

# **COMPARATIVE EVALUATION OF TRANSFORMER-BASED LANGUAGE MODELS FOR NEPALI LANGUAGE**

by

Suyogya Ratna Tamrakar

A Research Study Submitted in Partial Fulfillment of the Requirements for the Degree  
of Master of Engineering in Computer Science

Examination Committee: Dr. Chaklam Silpasuwanchai (Chairperson)  
Prof. Matthew N. Dailey  
Dr. Mongkol Ekpanyapong

Nationality: Nepalese  
Previous Degree: Bachelor of Engineering in Computer Engineering  
Kathmandu University  
Nepal

Scholarship Donor: AIT Partial Scholarship


Asian Institute of Technology  
School of Engineering and Technology  
Thailand  
December 2022

## **AUTHOR'S DECLARATION**

I, SUYOGYA RATNA TAMRAKAR, declare that the work carried out for this research was in accordance with the regulations of the Asian Institute of Technology. The work presented in it are my own and has been generated by me as the result of my own original research, and if external sources were used, such sources have been cited. It is original and has not been submitted to any other institution to obtain another degree or qualification. This is a true copy of the research including final revisions.

Date: November 20, 2022

Name: Suyogya Ratna Tamrakar

Signature: 

## ACKNOWLEDGEMENTS

First and foremost, I would like to thank Asian Institute of Technology (AIT) for providing me with partial scholarships for the duration of my studies and for allowing me to be a part of this beautiful university.

I would want to express my heartfelt appreciation to Dr. Chaklam Silpasuwanchai, who served as both my supervisor and the committee chairman and was a valuable resource to me during my research. Many thanks to him for being the most amazing lecturer at AIT who would effortlessly blend into the activities of his fellow students in order to be a part of it. He has persuaded me that research can be a part of everyday life.

I would want to express my heartfelt gratitude to my research defense committee, Prof. Matthew N. Dailey and Dr. Mongkol Ekpanyapong, for offering insightful comments and recommendations on my work.

I would also like to thank my other friends and lab members for their continuous encouragement and insights into the study. I am also grateful to be able to use our AIT facilities, such as the AIT Brain Lab computer resources and the Puffer server, without which my research experiments would take months to complete.

Most importantly, I would want to thank my family in Nepal for their unconditional love and support throughout my academic journey and stay in Thailand.

Finally, I would want to give credit where credit is due: to myself, for the hard work, effort, and, most of all, patience that led to my success in earning a Master's degree from AIT.

## ABSTRACT

Large pre-trained transformer models using self-supervised learning have achieved state-of-the-art performances in various NLP tasks. However, for low-resource language like Nepali, pre-training of monolingual models remains a problem due to lack of training data and well-designed and balanced benchmark datasets. Furthermore, several multilingual pre-trained models such as mBERT and XLM-RoBERTa have been released, but their performance remains unknown for Nepali language. Nepali monolingual pre-trained transformer models were compared with multilingual models to determine their performance using a Nepali text classification dataset as a downstream task based on different number of classes and data sizes, taking machine learning (ML) and deep learning (DL) algorithms as baselines. Under-representation of Nepali language in mBERT resulted in overall poor performance, but, XLM-RoBERTa, which has a larger vocabulary size, produced state-of-the-art performance which is relatively similar to that of Nepali DistilBERT and DeBERTa, which outperformed all of the baseline algorithms. Bi-LSTM and SVM from the baselines also performed very well in variety of settings. Moreover, to assess the cross-language knowledge transfer for the cases when monolingual models are not available, HindiRoBERTa, a monolingual Indian language model was also evaluated on Nepali text dataset. This research mainly contributes to the Nepali NLP community by creation of news classification dataset with 20 classes, with over 200,000 articles and performance evaluation of various pre-trained monolingual Nepali transformers with multilingual transformers, DL and ML algorithms.

# CONTENTS

	Page
<b>ACKNOWLEDGEMENTS</b>	<b>iii</b>
<b>ABSTRACT</b>	<b>iv</b>
<b>LIST OF TABLES</b>	<b>vii</b>
<b>LIST OF FIGURES</b>	<b>viii</b>
<b>CHAPTER 1 INTRODUCTION</b>	<b>1</b>
1.1 Background of the Study	1
1.2 Statement of the Problem	4
1.3 Research Questions	5
1.4 Objectives	5
1.5 Organization of the study	6
<b>CHAPTER 2 RELATED WORK</b>	<b>7</b>
2.1 Text classification	7
2.2 Pre-trained transformer models for high-resource languages	10
2.2.1 For English language	10
2.2.2 High-resource languages other than English	11
2.3 Pre-trained transformer models for low-resource languages	12
2.3.1 For Nepali language	12
2.3.2 Low-resource languages other than Nepali	13
<b>CHAPTER 3 METHODOLOGY</b>	<b>16</b>
3.1 Data Collection	18
3.2 Pre-processing and Data Selection	19
3.2.1 Pre-processing	19
3.2.2 Data Selection	20
3.3 Dataset	22
3.3.1 Creation	22
3.3.2 Splits	22
3.4 Tokenization	22
3.5 Pre-trained tokenizers	23
3.5.1 WordPiece Tokenizer	24
3.5.2 SentencePiece Tokenizer	24

3.5.3	Byte-level Byte Pair Encoding	25
3.6	Pre-trained transformer models	25
3.6.1	BERT	26
3.6.2	RoBERTa	27
3.6.3	DistilBERT	27
3.6.4	DeBERTa	27
3.6.5	mBERT	27
3.6.6	XLM-RoBERTa	28
3.6.7	HindiRoBERTa	28
3.7	Baseline algorithms	28
3.7.1	Multinomial Naive Bayes (MNB)	28
3.7.2	Random Forest (RF)	28
3.7.3	Support Vector Machine (SVM)	29
3.7.4	Bidirectional Long Short Term Memory (Bi-LSTM)	29
3.8	Fine-tuning and classification	29
<b>CHAPTER 4</b>	<b>RESULTS AND DISCUSSION</b>	<b>30</b>
4.1	Experiment 1: Intuitive Evaluation	30
4.2	Experiment 2: Evaluation on text classification dataset	31
4.2.1	Hyperparameter Selection	31
4.2.2	Performance on Test split by number of classes	33
4.2.3	Performance on Test split by data size	35
4.2.4	Weighted precision analysis on Test split	36
4.3	Discussion	37
4.3.1	Important findings	37
4.3.2	Comparison with past research	40
4.3.3	Useful experimental decisions	40
4.3.4	Practical recommendations	41
4.3.5	Limitations and future work	41
<b>CHAPTER 5</b>	<b>CONCLUSION</b>	<b>43</b>
<b>REFERENCES</b>		<b>44</b>

## LIST OF TABLES

Tables	Page
Table 3.1 20 news categories	19
Table 3.2 Top frequent words in each news category	21
Table 3.3 Handling of Nepali special characters and digits	23
Table 3.4 Summary of pre-trained tokenizers and models	26
Table 4.1 Hyperparameters and training time of transformer models for <i>np20ng_30K</i>	34
Table 4.2 Hyperparamters and training time of Bi-LSTM model	35
Table 4.3 Hyperparamters and training time of ML models	35
Table 4.4 Test set accuracy on <i>np16ng_30K</i> and <i>np20ng_30K</i> dataset	36
Table 4.5 Test set accuracy on 20-class dataset ( <i>np20ng</i> ) with varying data sizes	37
Table 4.6 Weighted average precision for <i>np20ng_30K</i> dataset	38
Table 4.7 Weighted average precision for <i>np16ng_30K</i> dataset	39

## LIST OF FIGURES

Figures	Page
Figure 3.1 Rating model as “ <i>Best</i> ” based on exact prediction	16
Figure 3.2 Rating model as “ <i>Acceptable</i> ” based on mean similarity of masked word and predicted words	17
Figure 3.3 The overall methodology for Experiment 2	18
Figure 3.4 News document distribution among categories	19
Figure 3.5 The snapshot of 20 Nepali Newsgroup ( <i>np20ng</i> ) dataset	20
Figure 3.6 WordPiece tokenization	24
Figure 3.7 SentencePiece tokenization	24
Figure 3.8 Byte-Level BPE tokenization	25
Figure 4.1 Example of masked word predictions in a short sentence.	31
Figure 4.2 Example of masked word predictions on a slightly long sentence.	32
Figure 4.3 Model quality ratings obtained from evaluation done on 50 sentences	33
Figure 4.4 Distribution of mean cosine similarity scores across models	33
Figure 4.5 The Bi-LSTM architecture	34
Figure 4.6 Test accuracy scores based on number of classes	36
Figure 4.7 Test accuracy scores based on size of data on <i>np20ng</i> dataset	37



# CHAPTER 1

## INTRODUCTION

### 1.1 Background of the Study

Large pre-trained models through self-supervised learning such as Bidirectional Encoder Representations from Transformers (BERT) (Devlin, Chang, Lee, & Toutanova, 2018) has attracted a lot of interest from researchers. While low-resource languages like Nepali are still in the early stages mostly due to a lack of high-quality data, high-resource languages such as English are dominating recent advancements in Natural Language Processing (NLP). There are just a few studies in Nepali that use current state-of-the-art transformers, despite the fact that there are a number of studies done for NLP in Nepali using machine learning and deep learning algorithms. However, Nepali being a morphologically rich language with fairly complex grammatical structures, inability to properly address its linguistic features appears to have an impact on the performance of machine learning models. Furthermore, such models might not be suited to perform NLP tasks involving longer sequences of text data compared to deep learning models that can extract the context and semantics from long texts. A huge amount of unstructured text data can be used to learn language patterns, and no matter what language it is, deep learning and attention mechanisms from transformers have been dramatically outperforming traditional machine learning models in this regard.

A wide range of traditional machine learning algorithms and deep learning models have been trained to perform text classification tasks in Nepali, mostly news group classification (Basnet & Timalsina, 2018; Kafle, Sharma, Subedi, & Timalsina, 2016; Koirala & Niraula, 2021; T. B. Shahi & Pant, 2018; Singh, 2018; Subba, Paudel, & Shahi, 2019; Thakur & Singh, 2014; Wagle & Thapa, 2021) and sentiment analysis (Piryani, Piryani, Singh, & Pinto, 2020; Regmi, Bal, & Kultsova, 2017; T. Shahi, Sitaula, & Paudel, 2022; Sitaula, Basnet, Mainali, & Shahi, 2021; Tamrakar, Bal, & Thapa, 2020; Thapa & Bal, 2016). Several studies (Aggarwal, Chauhan, Kumar, Mittal, & Verma, 2020; Al-Yahya, Al-Khalifa, Al-Baity, AlSaeed, & Essam, 2021; Terechshenko et al., 2020) show that transformer models give significantly better performances than the former approaches due to their ability to attend to longer sequences of text using attention mechanisms. In any case, there are only a few research in Nepali that utilize current state-of-the-art

transformers (Koirala & Niraula, 2021; Maskey, Bhatta, Bhatta, Dhungel, & Bal, 2022; Wagle & Thapa, 2021), despite the fact that a large number of NLP studies in Nepali have been performed using machine learning and deep learning algorithms. There are few pre-trained monolingual transformer models for Nepali that have been trained on a relatively low number of data compared to high-resource languages, and there are few multilingual models whose performance has not yet been thoroughly evaluated for Nepali. In addition, training and evaluating a model requires a well-designed dataset for improved generalization.

The deep learning models are very domain-specific and need a lot of labeled data for training which might lead the model to lack generalization. Contrary to this issue, the transformer models have made significant progress in the NLP field recently. They are best-known for their potential to generalize specific linguistic patterns and features in a language through extensive pre-training in an unsupervised manner. However, pre-training a transformer model from scratch is computationally expensive and resource-intensive and are trained effectively by bigger organizations. Studies show that transformers models give significantly better performances than machine learning and deep learning techniques (Aggarwal et al., 2020; Al-Yahya et al., 2021; Terechshenko et al., 2020) mostly due to their ability to attend to longer sequences of text using attention mechanisms.

Training a machine learning and deep learning model for Nepali language can be quite challenging due to its complex linguistic features. A series of pre-processing steps need to be taken for extracting the features and preparing a model from Nepali text. The most common but important steps is to reduce the vocabulary size by stemming, removing stopwords and tokenization. Normalization is also important as Nepali language consists of different vowel modifiers, which when spoken are indistinguishable from others, but text data may contain multiple version of these words which in turn adds noise to the data (Koirala & Niraula, 2021). In addition to these pre-processing requirements, the Nepali language also lacks generic pre-processing tools which takes into account every aspect of the language, without which the performance of traditional models seem to be affected with inability to address the linguistic features properly.

Therefore, one of the big advantages of using pre-trained transformers is that, even with a minimal pre-processing, fine-tuning on a specific task can produce significant

results. Transformers model reduces pre-processing overhead by addressing the Out-Of-Vocabulary (OOV) words using several sub-word tokenization algorithms like word-piece, sentence-piece, etc that tokenizes the words into sub-words which could be common suffixes for other rare or uncommon words. This can be a huge advancement for low-resource languages like Nepali which lack proper pre-processing tools. One such study is done by Aggarwal et al. (2020), where BERT model which was fine-tuned using very little pre-processing outperforms traditional XGBoost and LSTM model for the task of news classification.

There are a limited number of pre-trained Nepali models available. Recently, a few number of Nepali pre-trained models have been trained namely DistilBERT (Distilled version of BERT) and DeBERTa (Decoding-enhanced BERT with Disentangled Attention), and their performances are being compared with each other by Maskey et al. (2022) where they fine-tuned their models for news classification task with 16 classes. However, the dataset *16NepaliNews*<sup>1</sup> used by them is highly unbalanced and most likely to produce improper implications. For example, around 50% of total news documents belong to a single class and few classes have relatively very low numbers of documents. Furthermore, the fine-tuning methodology was not documented in detail and they did not report a balanced-accuracy score despite using an unbalanced dataset. However, they have investigated thoroughly appropriate methods of pre-training transformers model for Nepali language.

Moreover, some of the Nepali text classification datasets which are available publicly are either highly imbalanced, or have comparatively less amount of data. Such as number of documents in each class for *16NepaliNews*<sup>1</sup> dataset ranges from 16 to 7,452, and 10 out of 16 classes have less than 300 documents, given that it has only 14,364 documents in total. Furthermore, *Nepali Linguistic Dataset*<sup>2</sup> consists of some classes which are website-specific such as “*Koseli*” which means “*Souvenir*” in English. With some observation we found that this category consists of news from other categories like *Literature*, *Entertainment*, *Music* and *Society* which are also the classes in this dataset. Although these kinds of classes can be used for training and evaluating a classifier, a well-balanced, generic dataset needs to be created with high quality data for better gen-

---

<sup>1</sup><https://github.com/sndsabin/Nepali-News-Classifier>

<sup>2</sup><https://ieee-dataport.org/documents/nepalilinguistic>

eralization.

