CS-GY 6053: FOUNDATIONS OF DATA SCIENCE PROJECT PROPOSAL

A Machine Learning approach to Fake News Detection

Dhanesh Baalaji Srinivasan (**ds7636**)

Vaishnavi Chellappa (**vc2495**)

Venkataramanan Venkateswaran (**vv2265**)

What is the problem?

Detecting and analysing the accuracy of machine learning and deep learning models on fake news detection, and to increase the efficiency of classifying fake news.

Motivation

In today's age where information is mainstream and available for free, the spread of fake news has become a growing cause for concern in our society. This deliberate misinformation has put individuals and specific communities at risk. As Data Scientists, we are keen on applying the Data Science workflow to solve a ubiquitous problem.

Ideal Outcome

Short-term goal: Achieve significant accuracy on differentiating fake news from real news.

Long-term goal: Create an app that efficiently performs fake news detection and classification on the go, and shows users only real, authentic articles.

How will you learn the background?

While we have established a foundational perception of the fake news detection process, we propose to obtain more domain knowledge by:

1. Clearly articulating the goals and boundaries of the project by understanding the problem we intend to address and the insights we aim to uncover.
2. Gathering and thoroughly examining the dataset so that we can unveil subtle nuances and plausible data-related issues.
3. Exploring academic literature, publications, and previous research pertinent to the project's field and acquiring a comprehensive understanding of the existing knowledge

and methodologies in the domain. For instance, Shu et al. (2017) explored the dynamics of fake news on social media, utilizing data mining techniques to dissect real-time information dissemination. Further, Ma et al. (2015) introduced a novel approach using tree-structured recursive neural networks to detect rumors on Twitter, addressing the unique challenges of analyzing information propagation on this platform.

4. Using fact checking websites like PolitiFact and FactCheck.org to verify news veracity

What kinds of data will you use?

The LIAR dataset was introduced in a 2017 paper titled "Fake News: A Survey of Research, Detection Methods, and Opportunities" by William Yang Wang. It was designed to provide researchers with a substantial and diverse collection of labeled data to help advance the development of fake news detection models. The LIAR dataset has already been split into three classes - train, test and validation. More details of the number of data in each split is given in the table below.

Description of the dataset

Features	Description	Type
Column 1 (ID)	the ID of the statement in the format ([ID].json), used to uniquely identify each record.	Nominal (unique identifier).
Column 2 (Label)	the label which we would use to classify each news. It has 5 values (barely true, false, half true, mostly true, pants on fire)	Ordinal (categorical, with six levels from "True" to "Pants on Fire").
Column 3 (Statement)	the actual news which we would want to classify	Textual data (natural language text).
Column 4 (Subject)	the topic in which the news belongs to	Nominal (categorical, representing the topic).
Column 5 (Speaker)	the speaker involved in the news	Nominal (categorical, identifying the speaker).
Column 6 (Speaker's Job Title)	the speaker's job title.	Nominal (categorical, describing the speaker's job).
Column 7 (State)	The state at which the news was published	Nominal (categorical, U.S. state).
Column 8 (Party Affiliation)	the party affiliation either democrat or republican	Nominal (categorical, political party).
Column 9 (Barely True Counts)	The count of barely true class	Numerical (integer).
Column 10 (False Counts)	The count of false class	Numerical (integer).

Column 11 (Half True Counts)	The count of half true class	Numerical (integer).
Column 12 (Mostly True Counts)	The count of mostly true class	Numerical (integer).
Column 13 (Pants on Fire Counts)	The count of pants on fire class	Numerical (integer).
Column 14 (Context)	Textual context of the statement	Textual data (natural language text).

Dimensions

	Total	Train Data	Test Data	Validation Data
Size	≈12000	10269	1283	1284

Visualization of the data

Train data:

The below graph shows the count of each label across the input texts in train data. We can see that we have a fairly even spread of each outcome.

