# (temp) Using Word Embedding for Citation Recommendation in FinTech Scientific Articles

by Ting-Chun Chen, BSc

Submitted to The University of Nottingham

September 2023

in partial fulfilment of the conditions for the award of the degree of

Master of Science in Data Science

I declare that this dissertation is all my own work, except as indicated in the text

# Contents

**1 Introduction** **1**

1.1 (About Citation) 1

1.2 Natural Language Processing 2

1.3 Text Embedding 3

1.4 Word Similarity and Sentence Similarity 5

1.5 Semantic Search Engine 6

1.6 Study Purpose 6

**2 Literature Review** **8**

2.1 Text Embedding 8

2.2 Text Embedding for Search Engine 9

**References** **11**

**Appendix A Supplementary Materials I** **12**

**Appendix B Supplementary Materials II** **13**

## Abstract

According to the guidelines the abstract should not exceed 300 words. However, it is unlikely that anyone will be counting when you submit. If you still wish to do so, then better use Emacs `count-words` command.

## Acknowledgements

It is good to thank here your supervisors and any sponsoring bodies, as well as any family, friends, cats, dogs etc. that have been supportive during your time at University.

# Chapter 1

# Introduction

## 1.1 (About Citation)

Citations are used to demonstrate research background, existing techniques and pieces of evidence for a statement, playing a critical role in scientific writing. Authors can properly acknowledge the source of information of a statement and avoid plagiarism with an accurate citation. It also serves as a verification that the idea of this statement is provided and supported by previous studies. These days, references and citations also include additional information for search engines and citation recommendation systems to recognise the subfields and similar research of this article.

There has been a noticeable increase in the number of scientific articles being published. 3.8 million scientific articles are being published in 2022, according to the Web of Science database. It is expected that there would be more scientific articles available on the internet. It can be harder for academics to absorb all the new perspectives with their exact sources.

When looking for reference papers, keywords are commonly used in most academic databases and search engines. The authors would then review the results of papers from search engines to evaluate their relevance to our research and reliability. It could cost a huge amount of time and effort to go through full papers, not to mention that the academic database and search engines might miss some relevant papers.

Some academic databases come with citation recommendation tools, which provide relevant articles that share the same category with or are similar to our research. Citation recommendation tools would have a bias towards published journals, impact factors, times being referred to, and the availability of the journal if it's open to everyone or subscription-only.

## 1.2 Natural Language Processing

Natural language processing (NLP) refers to the approach of giving computers the ability to understand the meaning and thoughts of words, sentences and articles just like human beings do, building a way of communication between computer and human language. However, human languages have several characteristics that make it tricky to process them. The variability and informality, such as different vocabularies, syntax and phrases in different cultures or individuals representing a similar meaning, bring difficulties to applying a general transformations pipeline for information extraction. A large amount of noise such as mis-spellings, irrelevant contexts and grammar errors confuse not only human readers but also machines and affect the performance of language processing algorithms, not to mention that people might have different understandings of the same sentence which yield even more inaccuracies. Some advanced circumstances such as sarcasm, irony and the speaker's intention, whose implied meanings are beyond plain text, are more difficult for algorithms to detect and extract straightforwardly. Problems in normal vector data such as missing values and lack of labelling occur in natural language as well.

NLP manages to overcome the challenges of processing human language mentioned above and enables machines to analyse, cluster or classify not only vector instances but also texts. Advanced applications of NLP, such as sentiment analysis, text clustering, text summarisation, human language generation and improving automatic translation, provide solutions and assistance to a wide range of tasks. It reduces our workloads and time-spending and helps improve the interaction between humans and computers by mimicking human language to express in an easy-understanding way for most o the people not only programmers.

## 1.3 Text Embedding

Text embedding, also known as text vectorisation and text featurisation, aims to extract information from words and sentences and represent the semantics and meaning of words by numbers and vectors. The process of extracting information has been through significant development since the year 2000. Various techniques have been proposed, including naive ways of bag-of-words model, TF-IDF encoding, LSA encoding, Word2Vec embeddings and GloVe, along with more advanced methods like BERT and GPT, which we are familiar with and commonly use today.

