Advanced Natural Language Processing (968G5): Propaganda Detection

Candidate Number: 277178

University of Sussex

ABSTRACT: In today's digital world, the rapid spread of information through online platforms has led to concerns about information integrity. This encompasses fake news, misleading and provocative content which represent challenges for both social media platforms and government bodies. In this report, different natural language processing techniques are explored and their effectiveness in detecting propaganda is evaluated. Additionally, a brief comprehensive related work has been discussed. Moreover, the report focuses on two primary detection approaches: binary classification and multiclass classification. Here, popular pre-trained large language models (BERT), A pretrained GloVe (Global Vectors for Word Representation) word embedding model, and LSTM have been utilized. We have discussed the mechanism of these models and evaluated their effectiveness in detail. Furthermore, a comparative analysis is conducted between binary and multi-class classification models, considering future improvements. Here, BERT based Neural Network model outperforms compared to other methods, achieving an accuracy of 73% and F1 score of 74% in binary classification of propaganda detection.

KEYWORDS: Propaganda, Natural Language Processing, BERT, Language Models, Deep Learning, Word Embedding, GloVe

1 INTRODUCTION

Propaganda refers to the expression of opinion or an action by individuals or groups, with specific intentional aim of influencing other individuals or groups [1]. With the rise of the internet, it has made it even easier for people to freely express their opinions and share information, some of which may be propagandist in nature and may lead to accept the misleading or provocative contents presented in articles. And social media further exacerbates this issue as the contents can rapidly reach to millions of users easily. Hence It is crucial to accurately identify instances of propaganda to prevent successful attempts to influence and persuade individuals. However, it's extremely challenging task to determine the propagandists contents due to its nature, which often blends with true information or news. Within this vast information in online contents can originate from various sources such as governments, media or individuals, which further makes it more difficult to detect the content with propaganda from actual facts.

This report addresses two tasks related to propaganda detection. Task 1 focuses on designing methods and evaluating natural language techniques to determine whether a given text contains propaganda to identify 'propaganda' or 'not_propaganda' instances. To accomplish this, we employ binary classification, where texts are propaganda or not propaganda based on the presence or absence of propagandistic elements.

Task 2 involves identifying snippets or spans of text known to contain propaganda, tokens <BOS> and <EOS> which indicates the beginning and end of the span of text within the sentence. Each snippet is annotated with 9 different labels such as:

- flag_waving
- appeal_to_fear_prejudice
- causal_simplification

- doubt
- exaggeration,minimisation
- loaded_language
- name_calling,labeling
- repetition
- not_propaganda

In order to perform the two tasks, we will built classifiers using 2 very different approaches. These are as follows:

- A pretrained GloVe (Global Vectors for Word Representation) word embedding model will be used to generate a vector representation of text, which will be then passed into LSTM for binary classification and Neural Network Classifier for solving multiclass probelem.
- A pretrained large language model BERT (Bidirectional Encoder Representations from Transformers) will be used to generate a contextualised embedding, which will be used downstream to perform text classification tasks by a neural network for both binary and multiclass classification.

Moreover, A GloVe-NN and BERT-NN will be used for approaching the binary classification of task 1 and A GloVe-LSTM and BERT-NN model will be utilized for solving multiclass classification of task 2.

The rest of the paper is organized as follows. Section 2 presents literature survey of related models for detecting propaganda techniques using natural language processing techniques. Section 3 discusses the dataset and preprocessing step before training the model. Section 4 discusses the different methods for solving the tasks and model architecture details. Then results and evaluation of the approached methods are discussed in section 5. Finally section 6 concludes with future directions.

2 Related Works

Different natural language models have been explored for propaganda detection in online news articles over the years. In [2], Yoosuf et al. utilised a sampling strategy, learning rates, masked logics and a fine-tuned BERT which is a well-kown pre-trained model, for identifying and categorising textual segments. He used the dataset with a 18 given propaganda techniques in news articles. In this model, head importance, average attention head, masking out layers, and the distribution of the entropy were investigated. The linguistically advanced pre-trained language model, RoBERTa (based on the Bidirectional Encoder Representations from Transformers, BERT), has been utilised in [3] to identify propaganda instances in online news. Using SemEvak-2020 task 11 dataset, the authors optimised the model which is able to detect by utilising the propaganda techniques curated by Martino et al. [4], achieveing F1 score of 60.2%. The model achieved better accuracy for some particular techniques such as ”Loaded language” and ”Appeal to authority” for binary classification task of propaganda detection.

