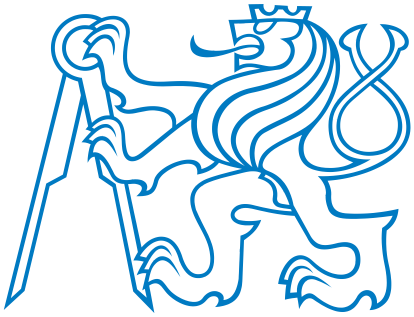
Czech Technical University in Prague

Faculty of Electrical Engineering
Department of Cybernetics
Czech Institute of Informatics, Robotics and Cybernetics



From FastText to Transformer Models, and their Application in Retrieval-Augmented Generation

Bachelor's Thesis

Yauheni Zviazdou

Prague, May 2024

Study programme: Open Informatics
Branch of study: Artificial Intelligence and Computer Science

Supervisor: Ing. Jan Šedivý, CSc.

Acknowledgments

Firstly, I would like to express my gratitude to my supervisor.

BACHELOR‘S THESIS ASSIGNMENT

I. Personal and study details

Student's name: **Zviazdou Yauheni** Personal ID number: **507333**

Faculty / Institute: **Faculty of Electrical Engineering**

Department / Institute: **Department of Cybernetics**

Study program: **Open Informatics**

Specialisation: **Artificial Intelligence and Computer Science**

II. Bachelor’s thesis details

Bachelor’s thesis title in English:

From FastText to Transformer Models, and their Application in Retrieval-Augmented Generation

Bachelor’s thesis title in Czech:

Od FastText k Transformer modelům a jejich aplikace v Retrieval-Augmented generování

Guidelines:

Review the representation of words, sentences, and paragraphs, progressing from traditional methods like FastText to advanced transformer-based models such as BERT. The primary focus is to evaluate selected representations using analogy tests and confusion matrix. Use the UPV corpus set for evaluation.
In the second part of the study, emphasis will shift towards selecting optimal representations for Retrieval-Augmented Generation (RAG) algorithms. The investigation will determine the most efficient embeddings and optimal text chunk size for question-answering tasks, particularly in the context of natural language answers generation from technical manuals. Conduct a comprehensive evaluation with a particular focus on suggesting an optimal representation model that balances factuality and CPU requirements.

Bibliography / sources:

[1] FastText documentation and tutorials, https://fasttext.cc/
[2] LangChain documentation, https://js.langchain.com/docs
[3] Word2Vec tutorials, https://www.tensorflow.org/text/tutorials/word2vec

Name and workplace of bachelor’s thesis supervisor:

Ing. Jan Šedivý, CSc. Big Data and Cloud Computing CIIRC

Name and workplace of second bachelor’s thesis supervisor or consultant:

Date of bachelor’s thesis assignment: **02.01.2024** Deadline for bachelor thesis submission: **24.05.2024**

Assignment valid until: **21.09.2025**

Ing. Jan Šedivý, CSc.	prof. Dr. Ing. Jan Kybic	prof. Mgr. Petr Páta, Ph.D.
Supervisor’s signature	Head of department’s signature	Dean’s signature

III. Assignment receipt

The student acknowledges that the bachelor’s thesis is an individual work. The student must produce his thesis without the assistance of others, with the exception of provided consultations. Within the bachelor’s thesis, the author must state the names of consultants and include a list of references.

. | Date of assignment receipt | Student’s signature

Declaration

I declare that presented work was developed independently, and that I have listed all sources of information used within, in accordance with the Methodical instructions for observing ethical principles in preparation of university theses.

Date ....

Abstract

This thesis investigates the application of word and sentence embeddings in Retrieval-Augmented Generation (RAG) for factual Question Answering (QA) tasks using technical manuals. The study explores the effectiveness of traditional FastText embeddings and advanced transformer-based models like Bidirectional Encoder Representations from Transformers (BERT) in capturing semantic relationships within text. We evaluate the quality of these representations using analogy tests and confusion matrix analysis on the UPV corpus set.

