Thesis No:

CSE 4000: Thesis/Project

CROSS-DOMAIN SENTIMENT ANALYSIS USING WORD EMBEDDINGS for BANGLA TEXT

By

Proloy Karmakar

Roll: 1907051

Department of Computer Science and Engineering

Khulna University of Engineering & Technology

Khulna 9203, Bangladesh

October, 2024

Cross-domain sentiment analysis using word embeddings for bangla text.

By

Proloy Karmakar

Roll: 1907051

A thesis pre-defense report submitted in partial fulfillment of the requirements for the degree of

"Bachelor of Science in Computer Science and Engineering"

Supervisor:

Dr. K. M. Azharul Hasan

Professor

Dept. of Computer Science and Engineering

Khulna University of Engineering & Technology

_____

Signature

Department of Computer Science and Engineering

Khulna University of Engineering & Technology

Khulna 9203, Bangladesh

October, 2024

Acknowledgement

First and foremost, I thank God, the Almighty, who created us and will always be with us. I am completing my thesis with the mercy of the Almighty. I am grateful to **Prof. Dr. K. M Azharul Hasan**, Department of Computer Science and Engineering, Khulna University of Engineering and Technology, for guiding and directing me through the proceeding of this thesis work. He made materials and tools available to me that will aid in the improvement of my thesis work. He is a well-behaved, well-disciplined individual who is assisting me in becoming more responsible.

Next, I want to express my gratitude to all of my class teachers who have shared their knowledge with us, which has aided us in completing our thesis. Also, a big thank you to my classmates for sharing their knowledge and always giving support, which helped us finish this thesis.

Authors

Abstract

Cross-domain sentiment analysis, a challenging task due to the variation in language usage and sentiment expression across different domains, is addressed in this research by leveraging advanced word embeddings. Traditional sentiment analysis models often struggle to capture the subtle differences in sentiment across diverse fields, such as product reviews, movie critiques, or financial reports. The proposed method enhances sentiment prediction across various domains by enriching word vectors with emotional tone and domain relevance. It captures subtle sentiment variations and improves performance in unseen domains where traditional models struggle. This work contributes to natural language processing by offering a flexible solution for cross-domain sentiment analysis, enabling more accurate sentiment predictions across a wide range of real-world applications.

Contents

List of Figures

List of Tables

CHAPTER I

Introduction

1.1 Background

Sentiment analysis is significant in fields like e-commerce, social media analysis, and marketing and is emerging as a prominent subject in Natural Language Processing (NLP) research. The advancement of deep learning techniques has greatly contributed to various NLP tasks, including text sentiment analysis. Within this domain, some researchers are dedicated to constructing sentiment analysis models based on neural networks to predict the emotional tone of textual content effectively.

A fundamental perspective frames review sentiment analysis as a classification challenge. In this scenario, distinctive attributes are extracted from review texts, serving as inputs to the classifier, while the anticipated sentiment categories form the corresponding outputs. A prevalent approach involves the utilization of pre-trained universal word vectors as representations for the review text's features. These vectors are then further refined during the training phase of the model [1]

Generic word vectors are created through unsupervised methods on extensive corpus datasets, utilizing contextual information. Numerous studies have demonstrated that these generic word vectors can effectively capture the semantic nuances embedded in words [2]. These word vectors are used later for sentiment analysis like review sentiment analysis.

1.2 Motivation

The motivation for conducting review sentiment analysis arises from the increasing importance of understanding and harnessing the sentiments and opinions expressed in textual reviews, feedback, and user-generated content. The review sentiment analysis provides a data-driven approach to gain insights into the collective sentiments, opinions, and emotions of customers and users. This information has far-reaching implications for businesses, marketing strategies, product development, and academic research. Review sentiment analysis in Natural Language Processing (NLP) is the process of using computational techniques to automatically determine the sentiment or emotional tone expressed in textual reviews, feedback, and user-generated content. This analysis involves classifying the sentiment of the text as positive, negative, or neutral, and sometimes even identifying specific emotions associated with the text. Recently, word embedding techniques have been improved and these low-dimensional word representations have a great impact on the sentiment analysis process. Word embedding is a powerful technique that can be used to improve the performance of review sentiment analysis. By learning the meaning of words in context, identifying words that are related to sentiment, and measuring the similarity between words, word embedding can help to accurately classify the sentiment of a text.

1.3 Problem Statement

Nonetheless, utilizing universal word vectors directly for sentiment classification tasks presents inherent limitations. Certain words, like ”good” or ”bad,” possess robust sentiment inclinations and share comparable contextual patterns within the training corpus. Consequently, the conventional word vector approach generates akin word representations for terms harboring opposing sentiments. Should these word vectors serve as input features for classification models, the model's accuracy would inevitably suffer due to inadequately captured features. The difference between general word embedding and the sentiment aware word embedding has been shown in Figure 3.1

