Received Date:21st December, 2024
Revision Date: 5th January, 2025
Accepted Date:21st January, 2025

Comparison Analysis of Nepali News Classifier

Koshish Shrestha[1]*, Kabita Khanal[2], Isha Shrestha[3], Neerav Shrestha[4], Kunjan Amatya[5]

[1]Dept. of Computer Engineering, Kathmandu Engineering College. Email: koshish62@gmail.com
[2] Dept. of Computer Engineering, Kathmandu Engineering College. Email: kbtkhnl@gmail.com
[3] Dept. of Computer Engineering, Kathmandu Engineering College. Email: ishashrestha246@gmail.com
[4] Dept. of Computer Engineering, Kathmandu Engineering College. Email: neeravsth11@gmail.com
[5]Assoc. Professor, Dept. of Computer Engineering, Kathmandu Engineering College. Email: kunjan.amatya@kecktm.edu.np

Abstract - With the growing volume of daily generated Nepali news content, which predominantly exists in unstructured formats, the need arises to effectively categorize and label this information. Considering this challenge, our system employs various Machine Learning algorithms such as Logistic Regression, Random Forest Algorithm, Neural Networks, etc., to automatically classify Nepali news into predefined categories and evaluate the efficiency of the model, which offers us a comparative analysis of these various algorithms.

Keywords - Nepali news, News Classification, Machine Learning, Logistic Regression, Random Forest Algorithm, Neural Network, Categories, Training, Testing, Efficiency

Introduction

News classification is the process of grouping news documents into some predefined categories. Due to the increasing volume of the Nepali news content being generated every day by thousands of online news portals, appropriate classification of these news items has become a necessity for the news readers.

The manual categorization of news articles proves to be time consuming and demanding, thus highlighting the necessity for an automated system capable of accurately classifying articles into predetermined categories. Leveraging the capabilities of natural language processing (NLP) and machine learning algorithms offers robust solutions for text classification tasks, ultimately enhancing the efficiency of news categorization systems. These news articles can be classified into various categories such as 'Business,' 'Politics,' 'Education,' 'Economy,' 'Sports,' 'Environment,' and more.

Objective

The objective of our project is to develop an accurate Nepali news classification system for efficiently categorizing online news, addressing the current challenges of manual classification by comparative analysis of multiple algorithms.

Related Works

A study tackled the task of Nepali News Classification by using different deep learning algorithms namely, Long Short-Term Memory (LSTM), Bidirectional Long Short-Term Memory (Bi-LSTM) and Transformer. News classification was done into 17 categories using around 200k news articles as a dataset. The LSTM model obtained a training score of 93.55% and a testing score of 92.94%. The Bi-LSTM model obtained a training score of 94.58% and a testing score of 93.65%.

Finally, the Transformer model has a training score of 96.54% and a testing score of 95.45%. The transformer was found to perform the best on the data however the transformer model required a lot of training data compared to the other models [1]. An analysis of Auto Encoding Transformer Language Models for Nepali Text Classification analyzed two auto-encoding transformer model, DeBERTa model focused on performance while DistilBERT model focused on being lightweight. During the classification task, DeBERTa model produced a highest accuracy of 88.93% while the DistilBERT model produced 88.31% accuracy [2].

A research paper was targeted to improve the Nepali news classification based on Recurrent Neural Networks. The model was compared with the Support Vector Machine based on the parameters Accuracy, Precision, Recall and F1 Score. The use of Long Short Term Memory Recurrent Neural Network has improved the precision with the use of

* *Corresponding Author*

word2vec model. The presented model in the research paper has achieved an accuracy of 84.63% and precision of 89% compared to the SVM where the accuracy was 81.41% and precision 85% [3].

