally visualised using a candlestick chart, where each "candle" represents the range from the 1st quartile to the 3rd quartile, effectively capturing the interquartile range ([IQR](#)), with the minimum and maximum values, providing a comprehensive view of the data distribution. Mathematically, for a normally distributed dataset, about 68% of the data falls within the first standard deviation ($mean \pm 1\sigma$), 95% within the second ($mean \pm 2\sigma$), and 99.7% within the third ($mean \pm 3\sigma$).

Observationally, the [LSTM](#) model seems to generally outperform the [RNN](#) and [CNN](#) models, boasting the highest median F1 score of approximately 0.9193. It's significant to mention that the [CNN](#) model registered the highest F1 score of 0.9251, suggesting its effectiveness in specific scenarios. Overall, understanding these distributions is essential for determining the consistency and potential peaks of each model's performance. The reduced candle size observed in the [LSTM](#) candle suggests its stability against changes in [word2vec](#) hyperparameters. This characteristic might position [LSTM](#) as a preferred option when maximum performance isn't imperative. Its capacity to deliver commendable results without the need to pinpoint the optimal [word2vec](#) hyperparameter combinations adds to its appeal.

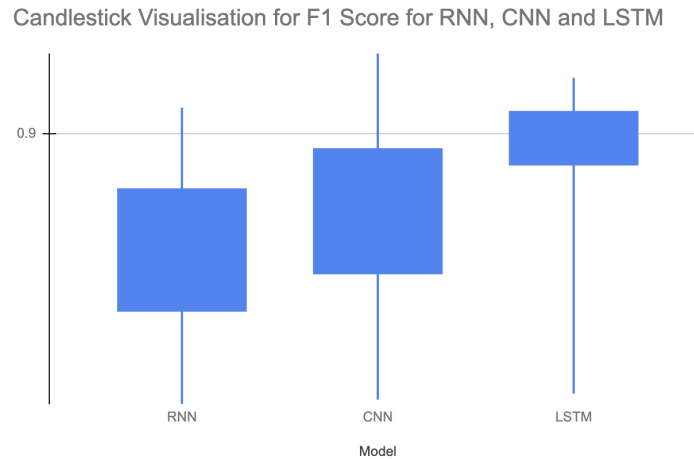


Figure 18: Graphical depiction of F1 score metrics in candlestick visualisation for [RNN](#), [CNN](#) and [LSTM](#)

To assess the statistical significance of the differences in F1 scores among the [DNN](#) models ([RNN](#), [CNN](#), and [LSTM](#)), a comprehensive analysis was carried out. The F1 scores, derived from various [word2vec](#) hyperparameter combinations, were analysed using several statistical tests. First, a one-way ANOVA (St et al., [1989](#)) test was conducted to determine if there were any significant differences in the mean F1 scores across the three models. The null hypothesis, which stated that there were no differences in the means, was tested. The ANOVA results showed a significant impact of the type of [DNN](#) model on the F1 scores, leading to the rejection of the null hypothesis and indicating that at least one model's performance differed significantly from the others. To identify which pairs of models differed significantly, a post-hoc Tukey HSD test was performed. The results indicated that:

- The [LSTM](#) model (mean F1 score = 0.919) significantly outperformed the Simple [RNN](#) model (mean F1 score = 0.8315), with a p – value < 0.001.
- The [CNN](#) model (mean F1 score = 0.9105) also showed a significant improvement over the Simple [RNN](#) model (mean F1 score = 0.8315), with a p – value < 0.001.
- There was no statistically significant difference between the [LSTM](#) and [CNN](#) models, though the [LSTM](#) had a higher median F1 score, suggesting greater stability and consistency.

This analysis highlights the superior performance and stability of the [LSTM](#) model compared to the Simple [RNN](#), and its comparable performance to the [CNN](#) model. These results support the experimental expectations and suggest that the

[LSTM](#) model is particularly robust for classification tasks using [word2vec](#) embeddings, maintaining high performance across a range of hyperparameters.

5.4.2. Visual performance evaluation of [DNN](#) architectures

Having analysed the statistical evaluation of the F1 score for various [DNN](#) architectures in relation to diverse [word2vec](#) hyperparameter combinations, we will now explore their comparative performance visually. Each test run illustrates the efficacy of a particular [word2vec](#) hyperparameter combination when applied to specific [DNN](#) architectures, namely [RNN](#), [CNN](#), and [LSTM](#). This analysis uses the stepped area chart which is a variation of the traditional area chart, where data points are connected by vertical and horizontal lines, creating a step-like progression instead of the typical diagonal lines. This type of chart is particularly useful for visualising data that changes at specific intervals or for emphasising the discrete nature of data points. One of its main strengths is its ability to clearly represent data that has sudden jumps or drops, making it easier for viewers to discern exact points of change and the duration over which a particular value remains constant.

The diagrams in this section presented below compare three models: [RNN](#) (represented in blue), [CNN](#) (in black), and [LSTM](#) (in yellow). The horizontal axis represents the varying values of the hyperparameter being examined, while the vertical axis shows the F1 score values from all trials. By observing the dominant colour in any section of the graph, one can infer the influence of a specific hyperparameter on a model's performance. For instance, if yellow predominates at the top, it suggests that the [LSTM](#) model excels for most values of that hyperparameter. Conversely, a blue dominance at the bottom indicates suboptimal performance of the [RNN](#) for that hyperparameter range. The x-axis displays hyperparameter values in descending order. For instance, the initial graph showcases various combinations of [word2vec](#) and its hyperparameters, using the experiment ID on the x-axis. In this context, the sequence is not crucial. However, in subsequent graphs, the order becomes significant as the x-axis represents a specific [word2vec](#) hyperparameter. When discussing the dimension size hyperparameter, for example, the x-axis arranges the dimension sizes in descending order. This arrangement provides a visual representation that is influenced by the x-axis sequence. A concentration of yellow on the right suggests that the [LSTM](#) model may not perform as well with smaller hyperparameter values. We will begin by examining the overall performance of various

[word2vec](#) hyperparameter combinations across different [DNN](#) architectures. The objective is to determine which [DNN](#) architecture is most robust to hyperparameter variations, which one achieves optimal performance, and which one is susceptible to these variations. From Figure 19, one can observe that the [CNN](#) performance denoted by black colour experiences considerable variance, suggesting a marked sensitivity of [CNN](#) to alterations in the [word2vec](#) hyperparameters. While [CNN](#) recorded the highest F1 score for specific hyperparameter configurations, it also exhibited pronounced inconsistencies, especially when compared with the [LSTM](#) network. Such variability underscores the potential repercussions on performance should there be adjustments in the [word2vec](#) parameters. The [LSTM](#) network is depicted by the yellow area. Notably, the yellow area is predominantly higher than blue and the black areas, reflecting [LSTM](#)'s inherent stability against shifts in the [word2vec](#) hyperparameters. Even though [LSTM](#) did not obtain the absolute highest F1 score in the tests, its median and mean F1 scores surpass those of both [RNN](#) and [CNN](#).

Conversely, the blue area, which corresponds to Simple [RNN](#), is primarily concentrated at the bottom spectrum of the graph. This positional dominance underlines the relative underperformance of Simple [RNN](#) in the face of diverse [word2vec](#) hyperparameter configurations, particularly when benchmarked against its counterparts, [CNN](#) and [LSTM](#).

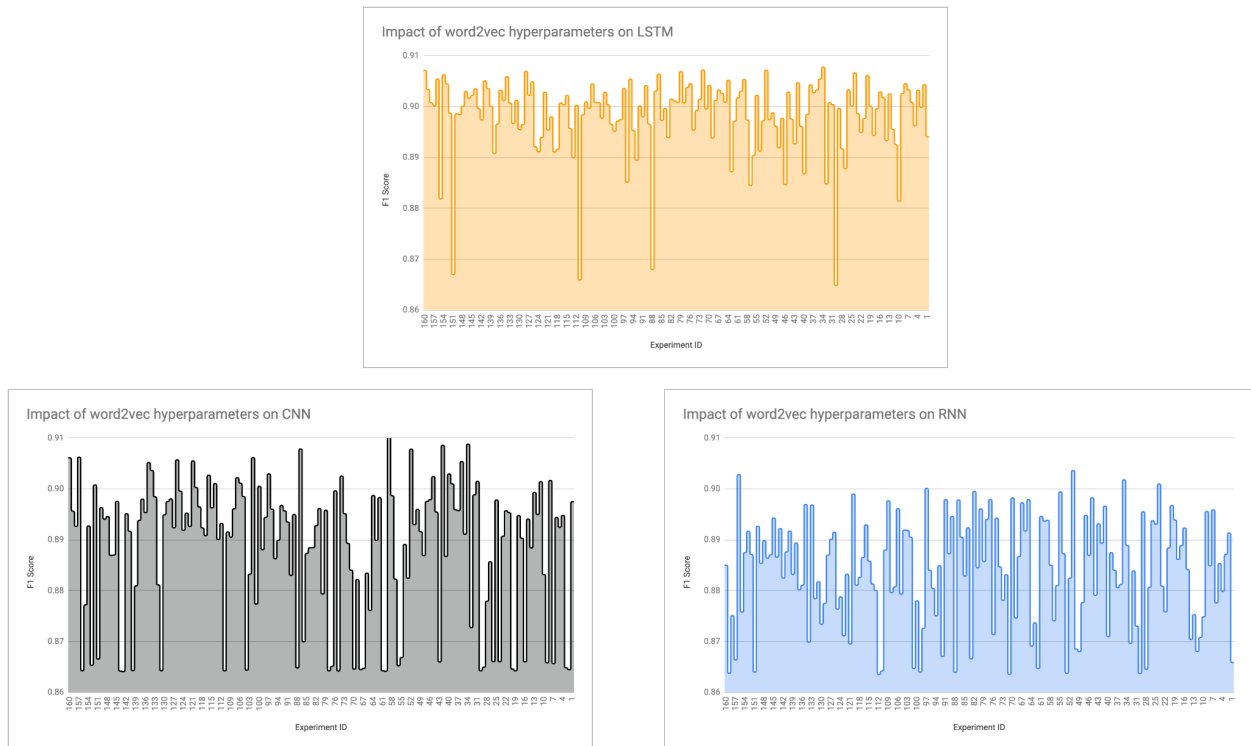


Figure 19: Variations in F1 scores for [RNN](#), [CNN](#), and [LSTM](#) with diverse [word2vec](#) hyperparameter settings.

5.4.3. Impact of different [word2vec](#) hyperparameters on [DNNs](#)

Having assessed the performance of different [DNN](#) architectures in relation to various [word2vec](#) hyperparameter combinations, we will now delve into the specific effects of each [word2vec](#) hyperparameter on the [DNN](#) architecture's performance. It is essential to understand that each hyperparameter is not tested in isolation. In every test run, different combinations are randomly selected. Thus, results from a single run shouldn't be used to draw final conclusions. Instead, we focus on a broader statistical analysis to identify if certain settings consistently influence the [DNN](#) performance, regardless of other settings. Given the random selection of [word2vec](#) settings, there may be gaps in the range of tested values and potential repetitions in the values of the parameters presented in the graphs. It is also important to denote that the area under the curve for all [DNN](#) architectures is similar for all graphs below. However, the distribution of performance will vary according to the value of the [word2vec](#) hyperparameter value.

5.4.4. [Word2vec](#) epoch hyperparameter

The number of epochs in [word2vec](#) denotes the number of iterations over the complete training dataset undertaken by the algorithm. Increasing the number of epochs can lead to better model performance, but at the risk of overfitting if done excessively.

Figure 20 showcases the influence of the [word2vec](#) hyperparameter "epoch" value on the performance of [RNN](#), [CNN](#), and [LSTM](#). The graph reveals that an extremely low epoch count adversely affects all [DNN](#) architectures. Additionally, a high epoch count appears to predominantly have a detrimental effect on these architectures. Specifically examining the [CNN](#), represented in black, it's evident that a smaller epoch count hampers its performance. Yet, there's a marked enhancement in [CNN](#)'s performance when epochs exceed 24. A similar trend is discernible with [LSTM](#), albeit to a milder degree.

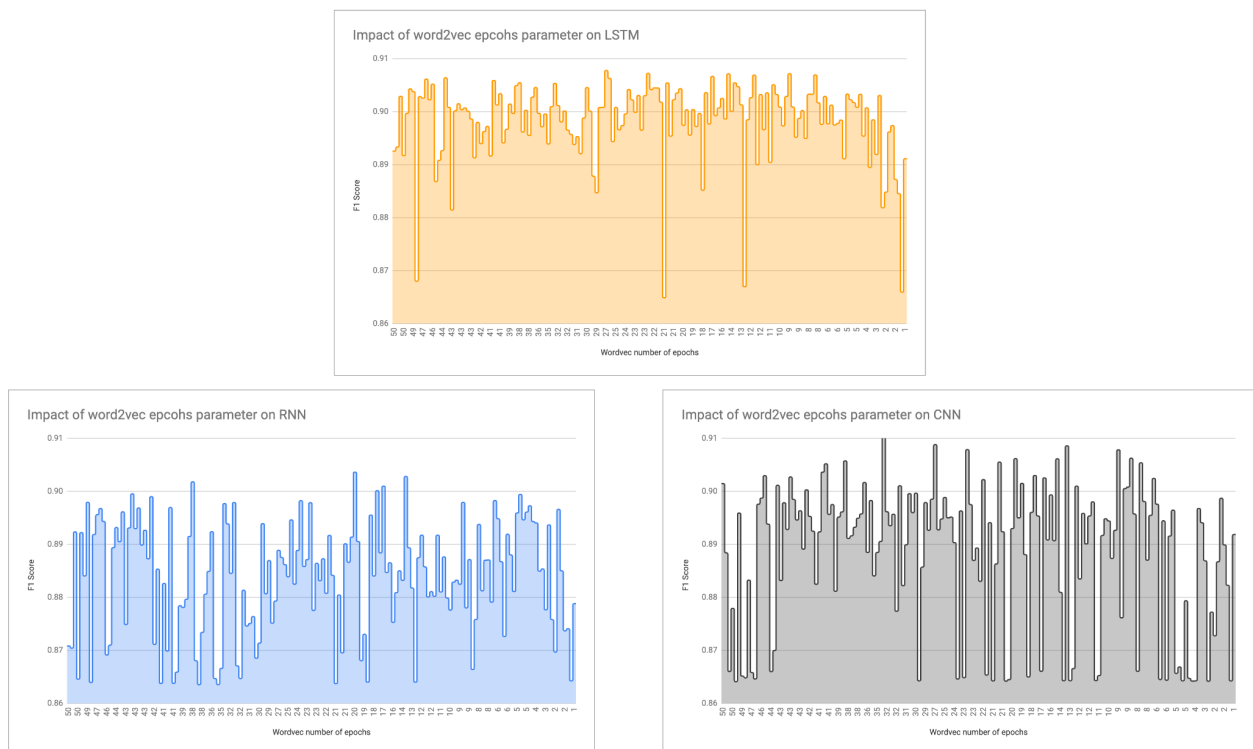


Figure 20: Impact of [word2vec](#) epoch parameter on [DNN](#) architectures.

5.4.5. [Word2vec](#) window size hyperparameter

The [word2vec](#) window size parameter determines the number of words around a target word that should be considered as its context. A larger window size captures a broader context, while a smaller size focuses on the immediate neighbouring words. Based on prior experiments, it was noted that window size directly correlates with interchangeability and relatedness tasks. For instance, In the interchangeability task, alternate names for underutilised crops are determined, including both vernacular and scientific terms. In contrast, the relatedness task is geared towards identifying characteristics associated with these underutilised crops. Notably, in the [word2vec](#) model, the window size as a hyperparameter can significantly influence the results of

these downstream tasks. In the current study, we investigate the effect of the window size hyperparameter on the classification task using various [DNN](#) architectures. A glance at Figure 21 clearly shows that [LSTM](#)'s performance is notably diminished at lower window size compared to previous observations. Yet, the performance of [LSTM](#) noticeably improves as the window size surpasses 12. This suggests a pronounced direct proportionality between window size and [LSTM](#)'s efficacy.



Figure 21: Impact of [word2vec](#) window size parameter on [DNN](#) architectures.

5.4.6. [Word2vec](#) negative sampling hyperparameter

Negative sampling in [word2vec](#) is a technique used to simplify the training process by updating only a subset of the model's weights. Instead of updating all weights for each training example, it updates the target word's weights and a small number of randomly chosen non-target words, termed as "negative samples". Within [word2vec](#)'s Skip-Gram approach, the goal is to foresee the neighbouring words of a given term. Negative sampling enhances this process by concentrating the weight adjustments on the actual context words and a select set of unrelated words, rather than the whole word list. This technique not only accelerates the training phase but also intensifies the differentiation between the central term and its contextual counterparts compared to arbitrary, non-associated terms. Upon examining the lower negative sampling

values in Figure 22, [LSTM](#) shows higher scores at lower values of negative sampling. [RNN](#) tends to perform relatively better at these reduced values.

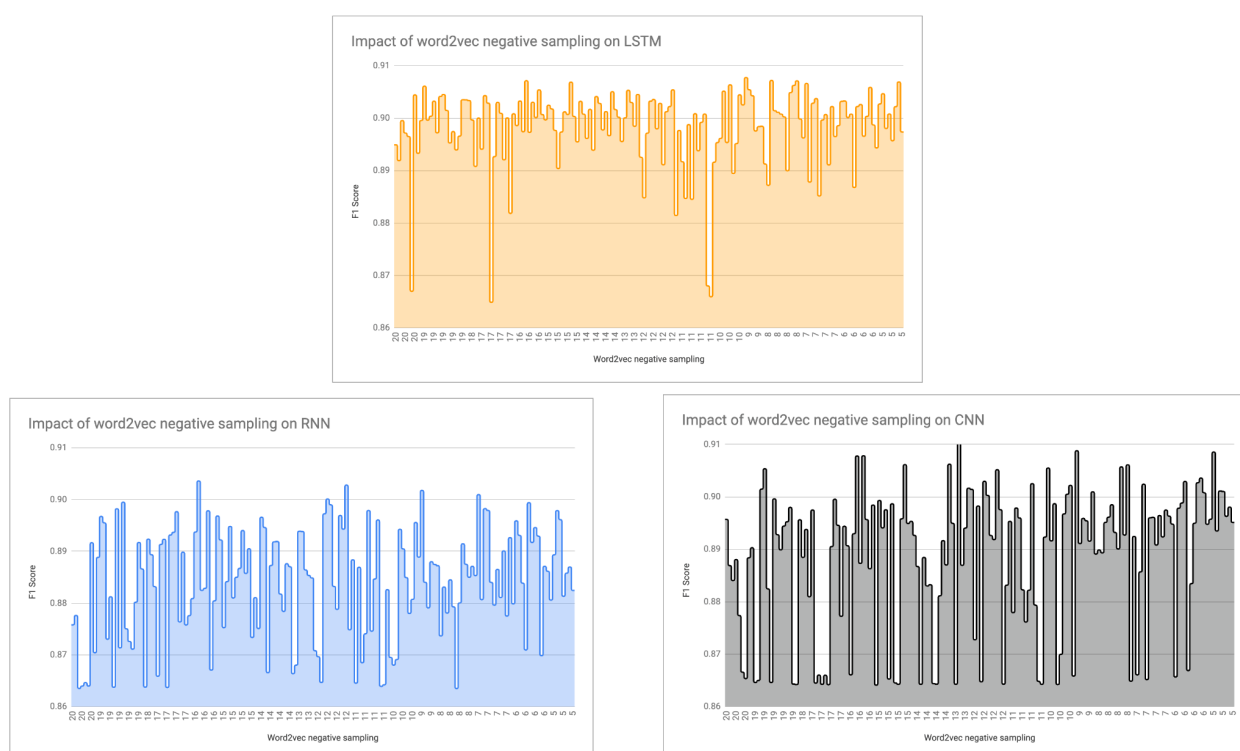


Figure 22: Impact of word2vec negative sampling parameter on [DNN](#) architectures.

5.4.7. [Word2vec](#) minimum count hyperparameter

The minimum count parameter in [word2vec](#) specifies the minimum number of occurrences a word must have in the dataset to be included in the model's vocabulary. Words that appear fewer times than the specified threshold are discarded, helping to eliminate rare words and reduce the dimensionality of the embeddings. While Figure 23 may initially suggest a lack of distinct performance trends among the [DNN](#) architectures, a closer examination reveals that the [LSTM](#) might show a noticeable improvement in performance at higher minimum count values.



Figure 23: Impact of [word2vec](#) min count parameter on [DNN](#) architectures.

5.4.8. [Word2vec](#) hierarchical softmax hyperparameter

Hierarchical softmax in [word2vec](#) is an optimization technique used to speed up the training process. Instead of computing the softmax over the entire vocabulary, it uses a binary tree representation of the vocabulary and calculates the probability of a word by traversing the tree, reducing the computational complexity. Figure 24 suggests that toggling hierarchical softmax (represented by 1 for 'on' and 0 for 'off') doesn't significantly influence the performance of any deep [DNN](#) architectures. This could be because hierarchical softmax primarily serves as an optimization tool to expedite training rather than directly impacting model performance.

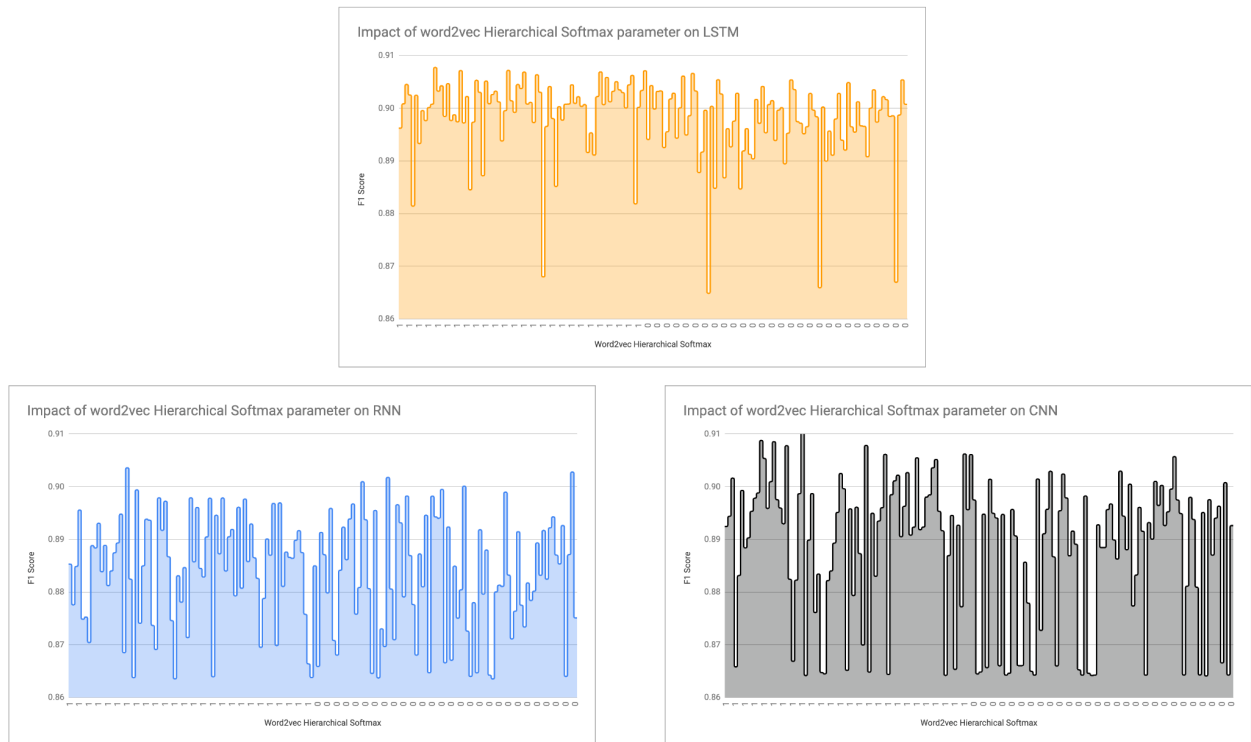


Figure 24: Impact of [word2vec](#) hierarchical softmax parameter on [DNN](#) architectures.

5.4.9. [Word2vec](#) Skip-Gram vs Bag of Words architecture

Skip-gram is a [word2vec](#) architecture where a target word is used to predict its surrounding context words in a given window. On the other hand, the Bag of Words (BoW[aaa](#)BOW) model represents a text by the frequency of its words, disregarding the order or sequence, making it a sparse representation of the original text. On the x-axis, 1 represents the Skip-Gram method and 0 denotes the Bag of Words approach. This visually presented as 1 = "left half of the chart", 0 = "right half of the chart. Figure 25 reveals that [RNNs](#) fare better with Skip-Gram, while relatively struggling with Bag of Words. Conversely, [CNNs](#) show a preference for Bag of Words and relatively underperform with Skip-Gram. [LSTMs](#) appear indifferent to the choice of architecture, maintaining consistent performance.



