Exploring the Impact of Deep Learning Models on Fake News Classification and the Introduction of Unseen Domain Data
CSE-256 Project Report

Atharv Sunil Biradar
abiradar@ucsd.edu

1 Introduction

In today's digital age, the rapid spread of misinformation has become a significant challenge. Fake news, designed to mislead readers, can influence public opinion, cause societal instability, and erode trust in information sources. Detecting fake news is crucial for maintaining the integrity of online platforms, but the task is complex due to the subtle language patterns and context often used in fake news articles.

This project explores various techniques for detecting fake news using Natural Language Processing (NLP). The objective is to analyze the content of news articles and build a system capable of identifying fake news accurately. By leveraging machine learning models and advanced NLP methods, the project aims to contribute to combating misinformation effectively.

1.1 Goals and Status

- **Collected and preprocessed dataset:** DONE
- **Built and trained a baseline model on the dataset:** DONE
- **Explored advanced NLP techniques to improve detection accuracy:** DONE
- **Evaluated and compared model performance:** DONE

2 Related Work

Fake news detection has gained significant attention in recent years, given the rise in misinformation and its potential societal impacts. Fake news can be defined as fabricated information that mimics the format of legitimate news media content but lacks the same organizational processes or intent (Shu et al., 2017). Numerous approaches have been proposed to address this issue.

Shu et al. (Shu et al., 2017) explored a variety of methods for fake news classification, including user-based, knowledge-based, social network-based, and style-based approaches, providing a comprehensive analysis of this multifaceted problem. Julio et al. (Julio and Edgar, 2018) introduced a new set of features and evaluated the performance of existing methods, emphasizing the critical role of feature selection in improving classification accuracy.

Social media platforms have also been a focus of fake news research. Daniel et al. (Castillo et al., 2011) conducted an analysis of information credibility on Twitter, providing insights into the role of social media in the spread of misinformation. Heejung et al. (Heejung et al., 2020) applied the Bidirectional Encoder Representations from Transformers (BERT) model to detect fake news by analyzing relationships between news headlines and their body text.

The use of specific datasets has been crucial for advancements in this field. Mohammad Hadi et al. (Hadi and Ghasemzadeh, 2020) utilized n-gram features extracted from the ISOT fake news dataset to enhance detection performance. Similarly, Saqib et al. (Saqib and Raza, 2020) proposed an ensemble classification model, achieving state-of-the-art accuracy on the ISOT dataset. Sebastian et al. (Sebastian and Hadi, 2020) employed a neural network-based approach for text analysis and fake news detection, further validating the usefulness of the ISOT dataset.

These studies highlight the diversity of techniques and datasets used for fake news detection, reflecting the complexity and importance of addressing this issue effectively.

3 Dataset

For this project, I used the ISOT Fake News Dataset (ISOT, 2020), which includes both real and fake news articles. The real news comes from Reuters, while the fake news was collected from unreliable sources flagged by Politifact and Wikipedia. Most articles are about politics and world events, with data collected from 2016 to 2017.

3.1 Dataset Statistics

The dataset has two main files:

- **True.csv**: Over 12,600 real news articles from Reuters.
- **Fake.csv**: Over 12,600 fake news articles from various sources.

Each article includes the title, text, type (real or fake), and publication date. Some fake news articles may contain punctuation mistakes, which were kept to reflect real-world errors.

Here's a summary of the dataset:

News Type	Number of Articles
Real-News	21,417
World-News	10,145
Politics-News	11,272
Fake-News	23,481
Government-News	1,570
Middle-East	778
US News	783
Left-News	4,459
Politics	6,841
Other News	9,050

Table 1: Category Breakdown of the ISOT Dataset

3.2 Task and Challenges

The task is to classify news articles as either real or fake. This is challenging because both real and fake news can cover similar topics, and fake news often contains subtle errors or misleading information. Distinguishing between biased, real news and fabricated content is difficult.