## 1.2 Statement of the Problem

With the popularity of transformer architecture in NLP, a large pool of pre-trained models are becoming available day by day. While there are a very limited number of such models for low-resource language like Nepali, bigger organizations have released multilingual models that are pre-trained on over hundred languages at once such as Google and Facebook released mBERT (multilingual BERT) and XLM-RoBERTa (Cross Lingual Model - Robustly Optimized BERT) respectively. However, depending on the proportion of language-specific data these multilingual models are trained on, their performance may vary from language to language. As very few monolingual language models have been pre-trained for Nepali language such as BERT, RoBERTa, DistilBERT and DeBERTa, the performance of these models need to be compared with the performance of multilingual models.

Therefore, the two main problems this study addresses are:

1. Relatively less amount of high quality data and balanced labelled dataset for performing Nepali text classification tasks
2. No detailed comparison of Nepali pre-trained models with traditional machine learning and deep learning algorithms as well as multilingual transformer models with better quality data.

First, the issue of having relatively less data in Nepali to perform text classification is addressed by creating a balanced dataset with much more data. This dataset can also be applied for sequence-to-sequence tasks such as headline generation from Nepali news documents. Second, taking ML and DL algorithms as baselines, namely Multinomial Naive Bayes (MNB), Random Forest (RF), Support Vector Machine (SVM) and Bidirectional Long Short Term Memory (Bi-LSTM), Nepali pre-trained transformer models were compared with with multilingual models by fine-tuning them on a text classification task using our created dataset. In addition to that, a different language model, HindiRoBERTa is also used to fine-tune on Nepali text to see its cross-lingual performance. This approach will be helpful in the scenario when there are no properly pre-trained language models available for specific language.

### 1.3 Research Questions

This research evaluates the performance of monolingual Nepali pre-trained transformer models when compared with multilingual models along with ML and DL algorithms as baselines in a number of settings such as class sizes and data sizes. The more specific research questions for this study are:

1. Which models perform better when there is a limited amount of data?
2. Which models are good when there are varying numbers of classes?
3. Do transformer models perform better than ML and DL algorithms?
4. Between monolingual and multilingual transformer models, which performs the best?

### 1.4 Objectives

The overall objective of this research study is to compare different Nepali pre-trained transformers models by fine-tuning them on multi-class text classification datasets in Nepali language and evaluate them based on their performance metrics using ML and DL algorithms as the baselines.

The specific objectives of this study are listed below:

1. To compare monolingual Nepali transformer models (BERT, DistilBERT, RoBERTa and DeBERTa), with multilingual versions (mBERT and XLM-RoBERTa) and a different language model, HindiRoBERTa.
2. To create a balanced labeled newsgroup dataset with more data size suited specifically for multi-class text classification tasks in Nepali.
3. To prove the effectiveness of minimally-preprocessed pre-trained Nepali transformers by comparing them with traditional ML and DL algorithms namely, MNB, RF, SVM and Bi-LSTM as baselines.

The pre-trained transformer models are evaluated using two different ways. First, an intuitive evaluation of models is done based on word masking where the effectiveness of the model was determined based on their ability to predict the masked word. Second, a huge number of news documents labelled into 20 different categories were scraped to create a well-balanced dataset, where the models are fine-tuned and evaluated based on the best validation loss. A 20-class and 16-class subsets of datasets with varied data sizes, such as each class containing 1,500, 500, 250, and 50 documents, were created in order to compare the models under study based on varying number of classes and

amount of data respectively.

### **1.5 Organization of the study**

This research work is organized as follows. Chapter 2 discusses the related work in the field, Chapter 3 discusses the proposed methodology for dataset creation, text preprocessing and model fine-tunings. Similarly, Chapter 4 presents the outcomes of analyses and discussions on the experimental results along with limitations and future directions. The research report is concluded in Chapter 5.

## CHAPTER 2

### RELATED WORK

The majority of research on text classification has been conducted mostly in high-resource languages such as English, Italian, German, Spanish, and others, as well as Asian languages such as Arabic, Chinese, and Japanese. Nepali language being morphologically rich with fairly complex grammar, the effectiveness of a classification model depends on how well the model represents several linguistic features. Nonetheless, important advancements have been made in recent years to the field of Nepali text classification utilizing diverse methodologies, such as classical machine learning models and deep learning models. Furthermore, the existing works do provide a baseline for NLP researchers in carrying out further investigation of more reliable and effective approaches.

#### 2.1 Text classification

Nepali language being a low-resource language, the majority of text classification research in Nepali uses news classification datasets for experiments as they are readily available through reliable online sources and are subject to fewer grammatical mistakes. Thakur and Singh (2014) trained a Naive Bayes classifier for Nepali news classification task which was evaluated on a self-created dataset of news stories having five classes namely *Arts & Entertainment*, *Business & Economics*, *Politics*, *Science & Technology* and *Sports*. In order to improve the performance, they incorporated domain-specific lexical knowledge into the simple Naive Bayes classifier based on 100 manually added words and 1000 frequently appearing words. In comparison to the typical Naive Bayes implementation, which achieved an accuracy of 80.67%, they achieved an accuracy of 84.24%.

Kafle et al. (2016) compared three text classification models (SVM + TF-IDF, SVM + Word2Vec (Mikolov, Sutskever, Chen, Corrado, & Dean, 2013), Cosine Similarity with Latent Semantic Indexing (LSI)) for Nepali news classification task on their self-created dataset having 13 different classes. Different classes were *Automobiles*, *Finance*, *Crime*, *Employment*, *Entertainment*, *Health*, *Literature*, *Politics*, *Society*, *Sports*, *Technology*, *Tourism* and *Others*. They found that the proposed Word2Vec variant showed improvement over TF-IDF only method and LSI method in F1-score by 1.6% and 2.2% respectively.

Singh (2018) compared several variants of traditional machine learning algorithms with deep learning methods for the purpose of Nepali news classification. They used Nepali news corpus having 15,000 documents and 16 classes which were pre-processed using regular expressions. Performances of Logistic Regression (LR), SVM, MNB, Bernoulli Naive Bayes (BNB), K-Nearest Neighbor (KNN), Multi Layered Perceptron (MLP), etc were compared with deep learning classifiers namely, Recurrent Neural Networks (RNN) using LSTM, Gated Recurrent Unit (GRU) and Adaptive GRU which all used Word2Vec as an embedding. They found that a simple perceptron model outperformed GRU network due to limited amount of data size. In addition, deep learning methods failed to perform better than some traditional machine learning methods.

Using TF-IDF as a feature extractor, T. B. Shahi and Pant (2018) also compared Naive Bayes, SVM and Neural Networks for automated Nepali news classification on a self created dataset on Nepali news containing only 4,964 documents varied across 20 different categories. They concluded that SVM outperforms Naive Bayes and neural networks. Basnet and Timalsina (2018) used LSTM for Nepali news recommendation and compared with SVM. News data classified into 8 categories were collected from 5 different news portals and a series of pre-processing methods were applied such as stemming and stop word removal before feeding it to the model. Word2Vec was used for feature extraction for LSTM models and models were compared with different variations in hidden layers. The authors concluded that the LSTM model showed remarkable improvement over SVM in accuracy of 85.6% and precision of 88% when the number of units in hidden layers were increased.

The Nepali NLP community is quickly advancing with the most recent NLP trends due to the transformer models that are delivering better results in a variety of tasks. Wagle and Thapa (2021) trained their own BERT model with 68M parameters and compared with LSTM and bi-LSTM using a Nepali news classification dataset with over 200K articles categorized into 17 classes. Although the BERT model showed slightly better results with a weighted average F-score of 95.45%, the other two baseline models namely LSTM and Bi-LSTM gave a good score with a weighted average F1-score of 92.94% and 93.65% respectively. Maskey et al. (2022) made an important contribution for Nepali NLP by fine-tuning various Nepali pre-trained transformer language models on a news classification task. They pre-trained from scratch two Nepali transformer



models, DeBERTa and DistilBERT, and compared with other available Nepali models. They found out that their models outperform other transformer models due to addressing language-specific features during the pre-training phase. DeBERTa and DistilBERT models fine-tuned on 16NepaliNews dataset obtained accuracy of 88.93% and 88.31% respectively.

Apart from news classification tasks in Nepali language, sentiment classification is another well-explored field that has used ML and DL approaches. Thapa and Bal (2016) performed a document level sentiment analysis for movie reviews using SVM, MNB and Logistic Regression which used TF-IDF and BoW as feature extractors. Moreover, their data consisted of a very low number of samples, i.e. 179 for positive and 205 for negative, and MNB outperformed other methods. Over a few years, several research (Pirayani et al., 2020; Regmi et al., 2017; Tamrakar et al., 2020) have been conducted using traditional ML and DL models for Nepali sentiment classification.

A sentiment analysis of Nepali tweets related to COVID-19 using three different variants of CNN model was done by Sitaula et al. (2021) where they used three CNN models ensembled together, trained to capture three different information namely contextual information using FastText (Bojanowski, Grave, Joulin, & Mikolov, 2017) embeddings for COVID-19 related tweets, domain-agnostic vectors to capture features from non-COVID-19 related documents using a news dataset, and domain-specific vectors to capture features from COVID-19 related documents itself. Moreover, they have done a comparative analysis of their proposed model with traditional machine learning algorithms such as Extreme Gradient Boosting (XGBoost), SVM with Linear & RBF (Radial Basis Function) kernel, Artificial Neural Network (ANN), RF, etc. They conclude that, though deep learning models tend to be more accurate text classifiers than conventional methods, the training methodology utilized by high-resource languages, which predominantly use the Latin script, may not be applicable to low-resource languages such as Nepali, which uses the Devanagari script.

Aside from text classification in Nepali language, there is numerous research for high-resource languages. Ranjan, Ghorpade, Kanthale, Ghorpade, and Dubey (2017) used LSTM with TF-IDF and BOW methods for feature extraction in classifying English news documents using 20 Newsgroup dataset. A range of preprocessing schemes such as tokenization, stop word removal, lemmatization and feature selection were implemented

before training a neural network classifier. Their classifier achieved an accuracy of 92% in 25 epochs of training and concluded that LSTM approaches prove to be successful in categorizing text documents more efficiently. With advancement in the NLP domain, traditional feature extraction methods are slowly being replaced with deep learning based word embeddings. In contrast to this, Chirra, Maddiboyina, Dasari, and Aluru (2020) trained a deep learning network with GloVe (Pennington, Socher, & Manning, 2014) embeddings for feature extraction where they used two datasets and achieved 95.02% accuracy using RNN for multi-class classification on a news classification dataset and 98.5% accuracy using CNN for binary classification on email classification dataset.

Aggarwal et al. (2020) fine-tuned a BERT model on fake news detection task and compared its performance with XGBoost and LSTM networks on NewsFN dataset. Their BERT model outperformed by a huge margin achieving 97% accuracy compared to 89.4% and 86.23% of XGBoost and LSTM models respectively. Most importantly they showed that transformer models are capable of performing significantly better than typical machine learning and deep learning algorithms with less pre-processing efforts. Because there are few pre-processing tools for the Nepali language, huge pre-trained transformer-based language models could be promising. This represents one aspect in which our study is particularly relevant to this research.

Dogra and Varun (2021) proposed a hybrid model for Indian Banking News where the DistilBERT model was integrated with machine learning algorithms. They compared the effectiveness of contextual language representation using DistilBERT with the widely used TF-IDF which is context-independent. In addition, They compared supervised ML algorithms such as Logistic regression, Linear SVC, Decision tree and Random forest. As random forest outperformed other supervised methods achieving an accuracy of 98%, they created a hybrid model with DistilBERT fine-tuned with RF classifier which was then compared with a similar model with domain-specific rules, where the later model achieved the highest accuracy of over 99% for their domain.

## **2.2 Pre-trained transformer models for high-resource languages**

### ***2.2.1 For English language***

Adoma, Henry, and Chen (2020) used an emotion recognition ISEAR dataset to analyse the efficacy of transformer-based pre-trained language models such as XLNet, DistilBERT, BERT and RoBERTa to classify seven different categories of emotions. Their

findings indicate that refining pre-trained models based on transformers is useful for identifying emotion in text. In their experiments, RoBERTa showed highest accuracy of 74.31% followed by XLNet with 72.99%, BERT with 70.09% and DistilBERT being the least accurate with 66.93% accuracy. However, they also compared these models based on their demand of computational resources where DistilBERT came to be fastest compared to the XLNet model which is most expensive or slowest.

Cortiz (2021) investigated the performance of five different transformer-based pre-trained model namely DistilBERT, BERT, XLNet, ELECTRA and RoBERTa on GoEmotion (Demszky et al., 2020) dataset which consisted of 58,000 English Reddit comments labelled with 27 different categories of emotion. They kept BERT as a baseline model and evaluated the performance of classification task using F1-score and training time duration. Their experiments showed that the RoBERTa model achieved the highest macro-average f1-score with 0.49 and got best outcomes for 14 classes, followed by DistilBERT, XLNet and ELECTRA. With these experiments, he concluded that ELECTRA resulted in being an inaccurate model for the GoEmotion dataset with F1-score of 0.33 and bad performance in 18 of the classes. However, experiments based on computational resource requirement BERT required the longest time to train, followed by XLNet, RoBERTa and DistilBERT. Surprisingly, the ELECTRA model despite giving the worst performance was able to train the fastest in just 13 minutes compared to 2 hours and 40 minutes of training BERT.

Gupta, Gandhi, and Chakravarthi (2021) built a fake review classifier based on pre-trained models of BERT family namely BERT, RoBERTa, ALBERT and DistilBERT and made an analysis based on 10% and 50% of the Yelp reviews dataset due to limited computational power. A restaurant domain-specific model was considered to be the baseline model where authors achieved an accuracy of 67.8% using SVM. RoBERTa outperformed the baseline model by obtaining an accuracy of 69% followed by DistilBERT with 67%, BERT with 65% and ALBERT with 64%. As earlier, BERT was the model with the slowest time to train, followed by RoBERTa and ALBERT whereas DistilBERT was the fastest.

