MSc in Business Administration and Data Science
Final Project Natural Language Processing and Text Analytics

News Article Classification: A Comparative Study of Different Models

Alexandros Kyriakopoulos; 167290
Arthur Harry Fenton Söhler; 167336
Julian Irigoyen Safar; 167279

Course Coordinator: Rajani Singh
Date: 30.05.2024
Pages: 10
Characters: 26962
Link to Dataset:
https://www.kaggle.com/datasets/rmisra/news-category-dataset
Link to GitHub Repository:
https://github.com/juligoat/NLP_Final_Assignment

Abstract

As natural language processing (NLP) techniques advance, their application in text classification has become increasingly popular due to their applicability in areas such as news recommendation. This study focuses on a HuffPost dataset containing over 200,000 news articles from 2012 to 2022, labeled across various categories. It is aimed at developing different machine-learning models to classify these articles. The research question addresses which models have the best performance and how they compare to one another when assigning a category to a specific news article. Our approach includes EDA and data preprocessing on the raw data, followed by various tokenization and vectorization techniques and hyperparameter tuning for each model. We use and evaluate six models: Multinomial Naïve Bayes, Multinomial Logistic Regression, Multi-Layer Perceptron (MLP), MLP with Word2Vec Embeddings, Bidirectional Long Short-Term Memory (Bi-LSTM), and Bidirectional Encoder Representations from Transformers (BERT). While most algorithms achieved similar accuracies (from 77% to 80%), the BERT model obtained the highest accuracy (84%) but at a significant computational cost. This study discusses the trade-offs between model performance and computational efficiency.

Keywords: Natural Language Processing (NLP), News Article Classification Huffington Post, Machine Learning, Transformers, Bidirectional Encoder Representations from Transformers (BERT), Word Embeddings, Word2Vec, Bidirectional Long Short-Term Memory (Bi-LSTM), Multi-Layer Perceptron (MLP), Multinomial Logistic Regression, Multinomial Naïve Bayes.

1 Introduction

This study focuses on classifying news articles based on a HuffPost dataset. This dataset contains articles humanly labeled with their respective category. This paper aims to develop progressively complex machine-learning models that can effectively utilize diverse natural language processing techniques to classify news articles into their respective categories. We aim to identify the most successful techniques for this task by exploring different NLP techniques for preprocessing and model development. The study will also explore how newer and more advanced NLP techniques, such as word embeddings and transformers, affect the classification task's accuracy and other performance metrics. Finally, our goal is to understand better how real-world data can behave when state-of-the-art natural language processing techniques are applied and gain insight into how further work on this topic could look like

The business value of this project originates from the fact that media companies like the HuffPost generate vast amounts of content daily, and categorizing this content manually would be both costly and time-consuming. Therefore, a need exists for automated, efficient, and accurate classification of news into categories. Moreover, correctly and well-classified articles could assist in having better-personalized news feeds, which can be used to make readers better informed and engaged with the news articles. From an academic and research perspective, the interest in this project results from leveraging different state-of-the-art

NLP techniques to solve a real-world problem. The results provide insights into different models' performance, the limitations of this project, and the models for this task.

The remainder of this paper is organized as follows: Section 2 reviews relevant related literature. Section 3 describes the methodology, including the data analysis workflow, dataset description, data analysis workflow process, and modeling selection. Section 4 discusses the hyperparameter tuning of each model, and Section 5 presents the results. These results are discussed in Section 6, including the limitations of the models and the project. Finally, Section 7 concludes the paper and suggests directions for future work.

2 Related Work

In a similar approach to ours, Hussain et al. (2020) compare the performance of four supervised machine learning methods to predict five different news categories (Sport, Tech, Entertainment, Politics, Business) of a dataset of the British Broadcasting Corporation (BBC). In their work, they use the Multinomial Naïve Bayes classifier, Random Forest, K-Nearest Neighbors (KNN), and Support Vector Machine (SVM), achieving up to 97.3% accuracy. In contrast, our research uses more advanced models and more news categories, increasing the complexity of the project.

Nugroho et al. (2021) do not use simpler models but instead choose to use different implementations of BERT to classify four news categories from the AG data set containing news articles from different sources. Their best model, BERT-Base, achieved 92.53% accuracy.

