# TRIBHUVAN UNIVERSITY

# INSTITUTE OF ENGINEERING

**Khwopa College Of Engineering**

Libali, Bhaktapur

**Department of Computer Engineering**

A PROPOSAL ON

## Word Embeddings in Nepali Language

*Submitted in partial fulfillment of the requirements for the degree*

**BACHELOR OF COMPUTER ENGINEERING**

Submitted by

| | |
|---|---|
| Manish Pyakurel | KCE077BCT020 |
| Rupak Neupane | KCE077BCT028 |
| Sarjyant Shrestha | KCE077BCT033 |
| Srijan Gyawali | KCE077BCT036 |

**Khwopa College Of Engineering**

Libali, Bhaktapur

2024

# Abstract

Suspendisse vitae elit. Aliquam arcu neque, ornare in, ullamcorper quis, commodo eu, libero. Fusce sagittis erat at erat tristique mollis. Maecenas sapien libero, molestie et, lobortis in, sodales eget, dui. Morbi ultrices rutrum lorem. Nam elementum ullamcorper leo. Morbi dui. Aliquam sagittis. Nunc placerat. Pellentesque tristique sodales est. Maecenas imperdiet lacinia velit. Cras non urna. Morbi eros pede, suscipit ac, varius vel, egestas non, eros. Praesent malesuada, diam id pretium elementum, eros sem dictum tortor, vel consectetuer odio sem sed wisi.


**Keywords**: *Khwopa college of engineering*

# Contents

# List of Tables

# List of Figures

# List of Symbols and Abbreviation

| | |
|---|---|
| AI | Artificial Intelligence |

## CHAPTER 1

# Introduction

## 1.1 Background Introduction

NLP is a branch of linguitics, computer science, and artificial intelligence concerned with computer human interaction, mainly how to design computers to process and evaluate huge volumes of natural language data [2]. Pre-training of an NLP model plays an essential role in transfer learning, where a language model will be trained on a vast corpus set and later fine-tune the model for a specific purpose [3]. Word embedding is a fundamental concept in NLP. It is a real-valued vector representation of words by embedding both semantic and syntactic meanings obtained from unlabeled large corpus [4]. It is of n-dimensional distributed representation of a text that attempts to capture the meanings of the words [2]. Word embeddings can be obtained using language modeling and feature learning techniques, where words or phrases from the vocabulary are mapped to vectors of real numbers [5]. Pre-trained word embeddings encode general word semantics and lexical regularities of natural language, and have proven useful across many NLP tasks, including word sense disambiguation, machine translation, and sentiment analysis, to name a few [6]. Word embeddings have been found to be very useful for many NLP tasks, including but not limited to Chunking [7], Question Answering [8], Parsing and Sentiment Analysis [9]. [10]

**Types of Word Embedding Techniques** [11]

**Traditional Embeddings**: Traditional word embeddings represent words as fixed vectors in an n-dimensional space, capturing semantic relationships between words. These embeddings are static and do not change based on context or training data.

**Static Embeddings**: Static word embeddings are pre-trained on a large corpus of text and do not change during model training. They are useful for tasks where word meanings remain constant across different contexts.

**Contextualized Embeddings**: Contextualized word embeddings, like BERT, are based on transformer models that can capture word meanings based on the context in which they appear. These embeddings provide more accurate representations of words by considering the surrounding context during training.

**Combined Word Embedding and Neural Network Models**: Combining word embeddings with neural network models can enhance model accuracy in various natural language processing tasks such as sentiment classification, text categorization, and phrase prediction.

Nepali is one of the languages that uses Devanagari, a script used in many languages spoken in Asian countries. It is spoken by more than 20 million people, mainly in Nepal, and other places in the world including Bhutan, India and Myanmar [12]. It has been rarely used for Natural Language Processing services. Nepali can be quite complex due to its many sounds, grammar rules, and ways to change words. Due to its complex grammatical structure and rich characters, extracting fruitful information from the corpus has been challenging [3].

