PROJECT REPORT

A Comparative Analysis of HAN, SVM and XGBoost for Fake News Detection

CS F407 - Artificial Intelligence
Birla Institute of Technology and Science, Pilani
April 22, 2025

Group Members

Aditya Agrawal	2022A7PS0160P
Shikhar Singh	2022A7PS1170P

Abstract

The proliferation of fake news on digital platforms poses significant threats to public trust and social stability, necessitating robust and scalable detection methods. This project presents a comparative analysis of three prominent approaches for fake news detection: **Hierarchical Attention Networks (HAN)**, **Support Vector Machines (SVM)**, and **Extreme Gradient Boosting (XGBoost)**. We systematically evaluate these models using a benchmark dataset, focusing on their ability to differentiate between authentic and fabricated news articles based on textual features. SVM and XGBoost represent traditional machine learning paradigms, while HAN leverages deep learning's hierarchical attention mechanisms to capture nuanced semantic and syntactic information from news content. Our experiments reveal that HAN consistently outperforms SVM and XGBoost in terms of accuracy and F1-score, which may be due to its capacity to model complex linguistic structures. However, SVM and XGBoost demonstrate competitive performance with lower computational requirements, making them viable for resource-constrained scenarios. The findings highlight the trade-offs between computational efficiency and detection accuracy across these models.

Problem Statement

Problem Statement: Given a news article, determine if it is legitimate or fake.

Experimental Setup

Datasets

For this project, we utilized the WELFake dataset with a 80/20 train-test split. The WELFake dataset consists of 72,134 news articles with a balanced distribution: 35,028 articles labeled as real news (class 1) and 37,106 articles labeled as fake news (class 0). This balanced nature of the dataset is advantageous for training models without class imbalance issues that could bias classification results.

FastText Embeddings

We used two different FastText embedding approaches:

1. **Custom-trained FastText Model:** We trained our own FastText model on the WELFake training dataset, allowing the embeddings to be specialized for the specific language patterns and terminology present in news articles.
2. **Pre-trained FastText Embeddings:** We also utilized the publicly available cc.en.300.bin model, which contains 300-dimensional vectors trained on Common Crawl and Wikipedia data.

FastText was chosen for its ability to handle out-of-vocabulary words by utilizing subword information, making it particularly effective for news text that may contain newly coined terms, named entities, and other words not seen during training.

Models Implemented

1. Support Vector Machine

We implement a Support Vector Machine classifier using scikit-learn's LinearSVC. This configuration is particularly effective for processing the high-dimensional FastText embeddings generated from the textual data. The LinearSVC is wrapped with CalibratedClassifierCV to enable probability outputs, as standard SVMs only provide distance from the decision boundary rather than calibrated probabilities.

The model directly processes the sentence embeddings generated by the FastText model, which captures semantic information from the title and text fields. This approach leverages SVM's effectiveness in high-dimensional spaces, allowing the classifier to identify optimal hyperplanes that separate the different document categories based on their dense vector representations rather than sparse word features. The combination of FastText's contextual embeddings with SVM's margin-optimization approach provides a computationally efficient method for text classification.

2. XGBoost

The experimental setup employs XGBoost, a powerful gradient boosting framework, to classify document embeddings. The implementation utilizes the XGBClassifier with carefully selected hyperparameters: 100 gradient-boosted decision trees, a moderate learning rate of 0.1, and tree depth limited to 5 levels to prevent overfitting. The model incorporates regularization strategies through parameters like subsample and colsample_bytree (both set to 0.8), which randomly select 80% of training instances and features respectively for building each tree, enhancing model generalization.

The classifier is configured with a binary logistic objective function appropriate for the fake/real news classification task, and implements early stopping after 10 rounds without improvement to optimize training efficiency. During training, the model processes the same FastText-generated document embeddings used in the SVM approach, leveraging XGBoost's ability to handle high-dimensional data while capturing complex non-linear relationships between features that might be missed by linear classifiers. This gradient boosting approach sequentially builds trees that correct errors made by previous trees, creating an ensemble that typically achieves higher accuracy than single-model approaches for text classification tasks.

