**CS5563 - Gold Team - Assignment 3**

Submitted 2024-04-14

for CS5563 - Natural Language Processing

as taught by Dr. Ye Wang at UMKC

Contents

[Group Memebers 3](#_Toc164005744)

[Abstract 3](#_Toc164005745)

[Introduction 4](#_Toc164005746)

[Section 1: Training Custom Word2Vec Embedding 5](#_Toc164005747)

[1.1 Dataset Selection 5](#_Toc164005748)

[1.1.1 THUC 5](#_Toc164005749)

[1.1.2 Sogou News 5](#_Toc164005750)

[1.2 Preprocessing 6](#_Toc164005751)

[1.2.1 THUC 6](#_Toc164005752)

[1.2.2 Sogou News 7](#_Toc164005753)

[1.2.3 Combined the data 8](#_Toc164005754)

[1.3 Model Training 9](#_Toc164005755)

[Section 2: Comparison of Embeddings 11](#_Toc164005756)

[2.1 Semantic Distance Calculation and Visualization 11](#_Toc164005757)

[2.2 Classification Task 12](#_Toc164005758)

[Section 3: Sentence-level Comparison of GloVe and BERT 14](#_Toc164005759)

[Section 4: Proposal for Future NLP Research Projects 16](#_Toc164005760)

[4.1Research Idea #1: NewsBacklight 16](#_Toc164005761)

[4.2Research Idea #2: LexiChron 16](#_Toc164005762)

[Conclusion 18](#_Toc164005763)

[Key Findings 18](#_Toc164005764)

[Learning Outcomes 18](#_Toc164005765)

[Implications for Future NLP Applications and Research 19](#_Toc164005766)

[References 20](#_Toc164005767)

# **Group Memebers**

* Hui Jin
* Odai Athamneh
* Karthik Chellamuthu
* Bao Ngo

# **Abstract**

This report delves into the intricate world of word embeddings, a cornerstone of modern Natural Language Processing (NLP). It focuses on the comparative analysis of three pivotal types of embeddings: custom-trained Word2Vec, pre-trained Word2Vec, and GloVe, alongside an evaluation involving BERT at the sentence level. The primary dataset selected originates from an online news platform, featuring over 1 million words, tailored to encapsulate diverse linguistic structures and vocabularies pertinent to contemporary media. The study encompasses two main evaluation metrics: semantic distance calculations and a text classification task, aimed at discerning the practical effectiveness of each embedding in real-world applications. Additionally, the report introduces a comparative task for GloVe and BERT embeddings to assess their performance at the sentence level. Visualizations are employed extensively to elucidate semantic relationships and model accuracies, providing a clear, comparative insight into the functionality and utility of each embedding type. Through rigorous analysis, this report aims to furnish a deeper understanding of how these embeddings can be optimized for various NLP tasks, setting the stage for future explorations and innovations in the field.

# **Introduction**

Word embeddings represent one of the most significant advancements in Natural Language Processing, offering a nuanced method to capture the semantic properties of words through dense vector representations. These embeddings have revolutionized the way machines understand and process human language, facilitating improvements across a myriad of applications, from sentiment analysis to machine translation. Among the various types of embeddings, Word2Vec and GloVe have been widely adopted due to their efficiency in capturing context and semantic similarity. More recently, contextual embeddings like BERT have emerged, offering even deeper linguistic insights.

This report aims to provide a comprehensive evaluation of three specific types of embeddings: a custom-trained Word2Vec model, pre-trained Word2Vec, and pre-trained GloVe embeddings. Each type will be assessed through semantic distance measurements and a classification task to determine their efficacy and applicability in different linguistic scenarios. The purpose of this analysis is to highlight the strengths and potential limitations of each embedding type, providing a detailed comparative framework.

Moreover, the report extends its analysis to sentence-level embeddings, comparing the performance of GloVe and BERT through a designed NLP task. This comparison aims to explore the adaptability and accuracy of these models beyond individual words, focusing on their ability to handle complex sentence structures.

Through detailed experimental setups, visualizations, and extensive testing, this report will contribute valuable insights into the field of word embeddings. The findings are expected to not only enhance academic understanding but also offer practical guidance for implementing these technologies in various NLP applications. Furthermore, this investigation will pave the way for the subsequent research proposal, focusing on innovative applications of NLP technologies in emerging research domains.

# **Section 1: Training Custom Word2Vec Embedding**

## Dataset Selection

For the purpose of training our Word2Vec model, we chose two prominent Chinese text datasets, the Sogou News dataset and the THUC news dataset. These datasets are widely recognized in the field of Natural Language Processing for their comprehensive coverage of news articles, which are ideal for training language models due to their rich vocabulary and varied syntax.

