4AL3 Progress Report:
Fake News Detection
Group 68

Chiyu Huang, Jingyao Sun, Guanqi Wang
{huanc10, sun250, wangg97}@mcmaster.ca

1 Introduction

Fake news have become a significant threat nowadays, as misinformation spreads quickly across online platforms. It can mislead the public, distort health behaviors, and create social unrest. Children and teenagers, who are increasingly exposed to online content at an early age, are particularly vulnerable to these effects due to their limited critical thinking skills. The goal of this project is to build a machine learning classifier that can automatically distinguish between real and fake news articles. The challenge is that fake news articles often mimic real journalism, using ambiguous, exaggerated, or misleading language. Detecting them requires models to capture subtle linguistic and semantic cues beyond surface-level text classification, making the task substantially more complex than traditional text categorization.

2 Dataset

We use the LIAR dataset for fact-checking and veracity classification. The dataset contains 12,836 short statements collected from PolitiFact.com, each labeled into one of six truthfulness categories: *pants-on-fire*, *false*, *barely-true*, *half-true*, *mostly-true*, and *true*.

Each record includes both the statement text and metadata such as speaker name, party affiliation, state, subject/topic, and context of the claim.

The dataset is pre-divided into train (10,240 samples), validation (1,284), and test (1,267) splits. Figure 1 shows the label distribution, where *half-true* and *false* are the most common categories.

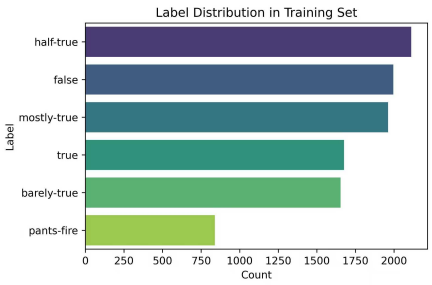

Figure 1: Label distribution of the LIAR dataset.

Before model training, we performed several cleaning and normalization steps on the raw dataset using Python libraries such as `pandas`, `re`, and `matplotlib`:

Firstly, we removed missing or duplicated samples based on statement, speaker, and context. Secondly, we normalized text by converting to lowercase and removing non-alphanumeric characters (except %, ?, !, and $). Thirdly, we also filtered out extremely short statements (<3 tokens) to reduce noise. Lastly, we generated exploratory visualizations to better understand data characteristics, including the label distribution and statement length distribution.

Figure 2 below shows the distribution of statement lengths, confirming that most statements contain between 5 and 25 tokens, which supports treating LIAR as a short-text classification task.

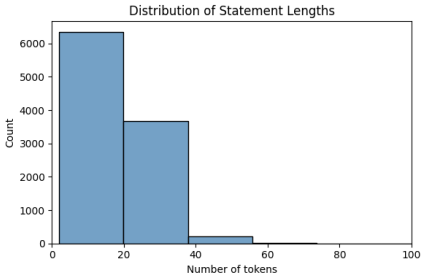

Figure 2: Statement length distribution in tokens.

3 Related Work

The benchmark dataset LIAR, introduced by (Wang, 2017), established a fine-grained six-class veracity classification task and laid the foundation for subsequent research on automated fake news detection. This paper directly inspired our project design. From this work, we learned how veracity can be quantitatively measured through fact-checking categories and how metadata (such as speaker identity or party affiliation) can enrich classification beyond text alone. We plan to follow a similar structure and replicate this benchmark to evaluate model accuracy across multiple truthfulness levels.

Early studies relied on traditional machine-learning approaches such as logistic regression and support vector machines (SVM) with TF-IDF and metadata features to create strong yet interpretable baselines (Patel et al., 2021). As deep language models became dominant, transformer-based architectures (e.g., BERT, RoBERTa, and DistilBERT) demonstrated substantial performance improvements on news and misinformation datasets(Kaliyar et al., 2021; Raza, 2021; Li and Zhou, 2020). Recent works further explored hybrid and ensemble frameworks that combine classical ML and transformer models to enhance robustness and generalization across domains(Kaliyar et al., 2021).

4 Features

For the baseline model, we represent each statement using TF-IDF unigram features, which capture how important each word is across the dataset while keeping the model simple and interpretable. At this stage, we only use textual inputs, though metadata such as party affiliation and state are available for possible later integration. The resulting TF-IDF matrix serves as input to our Logistic Regression baseline, and the same representation will be reused when we add SVM in the next phase for comparison.

5 Model Implementation

We implemented two models(Majority-class baseline model and TF–IDF & Logistic Regression model) as our primary baseline for fake news classification.

1. Majority-class baseline. A simple classifier that always predicts the most frequent label (*half-true*) achieves about 21% accuracy and 0.06 macro-F1.

2. TF–IDF & Logistic Regression baseline. Each statement is transformed with TF–IDF (unigrams, min_df=2, max_df=0.95). We use multiclass Logistic Regression with `class_weight="balanced"` and the `liblinear` solver.

Our model uses logistic loss, which is the classical objective function optimized by Logistic Regression. An L2 regularization term is added (controlled by the parameter C=1.0) to prevent overfitting on the sparse TF-IDF features.