### ***2.2.2 High-resource languages other than English***

Al-Yahya et al. (2021) compared transformer based models based on Arabic language namely AraBERT, AraELECTRA, AraGPT2, QARiB, Arbert and Marbert with neural

network based methods such as CNN, RNN, and GRU on Arabic COVID-19 pandemic tweets for detecting fake news. Moreover, they also compared with linear embedding-based models. These models were fine-tuned on four datasets namely ArCOV19-Rumors (an Arabic COVID-19 Twitter dataset with manual annotations), Covid-19-Fakes (a COVID-19 Twitter dataset with automatic annotations), AraNews (Arabic misinformation dataset), ANS (a corpus having with news titles). Their experimental results suggest that linear models with Word2Vec and FastText embeddings, having trained on a small in-domain dataset, achieved a limited performance of accuracy of approx. only 83%. Transformers models on the other hand were trained using various learning rates, outperformed all deep learning models, with QARiB attaining best accuracy of over 95.8% followed by AraBERT with 95.3%. Despite being a text generation (sequence-to-sequence) model, AraGPT2 achieved accuracy of 92%. However, to evaluate the generalizability of the models, when fine-tuned on COVID-19-Fakes dataset, these best performing models were not robust enough mainly due to the class imbalance issue. Transformer-based models outperformed neural network-based methods, with the F1 score increasing from 0.83 (best GRU) to 0.95 (best transformer-based model, QARiB) and accuracy increasing by 16%.

Although there exist respective multilingual models based on BERT and RoBERTa such as mBERT and XLM-RoBERTa, there also exist number of monolingual variants which are language-specific such as ALBERTo (Polignano, Basile, De Gemmis, Semeraro, & Basile, 2019) (Italian), FlauBERT (Le et al., 2019) (French), Bertje (de Vries et al., 2019) (Dutch), BERTimbau (Souza, Nogueira, & Lotufo, 2020) (Brazilian/Portugese), HuBERT (Ács, Lévai, Nemeskey, & Kornai, 2021) (Hungarian), and so on. The results from these monolingual models show that, with adequate amounts of pre-training data, they are able to outperform the multilingual counterparts.

## **2.3 Pre-trained transformer models for low-resource languages**

### ***2.3.1 For Nepali language***

Despite few monolingual transformer models based on BERT and RoBERTa are available for Nepali language, they seem to lack a appropriate pre-training strategy because they ignore a number of Nepali vowel modifiers, as a consequence resulted in lots of grammatically incorrect or meaningless words in the vocabulary. However, Maskey et al. (2022) pre-trained two large monolingual models namely DistilBERT and DeBERTa

for Nepali language analyzing minute details of the linguistic properties of Nepali language. Furthermore, they fine-tuned their models with available monolingual Nepali transformer models as well as a multilingual XLM-RoBERTa on a news classification task. However, they did not compare with Nepali RoBERTa model and other traditional algorithms like SVM. Moreover, they have used a highly unevenly distributed dataset for multi-class classification and did not report the metrics such as balanced accuracy, precision, recall or F1-scores for their models. In contrast to this research, our study makes use of their pre-trained models and compares them with existing monolingual Nepali models as well as two of the popular multilingual language models mBERT and XLM-RoBERTa.

### ***2.3.2 Low-resource languages other than Nepali***

Lehečka and Švec (2021) pre-trained two monolingual transformers models for Czech language namely FERNET-C5 (suitable for wide range of domains) on C5 (Czech Colossal Clean Crawled Corpus) dataset, and FERNET-News which was suitable for single domain of news classification task. FERNET-C5 is a BERT model and instead of using BERT’s internal WordPiece tokenizer, they trained SentencePiece tokenizer with vocabulary size of over 100K. FERNET-News is a RoBERTa model trained using a byte-level BPE tokenizer with vocabulary size of 50K tokens. Moreover, they compared their proposed models with four multilingual models namely MultiBERT, SlavicBERT, XLM-RoBERTa-base and XLM-RoBERTa-large. They fine-tuned all the models on three sentiment analysis datasets namely CSFD (Czech-Slovak Movie Database), MALL (product review dataset from Czech e-shop mall) and FCB (Facebook dataset from Czech pages), and two news classification datasets namely CTDC (Czech Text Document Corpus) and CN (a labelled news dataset) suited for multi-label topic identification task. FERNET-C5 model showed best F1 scores of 85.36, 79.75, 81.07 and 91.25 on CSFD, MALL, FCB and CTDC datasets respectively, whereas, FERNET-News outperformed other models on CN dataset. Despite XLM-RoBERTa-large being a relatively huge model, it did outperform their model, but with a very small margin of around 1%. With this results, they say that use of multilingual models may result in limited performance and low-resource languages might be under-represented in such models, so training of monolingual models give better results in most of the scenarios.

Kumar and Albuquerque (2021) obtained zero-shot transfer learning while predicting from English to Hindi for the classification of tweet sentiment. They evaluated the XLM-RoBERTa model on two Hindi language datasets, IIT-Patna Movie Review and Product Review, after refining it on the Semeval dataset, which contained English tweets. Even without any exposure to Hindi during training, the model was able to achieve an accuracy of 60.93%. For low-resource languages that may lack large labeled and unlabeled data, this strategy can be of great value, according to the researchers. A similar technique was applied by Ranasinghe and Zampieri (2020) where cross-lingual embeddings from XLM-RoBERTa were used to make predictions on low-resource languages namely Bengali and Hindi where 0.8415 and 0.8568 macro F1 scores were reported respectively for the task of offensive language identification. Moreover, they also reported results for Spanish and achieved a macro F1-score of 0.7513. With the results, they contrasted that XLM-R with transfer learning method outperforms other models under study.

Wongso, Lucky, and Suhartono (2022) pre-trained and released three monolingual transformer models namely RoBERTa, BERT and GPT-2 for Sundanese language on OSCAR, CC-100, C4 and Wikipedia data. These models were evaluated based on an emotion classification dataset of Sundanese tweets keeping linear SVC (Support Vector Classifier) with TF-IDF, multilingual BERT, XLM-RoBERTa and IndoBERT as baseline. As Indonesian and Sundanese have comparable linguistic structures, the IndoBERT model was compared and, interestingly, performed slightly lower than the Sundanese BERT model although having been trained on Indonesian data. Sundanese RoBERTa achieved the highest accuracy of 98.41% and F1-macro score of 98.43% outperforming all the models in comparison. Moreover, mBERT outperformed their monolingual BERT variant very slightly by 0.01% in accuracy, but XLM-RoBERTa being a larger model was outperformed by all the models except GPT-2. This may be due to the fact that Sundanese language could be under-represented in XLM-RoBERTa. GPT-2 on the other hand being a sequence-to-sequence model, even linear SVC outperformed it by 1.6% accuracy for the text classification task. SVC attained a promising accuracy of 96.43% with only 30K parameters in comparison to millions of parameters in other models, relatively due to intensive pre-processing steps prior to tokenization. They conclude that even though multilingual models are pre-trained with more parameters and substantially bigger corpora than monolingual ones, the model's ultimate performance on downstream tasks for specific language is greatly influenced by the amount of pre-training corpus used for that

language.

Due to the varying amounts of pre-training data, studies have shown that monolingual models are typically more effective than multilingual models. This is also clearly shown by the fact that pre-trained monolingual models for a variety of low-mid resource Asian languages, such as IndoBERT (Wilie et al., 2020) for Indonesian, PhoBERT (Nguyen & Nguyen, 2020) for Vietnamese, AraBERT (Antoun, Baly, & Hajj, 2020) for Arabic, WangchanBERTa (Lowphansirikul, Polpanumas, Jantrakulchai, & Nutanong, 2021) for Thai, ParsBERT (Farahani, Gharachorloo, Farahani, & Manthouri, 2021) for Persian, etc. typically outperforming their multilingual counterparts on downstream tasks.

Few such pre-trained monolingual Nepali transformer models have also been released based on BERT, DistilBERT, RoBERTa and DeBERTa. Most of these Nepali models are pre-trained on Nepali news corpus, OSCAR Corpus or Wikipedia. It is worth evaluating Nepali monolingual models due to the fact that they are pre-trained on different data sources. In this research study, BERT-based models pre-trained in Nepali language are fine-tuned for Nepali news classification task and are comparatively evaluated among each other.

## CHAPTER 3

### METHODOLOGY

The overall evaluation of transformer models were carried out with two different kinds of experiments. The first experiment was intuitive evaluation of transformer models, where we observed the Masked Language Modeling (MLM) capability of the models. Every transformer models under study were pre-trained for MLM objectives where they were required to predict or recover randomly masked words. Similarly, in this experiment, for each model, we gave a set of Nepali text document or a sentence with masking a single word and let the models predict that word. Then, the transformer models were evaluated intuitively with the human-knowledge whether the predicted word is exactly the same word or in any way matches the context of the text around it.

In addition, this evaluation was validated with a more quantitative technique where among the top five words predicted by the model, we determined whether they match the masked word or are semantically related to it. For this, we utilized Nepali Word2Vec embeddings (Koirala & Niraula, 2021) to calculate the cosine similarity between the words in the set. Cosine similarity score helps us identify whether or not two words are close in embedding space. Based on the mean similarity values of the five predictions, the models were then classified into four distinct categories: *Best*, *Good*, *Acceptable*, and *Poor*. In addition, if a model is able to make a perfect prediction of the masked word as in Figure 3.1, the mean similarity value is discarded and the model is labeled as *Best* otherwise is labeled by the mean value of five similarities as shown in Figure 3.2.

**Figure 3.1**

*Rating model as “Best” based on exact prediction*

```
Sentence: कोभिड महाव्याधिको पीडाबाट मुक्त नहुँदै मानव समाजलाई यतिबेला युद्धको विध्वंसकारी क्षतिको चिन्ताले चिन्तित बनाएको छ
Masked Word: चिन्तित
Masked Sentence: कोभिड महाव्याधिको पीडाबाट मुक्त नहुँदै मानव समाजलाई यतिबेला युद्धको विध्वंसकारी क्षतिको चिन्ताले <mask> बनाएको छ
Cosine Similarity between masked word and predicted words:
चिन्तित, चिन्तित ==> 1.0000
चिन्तित, आक्रान्त ==> 0.4852
चिन्तित, स्तब्ध ==> 0.4729
चिन्तित, त्रसित ==> 0.6415
चिन्तित, निराश ==> 0.6721

Mean: 0.6543
DistilBERT: Sakonii/distilbert-base-nepali ==> Best
```

For the second experiment, we fine-tuned transformer models on a text classification dataset. The overall fine-tuning methodology is depicted in Figure 3.3. Over 200,000



**Figure 3.2**

*Rating model as “Acceptable” based on mean similarity of masked word and predicted words*

```
Sentence: कोभिड महाव्याधिको पीडाबाट मुक्त नहुँदै मानव समाजलाई यतिबेला युद्धको विध्वंसकारी क्षतिको चिन्ताले चिन्तित बनाएको छ
Masked Word: चिन्तित
Masked Sentence: कोभिड महाव्याधिको पीडाबाट मुक्त नहुँदै मानव समाजलाई यतिबेला युद्धको विध्वंसकारी क्षतिको चिन्ताले <mask> बनाएको छ
Cosine Similarity between masked word and predicted words:
चिन्तित, बाध्य ==> 0.3906
चिन्तित, गम्भीर ==> 0.3560
चिन्तित, कमजोर ==> 0.3513
चिन्तित, पीडा ==> 0.2556
चिन्तित, दुःख ==> 0.2993

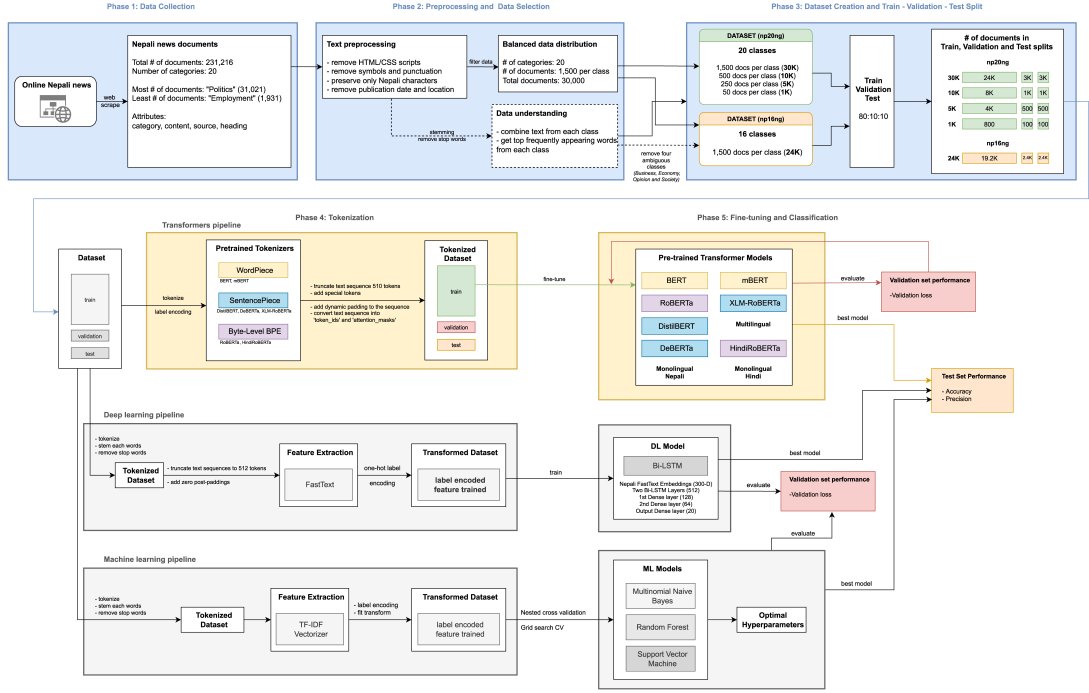
Mean: 0.3306
XLMRoBERTa: xlm-roberta-base ==> Acceptable
```

Nepali news documents were scraped from popular online news portals such as *Onlinekhabar*, *Nepalkhabar*, *Aarthiknews*, *Ekantipur*, *Setopati*, etc. These documents were labeled into their appropriate categories based on the category-specific URLs from respective sources. Altogether documents belonging to 20 categories were retrieved (Table 3.1). Total amount of documents scraped for each category is shown in Figure 3.4. Furthermore, some categories such as *Economy*, *Society*, *Opinion*, *Business* were ambiguous as they represented news articles belonging to other categories in the dataset like *Market*, *Bank*, and *Health*. Therefore, in order to see the robustness of models across different classes, two variations of our dataset were taken, one having 20 classes and the other having 16 classes (removed 4 ambiguous categories) with a well-balanced sample of 1,500 documents each per class. In addition to that, for the 20-class dataset, we sampled datasets having 500, 250 and 50 numbers of documents each per class, in order to see how each model performs when data sizes are limited.

