Leveraging the Fusion-in-Decoder for Label Classification

Azumi Okuda Hideya Mino Taro Miyazaki Jun Goto
NHK Science and Technology Research Laboratories
{okuda.a-gc, mino.h-gq, miyazaki.t-jw, goto.j-fw}@nhk.or.jp

Abstract

Text classification is an important technique in natural language processing for categorizing text into appropriate domains. With the increasing amount of textual data, robust text classification is in high demand. This paper focuses on single-label classification of text for scholarly articles, aiming to analyze a large number of papers. Inspired by the successful Fusion-in-Decoder method used in question-answering tasks, we propose an accurate method suitable for long articles. We evaluate the effectiveness of our method through experiments on single-label classification with scholarly articles, demonstrating its high F1 scores.

1 Introduction

Text classification plays a significant role in natural language processing, and several methods have been proposed (Kim, 2014; Zhang and LeCun, 2016; Zheng and Yang, 2019; Minaee et al., 2021). This paper focuses on a classification task for scholarly articles. The rapid growth of scholarly articles, for instance over 370,000 papers on COVID-19 published by 2022, necessitates efficient analysis. Pre-trained language models such as BERT (Devlin et al., 2019) face challenges in processing long texts such as scholarly articles. They are often limited by input length, leading to token overflow and utilization of only the initial part of the text. Additionally, these models do not consider the importance of each sentence in the full text for accurate label classification.

To address these limitations, we propose a method that leverages techniques from question answering (QA) tasks to enhance label classification accuracy for long texts. Specifically, we extract a set of sentences deemed informative based on the summary section, which represent a collection of important information in the paper. We combine the vector representations of these sentences using Fusion-in-Decoder (FiD) (Izacard and Grave, 2021b), a high-performing approach in QA tasks, to estimate the label. Although FiD was originally proposed for QA tasks, we applied it to paper analysis because it can extract important sentences from long documents and implement them in neural networks. We evaluate our proposed method using the CORD-19 dataset (Beltagy et al., 2020) of scientific papers on COVID-19. The results of our evaluation experiment demonstrate the effectiveness of our approach.

2 Related Work

Existing pre-trained language models generally used in text classification, such as BERT (Devlin et al., 2019), have limitations on the input length, preventing the utilization of all information in long documents during fine-tuning. While Longformer (Beltagy et al., 2020) and BigBird (Zaheer et al., 2021) are notable approaches for handling long documents, they still have limitations regarding input length.

When considering the handling of a large amount of data, QA tasks can provide valuable insights. Karpukhin et al. (2020) proposed Dense Passage Retriever (DPR), which retrieves relevant passages from a large number of documents, achieving high accuracy in an open-domain QA task.

Lee et al. (2019) introduced the Open-Retrieval Question Answering (ORQA) model for open-domain QA tasks. The ORQA model comprises a retriever that identifies relevant sentences from external knowledge and a reader that extracts answers from the retrieved sentences and questions. Building on the ORQA model, Izacard and Grave (2021b) proposed Fusion-in-Decoder (FiD) by improving the reader part. Additionally, Izacard and Grave (2021a) proposed a method to train the retriever using the knowledge from the reader, leading to improved accuracy in QA tasks. In this work, we adapt Izacard et al.'s FiD to the task of single-label classification.

Proceedings of the 2nd Workshop on Information Extraction from Scientific Publications, pages 68–73

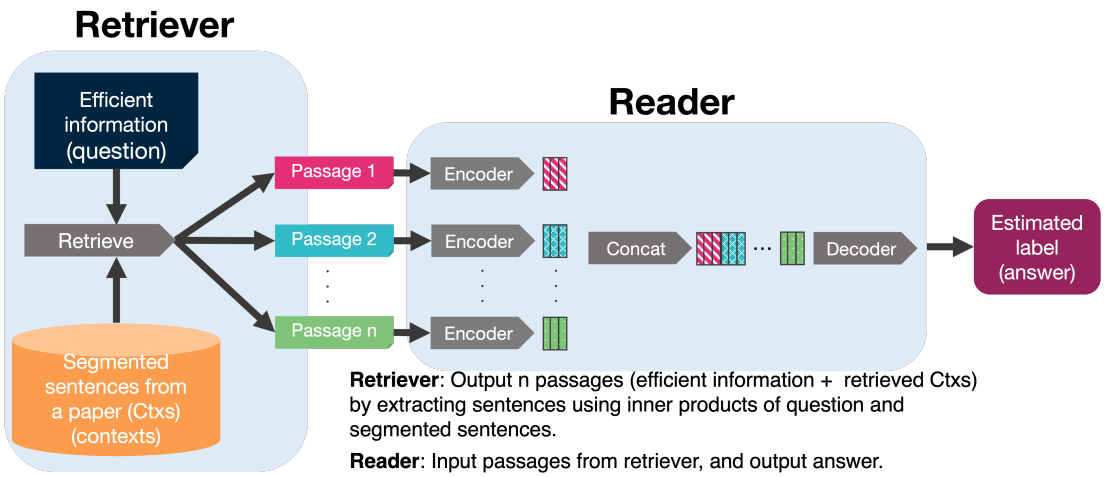

Figure 1: Overview of the proposed method.