Dewantara et al. [5] used three different deep learning models such as LSTM, combination of LSTM and CNN and CNN models. The combined model of LSTM and CNN outperform the other models with 79.4% F1 score. Another research by [6] only used BERT for classifying propaganda techniques in news article. In another approach in [7], an LSTM model with GloVe word embeddings was used without human-engineered features for representation learning. This model achieved macro-F1 score of 0.423 on the development set and F1 score of 0.406 on the test dataset.

A framework (ProSOUL) for propaganda detection is presented in [8], which is for online urdu language contents in news. A total of 11,754 urdu news are labeled for this model for training the classifiers. They modified different classifiers such as BERT, ngram, Word2Vec, News Landscape (NELA), and also built different hybrid model. The model achived 91% accuracy for the hybrid model of NELA, n-gram for text classification. Here, it is also observed in the article that the word2Vec model outperformed the BERT classifier with 87% accuracy for Urdu text classification.

3 Dataset and Preprocessing

The dataset used in this project is obtained from given training and validation dataset. The training set contains – 2414 records and validation dataset contain 580 records. Each training and validation dataset consists of 2 columns namely – 'label' and 'tagged_in_context'. The 'tagged_in_context' column contains <BOS>and <EOS> which indicate the beginning and end of the span of text (within the sentence). This is actually annotated with the one of the 9 propaganda techniques. The 9 different techniques are labeled as: 1. flag waving, 2. appeal to fear prejudice, 3. causal simplification, 4. Doubt, 5. exaggeration,minimisation 6. loaded language, 7. name calling,labeling, 8. Repetition, 9. not propaganda. The distribution of data in the training and validation datasets are shown in fig.

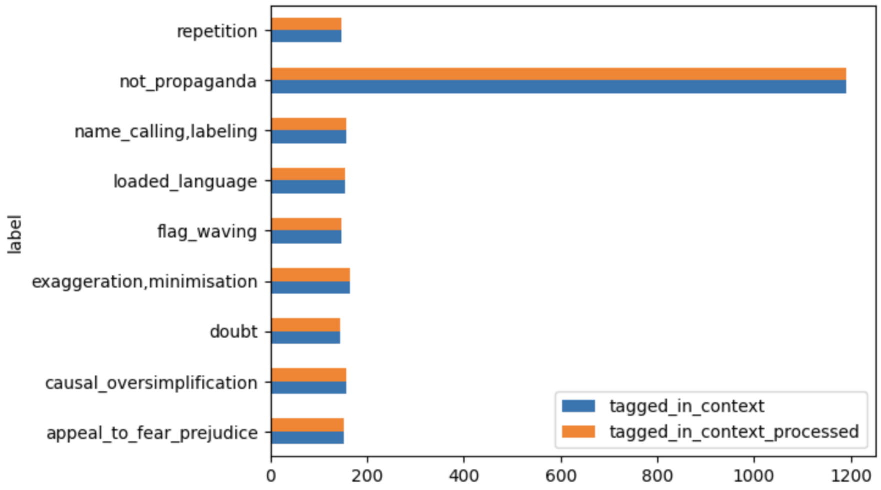
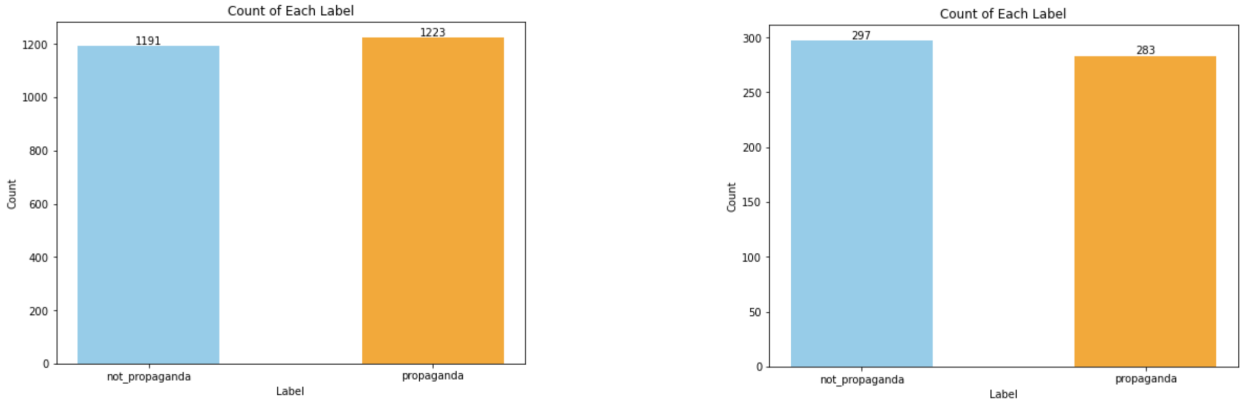