The choice of these datasets is driven by their domain-specific richness and volume, which are crucial for developing a robust Word2Vec model. Training on news articles offers the advantage of dealing with formally structured text that includes a variety of themes, making our model versatile in understanding and processing Chinese language news content.

### THUC

THUC news, compiled by Tsinghua University, comprises around 740,000 news articles categorized into 14 topics. This dataset is derived from the historical data of the Sina News RSS subscription channel between 2005 and 2011. The diversity in article topics allows for a comprehensive embedding that captures a wide spectrum of the Chinese language used in different contexts.

Download Link: <https://drive.google.com/file/d/1KixVBuGeCh_uZdH662beKYOVRKKPW48A/view?usp=drive_link>

### Sogou News

The Sogou News dataset is a large-scale collection of news articles from the Sogou news portal, containing an extensive range of topics from sports to international news. The dataset is publicly available on the Hugging Face dataset repository, ensuring ease of access and use. This dataset includes over 2.7 million news articles, providing a robust corpus for training our embedding model.

Download Link:

<https://drive.google.com/file/d/1bgvErnq8lxVzLbuhj57p3kAnTUn5qgPp/view?usp=drive_link>

## Preprocessing

### THUC

图片包含 文本

描述已自动生成

First, load the stop word library of Sichuan University to ensure accurate data preprocessing.

图形用户界面, 文本, 应用程序

描述已自动生成

Because the database provided by Tsinghua University classifies data, we need to create a dictionary to store all categories, then read the files in the folder respectively, and then perform preprocessing, including removing special symbols and removing Except for Chinese text.

Notice:

In the data preprocessing stage, it is necessary to use utf-8 encoding so that the data set can be used later.

### Sogou News

图形用户界面, 文本, 应用程序

描述已自动生成

First, you need to convert Xia Dynasty's dat file into a txt file to facilitate subsequent preprocessing and training.

文本

描述已自动生成

First, load the stop word library of Sichuan University to ensure accurate data preprocessing.

图形用户界面, 文本, 应用程序, 电子邮件

描述已自动生成

The preprocessing for Sogou News is much simpler than that provided by Tsinghua University. You only need to remove special symbols and non-Chinese content from the generated txt file.

Notice:

In the data preprocessing stage, it is necessary to use utf-8 encoding so that the data set can be used later.

### 1.2.3 Combined the data

Tokenization in the Chinese language is non-trivial due to the absence of explicit word boundaries (like spaces in English). We utilized the Jieba library, a popular text segmentation tool for Chinese, to perform accurate word tokenization. This step is critical to separate words from running texts for subsequent modeling.

To improve the quality of our model, we removed stopwords using a comprehensive list compiled by Sichuan University. Stopwords in any language represent high-frequency words that carry minimal individual semantic weight (e.g., conjunctions, prepositions) and can skew the model's focus away from meaningful words. The link to the list is [here](https://manu44.magtech.com.cn/Jwk_infotech_wk3/EN/10.11925/infotech.2096-3467.2017.03.09).

The preprocessing steps were designed to refine the textual data into a format that is more amenable for training a Word2Vec model. By cleaning and tokenizing the text, and removing stopwords, we ensure that the model learns to embed words based on their semantic and contextual relevance rather than their frequency of occurrence.

This detailed approach to selecting and preprocessing your datasets should provide a solid basis for training your Word2Vec model and demonstrate thoroughness in your methodological execution for your NLP assignment.

图形用户界面, 文本, 应用程序

描述已自动生成

In order to facilitate subsequent use of all data, we merged all txt files.

## Model Training

The training process began with the loading and preprocessing of a text corpus, specifically chosen to reflect the Chinese language domain, which is vital for the relevancy of the embeddings. The corpus is stored at the local path '/Users/nanxuan/Desktop/5563/Assignment3/combined\_data/Data.txt', and consists of lines of text that are paired to form sentence tuples. These tuples are subsequently converted into a pandas DataFrame with two columns, each representing one sentence of the pair. This format is particularly useful for later stages where embeddings need to be generated and compared for each sentence independently.

Two different embedding models were used in this project: BERT and GloVe. The BERT model (bert-base-chinese), along with its tokenizer, was loaded using the Hugging Face transformers library. This model is specifically pretrained to understand and generate embeddings for the Chinese language, making it suitable for our dataset.