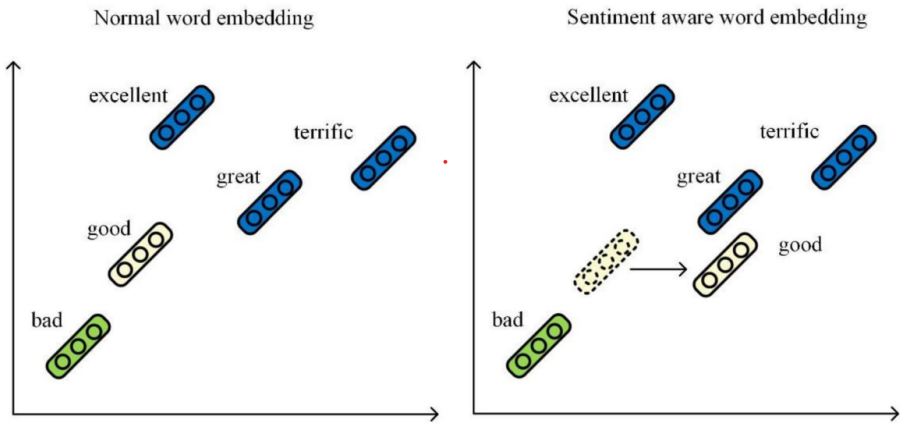


Figure 1.1: Comparison between normal and sentiment-aware word embeddings

Furthermore, the interpretation of specific sentiment-laden words hinges on the contextual domain they inhabit. For instance, consider the term ”lightweight.” An evaluation of an electronic product conveys a sense of positivity by indicating that the product is more portable and user-friendly. However, when applied within the context of a film review, the same word portrays a negative sentiment, suggesting a lack of depth in the movie's plot that fails to engage the audience. Another illustrative case involves the word ”simple.” While this term typically evokes a positive sentiment when describing an electronic product's ease of use, its connotation turns negative when employed to characterize a movie, hinting at an absence of complexity that contributes to a disappointing viewing experience. Universal word embedding models fail to incorporate the domain and sentiment information at a time in the word vector representation.

1.4 Objectives

The objective of this thesis is to implement a method for incorporating the sentiment information and the domain information. So, in short, our thesis objectives are:

- To implement a word embedding model that includes both sentiment information and domain-specific information.
- To improve the performance of sentiment classification for review.
- To establish a model for performing cross-domain sentiment analysis tasks for the Bangla language.

1.5 Scope

- **Word2vec:** Word2vec is a family of two related models that are used to produce word embeddings. These models are shallow, two-layer neural networks that are trained to reconstruct linguistic contexts of words. Word2vec takes as its input a large corpus of text and produces a vector space, typically of several hundred dimensions, with each unique word in the corpus being assigned a corresponding vector in the space. The two models in the Word2vec family are:
  - **Continuous Bag-of-Words (CBOW):** The CBOW model predicts the current word given its context words. For example, the CBOW model might be trained on the sentence ”The cat sat on the mat.” The context words for ”cat” would be ”The” and ”on the mat.” The CBOW model would then learn to associate the vector for ”cat” with the vectors for ”The” and ”on the mat.”

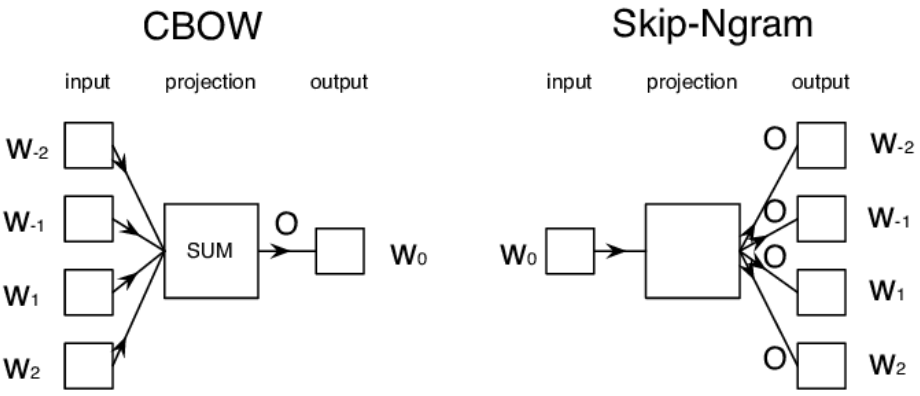


Figure 1.2: CBoW and Skip-gram model

  - **Skip-gram:** The skip-gram model predicts the context words given the current word. For example, the skip-gram model might be trained on the sentence ”The cat sat on the mat.” The current word for the skip-gram model would be ”cat.” The skip-gram model would then learn to associate the vectors for ”The”, ”on”, and ”mat” with the vector for ”cat.”

However, these models fail to capture the actual context in the text given. For example:

Sentence 1: I drink water **Sentence 2: I water the plants**

In sentence 1 water is a noun whereas in sentence 2 water is a verb. But the traditional

word embedding models consider these two **water** as the same and so give the same embeddings for these two **water**.

There comes the ELMo model in the picture which treats the two **waters** in the given example as different. As a result, the embeddings of the noun **water** and the verb **water** are also different.

- **ELMo:** ELMo stands for Embeddings from Language Models. It provides **contextualized word representations**, meaning the embedding of a word changes based on its surrounding words. Unlike Word2Vec, which gives a single static vector for each word regardless of context, ELMo captures both semantic meaning and syntactic roles, making it better at handling word ambiguity and polysemy. Additionally, ELMo's bidirectional nature enables it to understand both preceding and following context, allowing for more accurate sentiment prediction, especially in complex sentences where word meaning can shift depending on the sentence structure. This makes ELMo highly suitable for tasks like sentiment analysis, where context plays a crucial role in determining the polarity of a word.

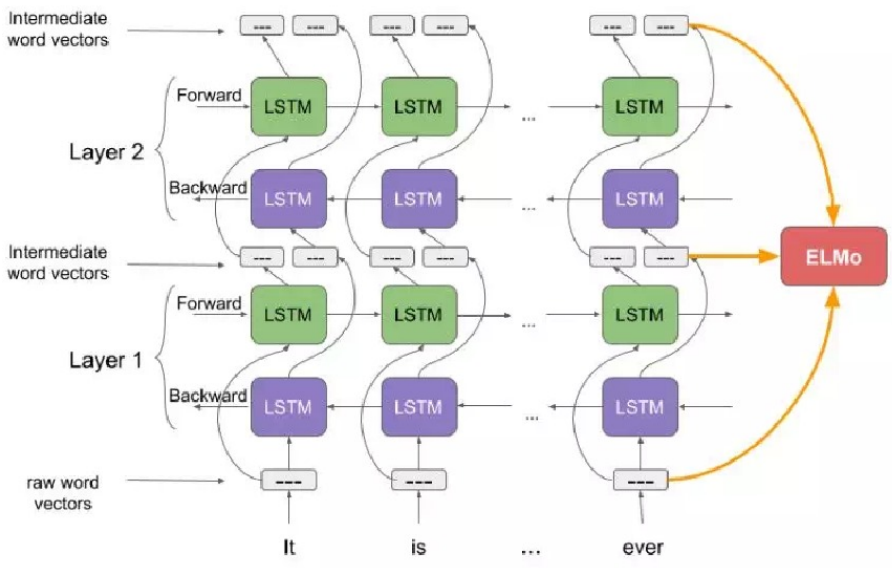


Figure 1.3: ELMo model

- **AdaGrad Optimizer:** Adagrad is an adaptive learning rate optimization algorithm. It is a type of stochastic gradient descent (SGD) that automatically adapts the learning rate for each parameter during training. The learning rate is adjusted based on how

frequently each parameter is updated. Parameters that are updated more frequently are given a smaller learning rate, while parameters that are updated less frequently are given a larger learning rate.