Here are some examples from the dataset:

- **Input:** ”President Trump talks about tax reform.”
- **Output:** `Real News`
- **Input:** ”NASA discovers aliens on Mars!”
- **Output:** `Fake News`

The goal is to train a model that can handle these challenges and accurately classify the articles.

3.3 Data Preprocessing

The goal of this step is to prepare the data for model training by using NLP techniques to clean and transform the input data.

The dataset consists of news titles and texts. Each title has about 12 words, and each text has about 405 words. Only the titles for fake news detection are used, as the texts are too long and detailed for efficient training and may distract the model.

To preprocess the data, I built a pipeline with the following steps:

- Replace characters that are not between `a-z` or `A-Z` with whitespace.
- Convert all characters to lowercase.
- Remove inflectional morphemes like “ed”, “est”, “s”, and “ing” (e.g., ”confirmed” → ”confirm”).
- Crop titles into sentences with a maximum length of 42 words to balance the dataset and avoid extreme-length titles.

3.3.1 Word Embedding

For the models, I needed to convert the text into a numerical format that the models can process.

For LSTM, Bidirectional LSTM, and CNN models:

- I use a Tokenizer to break the text into words and create sequences of tokenized words.
- Each sequence is zero-padded to a length of 42.
- An Embedding layer is used to convert the sequences into dense vector representations that the models can learn from.

For BERT:

- I use BERT's tokenizer to split the text into smaller subwords if needed (e.g., ”embedding” → [’em’, ’bed’, ’ding’]).

- Two special tokens are added: [CLS] at the start of the sentence and [SEP] to separate sentences or mark the end.

- The input is converted into three types of tensors: token tensors (indices of words), segment tensors (sentence identifiers), and mask tensors (token concentration after zero-padding).

4 Baseline

For the baseline model, I used a Long Short-Term Memory (LSTM) network to perform the fake news classification task. The model architecture is as follows:

- The model uses an embedding layer with 40-dimensional word vectors.

- A dropout layer with a rate of 0.3 is applied to prevent overfitting.

- The LSTM layer contains 100 units to capture sequential dependencies in the text.

- Another dropout layer with a rate of 0.3 is added after the LSTM layer.

- A dense output layer with a sigmoid activation function is used for binary classification.

The model is compiled with binary cross-entropy loss and the Adam optimizer, with accuracy as the evaluation metric.

The dataset was split into training (60%), validation (20%), and test (20%) sets, with shuffling. The original dataset consists of 44,898 articles. I performed two experiments: one with a balanced dataset and one with an imbalanced dataset. The imbalanced dataset was created by reducing the number of fake news articles to 10% of the original size to simulate real-world conditions where fake news is less frequent.

Dataset Type	Training Set	Validation Set	Test Set
Balanced Dataset	26,939	8,980	8,979
Imbalanced Dataset	14,260	4,753	4,754

Table 2: Dataset Split for Balanced and Imbalanced Data

I assumed a more realistic scenario in the imbalanced dataset, where fake news is less common. This allowed to compare the performance of the model under both conditions.

5 Experiments and Results - Approach

5.1 Conceptual Approach

The goal is to detect fake news based on news titles using different deep learning models. I employed four models: LSTM, Bidirectional LSTM (BiLSTM), CNN-BiLSTM, and BERT.

The LSTM model serves as the baseline. It works by processing the input text sequentially, capturing the context from past words, and making predictions based on the learned sequence. This approach helps in identifying fake news but has some limitations in capturing complex context.

I then explore Bidirectional LSTM (BiLSTM), which enhances the LSTM by processing the text in both forward and backward directions. This allows the model to consider the entire context of a sentence, improving the understanding of word dependencies in both directions.

Next, I introduce CNN-BiLSTM, combining Convolutional Neural Networks (CNN) with BiLSTM. The CNN layers help in extracting important features from the text, while BiLSTM ensures that the model can make predictions based on the complete context, both forward and backward.