Subsequently, we will select optimal representations for RAG algorithms and assess their impact on factual accuracy and computational efficiency during QA. By analyzing the performance with different text chunk sizes, we aim to identify the optimal configuration for factual RAG in technical domains. This research contributes to the field of Natural Language Procession (NLP) by providing insights into selecting effective representations that balance factual accuracy and computational efficiency for QA systems.

Keywords NLP, Word Embedding, Transformers, RAG, QA, Semantic Textual Similarity (STS)

Abstrakt

Tato práce zkoumá aplikaci embeddingů slov a vět v modelu Retrieval-Augmented Generation (RAG) pro úlohy Question Answering (QA) zaměřené na fakta s využitím technických příruček. Studie se zabývá účinností tradičních embeddingů FastText a pokročilých modelů založených na transformátoru, jako je Bidirectional Encoder Representations from Transformers (BERT), při zachycování sémantických vztahů v textu. Kvalitu těchto reprezentací hodnotíme pomocí analogových testů a analýzy confusion matrix na korpusu UPV. Následně vybereme optimální reprezentace pro algoritmy RAG a posoudíme jejich vliv na faktickou přesnost a výpočetní efektivitu během QA. Analýzou výkonu s různými velikostmi textových fragmentů se snažíme identifikovat optimální konfiguraci pro faktické RAG v technických oborech. Výzkum přispívá k oblasti zpracování přirozeného jazyka (Natural Language Procession (NLP)) tím, že poskytuje poznatky o výběru efektivních reprezentací, které vyvažují faktickou přesnost a výpočetní efektivitu pro systémy QA.

Klíčová slova NLP, Word Embedding, Transformátory, RAG, QA, Sémantická podobnost textu

Abbreviations

RAG Retrieval-Augmented Generation

NLP Natural Language Procession

STS Semantic Textual Similarity

QA Question Answering

BERT Bidirectional Encoder Representations from Transformers

CBOW Continuous Bag-of-Words

GloVe Global vectors

ML Machine Learning

NN Neural Network

mBERT Multilingual Bidirectional Encoder Representations from Transformers

MTEB Massive Text Embedding Benchmark

Contents

1 Introduction

1.1 Text representation

The human language, with its nuances and complexities, presents a significant challenge for machines to understand. Natural Language Procession (NLP) bridges this gap, and at its core lies the critical concept of text representation. This process acts as a translator, bridging the gap between the richness of text and the numerical language that machines understand. By effectively capturing the meaning within words and their relationships, text representation empowers NLP models to leverage machine learning's capabilities. From sentiment analysis to machine translation, this ability to represent meaning fuels the advancements in NLP, enabling machines to interact with and decipher human language with ever-increasing accuracy.

1.2 Evolution of text representation methods

NLP has undergone a significant transformation in its approach to text representation. Early methods, such as one-hot encoding, while simple to implement, suffered from limitations in efficiency due to dimensionality and sparsity issues.

Word embedding techniques (e.g., Word2Vec, GloVe, FastText) offered a significant improvement by capturing semantic relationships between words through high-dimensional word vectors. However, these techniques primarily focused on local context within a limited window, hindering their ability to capture complex relationships within sentences or documents.

The emergence of deep learning architectures, particularly transformer-based models like Bidirectional Encoder Representations from Transformers (BERT), revolutionized the field of text representation. These models allows to not only understand the meaning of individual words but also consider their interaction and context within a sentence or document.

1.3 Research objective

This research aims to evaluate the effectiveness of various word, sentence, and paragraph representations for their subsequent application in Retrieval-Augmented Generation (RAG) algorithms, with a specific focus on the domain of technical Question Answering (QA).

