Master's Thesis Presentation for the 142th Dissertation Evaluation

A Study of Data Augmentation for Dense Passage Retrieval using Corpus-Passage Frequency-Based Token Deletion

밀집 벡터 구절 검색을 위한 코퍼스-구절 빈도 기반 토큰 삭제 방식 데이터 증강 연구

Kyumin Kim

kyumin3537@gmail.com

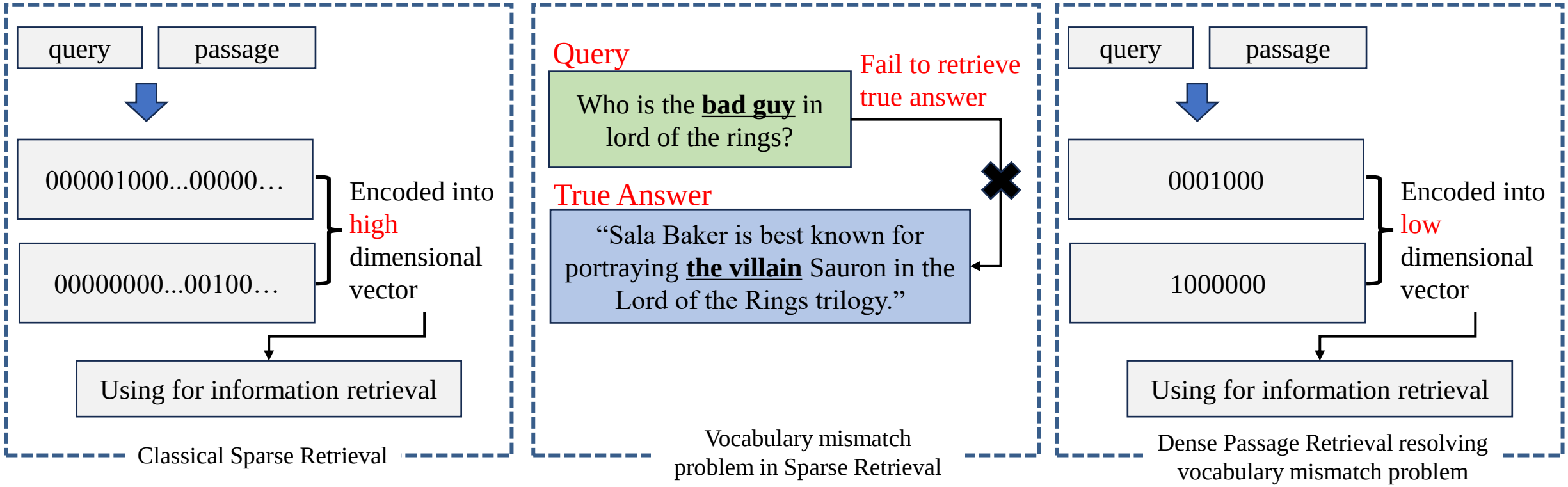
13th November, 2024

Contents

Introduction

Passage Retrieval

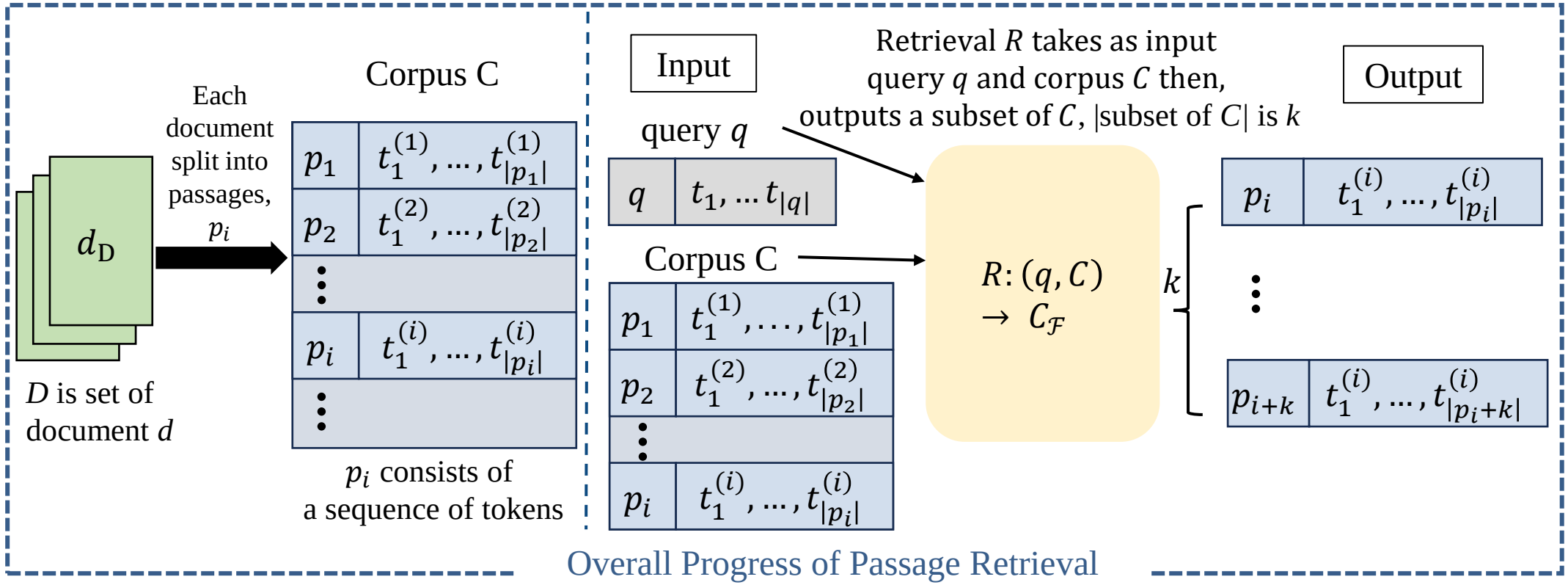
- Question answering aims to find the correct answer to a given question and at the same time, information retrieval techniques are presented as solutions. It's important to search for passages that contain key information on which to base correct answer.
- Unlike Classical Sparse Retrieval, Dense Passage Retrieval (DPR) computes similarity based on a dense vector representation of the query and passages to find the passages that it believes are most relevant to the query and reduce the vocabulary mismatch problem.



Introduction

Objective and Standard Procedure

- Unlike Classical Sparse Retrieval, Dense Passage Retrieval [2] takes input queries and a corpus of passages and outputs a subset of the corpus that are considered correct answers to the query. Dense vector representation is obtained by embedding.
- To train retrieval, a large number of question-answer pairs are needed. The goal is to find the document with the embedding closest to the embedding of the query in the entire corpus for an input query.