Figure 25: Impact of [word2vec](#) Skip-Gram or Bag-of-Words parameter on [DNN](#) architectures.

5.4.10. [Word2vec](#) dimension size architecture

In [word2vec](#), the size of the dimension indicates the count of neurons present in the embedding layer, which establishes the vector length for every word. A larger dimension size captures more nuanced semantic relationships between words but may require more data to train effectively and avoid overfitting. Figure 26 suggests that smaller embedding dimensions adversely affect all models' efficacy, even with [LSTMs](#) that clearly demonstrated previously to be robust against hyperparameter changes. From a dimensionality of 44 onwards, there's a noticeable improvement in performance with all [DNN](#) architectures. Yet, for [CNNs](#) and [RNNs](#), larger dimensions can be detrimental. [LSTMs](#), however, manage larger dimensions more effectively.

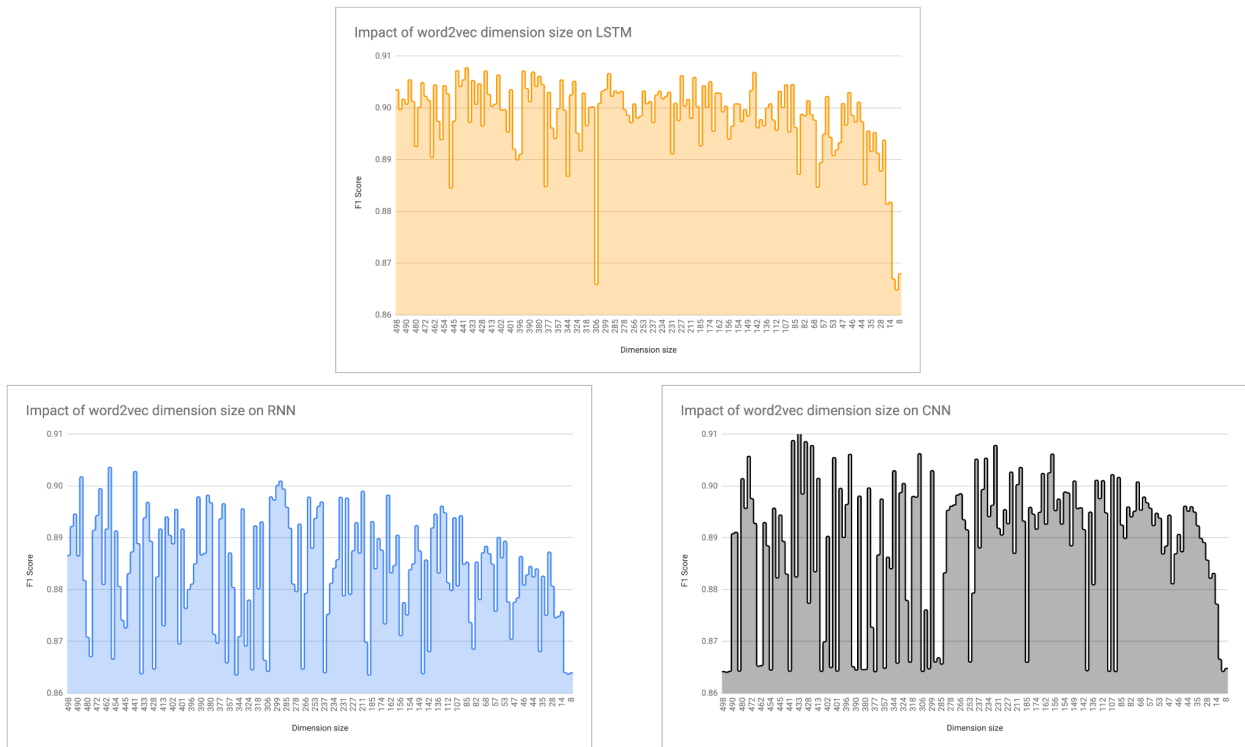


Figure 26: Impact of [word2vec](#) dimension size parameter on [DNN](#) architectures.

5.5. Conclusion

In this experiment, we delved deep into the realm of word embeddings, leveraging the robust capabilities of [word2vec](#) to craft mathematical representations of textual content. Our primary focus was on understanding the importance of hyperparameters and their impact on the quality of word embeddings. By employing [word2vec](#), we navigated through a myriad of hyperparameter combinations, emphasising the importance of each in the context of the model's performance. Our exploration was not limited to just the creation of embeddings; we further integrated these embeddings into deep neural network classifiers, such as simple [RNN](#), [CNN](#), and [LSTM](#), to discern their efficacy in text classification tasks.

This research underscores the significance of hyperparameter tuning in the realm of word embeddings. While the vastness of potential combinations presents a computational challenge, strategies like random search offer a viable solution, as corroborated by various academic studies. Furthermore, our investigation into document classification illuminated the pivotal role of word embedding layers in influencing the accuracy of [DNN](#) classifiers. By comparing various [DNN](#) architectures, we aimed to uncover the optimal blend of hyperparameters and network structures that achieve the highest performance.

In essence, our study serves as a testament to the interaction between word embeddings and [DNNs](#) in the domain of text classification. The findings accentuate the need for carefully selected hyperparameters and the profound impact of embedding layers on the overall performance of classifiers. As we move forward, it is imperative to continue refining our understanding of these parameters and their interplay, ensuring that we harness the full potential of [ML](#) in text analysis and beyond.

In this research, we delineated the diverse methodologies employed throughout our experiments. Initially, we harnessed Google search engine data, complemented by text mining techniques, to discern cropping patterns. Progressing further, we incorporated [NIR](#) and word embeddings for extracting explicit information. This enhanced approach facilitated high quality information extraction, such as the associations between specific crops and countries, vernacular and scientific crop names, and the soil requirement for underutilised crops.

A significant portion of our research was dedicated to the determination of word embedding hyperparameter combinations. This was pivotal in securing high-calibre embeddings, which in turn bolstered the performance of subsequent tasks, notably classification.

In the literature review, the crucial role of hyperparameter tuning for specific downstream tasks was highlighted, particularly in the context of text classification. The findings from this experiment with the Amazon Customer Review dataset can be generalized to other datasets and downstream tasks. This research extensively utilised [word2vec](#) for crafting word embeddings and explored a wide array of hyperparameter combinations to understand their impact on model performance. This process included integrating these embeddings into various [DNN](#) classifiers like simple [RNN](#), [CNN](#), and [LSTM](#), thereby revealing the profound influence of hyperparameter tuning on text classification accuracy. The significance of hyperparameter tuning, as demonstrated by our study, is not confined to the Amazon Customer Review dataset alone. The strategies and insights gained from this dataset, such as the efficacy of random search in hyperparameter optimization, are applicable to other datasets as well. For instance, the importance of selecting optimal hyperparameters for embedding layers extends to any text classification task, whether it involves identifying cropping patterns from search engine data or extracting relationships between crops and countries from scientific literature. Moreover, our findings on the interaction between word embeddings and [DNN](#) architectures suggest

that the methodologies we developed can be effectively transferred to other domains. This includes tasks beyond text classification, such as extracting high-quality information about underutilised crops or understanding the chronological evolution of scientific knowledge. By continually refining our understanding of hyperparameters and their interplay, we ensure that the robust techniques validated on the Amazon dataset can enhance the performance of various machine learning applications in different contexts, making our conclusions broadly applicable.

As this research culminated, we broadened our data spectrum, transitioning from a sole focus on crop science to encompassing broader scientific domains, including physics. The pinnacle of our experiment aimed to explore the chronological evolution of knowledge. Our objective was to carefully craft a methodology that would illuminate significant scientific events and elucidate prevailing knowledge trajectories. Such insights can equip decision-makers with a data-driven approach, augmenting their intuitive understanding of scientific research directions.

In this study, we introduced several approaches to hyperparameter optimization and text classification using word embeddings. We utilised Gensim, a highly efficient open-source Python library, to generate word embeddings through algorithms like [word2vec](#), Fasttext, and Latent Dirichlet Allocation. To address the challenge of optimizing hyperparameters amidst numerous potential combinations, we employed random search. This method allowed us to efficiently explore 1000 hyperparameter combinations for the [word2vec](#) model, aiming to find the most effective configurations without incurring excessive computational costs. We meticulously optimized key [word2vec](#) hyperparameters, including the number of training epochs, the skip-gram (sg) indicator, window size, vector dimensionality, hierarchical softmax (hs) usage, negative sampling, and the minimum word count (min_count). The optimized word embeddings were then integrated into various [DNN](#) classifiers, specifically Simple [RNN](#), [CNN](#), and [LSTM](#) architectures, to perform text classification tasks. This integration highlighted the significant impact of word embedding layers on the overall performance of the classifiers. The evaluation of our models was rigorous, employing a range of metrics such as accuracy, precision, recall, and F1 score to provide a comprehensive assessment of their performance. One of the key technical challenges we faced was managing the computational demands of testing numerous hyperparameter combinations, which required careful balancing of model complexity and efficiency. Another challenge was selecting the appropriate dimensionality for

word vectors to capture detailed semantic nuances without overburdening the computational resources. We also had to navigate the complexities of training different neural network architectures, each presenting unique challenges such as the vanishing gradient problem in [RNNs](#) and the high computational requirements of [CNNs](#). Furthermore, integrating the word embedding layers into [DNN](#) models while maintaining their effectiveness was crucial. Balancing the number of training epochs to optimize model performance without excessive computational expense was another critical consideration. Through addressing these challenges, our study underscored the importance of hyperparameter tuning in enhancing the quality of word embeddings and their application in text classification. The findings emphasize the need for well-chosen hyperparameters and demonstrate the profound influence of embedding layers on the accuracy of [DNN](#) classifiers, offering valuable insights for future research in machine learning and text analysis.

The conducted experiments provide valuable insights into the performance and stability of various [DNN](#) architectures ([RNN](#), [CNN](#), and [LSTM](#)) in response to different [word2vec](#) hyperparameter settings, aligning well with established research findings. The tests, conducted across 210 runs using different [word2vec](#) hyperparameter combinations, reveal distinct performance patterns for each [DNN](#) architecture. Regarding vector size (dimensionality), the literature (Mikolov et al. [2013a](#)), (Goldberg et al., [2014](#)), (Le et al., [2014](#)) and (Kim, [2014](#)) typically recommends a range from 100 to 300 dimensions, as larger sizes capture more semantic information. The results confirm that smaller embedding dimensions negatively affect performance across all models, with significant performance improvements noted from dimensions of 44 onwards, especially for [LSTM](#), which manage larger dimensions effectively. The window size parameter, balancing local and broader context capture, is commonly set between 5 to 10 in literature. The experiment shows that [LSTM](#) performance improves with larger window sizes, notably beyond 12, indicating that [LSTMs](#) benefit from a broader context in capturing semantic relationships. Negative sampling rate in the literature ranges from 5 to 20, improving quality with more samples but increasing computational cost. The findings indicate that [LSTMs](#) perform better at lower negative sampling values, while [RNNs](#) also benefit from reduced values, highlighting the importance of optimizing this parameter for different architectures. Concerning the number of epochs, the literature suggests that more epochs generally lead to better performance but risk overfitting. The results show that very low epoch counts harm all

architectures, but excessively high counts also have adverse effects, particularly on [CNNs](#). Optimal performance is achieved with moderate epoch counts. The minimum count parameter helps reduce dimensionality by ignoring rare words. The experiment reveals that [LSTMs](#) might benefit from higher minimum count values, suggesting that eliminating rare words aids in enhancing model performance. Hierarchical softmax is primarily an optimization tool to speed up training, with no significant impact on model performance according to the literature. The results confirm this, as toggling hierarchical softmax showed no notable influence on the performance of any [DNN](#) architecture. When comparing Skip-gram and Bag of Words, the literature typically prefers Skip-gram for capturing word context effectively. The results indicate that [RNNs](#) perform better with Skip-gram, while [CNNs](#) excel with BoW. [LSTMs](#) maintain consistent performance regardless of the architecture, demonstrating their robustness. Overall, the experimental results corroborate the established understanding of [word2vec](#) hyperparameters' impact on [DNN](#) performance for classification tasks. [LSTMs](#)' robustness and stability make them a reliable choice, while [CNNs](#), although capable of high peak performance, require careful tuning of hyperparameters to avoid variability. Simple [RNNs](#)' lower performance aligns with the known limitations of this architecture compared to more advanced models like [LSTMs](#) and [CNNs](#).

The primary objective of this experiment was to optimise the hyperparameters of the [word2vec](#) model to produce high quality embeddings. These embeddings were then integrated into the embedding layers of various [DNN](#) architectures, specifically [RNN](#), [CNN](#), and [LSTM](#), for the classification of Amazon reviews. Among the [DNN](#) architectures tested, [CNN](#) achieved the highest F1 score. However, [LSTM](#) showcased remarkable consistency, with the highest average and median F1 scores, suggesting its resilience to hyperparameter variations in [word2vec](#)-generated embedding layers. In contrast, Simple [RNN](#) lagged. Visualisation tools, such as the candlestick graph and stepped area chart, were employed to visually represent the performance of each [DNN](#) architecture. The research investigated thoroughly the effects of various [word2vec](#) hyperparameters on [DNN](#) performance. Findings suggest that [CNN](#)'s performance is sensitive to changes in [word2vec](#) hyperparameters, showing both highs and lows. [LSTM](#) displayed stability against shifts in [word2vec](#) hyperparameters, consistently performing well. Simple [RNN](#) generally underperformed relative to the other models, especially when faced with diverse [word2vec](#) hyperparameter configurations. Specific

hyperparameters, such as the number of epochs, showed distinct effects on [DNN](#) architectures. For instance, both [CNN](#) and [LSTM](#) initially showed weak performance at a lower number of epochs, improved as the epoch count increased, but then deteriorated once again. Hierarchical softmax didn't significantly impact [DNN](#) performance, indicating its role as a training optimization tool rather than a direct performance influencer. [RNNs](#) performed better with the Skip-Gram method, while [CNNs](#) favoured the Bag of Words approach. [LSTMs](#) remained consistent regardless of the choice. Smaller embedding dimensions were found to be detrimental to all models. However, [LSTMs](#) handled larger dimensions more effectively than [CNNs](#) and [RNNs](#).

In summary, this research provides valuable insights into the intricate relationship between [word2vec](#) hyperparameters and [DNN](#) performance. While [CNNs](#) showed potential peaks in performance, [LSTMs](#) emerged as the most consistent and resilient architecture, especially against variations in [word2vec](#) hyperparameters. This positions [LSTM](#) as a reliable choice for text classification tasks, especially when the goal is consistent performance across varying conditions. After evaluating the data, the chosen hyperparameters for [word2vec](#) include: the Skip-Gram model, activation via hierarchical softmax, negative sampling at 5, an embedding dimensionality of 500, a context window of 20, and a total of 40 training cycles. The Skip-Gram model was selected due to its consistent high performance across [DNNs](#). Using hierarchical softmax combined with a negative sampling rate of 5 enhanced the model performance. The decision to use a 500-dimensional space was influenced by observations showing improved [DNN](#) performance with increased dimensions. A context window of 20 struck the right balance in capturing contextual relationships, and training for 40 epochs was found optimal, striking a balance between learning depth and avoiding overfitting from the observations.

This research significantly contributes to the field by providing a comprehensive analysis of the impact of [word2vec](#) hyperparameters on various [DNN](#) architectures for text classification. Consistent with existing literature, such as the studies by Minaee et al. (2021) and Mikolov et al., (2013a) it confirms that [LSTM](#) architectures are notably robust across varying conditions, showcasing remarkable stability in performance. In contrast, [CNNs](#) demonstrate sensitivity to hyperparameter variations, with potential for peak performance under optimized conditions. Our detailed hyperparameter optimization reveals that using the Skip-Gram model, hierarchical softmax, negative

sampling of 5, embedding dimensions of 500, context window of 20, and training for 40 epochs significantly enhances [DNN](#) performance. Specifically, the chosen hyperparameters in this study align with and, in some cases, surpass those used in other research, providing a detailed comparison: typical literature suggests embedding dimensions around 300, context windows between 5-10, and epochs ranging from 10-20 Shiri et al., ([2023](#)). This tailored approach underscores the importance of context in word embeddings and offers an effective methodological advancement through the use of visualization tools, contributing to more intuitive and accessible model performance analysis.

Chapter 6

Temporal Knowledge Evolution

6.1. Introduction

The realm of scientific research is dynamic, continuously shaped by emerging technologies and new discoveries. To navigate this evolving landscape, one must grasp the nuances of knowledge evolution within academic literature. This study aims to chart the trajectory of scientific knowledge by harnessing the potential of word embeddings and analyse over time. Our primary data set was drawn from the arXiv repository, encompassing scholarly publications from 2000 to 2020. To process this extensive textual data, we implemented a cloud-based big data pre-processing pipeline, facilitating the efficient refinement of the acquired information. This pre-processing not only began with data decompression but culminated in the creation of monthly files, each representing the academic literature released within its respective month. An integral step in this pipeline is the classification of documents, enabling the subsequent clustering of relevant articles. This classification relies on the word embeddings generated from a meticulous hyperparameter tuning experiment which focuses on developing a refined text classification model using word embeddings generated from a carefully tuned hyperparameter experiment. These high-quality embeddings are crucial for improving downstream tasks such as [NIR](#), classification, and temporal knowledge evolution. It is important to note that the embeddings obtained from this experiment were not used in this experiment and only the training and embedding parameters were utilised. Given the extensive scope of the arXiv corpus, an effective classification step is necessary to ensure that only pertinent information contributes to the temporal knowledge evolution and vector space creation. The goal is to curate relevant content from a vast collection of 2.3 million arXiv documents, thereby enhancing the Temporal Knowledge Evolution process with superior quality embeddings. The effectiveness of this filtration process will significantly impact the accuracy and relevance of the resulting timeline in the evolution process.

The organized monthly clusters subsequently serve as the foundation for developing refined vector spaces. Aligning these spaces is essential, especially when crafting

temporal vector spaces. With the temporal vector spaces established, our focus is centered on the temporal element of knowledge. A rigorously structured methodology was employed to scrutinize the specified components. The results underscore the prowess of examining temporal shifts in vector spaces to decode patterns within academic literature. This approach provides invaluable insights for scholars striving to fathom the dynamics of knowledge in their fields. Furthermore, it can act as a beacon for decision-makers channelling resources into scientific progression.

6.2. Method

6.2.1. Data pipeline and pre-processing steps

As the importance of the arXiv dataset grows within the scientific community, extracting insights from this vast and valuable resource presents significant challenges. The dataset, which contains raw text and figures in PDF format, is difficult to process using standard text processing techniques. Additionally, the large size of the dataset requires substantial computational resources, making it inaccessible to many researchers.

One of the primary challenges of working with the arXiv dataset is converting the raw data contained in the PDF files into a usable format. The PDF format includes special characters, which can complicate the accurate extraction and conversion of text. Furthermore, the PDF format preserves the document's layout, including images, tables, equations, and formatting, which can interfere with text extraction algorithms. Therefore, researchers require specialised tools to accurately extract text from the PDF files. Another significant challenge when working with the arXiv dataset is its size. Housing over 2.3 million articles, the dataset contains a massive amount of information, necessitating significant computational resources for processing. This requirement poses a challenge for many researchers who may not have access to the necessary computing resources, thereby complicating the extraction of insights from the dataset.

Numerous researchers have made significant efforts to simplify the pre-processing of the arXiv dataset. For instance, Saier et al. ([2023](#)) made a substantial contribution by developing the unarXive 2022 dataset, which includes 1.9 million publications. They have created a comprehensive citation network and enriched the

document structure. Brinn et al. ([2022](#)) have enhanced the accessibility of research on arXiv by proposing the integration of HTML formats alongside the existing PDF and TeX formats. Furthermore, Rastogi et al. ([2022](#)) have contributed to the ongoing debate about the use of preprints by conducting a survey-based study on the advantages and disadvantages of posting papers on arXiv.

In response to the challenges associated with the arXiv dataset, we have engineered a data processing pipeline that streamlines and standardises access to the wealth of knowledge encapsulated within the dataset. This pipeline, illustrated in Figure 27, incorporates a cloud-based script designed to download the complete set of PDFs accessible via the arXiv's bulk download service, in addition to a PDF-to-text converter tasked with transforming the PDFs into plaintext. PDF-to-text conversion programs use Optical Character Recognition and text extraction algorithms to convert PDFs into text files. We used “pdftotext” from the “Poppler” library. Initially, the tool identifies the document structure and extracts embedded text directly if the PDF contains digital text. For scanned or image-based PDFs, it is applied to recognize and convert images of text into machine-encoded text. The process involves analysing the layout, detecting text blocks, lines, and individual characters, and then translating visual data into text using pattern recognition techniques. The resulting text is then formatted and outputted into a text file, maintaining as much of the original document's structure and content as possible. This conversion methodology enabled us to extract 1.37 million raw text documents from an initial pool of 1.5 million PDFs. The reduction in document count can be attributed to multiple failures experienced during the conversion process. It is worth noting that the data extraction was executed at an earlier stage in the study, which is reflected in the number of articles. The resultant raw-text dataset is substantial, encompassing approximately 11 billion words and constituting around 64GB in size. Despite the imperfections in the extracted text, we believe that it serves as a robust foundation for numerous research tasks. We are continuously striving to improve the data processing pipeline in our future contributions.

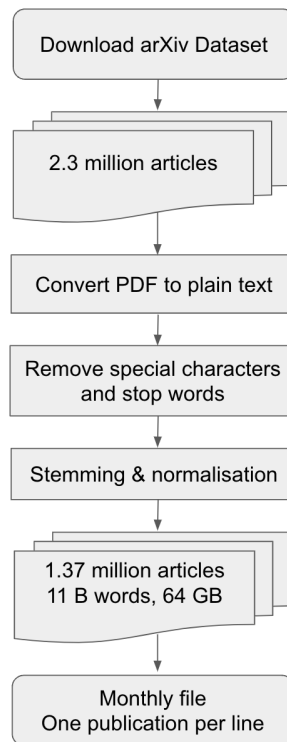


Figure 27 Data pipeline and pre-processing steps

Textual data pre-processing, a crucial step in harnessing the full potential of large, unstructured datasets, constitutes a significant component of the data processing pipeline. Text data often includes unwanted artefacts, such as special characters, stop words, and ASCII code, which can impede the precision of text analysis and subsequent steps. By employing effective data pre-processing techniques, these issues can be mitigated, thereby preparing the data for subsequent analysis. For instance, Camacho et al. (2017) investigated the impact of text pre-processing decisions on the performance of a standard neural text classifier. The authors conducted an extensive evaluation on standard benchmarks from text categorization and sentiment analysis, with their experiments highlighting the importance of this often-overlooked step in the pipeline. The authors were able to achieve a higher performance of 2.4% for the best-performing model, especially when the training data was not large enough to generalise.

In this research, an array of techniques was deployed for the pre-processing of textual data, including the application of regular expressions, stop word removal, stemming, and normalisation. Stop word removal is a process that involves discarding common words such as "and," "the," and "is," which are often considered to have less distinctive semantic value in the context of text analysis. In this research we utilised the NLTK stop word list and NLTK stemmer (Sarica et al., 2021). However, it's crucial

to note that while these words may contribute less specifically semantically in certain [ML](#) contexts, they still hold semantic value. Linguists, for instance, argue that these words play vital roles in sentence structure and meaning.

Considering the above information, along with the limitations of our computational resources and the size of our corpora, we have decided to remove stop words in our data analysis. This decision is informed by insights from both studies, as the impact of stop word removal appears to be context dependent. Our specific analysis scenario may differ from those in the prior research. We believe that removing stop words from our dataset will be suitable, balancing computational cost while preserving semantic meaning. Similarly, stemming is a technique used to condense words to their root form, which can help reduce data dimensionality and potentially enhance the efficiency of subsequent analyses. Finally, normalisation techniques are utilised to ensure the text data adheres to a consistent format, such as converting all text to lower case.

We organised the resultant text data into monthly files, with each file encapsulating one scientific paper per line. All lines representing a single month are concatenated into one file. The purpose of this segmentation is to prepare the data for the construction of a vector space continuum of embeddings, which is used for time series analysis. This analysis aims to detect events and uncover insights by identifying trends and shifts in the dataset over time. By segmenting the data in this manner, researchers can systematically analyse the evolution of research topics, trends, and other temporal patterns within the arXiv dataset. This data subsequently facilitated the construction of a vector space continuum of embeddings for time series analysis to detect events and uncover insights. The deployment of vector space continuum embeddings enabled the identification of trends and shifts in the arXiv dataset over time, thereby providing a more profound comprehension of the evolution of research in the field. In summary, the pre-processing of textual data was a crucial step in unlocking the potential of the arXiv dataset, paving the way for novel discoveries and insights that are set to potentially influence the scientific community.