3 Methodology

3.1 The Fusion-in-Decoder for Label Classification

The core idea of the proposed method involves incorporating the Fusion-in-Decoder (FiD) into a label classification task. Figure 1 shows a diagram of the proposed method with the inclusion of the FiD. The FiD comprises two components: the retriever and reader (Izacard and Grave, 2021b).

Retriever The retriever component requires *efficient information*, referred to as a question or query in the QA task, and segmented sentences of a paper (*Ctxs*), referred to as a context. The retriever is based on Dense Passage Retriever (DPR) (Karpukhin et al., 2020). In Figure 1, *efficient information* and *Ctxs* are embedded as dense vector representations by BERT (Devlin et al., 2019) networks. The retriever is trained to reflect the relevancy between representation vector of each sentence by the dot product. The objective of using dot product is to ensure that the inner product of the efficient information and the segmented sentences (*Ctxs*) produces appropriate value for retrieving relevant sentences from *Ctxs* with using efficient information as query. We take into consideration the cross-attention score in the reader, which we will be described in following section. This is because a sentence with several attentions in the previous reader, considered to be more useful for classification. Finally, the retriever outputs a set of sentences from *Ctxs*, which is referred to as passage retrieval. This component retrieves valuable *Ctxs* for label classification based on *efficient information*.

Reader The reader is based on a pretrained transformer-based sequence-to-sequence network. The reader component requires passages, which comprise both the *efficient information* and the retrieved *Ctxs*. More precisely, each sentence in *Ctxs* is concatenated with the *efficient information*, which is referred to as a passage. These passages are independently processed by each reader's encoder, which outputs an embedded expression for each passage. The encoded outputs are then concatenated and fed into the decoder. The decoder generates an *estimated label* for the paper as an answer in the QA task.

Repeated training During the process in the reader, the cross-attention scores are calculated between the *efficient information* and passage in the transformer model. Based on the assumption that passages with high cross-attention scores calculated by the reader contribute to accurate label estimation, the retriever calculates the inner product between the *efficient information* and segmented sentences within the passages and is trained to establish an association between them. During the process in the retriever, the passages used in the reader are updated. Due to their interdependence, the repeated training of the reader and retriever models leads to an improvement in the accuracy of each model.

Segmented sentences for retriever Izacard and Grave (2021b) incorporated external knowledge sources, such as Wikipedia, for the segmented sentences (*Ctxs*) in the QA task. In this paper, we utilize a scholarly article from the same domain that contains the entire paper as the *Ctxs*, which includes pertinent information for classifying pa-

pers. We assume that each scholarly article in training data comprises three components: an abstract, main text, and label that represent the genre of the article. Scholarly articles are often longer than typical model such as BERT can handle. FiD utilize shorter sentences that extracted using retriever, so it can handle such long articles.

3.2 Training Process

Based on Section 3.1, the proposed model is trained as follows:

1. Both the abstract and main text of an article are divided into sentences. During the initial training of the reader, the segmented abstracts which have correct labels, are utilized as the contexts. To simplify the training process, a fixed *efficient information* "What genre best describes this abstract?" is employed for all papers, instead of selecting essential information for label estimation as part of the question. A detailed analysis of appropriate *efficient information* is provided in Section 5.

2. The retriever is trained using the cross-attention scores calculated by the reader. The objective is to ensure that the inner product of the *efficient information* and the segmented sentences (*Ctxs*) produces an appropriate value. By optimizing this training objective, the retriever trains to select relevant *Ctxs* that align well with the given *efficient information*.

3. Using the retriever trained in step 2, the relevant sentences are retrieved from *Ctxs* using FAISS (Johnson et al., 2019). The objective is to extract highly relevant sentences as contexts and avoid extremely short sentences. This process, known as passage retrieval, helps identify and select the most informative and meaningful sentences from *Ctxs* for further analysis.

4. The *efficient information* and *Ctxs* extracted in step 3 are fed into the new reader model as a passage, and the reader is re-trained based on this input.

