**COMP4003 / COMP4031 / COMP4026 Project Plan**

**Student:** Ting-Chun Chen

**Student ID:** 20498346

**Course:** MSc Projects COMP4096

**1. Project title.**

This project is about building relations between sentences and scientific articles in FinTech based on text embedding and document summarisation techniques. We want to develop an information extraction method from texts that can best describe sentences in FinTech research articles and yield the best result of relating citation sentences and scientific articles.

**2. Statement of the research problem.**

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 the processes within NLP (natural language processing) pipeline, the word-embedding, sentence-embedding and BERT techniques would be mostly focused on in this research, which is to convert contents into meaningful numbers for algorithms to compute, summarise and compare. We decide to apply this research in the fintech field and answer the question “What feature extraction techniques can best represent a scientific sentence to derive an academic writing citation-hinting system in fintech field?”

**3. Related work.**

Information extraction from texts aims to represent the semantics and meaning of words with numbers and vectors. The process of extracting information has been through significant development since the year 2000. Various techniques have been proposed, including the 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. Information extraction methods can be roughly classified by the level of encoding unit (such as word, sentence, or a whole article), supervised or unsupervised, similarity measurement (such as Euclidean distance) and other specific technique supports 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 network and bigger training corpus could derive a better and more robust model but a huge time-consuming as well. Hence, another improvement of embedding methods would be the trade-off between better performance and reasonable training time.

Information extraction techniques are improved for specific objectives, fields and styles and applied to a wide range of tasks, including visualisation in medical field [4], topic modelling on social networks of Twitter and Reddit[5] and sentiment analysis on product reviews[6]. As for the application to scientific sentences, [7] used embedding to capture complex materials science concepts to extract knowledge and relationship from scientific literature. [8] used 15 text representation models to construct a similar article recommendation system in biomedical field. [9] proposed a sentence-level citation recommendation system that used CNN to recognise candidate citation sentences and FastText as the word embedding method to extract information from texts, without summarising an article to sentences.

**4. Methodology.**

We choose python 3.7 to be the main tool of this research due to its free cost, easy understanding and lots of libraries supported. The pyPDF2 library is used to convert PDF files to plain text. The pre-processing steps including noise removal, tokenization, stemming and sentence splitting are also applied to the data before word embedding. The citation sentences and their referencing articles would be extracted to form our own fintech dataset and separated into a training set and test set.

Since we would like to focus on text embedding techniques, we choose an unsupervised summarisation method to reduce the time-consuming of model training. To summarise candidate scientific articles that are to be referenced, we choose TextRank[10] to find the sentence that is most similar to any others to be the summary of this article. To extract information from sentences, we are applying several methods to citation sentences and summary sentences, including word-embedding techniques’ (such as word2vec, fastText[11][12] and FinText[13]), and sentence-embedding techniques’ (such as Sent2Vec[14] and InferSent[15]), to generate representation vectors. To evaluate the embedding model, cosine similarity between representation vectors of summary sentence and its citation sentence is calculated and we will analyse the performance of different text representation methods.

**5. Programme of work.**

* **Work Package 1 - Project Plan.** Determine research question and finish project plan with preliminary literature review. Breakdown the research into work packages and arrange a schedule. Milestone 1 (M1) – project plan completed.
* **Work Package 2 - Literature Review.** Perform literature review about word embedding techniques, sentence similarity, citation-based document summarisation researches and NLP on fintech scientific articles. Confirm the details of step-by-step methods and what techniques to use. Milestone 2 (M2) – introduction and literature review chapters of dissertation completed.
* **Work Package 3 - Dataset and Environment Preparation.** Download at least 1000 scientific articles from top business and financial journals. Convert PDF file to plain text. Finish preprocessing steps mentioned above. Extract citation sentences from articles. Prepare training and test dataset. Install and prepare the programming environment, libraries and toolkits. Milestone 3 (M3) – material section of dissertation completed.
* **Work Package 4 - Document Summarisation and Text Embedding.** Apply TextRank to articles to have their summaries. Apply text representation techniques to sentences. Convert all texts data into vectors. Milestone 4 (M4) – material and method chapter of dissertation completed.
* **Work Package 5 - Similarity Evaluation and Analysis.** Apply similarity to sentences and evaluate the performance between techniques. Generate tables and figures to best describe the results. Milestone 5 (M5) – result and analysis chapter completed.
* **Work Package 6 - Dissertation Complete and Proofreading.** Milestone 6 (M6) – conclusion and discussion chapter completed.

**6. Time plan.**

A screenshot of a computer

Description automatically generated with medium confidence

**983/1000 words**

**Reference**

1. Peters, M.E., Neumann, M., Iyyer, M., Gardner, M., Clark, C., Lee, K., Zettlemoyer, L. *Deep contextualized word representations*. in *Proceedings of NAACL*. 2018.

2. Brown, T., et al., *Language models are few-shot learners.* Advances in neural information processing systems, 2020. **33**: p. 1877-1901.

3. Niven, T., H.Y. Kao, and Acl. *Probing Neural Network Comprehension of Natural Language Arguments*. in *57th Annual Meeting of the Association-for-Computational-Linguistics (ACL)*. 2019. Florence, ITALY: Assoc Computational Linguistics-Acl.

4. Oubenali, N., et al., *Visualization of medical concepts represented using word embeddings: a scoping review.* Bmc Medical Informatics and Decision Making, 2022. **22**(1): p. 14.

5. Curiskis, S.A., et al., *An evaluation of document clustering and topic modelling in two online social networks: Twitter and Reddit.* Information Processing & Management, 2020. **57**(2): p. 21.

6. Onan, A., *Sentiment analysis on product reviews based on weighted word embeddings and deep neural networks.* Concurrency and Computation-Practice & Experience, 2021. **33**(23): p. 12.

7. Tshitoyan, V., et al., *Unsupervised word embeddings capture latent knowledge from materials science literature.* Nature, 2019. **571**(7763): p. 95-+.

8. Zhang, L., et al., *A comparative evaluation of biomedical similar article recommendation.* Journal of Biomedical Informatics, 2022. **131**: p. 14.

9. Wang, H.C., J.W. Cheng, and C.T. Yang, *SentCite: a sentence-level citation recommender based on the salient similarity among multiple segments.* Scientometrics, 2022. **127**(5): p. 2521-2546.

10. Mihalcea, R. and P. Tarau. *Textrank: Bringing order into text*. in *Proceedings of the 2004 conference on empirical methods in natural language processing*. 2004.

11. Bojanowski, P., et al., *Enriching word vectors with subword information.* Transactions of the association for computational linguistics, 2017. **5**: p. 135-146.

12. Joulin, A., et al., *Bag of tricks for efficient text classification.* arXiv preprint arXiv:1607.01759, 2016.

13. Rahimikia, E., S. Zohren, and S.-H. Poon, *Realised volatility forecasting: Machine learning via financial word embedding.* arXiv preprint arXiv:2108.00480, 2021.

14. Pagliardini, M., P. Gupta, and M. Jaggi, *Unsupervised learning of sentence embeddings using compositional n-gram features.* arXiv preprint arXiv:1703.02507, 2017.

15. Conneau, A., et al., *Supervised learning of universal sentence representations from natural language inference data.* arXiv preprint arXiv:1705.02364, 2017.