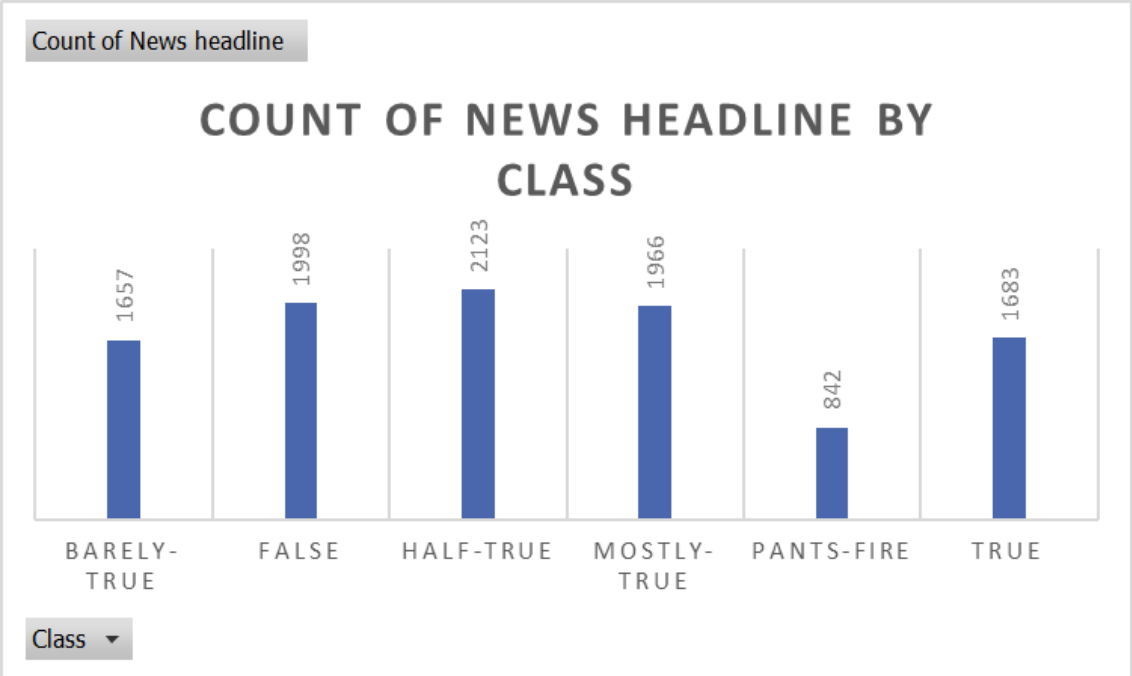


Fig 1: Count of data in each class in train set

Test and validation data:

The below graph shows the count of each label across the input texts in test and validation data. We can see that we have a fairly even spread of each outcome.