6.2.2. Systematic categorization of scholarly publications in scientific literature

The initial phase of our research resulted in a corpus of approximately 1.37 million articles, comprising an estimated 11 billion words. Our overarching objective is to

leverage this extensive body of text, partitioned into discrete monthly segments, to construct a series of vector spaces, a concept we have termed a 'vector space continuum'. This innovative framework will enable the systematic examination of the evolution of knowledge over time. However, the sheer magnitude of the corpus, encompassing a broad spectrum of topics from diverse disciplines such as mathematics, science, and physics, presents a significant challenge. The integration of such varied knowledge domains within the corpus carries the potential risk of introducing noise, which could inadvertently lead to unforeseen biases

To counteract this, we implemented an automated process to segregate the documents into three primary categories: mathematics, science, and physics. Given the absence of a pre-labelled dataset for classifier training, we resorted to word embedding techniques to identify the top-ranking documents most closely associated with each category. The procedure began with the determination of the vector representation for each category, followed by the generation of a unique vector for each article. We then employed cosine similarity to identify the 5,000 articles most closely aligned with each category. These 15,000 articles, spanning the categories of science, mathematics, and physics, constituted the training dataset for our classifier. Informed by our previous research findings, we chose to deploy a [LSTM](#) network as the classifier. The resulting [LSTM](#) network was subsequently used to categorise all the documents into the three designated categories. These categorised documents will serve as the foundation for the construction of the vector space continuum.

To validate our methodology, we researched several recent publications that utilise ([LSTM](#)) for document classification. For instance, Alfarizi et al. ([2022](#)) developed an innovative approach using [LSTM](#) for emotion-based document classification, achieving an impressive accuracy of 97.50%. In a similar vein, Wang et al. ([2022](#)) proposed a distinctive method that amalgamates sequence-based [LSTM](#) with traditional [API](#) call features for malware classification (Hua et al., [2020](#)). Their research highlights the effectiveness of [LSTM](#)-based hybrid classification in enhancing the dynamic analysis and classification of malware. These recent studies underscore the continued relevance and effectiveness of [LSTM](#) in research. The objective of this validation process is to standardize the methodology selection operations by utilizing recent studies on [LSTM](#) to confirm our choice, ensuring alignment with the latest findings.

6.2.3. Vector space continuum construction

Upon completion of the preliminary pre-processing stages on the arXiv dataset, the resultant files serve as the input for the training procedure. The objective of this process is to generate a unique set of vector space continua for each month of the scientific literature corpus. This procedure necessitates the training of a word embedding model, such as [word2vec](#). Each file corresponds to the publications released within a specific month, with each line within the file representing an individual publication. Our research primarily concentrated on the timeframe spanning from 2000 to 2020. Publications from the years 1991 to 1999 were deliberately omitted from the analysis due to their comparatively smaller volume in contrast to the publications post the year 2000. Upon successful training of the model, each resultant vector space signifies a distinct temporal period, with analogous words and concepts being closely clustered within the high-dimensional space. In the high-dimensional vector spaces generated by our trained model, words and concepts are grouped together based on their contextual and semantic similarities. Words with similar meanings or those frequently appearing in similar contexts within the text are positioned near each other in the vector space. This clustering results from the model capturing co-occurrence patterns, allowing it to group analogous terms together. As a result, words related to similar topics or concepts form distinct clusters, enabling the identification of trends and shifts in research focus over the analysed period. The generation of the vector space continuum introduces a significant predicament, notably the alignment issue within the temporal vector space continuum. This arises due to the inherent disassociation between each vector space, which represents a month of knowledge, and the rest of the vector space continuums when subjected to mathematical representation. Such a phenomenon is a by-product of the word embedding creation process and the random initialization of weights. In this research we investigated two strategies aimed at addressing these challenges: the development of alignment methodologies and the utilisation of relative positioning among entities. The alignment issue arises from the independent generation of vector spaces for each month, leading to disassociation between these temporal vector spaces. This misalignment results from the random initialization of weights during the word embedding process, causing inconsistencies where the same word may be represented by different vectors across different months. This disjointedness

complicates the tracking of linguistic and conceptual evolution over time. To address this, we explored two strategies: developing alignment methodologies to harmonize the vector spaces post-training and employing relative positioning to maintain the consistency of word relationships across different temporal embeddings.

Discrete knowledge evolution construction method

In our exploration of constructing a vector space continuum, we have proposed a novel approach, termed herein as the "Discrete Knowledge Evolution Construction Method". This method involves training separate word embedding models for each time period, such as monthly intervals. As a result, a set of vector spaces is created, each representing different periods of the scientific literature as illustrated in Figure 28. For example, a monthly vector space would represent scientific publications published within a specific month.

The input to each job is a file that contains the textual data from the corresponding time period. The output of each training job is an embedding vector space that represents the textual data for that time period. One potential limitation of this approach is that each vector space is independent of the others, meaning that the vector numerical values of different time periods cannot be compared or combined in a meaningful way. For example, arithmetic calculations cannot be performed across different vector spaces, as they are not necessarily aligned. This may limit the ability to track changes in semantic meaning across time or to identify overarching themes or concepts that persist over multiple time periods.

The generated vector spaces can be utilised to track the relative distances between specific entities across the vector spaces, instead of solely focusing on tracking a single entity across the vector space. For instance, by observing the relative distance between the entity "transformers" and the entity "language" within the vector spaces, it becomes possible to project the evolution of the relationship between these two entities over time, indicating their changing closeness or proximity over time. Nonetheless, the attempt to chronologically monitor the numerical representation of the word embedding vector of a given entity across time, for instance 'transformers,' may prove inconsequential. This arises from the premise that each numerical vector representation maintains autonomy from the rest. This situation could be described as a vector space alignment conundrum.

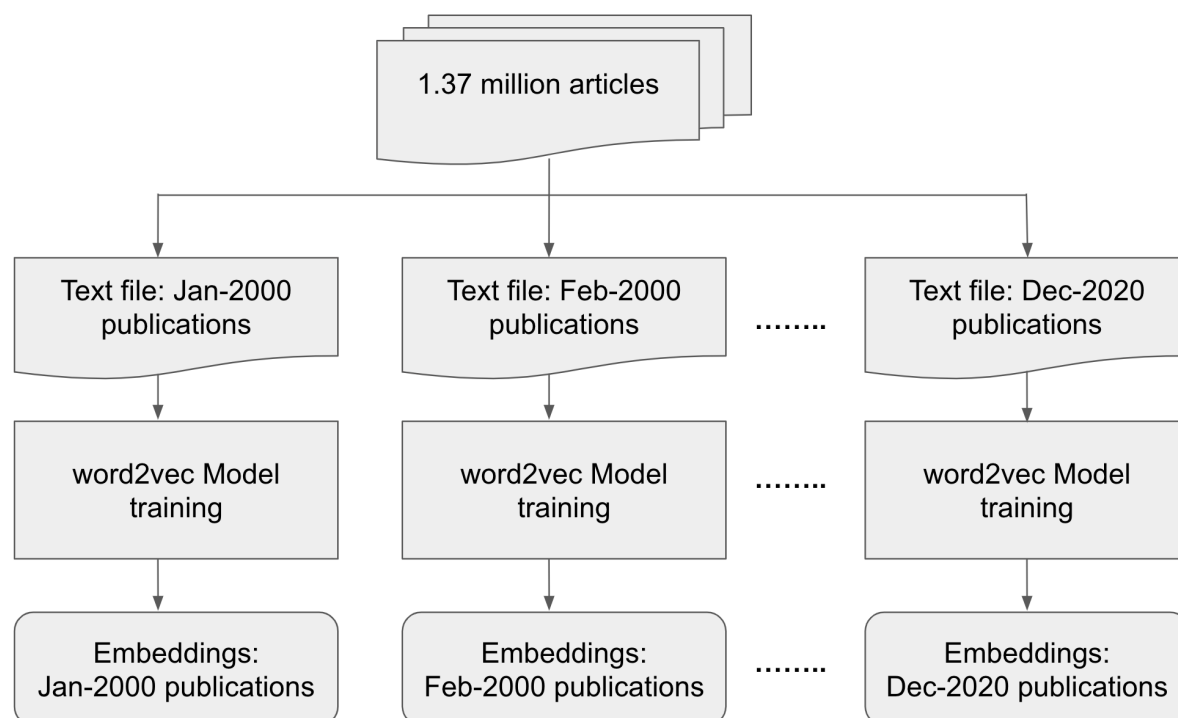


Figure 28: Discrete Knowledge Evolution Construction Method

Continuous knowledge evolution construction method

To address the vector space alignment conundrum, present within temporal vector spaces, and to facilitate consistent tracking of numerical weights in temporal word embeddings, we introduce a novel paradigm identified as the "Continuous Knowledge Evolution Construction Method". This method is designed to train a continuous function that maps time to the corresponding vector space. This can be achieved by building an initial vector space model based on the initial dataset for instance the first month of data, and then retraining the model by adding each month's data incrementally and extracting the corresponding embedding after each training step as illustrated in Figure 29. Each training cycle will incorporate data from all previous months along with the current month, utilizing a "continual training" approach to generate the vector space representation for the current month. By using this approach, it is possible to create a vector space continuum that captures changes in semantic meaning over time, providing a useful tool for applications such as event detection and trend analysis. These vector space continuum of embeddings allow for a richer representation of the underlying content, enabling researchers to track the evolution of scientific ideas and discoveries over time.

The generated vector space continuum, in this case, will remain consistent when representing a vector space for any given entity across time. Comparing the numerical vectors representing the same entity over time will coherently demonstrate the change in vector position over time. For instance, the numerical vector representing the entity "Apple" will experience significant changes over time, particularly when the company "Apple" was established. This is due to the underlying shift in meaning. Similarly, the vector representing the entity "transformer" will exhibit significant variance due to the introduction of the "transformers" model in the field of [NLP](#).

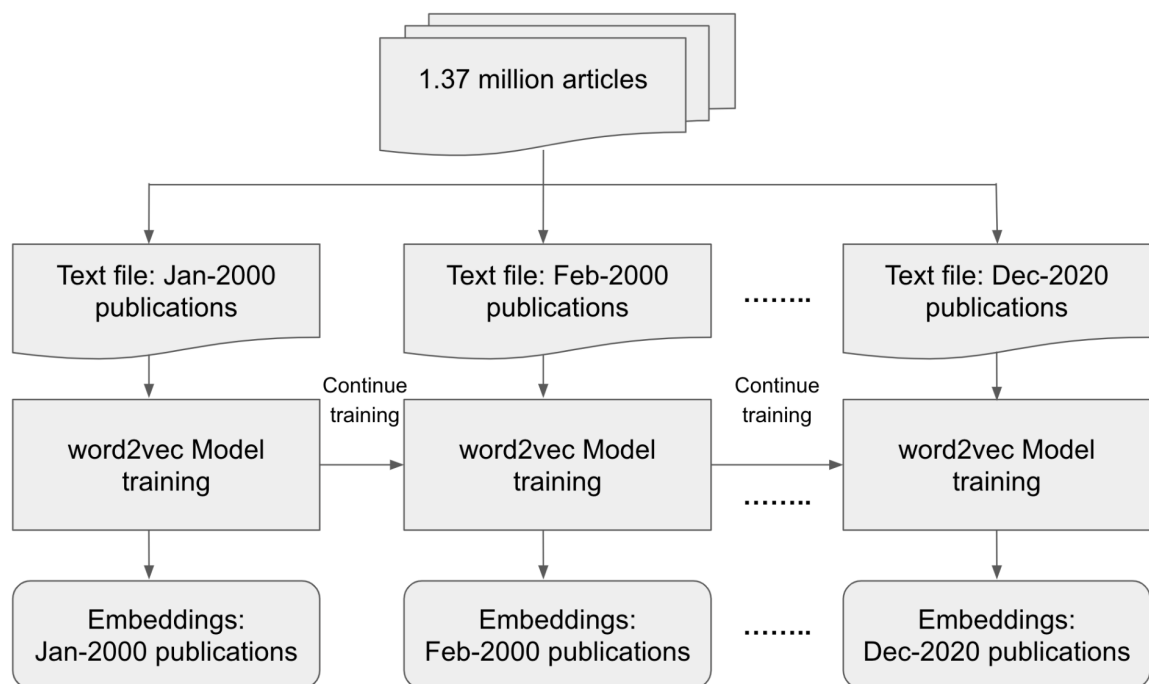


Figure 29: Continuous Knowledge Evolution Construction Method

Constructing the vector space continuum

Both the discrete and continuous knowledge evolution construction methods were evaluated for building the vector space continuum. While both methods are suitable for this purpose, we ultimately chose to utilise the continuous knowledge evolution construction method. This decision was based on the method's ability to examine the relative positions of entities to each other as well as the absolute position of a specific entity within the vector space. The following Algorithm 2 demonstrates the pseudocode of the implementation of both methods and how the system builds its knowledge.

Algorithm 2a Discrete Knowledge Evolution Construction Method

Require: *files* discrete_knowledge_evolution_construction_files

```
1: vector_spaces  $\leftarrow$  []
2: for file in files do
3:   text_data  $\leftarrow$  read_text_file(file)
4:   vector_space  $\leftarrow$  train_word_embedding_model(text_data)
5:   vector_spaces.append(vector_space)
6: end for
7: return vector_spaces
```

Algorithm 2b Continuous Knowledge Evolution Construction Method

Require: *initial_dataset*, *monthly_datasets* continuous_knowledge_evolution_construction_initial_dataset, monthly_datasets

```
1: vector_space  $\leftarrow$  train_word_embedding_model(initial_dataset)
2: vector_spaces  $\leftarrow$  [vector_space]
3: Combined_data  $\leftarrow$  initial_dataset
4: for monthly_data in monthly_datasets do
5:   combined_data  $\leftarrow$  combine_data(combined_data, monthly_data)
6:   vector_space  $\leftarrow$  retrain_word_embedding_model(vector_space, combined_data)
7:   vector_spaces.append(vector_space)
8: end for
9: return vector_spaces
```

Algorithm 2c Example usage

```
1: monthly_datasets  $\leftarrow$  [read_text_file("january_data.txt"),
2:   read_text_file("february_data.txt"), ...]
3: initial_dataset  $\leftarrow$  read_text_file("first_month_data.txt")
4: discrete_vector_spaces  $\leftarrow$  discrete_knowledge_evolution_construct(monthly_datasets)
5: continuous_vector_spaces  $\leftarrow$  continuous_knowledge_evolution_construct(initial_dataset,
6:   monthly_datasets)
```

Algorithm 2: Pseudo-code for the discrete and continuous knowledge evolution construction method

A model such as [word2vec](#) can be retrained on new data, with the embeddings for the same words likely to change due to the updated training process. This is because [word2vec](#) generates embeddings that are relative to the other words in the dataset and the context in which they appear, meaning that small changes in the dataset or the training parameters can result in significantly different embeddings. For example, if a new scientific discovery or a new scientific term becomes popular in the literature, retraining the [word2vec](#) model on the updated dataset will result in new embeddings for that term that reflect its updated context and meaning. Similarly, changing the hyperparameters of the [word2vec](#) model, such as the dimensionality of the embedding space or the context window size, will also lead to changes in the

resulting embeddings. Therefore, when working with time-series data, it is important to be aware that the embeddings are relative to the training dataset and parameters, and that retraining the model on new data will likely result in different embeddings for the same words. This can be particularly important for downstream applications that rely on the relative distances and relationships between the embeddings, such as clustering or classification. If a future term appears in the corpus, the embeddings will be entirely new and can be zeroed in all past instances where the entity didn't exist.

In the initial phases of our experiments, we employed both methods to generate vector spaces for each month of the scientific literature in the arXiv dataset. However, given the large size of the dataset and the computational resources required to train each model, we found that the Continuous Knowledge Evolution Construction method was more suitable for our purposes. This method involves training a foundational model on the initial dataset and then continuously retraining the model with the addition of new data, allowing for the creation of a seamless vector space continuum that reflects changes in the underlying knowledge over time. Both the Continuous and Discrete Knowledge Evolution Construction methods provide us with the capacity to maintain the relative positioning of entities. However, the Continuous Knowledge Evolution Construction further enables the preservation of the individual entity embedding representation's relevance throughout the temporal representation. Consequently, we opt to employ the latter methodology. It is, however, crucial to note that the adoption of the latter method introduces an additional computational resource burden due to the retraining process required for the aggregated data. We performed the training in chronological order, allowing for a more accurate reflection of the evolving knowledge over time in the vector space continuum.

To represent this mathematically through a three-dimensional matrix, also referred to as a tensor in the context of data representation, we created a mathematical construct to capture the multifaceted relationships to represent the temporal representation of the knowledge. The tensor encapsulates three distinct dimensions: entity representation, entity embedded representation, and temporal representation.

For the first dimension, each entity is a specific word, and we denote this dimension as E . Mathematically, we can represent an entity e within this dimension as E_e , where e is the index that corresponds to a unique entity. The second dimension represents

the embedded representation of the entities, expanding the information content through a 300-dimensional vector. This dimension, denoted as V , which carries a rich information about the entity that positions it in the vector space continuum. Each embedded representation can be notated as $V_{e,i}$, where e corresponds to the specific entity, and i is the component of the 300-dimensional vector. The third dimension embraces the temporal perspective, providing a time-lapse view of entity changes over a course of 240 months, denoted as T . Each time-step can be denoted as T_t , with t representing a specific month. Thus, the three-dimensional matrix, symbolised as M , that embeds these three dimensions can be mathematically defined as $M_{e,i,t}$, where $e \in E$, $i \in V$, and $t \in T$. This sophisticated representation enables a deep, multidimensional analysis of entities, encompassing not only the inherent characteristics of the entities through embedding but also their evolution over time.

6.2.4. Constructing temporal sequences of preselected entities: a synthesis of chronological data

Upon the establishment of the vector space continuum, its exploitation can initiate. The optimal scenario envisages a researcher tracing the trajectory of distinct terminologies, examining their evolution over time in relation to both them and other terms. This process has the potential to reveal anomalies, instigate event detection, guide the trajectory of research, confirm scientific hypotheses for evidence-based decision-making, and more. However, the corpus, consisting of more than 11 billion words, presents formidable computational challenges. These are especially pronounced when attempting to determine relationships among all the word pairs over our 240-month investigation period. Consequently, our preliminary strategy involves devising a filtering mechanism to curate the corpus of billions of words through a rigorous set of selection criteria.

Quantifying the relatedness of pre-selected entities

The objective of this study is to detect variations in the vector space continuum that may signify the occurrence of significant events, such as scientific discoveries or the emergence of new technologies and methodologies. Given the intricate fabric of the vector space, it can be challenging to monitor the independent entities and their variations over time. Therefore, in this experiment, we opted to calculate the Euclidean distance between pre-selected entities in the knowledge base. Specifically, we utilised

cosine similarity between 5000 pre-selected entities and computed the cosine similarity between each entity over a period of 240 months, resulting in a vector space of size 25,000,000 x 240. The matrix contains values representing the cosine similarity between the analysed entities. These values are relative to each other within the monthly vector spaces, regardless of the construction method employed, whether discrete or continuous knowledge evolution construction. It is important to emphasise that the cosine similarity is calculated irrespective of the construction technique used.

Entity selection criteria

The process of selecting the most relevant entities for the purpose of this study, which is to detect events such as scientific discovery or emergence of new methodologies, was carried out through the application of entity selection techniques. The process operates under the assumption that the pre-selected entities with the highest relevance to each other would make the most significant contributions in showcasing the system's ability to detect knowledge evolution events. However, the success of the event detection process is reliant on the human interpretation of the variation in the vector space over time and its alignment with scientific discoveries or events. Therefore, the selection of entities is subject to the knowledge and expertise of the team conducting the experiment. The methodology used to perform the entity selection process involved a series of steps illustrated in the Algorithm 3. Firstly, the most common English words were removed from the list. Secondly, regular expressions were used to eliminate unknown acronyms. Regular expressions were employed to eliminate unknown acronyms from the dataset. By defining a pattern to match sequences of letters, typically indicative of acronyms, these expressions facilitated the identification and removal of such terms from the text. Additionally, we curated a list of commonly used acronyms through manual extraction, allowing us to retain relevant acronyms while filtering out infrequent or unknown ones. This preprocessing step enhanced the dataset's quality for training the word embedding model. Thirdly, rare words that have minimal relevance to the domain knowledge were removed. Finally, a manual filtering process was employed where thousands of entities were evaluated based on their knowledge and expertise in the domain. All manual processes are performed and stored in a [CSV](#) format to be later integrated with the process.

Algorithm 3 Entities pre-selection criteria

```
1: Step 1: Remove most common English words from the list
2: common_words  $\leftarrow$  get_common_english_words()
3: entities_list  $\leftarrow$  remove_common_words(entities_list, common_words)
4: Step 2: Eliminate unknown acronyms using regular expressions
5: unknown_acronyms  $\leftarrow$  find_unknown_acronyms(entities_list)
6: entities_list  $\leftarrow$  remove_unknown_acronyms(entities_list,
   unknown_acronyms)
7: Step 3: Remove rare words with minimal relevance to domain knowledge
8: rare_words  $\leftarrow$  find_rare_words(entities_list, relevance_threshold)
9: entities_list  $\leftarrow$  remove_rare_words(entities_list, rare_words)
10: Step 4: Manual filtering Filtering process based on knowledge and expertise
11: selected_entities  $\leftarrow$  []
12: for entity in entities_list do
13:   if evaluate_entity(entity, team_knowledge, domain_expertise) then
14:     selected_entities.append(entity)
15:   end if
16: end for
17: Result: selected_entities contain the final list of entities
```

Algorithm 3: Pseudo-code for entities pre-selection criteria

Construction of temporal entity sequences

Upon the formulation of the preselected entity list, predicated on the designated criteria, we initiated the process of loading the 240 vector spaces into memory. This process was facilitated by leveraging the computational capabilities of high-performance machines in the cloud. For our computational needs, we developed our models on an [AWS](#) Elastic Compute Cloud (EC2) instance, powered with 128 [CPUs](#) and a memory of 256[GB](#). The expansive storage capabilities of Amazon [S3](#) buckets were harnessed for our data and model storage, offering us considerable latitude in terms of storage size. Subsequently, we embarked on calculating the interrelationships among the 5000 preselected entities. To illustrate, we selected two entities, entity 1 and entity 2, and computed the cosine similarity of their vectors across 240 disparate vector spaces. This computation generates a novel numerical vector comprising 240 numerical values, thereby encapsulating the relational dynamics between the two entities over a span of 240 months. To calculate the cosine similarity across the temporal dimension for two entities, we'd use the formula for cosine similarity, which is defined as the dot product of the two vectors divided by the product of their magnitudes.

The formula for cosine similarity between two vectors is typically defined as:

$$\text{CosSim}(a, b) = \left(\frac{a \cdot b}{\|a\| \|b\|} \right)$$

In this context, we have two entities e_1 and e_2 , and we want to calculate the cosine similarity of their embedded representation vectors across each time step. Let $M_{e_1, t}$ and $M_{e_2, t}$ represent the embedded vectors for entities e_1 and e_2 at time t , respectively. " $\cdot$ " denotes the dot product, and $\|\cdot\|_2$ is the $L2$ norm (Euclidean distance) of a vector. We can then compute the cosine similarity at each time t as:

$$[C_{e_1, e_2, t} = \frac{M_{e_1, t} \cdot M_{e_2, t}}{\|M_{e_1, t}\|_2 \cdot \|M_{e_2, t}\|_2}]$$

By iterating t over the span of 240 months, we would obtain a time series of cosine similarities between the two entities e_1 and e_2 . This series, denoted as $S_{e_1, e_2, t}$, would yield a robust depiction of the evolving relationship between these two entities over the considered temporal horizon. It provides a deep insight into the changes and dynamics of their semantic relatedness with respect to time.

Finally, to create a mathematical representation that captures the relationship between every possible pair of entities over time, we will generate a four-dimensional tensor S , where each element $S_{e_1, e_2, i, t}$ is the cosine similarity between the i th components of the embeddings of entities e_1 and e_2 at time t . To illustrate, we commence with the entity e_1 and proceed to compute the cosine similarity with successive entities $e_2, e_3, \dots, e_t$. This computational process is carried out iteratively across all temporal slices denoted by t where t represents time and i in this case, represents the month number. Hence:

$$S_{e_1, e_2, i, t} = \frac{M_{e_1, i, t} \cdot M_{e_2, i, t}}{\|M_{e_1, i, t}\|_2 \cdot \|M_{e_2, i, t}\|_2}$$

This tensor S enables an intricate understanding of how the relationships between any two entities evolve over time, captured through the angle subtended by their respective embedded representations. Algorithm 4 demonstrate the pseudocode for the construction of temporal entity sequences.