3 Methodology

3.1 Data Analysis Workflow

Figure 1 provides an overview of the data analysis pipeline followed throughout the analysis:

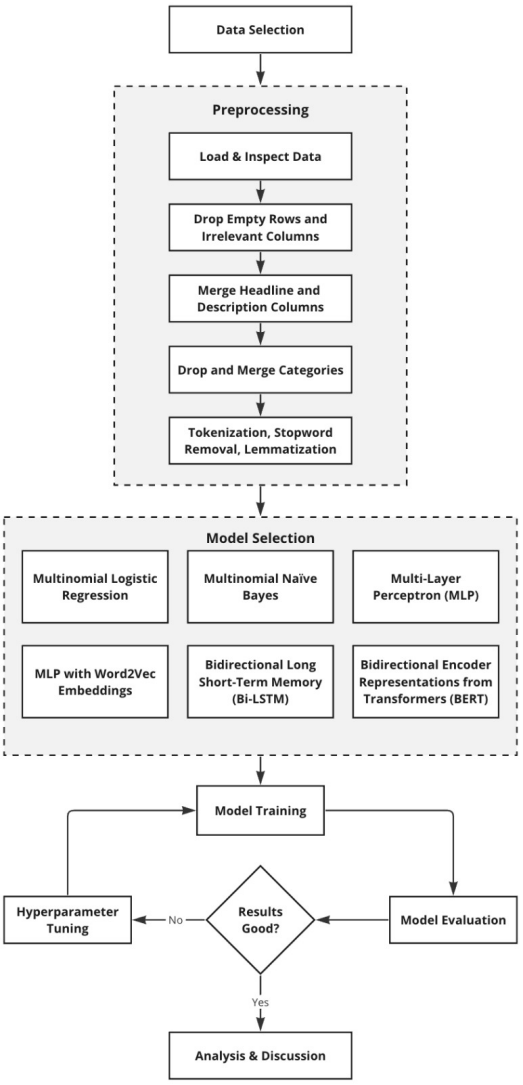

Figure 1: Overview of the data analysis pipeline.

3.2 Data Description

The dataset used in this project is the News Category Dataset which contains 210,000 news headlines, descriptions, and categories as well as additional metadata (author, date, URL) for each article from the HuffPost news website. The data was collected in 2022 by Rishabh Misra to be used for NLP tasks (Misra, 2022). The news articles were published from the years 2012 to 2022. In total, there are 42 different news categories in the dataset, an overview of which can be found in **Appendix A**. Below are a few examples of news headlines and descriptions along with their respective category:

Headline	Short Description	Category
Biden Says U.S. Forces Would Defend Taiwan If China Invaded	President issues vow as tensions with China rise	Politics
Las Vegas Aces Win First WNBA Title, Chelsea Gray Named MVP	Las Vegas never had a professional sports champion, until Sunday	Sports
'Our Hearts Are Broken'	Historic Front Pages Mark The Queen's Death: Both British and international newspapers honor the passing of the U.K.'s longest-reigning monarch	World News

3.3 EDA and Data Preprocessing

The preprocessing of the data mainly consisted of four parts:

1. Loading & Inspection of the data
2. Dropping empty rows and irrelevant columns for the analysis
3. Merging of Headline and Description columns
4. Remove categories and merge similar categories

The dataset is in JSON format, with each line representing one separate news article. We loaded this file to create a data frame. Then, we analyzed the data frame to see the data's characteristics. We dropped missing values in the 'headline' and 'short description' columns since we are using these as our model input. The dropped rows accounted for 9.59% of the dataset. After the NAs were removed, the columns were merged into one.

Another important step in our preprocessing that provided an edge in our metrics compared to similar efforts on the HuffPost dataset is the grouping of similar categories. The reasoning behind this grouping was that 42 categories are too many, as most news sites only have a handful. To reduce the number of categories, we merged them when they had similar articles and removed them when they did not represent real news categories and would therefore be difficult to categorize for the model, e.g., "Good News" or "Funny News". For categories such as Religion, Crime, and Queer Voices, the news articles within them were a mix of other categories such as World News, Politics, or Business Tech, making it difficult for any model to

classify. This is why we also decided to drop these kinds of categories. Finally, we removed some categories that had too few rows to be able to keep, such as Education or College. Once this process was concluded, eleven categories remained from the initial forty-two, as depicted in **Figure 2**.