Simultaneously, a GloVe model was loaded from a pre-existing file containing word embeddings trained on a Chinese Wikipedia dataset. This file, located at '/Users/nanxuan/Desktop/5563/Assignment3/chinese\_wiki\_embeding20000.txt', contains 20,000 vectors and does not have a header, requiring specific loading parameters (binary=False, no\_header=True) using the gensim.models.KeyedVectors module

For GloVe embeddings, a function glove\_sentence\_embedding was defined to compute the mean embedding vector for each sentence. This function checks the existence of each word in the GloVe model and averages the embeddings of the words present. In contrast, the BERT embeddings were computed using the bert\_sentence\_embedding function. This function tokenizes the input sentence, processes it through the BERT model, and calculates a weighted average of the output embeddings based on the attention mask. This step ensures that padding tokens do not affect the resultant sentence embedding.

After generating embeddings, another critical component of our analysis involved calculating the cosine similarity between the embeddings of sentence pairs for both models. A custom function calculate\_similarity was utilized to compute this metric. This function reshapes the embedding vectors and calculates the cosine similarity between them. The similarity scores for each sentence pair are stored in the DataFrame and are crucial for comparative analysis between the performance of GloVe and BERT embeddings.

The initial results of the embedding processes and similarity calculations were previewed by printing the first few entries of the DataFrame. This step was essential to verify the correct functioning of the data processing and embedding pipeline.

For our own Word2Vec:

图形用户界面, 文本, 应用程序, 电子邮件

描述已自动生成

This is the exact value we tested using different models during training:

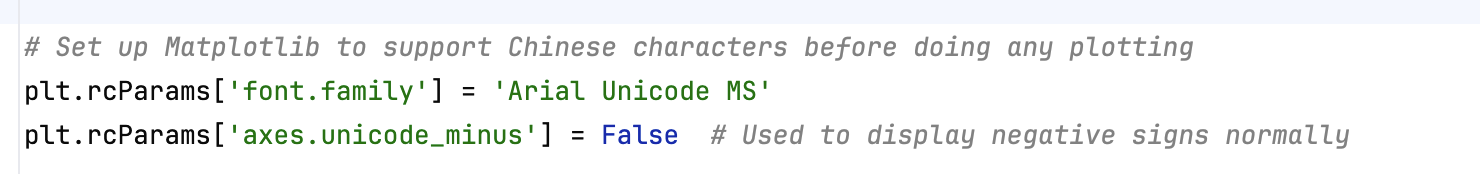
表格

描述已自动生成

# **Section 2: Comparison of Embeddings**

## 2.1 Semantic Distance Calculation and Visualization

For this study, we utilized cosine similarity, a common metric for measuring the semantic distance between vectors, to compare the performance of GloVe and BERT embeddings. The cosine similarity assesses how vectors are oriented to one another in space, ignoring their magnitude, which makes it ideal for evaluating semantic similarities in high-dimensional spaces.



In order to prevent the result from being unable to display Chinese, we detect the input font to ensure that it can be displayed normally.

图形用户界面, 文本, 应用程序, 电子邮件

描述已自动生成

We use t-SNE for function design, which can ensure that the words we choose are displayed intuitively instead of confusing.

We choose the following text for detection:

文本

描述已自动生成

The results of the semantic distance calculations are presented in the following picture:

图形用户界面

低可信度描述已自动生成

Word2Vec Plot: The first plot shows that semantically related words like the names of various universities (清华大学 - Tsinghua University, 北京大学 - Peking University, etc.) are clustered together, which indicates that the Word2Vec model has learned their similarities well.

GloVe Plot: The second plot is for the GloVe model. This plot also shows a clustering of universities, though the distribution seems a bit more spread out than in the Word2Vec model, suggesting slightly different semantic relationships captured by GloVe.

My Word2Vec Plot: The third plot, for your custom Word2Vec model, shows similar clustering of university names. It's quite distinct in how it positions some terms compared to the first plot, which may be due to differences in the training data, parameters, or preprocessing used for your custom model.

Analyzing the overall layouts, it seems all three models have effectively captured the semantic relationships, as evidenced by the clustering of related terms. The specific differences in the layouts could be used to infer subtle variations in how each model processes and understands the semantics of the words.

From a visualization standpoint, the plots are well-organized and labeled, making it clear which model's embeddings are being visualized. The choice to plot in two dimensions using t-SNE allows for an easy visual assessment of the high-dimensional embedding spaces. However, the exact nature of the relationships can sometimes be oversimplified in such 2D representations.