A typical pre-processing step such as cleaning of non-Nepali characters, symbols, and punctuations, removal of news location and publication dates, etc. were performed prior to fine-tuning. In addition, for ML and DL models, stemming was done and stop words were removed prior to tokenization. All variations of datasets were used to fine-tune on our Nepali news group classification dataset as a downstream task using monolingual Nepali transformer models namely BERT, RoBERTa, DistilBERT and DeBERTa. Two popular multilingual models namely mBERT and XLM-RoBERTa along with a different language (Hindi) model HindiRoBERTa were also evaluated. In addition to that, we used three typical ML algorithms namely MNB, RF and SVM as well as one DL algorithm namely Bi-LSTM as baselines to compare and evaluate their performances.

**Figure 3.3**

*The overall methodology for Experiment 2*



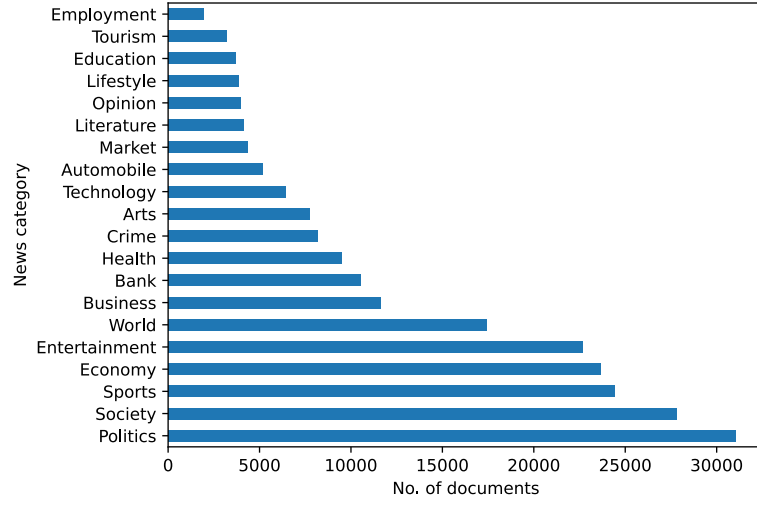
### 3.1 Data Collection

The first phase is data collection where a news scraper was developed that scraped news articles from several online Nepali news portals namely *Nagarik*, *Ekantipur*, *Onlinekhabar*, *Gorkhapatra*, *Nepalkhabar*, *Nepalipatra*, *Crimenews*, *Aarthiknews*, *Ratopati* and *Educationpati*. Altogether 231,216 news documents belonging to 20 different categories (Table 3.1) were retrieved using category-specific URLs. Category "Employment" has 1,931 articles and category "Politics" has 31,021 articles which are the minimum and maximum respectively among other categories. The snapshot of the dataset is shown in Figure 3.5 and news document distribution in each of the categories is shown in Figure 3.4.

This dataset is primarily designed for the purpose of multi-class text classification tasks in Nepali and is targeted to be utilized by Nepali NLP research community in future. Therefore, various other attributes such as news sources and headings were also retrieved in addition to basic content and category. Our full dataset<sup>1</sup> can be downloaded from HuggingFace datasets and source code is available in the GitHub repository<sup>2</sup>.

<sup>1</sup><https://huggingface.co/datasets/Suyogyart/np20ng>

<sup>2</sup><https://github.com/Suyogyart/nepali-transformers-evaluation>

**Figure 3.4***News document distribution among categories***Table 3.1***20 news categories*

Arts	Automobile	Bank	<i>Business</i>	Crime
<i>Economy</i>	Education	Employment	Entertainment	Health
Lifestyle	Literature	Market	<i>Opinion</i>	Politics
<i>Society</i>	Sports	Technology	Tourism	World

*Note.* The *italicized* news categories are ambiguous as they consist of news documents belonging to other classes, so they are not present in the dataset with 16 classes.

## 3.2 Pre-processing and Data Selection

### 3.2.1 Pre-processing

As the first step of pre-processing, the empty values, HTML/CSS scripts, English characters and extra whitespaces were removed. Then, the special symbols and punctuation symbols which are not part of Nepali language were discarded. Most of the news articles coming from specific source had a news location and publication dates, which were also removed to let the fine-tuned model learn more important language features of the dataset.

In previous studies for Nepali text classification (Basnet & Timalsina, 2018; Kafle et al., 2016; Ranjan et al., 2017; T. B. Shahi & Pant, 2018), training models which are not transformers-based, other text pre-processing techniques like removal of stop words

**Figure 3.5***The snapshot of 20 Nepali Newsgroup (np20ng) dataset*

	source	category	heading	content
92081	Onlinekhabar	Sports	ब्याक बक खुला क्लब क्रिकेट प्रतियोगिता बैठमा आयोजना हुने	१८ फागुन, काठमाडौं । प्लाड तानसेन तथा एनजीजीसी को आयोजनामा आगामी वैत ११...
131863	Nepalkhabar	Economy	छ महिनाको वैदेशिक व्यापारमा नेपाललाई छ खर्ब रुपैयाँ घाटा	नेपालले चालू आर्थिक वर्षको ६ महिनामा ६ खर्बभन्दा धेरैको वैदेशिक व्यापार...
74686	Nepalkhabar	World	गृहयुद्ध रोक्न 'कू' गरेको सुडानको सेनाको दाबी	सुडानको सेनाका प्रमुखले सैन्य 'कू'को बधाउ गरेका छन्। सेनाका जनरल अब्देल...
32178	Crimenews	Crime	गोदावरी जंगलमा बालिका बलात्कृत	घरेलु कामदारको रूपमा राखिएकी १३ वर्षीया बालिका बलात्कृत भएकी छिन् । ललि...
103991	Nepalkhabar	Society	विद्यार्थीको हात भौध्ने शिक्षक पक्राउ	गृहकार्य नगरेको भन्दै विद्यार्थीको हात भौध्ने सिरहाको विधैयास्थित स्...
104230	Nepalkhabar	Society	निर्वाचित नभएका व्यक्तिलाई सर्वोच्च कार्यकारी पदमा नलेजाउँ: नागरिक अगुवा	नागरिक समाजका अगुवाहरुले निर्वाचित नभएका व्यक्तिलाई कार्यकारी पदमा नलेज...
163434	Onlinekhabar	Politics	राजपालाई मोहरा बनाएर कांग्रेसले चुनाव सारेको एमालेको आरोप	२ असार, काठमाडौं । नेकपा एमालेका महासचिव ईश्वर पोखरेलले चुनाव हार्ने भ...
40973	Ekantipur	Opinion	विकल्पविनाको लोकतन्त्र	कोरोना महामारीका कारण आर्थिक-सामाजिक क्षेत्र आक्रान्त बनेको छ । दल अनि ...
69965	Onlinekhabar	World	अलिबाबासहितका केही यिनियाँ कम्पनीमाथि प्रतिबन्धको तयारीमा ट्रम्प	३२ साउन, काठमाडौं । अमेरिकी राष्ट्रपति डोनाल्ड ट्रम्पले अलिबाबासहितका अ...
70864	Onlinekhabar	World	अमेरिकामा कोरोनाबाट मृत्यु हुनेको संख्या ३० हजार नाघ्यो	काठमाडौं । अमेरिकामा कोरोना भाइरस (कोभिड-१९) बाट मृत्यु हुनेको संख्या ३०...

and stemming had to be done in order to reduce the vocabulary size. Similarly, we performed stemming and stop word removal steps prior to training ML and DL algorithms. However, for evaluating transformer-based language models, words in the document are conveniently tokenized using several sub-word tokenization algorithms such as Word-Piece tokenizer used by models like BERT. Such models have a pre-defined set of tokens obtained by pre-training over a huge corpus, which not only consists of single words but also part of words often referred to as sub-words. For example, the frequently used terms are not divided into smaller subwords by the subword-based tokenization methods. Instead, it breaks up the uncommon words into more meaningful subwords. For instance, with BERT tokenizer, the English word “tokenization” is divided into “token” and “##ization”. If some uncommon words appear such as “internationalization”, it will be tokenized into “international” and “##ization”. Even though “internationalization” as a whole might not be available in the model’s vocabulary, it will be represented by these two tokens where the later token can be a part of many other words as well.

### 3.2.2 Data Selection

According to Kaffle et al. (2016), the state-of-the-art models could also report poor performance if there is unbalanced data distribution among the classes. In our dataset, as articles in all the categories were not equally distributed, and moreover, taking memory constraints during fine-tuning into account, we selected the category having least articles, i.e. *Employment* with 1,931 articles and randomly sample 1,500 pre-processed articles from each category to create a balanced dataset. So, our main dataset consists of 30,000 news articles in total, equally categorized into 20 different newsgroups. Furthermore, we sampled balanced datasets with 10,000, 5,000 and 1,000 news articles to

evaluate model performances for varying data sizes.

**Table 3.2**

*Top frequent words in each news category*

Category	Top frequent words
Arts	book, poem, society, music, song, poet, story, love, culture, drawing
Automobile	showroom, scooter, car, bike, electrical, dealer, TVS, racing, Toyota
Bank	bank, insurance, dollar, euro, payment, governor, account, money
Business	bank, rupees, percentage, company, factory, sale, loan, market
Crime	crime, killer, gold, currency, bail, arrest, police, death
Employment	workers, help, company, abroad, U.K., Qatar, problem, law, committee
Economy	year, business, financial, project, invest, industry, price, tax, money
Education	school, student, education, local, major, district, college, board
Entertainment	film, song, play, artist, performer, movie, music, director, program
Health	problem, treatment, disease, medicine, doctor, study, cancer, consume
Lifestyle	fashion, hairstyle, treatment, movies, costume, exhibition
Literature	book, poem, language, poet, creation, culture, music, writer, novel
Market	service, percentage, discount, account, receive, sale, brand, material
Opinion	political, parliament, USA, organization, people, government
Politics	government, voting, committee, state, congress, communist, discussion
Society	district, office, program, village, arrest, hospital, municipality
Sports	game, player, league, club, goal, run, win, national, coach, team
Technology	company, mobile, Facebook, apps, telecom, online, internet, Google
Tourism	flight, travel, tourist, airlines, airways, hotel, ship, mountain
World	UK, USA, Russia, country, Trump, vaccine, army, nation, parliament

*Note.* The top frequent words shown in this table are translated into English from Nepali.

In addition to that, to have a better understanding of our dataset and check the correctness of category labels, text from each of the classes are combined and the most frequent words in those classes are analyzed. For this purpose, stemming was done and stop words were removed to highlight category-specific words. The top frequent words generated by this process is shown in Table 3.2. By analyzing this, we could see that some categories like *Market/Business/Bank/Economy*, *Opinion/Society/Politics* were related to each other and consisted of similar most-frequent words which are more

likely to be mis-classified by the models. Therefore, in order to check the robustness of models under study, we created one more subset of this dataset having 16 classes with the 4 ambiguous categories removed namely, *Business*, *Economy*, *Opinion* and *Society*. By evaluating our models on datasets with different numbers of classes, we contrasted the classification performance of typical context-independent machine learning models with context-dependent transformer-based language models.

### 3.3 Dataset

#### 3.3.1 Creation

As mentioned earlier, our main dataset consists of 30,000 news articles categorized into 20 different classes with 1,500 articles in each class. We denote this dataset as *Nepali 20 newsgroups (np20ng)* and data size of **30K**. In this third phase, we create one more subset of our main dataset *np20ng* by removing previously identified four ambiguous categories, namely *Business*, *Economy*, *Opinion* and *Society* and denote it as *Nepali 16 newsgroups (np16ng)*. So, *np16ng* consists of 24,000 news articles altogether categorized into 16 different classes and similarly denoted data size as **24K**. By fine-tuning the transformers models on these two datasets, we compared the model performances based on how accurately they can classify similar articles into correct categories. Moreover, see the robustness of models when data size is limited, we additionally sampled datasets with 10,000, 5,000 and 1,000 total news articles from *np20ng* and denote them as **10K**, **5K** and **1K** respectively.

#### 3.3.2 Splits

The datasets are shuffled and split into training, validation and testing sets in the ratio of 80:10:10 using stratified sampling to preserve the data-balance among each class in different splits. Splits were performed prior to any pre-processing to prevent potential data leakage.

### 3.4 Tokenization

Our dataset splits were tokenized and assigned a unique value to each token in a text sequence. The tokenization process depends on the type of model we use to tokenize (Table 3.4) the input sequence. Depending on the model, special tokens are added such as for BERT, [CLS] and [SEP] tokens represent beginning and end of sentence whereas for RoBERTa, <s> and </s> respectively.

For our experiments, we set the maximum sequence length to 512 (including two special tokens), so first, the text sequence is tokenized using pre-trained tokenizers into the length of 512 different tokens. Text sequences longer than 510 tokens are simply truncated whereas for those with shorter length, padding tokens are added by the tokenizer to fulfill the maximum input length of the models. For better model optimization and efficient training, dynamic padding is also implemented on the text sequences, which basically adjusts the input length to the maximum sequence length in the batch that is being trained on regardless of the length of 512. Attention masks were then computed to tell the model which part of the text sequence is required to be used and ignore the padding tokens. To summarize, attention masks, token ids and label-encoded category labels are compiled in the tokenized dataset.

### 3.5 Pre-trained tokenizers

Depending on the model architecture, different pre-trained tokenizers use different tokenization algorithms for tokenizing texts (Table 3.4). Similarly, depending on different pre-training approaches, these tokenizers handle the text sequence differently such as for Nepali language, some tokenizers preserve the important vowel modifier symbols but some do not (Table 3.3).

**Table 3.3**

*Handling of Nepali special characters and digits*

Tokenizer	Vowel Modifiers	Half character symbol	Nepali digits
BERT	No	No	No
RoBERTa	-	-	-
DistilBERT	Yes	Yes	Yes
DeBERTa	Yes	Yes	Yes
mBERT	Yes	No	Yes
XLM-RoBERTa	Yes	Yes	Yes
HindiRoBERTa	-	-	-

*Note.* As RoBERTa and HindiRoBERTa use Byte-level BPE tokenizer, and all the tokens and encoded, we cannot determine how tokens are tokenized.

### 3.5.1 WordPiece Tokenizer

WordPiece is a sub-word tokenization method where a word or a token is split into commonly occurring suffixes or word pieces. Nepali BERT model and mBERT use WordPiece tokenizers. In Figure 3.6, we can observe some of the similar tokens are being taken as the sub-words, but others are different, it is mainly due to the pre-training process of these tokenizers. They are trained in different corpora and use different pre-processing techniques.