Algorithm 4 Calculate Entity Relationships Using Cosine Similarity

```
1: Load vector spaces into memory
2:  $vector\_spaces \leftarrow load\_vector\_spaces()$ 
3: Initialise a dictionary to store entity relationships
4:  $entity\_relationships \leftarrow \{\}$ 
5: Calculate cosine similarity for each pair of preselected entities across all
   vector spaces
6: for  $i$  in range(5000) do
7:   for  $j$  in range( $i + 1$ , 5000) do
8:      $entity\_1 \leftarrow preselected\_entities[i]$ 
9:      $entity\_2 \leftarrow preselected\_entities[j]$ 
10:    Calculate cosine similarity across all vector spaces
11:     $cosine\_similarities \leftarrow [cosine\_similarity(vector\_spaces[k][entity\_1],$ 
       $vector\_spaces[k][entity\_2]) \text{ for } k \text{ in range}(240)]$ 
12:    Store the cosine similarities in the  $entity\_relationships$  dictionary
13:     $entity\_relationships[(entity\_1, entity\_2)] \leftarrow cosine\_similarities$ 
14:   end for
15: end for
```

Algorithm 4: The pseudocode for Construction of temporal entity sequences

Event detection in knowledge evolution

The results of the relatedness calculation have been stored in a big data platform as a vector database, in which the 240 months are represented as columns and the 25 million entity-to-entity relationships are represented as rows, over the span of 20 years of scientific literature grouped monthly. Table 6 provide an example of how the vector database that store entities and its relationship to each other.

Table 6: Example of how the vector database that store entities and its relationship to each other.

Entity_ID_1	Entity_ID_2	Month_1	Month_2	Month_3	...	Month_240
Entity_1	Entity_2	0.95	0.85	0.92	...	0.78
Entity_1	Entity_3	0.7	0.8	0.75	...	0.83
Entity_1	Entity_4	0.6	0.7	0.65	...	0.71
Entity_2	Entity_3	0.8	0.9	0.82	...	0.76
...	...	...	...	...	...	...

The big data platform that stores the pre-selected entities vectors is used to study specific entities and their evolution over time. The aim of this study is to detect how the relationships evolve over time. The hypothesis here is that the entities under study will exhibit changes in their vector spaces over time that represent an event like a new development of a methodology in a specific area like “transformers” with “language”.

The pre-selected entity vectors are stored in a big data platform, which is utilised for studying specific entities and their evolution over time. The objective of this study is to detect how the relationships between these entities evolve over time. The hypothesis is that the entities under investigation will undergo changes in their vector spaces over time, indicating significant events such as the development of a new methodology in a specific area, for example, the relationship between "transformers" and "language" in the field. By analysing the vector spaces of these entities over time, it becomes possible to observe and track the changes in their relationships and proximity. For instance, the vector representations of "transformers" and "language" can provide insights into the evolution of their connection, reflecting the advancements and developments in the field related to these concepts.

6.3. Experimental Design

The objective of the experiment is to examine the evolution of temporal knowledge using the arXiv dataset. This dataset is considered a large dataset, consisting of 1.7TB of PDF files, encompassing 2.3 million articles. Given the magnitude of the data, the experimental design has been carefully crafted to employ cloud-based resources for data processing, model training, and processing the subsequent datasets. In this section, we will provide a high-level overview of the process, followed by an elaboration on the cloud-based solution architecture. Later sections will dive deeper into the primary components of the experiments. Broadly, the experiment is segmented into three core subprocesses: text pre-processing, document classification, and the temporal knowledge evolution process.

6.3.1. arXiv DataStream: From PDFs to vector space analysis

The initial subprocess pertains to text pre-processing. The arXiv dataset is accessible for download from the Amazon [S3](#) bucket. However, it is noteworthy that individuals downloading this data will incur [AWS](#)-related charges. The procedure commences with downloading data from Amazon [S3](#) bucket. This data is retrieved in a compressed format, necessitating a decompression step. Given the large size of the dataset, the decompression process takes large computational resources and takes a long time to finish. Most arXiv files are in PDF format, so a subsequent step involves converting these PDFs into text files. This conversion presents certain challenges, including

issues with file layouts, special characters, linguistic support, among other complications. Following the conversion, we obtain text files that encapsulate the content extracted from the original PDFs. It is crucial to ensure the machine undertaking this operation possesses ample disk space to manage decompression, various processing stages, and the storage of multiple iterations. The processing for all 2.3 million articles was automated. It is essential to note that this phase posed various challenges, primarily due to the data's sheer size and inherent complexities in the stored PDFs. The automation yields a series of refined text files. These files then undergo several [NLP](#) operations, including tokenization, stemming, lemmatization, the elimination of stop words, conversion to lowercase, removal of punctuation and numerals, whitespace management, encoding and decoding adjustments, and spelling corrections. Encoding and decoding refer to the processes of converting text between different character formats to ensure consistency and accuracy during data processing. Encoding involves transforming text into a specific character encoding format, such as UTF-8, to accurately represent and store characters, including special symbols and multilingual text. Decoding is the reverse process, converting encoded byte data back into readable text characters, ensuring correct interpretation across different systems. In this experiment, encoding and decoding adjustments ensure that text extracted from PDFs is accurately represented, allowing for uniform processing in subsequent natural language processing tasks. The concluding step in this phase is to categorise the data into monthly files, with each file encapsulating the articles published in a distinct month, with each line representing a unique article.

The second subprocess focuses on document classification. This phase initiates with a manually curated list of categories and classifies the files for each category. This reduction ensures the derivation of high-quality embeddings while minimising potential noise from unrelated articles. Following this, a dataset is established, the particulars of which will be explained later in this chapter. This classified dataset is used to train a [LSTM](#) document classifier based on the parameters deduced from the results of the previous experiment in Chapter [5](#). The classification labels employed are Mathematics, Statistics, Physics, Quantitative Finance, Computer Science, Quantitative Biology, Electrical Engineering and Systems Science, Economics, Quantum Physics, and Cosmology. These classes were determined based on the

most common major topics in the dataset. Using this classifier, monthly arXiv files are processed, resulting in a series of classified monthly arXiv files.

The final subprocess encompasses the creation of a vector space continuum. This is achieved using the Continuous Knowledge Evolution Construction Method, constructing the vector spaces on a month-by-month basis, before integrating them to form a comprehensive vector space continuum. Due to the extensive nature of the data and the numerous models, this operation was carried out on a large-scale compute service via [AWS](#) cloud. The resultant vector spaces are then employed to study specific entities, the criteria for which will be detailed later in this chapter. Following this, a procedure is executed to evaluate the interrelations between these entities, storing the relational data on Amazon Athena which is a large-scale database. This database can subsequently be queried and interfaced with a cloud-based visualisation tool, enabling the examination of scientific trends and the identification of significant scientific events. Figure 30: Workflow of the temporal knowledge evolution framework, which is segmented into three stages: text pre-processing, document classification, and temporal knowledge progression.illustrates the three segments of the workflow of the temporal knowledge evolution framework.

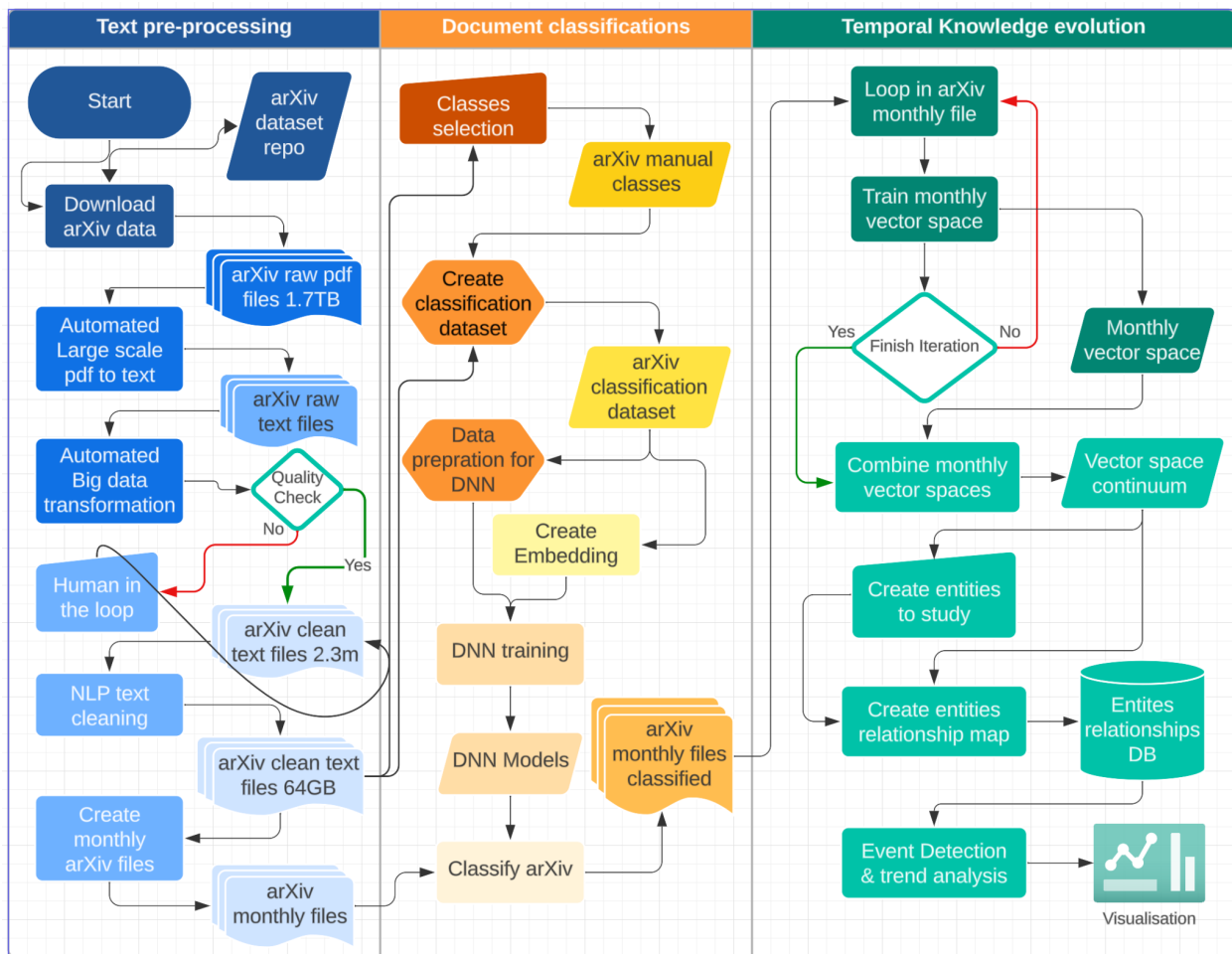


Figure 30: Workflow of the temporal knowledge evolution framework, which is segmented into three stages: text pre-processing, document classification, and temporal knowledge progression.

6.3.2. arXiv DataStream AWS solution architecture

This section outlines experimental approaches to process, classify, and analyse the large arXiv dataset, leveraging the state-of-the-art [AWS](#) advanced services. From data retrieval and text pre-processing to document classification and vector space creation, [AWS](#)'s suite of services - including Amazon [S3](#) bucket, [AWS](#) Batch, Amazon SageMaker, Amazon Elastic Container Service, and Amazon Athena, Amazon QuickSight - proved instrumental. Here's a deep dive into the nuanced processes and how we leverage [AWS](#) services for each step to drive scientific discovery and trend analysis as shown in Figure 31.

Text pre-processing using AWS services

The arXiv dataset is readily available in an Amazon [S3](#) bucket, or Simple Storage Service, is a scalable object storage solution by [AWS](#), which is widely used for backup,

archival, and web-scale applications. Utilising Amazon [S3](#) bucket, we initiate the process by downloading the dataset. Given that the dataset is compressed, the first task is decompression. Considering the enormity of the data, we leverage [AWS](#) Batch, a service that efficiently runs batch computing workloads, to handle the resource-intensive decompression task efficiently. Most of the arXiv files are in PDF format, which necessitates a conversion into text. This conversion, due to its intricacies such as layout differences and linguistic variations, is tackled using high-capacity Amazon Elastic Container Service ([ECS](#)). [ECS](#) is a container management service that allows easy scalability and high performance for containerized applications. Post conversion, the text files derived from the original PDFs are stored, and it's vital to ensure adequate disk space Amazon Elastic Block Store (EBS), offering the necessary disk storage in conjunction with [ECS](#), to manage different stages of processing and multiple versions. The final files are stored on Amazon [S3](#) bucket. The automation for processing the 2.3 million articles utilises [AWS](#) batch integrated with Amazon SageMaker, especially for its [NLP](#) functionalities like tokenization, stemming, lowercase, removing punctuation, and managing whitespace and lemmatization. Ultimately, the refined data is segmented into monthly files stored in Amazon [S3](#) bucket, with each file corresponding to articles of a specific month. Amazon SageMaker is a fully managed service that assists in building, training, and deploying [ML](#) models,

Document classification with AWS

For the document classification phase, we start with a manually curated category list to manage the number of files in each category. This reduction helps derive better embeddings by minimising potential noise. Using Amazon SageMaker, a dataset is curated, and then the [LSTM](#) document classifier is trained. Subsequently, the monthly arXiv files are classified through this model, resulting in a set of classified monthly files stored on Amazon [S3](#) bucket.

Vector space continuum creation using AWS

To create a vector space continuum, we implement the Continuous Knowledge Evolution Construction Method which will be discussed later in this chapter. Given the voluminous nature of the data and the multitude of models, this operation is executed on a large-scale using Amazon SageMaker. The resulting vector spaces are used for entity studies. The relationships between these entities are evaluated and stored on

Amazon Athena, an interactive query service, making data analysis seamless with its serverless architecture. Athena facilitates querying and is integrated with cloud visualisation tools Amazon QuickSight, [AWS](#)'s business intelligence service, enabling a deep dive into scientific trends and pivotal scientific occurrences.

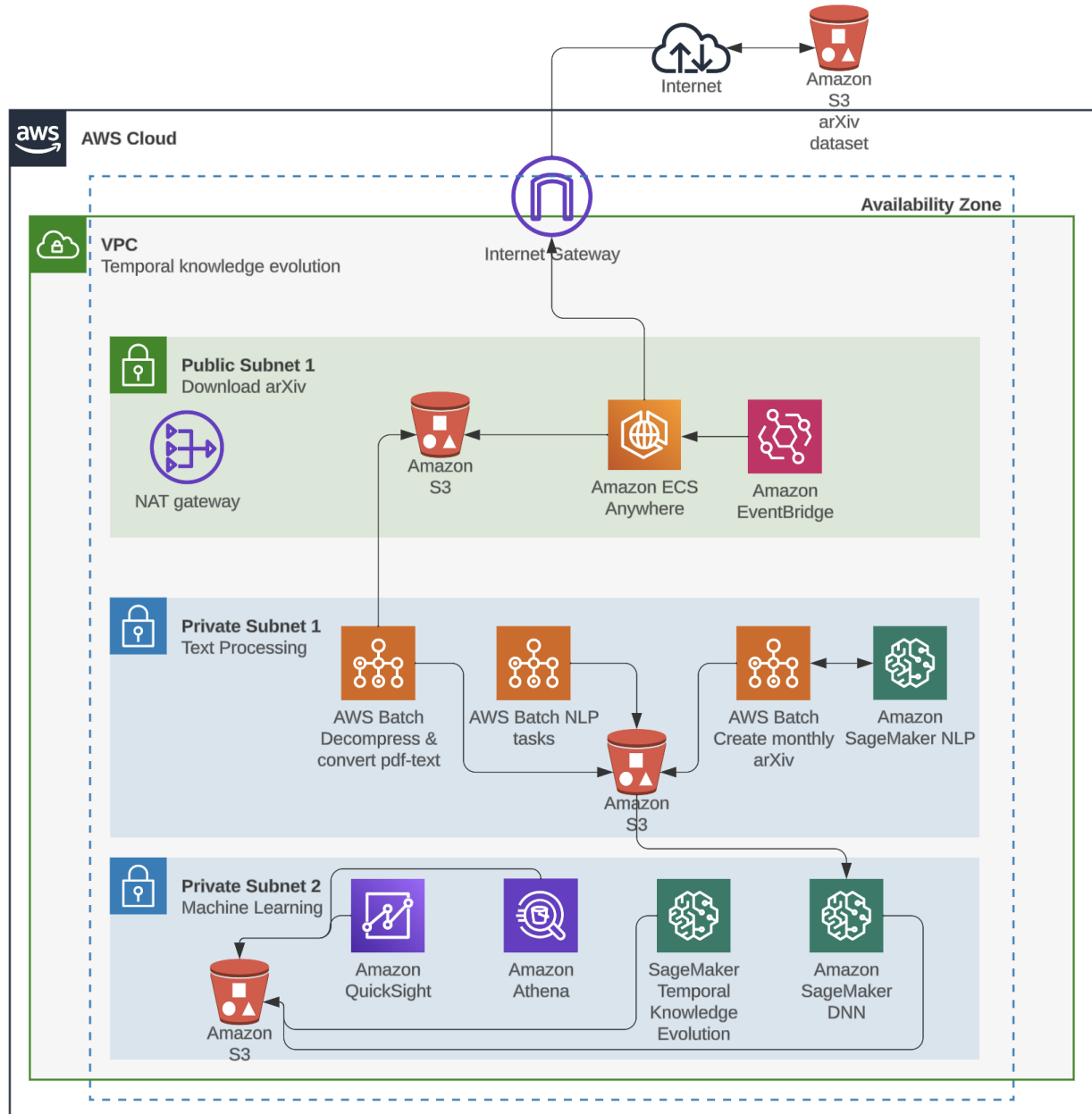


Figure 31: Solution architecture of the temporal knowledge evolution framework on [AWS](#) Cloud

6.4. Results

6.4.1. Sequence analysis of temporal entity relationships

The analysis of a vast dataset containing 25 million records of time series entity relationships necessitates meticulous processing. To tackle this, we employed a technique aimed at identifying significant trend curves that indicate changes in the relationships over time. Using an unsupervised k-means approach, we clustered the temporal sequences of entity relationships. We experimented with various values of k and determined that four predominant curve types effectively capture the majority of the observed trends: "EternalElevation," "TimelessTide," "PhantomFade," and "PhoenixRise." Below, we provide explanations of these major trends along with accompanying example trend graphs.

Polynomial regression analysis: a curve fitting approach for unveiling trends in data series

In order to visually represent the newly discerned key trends, we employed curve fitting, a form of regression analysis widely utilised in both mathematics and statistical modelling. This method involves identifying a curve or a mathematical function that aptly represents a series of data, or, in our specific case, the time series of temporal entity relationships. The objective is to encapsulate the data trend with a simple, yet analytical function that can then be deployed for interpolation or extrapolation as per the requirements. Curve fitting can be implemented using a variety of functions. The underlying trend and variations in the data principally determine the type of function to be employed. Key mathematical models encompass Linear Models, Polynomials, Exponential Models, Logarithmic Models, Sinusoidal Models, and Power Law Models. Given the trends evident in our dataset, we've chosen the Polynomial model. In mathematical terms, a polynomial is defined as a sum of terms, each consisting of variables raised to a power and multiplied by a coefficient. When it's in one variable, the polynomial can be represented as:

$$p(x) = a_n \cdot x^n + a_{n-1} \cdot x^{n-1} + \dots + a_2 \cdot x^2 + a_1 \cdot x + a_0$$

where:

- x is the variable
- n is the degree of the polynomial
- $a_0, a_1, \dots, a_n$ are the coefficients of the polynomial.

The degree of a polynomial function is identified by the highest power of the variable x present in the polynomial equation. For instance, in a polynomial in the above function, the degree is n , which is the largest exponent of x . Polynomial functions are selected for their versatility in fitting diverse data patterns, ranging from simple linear trends to complex non-linear relationships. This versatility allows them to model various data trends effectively. Moreover, polynomial functions are relatively simple mathematically, making them easier to manipulate and interpret compared to more complex models. The coefficients and terms in a polynomial equation provide direct insights into variable relationships, which enhances interpretability. The degree of the polynomial plays a critical role in analysis. Low-degree polynomials, like linear or quadratic functions, are simpler and less likely to overfit, making them easier to interpret. However, they might not capture complex trends or subtle data nuances, potentially leading to underfitting. In contrast, high-degree polynomials can capture intricate patterns more effectively and provide a closer fit to observed data points. However, they are more prone to overfitting, where the model becomes too tailored to the specific dataset, losing generalizability. Additionally, higher degrees increase model complexity, making interpretation more challenging.

In this study, polynomial functions with degrees around five are used to strike a balance between capturing complex, non-linear relationships and avoiding excessive overfitting while maintaining interpretability. This choice ensures that the models adequately capture the inherent trends in time series data and remain manageable for further analysis and application. This approach provides a robust framework for analysing the temporal evolution of entity relationships, allowing for interpolation within observed data ranges and reasonable extrapolation beyond them. This methodology is demonstrated in the thesis, where polynomial regression is applied to fit data trends and identify significant patterns and events in temporal sequences of entity relationships. By using polynomial degrees around five, detailed trends are effectively captured, avoiding overfitting and preserving the interpretability of results.

This research introduces several new terms to describe trends observed in the sequence analysis of temporal entity relationships. “EternalElevation” depicts a consistently upward relationship between two entities over time without any decline. “PhantomFade” describes a trend where a strong initial relationship gradually diminishes. “PhoenixRise” indicates a pattern where entities initially absent later

appear, showing an upward relationship trajectory, often seen with newly introduced terms or acronyms. Conversely, “PhoenixFall” represents entities that emerge later but then exhibit a downward trajectory in their relationships, also common among new terms or acronyms. Lastly, “TimelessTide” characterizes relationships that fluctuate continuously, akin to ocean tides, throughout the observed period.