5. Steps 2 to 4 are repeated alternately to iteratively train the reader and the retriever.

4 Experiments

4.1 Experimental Settings

4.1.1 Dataset

We conducted experiments using the CORD-19, which is a collection of papers released by the Allen Institute for AI[1]. The dataset is open access and includes papers sourced from PubMed, PubMed Central, the World Health Organization's COVID-19 database, and preprint servers such as bioRxiv, medRxiv, and arXiv. Each paper in CORD-19 is accompanied by metadata, including author names, submission dates, and acquisition sources, along with the abstract. For our experiments, we used 7,127 papers extracted from CORD-19, specifically from the bioRxiv. Each paper is associated with 25 research field labels. The average number of sentences in the full text of the papers is approximately 154.

4.1.2 Training Setup

We implemented our approach with FiD (Izacard and Grave, 2021b) [2].

The reader was initialized with the pretrained text-to-text transfer transformer (T5) base model (Raffel et al., 2020) with 220 million paremeters, available in the HuggingFace Transformers library.

During the initial training of the reader, the contexts (*Ctxs*) cannot be used because the retriever is not yet trained. We assumed that the abstract of each paper would be effective for label classification and used it as the initial value of *Ctxs*. We trained the readers for 20K steps.

The retriever was initialized with pretrained BERT base model (Devlin et al., 2019). For the retriever's output, we selected the top 20 sentences, excluding those comprising 10 words or less, from the search results of the paper database. We trained the retriever for 50k steps.

We fine-tuned the reader and the retriever using Adam (Kingma and Ba, 2017) with a constant learning rate 10^{-4} and dropout rate of 10%. The loss function and other settings related to learning followed the original FiD settings.

4.1.3 Evaluation

For the baseline, we used T5 where the full text of each paper was treated as a single passage to simulate T5 embeddings of full texts.

[1] https://allenai.org/data/cord-19
[2] https://github.com/facebookresearch/FiD

	Ctxs	Iteration	Micro-F1	Macro-F1
T5(baseline)	-	-	0.552	0.366
Proposed Method	abstract	0	0.555	0.362
	retrieved	1	0.562	0.298
	retrieved	2	0.554	0.339
	retrieved	3	**0.575**	0.377
	retrieved	4	0.570	**0.386**
	retrieved	5	0.551	0.350
	retrieved	6	0.541	0.305
	retrieved	7	0.552	0.354
	retrieved	8	0.540	0.319

Table 1: Experimental results.

For the evaluation, we employed the widely used classification metrics, namely Micro-F1 and Macro-F1, which provide insights into the overall performance on the classification tasks. We used NVIDIA V100 for training.

4.2 Experimental Results

Table 1 shows the experimental results of the label classification task. The "Iteration" column indicates the number of iterations of the reader and retriever.

Our method outperformed T5 in both Micro-F1 and Macro-F1. This is because the proposed method takes each sentence into account by the reader, which distinguishes it from both baseline methods performed. In the T5 baseline, all sentences were inputted as a single sentence, whereas each sentence was inputted separately for the proposed method. Our proposed method improved the accuracy by learning long sentences that exceed the T5 token limit.

We show F1 scores of the proposed method for each iteration in Table 1. This sequential process would allow the model to improve and make more accurate predictions. Comparing the initial learned reader with the T5 baseline, we observed that both Micro-F1 and Macro-F1 achieved similar levels of accuracy. This suggests that the process of cutting off long full texts and utilizing segmented abstracts as *Ctxs* is sufficient for achieving comparable performance in using the entire text.

In the proposed method, we utilize the segmented abstract as *Ctxs* for the first reader. The output of the retriever, which is trained based on the reader's output, serves as the input for the subsequent reader in the pipeline. During the repeated iterations, we observed that the Micro-F1 score remained relatively unchanged until the fourth iteration, while the Macro-F1 showed improvement. At the fifth iteration, both the Micro-F1 and Macro-F1 scores decreased. Based on these observations, we decided to stop the iteration. It is inferred that the reason why the accuracy improved by continuing the iteration is that it became possible to retrieve further important information written in full text. Therefore, it is thought that the improvement in accuracy saturated after several iterations. This pattern aligns with findings from previous studies on QA tasks using Fusion-in-Decoder (FiD), where it was reported that performance tends to improve up to approximately the fourth iteration (Izacard and Grave, 2021a). Similarly, in our study, it appears that the performance improvement reached a point of saturation (3rd or 4th iteration), beyond which further iterations did not lead to significant gains.