- **Pytorch:** Pytorch is an open-source software library for machine learning and artificial intelligence. It can be used across a range of tasks but has a particular focus on training and inference of deep neural networks.TensorFlow is a powerful tool for machine learning, but it can be complex to learn. It is important to have a good understanding of machine learning concepts before using Pytorch.

1.6 Unfamiliarity of the Problem/ Topic/Solution

In recent years, the field of Natural Language Processing (NLP) has witnessed significant advancements, particularly in the realm of sentiment analysis. One of the pressing challenges in this domain is the development of effective techniques for cross-domain sentiment-aware word embeddings. This entails the creation of word representations that not only capture semantic meanings but also account for sentiment variations across different domains. The correct classification of customer reviews, for instance, is a critical concern, as misclassification can have substantial consequences. To address this issue, researchers are diligently working on refining existing approaches. These approaches either emphasize the incorporation of sentiment information related to individual words or focus on handling domain-specific relevance. The complexity of cross-domain sentiment-aware word embeddings is examined in depth in this thesis, along with the developments that have been made, the difficulties that still need to be solved, and the prospective effects of improving sentiment analysis across other domains.

1.7 Project Planning:

The topic was finalized earlier this year, around March. The initial stages involve reading and interpreting relevant papers, establishing a foundational understanding of the concepts, acquiring fundamental theoretical knowledge, and grasping the basics of machine learning and deep learning by August. The primary focus will be on conducting the tasks using the English language initially. Subsequently, the same tasks will be replicated using Bangla text,

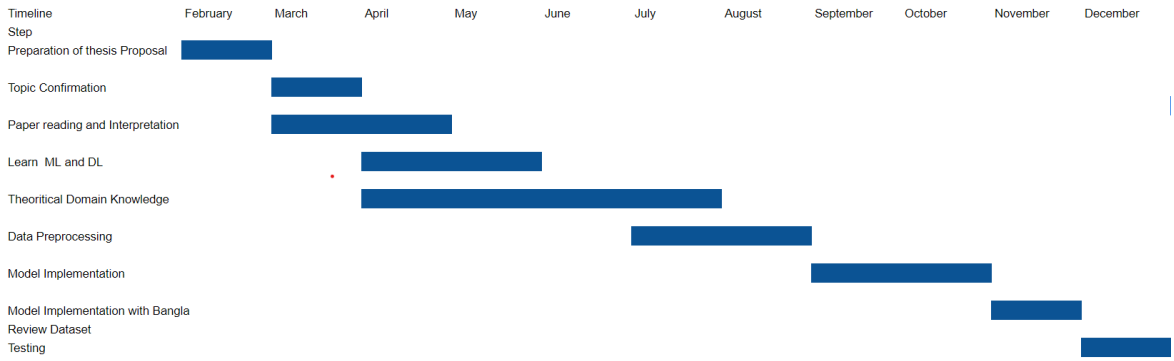


Figure 1.4: Gantt Chart

to complete this phase by December. If the work can be accomplished, this will help the e-commerce sector in Bangladesh significantly by helping the user choose the right product and by gaining their trust in e-commerce. 1.4

1.8 Applications of the work

Cross-domain sentiment-aware word embeddings are a type of word embedding that can be used to capture the sentiment of words in different domains. This can be useful for a variety of applications, such as:

- **Review Sentiment analysis:** Cross-domain sentiment analysis using word embeddings can be used to improve the performance of sentiment analysis models that are trained on data from multiple domains. This is because the embeddings can capture the different ways that sentiment is expressed in different domains.
- **Opinion mining:** Cross-domain sentiment analysis using word embeddings can be used to extract opinions from the text in different domains. This can be useful for tasks such as product reviews and customer feedback.
- **Fake news detection:** Cross-domain sentiment analysis using word embeddings can be used to detect fake news articles. This is because fake news articles often use different words and phrases to express sentiment than real news articles.
- **Social media monitoring:** Cross-domain sentiment analysis using word embeddings

can be used to monitor social media for sentiment about different topics. This can be useful for businesses to understand how customers are feeling about their products or services.

- **Market research:** Cross-domain sentiment analysis using word embeddings can be used to conduct market research by understanding the sentiment of customers about different products or services.

1.9 Organization of the Thesis

The rest of the thesis is organized as follows:

Chapter I provides an introduction to the background, problem statement, objectives, and organization of the thesis.

Chapter II presents short summaries of the related research works on word embeddings and review sentiment analysis.

Chapter III presents the methodology related to implement the work.

Chapter IV explains the dataset and work progress.

Chapter V discusses limitations, future works, and conclusion.