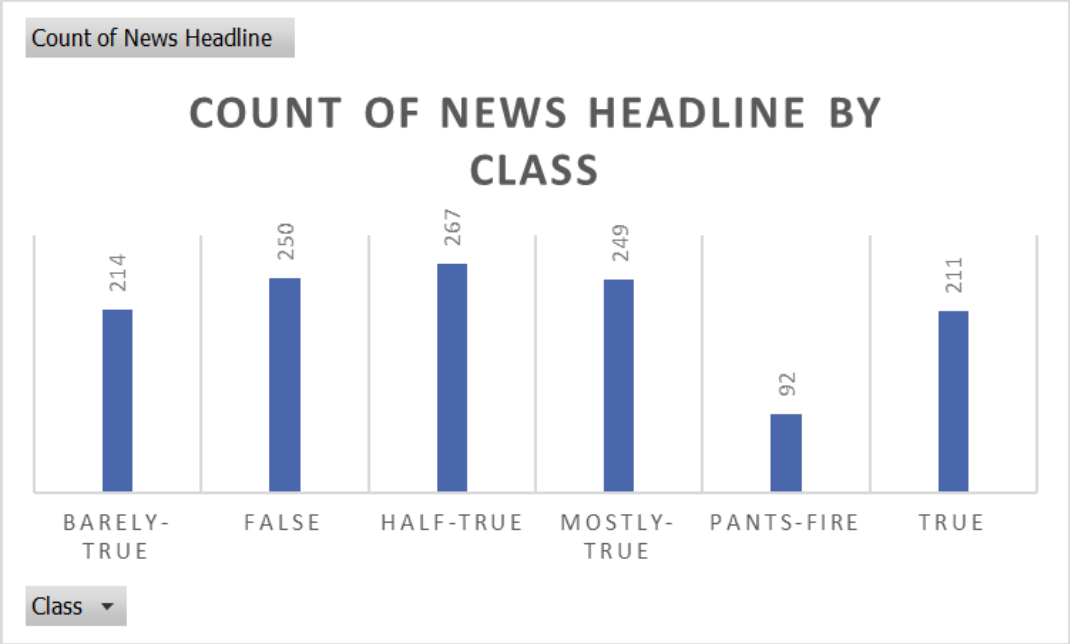


Fig 2: Count of data in each class in test set

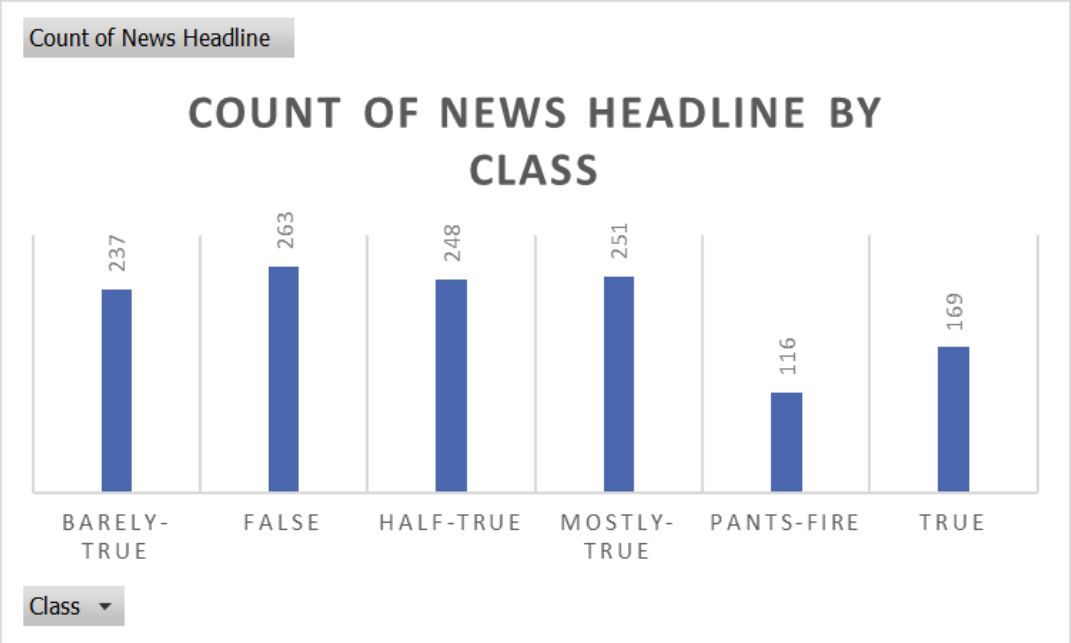
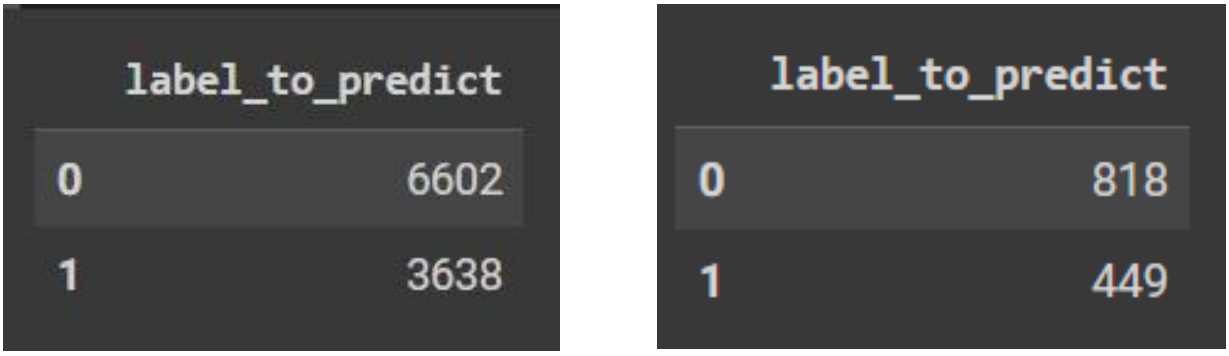


Fig 3: Count of data in each class in validation set

Featuring a well-balanced distribution of data across all possible outcomes, the LIAR dataset is an excellent choice for fake news detection, providing an ideal foundation for model development and evaluation in this domain.

