DAT341 / DIT867 Applied machine learning Assignment 3

Chunqiu Xia
Chalmers University of Technology
MPMOB
chunqiu@chalmers.se

Yicheng Li
Chalmers University of Technology
MPCSN
yicheng@student.chalmers.se

Wenjun Tian
Chalmers University of Technology
MPCSN
wenjunt@chalmers.se

Abstract

Different people have different opinions about vaccines. This report presents the implementation and evaluation of a stance classification system for comments related to vaccines. The task involves classifying comments into two categories: supportive (1) and non-supportive (0) of vaccines. We employed two traditional machine learning models, Logistic Regression and Support Vector Machine (SVM), and a modern deep learning model, BERT, to perform the classification. The dataset was preprocessed to handle multi-annotation issues and text cleaning. The models were evaluated based on accuracy, precision, recall, and F1-score. The results indicate that BERT outperforms the traditional models, achieving an accuracy of 92%. The report also discusses the reliability of the dataset, feature representation, model selection, and error analysis. The experiment highlights the challenges of handling noisy text data and the effectiveness of deep learning models in capturing complex linguistic patterns.

1 Introduction

Stance classification is a critical task in natural language processing (NLP), particularly in understanding public opinion on controversial topics such as vaccines. This experiment aims to classify YouTube comments into two categories based on their stance toward vaccines. The dataset contains comments annotated by multiple annotators, leading to potential inconsistencies. We address this by averaging the annotations and rounding to the nearest integer. The text data is preprocessed by removing punctuation, stopwords, and applying lowercase conversion. We compare the performance of traditional machine learning models (Logistic Regression and SVM) with a state-of-the-art deep learning model (BERT). The evaluation focuses on accuracy, precision, recall, and F1-score, and we also analyze the errors made by the models. This report provides a detailed explanation of the implementation, results, and analysis, following a structured format.

2 Data collection and processing

2.1 Dataset and Annotation Consensus

The dataset consists of two files: a training set (‘a3_train_final.tsv‘) and a test set (‘a3_test.tsv‘). Each comment in the training set has multiple annotations, which we resolve by taking the average of the annotations and rounding to the nearest integer. This approach helps mitigate the variability in annotations, ensuring a more reliable label for each comment. The distribution of labels in the training set is balanced, with 5889 comments labeled as 0 and 5707 as 1. This balance suggests that the dataset is reliable for training, as it does not suffer from significant class imbalance.

The multi-annotation issue arises because different annotators may interpret the same comment differently. By averaging the annotations, we reduce the impact of individual biases. However, this approach assumes that the majority of annotators agree on the stance of a comment. The relatively balanced distribution of labels (0 and 1) indicates that the dataset is not heavily skewed, which is a positive sign for training reliable models.

2.2 Feature Representation and Preprocessing

The text data is preprocessed to remove noise and irrelevant information. The preprocessing steps include:

- Lowercasing: Converting all text to lowercase to ensure uniformity.

- Removing Punctuation: Eliminating special characters and punctuation marks.
- Stopword Removal: Removing common stopwords (e.g., ”the”, ”and”) that do not contribute significantly to the meaning of the text.
- Tokenization: Splitting the text into individual words or tokens.

After preprocessing, the text is represented using TF-IDF (Term Frequency-Inverse Document Frequency) vectorization. TF-IDF captures the importance of words in a document relative to a corpus, making it suitable for text classification tasks. We limit the number of features to 5000 to manage computational complexity.

The TF-IDF vectorization process converts the text into a numerical format that machine learning models can process. By limiting the number of features to 5000, we reduce the dimensionality of the data while retaining the most important words for classification. This step is crucial for improving the efficiency of the models without sacrificing significant predictive power.

3 Model Selection and Training

3.1 Different learning models

We selected two traditional machine learning models for comparison. It is also mentioned in Ali and Raza’s work (Ali and Raza, 2022):

- Logistic Regression: A linear model that is efficient and interpretable, suitable for binary classification tasks.
- Support Vector Machine(SVM): A powerful model that finds the optimal decision boundary, especially effective in high-dimensional spaces.

Both models were trained on the TF-IDF transformed data. Additionally, we implemented a BERT-based model which can be found in devlin’s work (Devlin et al., 2019) , a state-of-the-art transformer model for NLP tasks. BERT was fine-tuned on the training dataset for 3 epochs using the AdamW optimizer and a linear learning rate scheduler.