CHAPTER II

Literature Review

2.1 Literature Review

Neural network-based deep learning methods revolutionized feature learning by extracting insights from input data instead of relying on handcrafted features [3]. Initially utilized in computer vision, deep learning has gained traction in NLP, particularly for text modeling and sentiment analysis [4–7]. Various neural network-based models have emerged to enhance text features. These models show that distributed word representations are highly effective across NLP tasks like parsing, language modeling, and sentiment analysis [8, 9]. The early neural network language model (NNLM) focused on language modeling, sometimes neglecting word vectors, and faced time-consuming training due to its complex architecture. To overcome this, shallow neural network structures were proposed to learn word embeddings efficiently. Unlike NNLM, these models aim to create low-dimensional word vectors using contextual information and preserving semantics [2]. Word2Vec, based on this approach, addresses the limitations of traditional bag-of-words (BoW) methods. GloVe combines global co-occurrence matrix decomposition and local context window methods for improved word representations [10]. However, existing word vector models primarily capture semantic and syntactic context, neglecting sentiment, leading to challenges in sentiment classification. To address this limitation, researchers proposed the sentiment-specific word embedding (SSWE) model, which combines semantic and sentiment information within word vectors [8, 11]. Another approach involves neural networks controlling emotions at document and word levels to train emotion-rich embeddings [12], while a refinement strategy aims for joint semantic-emotion word vectors [13]. Moreover, recognizing words' shifting meanings across domains, an unsupervised method captures domain-related word expressions, enabling feature learning for different domains [14, 15]. About [16], a method is introduced that employs canonical correlation analysis (CCA) in conjunction with both generic and domain-specific word embeddings. This amalgamation aims to extract domain-

specific semantic insights, and the efficacy of the generated word embeddings is gauged through tasks involving sentence classification. Similarly, in [17], a technique enhances the fundamental skip-gram model by introducing regularization and training across various sections of a corpus, culminating in the acquisition of embeddings that span multiple domains. Collectively, these approaches center on the alteration of word semantics in diverse domains. While [18] shows a provisioned approach for sentiment analysis using the ELMo model. Also indicates how the ELMo model creates contextualized word embeddings for the given text.

2.2 Discussion of research gap solution

Current strategies for enhancing word embeddings can be categorized into two groups. The first involves integrating sentiment information into word embeddings, while the second focuses on embedding domain-specific information. However, these prior approaches have not concurrently harnessed both sentiment and domain information.

The is no research available in the area of ”Cross-Domain Sentiment Analysis for Bangla Language using ELMo Model”. However, the CBOW model, the Glove model, and the Skip-Gram model are used to do this sentiment analysis task. However, the problem is these models are not context-dependent. So, they produce a fixed embedding for a particular word. For example: the word ”Present” will get the same embedding for both situations when it represents a ”birthday gift” as well as ”now (current situation)”. So, it is clear that these models can’t be able to capture the actual context of the text.

I am working with the ELMo model for CDSA for Bangla Text for the first time. The biggest advantage of the ELMo model is that it is context-dependent. And so, it can capture the actual context of the text. Details of the ELMo model are described in the ”Implementation” and ”Scope” sections.

CHAPTER III

Methodology

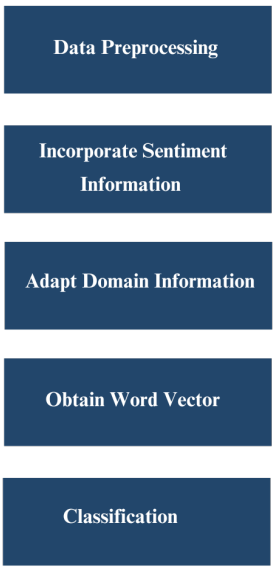
3.1 Introduction

In this research, the main target is to build a learning model that can take into account higher-level information such as sentiment information and domain relevance information. The methodology is described in [19].

The objective of this chapter is to provide a detailed description of the methodology and techniques that have been used, and how they will be applied in this research.

3.2 A General Framework

The process of cross-domain sentiment analysis using word embeddings is divided into five basic steps, as shown in Fig. 3.1



07

Figure 3.1: An Overview

- **Stage 1 (Data Pre-processing) :**The goal of data pre-processing includes collecting the data, text cleaning techniques, adding stopping words, and basic feature extraction. The context words along with the target word are paired.
- **Stage 2 (Incorporate Sentiment Information) :**This stage incorporates contextual details during the word vector generation phase. It enables the resulting word vectors to encapsulate the inherent sentiment characteristics present within the words.
- **Stage 3 (Adapt Domain Information) :** The semantic evolution of specific sentiment words across various domains is considered in this stage. This is achieved by evaluating the frequency of these words in two domains, allowing us to determine their significance within each domain.
- **Stage 4 (Obtain Word Vector and Sentiment Classification) :** The learned word embedding is used for sentiment classification and the obtained result will be checked against methods using a generic word embedding model.

3.3 Exploit The Sentiment Information

The sentiment incorporation is done by adding a sentiment module to the existing ELMo module. The hybrid model($CDSAWE_s$) combines the ELMo model with a sentiment prediction module. The ELMo model is used to learn the semantic and syntactic information of words, while the sentiment prediction module is used to learn the sentiment information of words. This hybrid model has two components:

- **The semantic module:** This module learns the semantic and syntactic information of words by predicting the center word from the context words.
- **The sentiment module:** This module learns the sentiment information of words by predicting the sentiment polarity of the context.

In Figure 3.2, the overview of the process is given: The model comprises two modules. The semantic module resembles the Embeddings from Language Models (ELMo) model: In the ELMo model, input sentences are tokenized and indexed into word representations. These word indices are passed into an embedding layer that generates context-sensitive word vectors by considering both preceding and following words. The hidden layers of the ELMo