Curve 1: EternalElevation

The EternalElevation curve represents a continuous and upward relationship between two entities over time, without any indication of decline. This curve in Figure 32 showcases a consistently increasing trend in the relationship between the entities without any downward movement. The polynomial function, representing the trend of 'EternalElevation', is deduced by averaging various polynomial functions identified throughout the study, as depicted below.

$$p(x) = 1.514e - 12 x^5 - 5.22e - 10 x^4 + 1.82e - 07 x^3 - 1.05e - 05 x^2 + 0.0008205 x - 0.1763$$

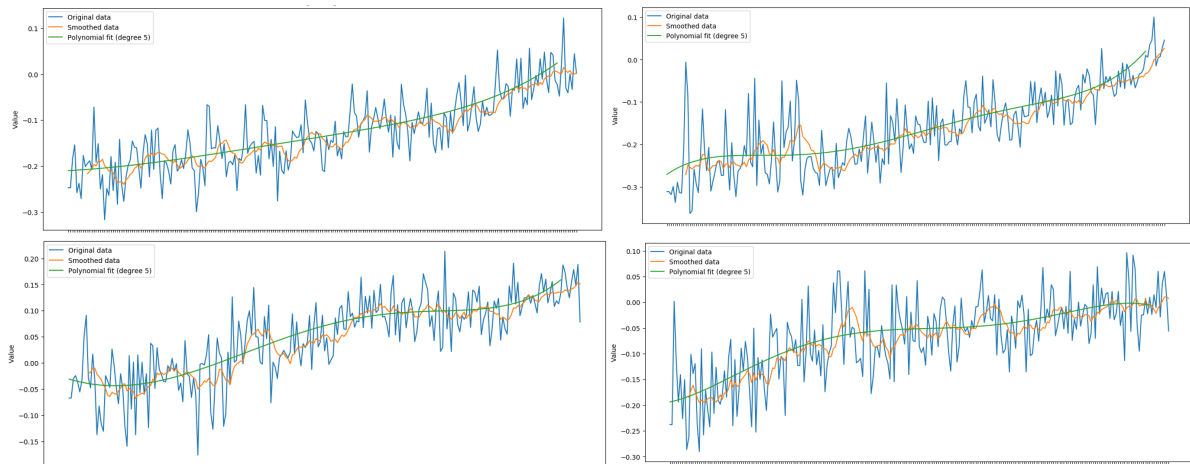


Figure 32: EternalElevation trend function examples

Curve 2: PhantomFade

The PhantomFade curve is a trend observed in the time series of a relationship between two entities, indicating a strong initial relationship as depicted in the literature, which gradually diminishes over time. This curve in Figure 33 signifies a fading of the relationship in the later stages of the timeseries. The polynomial function, representing the trend of PhantomFade, is deduced by averaging various polynomial functions identified throughout the study, as depicted below.

$$p(x) = -1.5e - 12 x^5 + 8.59e - 10 x^4 - 1.91e - 07 x^3 - 1.33e - 05 x^2 - 0.001525525 x + 0.543$$

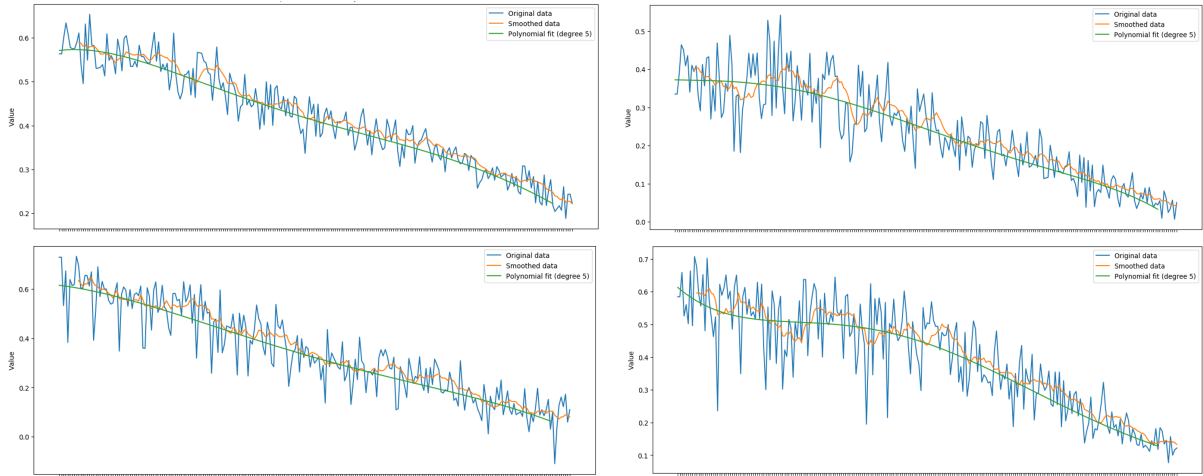


Figure 33: PhantomFade trend function examples

Curve 3: PhoenixRise

The PhoenixRise curve corresponds to a trend observed in the entity relationship time series, wherein one or two entities were absent in the initial stages of the literature corpus but later came into existence. This curve in Figure 34 indicates a lack of relationship at the early stages of the timeseries, followed by an upward trajectory in the relationships. This pattern is typically observed in newly introduced terms or acronyms. The polynomial function that represents the trend of ‘PhoenixRise’ is shown below.

$$p(x) = -1.25925e - 12x5 + 7.5275e - 10x4 - 1.7835e - 07x3 + 2.46125e - 05x2 - 9.289e - 04x + 0.00456775$$

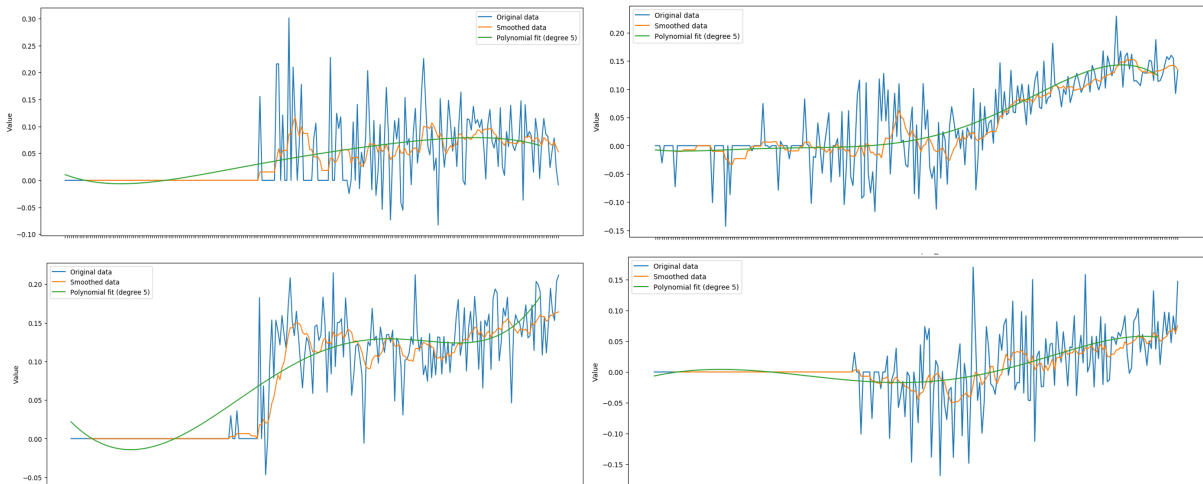


Figure 34: PhoenixRise trend function examples

Curve 4: PhoenixFall

The PhoenixFall curve corresponds to a trend observed in the entity relationship time series, wherein one or two entities were absent in the initial stages of the literature corpus but later appeared. This curve in Figure 35 indicates a lack of relationship at the early stages of the time series, followed by a downward trajectory in the relationships. This pattern is typically observed in newly introduced terms or acronyms. The polynomial function that represents the trend of 'PhantomFall' is shown below.

$$p(x) = (-2.899e - 12)x^5 - (1.883e - 09)x^4 + (3.386e - 07)x^3 + (9.12e - 05)x^2 - (0.000527)x + 0.0048537$$

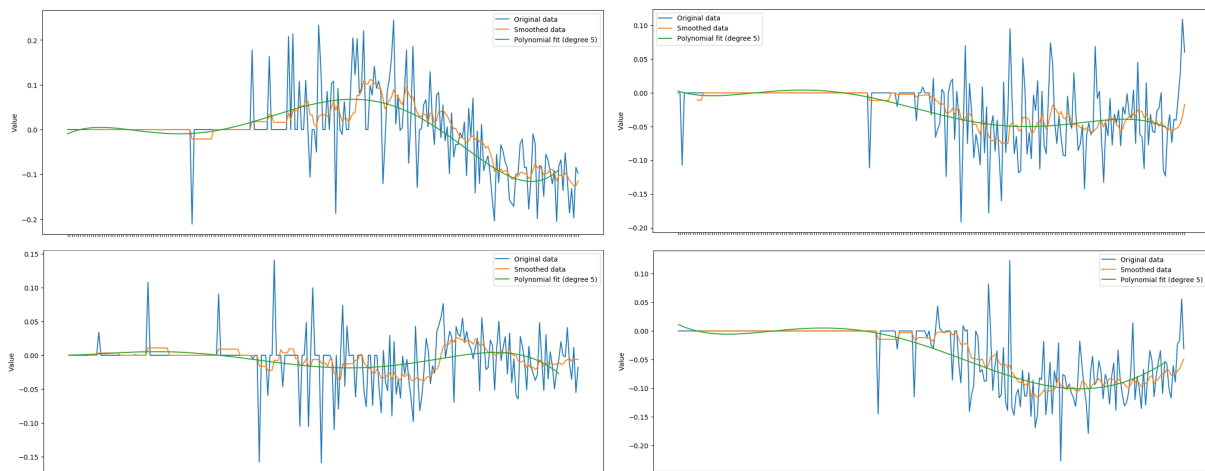


Figure 35: PhoenixFall trend function examples

Curve 5: TimelessTide

TimelessTide is a trend that characterises the relationship between two entities, resembling the oscillatory pattern of ocean tides. It exhibits a continuous fluctuation with high frequency throughout the observed time period. The example graphs in Figure 36 demonstrates slight variations in the curve, although their significance may not be crucial for trend analysis purposes. The polynomial function that represents the trend of 'TimelessTide' is shown below.

$$p(x) = 5.2275e - 13 x^5 - 2.5205e - 10 x^4 - 4.3575e - 08 x^3 + 3.8895e - 06 x^2 - 0.0003057 x + 0.0647375$$

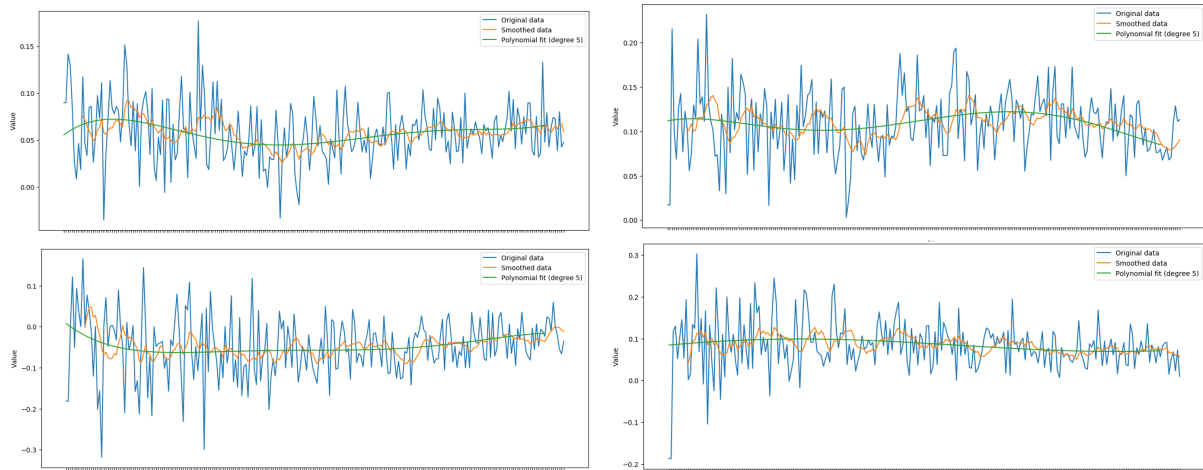


Figure 36: TimelessTide trend function examples

6.4.2. Capturing disruptive events and trends

Through our experiments, we were able to successfully identify significant scientific discoveries by studying the time series graph of a given target entity. In our approach, we utilised multiple co-entities to compare their similarity to the target entity, which enabled us to better understand the conformity of the vector space. By monitoring changes and anomalies in the time series, we can detect correlations to specific scientific discoveries or significant shifts in human behaviour towards a particular method or technology. In the following section, we will explore various events and analyse changes in the time series to identify potential correlations with significant scientific discoveries or shifts in research directions.

Given the substantial volume of the data, we have employed a cloud-based visualisation tool. This tool is intrinsically linked to a distributed query engine, specifically engineered to facilitate operations with big datasets. The visualisation that we used is the line graph, a graphical representation of data that is plotted on a two-dimensional plane, with the x-axis representing time and the y-axis representing the values of the measured data. The data points are connected by lines to depict a continuous line, allowing for a visual illustration of trends and patterns over time.

To illustrate the analysis, we will use one example of a specific entity, namely "bert", and examine its relationship with various other entities, including "transformer", "language", "image", "cancer", and "bidirectional". This comparison is facilitated by computing the cosine similarity between the vector representation of "bert" and the aforementioned entities over a timeframe of 240 months. Each pairing of "bert" with an individual term generates a line graph that denotes a mathematical correlation

between the two entities over time. An increase in this calculated value signifies a stronger correlation between the two entities.

The aim of our study is to unveil fluctuations within the chronology of vector space continuums. Such variations could potentially denote noteworthy occurrences, like the unveiling of innovative technological procedures, or shifts in academic research and literature towards particular methodologies. In the following section, we will focus on particular events that were clearly captured by our analysis and are consistent with the literature. By scaling this approach further, a researcher might be able to use the same method and analysis to obtain a quantified representation to support their research and opinions on a specific topic. Some of the events we spotlighted are well known to researchers in the field, and we used these instances to make sure our findings were accurate. We've also noted other events that are seen as a trend and are not categorised as events. These trending events demonstrate how our analysis can measure scientific trends to support a researcher's point of view.

Analysis of disruptive events

In this research, the term 'disruptive event' is defined as a noteworthy incident or phenomenon that triggers a profound alteration in the corpus of scientific literature. Such an event might synchronise with a significant breakthrough in a scientific domain, for example, the inception of a novel methodology, the discovery of an unprecedented material, or the redirection of the scientific community's focus towards a particular subject or individual.

To demonstrate the proficiency of our system, we examined specific entities of which we are aware of their scientific history and periods of substantial breakthroughs. Following their identification, we extracted the time series data of these selected entities under study, comparing it with unrelated and randomly selected entities to serve as a reference point. This approach allows us to determine if the trends we are measuring differ from those observed as reference points.

Figure 37 illustrates the interconnections between the entity "bert" and the entities "transformer", "language", "image", "cancer", and "bidirectional". The time series represented in the figure reveals a notable disruption in the vector space continuum occurring around October 2018, a date that coincides with the introduction of [BERT](#) (Bidirectional Encoder Representations from Transformers). This introduction occurred in a paper titled "[BERT](#): Pre-training of Deep Bidirectional Transformers for

Language Understanding", which was published by Google AI Language researchers and cited more than 73K times as of the time of writing. The time series representing the entities "bidirectional", and "transformer" show strong correlations, as well as a slightly shifted yet still strong correlation with the entity "language".

[BERT](#) is a pre-trained transformer-based model that can be fine-tuned for a variety of downstream tasks within the field of [NLP](#). These tasks encompass a broad spectrum, from question answering and sentiment analysis to named entity recognition. The model was initially trained on a vast corpus of unlabelled text, and thereafter fine-tuned for specific [NLP](#) tasks. This approach enabled it to achieve record-breaking performances on an array of benchmarks.

Despite being primarily tailored for [NLP](#) tasks, [BERT](#) has also been tested and adapted for image processing tasks. The pioneering research paper exploring this innovative application was titled "[ViLBERT](#): Pretraining Task-Agnostic Visiolinguistic Representations for Vision-and-Language Tasks" (Lu et al., [2019](#)). The paper introduced [ViLBERT](#) (Vision-and-Language [BERT](#)), a model designed to learn task-agnostic combined representations of image content and natural language. By observing Figure 37, particularly the time series line representing the relationship between the entities "image" and "bert", it becomes apparent that the disruption in the relationship between "image" and "bert" began to gain prominence in 2019, subsequent to the publication of the [ViLBERT](#) paper. As of the present date, this paper has received over 2500 citations.

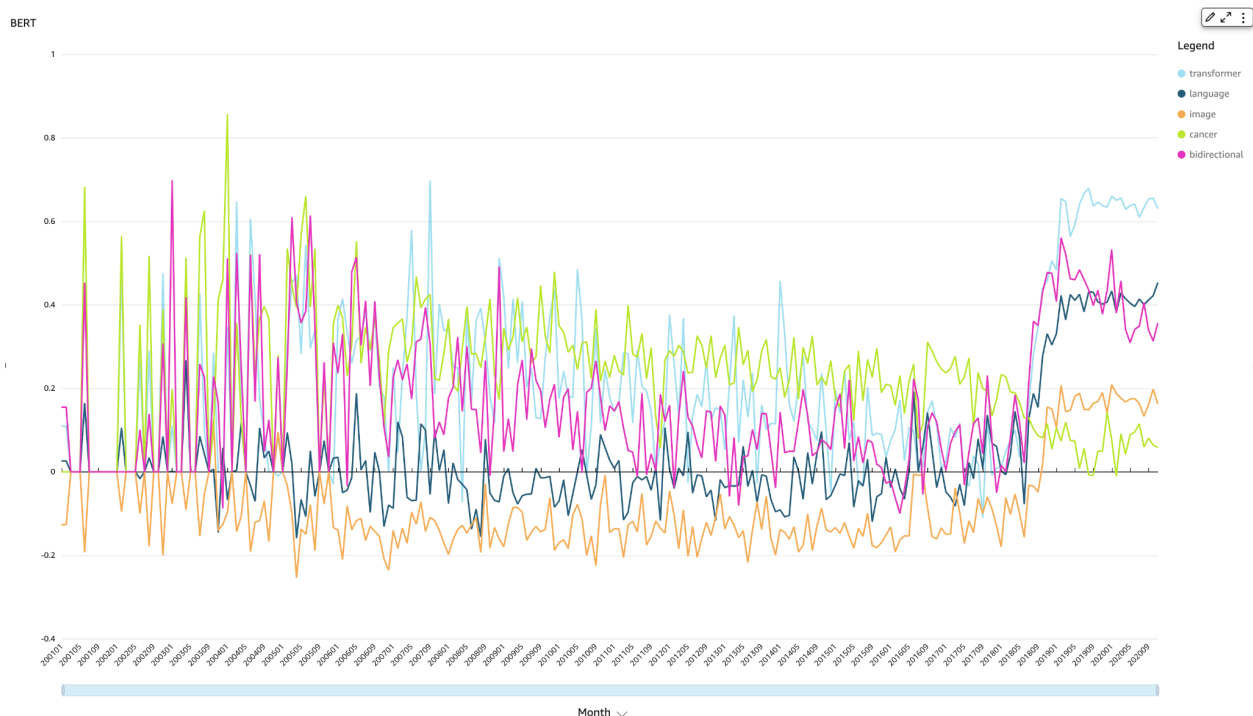


Figure 37: Time-series correlation between the entity “bert” and the entities “transformer”, “language”, “image”, “cancer” and “bidirectional”

Upon examining the graph in Figure 38, it is evident that a disruption in the correlation between the entities “Higgs” and “ATLAS” prominently occurs in the year 2012. This period aligns with the publication of the paper titled "Observation of a new particle in the search for the Standard Model Higgs boson with the ATLAS detector at the LHC". Authored by the ATLAS Collaboration, this paper delineates the analysis and results derived from proton-proton collision data, suggesting that the newly discovered particle aligns with the characteristics of the Higgs Boson at specific energies. As of the time of writing, the paper has obtained 12.5K citations, reflecting its significant impact on the scientific community. The time series analysis serves as a mirror of this disruption in the vector space continuum, effectively capturing the inception and continuation of the newly created correlation. This observation strongly confirms the underlying hypothesis that fluctuations within the vector space continuum can serve as a reliable metric for quantifying the emergence of novel events and discoveries.

ATLAS (A Toroidal LHC Apparatus) and CMS (Compact Muon Solenoid) are two of the four main particle detectors at the Large Hadron Collider (LHC) at CERN, the European Organization for Nuclear Research. They are used to study the properties of subatomic particles, such as protons and electrons, and to search for new particles and phenomena. Higgs boson was first proposed in 1964 by physicist Peter Higgs and

five other scientists as a way to explain why some particles have mass. The discovery of the Higgs boson was a major breakthrough in our understanding of the universe and was considered the final missing piece of the Standard Model of particle physics, which describes the behaviour of all known subatomic particles and their interactions.

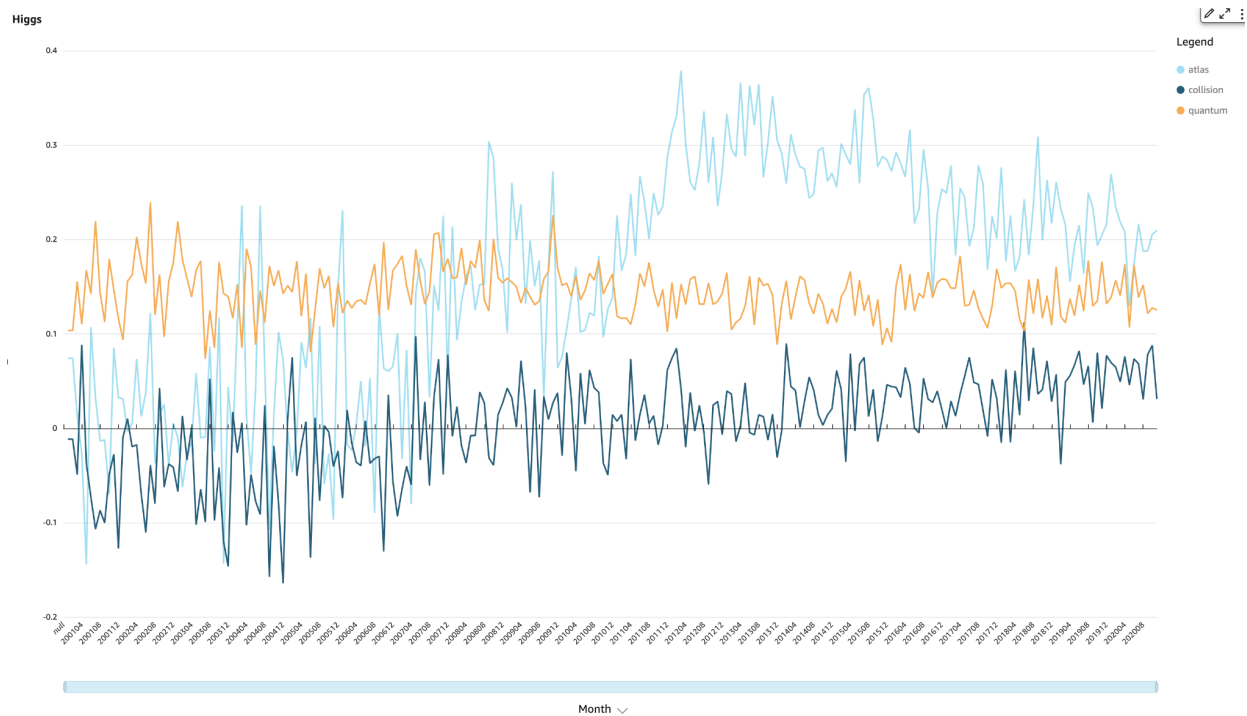


Figure 38: Time-series correlation between the entity “higgs” and the entities “atlas”, “collision” and “quantum”

Conventionally, [CNNs](#) are widely recognized for their prowess in image classification. However, in 2015, the landscape shifted with the publication of "Effective Use of Word Order for Text Categorization with Convolutional Neural Networks" by Johnson et al. ([2014](#)). This ground-breaking research, having garnered over a thousand citations to date, introduced a novel application of CNNs for text categorization. They innovatively applied [CNNs](#) directly to high-dimensional text data, thereby facilitating the learning of embeddings for small text regions instrumental in classification tasks. Observing Figure 39, the time-series analysis reveals a distinctive disruption between the entities “cnn” and “document” around 2015. This disruption precedes a noticeable increase in activity, leading to a stronger correlation between these entities, which emphasises the significant discovery made that year.

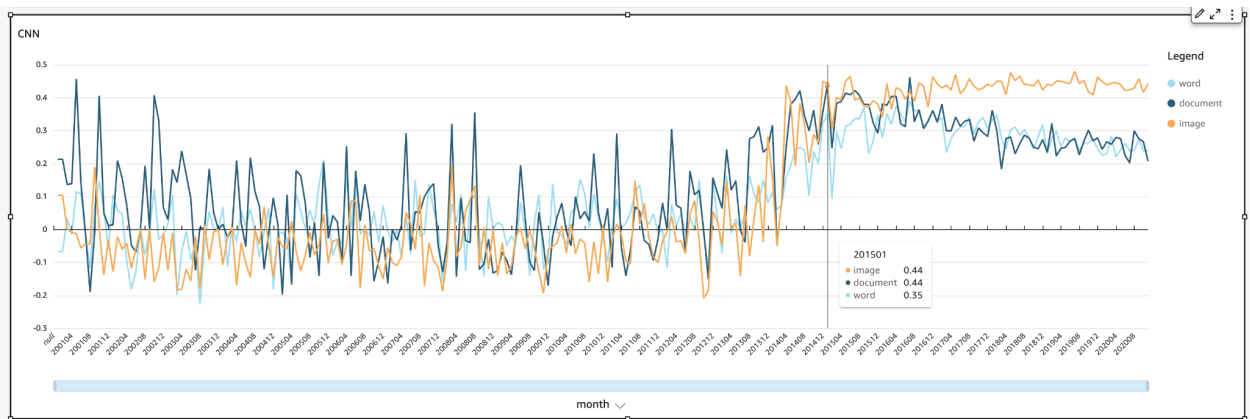


Figure 39: Time-series correlation between the entity “cnn” and the entities “word”, “document” and “image”

Analysis of trending events

In this research, the term 'trending event' signifies a progression or transformation occurring over an extended duration. This movement may capture the orientation of the scientific community, either leaning towards or diverging from a specific subject or approach. Such trends may be sparked by the introduction of novel technology or methodologies, or conversely, they may be triggered by the gradual decline of a formerly prevalent topic within the community.

To validate the system's capability of identifying trending events, we have pre-selected entities typically recognized for their upward or downward trajectories. Subsequently, a graphical representation of their time series analysis has been depicted to verify that the system can accurately reflect the anticipated scientific trends. In the following section, we will concentrate on a selection of specific entities under investigation, exhibiting either positive or negative trend patterns.