**Figure 3.6**

*WordPiece tokenization*

**Sentence:** काठमाडौंको नागार्जुनमा चितुवाको आक्रमणबाट पाँच जना घाइते भएका छन् ।

**Rajan/NepaliBERT**

काठमाडौंको नागा ##र ##जनमा चित ##वाको आक ##रमण ##बाट पाच जना घाइत भएका छन ।

**bert-base-multilingual-uncased**

का ##ठ ##मा ##ड ##ौ ##को न ##ाग ##ार ##जन ##मा च ##ित ##वा ##को आ ##करम ##ण ##बाट  
पाच जन ##ा घ ##ा ##इ ##त भ ##एका छन ।

### 3.5.2 SentencePiece Tokenizer

In sentence-piece tokenizer, an underscore-like character ‘\_’ (U+2581), “Lower One Eighth Block” denotes the start of the token and whitespaces are used to split the token into sub-words. For our language models under study, two Nepali tokenizers DistilBERT and DeBERTa and one multilingual XLM-RoBERTa are trained using a Sentence Piece Model (SPM). In Figure 3.7, we see the words which are possible suffixes in Nepali are split using the whitespace characters and SentencePiece tokenizer seems to perform better than Nepali WordPiece tokenizer.

**Figure 3.7**

*SentencePiece tokenization*

**Sentence:** काठमाडौंको नागार्जुनमा चितुवाको आक्रमणबाट पाँच जना घाइते भएका छन् ।

**Sakonii/distilbert-base-nepali**

\_काठमाडौंको \_नागार्जुन मा \_चितुवा को \_आक्रमण बाट \_पाँच \_जना \_घाइते \_भएका \_छन् \_।

**Sakonii/deberta-base-nepali**

\_काठमाडौंको \_नागार्जुन मा \_चितुवा को \_आक्रमण बाट \_पाँच \_जना \_घाइते \_भएका \_छन् \_।

**xlm-roberta-base**

\_काठमाडौं को \_ना गा र्ज ुन मा \_ चित ुवा को \_आक्रमण बाट \_पाँच \_जना \_घाइते \_भएका \_छन् \_।



### 3.5.3 Byte-level Byte Pair Encoding

A Byte-level BPE Gage (1994) tokenizer builds a vocabulary by merge rules. It marks the beginning of a new token with a special encoded unicode character like, ‘Ġ’ (a G with a dot). The RoBERTa tokenizer from Huggingface also use other characters to encode tokens such as ‘Ĵ’ (u/0134) ‘Ĺ’ (u/0139), and ‘Ɔ’ (u/0164). Nepali and Hindi RoBERTa model uses Byte-level BPE tokenizer and we can observe in Figure 3.8 that every character or symbol in the sentence is encoded into bytes-representation.

**Figure 3.8**

### Byte-Level BPE tokenization

**Sentence:** काठमाडौँको नागार्जुनमा चितुवाको आक्रमणबाट पाँच जना घाइते भएका छन् ।

amitness/nepbert

[illegible]

**flax-community/roberta-hindi**

ànḵ àn ¼ ànḵàḿ àn ¼ ànḵ àḵḵàḿ ànḵ àḵḵḵ Ḡàḿ àn ¼ ànḵ àn ¼ àḵ àḵḵ ànḵ àḵḵ àn ¼ Ḡàḵ àn àḵ àḵḵ àḵḵ Ḡàḵḵàḵ àḵḵ àḵ àḵḵàḵàḵ àn ¼ àḵ Ḡ àḵ àḵàḵàḵ ànḵ Ḡàḵḵàḵ àn ¼ Ḡàḵ àn ¼ àḵḵàḵ àḵḵ ḠàḵNàḵàḵ àn ¼ Ḡàḵḵàḵ àḵḵ Ḡàḵ àḵ

### 3.6 Pre-trained transformer models

Multi-lingual models address a problem where there is no pre-trained model available for a particular language and use cross-lingual transfer learning to solve the problem. Similarly in this study, we used mBERT and XLM-RoBERTa models to compare performances with monolingual pre-trained Nepali models. In addition to that we used a model from different language, i.e. Hindi language (India) to see its classification performance on Nepali text.

Nepali transformer models in this study are pre-trained on several corpus or datasets such as OSCAR<sup>3</sup>, CC-100<sup>4</sup>, Large Scale Nepali Corpus (Lamsal, 2020) and Wikipedia. Here is the brief overview of corpus or datasets:

A Large Scale Nepali Corpus contains unlabelled Nepali text from different news domains and consists of 90 million words (6.5 million sentences). *NepaliText*<sup>5</sup> is a language modeling dataset consisting of over 13 million Nepali text sequences extracted

<sup>3</sup><https://huggingface.co/datasets/oscar-corpus/OSCAR-2109>

<sup>4</sup><https://huggingface.co/datasets/cc100>

<sup>5</sup><https://huggingface.co/datasets/Sakonii/nepalitext-language-model-dataset>

from other sources like OSCAR, CC-100 and Wikipedia. CC-100 (Common Crawl) dataset consists of monolingual data for 100+ languages constructed using the urls and paragraph indices provided by CC-Net (Wenzek et al., 2020) repository by processing Commoncrawl snapshots. OSCAR (Open Super-large Crawled Aggregated coRpus) is a huge multilingual corpus obtained by language classification and filtering of the Common Crawl corpus using the Ungoliant architecture, a high-performance pipeline that provides tools to build corpus generation pipelines from CommonCrawl (Abadji, Suárez, Romary, & Sagot, 2021). This corpus consists of 3.7 GB of 177 million Nepali words (391,947 documents). This corpus is curated as a dataset by Huggingface and can be downloaded from here<sup>3</sup>. Wikipedia consists of articles in Nepali which are different from typical news articles. Incorporating more dimensions of language helps models to understand more sophisticated linguistic patterns.

**Table 3.4**

*Summary of pre-trained tokenizers and models*

Model	Tokenizer ID	Tokenizer	Vocab size	Pre-train data	Params
BERT	Rajan/NepaliBERT	WordPiece	50000	LSNC, OSCAR	82M
RoBERTa	amitnesh/nepbert	Byte-level BPE	52000	CC-100	83.5M
DistilBERT	Sakonii/distilbert-base-nepali	SentencePiece	24581	OSCAR, CC-100, Wikipedia	67M
DeBERTa	Sakonii/deberta-base-nepali	SentencePiece	24581	OSCAR, CC-100, Wikipedia	139M
mBERT	bert-base-multilingual-uncased	WordPiece	105879	Wiki, 102 languages	110M
XLM-RoBERTa	xlm-roberta-base	SentencePiece	250002	CC-100, Wiki	278M
HindiRoBERTa	flax-community/roberta-hindi	Byte-level BPE	50265	OSCAR, mC4, indic-nlp	125M

*Note.* *Tokenizer ID* refers to the identifier of the model in *HuggingFace*. This identifier can be used to load a pre-trained model and pre-trained tokenizer using respective classes for sequence classification. *LSNC* stands for *Large Scale Nepali Corpus* and *CC* stands for *Common Crawl*.

### 3.6.1 BERT

BERT is a transformer-based language model trained by Google AI (Devlin et al., 2018) on MLM and Next Sentence Prediction (NSP) objective. Nepali BERT model (*Rajan/NepaliBERT*) in this research is pre-trained on 6.7M lines of text from Large Scale Nepali Corpus & OSCAR Nepali corpus and has 82M parameters. It uses word-piece tokenizer with a vocabulary size of 50,000 tokens.

### 3.6.2 RoBERTa

RoBERTa is a transformer-based language model trained by Facebook AI (Liu et al., 2019). It addresses the known limitations of BERT by incorporating dynamic masking, where each single example presented to the model is masked randomly, providing some diversity in the training process and also making the training process a lot faster than BERT. Nepali RoBERTa model (*amitness/nepbert*) in this research is pre-trained on Nepali CC-100 dataset with 12M sentences using a Google Colab’s Tesla V100 GPU and has 83.5M parameters. It uses Byte-level BPE tokenizer and has a vocabulary size of 52,000 tokens.

### 3.6.3 DistilBERT

DistilBERT (a distilled version of BERT) is a transformer-based language model which is much smaller, lighter, cheaper and faster than BERT. Nepali DistilBERT model (*distilbert-base-nepali*) in this research is pre-trained on *NepaliText*<sup>6</sup> dataset consisting over 13M Nepali text sequences from OSCAR and Nepali CC-100 dataset using MLM objective and has 67M parameters. It uses sentence-piece tokenizer and has a vocabulary size of 24,581 tokens.

### 3.6.4 DeBERTa

DeBERTa is a novel transformer language model developed by Microsoft to give significant improvement over Google’s BERT model and Facebook’s RoBERTa model by introducing two improvement techniques, disentangled attention mechanism and an enhanced mask decoder (He, Liu, Gao, & Chen, 2020). Nepali DeBERTa model (*Sakonii/deberta-base-nepali*) in this research is pre-trained on *NepaliText*<sup>6</sup> dataset consisting over 13M Nepali text sequences. It uses Sentence Piece Model (SPM) for text tokenization and handles text sequences upto 512 tokens. It has a vocabulary size of 24,581 tokens and the model has 139M parameters. The creators of this model Maskey et al. (2022) have discussed that the model may not perform satisfactorily on shorter sequences.

### 3.6.5 mBERT

Multilingual BERT (mBERT) (Devlin et al., 2018) is a multilingual variant of BERT model and is pre-trained from concatenated monolingual Wikipedia corpora in 102 lan-

---

<sup>6</sup><https://huggingface.co/datasets/Sakonii/nepalitext-language-model-dataset>

guages (uncased variant) on MLM and NSP tasks. It is pre-trained using a shared word piece vocabulary of 105,879 tokens and has 110M parameters. We use an uncased version of the mBERT model in this research.

### **3.6.6 XLM-RoBERTa**

XLM-RoBERTa (Conneau et al., 2019) is a multilingual variant of the RoBERTa model released by Facebook AI team which is trained on over 100 different languages. It has a vocabulary size of 250,002 and is trained on 2.5TB of CommonCrawl data and Wikipedia. The architecture is the same as RoBERTa but the training procedure differs. It leverages the concept of cross-lingual transfer learning where a multi-lingual model is trained on a particular task for the English language and that model is used to solve a task in a different language. XLM-RoBERTa uses sentence piece tokenizer for tokenization. We use the base version of XLM-RoBERTa in this research.

### **3.6.7 HindiRoBERTa**

HindiRoBERTa<sup>7</sup> is a monolingual RoBERTa model pre-trained on Hindi language. It is trained on combination of several datasets such as OSCAR, mC4, IndicGLUE, Samanantar, Hindi Text Short and Large Summarization Corpus, Hindi Text Short Summarization Corpus and Old Newspapers Hindi datasets using Google Cloud Engine TPU. It uses Byte-level BPE tokenizer and has a vocabulary size of 50,265.

## **3.7 Baseline algorithms**

We evaluated the transformer models with three ML and one DL algorithms as baselines. They are as follows:

### **3.7.1 Multinomial Naive Bayes (MNB)**

MNB is a probabilistic approach for text classification tasks mostly preferred due to its fast training and prediction speeds. Moreover, it is not memory intensive. In our study, we use MNB as a baseline and select the best model through nested cross validation procedure.

### **3.7.2 Random Forest (RF)**

Random forest is a supervised machine learning algorithm, which is an ensemble of large numbers of individual decision trees preferred mostly due to its high prediction accuracy,

---

<sup>7</sup><https://huggingface.co/flax-community/roberta-hindi>

but it is relatively slow to train. In our study, we used a couple of hyperparameters such as *number of estimators and maximum features* to increase the overall prediction accuracy. Nested cross validation with grid search was used to obtain optimal hyperparameters.

### **3.7.3 Support Vector Machine (SVM)**

SVM is a supervised machine learning algorithm, which performs classification based on the hyperplane that differentiates two or more classes. It is mostly preferred due to its high prediction accuracy and is good for text classification. However, it can be memory intensive for larger datasets. In this study, several hyperparameters were tuned using Grid search to get the optimal parameters such as *kernel, gamma, penalty (C), etc*

### **3.7.4 Bidirectional Long Short Term Memory (Bi-LSTM)**

Bi-LSTMs are special types of Recurrent Neural Networks (RNNs) which solve the vanishing gradient problem from traditional RNNs. This architecture is preferred due to its capability to handle contexts of long sequential data by processing the input data in both forward and backward directions. However, its performance relies on the amount of input data. In this study, we train Bi-LSTM network using Nepali Fasttext embeddings (Grave, Bojanowski, Gupta, Joulin, & Mikolov, 2018) with softmax as a classifier.

## **3.8 Fine-tuning and classification**

We selected the optimal hyperparameters, fine-tuned our tokenized dataset using seven pre-trained models and evaluated the test set performance. For seven transformer-based models, the optimal hyperparameters and model checkpoint were identified based on the best validation loss and the best model was used to evaluate the test set performance. A series of experiments using varied learning rates such as  $2e-05$ ,  $3e-05$ ,  $5e-05$ ,  $5e-06$ , etc. were performed for every model. An optimal learning rate was found to be  $2e-05$  and training batch size to be 32. For three machine learning models, Grid Search Nested Cross Validation was used to select the optimal hyperparameters and TF-IDF vectorizer was used for feature extraction. For RNN, Bi-LSTM layers were used with Nepali FastText embeddings and softmax as activation function.

## CHAPTER 4

### RESULTS AND DISCUSSION

The evaluation of transformer-based language models were done in two different ways. Firstly, an intuitive evaluation was done by determining how the models fulfill simple Masked Language Modeling (MLM) tasks. Despite the fact that this method is a rapid way to see the usefulness of a transformer model, it lacks technical proof to justify its effectiveness. Therefore, secondly, the models were fine-tuned on a specific task of multi-class text classification to perform a more technical evaluation of their performance across various scenarios.

#### 4.1 Experiment 1: Intuitive Evaluation

This is a quick and relatively simple way of evaluating transformer models based on intuition. As all of the studied transformer models were pre-trained on the MLM objective, they were put to a test where they were given Nepali sentences with a masked word and had to predict it. Few top words predicted by some models were semantically similar to each other while some model's predictions were meaningless. For example, in Figure 4.1, given a short sentence in Nepali (NE) as input where the country "Thailand" is a context and a masked word being the name of the capital city "Bangkok", the top words predicted by seven transformer models are shown, with the majority of models able to predict the names of cities, some with the exact city as well.

Another example in Figure 4.2 is a slightly lengthy sentence in which the first part of the sentence expresses a context of a "worrying sentiment" caused by a pandemic, while a word in the second part of the sentence was masked with the intention of conveying the sentiment caused by "devastating war damage". Likewise, the majority of models predicted sentiment-based words with semantically similar meanings.

In order to obtain the mean performance of the models under study, a more detailed test was performed on a total of 50 Nepali sentences. Based on how semantically similar the predicted word was from the masked word, the models were given 4 different ratings. Figure 4.3 shows the proportion of ratings that transformer models predicted to be exact or nearly meaningful words according to the context of a sentence.