Figure 1: Different propaganda labels in training dataset

Here, several preprocessing techniques are conducted before training the model, which includes text leaning for removing special characters, punctuation, lowercase folding, stopword removal (removing common words like "the", "a" ect.) and tokenization. After applying preprocessing steps, the training and validation dataset is now cleaned, structured for the model. here, the training dataset has been randomly divided into two portions: 80% for training and 20% for testing..



(a) Training Dataset

(b) Total Count of Propaganda labels

Figure 2: Total binary labels in training dataset

4 Methodology

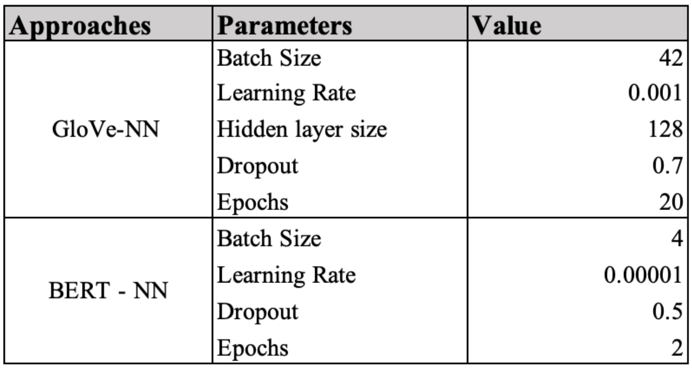
For task 1, I have utilized two approaches, one is combination of GloVe and Neural Network based classifier and another one is BERT and Neural Network based classifier. For task – 1, I've prepared the dataset as labeled as ‘propaganda’ and will use another label ‘not_propaganda’ for binary classification, which is later transformed into 0 or 1 for binary classification label.

4.1 Binary Classification

4.1.1 Task - 1.1: GloVe- NN based classifier

For the first approach, I have used pre-trained GloVe word embeddings in combination with a Feed-Forward Neural Network (FFNN) for propaganda detection. GloVe (Global Vectors for Word Representation) which is an unsupervised algorithm used for generating word embedding. By analyzing co-occurrence in a corpus, these embedding are able to capture semantic relationships between words. The process begins by constructing the co-occurrence matrix. Through these matrix factorizations, these occurrences can be converted into dense vector representations, which is then fed into simple feed-forward neural network. After preprocessing the data, the pre-trained GloVe model (’glove-wiki-gigaword-50’) has been used from genism library. After that I have built the word embedding function which takes input as text and pre-trained GloVe mode and creates a single vector representation for the whole sentence. Then the training (vector representation of each sentence obtained by averaging the word embeddings) and testing data (labels) are converted to PyTorch tensors and here, labels are encoded using scikit-learn’s ‘LabelEncoder’ library.

Here, for binary classification, FFNN classifier is created which is a Feed-Forward Neural Network and it has an input layer, a hidden layer with ReLU activation function and an output layer with softmax activation function. Here, forward method is used to forward pass of the neural network.



Approaches	Parameters	Value
GloVe-NN	Batch Size	42
	Learning Rate	0.001
	Hidden layer size	128
	Dropout	0.7
	Epochs	20
BERT - NN	Batch Size	4
	Learning Rate	0.00001
	Dropout	0.5
	Epochs	2

Table 1: Hyperparameter settings of GloVE+NN and BERT-NN models

The neural network consists of two fully connected layers which has input layer of 50 neurons, hidden layer of 128 neurons which determines the capacity of the model to learn the complex patterns in data and a output layer of 2 neurons corresponding to binary classification task. Here, these two layers are separated by a ReLU (Rectified Linear Unit) activation function and a dropout layer. The ReLU is applied after the first fully connected layer to introduce non-linearity in to the model and dropout is applied after ReLU to prevent any overfitting in the model with dropout value 0.7.