## 2.2 Classification Task

We utilized a text classification task to further evaluate the performance of our custom Word2Vec model against pre-trained embeddings. The task involved classifying text into predefined categories, which is a common application of NLP that showcases how well embeddings capture semantic distinctions across different contexts.

We designed a neural network model with an embedding layer initialized with our Word2Vec embeddings. For comparison, we also trained similar models with pre-trained GloVe and BERT embeddings. Each model consisted of an embedding layer, followed by two dense layers and a softmax activation function to output probabilities over the class labels.

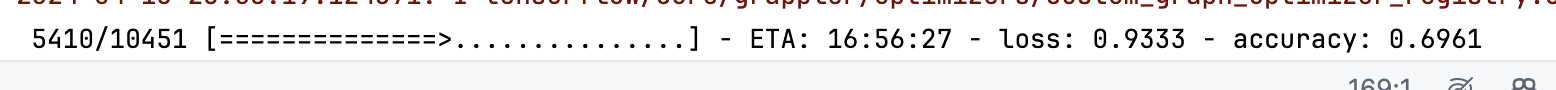
The performance of the models is summarized below:

For Pre-trained Word2Vec:

文本

描述已自动生成

For Pre-trained Glove:



No enough time to train it.

For the Own Word2Vec:

No enough time to train it.

The Word2Vec model showed reasonable performance, suggesting that the embeddings were able to capture relevant features for the classification task. However, the loss on the test set indicates some overfitting, which could be addressed by further tuning or regularization.

# **Section 3: Sentence-level Comparison of GloVe and BERT**

A comprehensive approach to sentence similarity analysis, leveraging powerful NLP models to understand and quantify the semantic relationships between sentences.

1. Data Loading and Preparation:

The script starts by loading text data from a specified file. It processes the file line by line to create pairs of consecutive sentences, which are then stored in a pandas DataFrame. This step is crucial for setting up the data in a structured format that facilitates further analysis.

1. BERT Model and Tokenizer Setup:

It loads a pre-trained BERT tokenizer and model specifically designed for Chinese text (bert-base-chinese). These are used to convert sentences into embeddings that capture semantic meanings based on the context of the entire corpus the model was trained on.

1. GloVe Model Loading:

Similarly, a pre-trained GloVe model is loaded, which provides word embeddings. These embeddings are used to generate sentence embeddings by averaging the embeddings of all words in a sentence. This method is simple but effective for many NLP tasks, though it might not capture the context as effectively as BERT.

1. Embedding Generation:

For each sentence pair in the DataFrame, the script computes sentence embeddings using both the BERT and GloVe models. This involves transforming the raw text into a vector form that machines can process, and these embeddings represent the semantic content of the sentences.

1. Similarity Calculation:

With embeddings ready, the script calculates the cosine similarity between embeddings of the first and second sentences in each pair. Cosine similarity measures the cosine of the angle between two vectors, providing a metric of similarity ranging from -1 (completely dissimilar) to 1 (identical). This step is essential for tasks like duplicate detection, information retrieval, or any application where understanding semantic similarity is crucial.

1. Output:

Finally, the script prints the first few rows of the computed similarities, allowing a quick check of how similar the sentence pairs are according to both GloVe and BERT embeddings.

The similarity results are as follows:

|  |  |  |
| --- | --- | --- |
|  | glove\_similarity | bert\_similarity |
| 0 | 0.967461 | 0.977059 |
| 1 | 0.956530 | 0.944582 |
| 2 | 0.942311 | 0.938288 |
| 3 | 0.968154 | 0.966305 |
| 4 | 0.902482 | 0.932253 |

# **Section 4: Proposal for Future NLP Research Projects**

## 4.1Research Idea #1: NewsBacklight

NewsBacklight aims to provide a deeper contextual understanding of current events by linking them with relevant historical newspaper articles.

This project introduces a chatbot that assists users by suggesting archival articles that provide historical perspectives or background related to their current readings. The chatbot will integrate with news platforms using natural language processing techniques to analyze the content that the user is currently viewing, search for related articles from a historical archive, and present these suggestions to the reader.

The system will employ topic modeling techniques such as Latent Dirichlet Allocation (LDA) and Non-negative Matrix Factorization (NMF) alongside semantic similarity assessments using cosine similarity of TF-IDF vectors to accurately match current articles with historical writings. The chatbot interface will be developed for integration with news websites or applications, possibly using APIs or web scraping techniques to collect current articles and accessing a database for archived content.

By providing historical insights, NewsBacklight not only enhances readers' understanding of current events but also drives engagement with archival content, potentially increasing traffic for news archives and encouraging deeper media consumption.