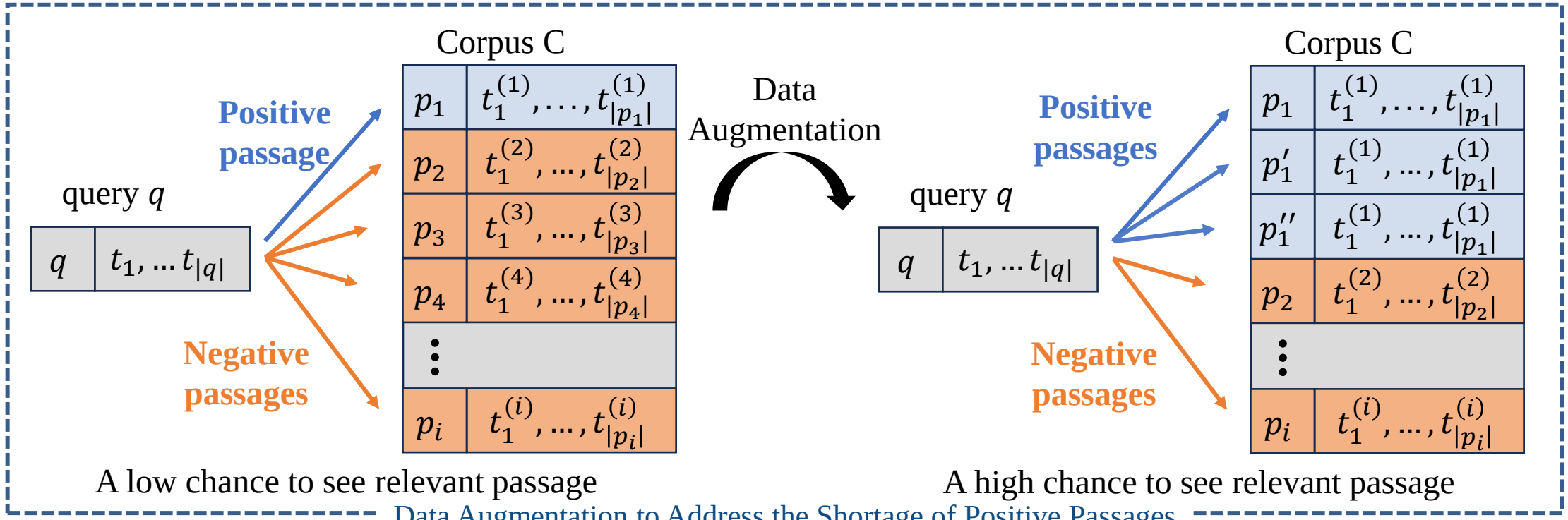


[1] Karpukhin, V., Oguz, B., Min, S., Lewis, P., Wu, L., Edunov, S., … & Yih, W. T. (2020, November). Dense Passage Retrieval for Open-Domain Question Answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)* (pp. 6769-6781).

Introduction

Problem and Strategy

- To obtain accurate embeddings, which are crucial for DPR, a substantial number of relevant passages for each query is required. However, in practice, the number of such passages is significantly limited, thereby constraining performance.
- As a strategy to overcome the scarcity of relevant documents for queries, data augmentation can be considered to synthesize additional relevant passages. Through this approach, the information retrieval performance of the DPR can be enhanced.

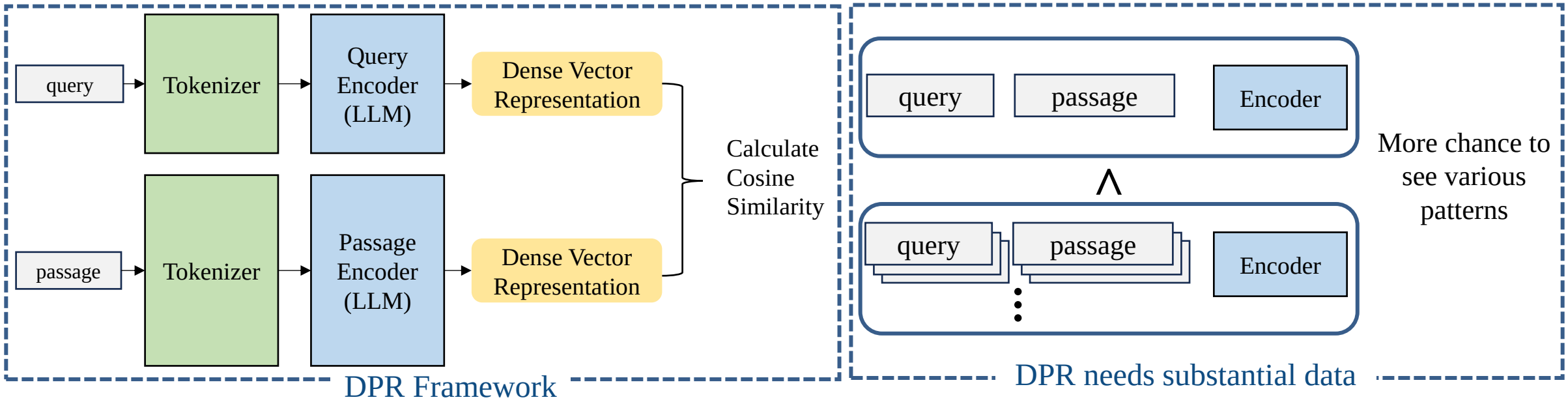


Data Augmentation to Address the Shortage of Positive Passages

Related Work

Dense Passage Retrieval [1]

- Dense passage retrieval (DPR) uses dense representations to retrieve information. This approach involves creating embeddings for both queries and documents to find the closest matches efficiently.
- In situations where training data is limited, both the query encoder and document encoder are unable to observe diverse patterns, resulting in reduced learning performance. To overcome this, data augmentation is used to increase the amount of training data.