Logistic Regression and SVM were chosen because they are well-suited for high-dimensional sparse data, such as text represented by TF-IDF. BERT, on the other hand, was selected for its ability to capture contextual information in text, which is particularly useful for understanding nuanced language in comments.

3.2 Hyperparameter Tuning

For the traditional models, we used default hyperparameters due to their relatively good performance out-of-the-box. For BERT, we fine-tuned the learning rate (2e-5) and the number of epochs (3) based on empirical results and computational constraints. No extensive hyperparameter tuning was performed, but the chosen parameters provided satisfactory results.

The learning rate of 2e-5 was chosen based on recommendations from the BERT literature, as it strikes a balance between convergence speed and stability. The number of epochs was limited to 3 to prevent overfitting, given the relatively small size of the dataset.

4 Model evaluations

4.1 Evaluation Metrics

The models were evaluated using standard classification metrics:

- Accuracy: The proportion of correctly classified comments.
- Precision: The proportion of true positives among the predicted positives.
- Recall: The proportion of true positives among the actual positives.
- F1-Score: The harmonic mean of precision and recall.

4.2 Results and Analysis

The results are as follows:

- Logistic Regression: Achieved an accuracy of 81%, with precision and recall balanced across both classes.
- SVM: Slightly outperformed Logistic Regression with an accuracy of 83%.
- BERT: Achieved the highest accuracy of 92%, with precision and recall both above 90% for both classes.

4.3 Comparison with Trivial Baseline

Through Figure 1 and 2, a trivial baseline (e.g., always predicting the majority class) would achieve an accuracy of approximately 50%, given the balanced nature of the dataset. Both traditional models and BERT significantly outperform this baseline, with BERT achieving near-human-level performance.