Additionally, the Cross-Entropy Loss function was used and Adam optimizer with learning rate 0.001 was used for updating the model parameters during training. The model is trained for 20 epochs with a batch size of 42 and for each iteration the optimizer calculates the gradients, updates the model parameters, and minimizes the loss using backpropagation. After the training, the model is evaluated on the test data and also classification report is produced to provide a overview of the models’ performance, presenting precision, recall, and F1-score for each propaganda technique class, alongside overall accuracy, which will be discussed in the result section 5.

4.1.2 Task - 1.2: BERT- NN based classifier

For the second approach, I have used the pre-trained BERT model for generating contextualized word embeddings. BERT, a new linguistic model which is a transformer-based neural network architecture. It was built in 2018 under the leadership of J. Devlin called BERT. It is designed to learn the bidirectional text representations by utilizing masked words prediction in text. Here, I have used the pre-trained ‘bert-base- uncased‘ BERT model. Comparing with GloVe, which only captures the word distributional semantics on co-occurrence patterns for embedding, BERT generates contextualized word embedding. These BERT built embedding facilitates the model to learn the word meaning pattern across different textual scenarios.

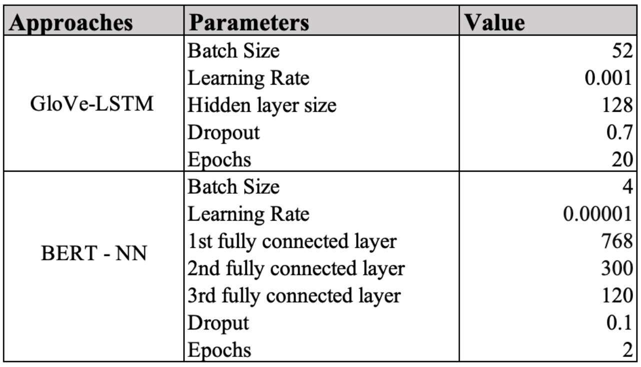
A dropout layer is applied with a value of 0.5 which is a regularization technique for preventing over-fitting in the model during training process. After that the dropout of the dropout then passed through the linear layer which has a input size of 768 and output size of 2. Then for making non-linearity into model and learning complex patterns, the activation function, ReLU is applied to the output of the linear layer. During training the model takes the input of epochs of 2, batch size of 4. Additionally, the model is trained using the Adam optimizer and evaluated using the Cross Entropy Loss function with learning rate 0.00001.

4.2 Multiclass Classification

4.2.1 GloVe - LSTM based classifier

For this task, I have implemented a multiclass classification model for propaganda detection using GloVe embeddings and LSTM (Long Short-Term Memory) neural networks. Long short-term memory (LSTM) is a popular deep learning technique which extensively used in the field of text classification [6]. This is because LSTM is capable of detecting contextual information in the long text sequence.

The architecture of the model consists of an embedding layer, an LSTM layer, a fully connected layer and an output layer. Initially, the embedding layer encodes each word by utilising pre-trained word embeddings – GloVe model. The LSTM layer is implemented to obtain contextual information. Here, ‘glove-wiki-gigaword-50‘ model has been used for word embedding. The LSTM-based classifier has input_size of 100, hidden_size of 128 and output_size of 8 which is the of the multi-class labels. Here, Dropout value of 0.7 is used for preventing overfitting in the model. After that, training, validation and test data are loaded by utilizing Data loader objects with a batch size of 16. The model is trained using the Cross Entropy Loss function and Adam optimizer with learning rate 0.001 and batch size of 52.