Text embedding methods can be roughly classified by the level of encoding unit (embed in a word, sentence, or article level), supervised or unsupervised, similarity measurement (such as Euclidean distance or cosine similarity) and other specific technique support such as term-based embedding in particular fields and graph-based embedding. Some advanced methods such as ELMo[1] and GPT[2] use autoregressive models and BERT[3] uses an autoencoder to do bidirectional context encoding. A more complex structure of neural networks and a bigger training corpus could derive a better and more robust model but a huge amount of time to train models as well. Hence, another improvement of embedding methods would be the trade-off between better performance and reasonable training time.

*TF-IDF*

Sample text sample text sample text sample text sample text

*LSA Encoding*

Sample text sample text sample text sample text sample text

*Word2Vec Encoding*

Sample text sample text sample text sample text sample text

*Doc2Vec Encoding*

Sample text sample text sample text sample text sample text

*Universal Sentence Encoder*

Sample text sample text sample text sample text sample text

*GloVe*

Sample text sample text sample text sample text sample text

*Long-Short Term Memory*

Sample text sample text sample text sample text sample text

*BERT*

Sample text sample text sample text sample text sample text

*GPT*

Sample text sample text sample text sample text sample text

*ELMo*

Sample text sample text sample text sample text sample text

## 1.4 Word Similarity and Sentence Similarity

After transforming every word into a vector, similarities between two words are to be calculated. A pair of similar words should act similarly in most of the features we extracted, while a good similarity metric can aggregate the difference in every feature into a single value, which is comparable between each pair of words. The measurements of word similarity are used in word-embedding techniques to evaluate the similarity between prediction value and real value to update parameters in NN model training. Besides Euclidean distance and cosine similarity as semantic similarity measurements mentioned in Chapter 1.2, WordNet also provides path similarity to evaluate similarities of words with a lexical hierarchical structure.

Sentence similarity measurements are more complex and various compared to word similarity since sentences can be considered as combinations of words. The most straightforward approach is to aggregate words in the sentence as a representation to compare with other sentences, which can be seen as a baseline measurement of sentence similarity. The steps are to take averages of every word in each sentence, yield a single vector for each sentence and calculate the Euclidean distance or cosine similarity between two sentences with average vectors. This approach, however, doesn't consider the order of words, while it can have a huge effect on a sentence's meaning when the word order differs. Several algorithms are proposed to map a sentence into a vector and calculate the similarity with the derived value directly from the embedding process. Much of them are extended from an existing word embedding method to apply to sentences or even documents, such as Word2Vec, Sent2Vec and Doc2Vec. Some Approaches consider different levels of embedding at the same time, such as word embedding and sentence embedding of BERT.

*Lexical Relation and WordNet*

Sample text sample text sample text sample text sample text

## 1.5 Semantic Search Engine

Techniques about search engines are a good starting point when it comes to matching a sentence to related articles. Search engines, especially full-text search engines, enable users to look for relative information in a huge full-text database. Among the tasks in a search process, the comparison between keyword and candidate results can be improved with text embedding techniques to retrieve semantic information from keyword and candidate documents, which is known as the semantic search engine.

Semantic search engines enable users to provide a sentence or description instead of just a keyword for querying, while text embedding methods are applied normally to both query texts and candidate documents to evaluate the similarity between their embedded vectors. Semantic search can better catch the contextual meaning within texts and understand the user's intent. Some advanced techniques such as WordNet can also be applied to the search engine to improve the accuracy and coverage of querying results and can serve partly as query expansion that expands a search query with its synonym words or semantic relative words.

## 1.6 Study Purpose

This project is about building a semantic search engine for scientific articles in the fintech field and comparing the performance of using different text embedding techniques. We want to discover and develop a text embedding method that can best describe sentences in FinTech research articles and yield the best result of finding related scientific articles by a citation sentence. This project aims to help academics find proper references for their statement when writing literature reviews and introductions of scientific reports.

Citations are commonly used in writing scientific articles to acknowledge the source of a statement in our work. Though we usually file and organize papers well while doing literature reviews, it happens that some statements are based on months or years of experience or accumulated knowledge in a particular subject. It could be relatively hard work to find an appropriate source for this idea. Hence, we are proposing a method to find the best math articles for our writing. We want to look for articles that mainly describe similar thoughts to the statement we write down in a certain domain in scientific writing. Among all the processes in the natural language processing pipeline, the text embedding techniques would be mostly focused on in this research, which is to convert contents into meaningful numbers and vectors for algorithms to compute, transform and compare. We decide to apply this research in the fintech field and answer the question "What text embedding techniques can best represent a scientific sentence to derive a semantic search engine for the fintech academic articles?"