Finally, I use BERT, a transformer-based model that revolutionized natural language processing. BERT is pre-trained on a massive corpus of data and uses attention mechanisms to understand the context of words from both directions. By fine-tuning BERT for my task, I aim to leverage its powerful understanding of language while reducing the training time compared to training from scratch.

5.2 Working Implementation

I successfully implemented all the models in my project. The files related to these models are available in the GitHub project repository.[1]

5.3 Compute

All experiments were run on Kaggle and Google Colab. These platforms provided access to powerful GPUs, including the Tesla P100 and

[1] https://github.com/atharv2802/CSE256_Project_Fake_News_detection

Tesla T4, which significantly accelerated the training process. For the BERT model, I used PyTorch and HuggingFace libraries, as they are optimized for transformer models like BERT. In Google Colab, I had to adjust the batch size to avoid running into memory issues during BERT fine-tuning.

5.4 Runtime

Training times varied depending on the model:

- LSTM: Approximately 20 minutes.
- Bidirectional LSTM (BiLSTM): Approximately 25 minutes.
- CNN-BiLSTM: Approximately 30 minutes.
- BERT: Fine-tuning took around 1.5 hours due to the size of the pre-trained model and dataset.

5.5 Domain-Specific Data Augmentation - Technique Exploration

To improve my model's ability to classify news more accurately, I decided to try something new. I used GPT to generate 3,500 additional samples of sports news, with 1,500 fake news articles and 2,000 real ones. These samples were added to the original ISOT dataset to see if introducing a **new domain (sports news)** would help the model perform better.

The idea behind this approach was simple: by exposing the model to news from a different domain, I hoped it would learn to classify articles more effectively, even if they come from various topics. Sports news, being quite different from the original dataset, introduces new terms and patterns, giving the model more variety to learn from.

After adding these new sports news samples, I retrained all of my models—LSTM, BiLSTM, CNN-BiLSTM, and BERT—on the combined dataset. My main goal was to see if including this new domain knowledge would help the models become better at distinguishing between real and fake news, especially when dealing with articles from different topics.

In this experiment, I wanted to test if the models would perform better with a broader range of data, as opposed to training them only on the original ISOT dataset. I was curious to find out if expanding the type of news the models saw would help them generalize better and improve their overall accuracy.

5.6 Results

The table 3 shows how the different models perform across the various datasets. Starting with the baseline LSTM model, it does a good job across all the datasets. It performs well on both the balanced and unbalanced datasets, and slightly improves when I add the sports data in the augmented dataset. This shows that the LSTM can handle extra data fairly well.

The Bidirectional LSTM and CNN-BiLSTM models perform a bit better than LSTM. The BiLSTM is consistently strong, while the CNN-BiLSTM shows a slight improvement, especially on the unbalanced dataset. This might mean that the CNN layers help capture some additional patterns when the data is not evenly distributed.

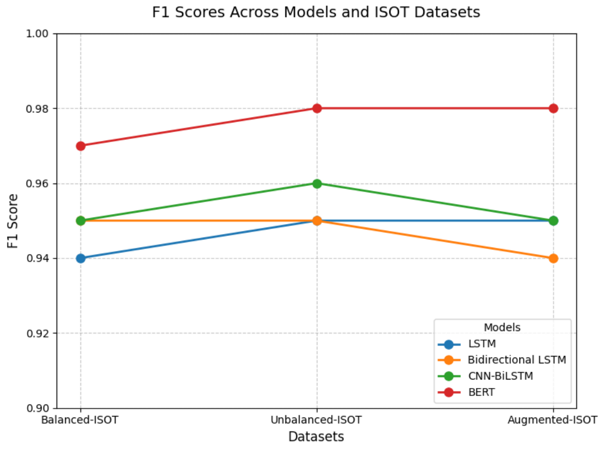

Figure 1: Model Comparison

Finally, BERT stands out as the top performer across all datasets, with the highest F1 scores. This shows how powerful BERT is for this type of text classification task, especially when I add the new sports-related data. In summary, as figure 1 demonstrates all models show some improvement with the augmented dataset, BERT stands out as the best model.