The advancement of NLP technologies adapted to individual languages, like Nepali, hold immense potential for empowering communities and enhancing accessibility to digital resources for Nepali language. By filling the gap between technological innovation and linguistic diversity, we can unlock new possibilities for communication and education.

## 1.2 Problem Statement

Even though Word Embeddings can be directly learned from raw texts in an unsupervised fashion, gathering a large amount of data for its training remains a huge challenge in itself for a low-resource language such as Nepali [13]. Despite having breakthroughs in the field of NLP, productive results with the Nepali language have not been achieved.

One of the primary reasons for this is the need for more computational resources [3]. As mentioned in the most recent study in this topic (i.e NepaliBERT [3]), there is a lack of larger, more diverse, and context-rich dataset to enhance the accuracy and robustness of the word embeddings in Nepali language. This research study seeks to construct a more finely tuned model capable of generating embeddings for the Nepali corpus. It is seen that there is reduction of perplexity by using XLM.

## 1.3 Objective

The main aim of this project is:

- To develop context dependent word embedding for Nepali language.

# CHAPTER 2

# Literature Review

Most of the research that has been undertaken on the Nepali corpus was focusing on generating embeddings through traditional approaches like TFIDF, Wod2Vec and other embedding methods. [14] [15] [16] and [17] implemented TF-IDF on Nepali text for text classification and other purposes such as sentiment analysis. Similarly, Word2Vec approach in nepali corpus was implemented by [14] [18] and [19]. 300-Dimensional Word Embeddings for Nepali Language [20] has pre-trained Word2Vec model having 300-dimensional vectors for more than 0.5 million Nepali words and phrases. The embeddings generated using the methods described above are static, implying that each word retains only one vector representation regardless of its context of use. However, contemporary trends emphasize the adoption of contextual-dependent embeddings over their contextual-independent counterparts. As highlighted earlier, there have been limited studies on BERT within the Nepali context. [13] claimed to provide an efficient Nepali BERT embedding, but despite having a huge dataset they were short of computational resources due to which they had to compromise on the different BERT parameters. They modified the BERT model by averaging the hidden states from the last two hidden layers to get the embeddings, whereas, for getting the baseline results, instead of using any pre-trained word vectors, a trainable Keras embedding layer was used in front of the architecture mentioned above which automatically learns the word embeddings by only using the provided training examples. [21] and [22] also tried the capacity of BERT for cross-lingual in Natural Language Processing.

There are also studies done in XLM [23]. The paper compares a Nepali language model with a cross-lingual language model trained in Nepali but enriched with different combinations of Hindi and English data, showing how leveraging data from related languages can benefit low-resource languages like Nepali. 300-Dimensional Word Embeddings for Nepali Language [20] has pre-trained Word2Vec model having 300-dimensional vectors for more than 0.5 million Nepali words and phrases. The embeddings generated using the methods described above are static, implying that each word retains only one vector representation regardless of its context of use. However, contemporary trends emphasize the adoption of contextual-dependent embeddings over their contextual-independent counterparts. As highlighted earlier, there have been limited studies on BERT within the Nepali context. [13] claimed to provide an efficient Nepali BERT embedding, but despite having a huge dataset they were short of computational resources due to which they had to compromise on the different BERT parameters. They modified the BERT model by averaging the hidden states from the last two hidden layers to get the embeddings, whereas, for getting the baseline results, instead of using any pre-trained word vectors, a trainable Keras embedding layer was used in front of the architecture mentioned above which automatically learns the word embeddings by only using the provided training examples. [21] and [22] also tried the capacity of BERT for cross-lingual in Natural Language Processing.

There are also studies done in XLM [23]. The paper compares a Nepali language model with a cross-lingual language model trained in Nepali but enriched with different combinations of Hindi and English data, showing how leveraging data from related languages can benefit low-resource languages like Nepali.