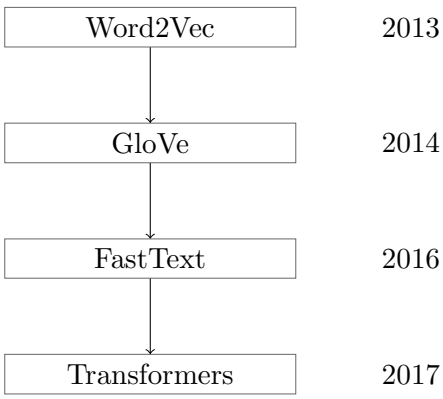


Figure 1.1: Evolution of the text representation methods.

2 Literature Review

2.1 Traditional word embedding methods

Word2Vec [3]

Word2Vec is algorithm that generates word embedding using information about target word (context). Word2Vec uses Neural Network (NN) and Machine Learning (ML) techniques to generate word embedding for every word in vocabulary during training. As NN architecture are used Continuous Bag-of-Words (CBOW) and Skip-gram, Fig. 2.1.

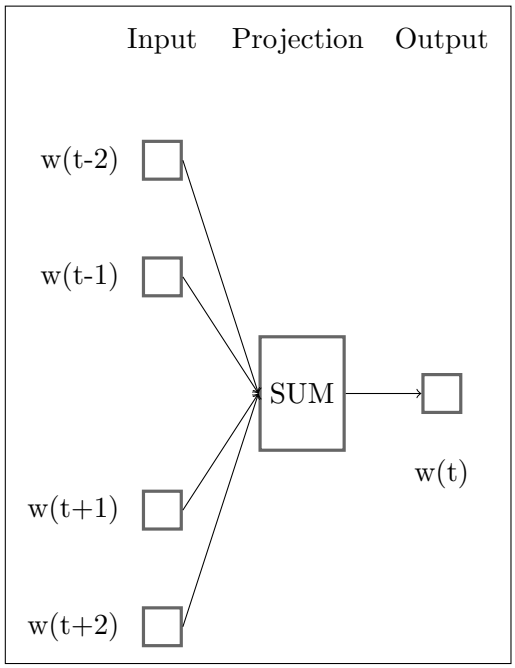


Figure 2.1: CBOW and Skip-gram schemes respectively

Due to its algorithmic simplicity and efficiency, Word2Vec has established itself as a strong baseline for numerous NLP tasks. Compared to more recent and complex models, Word2Vec requires minimal hyperparameter tuning, making it a relatively straightforward approach.

However, it is important to acknowledge that Word2Vec has limitations. These include its inability to capture **global information** within a document, its challenges in effectively handling **morphologically rich languages** (languages with many word variations), and its lack of awareness of the **broader context** beyond a limited window of surrounding words.

GloVe [2]

GloVe leverages the co-occurrence statistics of words within a corpus to learn vector representations. This approach involves constructing a co-occurrence matrix, where each entry reflects the frequency of two words appearing together within a predefined window size. This matrix essentially captures the relative importance of various word pairings.

A core principle of GloVe lies in the notion that word vectors should effectively encode the ratios between co-occurrence probabilities of words. By analyzing these ratios, GloVe can identify semantic relationships between words. This is achieved by factorizing the co-occurrence matrix into a lower-dimensional space, allowing for efficient representation and manipulation of word meanings.

To optimize the learned word embeddings, GloVe employs a weighted least squares objective function. This function aims to minimize the discrepancy between the dot product of two word vectors and the logarithm of their co-occurrence probability. Through iterative adjustments of the word vectors, GloVe converges on a solution that yields the desired word embeddings.

- **FastText** [1]

2.2 Transformer-based models

- Discuss traditional word embedding methods like FastText and their limitations.
- Explain the concept of transformer-based models like BERT and their advantages for text representation.
- Review related work on RAG algorithms and their dependence on effective text representations.Discuss existing research on evaluating text representations using analogy tests and confusion matrices.
- Briefly mention the UPV corpus set as the chosen evaluation benchmark.

3 Methodology

3.1 Evaluation process

This work specifically targets the evaluation of word, sentence, and paragraph representation methods on datasets in the Czech language. This focus on Czech allows for a deeper understanding of how these methods perform in a language with specific characteristics, such as a rich inflectional morphology and the presence of diacritics.