6 Error analysis

The baseline model struggled with headlines that used sensational language or exaggerated claims. Phrases like ”victory lap” or ”BIG NAME” were especially tricky because they hid subtle biases behind attention-grabbing words. These challenges were both about the way the sentences were structured—like using emotionally charged phrases—and the meaning behind them, such as biased or misleading framing. Even

Model	Balanced - ISOT	Unbalanced - ISOT	Augmented - ISOT (With sports samples)
LSTM	0.94	0.95	0.95
Bidirectional LSTM	0.95	0.95	0.94
CNN-BiLSTM	0.95	0.96	0.95
BERT	0.97	0.98	0.98

Table 3: F1 Scores for Various Models on ISOT Datasets

though the augmented approach helped improve the performance, it still struggled with language that was more subtle or manipulative. This shows that there's still room to improve and refine the model.

Various examples of misclassification are as follows:

- Sentence: "Obama does victory lap for car industry but it is different from what he hoped."
  - **Input sentence type:** Ambiguous or sensational language
  - **True Label:** 1 (true news)
  - **Prediction:** 0 (fake news)
  - **Reason:** Sensational phrasing such as "victory lap" introduces ambiguity, making it difficult for the model to distinguish between factual and exaggerated content.
- Sentence: "Tom Brady announces retirement but hints at possible return."
  - **Input sentence type:** Ambiguous or sensational language
  - **True Label:** 0 (fake news)
  - **Prediction:** 1 (true news)
  - **Reason:** The sensational phrasing "hints at possible return" made the model classify it as true news, even though the headline was speculative and lacked concrete information.
- Sentence: "IBM ANNOUNCES 2,000 Jobs For Vets After Meeting With President Trump."
  - **Input sentence type:** Plausible yet misleading claims
  - **True Label:** 0 (fake news)
  - **Prediction:** 1 (true news)
  - **Reason:** The headline's plausible yet exaggerated claim misled the model, which failed to account for the lack of context or scale.
- Sentence: "LeBron James promises Lakers' fans a championship in 2024."
  - **Input sentence type:** Plausible yet misleading claims
  - **True Label:** 0 (fake news)
  - **Prediction:** 1 (true news)
  - **Reason:** The claim appeared credible, but the model failed to recognize the exaggeration in the word "promises," which implied certainty without evidence.
- Sentence: "Bilderberg to Meet Next Week in Chantilly, Virginia."
  - **Input sentence type:** Clickbait and distorted events
  - **True Label:** 0 (fake news)
  - **Prediction:** 1 (true news)
  - **Reason:** While referencing a real location, the sensational framing confused the model, making it difficult to classify accurately.
- Sentence: "Cristiano Ronaldo to join MLS team for record-breaking salary."
  - **Input sentence type:** Clickbait and distorted events
  - **True Label:** 0 (fake news)
  - **Prediction:** 1 (true news)
  - **Reason:** The model was misled by the plausible yet exaggerated claim about a "record-breaking salary," failing to recognize it as clickbait with distorted details.

- Sentence: “Democratic Rep. Introduces Bill That Would Drug Test Rich Welfare Recipients.”
  - **Input sentence type:** Overgeneralized or biased claims
  - **True Label:** 0 (fake news)
  - **Prediction:** 1 (true news)
  - **Reason:** The model struggled to recognize political bias and exaggerated framing, treating the headline as factual.
- Sentence: “College athlete breaks multiple records in stunning performance.”
  - **Input sentence type:** Overgeneralized or biased claims
  - **True Label:** 1 (true news)
  - **Prediction:** 0 (fake news)
  - **Reason:** The model dismissed the headline due to its generality, failing to recognize the underlying factual content about the athlete’s performance.
- Sentence: “SENATOR WANTS ANSWERS ON MISSING TSA BADGES AFTER TSA TRIES TO BLOCK INVESTIGATION.”
  - **Input sentence type:** Manipulative political headlines
  - **True Label:** 0 (fake news)
  - **Prediction:** 1 (true news)
  - **Reason:** The sensational tone and framing made the headline appear more scandalous than factual, confusing the model.
- Sentence: “FIFA president under fire for handling of corruption allegations.”
  - **Input sentence type:** Manipulative political headlines
  - **True Label:** 0 (fake news)
  - **Prediction:** 1 (true news)
  - **Reason:** The dramatic tone of “under fire” and “corruption allegations” led the model to misclassify the headline as true news, despite its speculative nature.