Approaches	Parameters	Value
GloVe-LSTM	Batch Size	52
	Learning Rate	0.001
	Hidden layer size	128
	Dropout	0.7
	Epochs	20
BERT - NN	Batch Size	4
	Learning Rate	0.00001
	1st fully connected layer	768
	2nd fully connected layer	300
	3rd fully connected layer	120
	Droput	0.1
	Epochs	2

Table 2: Hyperparameter settings of GloVe-LSTM and BERT-NN models

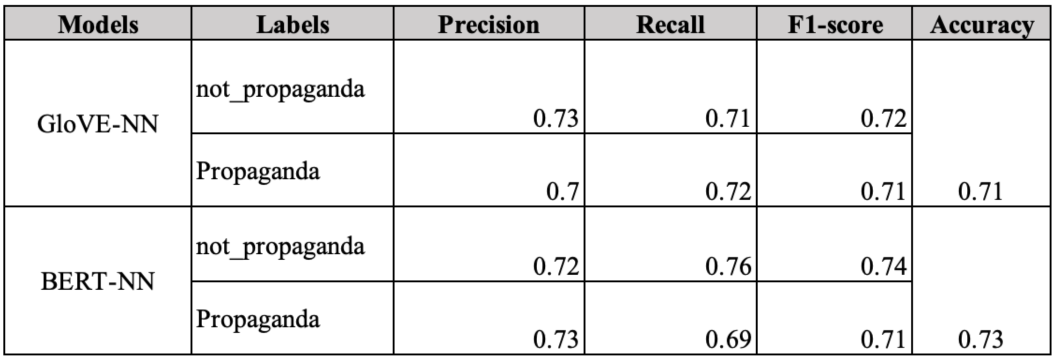
4.2.2 BERT- NN based classifier

For second approach, I have used the pre-trained BERT model – ‘bert-case-uncased‘ for extracting the contextualized word embeddings from the text. After that the processed embeddings are then passed through the 3 fully connected linear layers. Here, the first fully connected layers has input size of 768 neurons, 2nd fully connected layer’s input size of 300 and output size of 120. The third fully connected layer has input size of 120 neurons and output size of 8 (multiclass labels).

A dropout regularization with value 0.1 is applied for preventing over-fitting issue. The last fully connected layer is then processed by the softmax activation function and generated class probabilities. After that the training function trains the model with given training data and assessed on validation data. During training Cross Entropy Loss function is applied for determining the loss. The model is trained using the Cross Entropy Loss function and the optimizer ‘Adam‘ is used with learning rate 0.000001 and batch size of 4. Finally, the evaluation function evaluates the trained model performance using test dataset. By contrasting the model’s predictions with the actual labels in the test dataset, it evaluates the model’s accuracy which will be discussed in section 5.

5 Results and Evaluation

In Task 1, as illustrated in the table 3, the GloVe-NN model achieves an accuracy of 71% on the test set. For the ’not_propaganda’ class, it exhibits precision, recall, and F1-score of 0.73, 0.71, and 0.72, respectively. Here, for the ’propaganda’ class, these metrics are reported as 0.70, 0.72, and 0.71. This implies a slightly higher precision for ’not_propaganda’ instances and slightly greater recall for ’propaganda’ instances. The macro-averaged precision, recall, and F1-score are all 0.72, indicating a balanced performance across both classes. Similarly, the weighted average metrics for recall, accuracy, and F1-score, all at 0.72, suggest consistent performance across the dataset.



Models	Labels	Precision	Recall	F1-score	Accuracy
GloVE-NN	not_propaganda	0.73	0.71	0.72	0.71
	Propaganda	0.7	0.72	0.71	
BERT-NN	not_propaganda	0.72	0.76	0.74	0.73
	Propaganda	0.73	0.69	0.71	

Table 3: Binary classification report of GloVe+NN and BERT-NN models

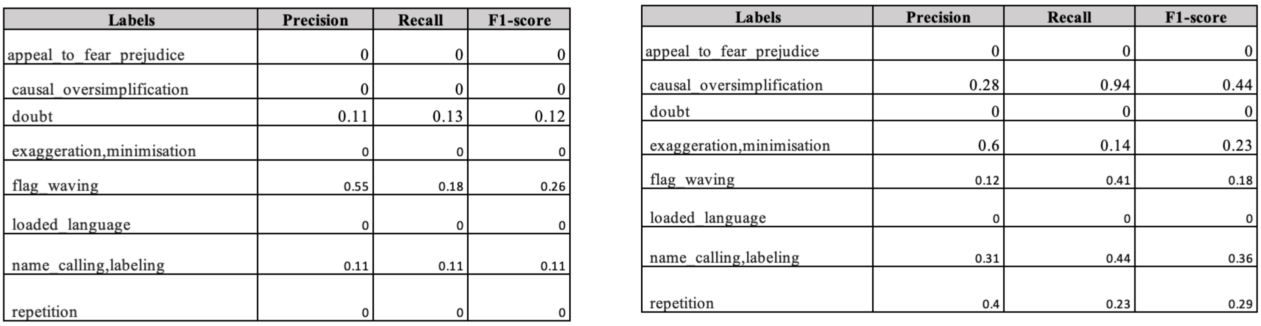
On the other hand, the BERT-NN model achieves an accuracy of 73% on the test dataset. It demonstrates precision, recall, and F1-score of 0.72, 0.76, and 0.74, respectively for the ’not_propaganda’ class, while for the ’propaganda’ class, these metrics are reported as 0.73, 0.69, and 0.71. Here too, there’s a slight higher precision for ’propaganda’ instances and higher recall for ’not_propaganda’ instances. The macro-averaged precision, recall, and F1-score are noted as 0.71, 0.70, and 0.70, respectively, suggesting balanced performance across both classes.