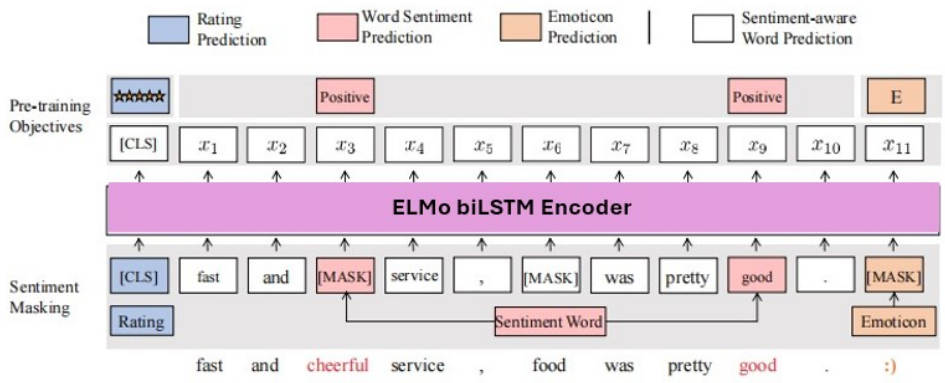


Figure 3.2: A diagram of the $CDSAWE_s$ model

model utilize bidirectional LSTMs to capture deep contextualized representations, ensuring that the word vectors reflect the semantic nuances based on their position in the sentence. The output layer computes probabilities for each word in the vocabulary, predicting the next word or reconstructing masked words based on the learned context. Simultaneously, a sentiment analysis module can take these contextualized word representations as input, using them to extract sentiment-related features from the sentence. These features influence the embedding computations and help classify the overall sentiment of the context. The model's predictions align with sentiment categories, with the updated word vectors reflecting the context's semantic meaning and emotional tone. The definition of the corrupted word vector $\tilde{x}$is provided in equation 3.1.

$$\tilde{x} = \begin{cases} 0, & \text{with probability } p \\ \frac{x}{1-p}, & \text{otherwise} \end{cases} \tag{3.1}$$

The feature of context after the corruption operation $\tilde{X}$ can be calculated using 3.2

$$\tilde{X} = \frac{1}{N} \sum_{i=1}^{N} \tilde{x}_i \tag{3.2}$$

The probability of predicting the sentiment-aware word by the ELMo model based on the context is calculated using 3.3 where $\hat{x}_i$ is the corrupted word of a sentence and x_i is the predicted word at the position i, from V which refers to the Vocab, the vocabulary of the basic ELMo model.

$$P(x_i \mid \hat{x}_i) = \text{Softmax}(W_w \cdot h_i + b_w) \tag{3.3}$$

Here W_w is the weight vector and b_w is the bias.

The probability of predicting the sentiment of the word by the ELMo model based on the context is calculated using 3.4 where $\hat{x}_i$ is the corrupted word of a sentence and s_i is the predicted sentiment at the position i by the basic ELMo model.

$$P(s_i \mid \hat{x}_i) = \text{Softmax}(W_s \cdot h_i + b_s) \tag{3.4}$$

Here W_s is the weight vector and b_s is the bias.

The above tasks focus on learning the token-level sentiment knowledge. Ratings represent the overall sentiment score of the reviews at the sentence level. Inferring the rating will bring in sentence-level sentiment knowledge. The rating is predicted using 3.5 where $\hat{x}_i$ is the corrupted word of a sentence and r is the predicted sentiment at position i by the basic ELMo model.

$$P(r \mid \hat{x}_i) = \text{Softmax}(W_r \cdot h_{[CLS]} + b_r) \tag{3.5}$$

Here W_r is the weight vector and b_r is the bias.

To express predictive emotions using context-semantic representation $\tilde{X}$, a sentiment classification model must be constructed, as shown by the sentiment module in Fig. 3.2. The general cross-entropy loss function model prediction of the sentiment component is as follows:

$$\mathcal{L} = -\frac{1}{|\hat{X}|} \sum_{\hat{x} \in \hat{X}} \frac{1}{|\hat{x}|} \sum_{i=1}^{|\hat{x}|} \log(Prob) \tag{3.6}$$

However, the cross-entropy loss for the sentiment word prediction can be done by the following equation:

$$\mathcal{L}_w = -\frac{1}{|\hat{X}|}\sum_{\hat{x}\in\hat{X}}\frac{1}{|\hat{x}|}\sum_{i=1}^{|\hat{x}|}\log\left(P(x_i \mid \hat{x}_i)\right) \tag{3.7}$$

Where $P(x_i \mid \hat{x}_i)$ defines the probability of predicting the sentiment word at position i. This value varies from 0 to 1. The closer the value is to 1, the better the prediction of the masked sentiment word, And so, the summation of logarithms of all the masked word predictions would be as minimal as possible.

Whereas, the cross-entropy loss for the prediction of masked word sentiment can be done by the following equation:

$$\mathcal{L}_s = -\frac{1}{|\hat{X}|}\sum_{\hat{x}\in\hat{X}}\frac{1}{|\hat{x}|}\sum_{i=1}^{|\hat{x}|}\log\left(P(s_i \mid \hat{x}_i)\right) \tag{3.8}$$

Where $P(s \mid \hat{x}_i)$ defines the probability of predicting the sentiment word at position i. This value varies from 0 to 1. The closer the value is to 1, the better the prediction of the masked sentiment word, And so, the summation of logarithms of all the masked word predictions would be as minimal as possible.

And lastly, the cross-entropy loss for the rating prediction can be done using the following equation:

$$\mathcal{L}_r = -\frac{1}{|\hat{X}|}\sum_{\hat{x}\in\hat{X}}\log\left(P(r \mid \hat{x})\right) \tag{3.9}$$

Where $P(r \mid \hat{x})$ defines the probability of predicting the rating at position i. This value varies from 0 to 1. The closer the value is to 1, the better the prediction of the masked sentiment word, And so, the summation of logarithms of all the masked word predictions would be as minimal as possible.

The total cross-entropy loss is then calculated using the following equation through which the model learns:

$$\mathcal{L}_t = \mathcal{L}_w + \mathcal{L}_s + \mathcal{L}_r \tag{3.10}$$

The process of acquiring sentiment-aware ELMo embeddings involves combining both semantic and sentiment components within the model's loss function. This is regulated by a hyperparameter, denoted as $beta$, which controls the importance of sentiment information. The value of $beta$ ranges between 0 and 1. The final loss function of the model incorporates both the prediction of masked words and sentiment learning, as illustrated in Equation (7). Notably, when $beta$ is set to 0, the model reduces to the traditional ELMo model, focusing solely on capturing semantic and syntactic information from context.

$$loss_s = \beta.loss_{semantic} + (1 - \beta)loss_{predict} \tag{3.11}$$

Ultimately, by employing a gradient descent-based optimization algorithm, the error can be propagated backward through the entire network. This process allows for the refinement of network parameters, leading to the acquisition of an optimized word embedding matrix through parameter updates.