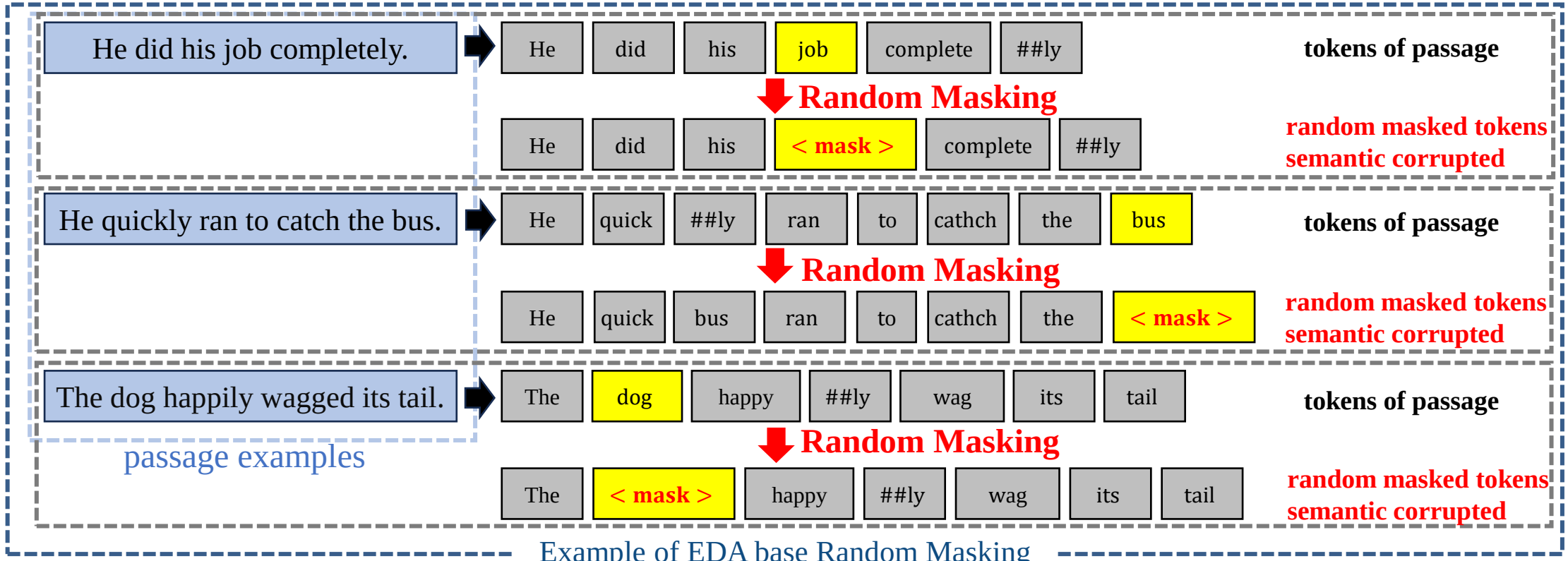


[1] Karpukhin, V., Oguz, B., Min, S., Lewis, P., Wu, L., Edunov, S., … & Yih, W. T. (2020, November). Dense Passage Retrieval for Open-Domain Question Answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)* (pp. 6769-6781).

Related Work

Easy Data Augmentation [2]

- Easy Data Augmentation (EDA) [3] methods are often considered as augmentation techniques for text data. These methods include random deletion, random swapping, etc. In this approach, random deletion method is selected as a target for improvement.
- Because random tokens are deleted, there is a possibility of deleting meaningful tokens during the information retrieval process. Therefore, it is necessary to take account on the importance of the tokens and to delete the tokens that are less important for semantic.



[2] Wei, J., & Zou, K. (2019, November). EDA: Easy Data Augmentation Techniques for Boosting Performance on Text Classification Tasks. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP) (pp. 6382-6388).