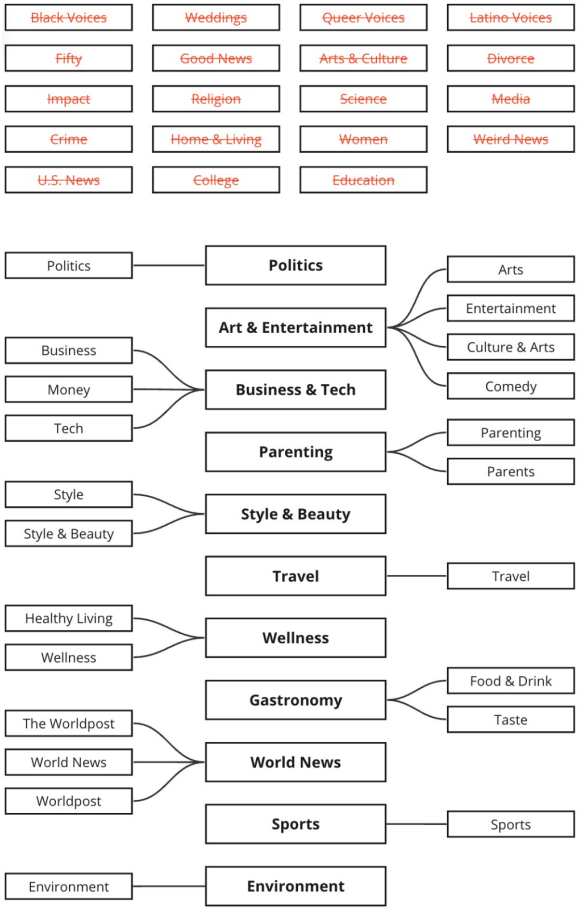

Figure 2: Transformation of news categories.

After all of our preprocessing, we dropped 31.66% of the rows in total. This includes the initial dropping (9.59%) and the dropping due to merging and eliminating categories (31.07%). Therefore, 68.34% of the original dataset was available for model training. Following this, we used NLTK and spaCy for tokenization, stopword removal, and lemmatization. We then compared how each performs and found that spaCy performs better. For this reason, this library was employed for data preparation in our models.

3.4 Model Selection

The following six models are used to classify news into eleven categories:

Multinomial Naïve Bayes: a linear model well suited for large-scale classification using L1 and L2 regularization (Yuan et al., 2012). It is also called Softmax Regression, as it utilizes the Softmax activation function to model the probabilities of each possible outcome category.

Multinomial Logistic Regression: a simple probabilistic classifier based on Bayes' theorem, which predicts the class of given inputs based on feature (words) frequency. It assumes independence between the features. Hyperparameter tuning is conducted for the smoothing parameter used in the algorithm.

Multi-Layer Perceptron (MLP): an extension of the single perceptron, which solves the XOR classification problem by introducing hidden layers. It consists of a fully connected, feed-forward network that allows modeling complex decision boundaries. A tuner based on validation accuracy is built to optimize the model's performance. The best combination is found among the number of layers, neurons (TLUs) in each layer, activation functions, learning rate and dropout rate, and L2 to introduce regularization. Furthermore, early stopping is introduced to ensure better convergence.

MLP with Word2Vec Embeddings: following the previous model, an enhanced MLP model was

created using word2vec pre-trained embeddings as inputs. Word2Vec summarizes word contexts as dense vectors, which work better in most NLP tasks than sparse vectors. These are represented in a continuous vector space, where semantically similar words are mapped to nearby points.

Bidirectional Long Short-Term Memory (Bi-LSTM): BiLSTMs are an advanced form of Recurrent Neural Networks (RNNs). They consist of two LSTMs, one that processes the input in a forward direction and another one that processes the input in a backward direction. They are then combined using concatenation. This way, the RNN can also use information from future states (Schuster Paliwal, 1997), useful for tasks where context is important, such as text classification.