In summary, the models BERT, DistilBERT, and DeBERTa for monolingual and XLM-

**Figure 4.1**

*Example of masked word predictions in a short sentence.*

**Sentence (NE):** थाइल्याण्डको राजधानी बैकक हो ।

**Sentence (EN):** The capital of Thailand is Bangkok.

**Masked Sentence (NE):** थाइल्याण्डको राजधानी <mask> हो ।

**Masked Sentence (EN):** The capital of Thailand is <mask>.

Top words predicted by the models (descending order by scores)

<b>BERT</b>	<b>RoBERTa</b>	<b>DistilBERT</b>	<b>DeBERTa</b>
पनि (also)	एक (one)	कालालम्पुर (Kuala Lumpur)	<b>बैकक (Bangkok)</b>
राजधानी (capital)	त (meaningless)	सहर (city)	सहर (city)
यही (this)	समय (time)	<b>बैकक (Bangkok)</b>	शहर (city)
पोखरा (Pokhara)	सहर (city)	काठमाडौं (Kathmandu)	पनि (also)
प्रदेश (state)	शहर (city)	नयाँदिल्ली (New Delhi)	थाइल्याण्ड (Thailand)

<b>mBERT</b>	<b>XLmRoBERTa</b>	<b>HindiRoBERTa</b>
नेपाल (Nepal)	काठमाडौं (Kathmandu)	शहर (city)
दिल्ली (Delhi)	काठमाण्डौ (Kathmandu)	न (meaningless)
राजधानी (capital)	काठमाडौं (Kathmandu)	कब (when)
शहर (city)	नेपाल (Nepal)	नगर (city)
पनि (also)	यही (this)	और (or)

RoBERTa for multilingual performed relatively well in terms of predicting syntactically and semantically significant words, however RoBERTa and mBERT did not. Furthermore, the HindiRoBERTa model was unable to predict Nepali words, which is acceptable for this scenario due to the fact that it is a monolingual Hindi model. The mean cosine similarity scores are reported in Figure 4.4. Although certain models did not perform well in this evaluation method, they are evaluated more technically in Section 4.2.

## 4.2 Experiment 2: Evaluation on text classification dataset

### 4.2.1 Hyperparameter Selection

For transformer models, hyperparameters were selected based on a series of experiments. After experimenting with a few settings in the initial phase, the most important parameter was found to be the learning rate and training batch size accordingly. Moreover, the training batch size was depended on the GPU, i.e. NVIDIA RTX A6000 (48GB). The total train epochs were set to 5 for bigger-sized models like mBERT and XLM-RoBERTa, and to 10 for other models. However, once the validation loss stopped improving, the training process was stopped by early stopping callback. Table 4.1 gives a summary of hyperparameters selected to fine-tune these models for *np20ng\_30K*

**Figure 4.2**

*Example of masked word predictions on a slightly long sentence.*

**Sentence (NE):** कोभिड महाव्याधिको पीडाबाट मुक्त नहुँदै मानव समाजलाई यतिबेला युद्धको विध्वंसकारी क्षतिको चिन्ताले **चिन्तित** बनाएको छ ।

**Sentence (EN):** Human society is not freed from the pain of the covid pandemic, but is now **worried** about the devastating damage of war.

**Masked Sentence (NE):** कोभिड महाव्याधिको पीडाबाट मुक्त नहुँदै मानव समाजलाई यतिबेला युद्धको विध्वंसकारी क्षतिको चिन्ताले **<mask>** बनाएको छ ।

**Masked Sentence (EN):** Human society is not freed from the pain of the covid pandemic, but is now **<mask>** about the devastating damage of war.

Top words predicted by the models (descending order by scores)

<b>BERT</b>	<b>RoBERTa</b>	<b>DistilBERT</b>	<b>DeBERTa</b>
निराश (disappointed) प्रभावित (impacted) <b>चिन्तित (worried)</b> खुसी (happy) कमजोर (weak)	सहज (easy) घर (house) सफल (successful) असफल (unsuccessful) सडक (road)	<b>चिन्तित (worried)</b> आक्रान्त (agonizing) स्तब्ध (shocked) त्रसित (freaked out) निराश (disappointed)	आक्रान्त (agonizing) <b>चिन्तित (worried)</b> मुक्त (free) ग्रसित (afflicted) पीडित (victimized)
<b>mBERT</b>	<b>XLNet</b>	<b>HindiRoBERTa</b>	
[UNK] (meaningless) न (meaningless) रपमा (in a way) " (meaningless) पनि (also)	बाध्य (compulsive) गम्भीर (serious) कमजोर (weak) पीडा (pain) दुःख (sadness)	कर (meaningless) क (meaningless) न (meaningless) न (meaningless) स (meaningless)	

dataset.

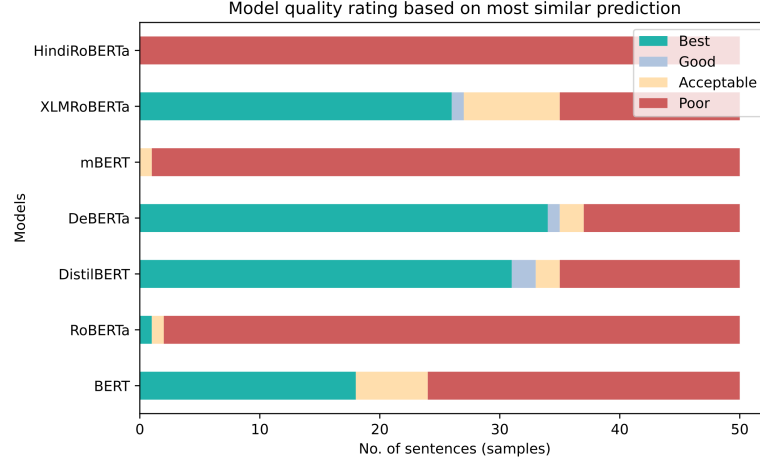
For Bi-LSTM model, Nepali FastText embeddings (Grave et al., 2018) of 300 dimensions were used to construct an embedding matrix with maximum length of input sequence set to 512. Two bidirectional LSTMs each having 512 units were connected together with the dropout of 0.2 each followed by two fully connected hidden dense layers having 128 and 64 units. The hidden layers used the ReLU activation function. Categorical cross entropy was used as a loss function with Adam optimizer was used. There were a total of 66.83M parameters out of which 2.79M parameters were trainable. The model was trained for 50 epochs with early stopping for best validation loss. All Bi-LSTM models were trained using the Tesla T4 (12 GB) GPU from Google Colab. An overview of its architecture is shown in Figure 4.5. The hyperparameters used to train this model are shown in Table 4.2 with some additional training results.

For machine learning algorithms, a widely used vectorizer, TF-IDF, was used for feature extraction with the n\_gram range of 1 to 4 and maximum of 2000 feature vectors. Nested cross validation using grid search was performed using 3 inner loops and 5 outer loops to find optimal hyperparameters. This procedure was applied in each of the data



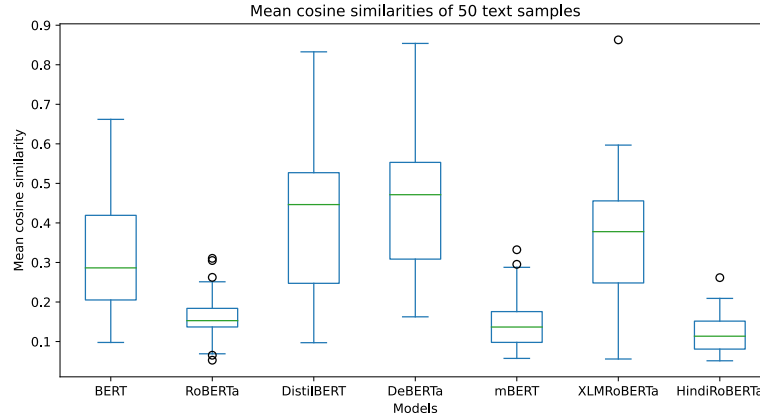
**Figure 4.3**

*Model quality ratings obtained from evaluation done on 50 sentences*



**Figure 4.4**

*Distribution of mean cosine similarity scores across models*



sizes separately to get the best parameters for respective data sizes. As nested cross validation procedure is quite expensive when there are more parameters to check, we just experimented with a few parameters, yet it took a considerably large amount of time, i.e approximately 6.3 hours for *np20ng\_30K* dataset. The training split was used to fit the ML models, while validation and test splits were used to evaluate the model. The parameters obtained are summarized in Table 4.3 and their overall performance on the test set is shown in Figure 4.7. The specification of the machine used for performing nested CV and model fitting is Intel(R) Core(TM) i5-9400F Hexa-core CPU 2.90GHz.

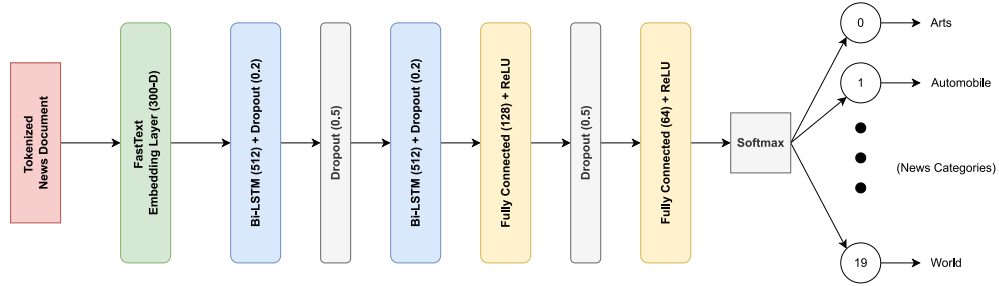
#### **4.2.2 Performance on Test split by number of classes**

As seen in Figure 4.6, all the models under study showed higher classification accuracy for the 16-class dataset compared to its 20-class variant. While monolingual DistilBERT

**Table 4.1***Hyperparameters and training time of transformer models for np20ng\_30K*

	Train batch size	Learning rate	Epochs trained	Training time
BERT	32	2e-05	3.73	32m 37s
RoBERTa	32	2e-05	5.33	43m 11s
DistilBERT	32	2e-05	5	40m 16s
DeBERTa	8	2e-05	1.6	1h 27m 37s
mBERT	8	2e-05	2	1h 32m 57s
XLM-RoBERTa	32	2e-05	5.33	1h 16m 44s
HindiRoBERTa	32	5e-05	3.2	34m 56s

*Note.* The training times depend on the batch size set for training. Lower batch sizes were taken due to limited hardware resources.

**Figure 4.5***The Bi-LSTM architecture*

achieved highest accuracy of 91.33% on the 16-class dataset and 86.63% on the 20-class dataset, multilingual model XLM-RoBERTa outperformed several monolingual models namely BERT, RoBERTa, and DeBERTa on both 16-class and 20-class datasets, reaching an accuracy of 90.46% and 86.7% respectively. Furthermore, the baseline model SVM achieved an accuracy of 87.5% on 16-class dataset exceeding monolingual BERT and RoBERTa by slight margins, including another DL baseline Bi-LSTM which scored only 85.12%. Although mBERT and HindiRoBERTa were able to get an accuracy above 80% on 16-class dataset, their performance dropped significantly on 20-class dataset such that MNB and RF also outperformed them. Table 4.4 shows the overall accuracy scores for both the datasets with 30K rows.

**Table 4.2***Hyperparameters and training time of Bi-LSTM model*

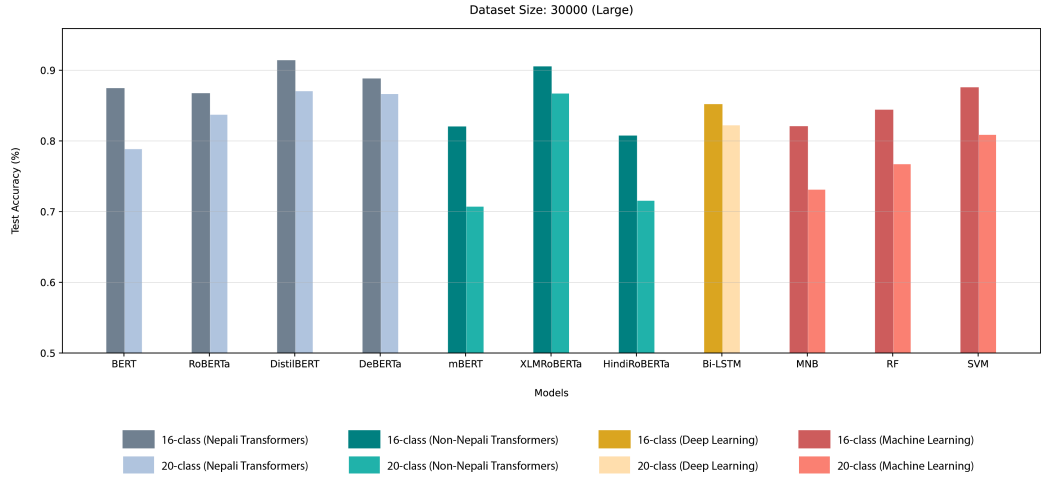
	Vocab size	Embedding dim	Learning rate	Batch size	Best epoch	Training time
np16ng_30K	179943	300	0.001	64	12	35m 36s
np20ng_30K	213462	300	0.001	64	13	39m 28s
np20ng_10K	109485	300	0.001	64	13	16m 41s
np20ng_5K	70822	300	0.001	64	21	14m 49s
np20ng_1K	25591	300	0.0001	32	25	23m 4s

**Table 4.3***Hyperparameters and training time of ML models*

	MNB		RF		SVM		Nested CV Duration
	<i>alpha</i>	<i>max_depth</i>	<i>n_estimators</i>	<i>C</i>	<i>kernel</i>	<i>gamma</i>	
np16ng_30K	0.1	200	1000	10	RBF	2	6.86 hr
np20ng_30K	0.01	300	1000	50	RBF	1	6.3 hr
np20ng_10K	1	100	1000	10	RBF	1	1.12 hr
np20ng_5K	0.1	None	1000	10	RBF	1	19.65 min
np20ng_1K	1	None	500	10	RBF	1	2.86 min

**4.2.3 Performance on Test split by data size**

As seen in Figure 4.7, the majority of models for the 20-class dataset exhibited a positive correlation between test accuracy and data size; that is, as the amount of data was increased, so did the models' accuracy. Most monolingual models and XLM-RoBERTa for multilingual models outperformed baseline models when data size is 5,000 and higher. However, ML baselines RF and SVM outperformed BERT, RoBERTa, mBERT, and Bi-LSTM on datasets with 1,000 rows, whereas RF and SVM outperformed them scoring an accuracy of 68% and 69% respectively. Furthermore, SVM performed exceptionally well as data amount was increased. Although Bi-LSTM model performance was comparatively higher for larger datasets, it did not perform well on the smallest dataset. Similarly, mBERT and HindiRoBERTa performed poorly in comparison to ML baseline models regardless of the data size.