# CHAPTER 3

# Theoritical Background

## 3.1 Transformers

A transformer model is a neural network that learns context and thus meaning by tracking relationships in sequential data like the words in this sentence. Transformer models apply an evolving set of mathematical techniques, called attention or self-attention, to detect subtle ways even distant data elements in a series influence and depend on each other. First described in a 2017 paper from Google [1], transformers are among the newest and one of the most powerful classes of models invented to date. They're driving a wave of advances in machine learning some have dubbed transformer AI. Stanford researchers called transformers "foundation models" in an August 2021 paper [24] because they see them driving a paradigm shift in AI. The "sheer scale and scope of foundation models over the last few years have stretched our imagination of what is possible," they wrote.

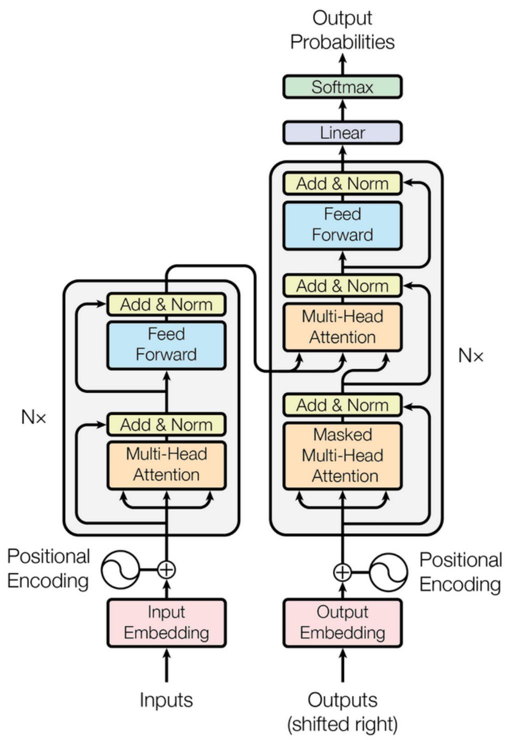


Figure 3.1: The Transformer - model architecture. [1]

## 3.2 BERT

BERT stands for Bidirectional Encoder Representations from Transformers. It is designed to train deep bidirectional representations from unlabeled text by jointly conditioning on both left and right context.

# CHAPTER 4

# METHODOLOGY

## 4.1 SOFTWARE DEVELOPMENT APPROACH

Agile development is a software development approach that emphasizes incremental progress and rapid cycles. It involves releasing small increments of functionality that build upon previous versions. Thorough testing is conducted for each release to ensure software quality. Agile is often employed for time-critical applications. Although this project is not time-critical this model seems to be the most optimal and practical in our case.

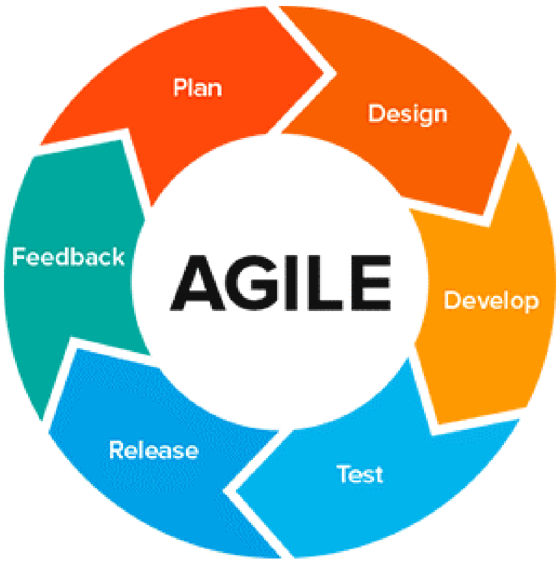


Figure 4.1: Agile Model for Software Development

*source:* *https://mobilelive.medium.com/agile-development-a-comprehensive-guide-for-the-modern-era-d2fe9ae7b395*