Related Work

Common Drawback and Solution

- Because previous studies randomize token deletion, they may delete tokens that are important for phrase formation. The probability of deletion follows distribution reflecting the importance of each token, rather than uniform distribution.
- To achieve this, passage frequency and corpus frequency are used to consider the relative importance of each token in the passage, which is reflected in the probability of deletion.

Algorithm	Author, year	Journal / Conference	Summary	Drawback
Dense Passage Retrieval	Karpukhin (2020)	EMNLP	It leverages a large language model for question answering	Due to the nature of the question answering dataset, there is a lack of data for training because the answers to the questions are matched as one
Easy Data Augmentation	Wei, J (2019)	EMNLP	Augment text data by deleting random tokens	Random deletion can delete important tokens in the question answering process

Proposed Method

Notation and Definition

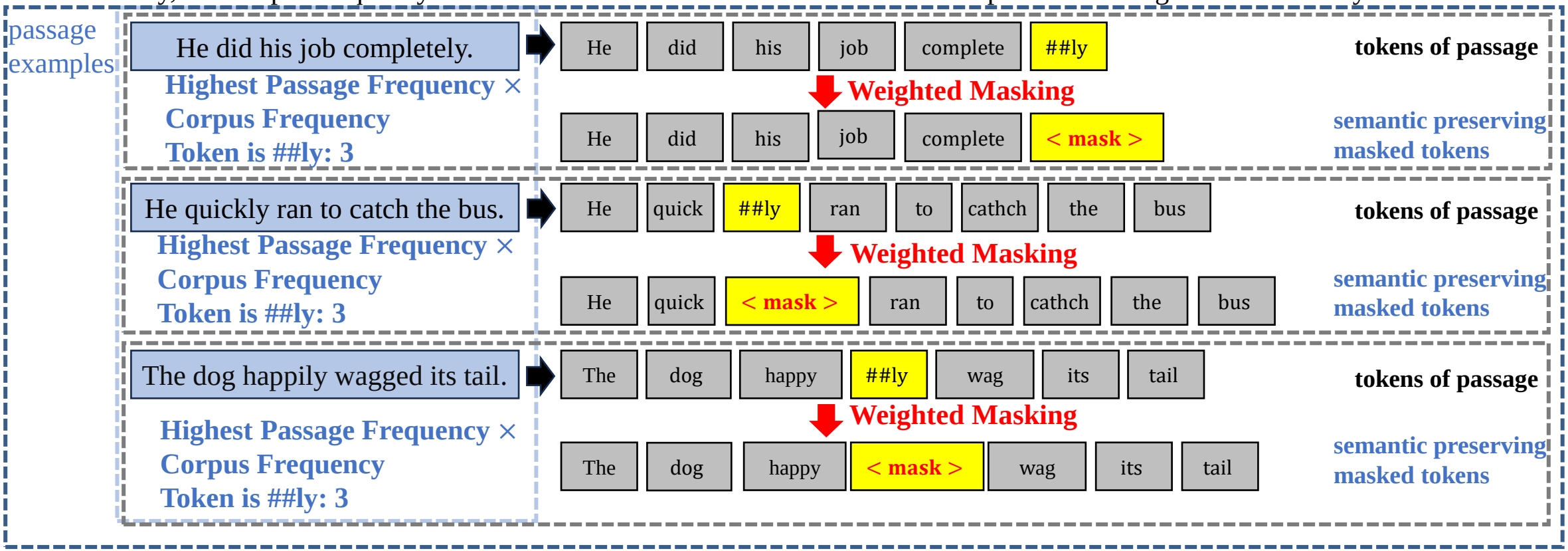
- Given a collection of D documents, each split into text passages of equal length, creating total of M, total documents.
  Corpus is denoted as $C = \{p_1, p_2, \dots, p_M\}$, where each passage $p_i = < t_1^{(i)}, \dots, t_{|p_i|}^{(i)} >$.
- Query q ,consists of sequence of tokens, and a corpus C are taken as input of retrieval function $R: (q, C) \rightarrow C_{\mathcal{F}}$ which returns a much smaller $filtered\ set$ of texts $C_{\mathcal{F}} \subset C$, where $|C_{\mathcal{F}}| = k \ll |C|$.