3.4 Extract The Domain Information

After enhancing the basic ELMo model to include sentiment information, the model is further extended to include domain relevance information. The extended model learns cross-domain word vectors for words that appear in both domains while ignoring words that appear in one domain. The goal is to train cross-domain word vectors related to domain Q based on domain P-trained word vectors. The equation 3.12 is used to learn the word vector of each word in domain Q.

$$loss_Q = loss_P + \sum_{w \in P \cap Q} t_w.dist(w_p - w_q) \tag{3.12}$$

The t_w definition is shown in equation 3.13.

$$t_w = \sigma(\lambda.\phi(w)) \tag{3.13}$$

$\phi(w)$ is generated when domain adaptations occur. The definition of the domain correlation $\phi(w)$ is based on the Sørensen-Dice coefficient, which is usually used to measure the similarity. The detailed definition is shown in 3.14

$$\phi(w) = \frac{2.F_p(w).F_Q(w)}{F_p(w) + F_Q(w)} \tag{3.14}$$

$F_P(w)$ and $F_D(w)$ are the frequency of the occurrence of the words after standardization in two domains as defined by 3.15

$$F_D(w) = \frac{f_D(w)}{f_D(w^k)} \tag{3.15}$$

CHAPTER IV

Implementation, Results and Discussions

4.1 Implementation and Results

4.1.1 Dataset Pre-processing:

The experimental dataset comprises product reviews sourced from the Amazon website, originally compiled by McAuley et al. [20]. This dataset encompasses reviews from diverse domains, each assigned a rating within the 1 to 5-point range. This study focuses on three specific domains—books (B), restaurants (R), and electronics (E)—while filtering out short reviews (less than 7 words) and punctuation. A stop-word list is also employed to eliminate frequently occurring yet less informative words. The sample of the dataset is in Figure 4.1

	Rating	Review	Product Name	Product Category	Emotion	Data Source	Sentiment
0	5.0	অসাধারণ ফোন অনেক পছন্দ হয়েছে একদম অথেনটিক শাও...	Redmi 12C (4/128GB)	Electronics	Happy	Daraz	Positive
1	5.0	অল্প দামে দারুন একটা স্মার্টফোন	Redmi 12C (4/128GB)	Electronics	Love	Daraz	Positive
2	3.0	ডেলিভারী বা প্রডাক্ট নিয়ে কোন কথা নেই বেশ ভালো...	Redmi 12C (4/128GB)	Electronics	Sadness	Daraz	Negative
3	2.0	এখনো ভালো আছে ভবিষ্যতে কি হবে সেটা দেখার বিষয়	Redmi 12C (4/128GB)	Electronics	Fear	Daraz	Negative
4	5.0	অসাধারণ	Redmi 12C (4/128GB)	Electronics	Love	Daraz	Positive

Figure 4.1: A sample of dataset

A training corpus is randomly sampled to facilitate the acquisition of cross-domain sentiment analysis using word embeddings, containing 100,000 polarity reviews from each domain. The initial step involves learning sentiment word embeddings using the polarity reviews of an individual domain. Subsequently, the corpus data from a different domain is incorporated to facilitate the learning of cross-domain word embeddings 4.1.

We split the data into training and test sets; 80% of the data will be used to train the sentiment classifier, and 20% of the data will be used for testing 4.2.

Dataset	Pos.reviews	Neg.reviews	Vocab.size(k)	tokens(M)
Book	50000	50000	136	16
Restaurant	50000	50000	153	17
Electronics	50000	50000	93	11

Table 4.1: Dataset to learn word embeddings

Dataset	Reviews	Avg.length	Vocab.size(k)	Test set(%)
Book	40000	177	56	10
Restaurant	40000	187	59	10
Electronics	40000	122	47	10

Table 4.2: Dataset sentiment classification

4.1.2 Basic ELMo Model Implementation:

The proposed work extends the basic ELMo model. The basic ELMo model is understood and learned and further modified it achieve the goal. The steps of implementing the ELMo model are given below:

- ELMo (Embeddings from Language Models) generates word embeddings that are contextually aware, capturing the meaning of words based on their usage in the sentence.
- The model uses a bidirectional LSTM (biLSTM) to process the text in both forward and backward directions.
- First of all, the raw words are passed to a Convolutional Neural Network (CNN) to get the primary word embeddings which are context-independent. Then they are passed to the LSTM network.
- In the forward LSTM, each word is passed through the model in sequence, and its representation is updated based on the words that came before it.
- In the backward LSTM, the sentence is processed in reverse, and the word representations are updated based on the words that follow it.
- The final word embeddings are a combination of the forward and backward LSTM representations, providing deep contextualization of each word.

- ELMo representations are task-specific, meaning they can be fine-tuned for various NLP tasks by weighting the layers differently.

Below is an example of how the ELMo model works with bi-directional LSTM for the sentence **"I read a book yesterday"**.

Figure 4.2: Forward LM working

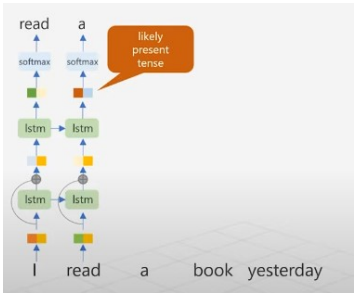


Figure 4.3: Forward LM working

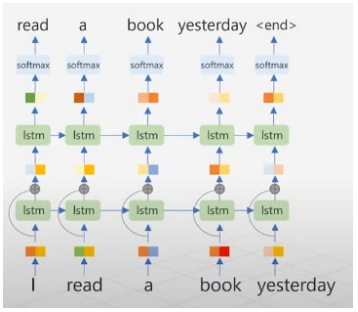


Figure 4.4: Forward LM working

This is how the **Forward LM** predicts the next words given the current words. Now let's look at the **Backward LM** and how it exactly works.