## 4.2 Proposed System Block Diagram

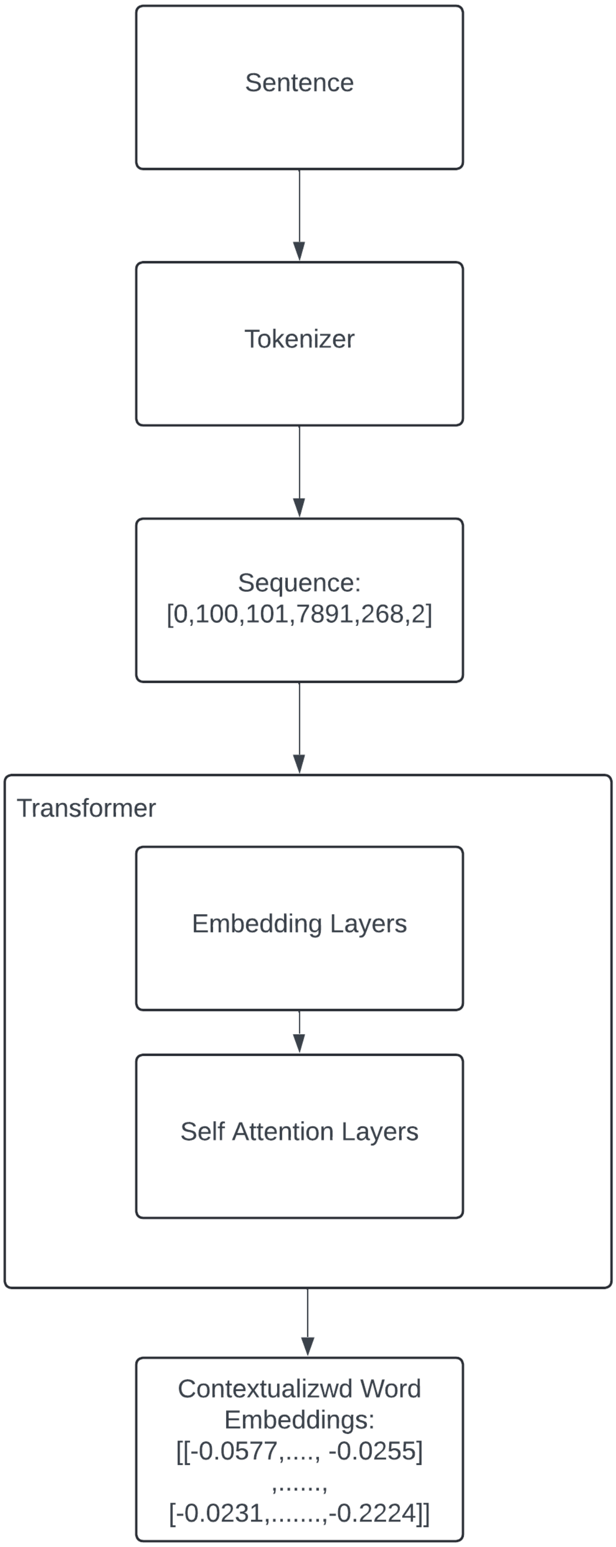


Figure 4.2: Block diagram of Proposed Sytem

## 4.3 Description of Working Flow of Proposed System

## 4.4 Performance Evaluation Metrics

# CHAPTER 5

# SYSTEM DESIGN

## 5.1 Requirement Analysis

### 5.1.1 FUNCTIONAL REQUIREMENT

These are specifications that describe the fundamental capabilities and behaviors a system or product must exhibit to meet the users' needs and achieve its intended purpose.

#### 5.1.1.1 Tokenization

The system should be able to tokenize Nepali text into individual words or subword units, considering the complexities of the Nepali script.

#### 5.1.1.2 Embedding Generation

Generate dense vector representations (embeddings) for each word or subword unit in the Nepali vocabulary. These embeddings should capture semantic relationships between words.