Symbol	Name of Symbol	Explain
D	Documents	A collection of documents
M	Number of passages	Number of passages originated from documents
C	Corpus	A set of passages
p_i	i^{th} passage	A passage consists of sequence of tokens
t	token	A token from passage
q	query	A query consists of sequence of tokens
R	Retrieval function	Retrieval function takes input query and Corpus, and outputs subset of Corpus
$C_{\mathcal{F}}$	Filtered Corpus	A subset of Corpus

Proposed Method

Procedural Overview

- As a way to consider the importance of tokens, concept of passage frequency and corpus frequency was selected. This concept is a way to statistically calculate how important a token in a passage is in the entire corpus.
- Low passage frequency tokens are crucial for distinguishing and understanding passages, as they often indicate unique content. Similarly, low corpus frequency tokens are share these characteristics and are emphasized through statistical analysis.



Proposed Method

Corpus-Passage Frequency- based Deletion

- For each token, the importance is calculated by considering the passage it belongs to and the entire corpus. To do this, the passage frequency and corpus frequency formulas are created and geometric mean is applied to determine the weight.
- Not only stop words are deleted, but the importance of tokens is also calculated by reflecting the characteristics of the corpus. For example, in a dataset about "song," the frequently occurring word "song" will have lower importance.

$$PF(t, p_i) = \frac{f_{t,p_i}}{\sum_{p \in d} f_{t,p_i}} \qquad CF(t, C) = log \frac{|C|}{|p_i \in P : t \in P_i|} \qquad Weight(t, p, C) = GM(PF(t, p), CF(t, C))$$

p_i	the British-American rock band **Foreigner**.

→ Weights are calculated to figure out how important each token is.

p_i	<mask> British-American rock band **Foreigner**.

→ New passage generated by weight base deletion

Proposed Data Augmentation

Proposed Method

Pseudo Algorithm

Algorithm: Data augmentation considering importance of tokens

Require: Passage dataset $C = \{p_1, p_2, \dots, p_M\}$, Question dataset $Q = \{q_1, q_2, \dots, q_{|Q|}\}$, Tokenizer $\mathcal{T}_{BERT}$, counter

Ensure: Answer passages for given queries $C' = \{p'_1, p'_2, \dots, p'_{|C'|}\}$

Initialize dictionary cf count
  for $i = 1$ to $|C|$ **do**
    List of tokens: $T_i \leftarrow \mathcal{T}_{BERT}$.tokenize($p'_i$)
    for Token t in T_i:
      if t in cf count: cf count[t] += 1
      else: cf count[t] = 1
  for $j = 1$ to $|C|$ **do**
    weight ← $GM(T_i$.counter(t), cf count(t))
    normalized weight ← (weight of t − min weight of T_i)/(max weight of T_i – min weight of T_i) + epsilon
    probability ← normalized weight × scaling factor
    $C' \leftarrow$ deletion each token as acquired probability
    $C \leftarrow C + C'$
  end for
  return C

Pseudocode of the proposed method

Proposed Method

Details

- The inputs needed by the algorithm, which include the Passage dataset $C = \{p_1, p_2, \dots, p_M\}$, Question dataset $Q = \{q_1, q_2, \dots, q_{|Q|}\}$, Tokenizer $\mathcal{T}_{BERT}$, counter

Require: Passage dataset $C = \{p_1, p_2, \dots, p_M\}$, Question dataset $Q = \{q_1, q_2, \dots, q_{|Q|}\}$, Tokenizer $\mathcal{T}_{BERT}$, counter

- Answer passages for given queries $C' = \{p'_1, p'_2, \dots, p'_{|C'|}\}$ are augmented and defined as the output of the algorithm.