A research evaluates various machine learning techniques, mainly Naive Bayes, SVM and Neural Networks, for automatic Nepali news classification problems. The average empirical results showed that the SVM with RBF kernel outperformed the other algorithms with the classification accuracy of 74.65% followed by linear SVM with accuracy 74.62%, Multilayer Perceptron Neural Networks with accuracy 72.99% and the Naive Bayes with accuracy 68.31% [4]

A study aimed to upgrade Nepali News document classification based on Long Short-Term Memory, Recurrent Neural Network and Global Vectors for Word Representation. The LSTM model produced the highest accuracy of 95.36%, followed by CNN with 93.97% accuracy and finally DNN with 90.75% [5].

In a paper, data from five distinct news portals were collected and analyzed using five distinct models: LSTM, BiLSTM, GRU, BiGRU using neural networks, and BERT. These 5 models are compared based on parameters like accuracy, precision, recall and F1-score where BERT emerged as the most effective model, boasting an accuracy of 95% [6].

Methodology

A. System Block Diagram

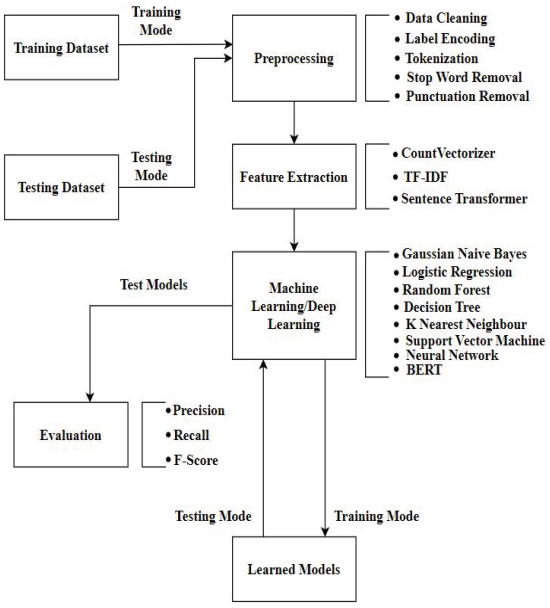

Fig. 1: System Block Diagram

B. Data Collection

In the data collection step, we gathered a diverse and representative dataset of Nepali news articles from various sources. We collected news articles from online news portals, blogs, social media platforms, and other relevant sources that publish news articles in Nepali. To ensure comprehensive classification, we made sure that the dataset covers a wide range of topics such as politics, business, health, sports, entertainment, accidents, education, environment, finance, and more.

Dataset we obtained had a total of 211191 news articles. There was data of 20 categories each containing a large number of news articles.

Table i

Dataset count before compilation

S.N.	Categories	Content
1	Arts	7712
2	Automobile	5152
3	Bank	10526
4	Business	11645
5	Crime	8173
6	Economy	23653
7	Education	3709
8	Employment	1931
9	Entertainment	22664
10	Health	9469
11	Lifestyle	3856
12	Literature	3856
13	Market	4373
14	Opinion	3945
15	Politics	31021
16	Society	27802
17	Sports	24392
18	Technology	6427
19	Tourism	3161
20	World	17431

C. Data Sampling

To enhance the accuracy of our model, we addressed two key issues within the dataset. First, we encountered significant class imbalance, where data within certain categories were disproportionately distributed. Second, we identified repetitive categories that could potentially skew the model's performance.

To rectify these issues, we implemented a data pruning strategy. Specifically, we decided to limit the number of instances per category to a maximum of 3000 data points. This approach aimed to achieve a more balanced distribution across all categories, thereby preventing the model from being biased towards over represented classes. By capping the data for each category, we aimed to create a more equitable training environment where all classes had a comparable influence on the model's learning process.