Geoffrey Hinton, often referred to as the "Godfather of Deep Learning," is a highly respected cognitive psychologist and computer scientist, acclaimed for his pioneering work in [ANN](#). Figure 40 illustrates the relationship between the entity labelled as “Hinton” and various other entities. Our analysis begins from a reference point, where the relationship between the entities “acoustic” and “Hinton” is explored. It follows a curve type termed “Timeless Tides,” which suggests there isn't a significant relationship or trend.

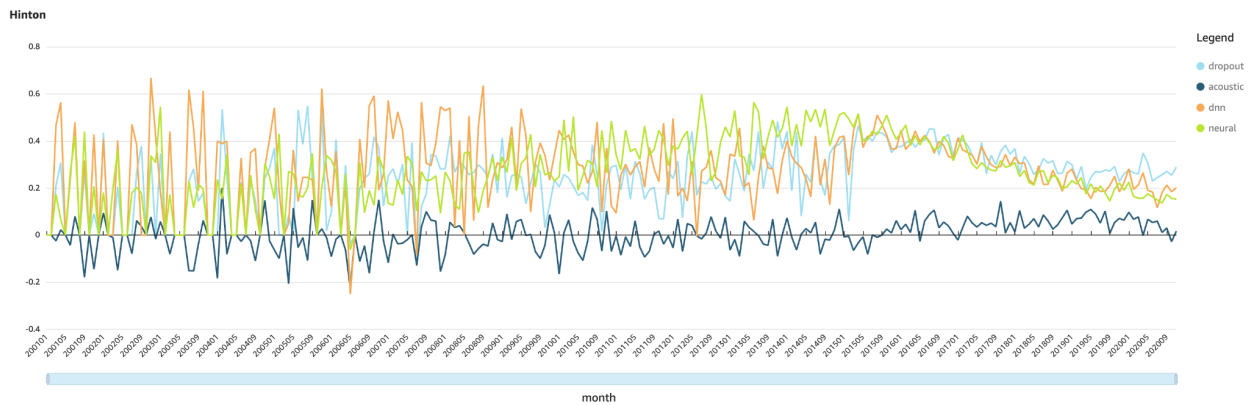


Figure 40: Time-series correlation between the entity “hinton” and the entities “dnn”, “neural”, “dropout” and “acoustic”

However, when examining the relationships between “Hinton” and “dropout”, [“dnn”](#), and “neural”, distinct differences emerge. A notable shift occurred around the year 2015, following the publication of the influential paper titled “Deep Learning” by LeCun, et., al ([2015](#)) which has been cited over 67K times at the time of this writing. This shift is also captured in the [“dnn”](#) and “neural” graphs. The observed increase in variance prior to 2015 possibly indicates a lower frequency of mentioning the entities across the timeline, but there appears to be a clear trend. The neural network struggles to determine if the entities are closely related, given the sporadic occurrence. of both entities together. The primary aim of these graphs, however, is to highlight the apparent downward trend, suggesting a diminished interest in “hinton” within the scientific community. This trend is corroborated when comparing the temporal progression of the graphs of [“dnn”](#), “dropout”, and “neural”.

Visualising these trends equips researchers with data to complement their intuitions. Further analysis of these graphs can reveal additional shifts or significant events, such as the change in the trend of the entity “neural” after 2006 (Hinton et al., [2006](#)). This coincides with the publication of Hinton's paper titled Reducing the Dimensionality of Data with Neural Networks, cited over 20K times at the time of this writing.

Figure 41 outlines the temporal progression of the entity "translate" in comparison with other entities, namely "crystal", "transformer", and "rnn". A clear upward trend, characterized by a consistent and significant increase in values or frequency, is immediately noticeable in the trajectory of the entity "transformer" around the year 2017, a pattern that aligns with the publication of the influential paper "Attention is All You Need" by Vaswani et al. ([2017](#)). This paper introduced the

Transformers model, an innovative architecture for sequence transduction tasks such as machine translation, which relies on attention mechanisms. The authors reported that their model surpasses the performance of existing models.

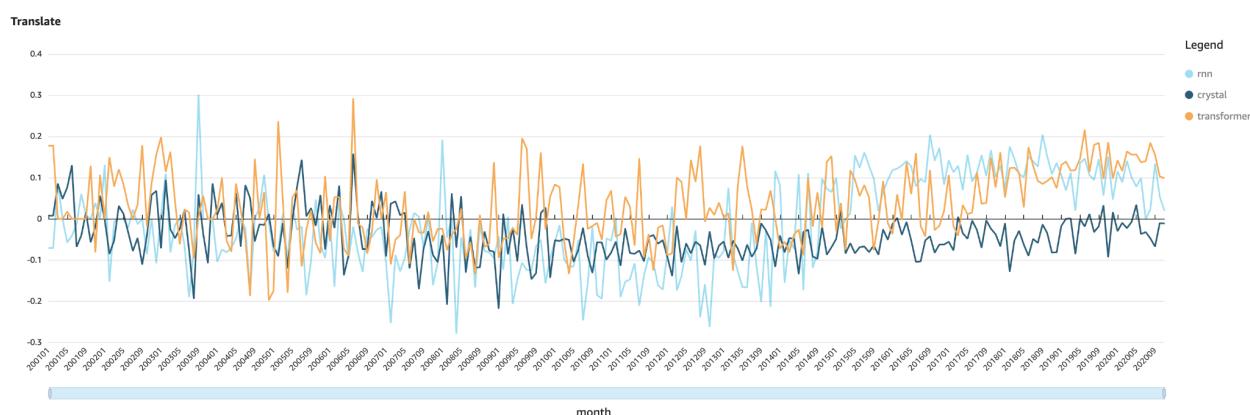


Figure 41: Time-series correlation between the entity “translate” and the entities “crystal”, “transformer” and “rnn”

The graph depicting the "rnn" entity reveals a noticeable upward trend starting from 2014, characterized by a consistent and significant increase in values or frequency, coinciding with the publication of the paper "Learning Phrase Representations using [RNN](#) Encoder–Decoder for Statistical Machine Translation" (Cho et al., [2014](#)). This paper proposed the use of two [RNNs](#) to encode a sequence of symbols into a fixed-length vector representation, which is subsequently decoded into another sequence of symbols for translation tasks. Interestingly, the popularity of "rnn" appears to decline in parallel with the rise of the use of transformers in translation, suggesting a shift in preference within the scientific community. Finally, the entity "crystal" serves as a reference point, predicated on the underlying hypothesis that it is relatively unrelated to the “translate” entity. Figure 41 confirms this assumption, illustrating a "Timeless tides" graph type, which indicates the absence of a significant relationship between the two entities.

Figure 42 illustrates a trending curve exemplifying the entity "linguistics." Upon careful observation of the correlation between this entity and "research," a distinct downward trend is evident, characterised as the "PhantomFade" curve type. This trend aligns with the prevailing consensus that linguistics is experiencing reduced traction due to the growing prominence of transformers models and large language models. A comparison with the curve representing the entity "semantic" reveals a lesser degree of correlation. Furthermore, the curve representing the entity "[RNN](#)" demonstrates a significant decline in correlation. Finally, we utilised the entity "boltzmann" as a

baseline to illustrate an insignificant relation, represented by the curve type "Timeless tides". From an academic perspective, these findings provide valuable insights into the changing dynamics within the field of linguistics and its interplay with other entities.

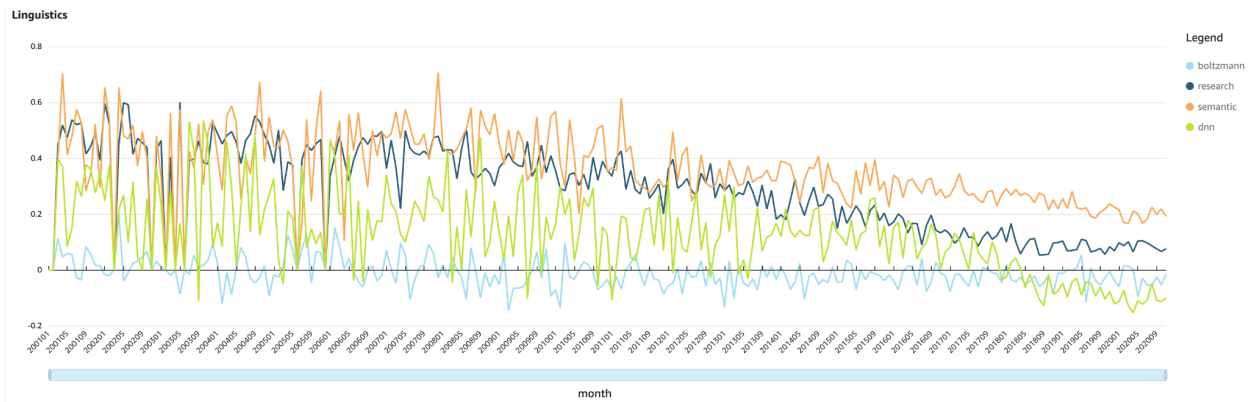


Figure 42: Time-series correlation between the entity “linguistics” and the entities “boltzmann”, “research”, “semantic” and “dnn”

6.4.3. Analysis of Temporal Knowledge Evolution Performance

Temporal Knowledge Evolution is a cutting-edge research domain focused on understanding how knowledge and relationships between entities evolve over time. By examining these temporal elements of the evolution of the relationships, we can uncover significant scientific discoveries, shifts in research trends, and patterns in technological advancements. This novel experiment aims to investigate these evolutionary processes by analysing the temporal relationships between different entities, providing insights into how knowledge develops and transforms within a specific domain. To obtain a holistic view of the performance of the temporal knowledge evolution framework, we designed an assessment method to evaluate its performance. Given the pioneering nature of this experiment, we cannot compare our results with state-of-the-art findings, as no similar studies have been conducted to date. Consequently, our assessment framework serves as a foundational benchmark, paving the way for future research in temporal knowledge evolution. We designed the assessment framework in a way to allow us to quantify the results for the temporal knowledge evolution. We measured 21 different relationships among various entities and collected the distance between the expected and observed data and trend. For each relationship, we identified a major event in the scientific field, for example, the publication of [BERT](#) in October 2018. This data point is considered expected data for the event. The temporal knowledge evolution experiment achieves its target by being able to detect an event approximately around the date of the paper’s publication. To

identify an event, we used the number of citations of a specific publication as an indication of its impact on the community and hence considered it a disruptive event.

The data collection involved 21 relationships between various entities. For each pair, we studied two main factors: the event date and the event trend. Following our categorization of the observed trends listed previously—“EternalElevation,” “PhantomFade,” “PhoenixRise,” “PhoenixFall,” and “TimelessTide”—we categorized each pair’s trends as observed in the data and as expected from the literature. For example, “bert” and “transformer” are expected to follow the trend of “PhoenixRise,” and the observed data trend is aligned with the expected trend. The event trend relationships were coded as “EternalElevation:1,” “PhantomFade:2,” “PhoenixRise:3,” “PhoenixFall:4,” and “TimelessTide:5.” The trend relation error metric was calculated based on the difference between the expected and observed trend relationships, with zero relationship distance indicating better conformity and any other value indicating non-conformity. For the event date, we recorded the expected date of the event from literature, supported by citation counts indicating the impact on the scientific community. We recorded the observed event date based on an anomaly detection method that will be explained later. *Table 7* provides example data for the entities and relationships and the example values between expected and observed event date and trend. The dataset included columns for Entity 1 and Entity 2 (pairs of entities being analysed), Expected relationship trend category, observed relationship trend category, error distance, event expected date, event observed date, and event date error metric.

Table 7 Example of data collections for the relationships among entities

Entity 1	Entity 2	Expected Relationship Trend Category	Observed Relationship Trend Category	Expected to Observed Trend Distance (Relationship Error Metric) (small is better)	Expected Event Date from Literature	Observed Date by Anomaly Detection	Expected to Observed Date Error Metric (0: within 3 months, 1: within 6 month, 2: within a year, 4: not detected)
Bert	Transformer	3	3	0	201810	201810	0
Language	Bert	3	3	0	201810	201810	0
Hinton	Dropout	2	2	0	201503	201504	0
Neural	Hinton	2	2	0	201503	201504	0
Linguistics	DNN	2	2	0	201503	201512	2

In this assessment we relied on anomaly detection to identify significant events such as the evolving relationship between “bert” and “transformer,” we selected the Random Cut Forest (RCF) method. This decision was influenced by the numerous advantages RCF provides over other anomaly detection algorithms, as highlighted by recent comparative studies. RCF was chosen for its efficiency and scalability, which is critical for our research that involves continuous monitoring of dynamic relationships over time. RCF is known for its computational efficiency and minimal memory usage, making it highly suitable for large datasets. Additionally, RCF’s robustness to noise ensures that the detected anomalies are genuine significant events rather than random fluctuations or noise in the data. This reliability is essential for accurately identifying meaningful changes in the dataset. Moreover, RCF’s versatility allows it to be applied to various types of data, including high-dimensional and time-series data. This versatility aligns perfectly with our dataset, which involves complex relationships between multiple entities that evolve over time. A comprehensive study conducted by Freeman et al. ([2022](#)) evaluated twelve time-series anomaly detection algorithms, including RCF, using performance metrics such as the Area Under the Curve (AUC), windowed F-score, and the Numenta Anomaly Benchmark (NAB) scoring model. The study revealed that while no single algorithm is optimal for all domains, RCF consistently performed well across different scenarios due to its isolation-based approach that effectively distinguishes anomalies by leveraging random cuts in the data space. Similarly, another study by Bouman et al. ([2024](#)) assessed 14 unsupervised anomaly detection algorithms on 47 tabular datasets. The findings indicated that RCF, particularly in its extended form (EIF), demonstrated superior performance, proving its effectiveness in detecting both global and local anomalies without extensive hyperparameter tuning.

In our study, we applied RCF through the following steps. We began with data preparation, collecting and preprocessing time-series data on the relationships between entities such as “bert” and “transformer,” ensuring data normalization and noise reduction. Next, we trained the RCF model using historical data, constructing a forest of trees with random cuts to capture the underlying patterns and anomalies. The trained model was then deployed to detect the anomaly for the selected entities as chosen earlier in the experiment design, calculating anomaly scores for each new data

point. Anomaly scores were compared against a predefined threshold to flag significant deviations, such as the notable development in the relationship between “bert” and “transformer” around late 2018. Finally, detected anomalies were analysed and validated to confirm they represented genuine significant events, supported by cross-referencing with publication date as listed online and referenced in the previous results section.

Figure 43 provides a visual representation of the event prediction error metric between expected and observed relationships among different entity pairs, categorized into low, medium, and high levels. The objective is to measure and analyse these relationship error metrics over time to identify significant scientific discoveries and shifts in research trends. The chart analysis showed that low error (0) dominated the chart at 76.2%, indicating that most entity pairs had a small distance between expected and observed relationships. Medium error band (1) accounted for 19.0% of the entity pairs, suggesting moderate discrepancies. High error band (2, 3) was present in only 4.8% of the entity pairs, indicating a few significant deviations.

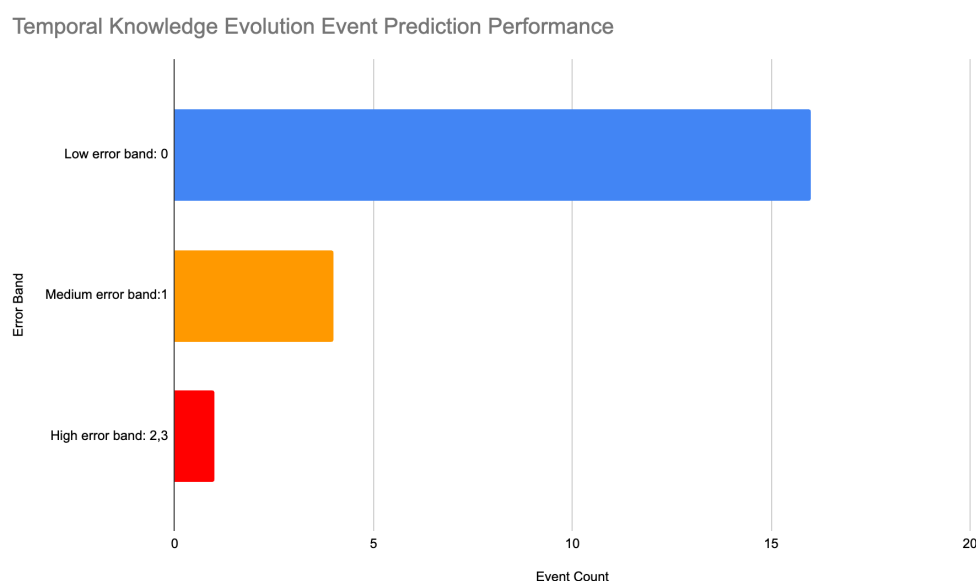


Figure 43 Temporal Knowledge Evolution Event Prediction Performance

A detailed breakdown of entity pairs and their distances revealed that low error band included pairs such as Bert-transformer, Bert-language, Bert-image, Bert-bidirectional, Higgs-atlas, CNN-word, Hinton-dropout, Hinton-dnn, Hinton-neural, Translate-rnn, Translate-transformer, Linguistics-research, and Linguistics-semantic. Medium error band included pairs like Higgs-quantum, CNN-document, Translate-rnn,

and Linguistics-dnn. High error band included only Linguistics-dnn with a error band of 2.

In conclusion, the majority of the entity pairs showed low error, suggesting a high level of accuracy in predicting observed relationship based on expected values. The relatively few medium and high errors indicate that the methodology used for the analysis is largely reliable. This approach allows researchers to quantify and support their findings by monitoring vector space continuums and identifying significant events and trends in scientific research.

6.5. Discussion

This research elucidates the utilisation of temporal vector space continua to encapsulate temporal variations in the relationships between different entities. The primary objective is to identify and highlight significant events and emergent trends within the corpus of literature under investigation. The primary challenge encountered during this study pertained to the data size and the computational capabilities necessary to execute the experiment, considering the extensive pre-processing and postprocessing steps, along with the training objectives themselves. In our approach, we utilised the full text of publications, a deviation from the method commonly adopted by researchers, who typically use the abstracts of scientific papers. While this augmented the demand for computational power, it simultaneously facilitated the generation of a high-quality vector space continuum representation capable of capturing significant events and discoveries, as documented in our findings.

A potential avenue for future research could involve the development of a real-time inference system. This would empower researchers across various fields to leverage the system and derive insights, further enabling them to investigate the detection mechanism for their entities of interest. We anticipate that providing such a platform to the community will equip the scientific realm with a tool for knowledge extraction and intuitive insight discovery. It should be noted that the main highlighted discoveries of the present study summarised in the results section were inherently constrained by the team's limited knowledge, emphasising the potential breadth of insights that could be derived from a more diverse set of perspectives.

6.6. Conclusion

In this chapter, we navigated the vast landscape of data-driven methodologies to shed light on the significance of underutilised crops in global agricultural strategies. We initiated our exploration by harnessing diverse datasets, with key findings stemming from sources such as arXiv, Wikipedia, and the [BG](#) datasets. These datasets were pivotal in setting the foundation for our research, with some serving as primary sources of insights, while others, like news categories and Amazon Customer Review datasets, were instrumental for preliminary experiments and hyperparameter optimisation tasks.

Transitioning from datasets, we delved into the methodologies employed throughout our research. The initial focus was on discerning cropping patterns of underutilised crops using global search engine data, aiming to understand public interest and potential cropping patterns, and the region where the crop is being cultivated. This approach was not only innovative but also reduced manual data collection efforts significantly. As we progressed, our research journey transitioned from traditional text mining techniques to the more advanced realm of [NIR](#). Leveraging the capabilities of word embeddings, we effectively analysed the large Wikipedia corpus, extracting valuable associations between crop names and countries, and discerning the relationship between crops and their soil requirements. We started from the English Wikipedia corpus and even widened the scope to capture all 178 languages of Wikipedia.

A portion of our chapter was dedicated to the interaction between word embeddings and [DNNs](#) in the domain of text classification. We emphasised the profound impact of hyperparameter tuning on the quality of word embeddings and how this directly influenced the performance of downstream tasks. Our exploration was comprehensive, ranging from the creation of embeddings to their integration into various [DNN](#) architectures, such as simple [RNN](#), [CNN](#), and [LSTM](#).

As our research narrative unfolded, we expanded our scope beyond crop science to broader scientific domains. The pinnacle of our exploration was the construction of the temporal evolution of knowledge in scientific literature. Harnessing a vast dataset from the arXiv repository, we carefully crafted a 'vector space continuum' that captured the essence of knowledge evolution over time which provided a foundation for tracing the trajectory of specific terminologies and their evolution.

In summary, this chapter offers a holistic overview of our methods and approaches used in this research, exploring the journey from text mining, going through [NIR](#), and stopping by hyperparameter tuning, building data pipeline processing over public cloud and constructing temporal knowledge evolution to study them thoroughly. Through our carefully designed methodologies and exploration, we aim to bridge knowledge gaps and provide valuable insights into the ever-evolving landscape of knowledge extraction and evolution.

The study presents a detailed approach to constructing a representation of the temporal evolution of knowledge in scientific literature. By sourcing the large dataset from the arXiv repository, spanning two decades, we carefully processed and analysed the data to construct a 'vector space continuum' that captures the essence of knowledge progression over time. Considering the volume of the data, utilising the capacity of the public cloud proved to be highly beneficial. The establishment of this vector space continuum provides a foundation for researchers to trace the trajectory of specific terminologies and examine their evolution in relation to other terms. Despite the computational challenges posed by the vast corpus of data, a rigorous selection criterion was devised to curate the most relevant entities for analysis. The subsequent construction of temporal entity sequences and the calculation of their interrelationships offer insights into the dynamics of their semantic relatedness over time. The big data platform, which stores the pre-selected entity vectors, serves as a tool for studying the evolution of specific entities and detecting significant events in the field. In chapter 6, the details of the experimental design of the temporal knowledge evolution will be discussed.

The adoption of the neural network model, [word2vec](#), has gained momentum amongst researchers across diverse disciplines to generate vector spaces for their respective textual corpora. [Word2vec](#) employs unsupervised learning to create word embeddings that embody words as vectors within a high-dimensional space. This facilitates the creation of a vector space encapsulating semantic relationships between words and paving the way for complex textual data analyses. Its implementation spans across social media (Biggers et al., [2023](#); Gao et al., [2023](#)), political science (Aydin, [2023](#)), medical science (Chen et al., [2021](#)), finance (Jiang et al., [2018](#)), and biomedicine (Wang et al., [2016](#)). The versatility of [word2vec](#) in myriad fields and applications, including text classification (Kim, [2014](#)), information retrieval (Yeshambel

et al., [2023](#)), and natural language generation (Vaswani et al., [2017](#)), underscores its popularity in textual data vector space creation.

Our exploration led us to two distinct methodologies for constructing the vector space continuum: the Discrete Knowledge Evolution Construction Method and the Continuous Knowledge Evolution Construction Method. While both methods presented their unique advantages, the continuous method emerged as the more robust choice, offering a consistent representation of knowledge evolution. By employing techniques such as word embeddings and the continuous knowledge evolution construction method, we have been able to create an aligned vector space continuum that captures the changes in knowledge over time.

This research expounded two techniques for constructing a vector space continuum, specifically the "Discrete Knowledge Evolution Construction Method" and the "Continuous Knowledge Evolution Construction Method." The former entails the training of separate word embedding models for each time slice, yielding a set of vector spaces representative of different scientific literature periods. Conversely, the latter trains a continuous function mapping time to the corresponding vector space. Our experiments with the arXiv dataset indicated the superior performance of the Continuous Knowledge Evolution Construction method in relation to the Discrete Knowledge Evolution Construction method for our study objectives.

We employed a vector-based similarity metric based on the cosine of the angle between 5000 pre-selected entities over a span of 240 months, which resulted in a vector space dimension of 25,000,000 x 240. The relatedness calculator outcomes were saved in a database, with the 240 months and 25 million entity-to-entity relationships represented as columns and rows, respectively. For data analysis and anomaly detection, we utilised a cloud-based analytics intelligence tool to delineate several recognised scientific discoveries and undertake manual extraction and graphing. Our intent was to associate patterns of relations to specific scientific discoveries or significant shifts in researchers' behaviours associated with a particular method or technology by studying changes and anomalies in the time series. Our methodology to pinpoint notable scientific discoveries involved analysing the time series graph of a targeted entity and using multiple co-entities for comparison in relation to the target entity's similarity. This strategy enhances our understanding of vector space conformity and enables us to monitor changes and anomalies in the time

series to identify correlations to significant scientific discoveries or human behavioural shifts.