Bidirectional Encoder Representations from Transformers (BERT): a model with excellent capabilities for natural language understanding. It has been pre-trained on a corpus, including Wikipedia and BookCorpus, and has showcased state-of-the-art performance on many NLP benchmarks (Devlin et al., 2019). Although the model is pre-trained, it provides many hyperparameters to tune. BERT is effective for various tasks, including multiclass classification of the HuffPost articles. For this task, we used the TFBertForSequenceClassification library. BERT has a variety of libraries to choose from depending on the task it is built for; for the multiclass classification task, the aforementioned library is the appropriate one. Additionally, BERT includes a tokenizer; it first applies basic tokenization and then wordpiece tokenization (Text.BertTokenizer | Text, n.d.).

4 Model Tuning

This section goes through the hyperparameter tuning performed in each model:

Multinomial Naïve Bayes: we experimented with CountVectorizer and TfidfVectorizer for input representation. The TF-IDF approach, which considers the importance of words in the context of the entire document corpus, achieved better accuracy than its counterpart. Moreover, Grid Search was conducted only to find the best smoothing parameter (alpha)(MultinomialNB, n.d.). The best-performing model was found to have an alpha value of 0.1, indicating that Laplace smoothing in our model only adds a small increment of 0.1 to feature counts. This implies that the model leans more towards relying on raw data while achieving good generalization without the necessity of aggressive smoothing.

Multinomial Logistic Regression: similar to our previous model, better performance was achieved by employing TF-IDF and spaCy to remove stopwords, tokenize, and lemmatize the news texts. The hyperparameter tuning, also based on Grid Search, indicated that the best-performing model utilizes the L-BFGS solver and Ridge Regularization (L2) set to a strength of C=1. This is expected as news articles vary widely in length, tone, and topic, which may introduce outliers in the feature space. A moderate L2 regularization increases the algorithm's robustness to outliers, while the efficient optimization provided by the L-BFGS solver is advantageous in large classification tasks with noisy data (Géron, 2019).

Multi-Layer Perceptron (MLP): up to this point, we have utilized sparse matrices generated

by TfidfVectorizer as inputs for our models. However, the BatchNormalization layer from our MLP models expects dense input tensors. To address this issue, we converted the sparse matrices into dense arrays using numpy. Additionally, we considered applying feature scaling to standardize the input features before passing them to the BatchNormalization layer, but we discarded this approach as it resulted in poorer performance.

During the hyperparameter tuning process, the performance of both label encoding and one-hot encoding was explored for the outcome variable. Even though the results were similar, the models trained with label encoding performed slightly better. As a result, label encoding was selected for the final model as well as for all further models. The best hyperparameters for the MLP reveal an architecture of an input layer with 63,277 neurons, representing the total amount of unique words in the corpus, followed by a Batch Normalization layer to scale the inputs and ensure easier convergence. Following this, two hidden layers of 96 and 480 neurons, respectively employed the tanh activation function. The output layer comprised 11 neurons with a softmax activation function, adequate for multiclass classification. This architecture can be seen in **Figure 3**. These specifications, alongside the feature dropout rate of dropout layers and the optimal L2 regularization values, were tuned through a RandomSearch of 30 trials of 10 epochs. The Adam optimizer was employed with varying learning rates throughout the entire neural network, and sparse categorical cross entropy was used as the loss function. To optimize training time, an Early Stopping callback was introduced.

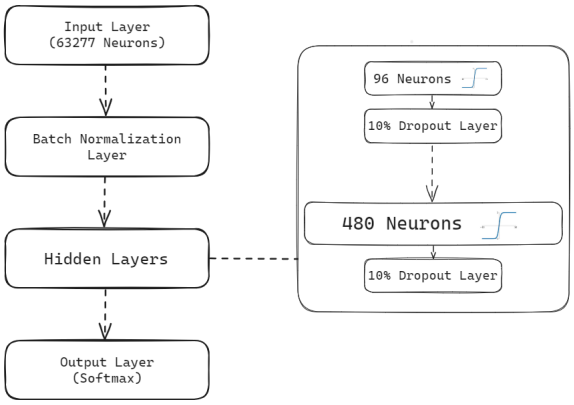

Figure 3: MLP model architecture

MLP with Word2Vec Embeddings: similar to the previous model, but utilizing Word2Vec embeddings as inputs, specifically employing the pretrained Google News dataset model. This model, trained on approximately 100 billion words, generates word embeddings in 300 dimensions and is accessed through the gensim API. To handle out-of-vocabulary (OOV) words, we appended a special token vector in our defined embedding function. Additionally, we employed the average pooling strategy instead of the maximum pooling to create a single embedding that captures the overall context of each news article. This approach ensures a balanced representation by averaging the word vectors. Moreover, label encoding of the outcome variable has also produced slightly better accuracies on this model's final results. Finally, contrary to the previous models, we found out that the results were better without conducting lemmatization.