Data Preprocessing

One caveat in the data distribution we observe is that we have very little data for 'pants-fire' class. In order to have an evenly distributed dataset, we simplified the labels into two categories: We set 'True' and 'mostly true' as 1 and the rest as 0. This way, we convert the ordinal to a categorical variable with two values "0" and "1" with which we build the classification models.



	label_to_predict
0	6602
1	3638

	label_to_predict
0	818
1	449

Fig 4: Distribution of labels in train and test datasets after preprocessing

How to process natural language into features for the model?

Natural Language Processing (NLP) models traditionally need numerical inputs to learn and classify the data. Word embeddings in Bidirectional Encoder Representations from Transformers (BERT) represent an advanced approach in NLP. Unlike traditional word embeddings like Word2Vec and GloVe, BERT generates embeddings that are contextually aware. This means that the representation of a word changes based on the context it appears in, allowing for a more nuanced understanding of language.

What kind of models will you build?

Machine Learning

We'll be using Support Vector Machines (SVM) for the text classification task because they are well-suited to work with textual data. They are capable of capturing non-linear relationships in the data, which can be valuable when dealing with fake news. SVMs are known for their efficiency in high-dimensional spaces, because when the data is projected onto a higher dimension, it might have a linear trend, and this can be easily captured using a SVM hyperplane, making them perfect for handling textual content and metadata.

Deep Learning

Deep Learning Models like BERT can be employed for fake news detection as they can be used to capture complex patterns in text data. BERT is adept at handling long-range dependencies in sequences. Data in LIAR dataset has considerable word ambiguity (i.e. many words have multiple meanings, and the correct interpretation often depends on the context) and semantic compositionality, (i.e. the meaning of the sentences is often not a simple linear combination of the meanings of its individual words). BERT's attention mechanism and its inherent resistance to the

vanishing gradient problem enable them to effectively capture and understand long-term dependencies present in the LIAR dataset.

We plan to employ these two models as it is crucial to understand relationships and dependencies that extend throughout the entirety of the news articles.

Metrics

We primarily focused on Precision and Recall. The reason for emphasizing precision as an evaluation metric is the critical nature of minimizing false positives in fake news detection. A high precision rate ensures that when our model labels news as true, it is highly likely to be correct, which is indispensable in sensitive domains.

We would also be using Precision as it gives the ability of a classification model to identify only the relevant data points. Precision is important as it is crucial for us to not classify real news as fake news and hence we would want our false positive rates to be low.

To get an overall picture of how the model is performing, we plan to use a confusion matrix, which summarizes the results of a classification problem, such as detecting fake news (positive class) and real news (negative class).

What assumptions are safe to make?

Features: The primary features used for fake news detection predominantly revolve around the textual content found in news articles or statements. The core feature typically under scrutiny is the "statement" itself.

We assume the data we get from the LIAR dataset is clean and reliable as the data is obtained from fact-checking websites like PolitiFact and GossipCop and the dataset is provided in a well-defined .tsv file. We assume that the fact-checking websites used in the LIAR dataset are credible and also assume that the dataset has a sufficient distribution of data to build a model that will generalize well to other news articles. This assumption is justified by the fact that the LIAR dataset spans a decade and contains 12.8k records, providing a substantial representation of news over that period. Consequently, we assume that there won't be any confounding variables.

Problem of overfitting

Within the LIAR dataset, there are a few variables that can influence the classification of whether a news is fake or real. For example, the "Speaker" and their "Job Title" can impact whether a statement is considered true or false because they affect the speaker's credibility and potential biases. Similarly, the "Subject" of a statement can serve as another variable that can create overfitting. Different subject matters may exhibit varying tendencies to contain fake news, and this factor should be accounted for in the analysis. We have to take into account the fact that the dataset may have a lot of fake news from a particular speaker or subject, and the model could overfit on that.

Observations and Results:

Our project employed two types of models: Support Vector Machines (SVM) and Bidirectional Encoder Representations from Transformers (BERT).

SVM (Baseline model) Performance:

Our SVM models, incorporating various combinations of metadata and text content, exhibited a range of accuracies. With full metadata utilization, the SVM model misclassified real news as fake 306 times, and fake news as real 203 times.