Text Data Preparation

In certain foreign languages, a common issue arises when individuals incorrectly write words by omitting diacritics or altering letters, Fig. 3.1. This problem is prevalent in social media, chatbots, and other informal written communications. As a result, embedding models face challenges in comprehending text without diacritics (hereinafter diacriticless). A potential solution involves adapting data representation to accommodate both formal and informal styles of writing.

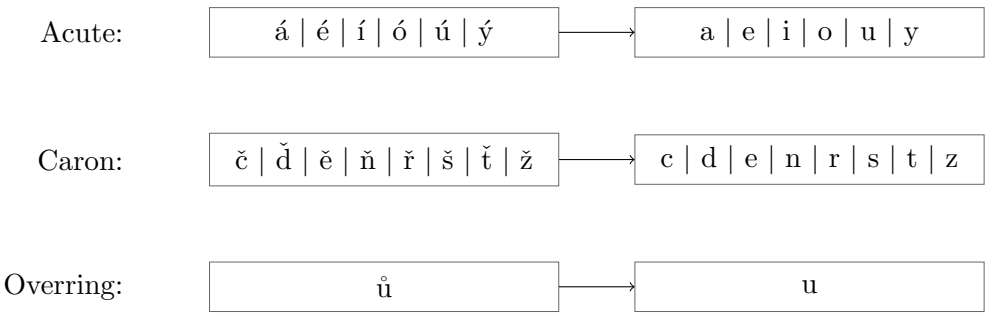


Figure 3.1: Usual changes in informal czech texts.

This study will employ two distinct text representations: text with diacritics and text without diacritics. To ensure optimal evaluation, the diacritic text will be assessed using datasets that preserve these diacritics, while the diacriticless text will be evaluated using datasets that lack diacritics.

As detailed in Lst. 3.1, this script is used for creating diacriticless versions of the datasets.

```
sed 's/.*/\L&/' "$1" | iconv -f utf-8 -t ascii//TRANSLIT > diacriticless/"$1"
```

Listing 3.1: Script for removing diacritics using Unix utilities

Datasets

UPV FAQ

Dataset is comprised of frequently asked questions (FAQs) and their answers from the Industrial Property Office of the Czech Republic website, organized into four distinct groups, as detailed in Table 3.1.

This work will evaluate two key metrics using UPV FAQ dataset

- **Question matching accuracy**: This metric involves calculating the cosine similarity between all possible question pairs within a dataset. A question is considered successfully matched if its second-highest cosine similarity score corresponds to another question belonging to the same class (i.e., the question with the highest similarity is likely the same question itself). The overall question matching accuracy is then computed as the ratio of successfully matched questions to the total number of question pairs evaluated.
- **Answer matching accuracy**: This metric assesses the system's ability to identify the correct answer for a given question. The system accomplishes this by directly comparing the question with pre-generated answer embeddings. By evaluating the similarity between the question and each answer embedding, the system classifies the question as corresponding to the answer with the highest similarity score. The overall answer matching accuracy is then calculated as the proportion of questions for which the system correctly identifies the corresponding answer.

Tests set	Number of tests
FAQv5	2054
FAQ50	561
FAQ76	2025
FAQ76v2	1965
All	6605

Table 3.1: Information about UPV FAQ tests

- **Baseline**

Due to the inherent morphological richness of the Czech language, this study adopts **FastText** as the baseline word embedding method. This decision is motivated by FastText's ability to effectively capture morphological variations within words, a characteristic that has been shown to be advantageous for languages like Czech. While other techniques like Word2Vec and GloVe have been explored for word embedding generation, they have demonstrated lower performance in this context.

Training parameters

- Architecture: CBOW
- Vector size: 300
- Loss function: Negative sampling loss
- Dictionary threshold (the frequency of the word to be included in the dictionary): 130

Training data

FastText model is trained on a segment of a preprocessed Common Crawl repository, which encompasses raw data from web pages, as well as metadata and text extractions. Given the nature of this dataset, it is expected to include misspellings and text lacking diacritics.