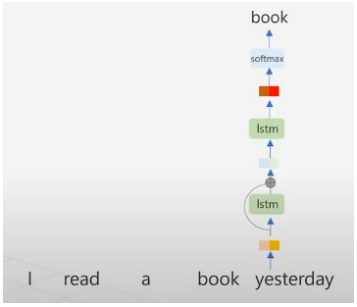


Figure 4.5: Backward LM working

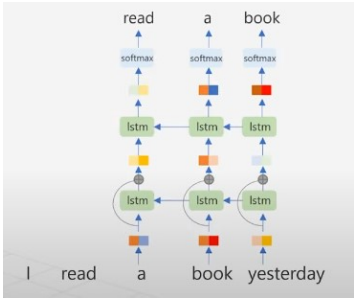


Figure 4.6: Backward LM working

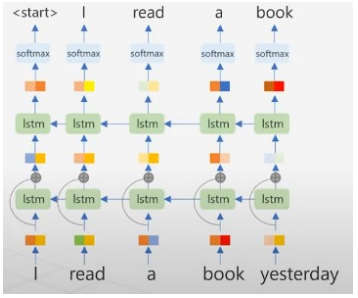


Figure 4.7: Backward LM working

This is how the **Backward LM** predicts the Previous words given the future words.

To get the word embeddings, all the vectors from internal states are multiplied by weights and then summed up.

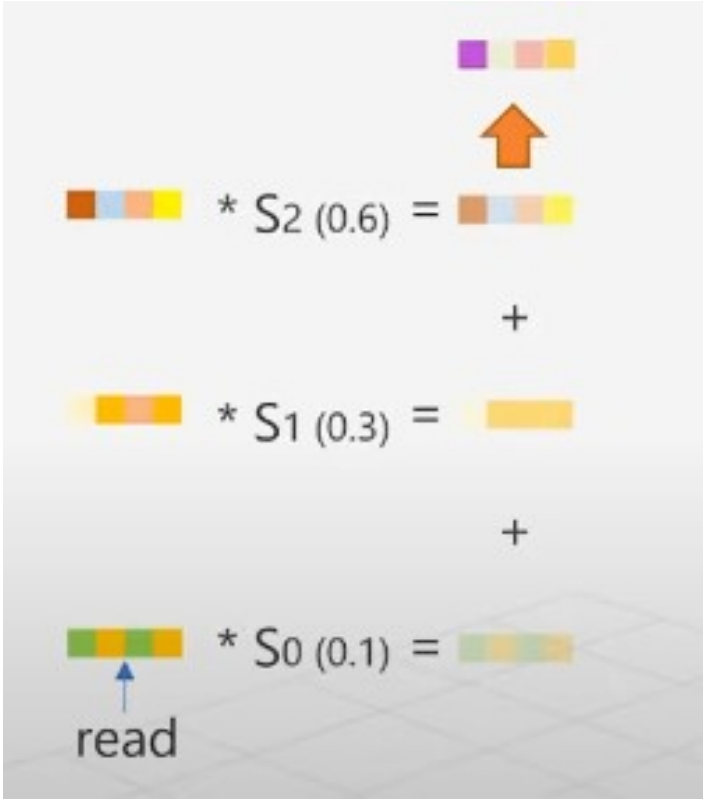


Figure 4.8: Pre-embedding

At the last stage, this embedding for each word is multiplied by a scaling parameter ”Gamma” which gives the actual contextualized embedding for every word.

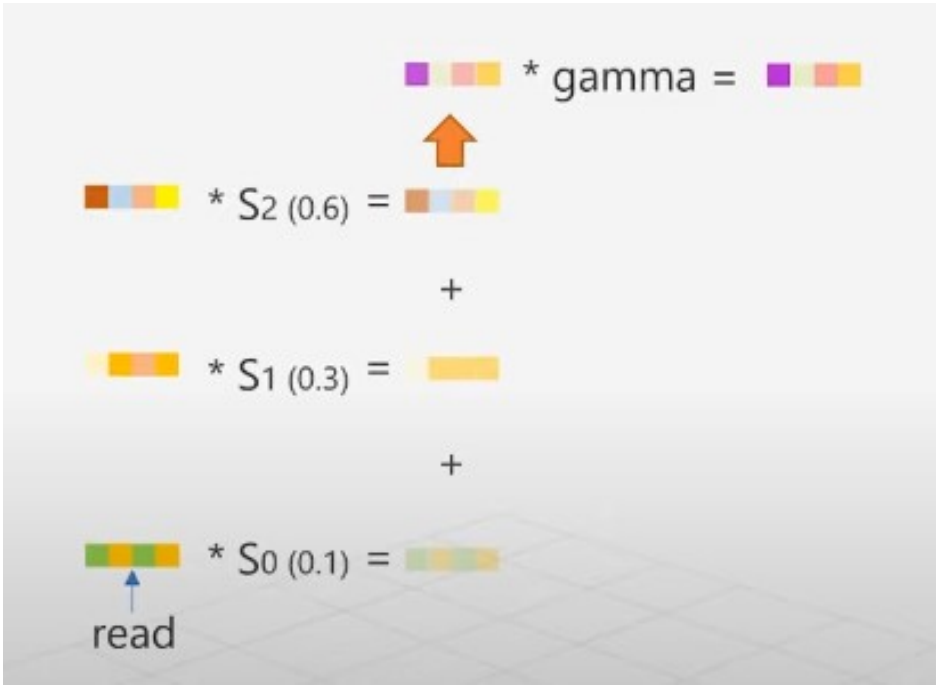


Figure 4.9: Final embedding

4.2 Objective Achieved

The major goals are to incorporate sentiment and domain relevance information in the word vector representation. The achieved objectives are now:

- The dataset has been processed to use.
- The related theory regarding understanding the work and implementing the work has been acquired.

4.3 Morality or Ethical Issues:

In the thesis, attention has been placed on providing accurate citations to ensure proper credit. The methodology utilized has also been acknowledged, demonstrating the commitment to recognizing sources and enhancing the clarity and understanding of the work for readers.