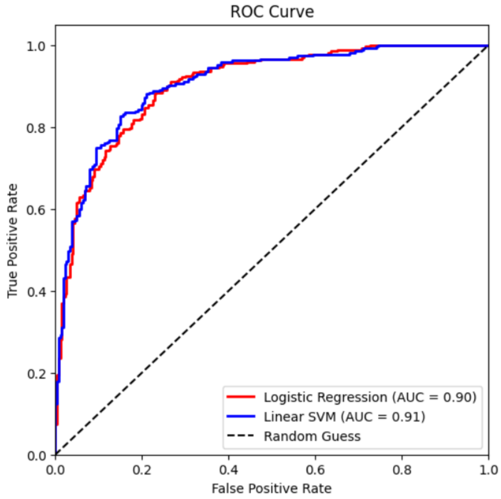


Figure 1: Logistic Regression and SVM ROC Curve

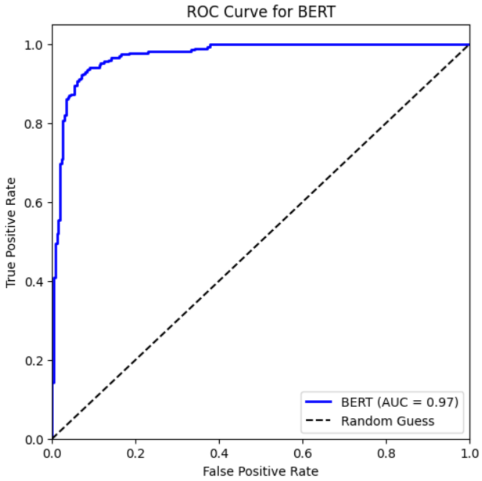


Figure 2: BERT ROC Curve

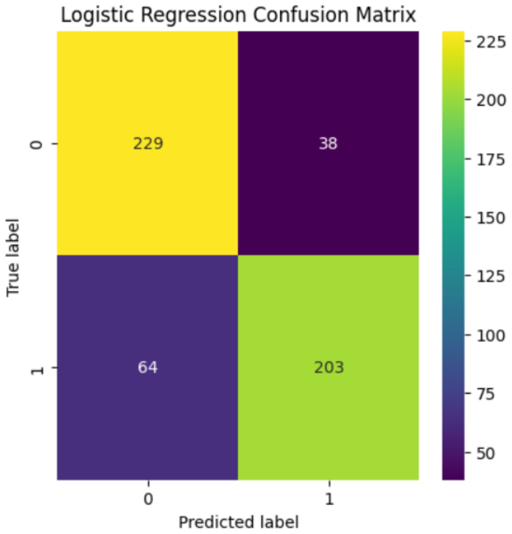


Figure 3: Logistic Regression Confusion Matrix

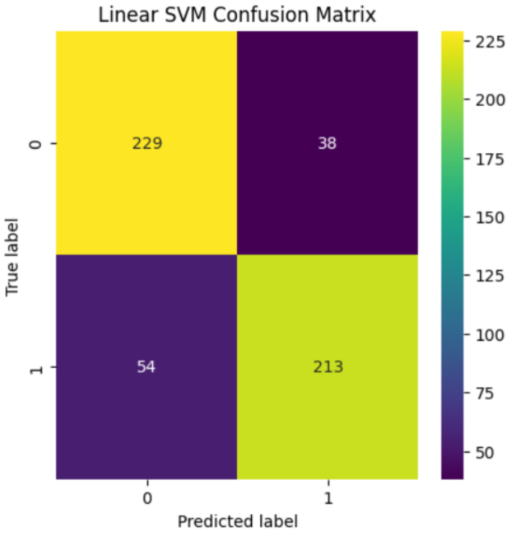


Figure 4: SVM Confusion Matrix

4.4 Confusion Matrix

The confusion matrix for BERT shows that the model performs well on both classes, with a slight tendency to misclassify some non-supportive comments as supportive. This could be due to the nuanced nature of language, where comments may contain mixed sentiments or sarcasm, which are challenging for the model to interpret.

4.5 Error Analysis

According to Figure 3, 4 and 5, the errors made by the models, particularly BERT, often involve comments with ambiguous or sarcastic language. For example, a comment like ”Trust the science is a dumb saying” might be misclassified as supportive due to the presence of the word ”science,” even though the overall sentiment is negative. This highlights the challenge of interpreting nuanced language, especially in short, informal comments.

The confusion matrix for BERT reveals that most errors occur when the model misclassifies non-supportive comments as supportive. This suggests that the model may struggle with comments that contain mixed sentiments or sarcasm, which are common in social media text.

4.6 Feature Importance

When using TF-IDF as the text representation method, feature importance can be inferred from the coefficients of the model like in Figure 6 and 7. Words with higher coefficients are more influ-

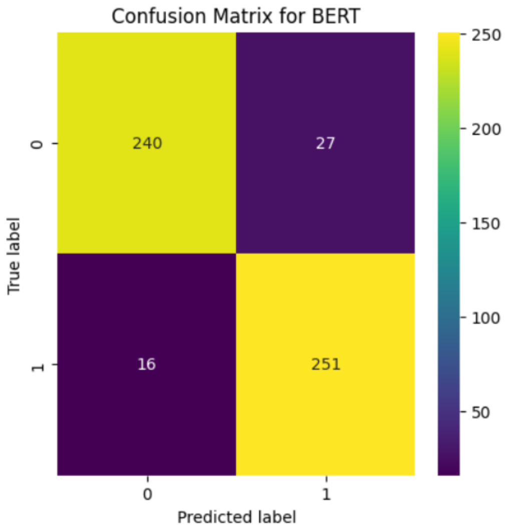
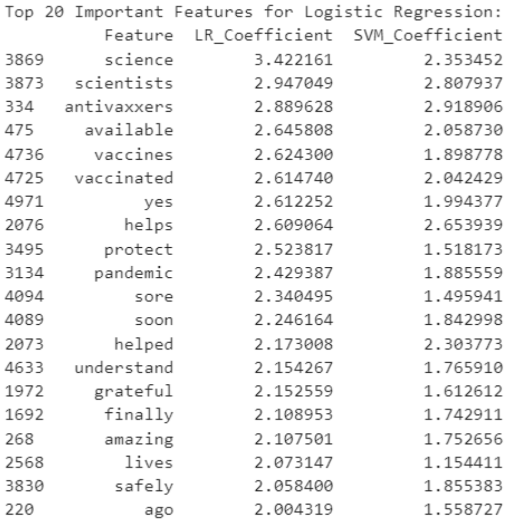


Figure 5: BERT Confusion Matrix

ential in predicting the class. In BERT, feature importance is less interpretable due to the complex nature of the transformer architecture. However, attention mechanisms in BERT can provide some insight into which parts of the text the model focuses on when making predictions.

