CS445 Milestone Report

Group No: 32

Alper Cimşit 28923
Kağan Kağanoğlu 29482
Mehmet Berke Bakan 28940

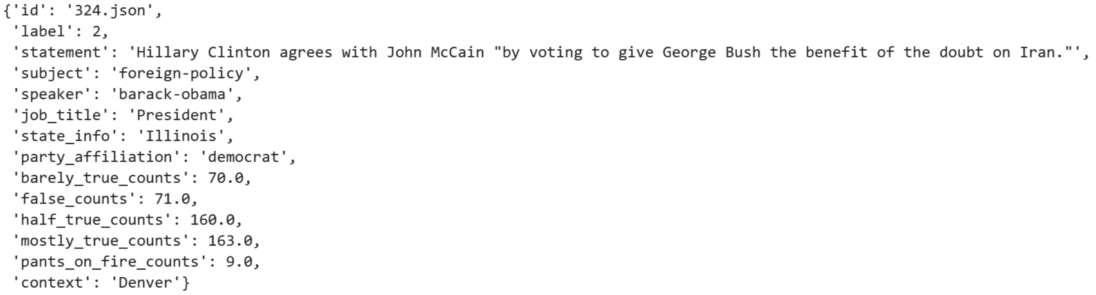
1. Introduction

The increase in misinformation can be seen in the nearly 10,000% rise in the number of academic articles including the term "fake news" between 2015 and 2018 (Figueira, Álvaro, & Guimarães, 2019). Due to the increasing number of fake news on the Internet, fake news detection has become an important task in the field of NLP. In the era of postmodernism where the truth becomes ambiguous, fake news detection can contribute to differentiate fake news from true ones. The purpose of our task is to develop a model that determines the “truth rates” of given information using machine learning algorithms, which can help to identify misinformation.

So far, we have investigated various research papers on fake news detection, so that we would have a broader understanding of the existing approaches. After scanning papers, we decided on a paper with a comprehensive dataset and various implementations. After examining the characteristics of the dataset too, we discussed our methodology and task division between team members regarding our next steps.

2. Dataset Selection

We are using the LIAR dataset, introduced by Wang (2017) in the paper “Liar, Liar Pants on Fire: A New Benchmark Dataset for Fake News Detection.” The dataset is composed of political statements collected from PolitiFact.com. We selected this dataset because it is widely used in the literature, providing a strong benchmark for fake news detection. Many papers have built different approaches on it, such as the BiDirectional Long Short-Term Memory (BiLSTM) approach (Alhindi, Petridis, & Muresan, 2018). The dataset includes six target categories for truthfulness, with short statements, making it a challenging task for classification.



```
{'id': '324.json',
 'label': 2,
 'statement': 'Hillary Clinton agrees with John McCain "by voting to give George Bush the benefit of the doubt on Iran."',
 'subject': 'foreign-policy',
 'speaker': 'barack-obama',
 'job_title': 'President',
 'state_info': 'Illinois',
 'party_affiliation': 'democrat',
 'barely_true_counts': 70.0,
 'false_counts': 71.0,
 'half_true_counts': 160.0,
 'mostly_true_counts': 163.0,
 'pants_on_fire_counts': 9.0,
 'context': 'Denver'}
```

Figure 1: Example Data Point from LIAR dataset

The image displays a sample data point from the LIAR dataset. For our task, the relevant fields include “label,” “statement,” and “speaker.” The “label” is the target categorical column indicating the truthfulness of the statement, with values ranging from 0 to 5, where 0 represents a blatant lie and 5 represents the full truth. “statement” and “speaker” are the input fields we will use to train our model.

The LIAR dataset is pre-split into three parts: training (10.269 examples), validation (1.284 examples), and test (1.283 examples). We will maintain this split to ensure the proper benchmarking and compatibility with other studies.

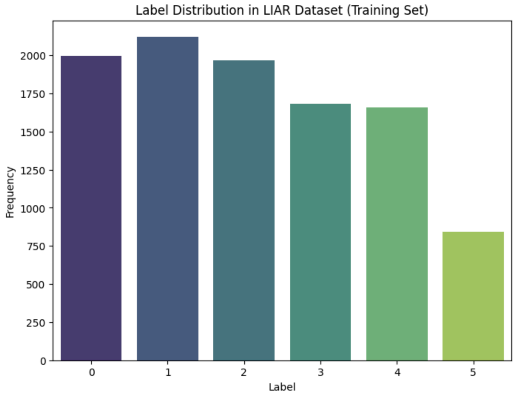


Figure 2: Label Distribution of Train Set

The bar graph illustrates the distribution of the labels in the training data, where the labels range from 0 (least truthful) to 5 (most truthful). From the graph, we can observe that the distribution is fairly balanced across most categories, except for the most truthful label (5), which is relatively underrepresented.