(i) Eliminating duplicate entries
(ii) Filtering out lines containing fewer than 9 characters

(iii) Excluding URLs
(iv) Breaking lines into individual sentences
(v) Converting all text to lowercase
(vi) Removing lines containing words exceeding 30 characters
(vii) Excluding HTML tags with less than 100 characters
(viii) Removing lines surpassing 500 characters
(ix) Omitting words longer than 21 characters

- **Chosen transformer models**

The evaluation process will leverage a curated selection of embedding models. This selection encompasses two categories:

(i) **Existing Czech Embedding Models**: To assess the performance of established solutions within the Czech NLP community, we will incorporate existing Czech embedding models into the evaluation.
(ii) **Models from Massive Text Embedding Benchmark (MTEB)**: We will also evaluate the effectiveness of all multilingual and highly regarded, open-source, English models available through the MTEB. This inclusion allows for a comparative analysis of how these models generalize to the Czech language.

mBERT

Multilingual Bidirectional Encoder Representations from Transformers (mBERT) is a BERT model, trained on a Wikipedia dump of 100 languages. The model performs best on high-resource languages such as English, French and Chinese, since lower-resource languages are underrepresented in the training data. We test whether M-BERT's pre-training on a wide range of languages, and thus a wide range of culture-specific analogies, might enhance the model's general analogy understanding.

CZERT

multilingual-e5

UAE-Large-V1

distiluse-base-multilingual-cased-v2

paraphrase-multilingual-MiniLM-L12-v2

paraphrase-multilingual-mpnet-base-v2

bge

gte

LaBSE

xlm-roberta

snowflake-arctic-embed

nomic-embed-text-v1.5

Seznam's models

- Describe the evaluation process for different text representations.
- - Specify the chosen word, sentence, and paragraph representation models (e.g., FastText, BERT variants).
  - Explain the usage of analogy tests and confusion matrices for evaluation.
  - Detail the selection process for the UPV corpus set and its suitability for technical QA tasks.
- Outline the second part of the study focusing on RAG for technical QA.
- - Explain the RAG algorithm and its reliance on text representations.
  - Describe the evaluation approach for selecting optimal representations for RAG.
  - Mention the factors considered during evaluation, such as embedding efficiency, text chunk size, and factuality of generated answers.

4 Experiments and Results

- Present the results of the evaluation for different text representations using analogy tests and confusion matrices.
- Discuss the findings regarding the effectiveness of each representation model for capturing semantic relationships in technical text.
- Analyze the results from the RAG evaluation, highlighting the impact of different representations and text chunk sizes on answer generation quality and CPU efficiency.
- Identify the representation model that achieves a balance between factuality of answers and computational demands.

5 Discussion

- Interpret the overall findings and their implications for choosing suitable text representations for RAG in technical QA tasks.
- Discuss the strengths and limitations of the chosen evaluation methods.
- Address potential challenges encountered during the study and suggest improvements for future research.

6 Conclusion

Summarize the achieved results. Can be similar as an abstract or an introduction, however, it should be written in past tense.

- Summarize the key takeaways from the research, emphasizing the most effective text representation model for RAG in technical QA based on the evaluation criteria.
- Briefly mention the trade-offs between factuality, CPU usage, and other factors in selecting representations for RAG.
- Suggest potential future research directions, such as exploring other text representation methods or evaluating RAG performance on different datasets.

7 References

[1] P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov, "Enriching word vectors with subword information," *arXiv preprint arXiv:1607.04606*, 2016.

[2] J. Pennington, R. Socher, and C. D. Manning, "Glove: Global vectors for word representation," in *Empirical Methods in Natural Language Processing (EMNLP)*, 2014, pp. 1532–1543. [Online]. Available: http://www.aclweb.org/anthology/D14-1162.

[3] T. Mikolov, K. Chen, G. Corrado, and J. Dean, *Efficient estimation of word representations in vector space*, 2013. arXiv: 1301.3781 [cs.CL].

A Appendix A