Table ii
Dataset count after sampling

S.N.	Categories	Content
1	Automobile	2999
2	Business	3000
3	Crime	2999
4	Education	3000
5	Entertainment	3000
6	Health	3000
7	Literature	3000
8	Market	3000
9	Politics	3000
10	Sports	3000
11	Technology	2998
12	Tourism	2984
13	World	3000

D. Data Preprocessing

Data preprocessing is the process of cleaning and preparing the text for feature extraction and classification. This process for our project involved the following steps:

1) *Data Cleaning:* The data extracted from the news sites contained a mixture of Nepali script, English, and special characters. There were also occurrences of other Unicode characters within the text. To ensure that we work with Nepali language exclusively, we needed to perform data cleaning. This cleaning process involved removing English characters, special symbols, and unnecessary white spaces from the data.

2) *Label Encoding:* In this process, each unique category or label is assigned an integer value, which is useful when working with algorithms and models that require numerical input, as most machine learning algorithms operate on numerical data.

3) *Tokenization:* Tokenization is a process of separating a piece of text into smaller units called tokens. We explored techniques such as dictionary-based tokenization or statistical models. Vertical bar, question mark, and full stop can be used to break down the sentences while space and comma can be used to break down the words.

4) *Stop Words and Punctuation Removal:* We eliminated common and insignificant words (e.g., articles, prepositions, conjunctions) and punctuation that do not carry much semantic meaning, in order to improve the performance of the classification.

E. Feature Extraction

Feature Extraction process involves transforming the pre-processed text into numerical feature vectors that a machine learning algorithm can understand. We employed techniques such as:

1) *CountVectorizer:* CountVectorizer is a tool provided by the scikit-learn library in Python. It is used to transform a given text into a vector on the basis of the frequency (count) of each word that occurs in the entire text.

   CountVectorizer creates a matrix in which each unique word is represented by a column of the matrix, and each text sample from the document is a row in the matrix. The value of each cell is nothing but the count of the word in that particular text sample.

2) *TFIDF Vectorizer:* We transformed the preprocessed text into numerical feature vectors that machine learning algorithms can understand. The idea of TF-IDF is to reflect the importance of a word to its document or sentence by normalizing the words which occur frequently in the collection of documents.

   Term frequency (TF): It is the number of times a term has appeared in a document. The term frequency is a measure of how frequently or how common a word is for a given sentence.

   TF of a word is = N/n.....(i)

   N: total number of times the word appears in the document n: Total number of terms in a document

   Inverse Document Frequency (IDF): The inverse document frequency (IDF) is a measure of how rare a word is in a document. Words like "a", "an", "the" usually appears in almost all the documents but some of the rare words will not appear in all the documents of

the corpus. If a word appears in almost every document means the word is not significant for the classification.

IDF of a word is = $log(N/n)$.....(*ii*)

N: total number of documents

n: number of documents containing a term (word)

3) *Sentence Transformer:* Sentence Transformers is a Python framework for state-of-the-art sentence, texts as well as image embeddings. It is a type of model which transforms sentences or texts into fixed-size vectors. It offers pre-trained models based on transformer architecture for generating embeddings. These embeddings can be useful for various natural language processing (NLP) tasks.

4) *Bert Tokenizer:* The BERT tokenizer is a crucial component of the BERT (Bidirectional Encoder Representations from Transformers) model, which is a powerful pre-trained language representation model developed by Google. The tokenizer's primary function is to break down input text into individual tokens that can be processed by the BERT model. Overall, the BERT tokenizer plays a vital role in preparing text data for input into the BERT model, enabling it to generate high quality contextual embeddings that capture the semantics and nuances of the input text effectively.

Once the data undergoes the vectorization process, wherein the raw text was transformed into a numerical representation based on the frequency of individual words, the resulting feature vectors were then sent into a machine learning pipeline, initiating the training phase for the model. During this training process, the machine learning algorithm refined its internal parameters through iterative optimization, learning and adapting to the patterns and relationships encoded in the vectorized data, ultimately enhancing its ability to classify and make predictions based on the features derived from the original textual.