In Task 2, the classification report which is depicted in Figure 3 (a), outlines the performance of the GloVE-LSTM model across propaganda technique labels. Notably, the class ”flag_waving” displayed the highest precision, achieving a value of 0.55, indicating a relatively high proportion of true positive predictions among instances classified as ”repetition.” On the other hand, the class ”doubt” recorded the lowest precision, standing at 0.11, reflecting a lower accuracy in identifying instances of ”doubt.”

In terms of recall, the class ”flag_waving” emerged with the highest value of 0.18, implying that the model effectively captured a significant portion of actual instances belonging to this class. On the other hand, the class ”doubt” displayed the lowest recall, amounting to 0.13, suggesting a relatively lower ability to identify instances of ”doubt.”

The F1-score, which balances precision and recall, ranged from 0 for several labels like ”repetition”, “casual_oversimplification”, “loaded_language” and ”exaggeration,minimisation”. Furthermore, through error analysis, it is observed that the model struggles in classifying these instances. These difficulties may arise from the complexity and ambiguity inherent in language, potentially impacting the model’s classification accuracy for these particular classes.

For the BERT-NN model, it is observed from the figure 3(b) that the classes ”causal_oversimplification” and ”repetition” exhibit relatively higher precision, recall, and F1-score, with values ranging from 0.28 to 0.53. This indicates the model’s ability to accurately identify instances of these propaganda techniques. On the other hand, the classes ”appeal_to_fear_prejudice,” ”doubt,” and ”loaded_language” have precision, recall, and F1-score values of 0, which indicates that the model struggled to correctly classify instances belonging to these classes. Additionally, the class ”casual_oversimplification” and ”name_calling,labeling” demonstrate



Labels	Precision	Recall	F1-score
appeal_to_fear_prejudice	0	0	0
causal_oversimplification	0	0	0
doubt	0.11	0.13	0.12
exaggeration,minimisation	0	0	0
flag_waving	0.55	0.18	0.26
loaded_language	0	0	0
name_calling,labeling	0.11	0.11	0.11
repetition	0	0	0

(a) Classification report of GloVe-LSTM model

Labels	Precision	Recall	F1-score
appeal_to_fear_prejudice	0	0	0
causal_oversimplification	0.28	0.94	0.44
doubt	0	0	0
exaggeration,minimisation	0.6	0.14	0.23
flag_waving	0.12	0.41	0.18
loaded_language	0	0	0
name_calling,labeling	0.31	0.44	0.36
repetition	0.4	0.23	0.29

(b) Classification report of BERT-NN model

Figure 3: Multiclass classification report

a relatively high recall of 0.94 and 0.44 respectively, which suggests that the model has a better ability to capture true positive instances, but with a lower precision and F1-score. Overall, while the model demonstrates strengths in identifying certain propaganda techniques, such as ”causal_oversimplification” and ”repetition,”, it encounters challenges in accurately classifying other propaganda techniques which can be improved in future by utilizing different hyper-parameter tuning.

5.1 Comparison Analysis

In binary classification task, it is observed from the table 4 that BERT classifier exhibits outstanding performance even with minimum epochs of 4 and the model achieves an accuracy of 73%, compared to GloVe classifier's accuracy of 71% with higher epochs and batch size. This significant performance proves the BERT model's ability to provide contextualized word embeddings in various context of words. As BERT model is a pretrained model on vast corpus of texts, it can extract valuable semantic information from large amount of texts which enhances its classification ability. Thus, the BERT model can handle raw texts without preprocessing. On the other hand, GloVe model solely relies on the text processing utilizing tokenization, dramatization etc. for generating such embedding. Moreover, during training BERT model need typically fewer iteration for training the model as their embeddings are already fined tuned for preprocessing and thus it provides excellent accuracy.