Utilizing RandomSearch for hyperparameter tuning, a similar architecture to the previous model was achieved, as illustrated in **Figure 4**.

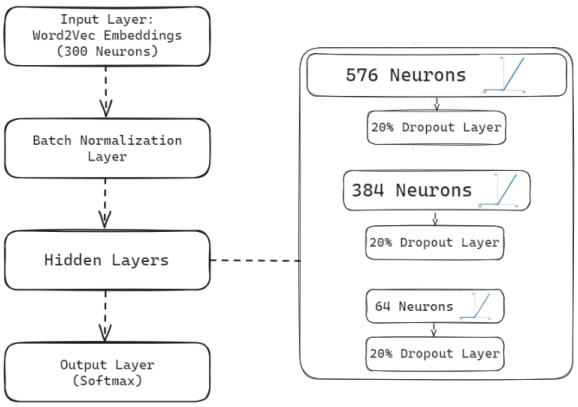

Figure 4: Word2Vec MLP model architecture

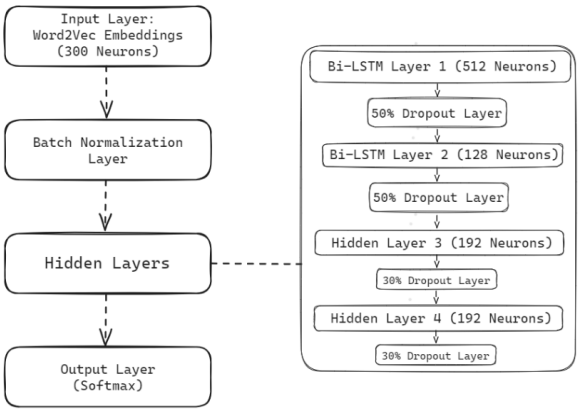

Figure 5: Bi-LSTM Model Architecture

This model's input layer consists of 300 neurons, corresponding to the fixed dimensions of the word embeddings. Additionally, it includes an extra hidden layer, and no L2 regularization was needed.

Bidirectional Long Short-Term Memory (Bi-LSTM): following the learnings from the MLP model, we used Word2Vec embeddings to prepare the data for the Bi-LSTM model. For the hyperparameter tuning, we tried both Hyperband and Random Search, but soon discarded Hyperband since it took much more time to find comparable results to Random Search. Therefore we used Random Search to perform the hyperparameter tuning. The search parameters were the number of Bi-LSTM layers, Dense layers, number of units for each, dropout, and learning rate. The tanh activation function had been consistently selected as the best one in previous searches, so we decided to keep that and not tune for other ones for the final search. The final Random Search was conducted over 30 trials with a maximum of 15 epochs each, using the adam optimizer and sparse categorical cross entropy as a loss function. Early stopping was added to prevent a trial from running unnecessarily long. As depicted in **Figure 5**: The final optimal hyperparameters were two Bi-LSTM layers with 512 and 128 units respectively, two Dense layers with 192 units, a learning rate of 0.001 as well as Dense dropout and Bi-LSTM dropout of 0.3 and 0.5, respectively.

Bidirectional Encoder Representations from Transformers (BERT): the model's initial parameters were set according to recommendations from its documentation (BERT, n.d.) and common practices. The BERT model's fine-tuning was carried out by analyzing the results of each run and manually tweaking relevant hyperparameters to evaluate changes in results.