F. Training the model

After all the previous processes, the data was ready for the model to be trained. We split the dataset so obtained in 80:20, where 80% of the dataset was used to train the model whereas the leftover 20 % of the data was used to test the model. On training the models, we used a variety of algorithms from Gaussian Naive Bayes, Random Forest, Logistic Regression, Neural Network, and so on. Each algorithm fed the obtained data from pre-processing and after the model was trained the model was tested using the test data. This process was run over several iterations and stopped if the desired accuracy was obtained or if the accuracy remained stagnant over multiple attempts. In order to increase the accuracy, the model was hyper-parameterized or tuned to find the optimal value of the hyper parameters which increased the overall performance of the model.

G. Algorithms

1) *Gaussian Naive Bayes Algorithm:* Gaussian Naive Bayes is a probabilistic classification algorithm based on Bayes' theorem, particularly suited for tasks where the features are continuous and follow a Gaussian (normal) distribution.

2) *Logistic Regression:* Logistic regression is one of the most popular Machine Learning algorithms that comes under the Supervised Learning technique. It is used for predicting the categorical dependent variable based on a given set of independent variables. The algorithm predicts the output of a categorical dependent variable. Therefore, the outcome must be a categorical or discrete value. Instead of providing the exact values, which can be either Yes or No, 0 or 1, True or False, it provides the probabilistic values, which fall between 0 and 1.

3) *Random Forest:* Random Forest is an ensemble learning method commonly used for both classification and regression tasks. It operates by constructing a multitude of decision trees during training and outputs the mode (for classification) or the average prediction (for regression) of the individual trees. The "random" in Random Forest comes from the fact that it introduces randomness in the tree-building process.

4) *K Nearest Neighbor:* K Nearest Neighbors (KNN) is a simple, yet effective, supervised machine learning algorithm used for both classification and regression tasks. It makes predictions based on the majority class (for classification) or the average value (for regression) of the k- nearest data points in the feature space.

5) *Support Vector Machine:* A Support Vector Machine (SVM) is a powerful and versatile supervised machine learning algorithm used for both classification and regression tasks. It works by finding the optimal hyperplane that separates data points of different classes in a high- dimensional space. SVM is particularly effective in cases where the decision boundary is nonlinear or complex.

6) *Neural Network:* Computing systems with interconnected nodes that function similar to neurons in the human brain are called neural networks. They can recognize hidden patterns and correlations in raw data, cluster and classify it, and continuously learn and improve over time. They interpret sensory data with a kind of machine perception, labeling or clustering raw unprocessed input. The patterns they recognize are numerical, that are contained in vectors, into which all real-world data, which may be images, sound, text, or time series, must be translated. Artificial neural networks are used for solving artificial intelligence (AI) problems and model connections of biological neurons as weights between nodes.

7) *BERT:* BERT stands for Bidirectional Encoder Representations from Transformers. BERT, which was introduced in October 2018 by researchers at Google. Notable for its noticeable improvement over previous state of the art models, it makes use of Transformer, an attention mechanism that learns contextual relations between words (or sub-words) in a text. In its vanilla form, Transformer includes two separate mechanisms — an encoder that reads the text input and a decoder that produces a prediction for the task. As compared to directional models that read the text input sequentially (left-to-right or right-to-left), the Transformer encoder however reads the entire sequence of words at once. Therefore, it is considered bidirectional. This particular characteristic allows the model to learn the context of a word based on all of its surroundings.

MLM, or Masked LM, is a technique used in BERT where 15% of the words in a word sequence are replaced with a [MASK] token. The model then tries to predict the original values of these masked words based on the context provided by the other words in the sequence. To make these predictions, a classification layer is added on top of the encoder output. The output vectors are multiplied by an embedding matrix to transform them into the vocabulary dimension. The probability of each word in the vocabulary is calculated using softmax. The BERT loss function focuses only on the prediction of the masked values and disregards the prediction of the non masked words. This causes the model to converge slower than directional models, but it gains increased context.