Models	Precision	Recall	F1-score	Accuracy
GloVe-NN	71.5%	71.5%	72.0%	71.0%
BERT-NN	72.0%	72.5%	72.5%	73.0%

Table 4: Comparison of Binary classification models report

However, for multi-class classifiers, the BERT model showcases accuracy of 24% for epochs 5 and the model were not able to classify three propaganda techniques. Although this model has achived better results for some propaganda techniques. This is because of the lack for model training iteration and limited computation power. On the other hand, GloVe-LSTM poorly performed with accuracy of 18% in detecting propaganda. With proper tuning of hyperparameters and also utilizing different regularization techniques can be explored in future for building better model.

6 Conclusion and Future Works

Machine learning and computation linguistics are essential for understanding and reducing the adverse impacts of propaganda in this digital world. This report explores the computational analysis of propaganda in texts through two tasks: detecting propaganda presence and classifying propaganda techniques. Here, two approaches are presented and evaluated for each task. For Task 1, GloVe-NN and BERT-NN are approached.

Here, from the results analysis it is observed that BERT-NN outperformed the GloVe-LSTM method, achieving an accuracy of 73% and F1 score of 74%. For Task 2, GloVe-LSTM and BERT-NN are approached. Here, BERT demonstrated superior performance with an accuracy of 26%, outperforming the GloVe-LSTM. However, performing hyperparameter tuning such as increasing training iteration, batch size, the BERT model can achieve significant better results. Additionally, loss function the Cross Entropy Loss and the Adam (Adaptive Moment Estimation) optimizer algorithm are utilized for model optimization.

Overall, the project enhances the techniques of computational analysis of propaganda in texts and further offers the opportunities for the advancement in building sophisticated model architecture beyond BERT such as GPT, or RoBERTa, for classifying different propaganda techniques. In future, improvement in the models can be achieved by proper tuning of hyperparameters and also utilizing different regularization techniques such as L1, L2, LeakyRelu can be explored to mitigate overfitting issue and improve model generalization.. Also different optimization techniques such as Bayesian, Grid Search, stochastic gradient descent (SGD) can be utilized in future which will enhance the model's ability to learn the complex text patterns related to propaganda.

7 Code Appendix

Please see file for code – 'propaganda_detection_277178.ipynb'

References

[1] Institute for Propaganda Analysis. 1938. How to Detect Propaganda. In Propaganda Analysis. Vol. I of the Publications of the Institute for Propaganda Analysis. chapter 2.

[2] Yoosuf, S. and Yang, Y., 2019, November. Fine-grained propaganda detection with fine-tuned BERT. In Proceedings of the second workshop on natural language processing for internet freedom: censorship, disinformation, and propaganda (pp. 87-91).

[3] Abdullah, M., Altiti, O. and Obiedat, R., 2022, June. Detecting propaganda techniques in english news articles using pre-trained transformers. In 2022 13th International Conference on Information and Communication Systems (ICICS) (pp. 301-308). IEEE.

[4] Martino, G.D.S., Cresci, S., Barrón-Cedeño, A., Yu, S., Di Pietro, R. and Nakov, P., 2020. A survey on computational propaganda detection. arXiv preprint arXiv:2007.08024.

[5] Dewantara, D.S. and Budi, I., 2020, November. Combination of lstm and cnn for article-level propaganda detection in news articles. In 2020 Fifth International Conference on Informatics and Computing (ICIC) (pp. 1-4). IEEE.

[6] Alhindi, T., Pfeiffer, J. and Muresan, S., 2019. Fine-tuned neural models for propaganda detection at the sentence and fragment levels. arXiv preprint arXiv:1910.09702.

[7] Dao, J., Wang, J. and Zhang, X., 2020. YNU-HPCC at SemEval-2020 task 11: LSTM network for detection of propaganda techniques in news articles. arXiv preprint arXiv:2008.10166.

[8] Kausar, S., Tahir, B. and Mehmood, M.A., 2020. ProSOUL: a framework to identify propaganda from online Urdu content. IEEE access, 8, pp.186039-186054.