3. Hierarchical Attention Network (HAN)

The HAN consists of an embedding layer followed by Bidirectional GRUs and Attention modules for word and then sentence level data. Output size of the embedding layer is set at 200 while the size of the hidden layer (for GRU output) is set at 50. The data was first tokenized using the PUNKT tokenizer in NLTK to attain a 3d tensor of size (num_examples, num_sentences=20, num_words=20) for the HAN. For training, Adadelta was used as the optimizer with learning rate 0.03. Binary cross entropy was used as the loss function. The model was trained for 30 Epochs on an MPS device, which took about 258 minutes. Training loss graph was as follows:

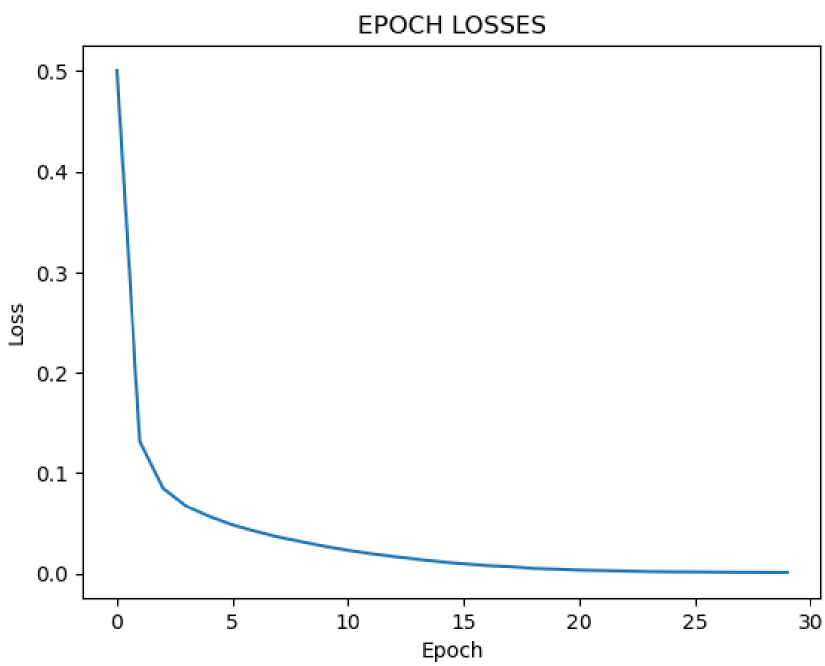

Fig. 1 HAN Training Losses

Results

The results from the models were as follows:

Model	Accuracy	Precision	Recall	F1-Score	MCC
SVM	0.9581	0.9581	0.9581	0.9581	0.9161
XGBoost	0.9558	0.9558	0.9558	0.9558	0.9115
HAN	0.9852	0.9852	0.9852	0.9852	0.9703