In BERT training, a technique called Next Sentence Prediction (NSP) is also used. The model receives pairs of sentences and learns to predict if the second sentence in the pair follows the first sentence in the original document. During training, 50% of the inputs consist of pairs where the second sentence is indeed the subsequent sentence in the original document, while the other 50% have a randomly chosen sentence as the second one. The idea is that the random sentence is unrelated to the first sentence.

Verification and Validation

A. Comparison of test accuracy of different models

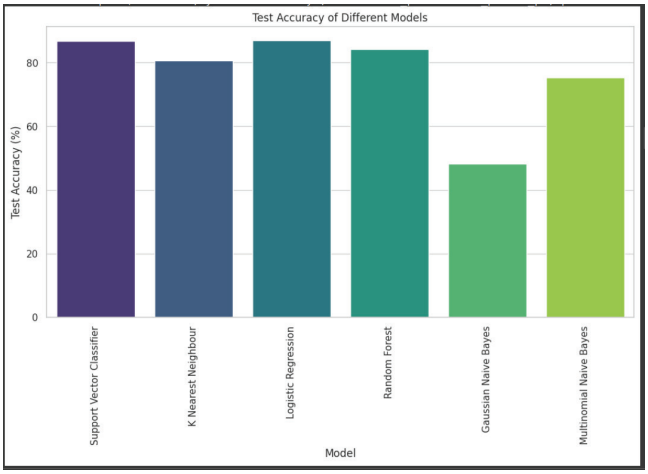

Fig. 2 Bar graph of accuracy of different models

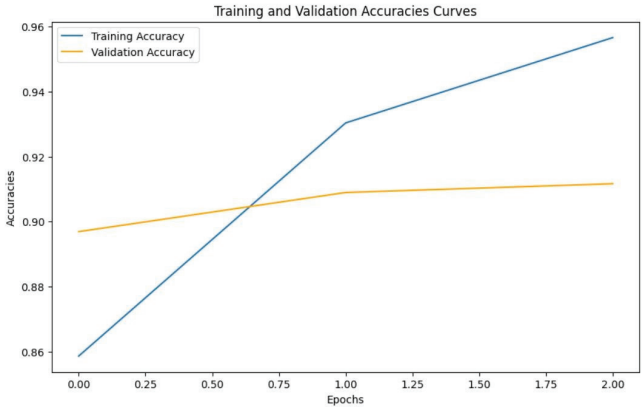

Fig. 3 Training and Validation Accuracy Curves of BERT Model

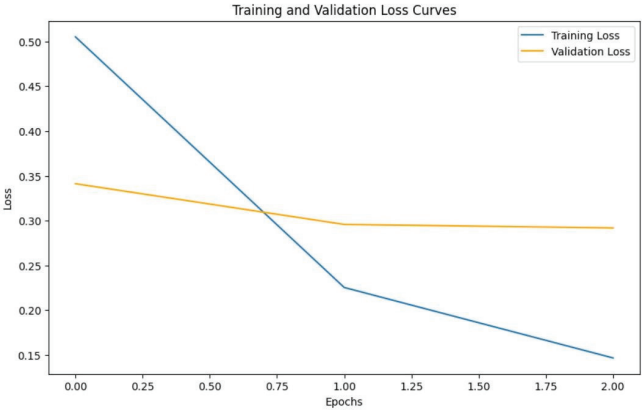

Fig. 4 Training and Validation Loss Curves of BERT Model

After conducting training and validation of the model, the process was limited to 2 epochs due to observed changes in performance metrics. Following the second epoch, there was a noticeable increase in loss and decrease in accuracy on the validation set. Recognizing that further epochs were unlikely to improve results and could lead to overfitting, the decision was made to halt training to preserve the model's generalization capability.

This choice was guided by careful monitoring of the training process, aiming to strike a balance between performance and the risk of overfitting. By limiting training to 2 epochs, the model achieved optimal performance within a practical timeframe and remained robust for real-world deployment.