4.4 Socio-Economic Impact and Sustainability:

This section explores the extensive implications of accurate review sentiment analysis, focusing on its socio-economic influence and role in fostering long-term sustainability. **Socio-Economic Impact:**

- Informed Consumer Decisions: Accurate sentiment analysis empowers consumers to make informed choices by understanding product and service sentiments,
- Brand Perception: Sentiment analysis gauges brand perception.
- Market Insights: Sentiment-derived insights inform businesses of consumer preferences, facilitating adaptive strategies based on evolving market trends.

Sustainability:

- Enhancing Products and Services: Accurate sentiment analysis informs product enhancements, aligning offerings with user preferences and needs.
- Minimizing Waste: Aligned with customer sentiment, businesses reduce waste by producing under expectations, minimizing environmental impact.

- Efficient Resource Allocation: Sentiment awareness guides resource distribution, optimizing operational efficiency, and minimizing waste.

In conclusion, accurate review sentiment analysis profoundly influences consumer decisions, brand perception, and market insights, thus bolstering economic growth.

CHAPTER V

Conclusions

5.1 Conclusion and Challenges Faced

The objective is to develop an enhanced word embedding model that encompasses not only syntactic and semantic aspects but also higher-level information. This endeavor involves utilizing reviews from three domains, namely books, restaurants, and electronics.

The primary challenge revolves around the expansion of the Elmo model. To tackle this, a comprehensive grasp of the foundational Elmo model's structure has been gained. The performance evaluation of the modified model will be conducted through sentiment classification tasks involving user review data. This assessment will serve to compare the modified model against various well-established word vector models in mainstream use.

The biggest challenges faced up to now can be described as the shortage of CPU, GPU, and datasets collecting, creating datasets, creating sentiment word lists, and creating stopwords lists.

5.2 Future Works

Initially, the focus lies on conducting this work using English review data. In the future, there are plans to curate a review dataset in Bangla, upon which this approach will be implemented and adapted. As a result, the ELMo model for the Bangla language needs to be trained. At last, I will make a comparison of the performance of my trained model to the existing models.

References

[1] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.

[2] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. *Advances in neural information processing systems*, 26, 2013.

[3] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *nature*, 521(7553):436–444, 2015.

[4] Li Dong, Furu Wei, Ke Xu, Shixia Liu, and Ming Zhou. Adaptive multi-compositionality for recursive neural network models. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 24(3):422–431, 2015.

[5] Abdalraouf Hassan and Ausif Mahmood. Convolutional recurrent deep learning model for sentence classification. *Ieee Access*, 6:13949–13957, 2018.

[6] Kim Schouten and Flavius Frasincar. Survey on aspect-level sentiment analysis. *IEEE transactions on knowledge and data engineering*, 28(3):813–830, 2015.

[7] Meng Joo Er, Yong Zhang, Ning Wang, and Mahardhika Pratama. Attention pooling-based convolutional neural network for sentence modelling. *Information Sciences*, 373:388–403, 2016.

[8] Duyu Tang, Furu Wei, Bing Qin, Nan Yang, Ting Liu, and Ming Zhou. Sentiment embeddings with applications to sentiment analysis. *IEEE transactions on knowledge and data Engineering*, 28(2):496–509, 2015.

[9] Yoshua Bengio, Aaron Courville, and Pascal Vincent. Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence*, 35(8):1798–1828, 2013.

[10] Xinghui Dong and Junyu Dong. The visual word booster: A spatial layout of words descriptor exploiting contour cues. *IEEE Transactions on Image Processing*, 27(8):3904–3917, 2018.

[11] Abdulrahman Almuhareb, Waleed Alsanie, and Abdulmohsen Al-Thubaity. Arabic word segmentation with long short-term memory neural networks and word embedding. *IEEE Access*, 7:12879–12887, 2019.

[12] Dong Deng, Liping Jing, Jian Yu, and Shaolong Sun. Sparse self-attention lstm for sentiment lexicon construction. *IEEE/ACM transactions on audio, speech, and language processing*, 27(11):1777–1790, 2019.

[13] Syeda Rida-E-Fatima, Ali Javed, Ameen Banjar, Aun Irtaza, Hassan Dawood, Hussain Dawood, and Abdullah Alamri. A multi-layer dual attention deep learning model with refined word embeddings for aspect-based sentiment analysis. *IEEE Access*, 7:114795–114807, 2019.

[14] Michael T Mills and Nikolaos G Bourbakis. Graph-based methods for natural language processing and understanding—a survey and analysis. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 44(1):59–71, 2013.

[15] Danushka Bollegala, Tingting Mu, and John Yannis Goulermas. Cross-domain sentiment classification using sentiment sensitive embeddings. *IEEE Transactions on Knowledge and Data Engineering*, 28(2):398–410, 2015.

[16] Prathusha K Sarma, Yingyu Liang, and William A Sethares. Domain adapted word embeddings for improved sentiment classification. *arXiv preprint arXiv:1805.04576*, 2018.

[17] Yanbin Hao, Tingting Mu, Richang Hong, Meng Wang, Xueliang Liu, and John Y Goulermas. Cross-domain sentiment encoding through stochastic word embedding. *IEEE Transactions on Knowledge and Data Engineering*, 32(10):1909–1922, 2019.

[18] Ya'ou ZHAO, Jiachong ZHANG, Yibin LI, Xianrui FU, and Wei SHENG. Sentiment analysis using embedding from language model and multi-scale convolutional neural network. *Journal of Computer Applications*, 40(3):651, 2020.

[19] Jie Zhou, Junfeng Tian, Rui Wang, Yuanbin Wu, Wenming Xiao, and Liang He. Sentix: A sentiment-aware pre-trained model for cross-domain sentiment analysis. In *Proceedings of the 28th international conference on computational linguistics*, pages 568–579, 2020.

[20] Julian McAuley, Christopher Targett, Qinfeng Shi, and Anton Van Den Hengel. Image-based recommendations on styles and substitutes. In *Proceedings of the 38th international ACM SIGIR conference on research and development in information retrieval*, pages 43–52, 2015.