5 Result Analysis

After selecting the best-performing hyperparameters for each model, every algorithm was re-trained and tested on the entire dataset using the same hardware. This process was timed to compare the resource requirements for each model. The evalua-

tion metrics employed to assess performances are Accuracy, Precision, Recall, and F1-score:

$$\text{Accuracy} = \frac{\text{Number of correct predictions}}{\text{Total number of predictions}}$$

$$\text{Precision} = \frac{\text{Number of true positive predictions}}{\text{Total number of predicted positives}}$$

$$\text{Recall} = \frac{\text{Number of true positive predictions}}{\text{Total number of actual positives}}$$

$$F_1 \text{ Score} = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$$

The following metrics were obtained from the results on the testing dataset:

Model	Acc.	Prec.	Rec.	F1	Time
MNB	77.04%	78%	77%	77%	1.3s
MLR	78.82%	79%	79%	79%	9.8s
MLP	79.25%	79%	79%	79%	11m26s
MLP + W2V	79.51%	80%	80%	79%	3m55s
Bi-LSTM + W2V	79.64%	80%	80%	79%	3h17m
BERT	84%	82%	82%	83%	25h33m

As displayed in the above table, the BERT model shows the best performance among all, with an accuracy of 84%. It is followed by the two models that use Word2Vec word embeddings: Bi-LSTM (79.6%) and MLP (79.51%). The base MLP model, using TF-IDF sparse vectors as input, follows (79.25%), and the Multinomial Logistic Regression (78.82%) and Naïve Bayes (77.04%) models complete the list.

Notably, none of our model's F1 scores surpass the Precision or Recall percentages by more than 1%. This indicates that each model effectively balances both metrics without compromising accuracy. Regarding computational efficiency, Naïve Bayes is the fastest, completing the task in little more than a second. In contrast, the most complex models require substantially more time to execute, with Bi-LSTM taking over 3 hours and the BERT model surpassing 25 hours.

6 Discussion

6.1 Model Comparison

As can be seen in the results, the BERT model has outperformed all other models across all metrics. However, this superior performance at multiclass classification comes at the cost of significantly higher computational time, as it took 25 hours to train and evaluate. This should be taken into consideration when comparing the models.

Our analysis revealed that in terms of the chosen metrics, even the simplest models - Naive Bayes and Logistic Regression - do not perform significantly worse than the most sophisticated ones while taking only a few seconds to run. The Multinomial Logistic Regression model, which has virtually the same scores as all of the more complex models except BERT, took only 9.8 seconds to run, compared to the Bi-LSTM model, which took over 3 hours. This time does not include the time spent refining the input data and the hyperparameter tuning, which similarly took much longer for the more complex models.

Similarly, the increase in accuracy does not prove to be proportional to the increase in running time for our models. With each model being more complex than the previous one, the expectation was that the more complex model would always

easily outperform the simpler model in terms of the chosen metrics. However, moving from Logistic Regression to MLP only granted a 0.43% increase in accuracy (0.69% with Word2Vec), and moving further to Bi-LSTMs merely increased it by a further 0.13%. Looking at the running times of 9.8 seconds compared to 3 hours and 17 minutes, 0.82% increased accuracy is arguably not enough.

After all, only the BERT model, with 25 hours of running time, could show a sizable increase in performance compared to all other models. When choosing models, one should be aware of these figures to decide what is best to use.

Overall, our results highlight the trade-off between model performance and computational time. While BERT offers the best performance, its high computational cost may sometimes be impractical. With limited computational resources and time, models like Multinomial Logistic Regression or MLP with Word2Vec embeddings shine with relatively high performance and manageable training times.

6.2 Limitations

While NLP applications such as our project show a lot of potential in making various tasks more straightforward, they have limitations that must be addressed to generate better results. One of the most evident ones is the computational resources required to train these models. As showcased by our model's results, achieving high accuracy scores takes a lot of time when solving a complex problem. Most of the models we implemented include hyperparameter tuning, except for BERT. Still, BERT achieved higher scores than the other models; however, it also took considerably more time to run. This difference in time to run is the result of BERT's architecture. With more computational resources, hyperparameter tuning could have been implemented for our BERT model, and potentially, its scores could have been increased substantially. However, this would again come at the cost of substantial increases in running time.

Another limitation is the ability of such models to generalize to new data and avoid the risks of overfitting. Various techniques were employed to deal with this for HuffPost data. However, a most interesting question remains: How would the models generalize to similar data but from different sources, and how well could they classify those articles in our categories if applicable? This could prove to be a most difficult challenge, considering that the models resulted in an accuracy of around 80% on our validation data, which also contained HuffPost articles.