#### 5.1.1.3 Embedding Lookup

Allow users to retrieve the embedding vector for any given word or subword unit in the Nepali vocabulary.

#### 5.1.1.4 Similarity Calculation

Calculate semantic similarity between words or subword units based on their embedding vectors. Users should be able to compare words and get similarity scores.

### 5.1.2 NON-FUNCTIONAL REQUIREMENT

These are the characteristics and qualities that describe how a system should behave and perform.

#### 5.1.2.1 Usability

The system should have a user-friendly interface or API that allows users to easily interact with word embeddings without requiring deep technical knowledge.

#### 5.1.2.2 Reliability

The system aims to be highly available, with minimal downtime, to ensure users can access word embedding functionalities when needed.

#### 5.1.2.3 Interoperability

The system ensures compatibility with various operating systems, programming languages, and NLP frameworks to facilitate integration with existing systems and workflows.

#### 5.1.2.4 Maintainability

The system with a modular architecture will enable easy maintenance, updates, and future enhancements.

### 5.1.3 FEASIBILITY STUDY

The following points describes the feasibility of the project.

#### 5.1.3.1 Economic Feasibility

The total expenditure of the project is just computational power. The computational resources can be fulfilled with the help of college. Therefore, the project is economically feasible.

#### 5.1.3.2 Technical Feasibility

While existing datasets on this topic are available, they are insufficient. But, by using the abundance of Nepali news articles, books, and literature accessible online can augment our corpus significantly through web scraping.

#### 5.1.3.3 Operational Feasibility

The operational processes, including web-scraping and model training, are well-defined and can be efficiently carried out by the project team. Additionally, the project aligns with the existing technical infrastructure and capabilities, making it operationally feasible.