SVM (Baseline model) Classification Report:

	precision	recall	f1-score	support
0	0.667752	0.751834	0.707303	818.000000
1	0.413295	0.318486	0.359748	449.000000
accuracy	0.598264	0.598264	0.598264	0.598264
macro avg	0.540524	0.535160	0.533526	1267.000000
weighted avg	0.577578	0.598264	0.584136	1267.000000

Fig 5: Classification report of SVM - Baseline model

SVM (Baseline model) Confusion Matrix:

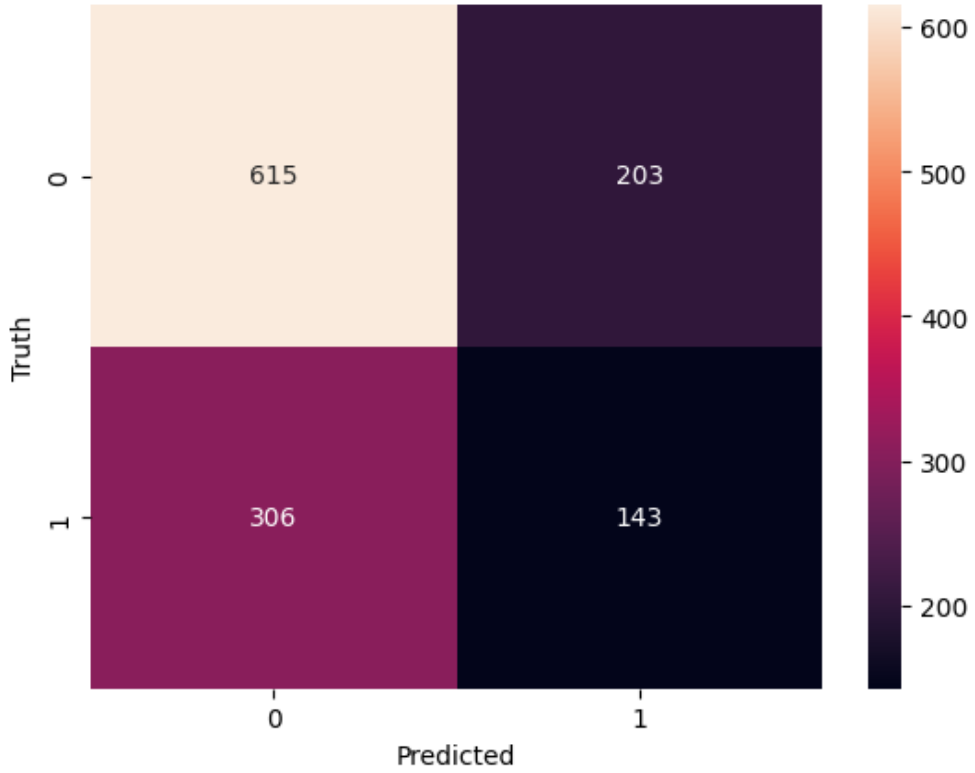


Fig 6: Confusion matrix for SVM - Baseline model

SVM (Improved) Performance:

The process of removing unwanted columns like 'subject', 'speaker', 'speaker_job_title', 'state_info', 'party_affiliation', and 'statement_context' from the train and test datasets is a pivotal step in refining the model's focus. By excluding these potentially extraneous or less influential factors, the approach zeroes in on the most pertinent features for effective model training and evaluation.

SVM (Improved) Classification Report:

	precision	recall	f1-score	support
0	0.646130	1.000000	0.785029	818.000000
1	1.000000	0.002227	0.004444	449.000000
accuracy	0.646409	0.646409	0.646409	0.646409
macro avg	0.823065	0.501114	0.394737	1267.000000
weighted avg	0.771534	0.646409	0.508405	1267.000000

Fig 7: Classification report of SVM - improved model

SVM (Improved) Confusion Matrix:

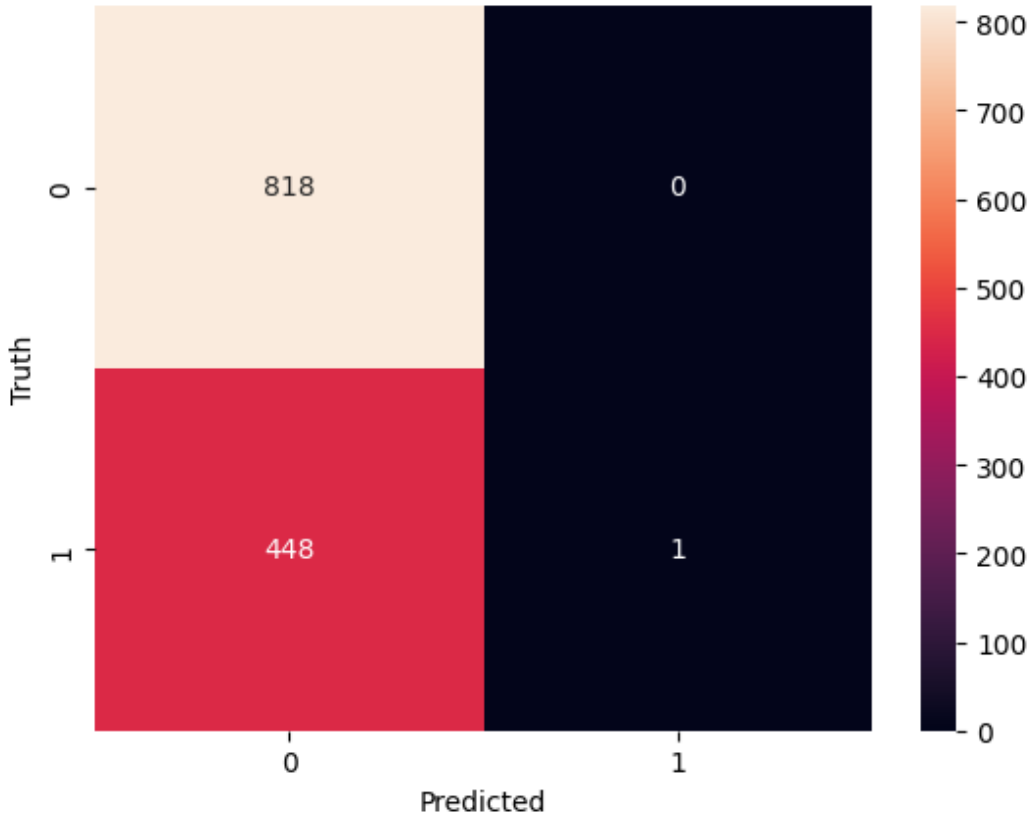


Fig 8: Confusion matrix for SVM - improved model

When metadata was excluded, the rate of misclassifying fake news as real news decreased, but the misclassification of real news as fake news increased to 448 instances. These variations highlight the impact of metadata on the SVM model's performance in distinguishing real from fake news.

BERT Performance:

The BERT model demonstrated a more balanced and effective performance in news classification. It correctly identified fake news 510 times, with a significantly lower misclassification rate for genuine news, which was incorrectly labeled as fake only 220 times. The results indicate BERT's superior capability in contextual understanding and language processing, making it a more reliable model for fake news detection compared to the traditional SVM approach.

BERT Classification Report:

	precision	recall	f1-score	support
0	0.698630	0.904255	0.788253	564.000000
1	0.587786	0.259259	0.359813	297.000000
accuracy	0.681765	0.681765	0.681765	0.681765
macro avg	0.643208	0.581757	0.574033	861.000000
weighted avg	0.660395	0.681765	0.640464	861.000000

Fig 10: Classification report for BERT model

BERT Confusion Matrix:

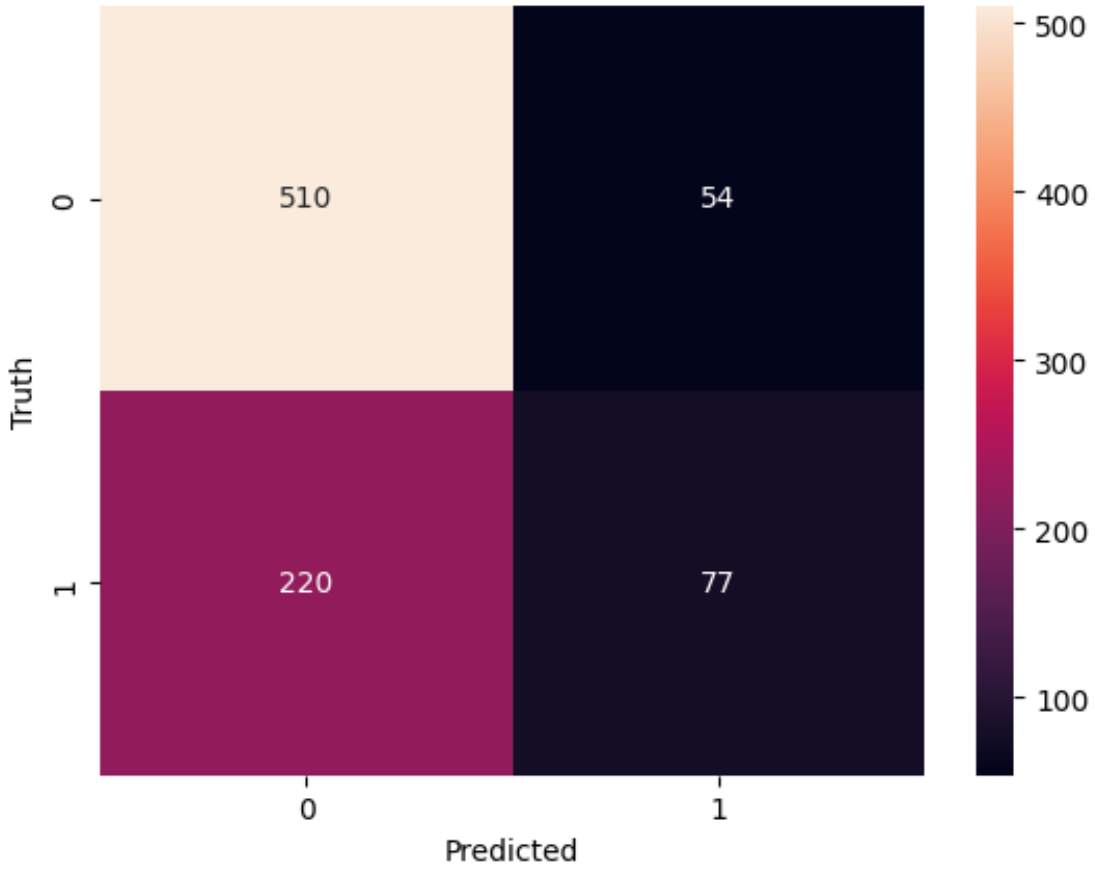


Fig 9: Confusion matrix for BERT model

In comparing the results of BERT and SVM models in fake news detection, distinct differences in performance and accuracy emerge. The SVM model, when leveraging full metadata, showed a higher tendency to misclassify real news as fake, and vice versa. This misclassification was particularly pronounced when metadata was excluded, indicating its influence on SVM's accuracy. On the other hand, BERT exhibited a more balanced performance with significantly lower misclassification rates, particularly in correctly identifying fake news. This suggests BERT's superior contextual understanding and language processing abilities, making it a more effective model for accurately detecting fake news compared to the traditional SVM approach.

Future scopes and next steps

Ensemble Methods: Exploring ensemble techniques by combining predictions from multiple models, such as SVM and BERT, can potentially improve overall performance.

Feature Engineering: Investigating additional features or representations that are derived from the text data that could enhance model performance, such as linguistic features, sentiment analysis, or topic modeling.

Hyperparameter Tuning: Optimizing hyperparameters for both SVM and BERT models to achieve better performance. We can utilize grid search, random search, or Bayesian optimization methods to fine-tune parameters effectively.

Deployment and Real-time Application: Focusing on deploying the model into real-world scenarios, integrating it into applications or systems for real-time fake news detection.

Conclusion

The comparative analysis of SVM and BERT models in our fake news detection project underscores the advanced capabilities of deep learning, especially BERT, in understanding and processing complex language patterns. While SVM shows varying levels of precision influenced by metadata, BERT's contextual awareness provides a more nuanced and accurate classification. This study contributes significantly to the field of fake news detection, suggesting a promising direction for future research and practical applications in media and information verification.

References

1. **Demystifying BERT: A Comprehensive Guide to the Groundbreaking NLP Framework**
2. **LIAR - Dataset**
3. **SVM - Introduction to Machine learning algorithms**

Team points

Name	Net ID	N-Number	Points
Vaishnavi Chellappa	vc2495	N19223004	4
Venkataramanan Venkateswaran	vv2265	N16495641	4
Dhanesh Baalaji Srinivasan	ds7636	N11671233	4