Fake news detection using augmented LLM

Yi Liu, Zhuoyan Ma, Ruozhu Wang

Mar 31, 2024

Abstract

The proliferation of fake news in modern media compromises information integrity. By incorporating professionally verified fact-checking data, our project seeks to leverage Large Language Models' extensive knowledge, reasoning abilities and accessibility to external sources to identify and combat fake news effectively.

Introduction

The proliferation of fake news in contemporary media landscapes presents a formidable challenge to the integrity and reliability of information dissemination. Fake news, which is fake material in a news media format but is not processed properly by news agencies [1], has accelerated dissemination through users' social media. Research in social science has suggested that the consumption of fake news leads to cognitive bias, which has become the main reason people make irrational decisions[10]. Thus, understanding the mechanisms underlying the composition of fake news and being able to distinguish them is crucial for combating its proliferation and mitigating its impact on public opinion and decision-making.

Early research concentrated on leveraging textual information derived from the article's content, such as statistical text features and sentimental analysis, which heavily relies on data pre-processing[14]. However, with the advancement of natural language processing (NLP) and deep learning algorithms, the detection of fake news has achieved greater accuracy while allowing automated feature extraction on high-dimensional data. Despite these advancements, these technologies still fall short in comprehensively analyzing fake news in a manner that parallels human logical reasoning.

The existence of large language models has taken the fake new detention techniques to a higher level. Previous research has shown that LLMs 1) contain a significant amount of world knowledge; 2) have powerful capacities in arithmetic reasoning, commonsense reasoning, and symbolic reasoning; 3) can be augmented with external knowledge, tools, and multimodal information, and can even operate as autonomous agents [4]. However, to our best knowledge, the reasoning ability of LLM hasn't been fully boosted via prompt engineering. There is not much literature using augmented LLM with external knowledge to detect fake news, and we hope our project can fill this gap and improve models' accuracy by enabling reasoning and searching from external sources.

In our project, we aim to take advantage of the Large Language Models (LLMs) to combat misinformation. Our input will be claims from newspaper articles, and output will be how likely the claims are true statements (i.e. not fake news). As LLMs contain a large amount of internal knowledge, we will first determine a naive truthfulness prediction. In the second step, we will conduct an augmented LLM by feeding the model with ground truth reasonings, which are scrapped from professional-verified fake news detecting websites. Then, to further boost the model, we will adopt the augmented LLM with external knowledge searched from simple search APIs to improve our prediction accuracy.

Related Work

Prior Fake news detector

The exploration of fake news detection has evolved significantly over the years, with early efforts largely centered on traditional machine learning. Those models focus on leveraging handcrafted features, like morphological, psychological, and readability features based on content [3] and lexicon-level, syntax-level, and semantic-level features based on the writing style [23], to train a classifier capable of distinguishing fake news from real statements. Despite achieving acceptable accuracy, these models struggled with the adaptability and complexity of fake news, necessitating the development of more sophisticated models. This led to the exploitation of pre-trained small language models (SLMs), like BERT [5], RoBERTa [13], ALBERT [11] to capture contextual meanings of words, enabling a deeper understanding of language nuances. However, their limitations in knowledge and capabilities also hinder further improvements in fake news detectors. For instance, BERT is pre-trained on text corpus like Wikipedia [5] and thus can't handle news that requires knowledge not included.

LLM as fake news detector

Compared with SLMs, Large Language models(LLMs) [2, 16, 15] are trained on the large-scale corpus, including Common Crawl, WebText2, and Wikipedia [2], enabling them to detect wrong information via internal knowledge base. Previous works also show that LLM has an emergent ability to answer questions truthfully [19] due to the huge amount of parameters and prompting strategies like Few-shot prompting [2] and Chain-of-thought[20]. Recent work [8] indicated that LLM with few-shot CoT prompting as a fake news detector on Weibo and GossipCop datasets underperforms fine-tuned SLMs. The limitation is expected to be rooted in hastily drafted prompting techniques. We'll further try advanced prompting strategies, like instruct prompting [16], self-ask [17] or fine-tune LLMs[9] on our datasets.

External augmentation for LLM

In advancing the battle against the proliferation of fake news, several pioneering studies underscore the potential of augmenting Large Language Models (LLMs) with Retrieval-Augmented Generation (RAG) systems or other external tools to verify the factuality of the content. A comprehensive survey on RAG demonstrates that RAG process not only informs the subsequent reasoning generation but also ensures that LLM's responses are grounded in retrieved evidence, thereby enabling the verification of news content for accuracy[6]. Additionally, RETA-LLm provides a general prototype of RAG-augmented LLM, including modules of document retrieval, relevant information extraction, and fact-checking [12]. This suggests that RAG can be a solution for LLM, as a fake news detector, to access up-to-date, factual information from beyond their initial training data, and they can more accurately assess the veracity of news content. Moreover, the ReAct model integrates reasoning with tool use by prompting LLMs to generate interleaved verbal reasoning traces and tool calls, which has been shown to effectively enhance LLMs' problem-solving capabilities [22]. Similarly, several studies examine that LLMs' limitations in factual lookup can be mitigated through tool use, and LLMs can decide which tools to use, when to use tools, and how to best incorporate the evidence-based response into future token predictions to a certain extent[18, 21]. These studies collectively illustrate the significant promise of leveraging external tools and knowledge bases to improve the detection of misinformation, providing a compelling direction for future research in the domain of automated fake news detection.