3. Approach Plan

In this project, after doing a review of literature, we decided to implement the hybrid Convolutional Neural Network model proposed by Wang (2017) in his paper "Liar, Liar Pants on Fire." Which is the paper that the dataset is also proposed. This approach has achieved the highest reported accuracy on this particular dataset, with a test accuracy of 27.4% in this six-class classification task. The model effectively combines textual features extracted from the statements with metadata of the speaker and the context.

This hybrid CNN model operates by merging two streams of data: the statement text and metadata fields. For the text, the model employs a CNN to discern patterns, including linguistic cues and phrasing. But metadata is processed independently; These outputs are subsequently concatenated and transmitted through a fully connected layer to yield the final prediction. However, this approach is complex and requires careful implementation. Although the integration of these data streams enhances accuracy, it also presents challenges.

To achieve this, we will separately process the text data and metadata. The text data will be represented using pre-trained word embeddings (like Word2Vec), which capture the semantic meaning of words. These embeddings will be input into a CNN layer, which extracts features like n-grams; this will be followed by max-pooling to reduce dimensionality. However, for the metadata, we will encode features into numerical vectors and utilize a distinct layer to process them. Once both streams of data are processed, they will be combined into a single feature representation. This representation will be passed to a dense layer for classification.

Although we will experiment with different hyperparameter settings (like filter sizes, dropout rates), the aim is to find the best-performing configuration. Finally, we will evaluate the model on the dev set and compare its performance to simpler baseline models (such as logistic regression or SVMs),to enhance this baseline, we also plan to evaluate our model against advanced architectures like BiLSTMs. Drawing inspiration from Qiao et al. (2020)

Although we finalized our most part of the project approach, we will be open for enhancements and changes as we review new papers and researches.

4. Next Steps

For the next steps, both textual and metadata processing will go through their respective procedures. The text will be represented using pre-trained embeddings, then will pass through a CNN which, in turn, goes through max-pooling to draw the vital characteristics. At the same time, the metadata will be encoded as vectors and will be processed in different layers to get the necessary information out. Next, the outcome of both streams is merged into a dense layer that actually does the classification. We will try different possibilities for each hyperparameter such as filter sizes and dropout rates, to see if we will be able to increase the performance. During the testing, the model will be compared against the baseline methods such as regression and SVMs. Thereafter, more complex models such as BiLSTMs will be examined. We will continue our research and use existing studies to improve the model. Finally, we will report our findings.

The project will be managed and executed collaboratively by all team members at every stage, with each member focusing on specific tasks. Berke will concentrate on text preprocessing and analysis, Kağan will focus on model implementation, and Alper will handle hyperparameter tuning and performance evaluation. Since we will be in contact working on our tasks, the task division is open to changes.

5. References

Alhindi, T., Petridis, S., & Muresan, S. (2018). Where is your evidence: Improving fact-checking by justification modeling. *Proceedings of the First Workshop on Fact Extraction and VERification (FEVER)*. https://doi.org/10.18653/v1/w18-5513

Figueira, Álvaro & Guimarães, Nuno & Torgo, Luís. (2019). A Brief Overview on the Strategies to Fight Back the Spread of False Information. Journal of Web Engineering (JWE). 18. 319-352. 10.13052/jwe1540-9589.18463.

PolitiFact. (n.d.). https://www.politifact.com/

Wang, W. Y. (2017). “Liar, Liar Pants On Fire”: A new benchmark dataset for fake news detection. *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*. https://doi.org/10.18653/v1/p17-2067

Yu Qiao, Daniel Wiechmann, and Elma Kerz. 2020. A Language-Based Approach to Fake News Detection Through Interpretable Features and BRNN. In *Proceedings of the 3rd International Workshop on Rumours and Deception in Social Media (RDSM)*, pages 14–31, Barcelona, Spain (Online). Association for Computational Linguistics.