# CHAPTER 6

# IMPLEMENTATION PLAN

## 6.1 Schedule(Gantt Chart)

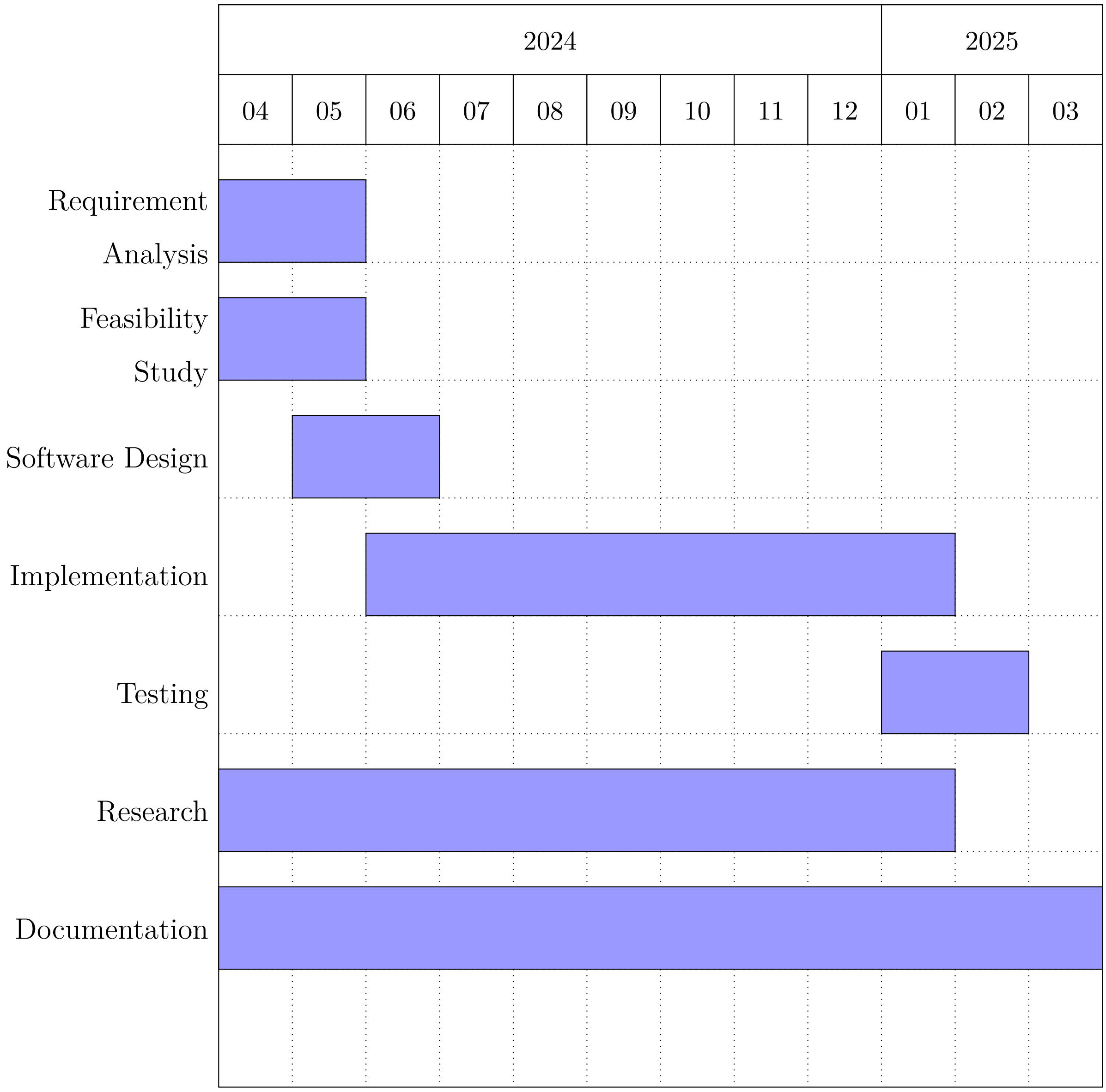


Figure 6.1: Gantt Chart

## 6.2 Hardware and Software requirements

### 6.2.1 SOFTWARE REQUIREMENT

This project requires following softwares:

#### Python

Python is our primary language for this project, chosen for its versatility and simplicity. All machine learning frameworks are imported using Python, leveraging its dominant position in data science and machine learning. This allows us to seamlessly integrate powerful libraries like TensorFlow and PyTorch for efficient model development. It also provides essential libraries like Scikit-learn, Numpy, Pandas, etc.

#### Pytorch

Pytorch is a an open-source machine learning framework. It provides a flexible and dynamic computational graph, which allows for easy experimentation and rapid development of deep learning models.

#### Natural Language Toolkit

Natural Language Toolkit (NLTK) is a free and open source Python library for natural language processing. NLTK provides stemming, lowercase, categorization, tokenization, spell check, lemmatization, and semantic reasoning text processing packages. It gives access to lexical resources like WordNet.

## 6.3 Cost Estimation

# CHAPTER 7

# EXPECTED OUTCOME

The primary outcome is the generation of high-quality word embeddings for the Nepali language that capture semantic relationships between words effectively.

# REFERENCES


[1] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need, 2023.

[2] Deepak S Asudani, Narendra K Nagwani, and Pradeep Singh. Impact of word embedding models on text analytics in deep learning environment: a review. *Artificial Intelligence Review*, 22(1):1–81, Feb 2023. Epub ahead of print.

[3] Shushanta Pudasaini, Subarna Shakya, Aakash Tamang, Sajjan Adhikari, Sunil Thapa, and Sagar Lamichhane. Nepalibert: Pre-training of masked language model in nepali corpus. pages 325–330, 10 2023.

[4] Bin Wang, Angela Wang, Fenxiao Chen, Yuncheng Wang, and C.-C. Jay Kuo. Evaluating word embedding models: methods and experimental results. *APSIPA Transactions on Signal and Information Processing*, 8:e19, 2019.