The findings substantiate the system's proficiency in encapsulating specific recognized events and accurately illustrating research trends. Such capabilities position our methodology as a potent tool for quantifying researchers' intuitions and substantiating their perspectives. The potential time-saving implications are substantial, which, in the broader research landscape, could translate into considerable efficiency gains. Moreover, the availability of an objective metric that can support subjective opinion offers immense value to the scientific community at large.

Chapter 7

Discussion and conclusion

This thesis investigates word embeddings, their applications, and their underlying complexities, specifically tailored around three pivotal sets of research questions. These questions not only revolve around the practical utility of word embeddings in domain-specific scenarios but also dive deep into the nuances of their technical aspects, including hyperparameter impacts, and their capability to shed light on temporal trends in knowledge domains. The following sections delve into these three main avenues of investigation, aiming to provide a comprehensive understanding of this study's scope, findings, and implications.

The research seeks to address key questions surrounding the application and effectiveness of word embeddings in information retrieval. Firstly, the aim is to evaluate the efficacy of using word embeddings in extracting domain-specific insights, especially in terms of understanding interchangeability and relatedness. Secondly, the exploration involves the feasibility of leveraging a domain-specific corpus to train a high-quality word embedding model. The domain-specific corpus includes the ECOCROP database, which was used to compare and validate findings from the Wikipedia corpus regarding crop characteristics. Additionally, big data resources from the Google search engine were employed to discern global interest in underutilised crops and identify their local names. Specific data on crop soil requirements for underutilised crops was also gathered to determine soil needs and other characteristics using word embeddings. Such a model would ideally be adept at addressing questions tailored to specific needs of crops, like crop soil requirements. The research also seeks to investigate the intricacies of word embedding hyperparameters. Specifically, this research centres around understanding how these hyperparameters influence the performance of various [DNN](#) architectures. When these hyperparameters are passed to the [DNN](#)'s embedding layer, how do they impact the network's efficacy on downstream tasks, such as classification? Additionally, there is an interest in exploring the implications of word embedding hyperparameters in the construction of a temporal knowledge evolution framework. Both questions aim to shed light on the pivotal role of hyperparameter tuning for word embeddings. For the last problem addressed in the research, the goal was to explore the efficacy of using

vector space continuum methodologies to conduct temporal analysis of word embeddings on a large corpus of data. The overarching objective was to determine if such an approach could be instrumental in studying the evolution of knowledge over time. Within the broader framework of temporal knowledge evolution, there was also an aim to understand the potential of this method in pinpointing critical scientific events. Additionally, there was interest in its capability to map out specific scientific trends and trajectories, offering a comprehensive view of the progression in a particular field or discipline.

In the context of our research questions focused on the application and effectiveness of word embeddings in information retrieval, the findings from our experiments in Chapter 4 provide substantial evidence. Our primary inquiry evaluated the capability of word embeddings in extracting domain-specific insights, particularly concerning understanding interchangeability and relatedness. The results from using the [word2vec](#) model trained on the vast English Wikipedia dataset successfully highlighted its potential. This model adeptly identified various crop attributes such as crop vernacular and scientific names and global crop dynamics, which further underscores the efficacy of word embeddings in gleaning domain-specific knowledge. Moreover, the subsequent phase of our study explored the feasibility of harnessing a domain-rich corpus for training an optimised word embedding model. The extended scope of our experiment, which incorporated Wikipedia datasets across 178 languages, was demonstrative of this feasibility. The precision with which the model predicted soil requirements for a myriad of crops, further validated against the established ECOCROP (Ramirez-Villegas et al., [2013](#)) soil classification database, underscored its efficacy. Achieving an accuracy rate of 76.11% is reflective of the model's capability to address specific questions tailored to crops. The emphasis on the careful selection of terminology and its resultant impact on prediction accuracy suggests a deeper potential. By fine-tuning the word embedding hyperparameters with a more specialised corpus, it's plausible to expect even greater alignment with domain-centric needs. In synthesising these results, it is evident that our investigations align coherently with our initial research queries. The outcomes not only validate the potential of word embeddings in domain-specific information retrieval but also illuminate the path for temporal knowledge evolution and that allows scientific event detection and scientific trend mapping.

This study demonstrates the application of [NIR](#) methods, particularly leveraging word embeddings like [word2vec](#), to extract domain-specific knowledge from large corpora such as the English Wikipedia. This approach has proven effective in identifying relationships between underutilised crops and various attributes, achieving a precision rate that underscores its utility. Compared to state-of-the-art methods, such as those discussed by Zhang et al. (2016) and Radford et al. (2019), which highlight the challenges of context understanding and domain specificity in traditional information retrieval systems, our approach stands out by addressing these limitations. Specifically, the use of [NIR](#) enables a more nuanced extraction of both explicit and implicit knowledge, thus surpassing conventional methods in handling domain-specific queries and providing more comprehensive insights into agricultural data.

This research and the study by Jatnika et al. (2019) both utilize the [word2vec](#) model to explore semantic relationships within large text corpora. Jatnika et al. focused on calculating word similarities using [word2vec](#), trained on a large corpus of English Wikipedia articles, employing both [CBOW](#) and Skip-Gram architectures, and testing various window sizes and vector dimensions to determine optimal configurations for general semantic similarity tasks. Their study achieved 66.5% for WordSim-353 and 28.4% for SimLex-999, indicating moderate to weak correlations. Similarly, our study leveraged the [word2vec](#) model, but with a specialized focus on agricultural data. We utilised a multilingual dataset derived from Wikipedia and the ECOCROP database to target crop soil requirements. Techniques such as tokenization, lemmatization, and hyperparameter tuning were applied to optimize the model's performance. Our approach resulted in a 76.11% accuracy in predicting soil classifications using scientific crop names, which is significantly higher than the 49.97% accuracy achieved with vernacular names. The domain-specific application of our research enhances its practical utility and relevance, directly comparing model predictions with the ECOCROP database to provide actionable insights for agricultural practices. While both studies effectively demonstrate the capabilities of [word2vec](#) in semantic analysis, our research sets a new benchmark in predicting crop soil requirements, showcasing the robustness and applicability of [word2vec](#) in a specialized domain.

In our exploration of word embedding hyperparameters and their intricacies, the research discerned significant implications on the performance of various [DNN](#) architectures as indicated in Chapter 5. Central to our investigation was understanding the nexus between these hyperparameters and the efficacy of [DNNs](#) in tasks such as

text classification when the embeddings were integrated into their layers. Our findings corroborated that [DNN](#) architectures, namely [CNNs](#), [LSTMs](#), and [RNNs](#), exhibited varying degrees of sensitivity and adaptability to the hyperparameter configurations of the [word2vec](#) model. For instance, [CNNs](#) manifested marked fluctuations in their performance, potentially achieving stellar outcomes in certain configurations. Contrastingly, [LSTMs](#) displayed commendable steadiness and resilience to these hyperparameter shifts, implying their robustness in various conditions. Simple [RNNs](#), in contrast, demonstrated a more constrained performance spectrum, particularly when exposed to diverse hyperparameter setups. As the research ventured deeper into the role of hyperparameters within the embedding layers of [DNNs](#), it became evident that optimal configurations are intrinsically linked to better outcomes in text classification. The selection of the Skip-Gram model for [word2vec](#) and the optimal embedding dimensionality, for instance, were pivotal in enhancing the classification capability of the [DNN](#) architectures. Understanding the impact of hyperparameters on word embeddings is vital for establishing a framework for temporal knowledge evolution. This comprehension aids in producing high-quality word embeddings essential for constructing the continuum vector space, the cornerstone of the temporal knowledge evolution framework. Collectively, the research underscores the indispensable role of careful selection of word embedding hyperparameters, emphasising its direct bearing on the efficiency and versatility of [DNN](#) models and the temporal knowledge evolution framework.

(Mikolov et al. [2013a](#)), (Goldberg et al., [2014](#)), (Le et al., [2014](#)) and (Kim, [2014](#)) recommended specific set of hyperparameters for [word2vec](#) when used for downstream tasks. Our results confirm that smaller embedding dimensions negatively impact performance across all models, with significant improvements observed from dimensions of 44 onwards, especially for [LSTMs](#), which handle larger dimensions more effectively than [RNNs](#) and [CNNs](#). The window size parameter, balancing local and broader context capture, is commonly set between 5 to 10 in literature. Our experiments show that [LSTM](#) performance improves with larger window sizes, particularly beyond 12, indicating that [LSTMs](#) benefit from a broader context. In contrast, [RNNs](#) and [CNNs](#) do not show the same level of improvement with increased window size, suggesting they capture context differently. Negative sampling rate, ranging from 5 to 20 in literature, enhances quality with more samples but increases computational cost. Our findings indicate that [LSTMs](#) perform better at lower negative

sampling values, while [RNNs](#) also benefit from reduced values. [CNNs](#), however, tend to show less sensitivity to this parameter, highlighting the importance of optimizing it for [LSTMs](#) and [RNNs](#) specifically. Regarding the number of epochs, literature suggests that more epochs generally improve performance but risk overfitting. Our results show that very low epoch counts harm all architectures, while excessively high counts also have adverse effects, particularly on [CNNs](#). [LSTMs](#) and [RNNs](#) achieve optimal performance with moderate epoch count, emphasizing the need for careful epoch selection to balance performance and overfitting risk. When comparing Skip-gram and Bag of Words (BoW), literature typically prefers Skip-gram for effectively capturing word context. Our results indicate that [RNNs](#) perform better with Skip-gram, while [CNNs](#) excel with BoW. [LSTMs](#) maintain consistent performance regardless of the architecture, demonstrating their robustness across different context-capturing methods.

In our pursuit to gauge the utility of vector space continuum methodologies for temporal analysis as in Chapter 6, this study uncovered intriguing patterns associated with significant scientific events. The term 'disruptive event' describes incidents leading to substantial changes in the body of scientific literature. By probing into known entities with evident historical scientific breakthroughs, and contrasting them against random references, the aim was to determine whether the trends uncovered were truly anomalous. One salient observation was the relationship between the entity "bert" and its associated entities. The pronounced disturbance in the vector space around October 2018 underscores the transformative introduction of [BERT](#). The profound connections between "bert", "bidirectional", and "transformer" coincide with the revolutionary achievements of [BERT](#) in the realm of [NLP](#). Furthermore, the shift in the relationship between "bert" and "image" post-2019 underscores the innovative adaptation of [BERT](#) to image processing tasks, as evidenced by the [ViLBERT](#) paper (Lu et al., [2019](#)). The exploration of the "higgs" entity paints a similar story. The marked disturbance between "Higgs" and "ATLAS" in 2012 is a reflection to the momentous discovery of a particle in alignment with the Higgs Boson's properties, a cornerstone in understanding the universe's fabric. This observation substantiates our hypothesis that disruptions in the vector space are potent markers for significant scientific milestones. Lastly, the dynamics between "cnn" and the entities "document", "word", and "image" are revelatory. The pronounced shift around 2015 marks a paradigm shift in the application of [CNNs](#), as they transitioned from being predominantly image-

based tools to formidable assets in text categorization. To encapsulate, our findings corroborate the potential of vector space continuum methodologies to effectively chronicle the temporal evolution of knowledge. By highlighting perturbations in the vector space, it's possible not only trace the genesis of ground-breaking discoveries but also anticipate the trajectories they may forge, providing a panoramic perspective of a discipline's evolution.

Our exploration into 'trending events' aims to decode the evolution of interest and focus regarding the scientific arena over extended periods as in Chapter [6](#). These events epitomise the ups and downs of attention towards specific subjects. A closer look at Geoffrey Hinton, a luminary in deep learning, presents a fascinating timeline. While Hinton's relationship with acoustics doesn't exhibit significant trends, his association with dropout, [DNN](#), and [ANN](#), does, following his influential work and contributions. Yet, a key observation is the continuous decline in attention to "Hinton" after 2015, which highlights the temporal nature of scientific trajectories and trends. The theme of translation brought forth interesting patterns, especially surrounding "transformer" and "[RNN](#)". The rise of "transformer" post-2017 is a testament to the ground-breaking Transformer model discussed in "Attention is All You Need" (Vaswani et al., [2017](#)). On the other hand, the gradual retreat of "[RNN](#)" amidst the transformer's ascent underlines the collective preferences in scientific methodologies. Our final deep dive into "linguistics" offers some captivating trends. The tapering alignment of linguistics with "research" hints at a broader shift, possibly triggered by the rise of transformer-based models and large language architectures. This shift, marked by the diminishing association with "semantic" and "dnn", reflects the ever-evolving priorities in linguistic studies today. In essence, our analysis paints a picture of the dynamic, ever-shifting terrain of scientific emphasis. In essence, the evidence generated through our experiments confirms that the vector space continuum methodologies are not only viable but also particularly potent in tracing the evolution of knowledge over periods and spotlighting critical moments of scientific breakthrough or shift.

This study on Temporal Knowledge Evolution offers a novel perspective on how knowledge and relationships between entities change over time. This innovative approach, which measures and compares expected versus observed correlations, sets our research apart since no comparable studies exist for direct comparison with current state-of-the-art methods. Analysing 21 different entity relationships, we found that 76.2% showed low error band, indicating a high level of predictive accuracy.

Medium error band was observed in 19.0% of the pairs, while only 4.8% exhibited high variance, demonstrating the robustness of our methodology. The predominance of low variance highlights the reliability of our method in tracking and understanding the dynamic evolution of knowledge. This study provides a foundational benchmark, offering a novel framework for future research in temporal knowledge evolution and aiding in the identification of significant scientific discoveries and shifts in research trends.

The research findings on word embeddings provide valuable insights for informed decision-making across multiple domains, especially through temporal knowledge evolution. This study highlights the capability of word embeddings to extract domain-specific knowledge and track the evolution of key terms and concepts over time. In agriculture, a model trained on a domain-rich corpus successfully predicted soil requirements for crops, demonstrating practical applications in resource management and agricultural planning. Beyond agriculture, these methodologies can be transformative in fields such as pharmaceuticals and politics. In the pharmaceutical industry, temporal knowledge evolution can be leveraged to predict emerging drug trends and identify potential breakthrough treatments. By analysing the evolving relationships between terms related to diseases, treatments, and drug efficacy, researchers can forecast shifts in medical research focus and prioritize the development of promising new medications. Similarly, in the political domain, temporal knowledge evolution can help predict significant political events and shifts in public sentiment. By examining changes in the vector space related to political figures, policies, and key issues, analysts can detect early indicators of emerging trends, such as the rise of new political movements or changes in voter priorities. For example, tracking the evolution of terms associated with climate change legislation can help policymakers anticipate future public support or opposition, allowing them to adjust their strategies accordingly. These applications illustrate the robust potential of temporal knowledge evolution in various fields, providing a scientific framework for strategic planning and decision-making based on the dynamic analysis of evolving information.

Drawing from our exploration into word embeddings, their practical applications, and associated intricacies, several key conclusions can be gleaned. Our research reaffirmed the utility of word embeddings, specifically in extracting domain-specific explicit and implicit knowledge. The accuracy achieved using specialised

datasets underscores the significance of choosing an appropriate training corpus for optimal results. Moreover, the intricate combinations of word embedding hyperparameters has emerged as a focal point, influencing the performance of various [DNN](#) architectures. Our findings reinforce the importance of prudently selecting hyperparameters, given their direct influence on the efficacy of models, especially in tasks like text classification and temporal knowledge evolution. Additionally, the potent capability of vector space continuum methodologies in chronicling the temporal evolution of scientific knowledge was discovered. This method proved instrumental in pinpointing significant scientific events, unveiling patterns that serve as markers for transformative milestones in scientific literature. Notably, our research on trending events offers a dynamic snapshot of scientific focus over time, exemplifying the fluid nature of scientific explorations and the ensuing shifts in research emphasis. In totality, our study not only consolidates the critical role of word embeddings in domain-specific scenarios but also emphasises the vast potential of vector space continuum methodologies in comprehending and predicting the temporal evolution of scientific knowledge.

This research delves into [NIR](#), hyperparameter optimization, and temporal knowledge tracking using a vector space continuum. However, inherent limitations shape its trajectory. While the study on multilingual crop names focuses on select languages, future research could broaden this scope, tapping into more diverse linguistic datasets. The methods applied are tailored for crop science, suggesting the need for further studies that test these methodologies across different domains. The text mining emphasis on specific crops datasets raises the possibility of integrating other data types in subsequent investigations. Our focus on interchangeability and relatedness analogies leaves the door open for exploring alternative relationships in future analyses. The concentration on a subset of scientific literature means other influential repositories could be spotlighted in later studies. Our knowledge classification, tailored for temporal knowledge evolution, could be adapted and expanded for other knowledge domains. The dataset generated for studying temporal knowledge evolution is already substantial in size. However, due to constraints in human and computational resources in this research, Chapter [6](#) experiments examined only a subset of this dataset. Future research aims to expand this investigation to encompass a larger set of entities and other scientific domains. Subsequent studies will explore the application of Large Language Models, using them

to recreate the temporal knowledge evolution framework. This will allow for an examination of its impact on the quality of scientific event detection and the mapping of scientific trends trajectories.

References

- Adewumi, T. P., Liwicki, F., & Liwicki, M. (2020). Word2vec: Optimal hyperparameters and their impact on nlp downstream tasks. arXiv preprint arXiv:2003.11645.
- Alfarizi, M. I., Syafaah, L., & Lestandy, M. (2022). Emotional Text Classification Using TF-IDF (Term Frequency-Inverse Document Frequency) And LSTM (Long Short-Term Memory). *JUITA: Jurnal Informatika*, 10(2), 225-232.
- Alotaibi, F. M. (2023). A Machine-Learning-Inspired Opinion Extraction Mechanism for Classifying Customer Reviews on Social Media. *Applied Sciences*, 13(12), 7266.
- Arora, V. S., McKee, M., & Stuckler, D. (2019). Google Trends: Opportunities and limitations in health and health policy research. *Health Policy*, 123(3), 338-341.
- AYDIN, Y. (2023). Sentiment Analyzing from Tweet Data's Using Bag of Words and Word2Vec. *Bitlis Eren Üniversitesi Fen Bilimleri Dergisi*, 12(2), 412-417.
- Baltrunas, L., Ludwig, B., & Ricci, F. (2011, October). Matrix factorization techniques for context aware recommendation. In *Proceedings of the fifth ACM conference on Recommender systems* (pp. 301-304).
- Bamler, R., & Mandt, S. (2017, July). Dynamic word embeddings. In *International conference on Machine learning* (pp. 380-389). PMLR.
- Baradad, V. P., & Mugabushaka, A. M. (2015, June). Corpus Specific Stop Words to Improve the Textual Analysis in Scientometrics. In *ISSI*.
- Bardak, Batuhan, and Mehmet Tan. "Prediction of influenza outbreaks by integrating Wikipedia article access logs and Google flu trend data." In *Bioinformatics and Bioengineering (BIBE), 2015 IEEE 15th International Conference on*, pp. 1-6. IEEE, 2015.
- Batista-Navarro, R., Zerva, C., & Ananiadou, S. (2016, September). Construction of a Biodiversity Knowledge Repository using a Text Mining-based Framework. In *SIMBig* (pp. 22-25).
- Benedetto, L. (2023). A quantitative study of NLP approaches to question difficulty estimation. arXiv preprint arXiv:2305.10236.
- Berger, M., McDonough, K., & Seversky, L. M. (2016). cite2vec: Citation-driven document exploration via word embeddings. *IEEE transactions on visualization and computer graphics*, 23(1), 691-700.
- Biggers, F. B., Mohanty, S. D., & Manda, P. (2023). A deep semantic matching approach for identifying relevant messages for social media analysis. *Scientific Reports*, 13(1), 12005.
- Bojanowski, P., Grave, E., Joulin, A., & Mikolov, T. (2017). Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5, 135-146.

- Bouman, R., Bukhsh, Z., & Heskes, T. (2024). Unsupervised anomaly detection algorithms on real-world data: how many do we need?. *Journal of Machine Learning Research*, 25(105), 1-34.
- Brinn, S., Cameron, C., Fielding, D., Frankston, C., Fromme, A., Huang, P., ... & Zhou, Q. (2022). A framework for improving the accessibility of research papers on arXiv. org. arXiv preprint arXiv:2212.07286.
- Burns, A., Srinivasan, K., Ainslie, J., Brown, G., Plummer, B. A., Saenko, K., ... & Guo, M. (2023). WikiWeb2M: A Page-Level Multimodal Wikipedia Dataset. *arXiv preprint arXiv:2305.05432*.
- Camacho-Collados, J., & Pilehvar, M. T. (2017). On the role of text preprocessing in neural network architectures: An evaluation study on text categorization and sentiment analysis. arXiv preprint arXiv:1707.01780.
- Caracciolo, Caterina, Armando Stellato, Ahsan Morshed, Gudrun Johannsen, Sachit Rajbhandari, Yves Jaques, and Johannes Keizer. "The AGROVOC linked dataset." *Semantic Web* 4, no. 3 (2013): 341-348.
- Carneiro, H. A., & Mylonakis, E. (2009). Google trends: a web-based tool for real-time surveillance of disease outbreaks. *Clinical infectious diseases*, 49(10), 1557-1564.
- Caselles-Dupré, H., Lesaint, F., & Royo-Letelier, J. (2018, September). Word2vec applied to recommendation: Hyperparameters matter. In *Proceedings of the 12th ACM Conference on Recommender Systems* (pp. 352-356).
- Chamberlain, B. P., Rossi, E., Shiebler, D., Sedhain, S., & Bronstein, M. M. (2020, September). Tuning word2vec for large scale recommendation systems. In *Proceedings of the 14th ACM Conference on Recommender Systems* (pp. 732-737).
- Chelba, C., Mikolov, T., Schuster, M., Ge, Q., Brants, T., Koehn, P., & Robinson, T. (2013). One billion word benchmark for measuring progress in statistical language modeling. arXiv preprint arXiv:1312.3005.
- Chen, Q., & Sokolova, M. (2021). Specialists, scientists, and sentiments: Word2Vec and Doc2Vec in analysis of scientific and medical texts. *SN Computer Science*, 2, 1-11.
- Chivenge, Pauline, Tafadzwanashe Mabhaudhi, Albert T. Modi, and Paramu Mafongoya. "The potential role of neglected and underutilised crop species as future crops under water scarce conditions in SubSaharan Africa." *International journal of environmental research and public health* 12, no. 6 (2015): 5685-5711.
- Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., & Bengio, Y. (2014). Learning phrase representations using RNN encoder-decoder for statistical machine translation. arXiv preprint arXiv:1406.1078.
- Choi, Hyunyoung, and Hal Varian. "Predicting the present with Google Trends." *Economic Record* 88, no. s1 (2012): 2-9.

- Clement, C. B., Bierbaum, M., O'Keeffe, K. P., & Alemi, A. A. (2019). On the use of arxiv as a dataset. arXiv preprint arXiv:1905.00075.
- Conneau, A., Kiela, D., Schwenk, H., Barrault, L., & Bordes, A. (2017). Supervised learning of universal sentence representations from natural language inference data. arXiv preprint arXiv:1705.02364.
- Deng, S., Qin, J., Wang, X., & Wang, R. (2023). Attention Knowledge Network Combining Explicit and Implicit Information. *Mathematics*, 11(3), 724.
- Department of Agriculture, forestry & fisheries, Forestry and Fisheries Private Bag X144, Pretoria, 0001 South Africa. "Production guideline for Bambara groundnuts." (2016). http://www.dalrrd.gov.za/phocadownloadpap/Brochures_and_Production_Guidelines/Bambara.pdf
- Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805.
- Ding, W., Li, J., Luo, L., & Qu, Y. (2024, May). Enhancing complex question answering over knowledge graphs through evidence pattern retrieval. In *Proceedings of the ACM on Web Conference 2024* (pp. 2106-2115).
- Ebert, Andreas W. "Potential of underutilised traditional vegetables and legume crops to contribute to food and nutritional security, income and more sustainable production systems." *Sustainability* 6, no. 1 (2014): 319-335.
- Elgeldawi, E., Sayed, A., Galal, A. R., & Zaki, A. M. (2021, November). Hyperparameter tuning for machine learning algorithms used for arabic sentiment analysis. In *Informatics* (Vol. 8, No. 4, p. 79). MDPI.
- Ethayarajh, K. (2019). How contextual are contextualized word representations? Comparing the geometry of BERT, ELMo, and GPT-2 embeddings. *arXiv preprint arXiv:1909.00512*.
- Firth, J. (1957). A synopsis of linguistic theory, 1930-1955. *Studies in linguistic analysis*, 10-32.
- Freeman, C., Merriman, J., Beaver, I., & Mueen, A. (2021). Experimental comparison and survey of twelve time series anomaly detection algorithms. *Journal of Artificial Intelligence Research*, 72, 849-899.
- Gao, Y., Wang, J., Li, Z., & Peng, Z. (2023). The Social Media Big Data Analysis for Demand Forecasting in the Context of Globalization: Development and Case Implementation of Innovative Frameworks. *Journal of Organizational and End User Computing (JOEUC)*, 35(3), 1-15.
- Geddes, Brad. *Advanced Google AdWords*. John Wiley & Sons, 2014.
- Ginsparg, P. (2011). ArXiv at 20. *Nature*, 476(7359), 145-147.
- Goldberg, Y., & Levy, O. (2014). word2vec Explained: deriving Mikolov et al.'s negative-sampling wordembedding method. arXiv preprint arXiv:1402.3722.
- Gong, H., Bhat, S., & Viswanath, P. (2020). Enriching word embeddings with temporal and spatial information. *arXiv preprint arXiv:2010.00761*
- Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep learning*. MIT press.