Ensure: Answer passages for given queries $C' = \{p'_1, p'_2, \dots, p'_{|C'|}\}$

- Initialize the dictionary named cf count before the start of the loops
- ***For* loop**: Iterates over each passages in corpus C'

Initialize dictionary cf count
for $i = 1$ to $|C'|$ **do**

Proposed Method

Details

- **List of tokens: $T_i \leftarrow \mathcal{T}_{BERT}$.tokenize($p_i'$)**: Uses the BERT tokenizer to convert the passages into list of tokens T_i

List of tokens: $T_i \leftarrow \mathcal{T}_{BERT}$.tokenize($p_i'$)

- For each token t in the token list T_i , if t is present in the cf count, increment the cf count value by 1. Conversely, if t is not present in the cf count, assign a value of 1. This procedure determines the cf count value for the entire passage

for Token t in T_i:
    if t in cf count: cf count[t] += 1
    else: cf count[t] = 1

- For each token, calculate its count within the token list to determine the Passage Frequency (PF). Subsequently, apply geometric mean to this TF value and the previously determined cf count to obtain the weight value

for $j = 1$ to $|C|$ **do**
    weight $\leftarrow GM(T_i.\text{counter}(t), \text{cf count}(t))$

Proposed Method

Details

- For each token, determine the minimum and maximum weights within the token list. Subtract the minimum weight from the weight of the token, subsequently divide the resulting value by the difference between the maximum and minimum weights, adding an infinitesimal epsilon value to prevent division by zero.

$$\text{normalized weight} \leftarrow (\text{weight of } t - \text{min weight of } T_i)/(\text{max weight of } T_i - \text{min weight of } T_i) + \text{epsilon}$$

- To convert min-max normalized values to probabilities, multiply by a scaling factor. Subsequently, proceed with the deletion process according to the probability assigned to each token

$$\text{probability} \leftarrow \text{normalized weight} \times \text{scaling factor}$$

deletion each token as acquired probability

- Update the corpus C with the generated passages and proceed with augmentation by combining the new passages with C'

$C' \leftarrow$ deletion each token as acquired probability

$C \leftarrow C + C'$

end for

return C

Experimental Settings

Dataset

- For NQ dataset, original train dataset was filtered because of mismatch between query and positive document. First column of train is the number of original and second column is filtered.
- The number of query for training is 58,880, also positive passage has the same. The number of entire passage is 11,736,475. It is split into train, dev, test.

Dataset	Train		Dev	Test
Natural Question	79,618	58,880	8,757	3,610

Hyperparameter Settings

- Batch size: 16
- Epochs : 10
- Loss function: Negative Log Likelihood Loss
- Optimizer : Adam
- Learning Rate: 2×10^{-5}

Evaluation Measures

- *Top-K Accuracy*

$$Top - K\ Accuracy = \frac{1}{N}\sum_{i=1}^{N} 1\{y_i \in \hat{y}_i^{(K)}\}$$

Experimental Results

Comparison

- The model trained with the proposed data augmentation method demonstrated better performance compared to the traditional random uniform probability deletion method or without data augmentation.
- Results demonstrate superior performance across all TOP-K accuracy metrics in NQ dataset, indicating the effectiveness of the proposed data augmentation method in information retrieval tasks.

Method	Top-10	Top-20	Top-30	Top-40
Proposed Method	**0.720**	**0.768**	**0.796**	**0.810**
DPR (base)	0.712	0.762	0.789	0.803
Uniform probability random deletion	0.708	0.761	0.786	0.802

Experimental Results

In-Depth Analysis

- The proposed method ensures that the essential information remains intact, which is crucial for accurate information retrieval. This demonstrates the advantage of considering token importance in the augmentation process.
- The proposed method does not simply remove stop words during the deletion process but also reflects the nature of the corpus. For example, if a corpus related to music is given, the weights of frequently occurring words related to music are increased.