[5] Wikipedia contributors. Word embedding — Wikipedia, the free encyclopedia, 2024. [Online; accessed 4-May-2024].

[6] Alejandro Moreo, Andrea Esuli, and Fabrizio Sebastiani. Word-class embeddings for multiclass text classification, 2019.

[7] Joseph Turian, Lev-Arie Ratinov, and Yoshua Bengio. Word representations: A simple and general method for semi-supervised learning. In Jan Hajič, Sandra Carberry, Stephen Clark, and Joakim Nivre, editors, *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 384–394, Uppsala, Sweden, July 2010. Association for Computational Linguistics.

[8] Stefanie Tellex, Boris Katz, Jimmy Lin, Aaron Fernandes, and Gregory Marton. Quantitative evaluation of passage retrieval algorithms for question answering. pages 41–47, 07 2003.

[9] Richard Socher, Jeffrey Pennington, Eric H. Huang, Andrew Y. Ng, and Christopher D. Manning. Semi-supervised recursive autoencoders for predicting sentiment distributions. In Regina Barzilay and Mark Johnson, editors, *Proceedings*

*of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 151–161, Edinburgh, Scotland, UK., July 2011. Association for Computational Linguistics.

[10] Felipe Almeida and Geraldo Xexéo. Word embeddings: A survey, 2023.

[11] A comprehensive review on word embedding techniques. pages 538–543, 2023.

[12] Nobal B. Niraula, Saurab Dulal, and Diwa Koirala. Linguistic taboos and euphemisms in nepali, 2020.

[13] Pravesh Koirala and Nobal B. Niraula. NPVec1: Word embeddings for Nepali - construction and evaluation. In Anna Rogers, Iacer Calixto, Ivan Vulić, Naomi Saphra, Nora Kassner, Oana-Maria Camburu, Trapit Bansal, and Vered Shwartz, editors, *Proceedings of the 6th Workshop on Representation Learning for NLP (RepL4NLP-2021)*, pages 174–184, Online, August 2021. Association for Computational Linguistics.

[14] Janardan Bhatta, Dipesh Shrestha, Santosh Nepal, Saurav Pandey, and Shekhar Koirala. Efficient estimation of nepali word representations in vector space. *Journal of Innovations in Engineering Education*, 3(1):71–77, Mar. 2020.

[15] Oyesh Mann Singh. Nepali multi-class text classification. 2019.

[16] Tej Bahadur Shahi and Ashok Kumar Pant. Nepali news classification using naive bayes, support vector machines and neural networks. In *2018 international conference on communication information and computing technology (iccict)*, pages 1–5. IEEE, 2018.

[17] Samujjwal Ghosh and Maunendra Sankar Desarkar. Class specific tf-idf boosting for short-text classification: Application to short-texts generated during disasters. In *Companion Proceedings of the The Web Conference 2018*, pages 1629–1637, 2018.

[18] Kaushal Kafle, Diwas Sharma, Aayush Subedi, and Arun Timalsina. Improving nepali document classification by neural network. 03 2018.

[19] Ashok Basnet and Arun Timalsina. Improving nepali news recommendation using classification based on lstm recurrent neural networks. pages 138–142, 10 2018.

[20] Rabindra Lamsal. 300-dimensional word embeddings for nepali language, 2019.

[21] Rajan. Nepalibert. `https://huggingface.co/Rajan/NepaliBERT`, 2021.

[22] Milanmg. Bert-nepali. `https://huggingface.co/Milanmg/Bert-Nepali`, 2022.

[23] Alexis CONNEAU and Guillaume Lample. Cross-lingual language model pretraining. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

[24] Rishi Bommasani, Drew A Hudson, Ehsan Adeli, Russ Altman, Simran Arora, Sydney von Arx, Michael S Bernstein, Jeannette Bohg, Antoine Bosselut, Emma Brunskill, et al. On the opportunities and risks of foundation models. *arXiv preprint arXiv:2108.07258*, 2021.