7 Conclusion

In conclusion, this project taught me a lot about how deep learning models can be used to classify news headlines as true or fake. We tested different models like LSTM, BiLSTM, CNN-BiLSTM, and BERT on the ISOT dataset, and we also tried improving the models by adding more sports news samples. What I learned is that while BERT, the more complex model, performed really well, simpler models like LSTM still showed decent results, especially when the dataset was balanced. Adding more data helped, but it also made it clear that handling sensational language and biased headlines is still a challenge.

One thing that was surprisingly difficult was dealing with headlines that use exaggerated or emotional language. Phrases like ”victory lap” or ”BIG NAME” made it hard for the models to catch the subtle distortions behind the words. These challenges were due to both how the headlines were written (emotionally charged language) and the misleading ideas they were trying to push. Even though the data augmentation helped, there’s still a lot of room to improve, especially when it comes to recognizing these types of biases.

The results were not too surprising, but they did highlight areas where we can do better. It was clear that more advanced models like BERT can better handle the complexities of news headlines, but they still struggle with more subtle issues, like biased or manipulative language.

If I were to keep working on this project, I’d want to expand the dataset even more to cover different topics and areas. I’d also focus on improving the models to handle biased or emotionally charged language more effectively. Looking into more advanced models or adding things like sentiment analysis could be helpful too. Ultimately, the goal would be to make the fake news classification process more accurate and reliable, especially when dealing with tricky headlines in real-world news.

8 Acknowledgements

In the preparation of this report, I adhered to the guidelines regarding the appropriate use of generative AI tools. Specifically:

- **Code Debugging:** I wrote all the code for the project independently. However, I used ChatGPT to assist in identifying and resolving errors in my code. The output provided by the AI was reviewed, validated, and, where necessary, modified to ensure alignment with the

project's objectives. These changes were *major*, as I customized the suggestions to suit the specific context of the code.

- **Research Assistance:** I used ChatGPT to research related work and understand background concepts. The information obtained was used to inform the content of my report but was paraphrased and expanded upon in my own words. The changes made to the AI-provided outputs were *major*, as I integrated them with my own insights and project-specific details.

No sections of the report were directly written, programmed, or illustrated by AI without significant modification or review on my part.

References

Castillo, C., Mendoza, M., and Poblete, B. (2011). Analyzing the credibility of information on twitter. *Proceedings of the 20th international conference on World wide web*, pages 675–684.

Hadi, M. and Ghasemzadeh, M. (2020). Fake news detection using n-grams: A case study on the isot dataset. *Journal of Information Systems*, 45:15–25.

Heejung, K., Chul, K., and Jae, P. (2020). Fake news detection using bert. In *2020 International Conference on Machine Learning*, pages 123–132.

ISOT (2020). Isot fake news dataset. Accessed: 2024-12-05.

Julio, C. and Edgar, R. (2018). Features and methods for automatic fake news detection. *Online Information Review*, 42(5):603–618.

Saqib, M. and Raza, S. (2020). Ensemble classification for fake news detection: A comparative study. *IEEE Access*, 8:15600–15612.

Sebastian, N. and Hadi, M. (2020). A neural network-based approach for fake news detection on isot dataset. *Journal of Data Science and Its Applications*, 10:45–54.

Shu, K., Sliva, A., Wang, S., Tang, J., and Liu, H. (2017). Fake news detection on social media: A data mining perspective. *ACM SIGKDD Explorations Newsletter*, 19(1):22–36.