- Gregory, P. J., Mayes, S., Hui, C. H., Jahanshiri, E., Julkifle, A., Kuppusamy, G., ... & Azam-Ali, S. N. (2019). Crops For the Future (CFF): an overview of research efforts in the adoption of underutilised species. *Planta*, 250, 979-988.
- Griggs, David, Mark Stafford-Smith, Owen Gaffney, Johan Rockström, Marcus C. Öhman, Priya Shyamsundar, Will Steffen, Gisbert Glaser, Norichika Kanie, and Ian Noble. "Policy: Sustainable development goals for people and planet." *Nature* 495, no. 7441 (2013): 305-307.
- Grover, A. & Leskovec, J. (2016, August). node2vec: Scalable feature learning for networks. In *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining* (pp. 855-864).
- Gupta, G., Rastegarpanah, B., Iyer, A., Rubin, J., & Kenthapadi, K. (2023). Measuring Distributional Shifts in Text: The Advantage of Language Model-Based Embeddings. *arXiv preprint arXiv:2312.02337*.
- Hajibabaei, P., Malekzadeh, M., Heidari, M., Zad, S., Uzuner, O., & Jones, J. H. (2021, October). An empirical study of the graphsage and word2vec algorithms for graph multiclass classification. In *2021 IEEE 12th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON)* (pp. 0515-0522). IEEE.
- Han, J. Song, Y. Zhao, W. X. Shi, S. & Zhang, H. (2018). hyperdoc2vec: Distributed representations of hypertext documents. *arXiv preprint arXiv:1805.03793*.
- Hasan, B. M. S., & Abdulazeez, A. M. (2021). A review of principal component analysis algorithm for dimensionality reduction. *Journal of Soft Computing and Data Mining*, 2(1), 20-30.
- Hinton, G. E., & Salakhutdinov, R. R. (2006). Reducing the dimensionality of data with neural networks. *science*, 313(5786), 504-507.
- Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural computation*, 9(8), 1735-1780.
- Holman Rector, L. (2008). Comparison of Wikipedia and other encyclopedias for accuracy, breadth, and depth in historical articles. *Reference services review*, 36(1), 7-22. http://cropbase.org/cropbase/tools/AI/data_eye.html,.
- Hua, B., Yangb, Y., Weic, J., & Wud, B. (2020). Research and Application of Malware Classification Method Based on LSTM. *Academic Journal of Computing & Information Science*, 5(8), 95-100.
- Internet live stats. "Google Search Statistics" [internetlivestats.com](http://www.internetlivestats.com). <http://www.internetlivestats.com/google-search-statistics> (accessed March 13, 2023).
- Jahanshiri, E., & Walker, S. (2015). Agricultural knowledge-based systems at the age of semantic technologies. *Inter. J. Know. Engin*, 1(1), 64-67. Jahanshiri, Ebrahim, and Sue Walker. "Agricultural Knowledge-Based Systems at the Age of Semantic Technologies." *International Journal of Knowledge Engineering-IACSIT* 1 (2015): 64-67.

- Jahanshiri, E., Walker, S., Ehsan, S. D., & Jahanshiri, E. (2014). Paper Reference Number: 47 Collaborative Agricultural Research Environment Based on Open Source Geospatial Web Technologies.
- Jatnika, D., Bijaksana, M. A., & Suryani, A. A. (2019). Word2vec model analysis for semantic similarities in english words. *Procedia Computer Science*, 157, 160-167.
- Ji, S. Yun, H. Yanardag, P. Matsushima, S. & Vishwanathan, S. V. N. (2015). Wordrank: Learning word embeddings via robust ranking. *arXiv preprint arXiv:1506.02761*.
- Jiang, J., & Li, J. (2018). Constructing financial sentimental factors in Chinese market using natural language processing. *arXiv preprint arXiv:1809.08390*.
- Johnson, R., & Zhang, T. (2014). Effective use of word order for text categorization with convolutional neural networks. *arXiv preprint arXiv:1412.1058*.
- Joulin, A. Grave, E. Bojanowski, P. Douze, M. Jégou, H. & Mikolov, T. (2016). Fasttext. zip: Compressing text classification models. *arXiv preprint arXiv:1612.03651*.
- Joulin, A., Grave, E., Bojanowski, P., & Mikolov, T. (2016b). Bag of tricks for efficient text classification. *arXiv preprint arXiv:1607.01759*.
- Khan, M. M. H., Rafii, M. Y., Ramlee, S. I., Jusoh, M., & Al-Mamun, M. (2021). Bambara groundnut (*Vigna subterranea* L. Verdc): A crop for the new millennium, its genetic diversity, and improvements to mitigate future food and nutritional challenges. *Sustainability*, 13(10), 5530.
- Khosa, S., Mehmood, A., & Rizwan, M. (2023). Unifying Sentence Transformer Embedding and Softmax Voting Ensemble for Accurate News Category Prediction. *Computers*, 12(7), 137.
- Kim, D., Yoon, S., Park, D., Lee, Y., Song, H., Bang, J., & Lee, J. G. (2023). One Size Fits All for Semantic Shifts: Adaptive Prompt Tuning for Continual Learning. *arXiv preprint arXiv:2311.12048*.
- Kim, Y. (2014). Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)* (pp. 1746-1751).
- Kiros, R., Zhu, Y., Salakhutdinov, R. R., Zemel, R., Urtasun, R., Torralba, A., & Fidler, S. (2015). Skip-thought vectors. *Advances in neural information processing systems*, 28.
- Kotiyal, B., Pathak, H., & Singh, N. (2023). Debunking multi-lingual social media posts using deep learning. *International Journal of Information Technology*, 1-13.
- Kurata, H., Tsukiyama, S., & Manavalan, B. (2022). iACVP: markedly enhanced identification of anti-coronavirus peptides using a dataset-specific word2vec model. *Briefings in Bioinformatics*, 23(4), bbac265.
- Kutuzov, A., Øvrelid, L., Szymanski, T., & Velldal, E. (2018). Diachronic word embeddings and semantic shifts: a survey. *arXiv preprint arXiv:1806.03537*.

- Le, Q. & Mikolov, T. (2014, June). Distributed representations of sentences and documents. In International conference on machine learning (pp. 1188-1196). PMLR.
- LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *nature*, 521(7553), 436-444.
- LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11), 2278-2324.
- Lee, J. (2022). The emotion is not one-hot encoding: Learning with grayscale label for emotion recognition in conversation. *arXiv preprint arXiv:2206.07359*.
- Levy, O., & Goldberg, Y. (2014). Neural word embedding as implicit matrix factorization. In *Advances in Neural Information Processing Systems* (pp. 2177-2185).
- Liao, S., Liang, S., Meng, Z., & Zhang, Q. (2021, March). Learning dynamic embeddings for temporal knowledge graphs. In *Proceedings of the 14th ACM International Conference on Web Search and Data Mining* (pp. 535-543).
- Lilleberg, J., Zhu, Y., & Zhang, Y. (2015, July). Support vector machines and word2vec for text classification with semantic features. In *Cognitive Informatics & Cognitive Computing (ICCI* CC), 2015 IEEE 14th International Conference on* (pp. 136-140). IEEE.
- Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., ... & Stoyanov, V. (2019). Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.
- Long, X., Zeng, J., Meng, F., Ma, Z., Zhang, K., Zhou, B., & Zhou, J. (2024). Generative Multi-Modal Knowledge Retrieval with Large Language Models. *arXiv preprint arXiv:2401.08206*.
- Lu, J., Batra, D., Parikh, D., & Lee, S. (2019). Vilbert: Pretraining task-agnostic visiolinguistic representations for vision-and-language tasks. *Advances in neural information processing systems*, 32.
- Luo, Y., Huang, K., Hong, M., Yang, K., Zhang, J., Wu, Y., & Nie, Z. (2023). Empowering ai drug discovery with explicit and implicit knowledge. *arXiv preprint arXiv:2305.01523*.
- Mandt, S. M., & Bamler, R. (2021). U.S. Patent No. 11,068,658. Washington, DC: U.S. Patent and Trademark Office.
- Mantovani, R. G., Rossi, A. L., Vanschoren, J., Bischl, B., & De Carvalho, A. C. (2015, July). Effectiveness of random search in SVM hyper-parameter tuning. In *2015 international joint conference on neural networks (IJCNN)* (pp. 1-8). IEEE.
- Maslova, N. & Potapov, V. (2017). Neural network doc2vec in automated sentiment analysis for short informal texts. In *Speech and Computer: 19th International Conference, SPECOM 2017, Hatfield, UK, September 12-16, 2017, Proceedings 19* (pp. 546-554). Springer International Publishing.

- Mayes, S., F. J. Massawe, P. G. Alderson, J. A. Roberts, S. N. AzamAli, and M. Hermann. "The potential for underutilised crops to improve security of food production." *Journal of experimental botany* 63, no. 3 (2012): 1075-1079.
- Mayes, S., Ho, W. K., Chai, H. H., Gao, X., Kundy, A. C., Mateva, K. I., ... & Azam-Ali, S. N. (2019). Bambara groundnut: an exemplar underutilised legume for resilience under climate change. *Planta*, 250, 803-820.
- Mesnil, G., Mikolov, T., Ranzato, M. A., & Bengio, Y. (2014). Ensemble of generative and discriminative techniques for sentiment analysis of movie reviews. *arXiv preprint arXiv:1412.5335*.
- Mikolov, T. Chen, K. Corrado, G. & Dean, J. (2013a). Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.
- Mikolov, T., Grave, E., Bojanowski, P., Puhersch, C., & Joulin, A. (2017). Advances in pre-training distributed word representations. *arXiv preprint arXiv:1712.09405*.
- Mikolov, T., Le, Q. V., & Sutskever, I. (2013b). Exploiting similarities among languages for machine translation. *arXiv preprint arXiv:1309.4168*.
- Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., & Dean, J. (2013c). Distributed representations of words and phrases and their compositionality. *Advances in neural information processing systems*, 26.
- Minaee, S., Kalchbrenner, N., Cambria, E., Nikzad, N., Chenaghlu, M., & Gao, J. (2021). Deep learning--based text classification: a comprehensive review. *ACM computing surveys (CSUR)*, 54(3), 1-40.
- Misra, R. (2022). News category dataset. *arXiv preprint arXiv:2209.11429*.
- Mitara, B., Craswell, N. (2017). an Introduction to Neural Information retrieval. *Foundations and Trends R in Information Retrieval*.
- Moody, C. E. (2016). Mixing dirichlet topic models and word embeddings to make lda2vec. *arXiv preprint arXiv:1605.02019*.
- Munková, D., Munk, M., & Vozár, M. (2014). Influence of stop-words removal on sequence patterns identification within comparable corpora. In *ICT Innovations 2013: ICT Innovations and Education* (pp. 67-76). Springer International Publishing.
- Musil, T. (2019). Examining structure of word embeddings with PCA. In *Text, Speech, and Dialogue: 22nd International Conference, TSD 2019, Ljubljana, Slovenia, September 11–13, 2019, Proceedings 22* (pp. 211-223). Springer International Publishing.
- Narayanan, A. Chandramohan, M. Venkatesan, R. Chen, L. Liu, Y. & Jaiswal, S. (2017). graph2vec: Learning distributed representations of graphs. *arXiv preprint arXiv:1707.05005*.
- Nastase, V., Roth, B., Dietz, L., & McCallum, A. (2019, June). Proceedings of the Workshop on Extracting Structured Knowledge from Scientific Publications. In *Proceedings of the Workshop on Extracting Structured Knowledge from Scientific Publications*.

- Nguyen, D. Luo, W. Vo, B. Nguyen, L. T. & Pedrycz, W. (2021). Con2Vec: Learning embedding representations for contrast sets. *Knowledge-Based Systems*, 229, 107382.
- Nikfarjam, A., Sarker, A., O'connor, K., Ginn, R., & Gonzalez, G. (2015). Pharmacovigilance from social media: mining adverse drug reaction mentions using sequence labeling with word embedding cluster features. *Journal of the American Medical Informatics Association*, 22(3), 671-681.
- Nizar, N. M. M., Jahanshiri, E., Tharmandram, A. S., Salama, A., Sinin, S. S. M., Abdullah, N. J., ... & Azam-Ali, S. N. (2021). Underutilised crops database for supporting agricultural diversification. *Computers and Electronics in Agriculture*, 180, 105920.
- NRCS, USDA. "The PLANTS Database (<http://plants.usda.gov>). National Plant Data Center, Baton Rouge." Accessed March 30 (2010): 2010.
- Okamura, K. (2022). Scientometric engineering: Exploring citation dynamics via arXiv eprints. *Quantitative Science Studies*, 3(1), 122-146.
- Onal, K. D., Zhang, Y., Altingovde, I. S., Rahman, M. M., Karagoz, P., Braylan, A., ... & Lease, M. (2018). Neural information retrieval: at the end of the early years. *Information Retrieval Journal*, 21, 111-182.
- Ozsoy, M. G. (2016). From word embeddings to item recommendation. arXiv preprint arXiv:1601.01356.
- Pennington, J. Socher, R. & Manning, C. D. (2014, October). Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)* (pp. 1532-1543).
- Pingali, Prabhu L. "CIMMYT 1999/2000 World maize facts and trends. Meeting world maize needs: Technological opportunities and priorities for the public sector." (2001).
- Plahar, W. "Marketing and processing of bambara groundnuts (West Africa). Final Technical Report." (2009).
- Polykalas, Spyros E., George N. Prezerakos, and Agisilaos Konidaris. "An Algorithm based on Google Trends' data for future prediction. Case study: German Elections." In *Signal Processing and Information Technology (ISSPIT), 2013 IEEE International Symposium on*, pp. 000069-000073. IEEE, 2013.
- Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., & Sutskever, I. (2019). Language models are unsupervised multitask learners. *OpenAI blog*, 1(8), 9.
- Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., ... & Liu, P. J. (2020). Exploring the limits of transfer learning with a unified text-to-text transformer. *The Journal of Machine Learning Research*, 21(1), 5485-5551.
- Ramirez-Villegas, J., Jarvis, A., & Läderach, P. (2013). Empirical approaches for assessing impacts of climate change on agriculture: The EcoCrop model and a case study with grain sorghum. *Agricultural and Forest Meteorology*, 170, 67-78.

- Rastogi, C., Stelmakh, I., Shen, X., Meila, M., Echenique, F., Chawla, S., & Shah, N. B. (2022). To ArXiv or not to ArXiv: A study quantifying pros and cons of posting preprints online. arXiv preprint arXiv:2203.17259.
- Rijsbergen, C. V. (1979). Information retrieval. Butterworth-Heinemann.
- Rijdsdijk, J., Wu, L., Perin, G., & Picek, S. (2021). Reinforcement learning for hyperparameter tuning in deep learning-based side-channel analysis. *IACR Transactions on Cryptographic Hardware and Embedded Systems*, 677-707.
- Rong, X. (2014). word2vec parameter learning explained. arXiv preprint arXiv:1411.2738.
- Rossanez, A., Dos Reis, J. C., Torres, R. D. S., & de Ribaupierre, H. (2020). KGen: a knowledge graph generator from biomedical scientific literature. *BMC medical informatics and decision making*, 20, 1-24.
- Saier, T., Krause, J., & Färber, M. (2023). unarxive 2022: All arxiv publications pre-processed for nlp, including structured full-text and citation network. arXiv preprint arXiv:2303.14957.
- Salama, A., Maul, T., Jahanshiri, E. (2017). "Detecting cropping patterns of underutilised crops using online big data", 2nd International conference on "Future Technology Conference". IEEE.
- Montemurro, M. A., & Zanette, D. H. (2019). Language entropy reveals political instability. *PLoS ONE*, 14(2), e0212037.
- Sanh, V., Debut, L., Chaumond, J., & Wolf, T. (2019). DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. arXiv preprint arXiv:1910.01108.
- Sarica, S., & Luo, J. (2021). Stopwords in technical language processing. *Plos one*, 16(8), e0254937.
- Savigny, J. & Purwarianti, A. (2017, August). Emotion classification on youtube comments using word embedding. In 2017 international conference on advanced informatics, concepts, theory, and applications (ICAICTA) (pp. 1-5). IEEE.
- Sebastiani, F. (2002). Machine learning in automated text categorization. *ACM computing surveys (CSUR)*, 34(1), 1-47.
- Shardlow, M., Batista-Navarro, R., Thompson, P., Nawaz, R., McNaught, J., & Ananiadou, S. (2018). Identification of research hypotheses and new knowledge from scientific literature. *BMC medical informatics and decision making*, 18, 1-13.
- Sharma, A., & Kumar, S. (2023). Ontology-based semantic retrieval of documents using Word2vec model. *Data & Knowledge Engineering*, 144, 102110.
- Sherry, Stephen T., M-H. Ward, M. Kholodov, J. Baker, Lon Phan, Elizabeth M. Smigielski, and Karl Sirotkin. "dbSNP: the NCBI database of genetic variation." *Nucleic acids research* 29, no. 1 (2001): 308-311.

- Shiri, F. M., Perumal, T., Mustapha, N., & Mohamed, R. (2023). A comprehensive overview and comparative analysis on deep learning models: CNN, RNN, LSTM, GRU. *arXiv preprint arXiv:2308.12345*
- Sievert, C., Parmer, C., Hocking, T., Chamberlain, S., Ram, K., Corvellec, M., & Despouy, P. (2016). plotly: Create interactive web graphics via Plotly's JavaScript graphing library [Software].
- Sommerauer, P., & Fokkens, A. (2019, August). Conceptual change and distributional semantic models: An exploratory study on pitfalls and possibilities. In *Proceedings of the 1st International Workshop on Computational Approaches to Historical Language Change* (pp. 223-233).
- Song, K., Tan, X., Qin, T., Lu, J., & Liu, T. Y. (2020). Mpnnet: Masked and permuted pre-training for language understanding. *Advances in Neural Information Processing Systems*, 33, 16857-16867.
- Sparck Jones, K. (1972). A statistical interpretation of term specificity and its application in retrieval. *Journal of documentation*, 28(1), 11-21.
- Srivamsi, D., Deepak, O. M., Praveena, M. A., & Christy, A. (2023, April). Cosine Similarity Based Word2Vec Model for Biomedical Data Analysis. In *2023 7th International Conference on Trends in Electronics and Informatics (ICOEI)* (pp. 1400-1404). IEEE.
- St, L., & Wold, S. (1989). Analysis of variance (ANOVA). *Chemometrics and intelligent laboratory systems*, 6(4), 259-272.
- Styawati, S., Nurkholis, A., Aldino, A. A., Samsugi, S., Suryati, E., & Cahyono, R. P. (2022, January). Sentiment analysis on online transportation reviews using Word2Vec text embedding model feature extraction and support vector machine (SVM) algorithm. In *2021 International Seminar on Machine Learning, Optimization, and Data Science (ISMOL)* (pp. 163-167). IEEE.
- Sumathi, B. (2020). Grid search tuning of hyperparameters in random forest classifier for customer feedback sentiment prediction. *International Journal of Advanced Computer Science and Applications*, 11(9).
- Sutskever, I., Vinyals, O., & Le, Q. V. (2014). Sequence to sequence learning with neural networks
- Swoboda, T., Hemmje, M., Dascalu, M., & Trausan-Matu, S. (2016, September). Combining Taxonomies using Word2vec. In *Proceedings of the 2016 ACM Symposium on Document Engineering* (pp. 131- 134). ACM.
- Tan, X. L., Azam-Ali, S., Goh, E. V., Mustafa, M., Chai, H. H., Ho, W. K., ... & Massawe, F. (2020). Bambara groundnut: An underutilised leguminous crop for global food security and nutrition. *Frontiers in Nutrition*, 7, 601496.
- Tang, J. Qu, M. Wang, M. Zhang, M. Yan, J. & Mei, Q. (2015, May). Line: Large-scale information network embedding. In *Proceedings of the 24th international conference on world wide web* (pp. 1067-1077).
- Taxonomy, GBIF Backbone. "doi: 10.15468/39omei." (2016).
- The International Rice Research Institute (IRRI). <http://irri.org/aboutus/our-mission>

- Tissier, J., Gravier, C., & Habrard, A. (2020). Dict2vec: Learning word embeddings using lexical dictionaries. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP) (pp. 635-645).
- Trevisan, Filippo. "Search engines: From social science objects to academic inquiry tools." *First Monday* 19, no. 11 (2014).
- Tripathy, G., & Sharaff, A. (2023). AEGA: enhanced feature selection based on ANOVA and extended genetic algorithm for online customer review analysis. *The Journal of Supercomputing*, 1-30.
- Tshitoyan, V., Dagdelen, J., Weston, L., Dunn, A., Rong, Z., Kononova, O., ... & Jain, A. (2019). Unsupervised word embeddings capture latent knowledge from materials science literature. *Nature*, 571(7763), 95-98.
- Van der Velde, Marijn, Linda See, Steffen Fritz, Frank GA Verheijen, Nikolay Khabarov, and Michael Obersteiner. "Generating crop calendars with Web search data." *Environmental Research Letters* 7, no. 2 (2012): 024022.
- Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). Attention is all you need. *Advances in neural information processing systems*, 30.
- Villa t al. (2005) Villa, Tania Carolina Camacho, Nigel Maxted, Maria Scholten, and Brian Ford-Lloyd. "Defining and identifying crop landraces." *Plant genetic resources: characterization and utilization* 3, no. 03 (2005): 373- 384.
- Walker, N. T., Ultes, S., & Lison, P. (2023). Retrieval-Augmented Neural Response Generation Using Logical Reasoning and Relevance Scoring. *arXiv preprint arXiv:2310.13566*.
- Wang, B., Chen, F., Wang, A., & Kuo, C. C. J. (2019, July). Post-processing of word representations via variance normalization and dynamic embedding. In 2019 IEEE International Conference on Multimedia and Expo (ICME) (pp. 718-723). IEEE.
- Wang, J., Zhang, J., An, Y., Lin, H., Yang, Z., Zhang, Y., & Sun, Y. (2016). Biomedical event trigger detection by dependency-based word embedding. *BMC medical genomics*, 9, 123-133.
- Wang, L., Li, Y., Aslan, O., & Vinyals, O. (2021). WikiGraphs: A Wikipedia text-knowledge graph paired dataset. *arXiv preprint arXiv:2107.09556*.
- Wang, W., Hu, K., & Sainath, T. N. (2022, May). Deliberation of streaming rnn-transducer by non-autoregressive decoding. In ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) (pp. 7452-7456). IEEE.
- Weston, J., Bordes, A., Chopra, S., Rush, A. M., Van Merriënboer, B., Joulin, A., & Mikolov, T. (2015). Towards ai-complete question answering: A set of prerequisite toy tasks. *arXiv preprint arXiv:1502.05698*.
- Williams, R. J., & Zipser, D. (1989). A learning algorithm for continually running fully recurrent neural networks. *Neural computation*, 1(2), 270-280.

- Wohlgenannt, G., & Minic, F. (2016). Using word2vec to Build a Simple Ontology Learning System. In International Semantic Web Conference (Posters & Demos).
- Yao, Z., Sun, Y., Ding, W., Rao, N., & Xiong, H. (2018, February). Dynamic word embeddings for evolving semantic discovery. In Proceedings of the eleventh acm international conference on web search and data mining (pp. 673-681).
- Yeshambel, T., Mothe, J., & Assabie, Y. (2023). Learned Text Representation for Amharic Information Retrieval and Natural Language Processing. *Information*, 14(3), 195.
- Yildiz, B., & Tezgider, M. (2021). Improving word embedding quality with innovative automated approaches to hyperparameters. *Concurrency and Computation: Practice and Experience*, 33(18), e6091.
- Yu, X., & Verma, N. (2022). Improving Model Training via Self-learned Label Representations. *arXiv preprint arXiv:2209.04528*.
- Zhang, Y., Rahman, M. M., Braylan, A., Dang, B., Chang, H. L., Kim, H., ... & Lease, M. (2016). Neural information retrieval: A literature review. *arXiv preprint arXiv:1611.06792*.