Table iii

Comparison of performance metrics for various models

Model	Test Accuracy	Precision	Recall	F1
Support Vector Classifier	88.10	0.88	0.88	0.88
K Nearest Neighbor	80.81	0.81	0.81	0.81
Gaussian Naive Bayes	48.31	0.48	0.48	0.48
Logistic Regression	87.01	0.87	0.87	0.87
Random Forest	84.29	0.84	0.84	0.84
Neural Network	81.09	0.81	0.81	0.81
BERT	91.16	0.91	0.91	0.91

Result and Conclusion

After extensive development and testing using various models, we concluded that BERT performed the best among all the models with accuracy of 91.16. Thus, we have successfully created a robust and accurate Nepali news classification system using BERT. This system efficiently categorizes online news articles, eliminating the need for manual classification. Our system allows users to input Nepali news articles, which are then classified by our trained model. The classified articles are then displayed on the portal's category-wise pages based on the categories provided by the classifier. Through rigorous testing and validation, we have achieved high accuracy in classifying Nepali news articles into relevant categories.

Table iv

Performance Metrics of BERT Model

Accuracy	91.1621344279
Precision	0.9118171493
Recall	0.91162134427
F1 Score	0.9113949489

TABLE V

Category-wise Accuracy

Category	Accuracy
Education	95.847176
World	92.786885
Sports	98.564593
Technology	85.908331
Crime	96.375617
Health	92.931034
Market	90.677966
Politics	93.019480
Automobile	92.720970
Tourism	92.991452
Literature	82.782608
Entertainment	88.244766
Business	81.847649

Acknowledgment

First of all, we wish to express our sincere gratitude towards the entire Department of Computer Engineering, Kathmandu Engineering College, Er. Sudeep Shakya, Head of Department, for providing us the opportunity of undertaking this project which has helped us implement the knowledge gained over these years. Without their valuable supervision and suggestions, it would have been a difficult journey for us. We are extremely grateful to our project supervisor Er. Kunjan Amatya, whose constant guidance throughout the duration of the project has been immensely helpful and motivating.

We would like to express our heartfelt gratitude to Lokantra Post for providing us with valuable information about the workings of the news industry on the online news platform. We are also thankful to Kantipur for providing us with the opportunity to visit their organization and learn more about the industry.

We would especially like to thank our year coordinator and our project coordinator Er. Sharad Chandra Joshi for his wholehearted and continuous support. We are also grateful to our teachers for their continuous support. Lastly, we would also like to thank all of our friends and every other character who have contributed directly or indirectly in this project.

References

[1] Wagle, S. S., & Thapa, S. (2021). Comparative analysis of nepali news classification using lstm, bi-lstm and transformer model.

[2] Maskey, U., Bhatta, M., Bhatt, S., Dhungel, S., & Bal, B. K. (2022, June). Nepali encoder transformers: An analysis of auto encoding trans former language models for nepali text classification. In Proceedings of the 1st Annual Meeting of the ELRA/ISCA Special Interest Group on Under-Resourced Languages (pp. 106-111).

[3] Basnet, A., & Timalsina, A. K. (2018, October). 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). IEEE.

[4] Shahi, T. B., & Pant, A. K. (2018). Nepali news classification using Naive Bayes, Support Vector Machines and Neural Networks. 2018 International Conference on Communication Information and Computing Technology (ICCICT). https://doi.org/10.1109/ICCICT.2018.8325883

[5] Thapa, S. K., & Pokhrel, S. (2021). Nepali News Document Classification using Global Vectors and Long Short Term Memory.

[6] Kafle, P., Chaitrakar, R., & Nemkul, K. (2022). Improving Nepali News Classification Using Bidirectional Encoder Representation from Transformers. In Artificial Intelligence and Sustainable Computing: Proceedings of ICSISCET 2021 (pp. 485-494). Singapore: Springer Nature Singapore.