Proposed Work

We plan to sequentially test LLM's ability to detect fake news. Firstly, we will use chain-of-thought prompting to guide LLMs to extract a list of assertions from a claim, then let LLMs determine the truthfulness of each assertion based on its internal knowledge, and decide the truthfulness scores or level based on the proportion of false assertions to all assertions. We'll compare LLMs' detection accuracy with the that of state-of-art detection model before the era of the large language model, fine-tuned BERT [5], to examine whether LLMs will be a good substitute for fake news detection tasks.

Secondly, we'll test the performance of LLMs augmented with the external knowledge base. When verifying the truthfulness of news is beyond LLM's own training memory, LLM can't distinguish the correctness of facts and may mislabel fake news as true. As our datasets contain reasoning processes written by professionals, treated as ground truth, we can construct an external knowledge base using all the links or documents embedded in reasoning steps and augment LLMs with RAG. We'll compare the augmented performance with pure LLMs using prompting approaches, and check whether LLMs correctly use those injected knowledge and result in more accurate estimation.

After that, as accessing external tools will further boost LLM's reasoning ability and factual reliability, we'll try the prompt techniques in the ReAct paper to augment LLM with action space. In the action space, LLM will implement information retrieval using search, lookup, and finish actions from a simple Google Search API or Wikipedia web API. Since there are experts' detailed reasoning processes for news truthfulness scores in our dataset, we will summarize the possible tools needed for detection and create a toolkit. We may need further prompts to check whether the LLM can decide when to use a tool and select the appropriate tool based on its planned steps.

Dataset

LIAR-PLUS is a dataset consisting of 12.8K human-labeled short statements from Politifact.com and each statement is evaluated for its truthfulness. The data set contains the following features:

- **Target label**: Pants-fire, False, Barely-true, Half-true, Mostly-true, True. Here, pants-fire is derived from "liar, liar, pants on fire", indicating the statement is extremely false.
- **Statement** (text): The entire PolitiFact article, in rare cases just the title.
- **Subject** (categorical, non-ordinal): Subject of the statement, ie.) health care, social security, economy, etc.
- **Job title** (categorical, non-ordinal): Job title of the speaker, ie.) President, presidential candidate, state representative, etc.
- **State info** (categorical, non-ordinal): US state where the speaker is based. ie.) Massachusetts, New York, Florida, etc.
- **Party affiliation** (categorical, non-ordinal): the party affiliation of the speaker republican, democrat, etc.
- **Context** (text): the context, such as the venue/location of the statement.
- **Justification** (text): the reasoning statement.

Data Cleaning and Preprocessing

- **Missing Value**: In our dataset, certain variables exhibit missing values. For columns with a missing value rate below 1%, such as *subject* and *affiliation*, we opted to remove rows that contained these missing values. On the other hand, columns like *job title* and *state info*, which have a relatively higher rate of missing values, presented a challenge where direct removal of missing entries would result in significant data loss. To address this, we employed data imputation techniques, leveraging the distribution of existing values within these columns to fill in the gaps.

- **Categorical Data**: We employed one-hot encoders to convert categorical variables into a binary matrix representation. For columns with a large number of subcategories, such as *job title* and *subject*, we strategically one-hot-encoded only the top n categories by frequency to prevent overly sparse data. The value of n was determined to capture the top 50% of the data volume, with the remaining less frequent subcategories aggregated under a single "Others" category.

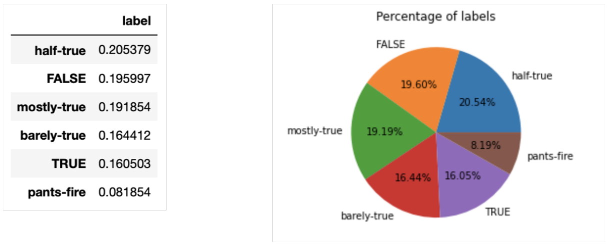
- **Text Cleaning**: We refined the texts by converting all words to lowercase, and removing punctuation, HTML tags, and stop-words (e.g., "the", "a") that contribute minimally to sentence meaning. By doing so, we can reduce the number of features coming from statements, while keeping most of the important information. For stop-words, we used the English stop words corpus offered by The Natural Language Toolkit (NLTK), which is a powerful and resourceful package commonly used for NLP.