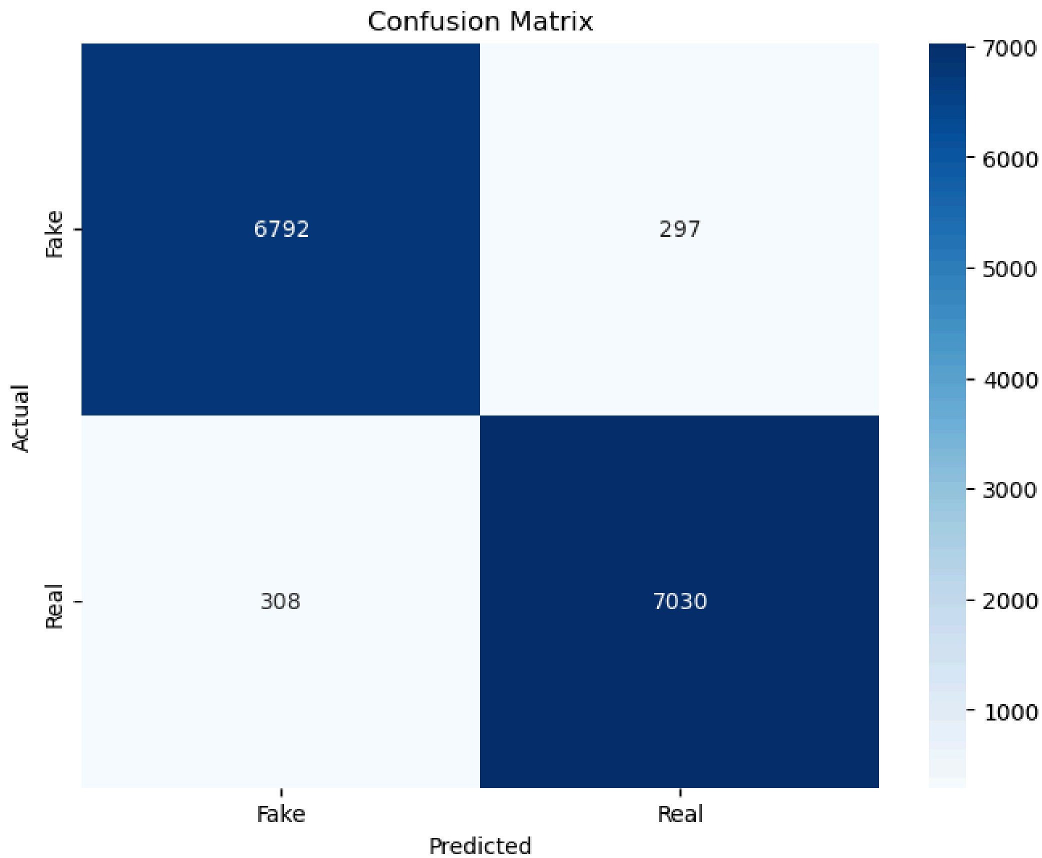

Fig. 2 SVM Confusion Matrix

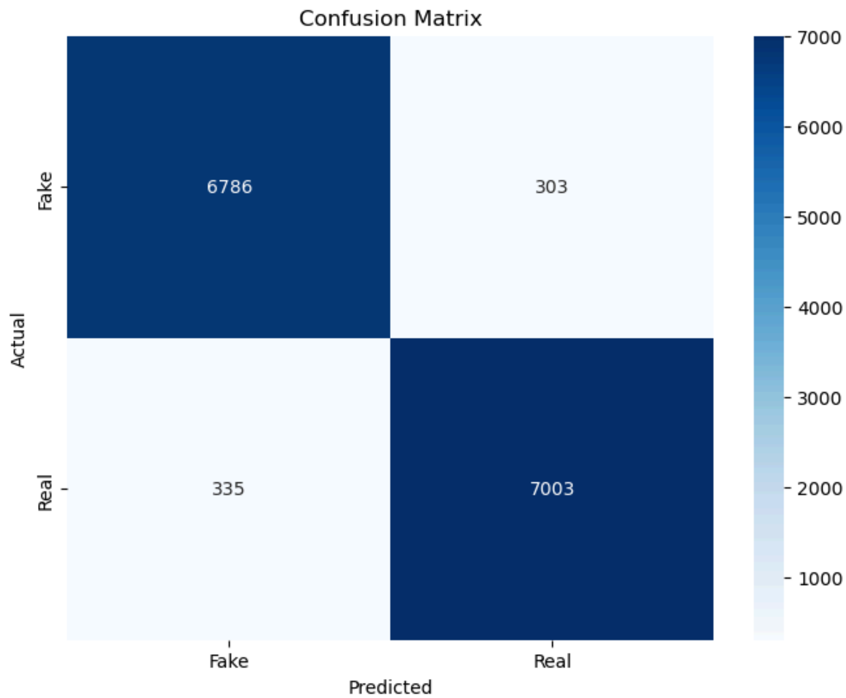

Fig. 3 XGBoost Confusion Matrix

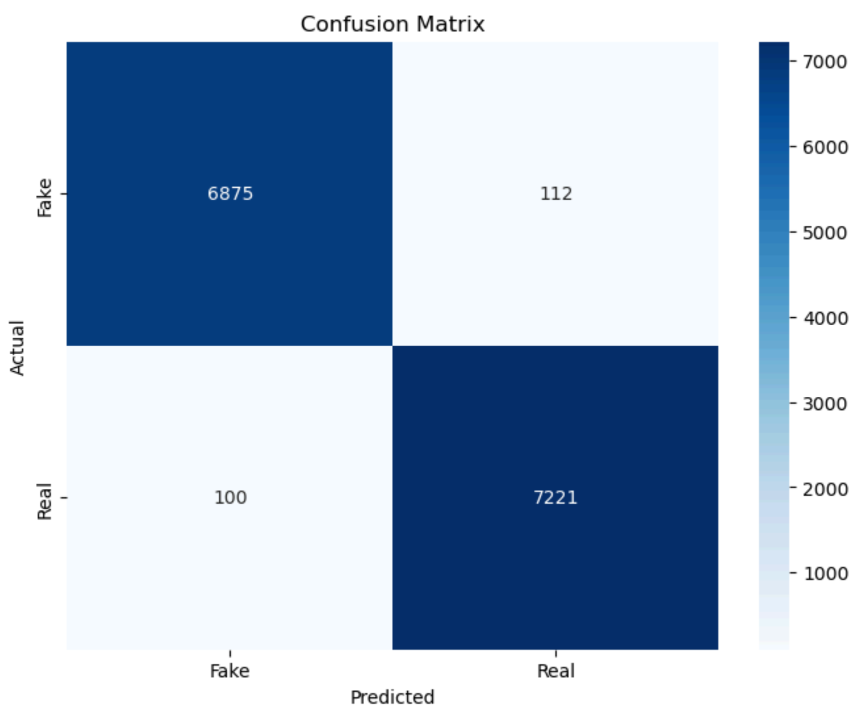

Fig. 4 HAN Confusion Matrix

Conclusion

Based on the evaluation of all three machine learning models, we see that the Hierarchical Attention Network (HAN) significantly outperforms both Support Vector Machine (SVM) and XGBoost approaches, achieving superior results across all performance metrics with an accuracy of 0.9852, precision and recall of 0.9852, F1-score of 0.9852, and Matthews Correlation Coefficient (MCC) of 0.9703. The confusion matrix for HAN reveals its exceptional classification capabilities with only 212 misclassifications, compared to SVM’s 605 and XGBoost’s 638. This substantial difference in misclassifications highlights HAN's ability to effectively capture semantic nuances and contextual relationships within news articles.

While SVM and XGBoost still demonstrated strong performance with accuracies of 0.9581 and 0.9558 respectively, the clear superiority of the HAN model suggests that deep learning approaches leveraging attention mechanisms are particularly well-suited for fake news detection tasks. The hierarchical structure of HAN appears to provide a significant advantage in understanding both word-level and sentence-level features.

References

1. **Hashmi, E., Yildirim Yayilgan, S., Yamin, M. M., Ali, S., & Abomhara, M.** (2024). Advancing Fake News Detection: Hybrid Deep Learning With FastText and Explainable AI. *IEEE Access, 12*, 44462–44466. https://doi.org/10.1109/ACCESS.2024.3381038