Query	Who sang waiting for a girl like you
Original passage	Waiting for a Girl Like You "Waiting for a Girl Like **You**" is **a** 1981 power ballad by the British-American rock band **Foreigner**. The distinctive synthesizer theme was performed by the then-little-known Thomas Dolby, and this **song** also marked a major departure from their earlier singles because their previous singles were mid to upper tempo **rock** songs while this song was a softer love song with the energy of a power ballad. It was the second single released from the album "4" (1981) and was co-written by Lou Gramm and Mick Jones. It has become one **of** the band's most.
Uniform probability random deletion	Waiting for a Girl Like You "Waiting for a Girl Like **<mask>**" is a 1981 power ballad by the British-American rock band **<mask>**. The distinctive synthesizer theme was performed by the then-little-known Thomas Dolby, and this song also marked a major departure from their earlier singles because their previous singles were mid to upper tempo **<mask>** songs while this song was a softer love song with the energy of a power ballad. It was the second single released from the album "4" (1981) and was co-written by Lou Gramm and Mick Jones. It has become one of the band's most.
Proposed method	Waiting for a Girl Like You "Waiting for a Girl Like You" is **<mask>** 1981 power ballad by the British-American rock band Foreigner. The distinctive synthesizer theme was performed by the then-little-known Thomas Dolby, and this **<mask>** also marked a major departure from their earlier singles because their previous singles were mid to upper tempo rock songs while this song was a softer love song with the energy of a power ballad. It was the second single released from the album "4" (1981) and was co-written by Lou Gramm and Mick Jones. It has become one **<mask>** the band's most.

Conclusion and Contribution

- This thesis considered a novel data augmentation method for Dense Passage Retrieval (DPR), focusing on token importance to improve retrieval performance.
- The proposed method mitigates the issues of random token deletion by considering passage frequency and corpus frequency, leading to better preservation of semantic meaning.
- Experimental results demonstrate that proposed method outperforms traditional methods, showing improvements in top-K accuracy metrics across multiple datasets, confirming the effectiveness of the augmentation strategy.

Achievement

- Conferences
  - An Efficient Fine-Tuning of Generative Language Model for Aspect-Based Sentiment Analysis, ***ICCE***, 2024
  - Improving Dense Passage Retrieval using Weighted Passage Perturbation, ***IEIE***, 2024
- Training and Event
  - The European Summer School on Artificial Intelligence, NSCR, Athens, **Greece**, 2024
  - 2023 Singapore K-Tech Festival - AI/ML Innovation Research Center Booth, Singapore, **Singapore**, 2023

Thank you

Reference

[1] Robertson, S., & Zaragoza, H. (2009). The probabilistic relevance framework: BM25 and beyond. Foundations and Trends® in Information Retrieval, 3(4), 333-389.)

[2] Karpukhin, V., Oguz, B., Min, S., Lewis, P., Wu, L., Edunov, S., ... & Yih, W. T. (2020, November). Dense Passage Retrieval for Open-Domain Question Answering. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP) (pp. 6769-6781).)

[3] Kenton, J. D. M. W. C., & Toutanova, L. K. (2019). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In Proceedings of NAACL-HLT (pp. 4171-4186).

[4] Goceri, E. (2023). Medical image data augmentation: techniques, comparisons and interpretations. Artificial Intelligence Review, 56(11), 12561-12605.

[5] Wei, J., & Zou, K. (2019, November). EDA: Easy Data Augmentation Techniques for Boosting Performance on Text Classification Tasks. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP) (pp. 6382-6388).

[6] Baevski, A., Edunov, S., Liu, Y., Zettlemoyer, L., & Auli, M. (2019, November). Cloze-driven Pretraining of Self-attention Networks. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP) (pp. 5360-5369).

[7] Lee, K., Chang, M. W., & Toutanova, K. (2019, July). Latent Retrieval for Weakly Supervised Open Domain Question Answering. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (pp. 6086-6096).

[8] Chen, D., Fisch, A., Weston, J., & Bordes, A. (2017, July). Reading Wikipedia to Answer Open-Domain Questions. In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers) (pp. 1870-1879).

[9] Wang, Z., Ng, P., Ma, X., Nallapati, R., & Xiang, B. (2019, November). Multi-passage BERT: A Globally Normalized BERT Model for Open-domain Question Answering. In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP) (pp. 5878-5882).

[10] Kwiatkowski, T., Palomaki, J., Redfield, O., Collins, M., Parikh, A., Alberti, C., ... & Petrov, S. (2019). Natural questions: a benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, *7*, 453-466.