Efficient information	Ctxs	Iteration	Micro-F1	Macro-F1
fix	abstract	0	0.555	0.362
fix	full text	0	0.590	0.380
abstract	abstract	0	0.570	0.396

Table 2: Results of additional experiments for analyzing the effects of changing efficient information.

5 Analysis

In Table 1, we used segmented abstracts as the initial contexts (*Ctxs*). To investigate the effect of using a larger amount of text, we replaced the abstracts with segmented full texts. The results (second row in Table 2) show that using full text yields higher Micro-F1 and Macro-F1 scores compared to using abstracts (first row in Table 2). This indicates that providing more context to the reader contributes to improved accuracy. However, it is important to note that increasing the context size also increases the computation time. In this study, training the reader with full text required approximately 9 hours, while training with abstracts only required approximately 3 hours.

In Table 2, we show the result of a supplementary experiment. In Table 1, our initial approach used a common phrase as *efficient information* in section 3.2. However, the original Fusion-in-Decoder (FiD) used a characteristic sentence for estimating the answer. The results without any iterations (second row in Table 1) estimate labels using abstracts without employing retrievers. The Micro-F1 score is similar to the baseline (first row in Table 1), indicating that abstracts contain useful information for label classification. Therefore, instead of using a common fixed phrase for effi-

cient information, we experimented using abstract as *efficient information*.

Compared to the experiment described in Section 4 without any iteration (first row of Table 2), both the Micro-F1 and Macro-F1 have improved the accuracy. This suggests that the selection of *efficient information* is significant in improving the accuracy of label classification.

6 Conclusion

We extended the Fusion-in-Decoder (FiD) approach, originally designed for question answering, to a label classification task for scholarly papers. Through experiments using papers related to COVID-19, we validated the effectiveness of the proposed method.

For future work, we plan to improve the retriever's performance by refining the input selection. Since a retriever model is trained using only cross-attention scores of a reader model for references, we will find new additional criteria to get more effective passages. Also, we will conduct experiments on other such datasets to confirm the effectiveness of the proposed method.

Limitations

We have not conducted human evaluation to confirm whether the output passages generated by a retriever model are the most effective information for a reader model.

Acknowledgements

We would like to thank our colleagues, Ichiro Yamada, Yuki Yasuda, Taichi Ishiwatari and Kinugawa Kazutaka for useful discussions and advice. We also thank Simon Clippingdale for proofreading our English and the anonymous reviewers for their comments and suggestions.

References

Iz Beltagy, Matthew E. Peters, and Arman Cohan. 2020. Longformer: The long-document transformer. *arXiv preprint arXiv:2004.05150*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Gautier Izacard and Edouard Grave. 2021a. Distilling knowledge from reader to retriever for question answering. In *The International Conference on Learning Representations*.

Gautier Izacard and Edouard Grave. 2021b. Leveraging passage retrieval with generative models for open domain question answering. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics*, pages 874–880.

Jeff Johnson, Matthijs Douze, and Hervé Jégou. 2019. Billion-scale similarity search with GPUs. *IEEE Transactions on Big Data*, 7(3):535–547.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6769–6781, Online. Association for Computational Linguistics.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1746–1751, Doha, Qatar. Association for Computational Linguistics.

Diederik P. Kingma and Jimmy Ba. 2017. Adam: A method for stochastic optimization. In *Proceedings of the 3rd International Conference on Learning Representations*.

Kenton Lee, Ming-Wei Chang, and Kristina Toutanova. 2019. Latent retrieval for weakly supervised open domain question answering. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6086–6096, Florence, Italy. Association for Computational Linguistics.

Shervin Minaee, Nal Kalchbrenner, Erik Cambria, Narjes Nikzad, Meysam Chenaghlu, and Jianfeng Gao. 2021. Deep learning-based text classification: A comprehensive review. *ACM Comput. Surv.*, 54(3):62:1–62:40.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67.

Manzil Zaheer, Guru Guruganesh, Avinava Dubey, Joshua Ainslie, Chris Alberti, Santiago Ontanon, Philip Pham, Anirudh Ravula, Qifan Wang, Li Yang, and Amr Ahmed. 2021. Big bird: Transformers for longer sequences.

Xiang Zhang and Yann LeCun. 2016. Text understanding from scratch. *arXiv preprint arXiv:1502.01710*.

Shaomin Zheng and Meng Yang. 2019. A new method of improving bert for text classification. In *Intelligence Science and Big Data Engineering. Big Data and Machine Learning: 9th International Conference, IScIDE 2019, Nanjing, China, October 17–20, 2019, Proceedings, Part II 9*, pages 442–452. Springer.