**Figure 4.6***Test accuracy scores based on number of classes***Table 4.4***Test set accuracy on np16ng\_30K and np20ng\_30K dataset*

	BERT	RoBERTa	DistilBERT	DeBERTa	mBERT	XLm-R	Hindi-R	Bi-LSTM	MNB	RF	SVM
np16ng_30K	0.8738	0.8667	<b>0.9133</b>	0.8875	0.8196	0.9046	0.8067*	0.8512	0.82	0.8433	0.875
np20ng_30K	0.7883	0.837	<b>0.8703</b>	0.8663	0.707*	0.867	0.7153	0.822	0.731	0.767	0.8086

*Note.* The highest test accuracy scores are bold-faced and the lowest scores are marked with an ‘\*’. XLM-R and Hindi-R stands for XLM-RoBERTa and HindiRoBERTa respectively.

#### 4.2.4 Weighted precision analysis on Test split

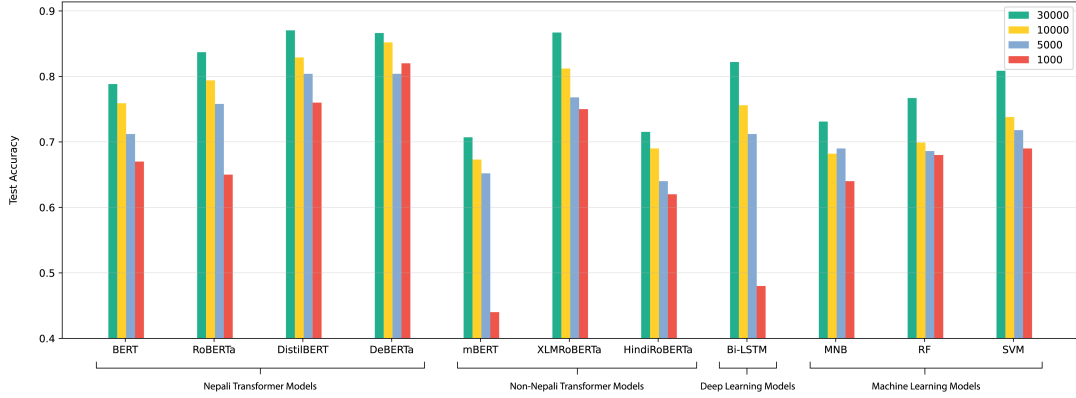
Precision metric is used to analyze how correct are the predictions given by the models. The category-wise results for each of the models on *np20ng\_30K* and *np16ng\_30K* dataset are shown in Table 4.6 and 4.7 respectively.

Nepali DeBERTa model was able to achieve best precision in 7 classes followed by BERT, DistilBERT and XLM-RoBERTa with highest precision in 4 classes each. RoBERTa, BiLSTM and SVM got highest precision for 1 class each. Similarly, HindiRoBERTa, mBERT and MNB were the models with worst precision scores among all the models. Some categories like *Business*, *Economy* and *Society* have relatively low weighted precision values for each model, which were also considered to be the ambiguous categories. Therefore, most of the models were not able to correctly classify documents under these categories.

We can observe overall increase in weighted precision values in *np16ng\_30K* dataset

**Figure 4.7**

Test accuracy scores based on size of data on np20ng dataset

**Table 4.5**

Test set accuracy on 20-class dataset (np20ng) with varying data sizes

	BERT	RoBERTa	DistilBERT	DeBERTa	mBERT	XLm-R	Hindi-R	Bi-LSTM	MNB	RF	SVM
<b>30K</b>	0.7883	0.837	<b>0.8703</b>	0.8663	0.707*	0.867	0.7153	0.822	0.731	0.767	0.8086
<b>10K</b>	0.759	0.794	0.829	<b>0.852</b>	0.673*	0.812	0.69	0.756	0.682	0.699	0.738
<b>5K</b>	0.712	0.758	<b>0.804</b>	<b>0.804</b>	0.652*	0.768	0.64	0.712	0.69	0.686	0.718
<b>1K</b>	0.67	0.65	0.76	<b>0.82</b>	0.44*	0.75	0.62	0.48	0.64	0.68	0.69

*Note.* The highest test accuracy scores are bold-faced and lowest scores are marked with an asterisk. XLM-R and Hindi-R stands for XLM-RoBERTa and HindiRoBERTa respectively.

(Table 4.7) compared to that of *np20ng\_30K* dataset (Table 4.6). When ambiguous categories were removed, there seemed to have acceptable ranges of precision scores for all the models. The *Sports* category constitutes the highest precision scores with all values over 90%. The findings of the overall baseline model demonstrated that ML and DL baselines are highly resilient in categorizing Nepali text and are suitable for situations when training time is critical. mBERT model still has moderate performance on 16-class dataset, and HindiRoBERTa did not manage to get acceptable precision scores for most of the classes.

### 4.3 Discussion

#### 4.3.1 Important findings

To evaluate the performance of context-dependent language models, we conducted experiments on datasets with varying numbers of classes to determine whether the model can effectively classify news documents despite the ambiguity of the document classes.

**Table 4.6***Weighted average precision for np20ng\_30K dataset*

Category	BERT	RoBERTa	DistilBERT	DeBERTa	mBERT	XLM-R	Hindi-R	Bi-LSTM	MNB	RF	SVM
Arts	0.718	0.827	<b>0.871</b>	0.802	0.728	0.868	0.589*	0.779	0.72	0.74	0.82
Automobile	0.879	0.878	<b>0.886</b>	0.848	0.757	0.853	0.834	0.833	0.73*	0.8	0.84
Bank	0.796	<b>0.898</b>	0.858	0.865	0.558*	0.888	0.722	0.824	0.69	0.78	0.8
<i>Business</i>	0.566	0.65	0.806	0.7	0.478	<b>0.814</b>	0.416*	0.62	0.46	0.66	0.52
Crime	0.779	0.95	0.953	<b>0.966</b>	0.856	0.892	0.688*	0.906	0.78	0.81	0.87
<i>Economy</i>	0.422	0.634	0.724	<b>0.776</b>	0.335	0.686	0.413*	0.651	0.49	0.59	0.57
Education	<b>0.911</b>	0.89	0.917	0.895	0.724*	0.907	0.891	0.849	0.82	0.75	0.87
Employment	<b>0.888</b>	0.842	0.881	<b>0.888</b>	0.789	0.87	0.826	0.767*	0.86	0.82	0.87
Entertainment	0.879	0.906	0.885	<b>0.972</b>	0.827	0.922	0.803*	0.922	0.85	0.91	0.88
Health	<b>0.85</b>	0.816	0.815	0.811	0.796	0.84	0.693*	0.799	0.72	0.74	0.77
Lifestyle	0.845	0.871	0.944	0.937	0.762	<b>0.951</b>	0.71	0.807	0.69*	0.76	0.85
Literature	0.759	0.81	0.818	0.798	0.669*	<b>0.856</b>	0.711	0.769	0.76	0.83	0.85
Market	0.744	0.816	0.857	0.838	0.804	0.824	0.784	0.831	0.72*	0.8	<b>0.87</b>
<i>Opinion</i>	0.878	0.897	0.928	<b>0.955</b>	0.771	0.953	0.739*	0.925	0.83	0.87	0.94
Politics	0.862	0.856	0.868	<b>0.933</b>	0.702*	0.893	0.817	0.899	0.75	0.73	0.87
<i>Society</i>	0.595	0.699	<b>0.781</b>	0.749	0.538	0.766	0.456*	0.804	0.62	0.61	0.66
Sports	0.954	0.948	0.973	0.974	0.937	0.98	0.966	<b>0.993</b>	0.95	0.89*	0.96
Technology	0.857	0.853	0.867	<b>0.913</b>	0.786	0.857	0.832	0.822	0.75*	0.76	0.86
Tourism	0.762	0.864	<b>0.932</b>	0.925	0.722	0.869	0.719*	0.851	0.73	0.74	0.82
World	<b>0.879</b>	0.85	0.838	0.861	0.769	0.848	0.722	0.805	0.66*	0.74	0.73

*Note.* The bold-faced values indicate the highest weighted precision obtained for each class whereas lowest values are marked with an ‘\*’. The categories which are *italicized* are the ones not present in the 16-class dataset. XLM-R and Hindi-R stands for XLM-RoBERTa and HindiRoBERTa respectively.

Consequently, as compared to the baselines, transformer-based language models were rather robust and effective at classifying news documents from both 16-class and 20-class datasets with the desired level of accuracy. According to the overall study, when given an adequate amount of data, transformer models were better than traditional ML and DL algorithms. Although we anticipated that, as a result of language-specific pre-training, monolingual large pre-trained transformer models would be more effective than multilingual models, the XLM-RoBERTa model for Nepali text classification was able to outperform several monolingual models by a significant margin. As a multilingual model, mBERT was unable to demonstrate a substantial improvement over any other model under study. It could be because the Nepali language is underrepresented in the multilingual BERT (Wu & Dredze, 2020).

We performed an intuitive evaluation of pre-trained transformer models by allowing

**Table 4.7***Weighted average precision for np16ng\_30K dataset*

Category	BERT	RoBERTa	DistilBERT	DeBERTa	mBERT	XLM-R	Hindi-R	Bi-LSTM	MNB	RF	SVM
Arts	0.727	0.783	0.81	0.784	0.723	<b>0.824</b>	0.653*	0.8	0.69	0.75	0.75
Automobile	0.912	0.916	0.942	<b>0.96</b>	0.924	0.935	0.928	0.848	0.83*	0.92	0.94
Bank	0.836	0.847	0.902	0.865	0.735*	<b>0.914</b>	0.86	0.895	0.84	0.87	0.85
Crime	0.979	0.933	0.979	<b>0.986</b>	0.906	0.953	0.875	0.89	0.88	0.84*	0.9
Education	0.916	0.921	0.973	0.92	0.821*	<b>0.993</b>	0.937	0.901	0.83	0.85	0.9
Employment	0.851	0.868	<b>0.903</b>	0.78	0.739	0.855	0.729*	0.814	0.81	0.85	0.84
Entertainment	0.83	0.809	0.826	<b>0.91</b>	0.732	0.894	0.716*	0.782	0.86	0.85	0.87
Health	0.87	0.766	0.852	<b>0.887</b>	0.794	0.853	0.739*	0.789	0.81	0.81	0.81
Lifestyle	0.82	0.831	<b>0.899</b>	0.848	0.887	0.894	0.759	0.756	0.72*	0.77	0.88
Literature	0.843	0.848	<b>0.922</b>	0.901	0.754	0.893	0.747*	0.835	0.81	0.88	0.88
Market	0.817	<b>0.887</b>	0.854	0.847	0.801	0.876	0.702*	0.827	0.81	0.85	0.87
Politics	0.931	0.892	<b>0.966</b>	0.944	0.884	0.922	0.864	0.887	0.88	0.85*	0.9
Sports	0.974	0.973	0.987	<b>1.0</b>	0.965	0.974	0.967	0.937	0.96	0.91*	0.97
Technology	0.885	0.881	<b>0.958</b>	0.886	0.807	0.908	0.854	0.899	0.81*	0.84	0.91
Tourism	<b>0.971</b>	0.913	0.959	0.956	0.847	0.932	0.814*	0.931	0.84	0.85	0.9
World	0.844	0.815	<b>0.902</b>	0.798	0.832	0.873	0.799	0.831	0.78*	0.82	0.86

*Note.* The bold-faced values indicate the highest weighted precision obtained for each class whereas lowest values are marked with an ‘\*’. XLM-R and Hindi-R stands for XLM-RoBERTa and HindiRoBERTa respectively.

them to predict the masked word from a Nepali text. While monolingual models such as DistilBERT and DeBERTa performed exceptionally well, others such as monolingual RoBERTa, mBERT, and different lingual HindiRoBERTa were unable to predict any meaningful words. However, when these models were fine-tuned for a downstream task, they were able to achieve some but not acceptable levels of accuracy. XLM-RoBERTa being a relatively larger model with bigger vocabulary size performed well and showed that for the low-resource languages where there is limited number of good pre-trained models and data, can benefit from multilingual models trained on significantly larger amounts of data. In multilingual models, however, numerous languages may share common characters and properties; for instance, in our research, both the Hindi and Nepali language models were pre-trained using the Devanagari script. On the contrary, subword tokenization may result in subwords that are similarly represented in the training of a multilingual model, establishing a subword learning bias towards certain languages over others (Wu & Dredze, 2020). Furthermore, as shown by DistilBERT and DeBERTa, a proper pretraining strategy facilitates a more accurate model representation of the language (Maskey et al., 2022) and they perform well in downstream tasks.

#### ***4.3.2 Comparison with past research***

Past research on text classification for high-resource language, English (Adoma et al., 2020; Cortiz, 2021; Gupta et al., 2021), out of various models compared, RoBERTa showed highest accuracy followed by BERT and DistilBERT. But, in the case of monolingual Nepali models, our comparative evaluation results are different than that of high-resource language, but are comparable to those of Maskey et al. (2022), where DeBERTa and DistilBERT demonstrated a significant performance improvement over BERT models. In addition, RoBERTa comparisons yielded similar results. In contrast to their results, the end-to-end performance of the multilingual model XLM-RoBERTa on our well-balanced dataset was quite noticeably comparable to that of the highest performing monolingual models. As previously discussed, the appropriate pre-training strategy may be the likely reason why Nepali models namely BERT and RoBERTa with improved architectures could not perform any better than DistilBERT and DeBERTa.

In the research conducted by Kumar and Albuquerque (2021); Ranasinghe and Zampieri (2020), the potential for zero-shot and cross-lingual language transfer is quite an interesting experiment for low-resource languages. They fine-tuned the XLM-RoBERTa model for the high-resource language English on a task-specific dataset, while using the learned weights as inputs for their different language datasets, such as Hindi, Bengali, Marathi, and still achieving the desired level of accuracy. They both concluded that XLM-RoBERTa, which was trained especially for cross-lingual language transfer in the multilingual domain, performs considerably better in a wide variety of scenarios. Similarly, our research indicates that the performance of the XLM-RoBERTa model on Nepali text classification tasks is comparable to that of Nepali monolingual transformer models. In addition, the different language model, HindiRoBERTa, that we used to fine-tune on Nepali language text shares comparable characteristics with the language-transfer domain and opens up endless possibilities for low-resource languages in general.