# Chapter 2

# Literature Review

## 2.1 Text Embedding

Text embedding techniques are improved for specific objectives, fields and styles and applied to a wide range of tasks, including various issues of tweets analysis[Mottaghinia et al., 2020], visualisation in the biomedical field [Oubenali et al., 2022] and sentiment analysis on movie reviews [Sivakumar et al., 2021].

[Khatua et al., 2019] identified crisis-related tweets during the 2014 Ebola and 2016 Zika outbreaks with pre-trained Word2Vec and GloVe models. They found a better classification performance to have a small domain-specific corpus from tweets and scholarly abstracts from PubMed participated in the model. They also observed a higher accuracy from a higher dimension of word vector and skip-gram model than CBOW.

[Lee et al., 2018] utilized SentiWordNet 3.0 to analyse the effect of several negative emotions in hotel reviews. SentiWordNet 3.0 [Baccianella et al., 2010] provided sentiment analysis as classification tasks and word embedding with a frequency-weighted bag-of-words model and the help of WordNet corpus. [Onan, 2020] presented a sentiment analysis approach to product reviews from Twitter. This deep-learning-based method applied TF-IDF weighted GloVe to the CNN-LSTM architecture to do word embedding and outperformed conventional deep-learning methods.

## 2.2 Text Embedding for Search Engine

With the increase of documents and web pages, traditional keywords-based search engines are thought to be powerless to correctly look for users' requirements. Text embedding techniques are applied to search engines to provide machine-readable web pages and semantic annotations to the algorithm to yield a more accurate search result. Many websites are applying text embedding to their search engines, including Google Search, Microsoft's Bing Search, Amazon Search and many e-commerce websites and platforms. Not to confuse the search engine using text embedding and the search engine using semantic web search language, in this paper, we refer to the semantic search as the searching approach with text embedding or other semantic retrieval techniques.

Regarding the search engines for scientific articles, [Eisenberg et al., 2017] proposed a semantic search for Biogeochemical literature that undergoes paper research by comparing the concepts extracted from queries and academic literature. However, the concept extraction component in the workflow of this research is annotated by domain experts, which can be done automatically by NLP approaches mentioned in the future directions chapter. [Fang et al., 2018] performed a biomedical article-searching approach called Semantic Sequential Dependence Model (SSDM) that combined semantic information retrieval techniques and the traditional SDM model. Word embedding techniques of the Neural Network Language Model (NNLM)[bengio et al., 2003], Log-Bilinear Language Model (LBL)[Mnih et al., 2007] and Word2vec are applied to the literature corpus to find synonyms by KNN classification algorithm and generate a domain-specific thesaurus. The thesaurus is then utilized to extend query keywords and the SDM played an important role in the combination strategy of the extended keywords for further comparison to documents.

Compare to the keyword embedding approaches, applying sentence embedding or other wider-level embedding techniques to search engines can better preserve information for query but be more challenging at the same time. [Palangi et al., 2016] managed to embed semantic information at a sentence level by using LSTM-RNN (Long Short-Term Memory - Recurrent Neural Network) framework to do web document retrieval, and compare the summarisation sentence vector and query sentence vector to yield the best searching result.

### 2.2.1 Second Level Heading

Sample text sample text sample text sample text sample text sample text sample text sample text sample text sample text sample text sample text sample text sample text sample text sample text sample text.

*Third Level Headings Do Not Appear in the Table of Contents*

Sample text sample text sample text sample text sample text sample text sample text sample text sample text sample text sample text sample text sample text sample text sample text sample text (Lamport, 1994)

# References

Lamport, L. (1994). *LaTeX : A document preparation system* (Second ed.). Addison-Wesley.

# Appendix A

# Supplementary Materials I

Sample text sample text sample text sample text sample text sample text sample text sample text sample text sample text sample text sample text sample text sample text sample text sample text sample text.

Appendix B

# Supplementary Materials II

Sample text sample text sample text sample text sample text sample text sample text sample text sample text sample text sample text sample text sample text sample text sample text sample text sample text.