Finally, another limitation of natural language processing techniques is ambiguity. NLP systems struggle to capture various elements of natural language, resulting in various categories of ambiguities that are not understood by such algorithms (Li et al., 2024). Scholarly work is being invested in solving ambiguity problems inherent in NLP systems, and according to Li et al., developing more nuanced benchmarks and better studying models' behavior could potentially strengthen their ability to capture ambiguities. Advancing the understanding of these ambiguities by algorithms could also potentially lead to better predictions in our models.

7 Conclusion

In this project, we have analyzed which models have the best performance and how they compare to one another when assigning a category to a specific news article. Our results show that, in our case, the more sophisticated models performed better. However, as mentioned before, the trade-off lies in substantial increases in running time. Depending on the scenario, a fast model with only slightly worse performance, such as Multinomial Logistic Regression, might be preferred.

Our findings can provide business value for media companies to be able to categorize their content efficiently. These automatically classified articles could, in turn, assist companies in having better-personalized news feeds for their customers, increasing their popularity.

In the future of this project, the models could be tested on the news from different websites and newspapers from the same categories to see if they manage similar accuracies to those we obtained by just using the HuffPost articles. We could train on different news sources to make the models more robust and generalized.

Finally, having tested all the different models, we can conclude from our results that the more recent techniques and models, such as BERT and Word2Vec, show better results, indicating a promising trajectory of NLP advancements in the future.

References

Bender, E. M., Friedman, B. (2018). Data Statements for Natural Language Processing: Toward Mitigating System Bias and Enabling Better Science. *Transactions of the Association for Computational Linguistics*, 6, 587–604. https://doi.org/10.1162/tacl_a_00041

BERT. (n.d.). Retrieved 29 May 2024, from https://huggingface.co/docs/transformers/model_doc/bert

Devlin, J., Chang, M.-W., Lee, K., Toutanova, K. (2019). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding (arXiv:1810.04805). *arXiv*. https://doi.org/10.48550/arXiv.1810.04805

Géron, A. (2019). Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow: Concepts, tools, and techniques to build intelligent systems (2nd ed.). O'Reilly Media.

Hussain, A., Ali, G., Akhtar, F., Khand, Z. H., Ali, A. (2020). Design and Analysis of News Category Predictor. *Engineering, Technology Applied Science Research*, 10(5), Article 5. https://doi.org/10.48084/etasr.3825

Li, M. Y., Liu, A., Wu, Z., Smith, N. A. (2024). A Taxonomy of Ambiguity Types for NLP (arXiv:2403.14072). *arXiv*. https://doi.org/10.48550/arXiv.2403.14072

Misra, R. (2022). News Category Dataset (arXiv:2209.11429). *arXiv*. https://doi.org/10.48550/arXiv.2209.11429

MultinomialNB. (n.d.). Scikit-Learn. Retrieved 31 May 2024, from https://scikit-learn/stable/modules/generated/sklearn.naive_bayes.MultinomialNB.html

Nugroho, K. S., Sukmadewa, A. Y., Yudistira, N. (2021). Large-Scale News Classification using BERT Language Model: Spark NLP Approach. *6th International Conference on Sustainable Information Engineering and Technology 2021*, 240–246. https://doi.org/10.1145/3479645.3479658

Schuster, M., Paliwal, K. K. (1997). Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing*, 45(11), 2673–2681. https://doi.org/10.1109/78.650093

Text.BertTokenizer | Text. (n.d.). TensorFlow. Retrieved 31 May 2024, from https://www.tensorflow.org/text/api_docs/python/text/BertTokenizer

Yuan, G.-X., Ho, C.-H., Lin, C.-J. (2012). Recent Advances of Large-Scale Linear Classification. *Proceedings of the IEEE*, 100(9), 2584–2603. https://doi.org/10.1109/JPROC.2012.2188013

Appendix

A. Original Data News Categories

U.S. News	Comedy
Parenting	World News
Culture & Arts	Tech
Sports	Entertainment
Politics	Weird News
Environment	Education
Crime	Science
Wellness	Business
Style & Beauty	Food & Drink
Media	Queer Voices
Home & Living	Women
Black Voices	Travel
Money	Religion
Latino Voices	Impact
Weddings	College
Parents	Arts & Culture
Style	Green
Taste	Healthy Living
The WorldPost	Good News
WorldPost	Fifty
Arts	Divorce

Appendix

B. Training Validation: Accuracy & Loss