#### ***4.3.3 Useful experimental decisions***

A few experimental choices aided us in selecting the models for comparison. First, a straightforward method for measuring the MLM performance of the transformer model using masked word prediction, enabled a quick assessment of their pre-training effectiveness. In addition, when the models predicted sensible words according to the context, the outcomes of this form of evaluation appeared to be consistent with those of



the fine-tuning task. Furthermore, comparing monolingual models with mBERT and XLM-RoBERTa was a wise choice, as we were able to see some interesting results from a multilingual model’s ability to adapt cross-lingual language learning. Similarly, by using HindiRoBERTa, a model trained in a totally different language, we discovered the possibility of having a decent substitute model for low-resource languages for conditions where language-specific models are either unavailable or poorly trained. As for the baseline model, SVM proved to be an excellent model for Nepali text classification tasks due to its ability to generalize effectively in higher-dimensional domains, such as text. In addition, we showed that when the dataset is well-balanced across all classes, traditional ML methods are also able to perform significantly better. Also, Bi-LSTM model, when given sufficient data and with correct hyperparameter tunings was able to perform significantly well compared to transformer models.

#### ***4.3.4 Practical recommendations***

This study compared the effectiveness of Nepali monolingual transformer models in terms of their ability to perform significantly well for text categorization tasks. According to the results, DistilBERT and DeBERTa were the most effective Nepali pre-trained models among currently available models. In addition, it is crucial to continue research on multilingual models and large language models that share language-characteristics because they can be incredibly valuable for addressing NLP tasks, such as the XLM-RoBERTa base model explored in this study. Moreover, among the baseline models, Bi-LSTM and SVM were by far the most effective, however the question is whether this is due to the nature of the dataset we are using. We must investigate further whether the performance of the models is unaffected by the usage of an imbalanced dataset.

#### ***4.3.5 Limitations and future work***

One of the most major limitations of this study is that all experiments were conducted using a balanced data set. This could be a prime reason why ML algorithms were robust-enough for our Nepali text classification task. So, the performance of transformer models should also be evaluated for imbalanced datasets in order to determine models which are superior in handling imbalanced scenarios. Furthermore, new transformer models can be proposed with subtle changes in the underlying architecture in a way that delivers more improved outcomes for a wide range of tasks. Furthermore, transformer models must be evaluated on the basis of more complex sequence-to-sequence tasks such as

machine translation, question answering, text summarization, etc. in order to reveal their true capabilities.

## **CHAPTER 5**

### **CONCLUSION**

Rapid progress is being made in the domain NLP for the Nepali language, and the development of huge pre-trained models not only benefits the community but also paves the way for a wide range of future research opportunities for the entire country and language. This study emphasized the necessity for transformer models for low-resource languages, which can benefit from large language models with similar linguistic features or multilingual models. In contrast to typical ML and DL algorithms, which require learning both the language and the task in order to perform a specific task, pre-trained transformer models are capable of achieving even better performance metrics with just fine-tuning on task-specific data, without the need to learn language-specific features.

This study has highlighted that when monolingual languages are limited or poorly trained, low-resource languages can gain out-of-the-box performances even from large multilingual models without enormous investment. Moreover, these types of languages can also benefit from other monolingual models which share a similar linguistic pattern with each other, a transfer-learning approach. Unlike typical ML and DL algorithms where performing specific task requires learning the language and task at the same time, fine-tuning large transformer models are versatile enough to produce interesting results on a number of downstream tasks as they already possess deep understanding of the language through pre-training on a massive text data. With this advantage, low resource languages like Nepali can highly benefit from their state-of-the-art capabilities.

## REFERENCES

- Abadji, J., Suárez, P. J. O., Romary, L., & Sagot, B. (2021). Ungoliant: An optimized pipeline for the generation of a very large-scale multilingual web corpus. In H. Lungen, M. Kupietz, P. Bański, A. Barbaresi, S. Clematide, & I. Pisetta (Eds.), (pp. 1 – 9). Mannheim: Leibniz-Institut für Deutsche Sprache. Retrieved from <https://nbn-resolving.org/urn:nbn:de:bsz:mh39-104688> doi: 10.14618/ids-pub-10468
- Ács, J., Lévai, D., Nemeskey, D. M., & Kornai, A. (2021). Evaluating contextualized language models for hungarian. *arXiv preprint arXiv:2102.10848*.
- Adoma, A. F., Henry, N.-M., & Chen, W. (2020). Comparative analyses of bert, roberta, distilbert, and xlnet for text-based emotion recognition. In *2020 17th international computer conference on wavelet active media technology and information processing (iccwamtip)* (pp. 117–121).
- Aggarwal, A., Chauhan, A., Kumar, D., Mittal, M., & Verma, S. (2020). Classification of fake news by fine-tuning deep bidirectional transformers based language model. *EAI Endorsed Transactions on Scalable Information Systems*, 7(27).
- Al-Yahya, M., Al-Khalifa, H., Al-Baity, H., AlSaeed, D., & Essam, A. (2021). Arabic fake news detection: comparative study of neural networks and transformer-based approaches. *Complexity*, 2021.
- Antoun, W., Baly, F., & Hajj, H. (2020). Arabert: Transformer-based model for arabic language understanding. *arXiv preprint arXiv:2003.00104*.
- Basnet, A., & Timalisina, A. K. (2018). Improving nepali news recommendation using classification based on lstm recurrent neural networks. In *2018 ieee 3rd international conference on computing, communication and security (icccs)* (pp. 138–142).
- Bojanowski, P., Grave, E., Joulin, A., & Mikolov, T. (2017). Enriching word vectors with subword information. *Transactions of the association for computational linguistics*, 5, 135–146.
- Chirra, V. R., Maddiboyina, H. D., Dasari, Y., & Aluru, R. (2020). Performance evaluation of email spam text classification using deep neural networks. *Journal homepage: <http://iicta.org/journals/rcs>*, 7(4), 91–95.
- Conneau, A., Khandelwal, K., Goyal, N., Chaudhary, V., Wenzek, G., Guzmán, F., ... Stoyanov, V. (2019). Unsupervised cross-lingual representation learning at scale.

*arXiv preprint arXiv:1911.02116.*

- Cortiz, D. (2021). Exploring transformers in emotion recognition: a comparison of bert, distillbert, roberta, xlnet and electra. *arXiv preprint arXiv:2104.02041*.
- Demszky, D., Movshovitz-Attias, D., Ko, J., Cowen, A., Nemade, G., & Ravi, S. (2020). Goemotions: A dataset of fine-grained emotions. *arXiv preprint arXiv:2005.00547*.
- Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.
- de Vries, W., van Cranenburgh, A., Bisazza, A., Caselli, T., van Noord, G., & Nissim, M. (2019). Bertje: A dutch bert model. *arXiv preprint arXiv:1912.09582*.
- Dogra, & Varun. (2021). Banking news-events representation and classification with a novel hybrid model using distilbert and rule-based features. *Turkish Journal of Computer and Mathematics Education (TURCOMAT)*, 12(10), 3039–3054.
- Farahani, M., Gharachorloo, M., Farahani, M., & Manthouri, M. (2021). Parsbert: Transformer-based model for persian language understanding. *Neural Processing Letters*, 53(6), 3831–3847.
- Gage, P. (1994). A new algorithm for data compression. *C Users Journal*, 12(2), 23–38.
- Grave, E., Bojanowski, P., Gupta, P., Joulin, A., & Mikolov, T. (2018). Learning word vectors for 157 languages. In *Proceedings of the international conference on language resources and evaluation (lrec 2018)*.
- Gupta, P., Gandhi, S., & Chakravarthi, B. R. (2021). Leveraging transfer learning techniques- bert, roberta, albert and distilbert for fake review detection. In (p. 75–82). New York, NY, USA: Association for Computing Machinery. Retrieved from <https://doi.org/10.1145/3503162.3503169> doi: 10.1145/3503162.3503169
- He, P., Liu, X., Gao, J., & Chen, W. (2020). Deberta: Decoding-enhanced bert with disentangled attention. *arXiv preprint arXiv:2006.03654*.
- Kafle, K., Sharma, D., Subedi, A., & Timalisina, A. K. (2016). Improving nepali document classification by neural network. In *Proceedings ioe graduate conference* (pp. 317–322).
- Koirala, P., & Niraula, N. B. (2021, August). NPVec1: Word embeddings for Nepali - construction and evaluation. In *Proceedings of the 6th workshop on repre-*

- sentation learning for nlp (repl4nlp-2021)* (pp. 174–184). Online: Association for Computational Linguistics. Retrieved from <https://aclanthology.org/2021.repl4nlp-1.18> doi: 10.18653/v1/2021.repl4nlp-1.18
- Kumar, A., & Albuquerque, V. H. C. (2021, jun). Sentiment analysis using xlm-r transformer and zero-shot transfer learning on resource-poor indian language. *ACM Trans. Asian Low-Resour. Lang. Inf. Process.*, 20(5). Retrieved from <https://doi.org/10.1145/3461764> doi: 10.1145/3461764
- Lamsal, R. (2020). *A large scale nepali text corpus*. IEEE Dataport. Retrieved from <https://dx.doi.org/10.21227/jxrd-d245> doi: 10.21227/jxrd-d245
- Le, H., Vial, L., Frej, J., Segonne, V., Coavoux, M., Lecouteux, B., ... Schwab, D. (2019). Flaubert: Unsupervised language model pre-training for french. *arXiv preprint arXiv:1912.05372*.
- Lehečka, J., & Švec, J. (2021). Comparison of czech transformers on text classification tasks. In *International conference on statistical language and speech processing* (pp. 27–37).
- Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., ... Stoyanov, V. (2019). Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.
- Lowphansirikul, L., Polpanumas, C., Jantrakulchai, N., & Nutanong, S. (2021). Wangchanberta: Pretraining transformer-based thai language models. *arXiv preprint arXiv:2101.09635*.
- Maskey, U., Bhatta, M., Bhatta, S. R., Dhungel, S., & Bal, B. K. (2022). Nepali encoder transformers: An analysis of auto encoding transformer language models for nepali text classification. In *Lrec 2022 workshop language resources and evaluation conference 20-25 june 2022* (p. 106).
- Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., & Dean, J. (2013). Distributed representations of words and phrases and their compositionality. *Advances in neural information processing systems*, 26.
- Nguyen, D. Q., & Nguyen, A. T. (2020). Phobert: Pre-trained language models for vietnamese. *arXiv preprint arXiv:2003.00744*.
- Pennington, J., Socher, R., & Manning, C. D. (2014). Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (emnlp)* (pp. 1532–1543).

- Pirayani, R., Pirayani, B., Singh, V. K., & Pinto, D. (2020). Sentiment analysis in nepali: Exploring machine learning and lexicon-based approaches. *Journal of Intelligent & Fuzzy Systems*, 39(2), 2201–2212.
- Polignano, M., Basile, P., De Gemmis, M., Semeraro, G., & Basile, V. (2019). Alberto: Italian bert language understanding model for nlp challenging tasks based on tweets. In *6th italian conference on computational linguistics, clic-it 2019* (Vol. 2481, pp. 1–6).
- Ranasinghe, T., & Zampieri, M. (2020). Multilingual offensive language identification with cross-lingual embeddings. *arXiv preprint arXiv:2010.05324*.
- Ranjan, M. N. M., Ghorpade, Y., Kanthale, G., Ghorpade, A., & Dubey, A. (2017). Document classification using lstm neural network. *Journal of Data Mining and Management*, 2(2), 1–9.
- Regmi, S., Bal, B. K., & Kultsova, M. (2017). Analyzing facts and opinions in nepali subjective texts. In *2017 8th international conference on information, intelligence, systems & applications (iisa)* (pp. 1–4).
- Shahi, T., Sitaula, C., & Paudel, N. (2022). A hybrid feature extraction method for nepali covid-19-related tweets classification. *Computational Intelligence and Neuroscience*, 2022.
- Shahi, T. B., & Pant, A. K. (2018). Nepali news classification using naïve bayes, support vector machines and neural networks. In *2018 international conference on communication information and computing technology (iccict)* (pp. 1–5).
- Singh, O. M. (2018). *Nepali multi-class text classification*.
- Sitaula, C., Basnet, A., Mainali, A., & Shahi, T. B. (2021). Deep learning-based methods for sentiment analysis on nepali covid-19-related tweets. *Computational Intelligence and Neuroscience*, 2021.
- Souza, F., Nogueira, R., & Lotufo, R. (2020). Bertimbau: pretrained bert models for brazilian portuguese. In *Brazilian conference on intelligent systems* (pp. 403–417).
- Subba, S., Paudel, N., & Shahi, T. B. (2019). Nepali text document classification using deep neural network. *Tribhuvan University Journal*, 33(1), 11–22.
- Tamrakar, S., Bal, B. K., & Thapa, R. B. (2020). Aspect based sentiment analysis of nepali text using support vector machine and naive bayes. *Technical Journal*, 2(1), 22–29.

- Terechshenko, Z., Linder, F., Padmakumar, V., Liu, M., Nagler, J., Tucker, J. A., & Bonneau, R. (2020). A comparison of methods in political science text classification: Transfer learning language models for politics. *Available at SSRN 3724644*.
- Thakur, S. K., & Singh, V. K. (2014). A lexicon pool augmented naive bayes classifier for nepali text. In *2014 seventh international conference on contemporary computing (ic3)* (pp. 542–546).
- Thapa, L. B. R., & Bal, B. K. (2016). Classifying sentiments in nepali subjective texts. In *2016 7th international conference on information, intelligence, systems & applications (iisa)* (pp. 1–6).
- Wagle, S. S., & Thapa, S. (2021). Comparative analysis of nepali news classification using lstm, bi-lstm and transformer model.
- Wenzek, G., Lachaux, M.-A., Conneau, A., Chaudhary, V., Guzmán, F., Joulin, A., & Grave, É. (2020). Ccnet: Extracting high quality monolingual datasets from web crawl data. In *Proceedings of the 12th language resources and evaluation conference* (pp. 4003–4012).
- Wilie, B., Vincentio, K., Winata, G. I., Cahyawijaya, S., Li, X., Lim, Z. Y., ... Syafri (2020). Indonlu: Benchmark and resources for evaluating indonesian natural language understanding. *arXiv preprint arXiv:2009.05387*.
- Wongso, W., Lucky, H., & Suhartono, D. (2022). Pre-trained transformer-based language models for sundanese. *Journal of Big Data*, 9(1), 1–17.
- Wu, S., & Dredze, M. (2020). Are all languages created equal in multilingual bert? *arXiv preprint arXiv:2005.09093*.