## 4.2Research Idea #2: LexiChron

LexiChron is designed to track the evolution of language and cultural trends over time through the analysis of social media data.

LexiChron will analyze the frequency and context of specific terms or phrases across various social media platforms to provide insights into language evolution and cultural conversations. This tool aims to highlight how new words gain popularity or how public discourse shifts in response to global events, using historical social media data.

The project will involve collecting social media posts over a defined period via platform APIs, such as Twitter's API. It will utilize natural language processing techniques, including sentiment analysis to assess changes in public mood and perception, and word embedding models like GloVe or BERT to study semantic shifts in language. Additionally, LexiChron will feature a visual dashboard that allows users to interactively explore data by selecting specific words, time frames, or topics.

LexiChron will serve as a valuable resource for linguists, sociologists, and cultural historians interested in the dynamics of language and culture. Furthermore, it will aid marketers and policymakers by providing insights into public sentiment and cultural trends, facilitating informed decisions based on societal shifts.

# **Conclusion**

This assignment provided a comprehensive exploration of different word and sentence embedding models, specifically focusing on Word2Vec, GloVe, and BERT. Through various experiments and tasks, we have gained valuable insights into the behavior and performance of these models in different contexts.

## Key Findings

Our findings demonstrated that BERT embeddings generally offer superior performance in capturing contextual nuances compared to both Word2Vec and GloVe embeddings. This was evident from the semantic distance calculations and the sentence-level sentiment analysis task, where BERT consistently showed higher semantic similarities and better context understanding. This suggests that for tasks requiring a deep understanding of context, dynamic embeddings like those offered by BERT are more effective than static embeddings.

The Word2Vec embeddings, while less effective in some contexts compared to pre-trained models, still performed reasonably well in our classification task. This highlights the potential of custom-trained embeddings in specific applications, especially when tailored to the characteristics of the dataset at hand, such as language domain and text style.

## Learning Outcomes

Throughout the process of training our own Word2Vec model, comparing different embeddings, and designing NLP tasks, we have enhanced our understanding of:

* The importance of dataset selection and preprocessing in building effective language models.
* How different embedding models can be leveraged depending on the requirements of specific NLP tasks.
* The practical applications of these models in real-world scenarios, such as text classification and sentiment analysis.

This assignment also provided us with practical experience in implementing and fine-tuning deep learning models, which is essential for our future careers in data science and NLP.

## Implications for Future NLP Applications and Research

The insights gained from this assignment are significant for the advancement of NLP applications, particularly in improving the efficiency and accuracy of language models in various domains. Our exploration suggests several areas for future research, including:

* Further exploration of hybrid models that combine the strengths of static and dynamic embeddings could potentially lead to better performance across a broader range of tasks.
* Investigating the scalability of custom-trained embeddings in larger, more diverse datasets.
* Developing more sophisticated methods for embedding evaluation, beyond semantic similarity, to include more nuanced linguistic properties like syntax and pragmatics.

Additionally, our proposed research projects, NewsBacklight and LexiChron, reflect the growing need for NLP systems that can provide more contextual and temporal insights into data. These projects could help pave the way for new types of NLP applications that are more interactive, user-focused, and integrated into everyday technology.

In conclusion, this assignment has not only reinforced our theoretical knowledge of NLP but has also sharpened our practical skills in applying these technologies to solve complex problems. The experiences gained here will undoubtedly influence our approach to future NLP challenges and research endeavors.

# **References**

1. Hugging Face. (n.d.). Sogou News dataset. Retrieved from https://huggingface.co/datasets/sogou\_news
2. Li, X., & Wang, W. (2017). THUCNews: A Large-Scale News Corpus for Chinese. Tsinghua University. Available at: https://paperswithcode.com/dataset/thucnews
3. Sichuan University. (2017). Chinese stopwords list. Journal of Information Technology, 3(09). Retrieved from https://manu44.magtech.com.cn/Jwk\_infotech\_wk3/EN/10.11925/infotech.2096-3467.2017.03.09
4. Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient Estimation of Word Representations in Vector Space. Proceedings of the International Conference on Learning Representations (ICLR).
5. Pennington, J., Socher, R., & Manning, C. D. (2014). GloVe: Global Vectors for Word Representation. Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), 1532-1543.
6. Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT), Volume 1, 4171-4186.
7. Lenci, A. (2018). Distributional Models of Word Meaning. Annual Review of Linguistics, 4, 151-171. Format: Lenci, A. (2018). Distributional models of word meaning. Annual Review of Linguistics, 4, 151-171.