2. **Berrondo-Otermin, M., & Sarasa-Cabezuelo, A.** (2023). Application of Artificial Intelligence Techniques to Detect Fake News: A Review. *Electronics, 12*(24), 5041. https://doi.org/10.3390/electronics12245041

3. **Hu, L., Wei, S., Zhao, Z., & Wu, B.** (2022). Deep learning for fake news detection: A comprehensive survey. *AI Open, 3*, 133–155. https://doi.org/10.1016/j.aiopen.2022.09.001

4. **Granik, M., & Mesyura, V.** (2017). Fake News Detection Using Naive Bayes Classifier. *2017 IEEE First Ukraine Conference on Electrical and Computer Engineering (UKRCON)*. https://doi.org/10.1109/UKRCON.2017.8100379

5. **Ozbay, F. A., & Alatas, B.** (2020). Fake news detection within online social media using supervised artificial intelligence algorithms. *Physica A: Statistical Mechanics and Its Applications*, 540, 123174. https://doi.org/10.1016/j.physa.2019.123174

6. **Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alex Smola, and Eduard Hovy.** (2016). Hierarchical Attention Networks for Document Classification. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1480–1489, San Diego, California. Association for Computational Linguistics.

7. **Facebook AI Research.** (n.d.). *Crawl-based multilingual word vectors*. fastText. https://fasttext.cc/docs/en/crawl-vectors.html

8. **P. K. Verma, P. Agrawal, I. Amorim and R. Prodan,** "WELFake: Word Embedding Over Linguistic Features for Fake News Detection," in IEEE Transactions on Computational Social Systems, vol. 8, no. 4, pp. 881-893, Aug. 2021, doi: 10.1109/TCSS.2021.3068519.