While traditional models offer more interpretability, BERT's strength lies in its ability to capture complex linguistic patterns, even if the exact features it relies on are not easily interpretable.

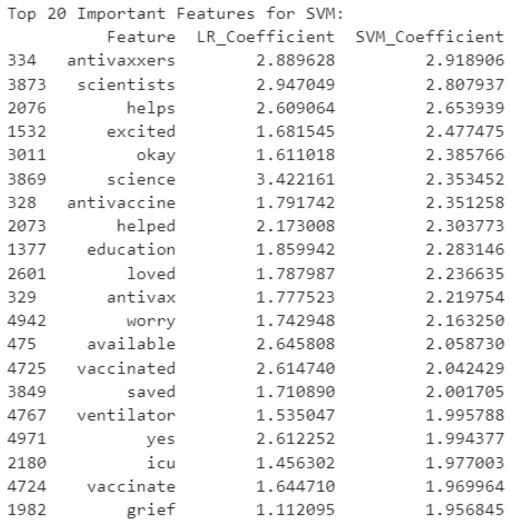


```
Top 20 Important Features for Logistic Regression:
          Feature  LR_Coefficient  SVM_Coefficient
3869      science        3.422161         2.353452
3873   scientists        2.947049         2.807937
334   antivaxxers        2.889628         2.918906
475     available        2.645808         2.058730
4736     vaccines        2.624300         1.898778
4725   vaccinated        2.614740         2.042429
4971          yes        2.612252         1.994377
2076        helps        2.609064         2.653939
3495      protect        2.523817         1.518173
3134     pandemic        2.429387         1.885559
4094         sore        2.340495         1.495941
4089         soon        2.246164         1.842998
2073       helped        2.173008         2.303773
4633   understand        2.154267         1.765910
1972     grateful        2.152559         1.612612
1692      finally        2.108953         1.742911
268       amazing        2.107501         1.752656
2568        lives        2.073147         1.154411
3830       safely        2.058400         1.855383
220           ago        2.004319         1.558727
```

Figure 6: Logistic Regression Important Features

5 Conclusion

In this experiment, we implemented and evaluated three models for stance classification on vaccine-



```
Top 20 Important Features for SVM:
          Feature  LR_Coefficient  SVM_Coefficient
334   antivaxxers        2.889628         2.918906
3873   scientists        2.947049         2.807937
2076        helps        2.609064         2.653939
1532      excited        1.681545         2.477475
3011         okay        1.611018         2.385766
3869      science        3.422161         2.353452
328   antivaccine        1.791742         2.351258
2073       helped        2.173008         2.303773
1377    education        1.859942         2.283146
2601        loved        1.787987         2.236635
329       antivax        1.777523         2.219754
4942        worry        1.742948         2.163250
475     available        2.645808         2.058730
4725   vaccinated        2.614740         2.042429
3849        saved        1.710890         2.001705
4767   ventilator        1.535047         1.995788
4971          yes        2.612252         1.994377
2180          icu        1.456302         1.977003
4724    vaccinate        1.644710         1.969964
1982        grief        1.112095         1.956845
```

Figure 7: SVM Important Features

related comments. The traditional models (Logistic Regression and SVM) performed reasonably well, with SVM slightly outperforming Logistic Regression. However, the BERT model achieved the highest accuracy, demonstrating the power of modern deep learning techniques in NLP tasks. The dataset, despite some annotation variability, proved reliable for training, and the preprocessing steps effectively prepared the text for modeling. Future work could involve more extensive hyperparameter tuning and exploring other transformer-based models to further improve performance. Additionally, addressing the challenge of interpreting nuanced and sarcastic language remains an open area for improvement. This experiment highlights the importance of leveraging advanced models like BERT for complex NLP tasks, while also acknowledging the limitations of current approaches in handling ambiguous text.

References

Tahir Ali and Mohsin Raza. 2022. Stance classification on covid-19 vaccine tweets using machine learning and deep learning techniques. *IEEE Access*, 10:120735–120746.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (NAACL-HLT)*, pages 4171–4186.