The model minimizes the following regularized logistic loss:

$$\text{Loss} = -\sum y \log(\hat{p}) + \lambda \|w\|_2^2 \qquad (1)$$

where $\lambda = 1/C$. This matches the default behavior of the liblinear solver used in our code. The model is optimized using the liblinear solver. It uses a coordinate descent method to minimize the convex logistic loss with L2 regularization. It iteratively updates one weight at a time while holding the others fixed, ensuring convergence to the global minimum. This solver is stable and efficient for sparse text data, which matches the structure of our TF-IDF representation.

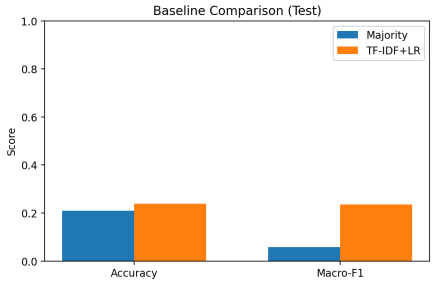

Figure 3: Statement length distribution.

6 Results and Evaluation

We evaluate using the LIAR train/validation/test splits. The model is trained on training data, validated on the dev set, and evaluated on the test set. We do not use cross-validation to keep a consistent baseline.

For evaluation, we focus on two key metrics — Accuracy and Macro-F1 — to measure overall and balanced performance across the six labels. We also include a classification report and a confusion matrix visualization to show per-class precision,

recall, and F1-scores and to analyze which labels are most frequently confused.

Model	Dataset	Accuracy	Macro-F1
Majority (half-true)	Test	0.209	0.058
TF-IDF + Logistic Regression	Validation	0.232	0.232
	Test	0.238	0.236

Table 1: Baseline model performance on validation and test sets.

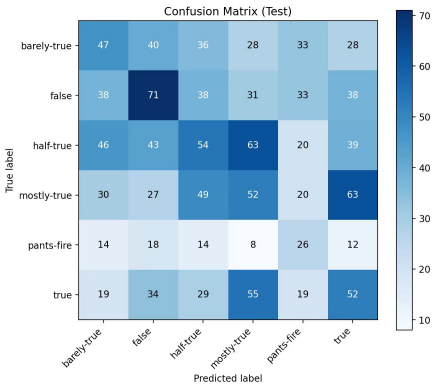

Figure 4: Statement length distribution.

7 Feedback & Plans

During our meeting with the Monday TA, we received positive feedback on our project's progress. The TA noted that our preprocessing and baseline implementation were on the right track and that our dataset handling appeared robust. The main suggestion was to add more visualizations to better illustrate both the data characteristics and model results.

Based on this feedback, we have added several visualizations: for the data, we now include charts of label distribution and statement length; for the model, we added a confusion matrix and a baseline performance comparison. In the next stage, we plan to extend these visualizations with training/validation curves and feature-importance plots for the logistic regression baseline.

In the next stage, we plan to add an SVM as an additional classical baseline and fine-tune a DistilBERT model to capture deeper contextual semantics. We will then explore a simple ensemble that combines SVM and DistilBERT outputs to compare traditional and transformer-based approaches. These extensions will allow us to systematically measure performance gains while maintaining a clear, minimal baseline for fair comparison.

Team Contributions

- **Guanqi Wang** was responsible for data cleaning and preprocessing, including collecting data and implementing the data normalization pipeline. He also wrote the Dataset, Introduction, and Related Work sections of the report.
- **Jingyao Sun** was responsible for model training and implementation, including developing the baseline models and configuring the experimental setup. She also wrote the Features and Model Implementation sections of the report.
- **Chiyu Huang** was responsible for evaluation, visualization, and integration. He verified that all scripts and models ran correctly, created the figures and tables, and wrote the Evaluation and Results and Feedback and Plans sections. He also assembled the final LaTeX report, ensuring consistent formatting and presentation.

References

R. K. Kaliyar, A. Goswami, and R. Narayan. 2021. A transformer-based architecture for fake news classification. *Social Network Analysis and Mining*, 11(1):59.

Q. Li and D. Zhou. 2020. Connecting the dots between fact verification and fake news detection. In *Proceedings of the 28th International Conference on Computational Linguistics (COLING 2020)*, pages 1889–1900. Association for Computational Linguistics.

J. Patel, A. Tiwari, and T. Ahmad. 2021. Fake news detection using support vector machine. In *Proceedings of the 10th International Conference on Pattern Recognition Applications and Methods (ICPRAM 2021)*. SCITEPRESS.

H. Raza. 2021. Automatic fake news detection in political platforms – a transformer-based approach. In *Proceedings of the Fourth Workshop on Challenges and Applications of Automated Extraction of Socio-political Events (CASE @ ACL 2021)*, pages 92–97. Association for Computational Linguistics.

William Y. Wang. 2017. "liar, liar pants on fire": A new benchmark dataset for fake news detection. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Short Papers)*, pages 422–426. Association for Computational Linguistics.