- **Text Tokenization**: Before feeding data into our model, it's essential to transform textual statements into numerical features. This process involves converting a collection of text documents into a matrix of token counts. In other words, each matrix row captures the frequency of token occurrences within a specific statement. To achieve so, we utilized the CountVectorizer from Sklearn, setting the *max_df* parameter to 0.8. This setting instructs the vectorizer to exclude words appearing in more than 80% of the statements from the vocabulary, given that such common words offer little unique information for analysis. Furthermore, to diminish the influence of words that appear excessively across the corpus —which may dilute the significance of more informative features— we applied a normalization technique. Specifically, we converted the count matrix into a *tf-idf* representation. Here, *tf* denotes the frequency of each word per document, while *idf* reflects the inverse proportion of documents containing each word, thereby highlighting the importance of less frequent terms. For this purpose, Sklearn's TfidfVectorizer was employed. Both CountVectorizer and TfidfVectorizer are trained on the development corpus and then applied to the testing corpus.

Data Exploration and Insights

Predicted Variable: Label

The distribution of labels is approximately balanced (Figure 1). False-related statements, including ”pants-fire,” false, and barely true, make up 44.23% of the data, while true-related statements, including half-true, mostly true, and true, make up 55.77% of the data, showing a slight skew towards true statements. Half-true statements have the highest count, while statements labeled as pants-fire have the lowest count.



	label
half-true	0.205379
FALSE	0.195997
mostly-true	0.191854
barely-true	0.164412
TRUE	0.160503
pants-fire	0.081854

Figure 1: Distribution of the predicted variable

Key Topics

In Figure 2, in terms of 1-grams, ”Says” emerges as the most prevalent word in these statements, with ”Obama” notably being the only politician to feature prominently among the most frequent words. Topics such as health care, health insurance, and social security garner significant attention. However, the breakdown of the most popular 4-grams doesn’t offer much additional insight compared to 3-grams, considering the limited size of our dataset. Nevertheless, discussions surrounding what President Obama expresses remain a highly popular topic in the statements.

Top Job Titles

We plot a bar chart to show the frequency of the top 30 job titles in the dataset (Figure 2). The job title ”President” has the highest frequency, suggesting that statements from or about the President are most common in our fake news dataset. ”U.S. Senator” ranks second, followed closely by ”Governor” and ”President-Elect”, indicating that high-ranking public officials are prominently featured in the data. The titles show a diverse representation of political and governmental positions, ranging from national roles (like ”U.S. Representative”) to more specific ones (such as ”State Representative of Ohio 10, Woman” or ”Candidate for U.S. Senate physician”).

Moreover, the variety extends to non-elective positions, including a ”Social media posting” and a ”Co-host on CNN’s ’Crossfire’”, which suggests that the dataset not only covers

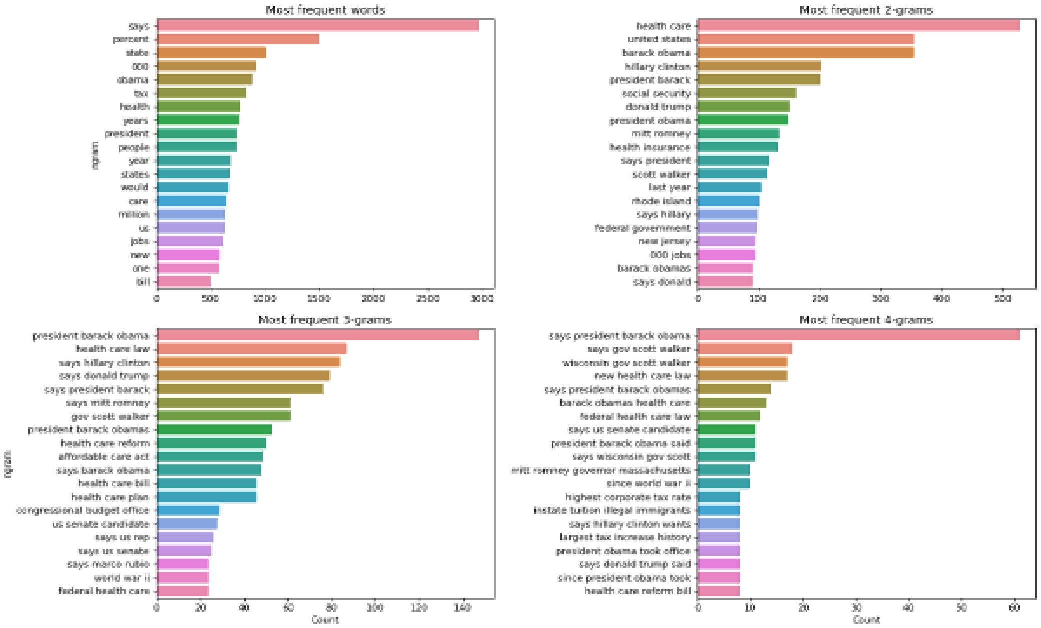


Figure 2: Distribution of the ngrams

elected officials but also includes a variety of influencers and commentators, who also play important role in delivery information to the public.

Top Speakers

In figure 4, we show the top 20 speakers from our dataset. The first bar, significantly higher than the others, represents statements attributed to Barack Obama, indicating he is the most quoted or referenced speaker in the dataset. Other prominent figures include Donald Trump and Hillary Clinton, but with notably fewer mentions than Obama. The descending order of the bars suggests a tapering frequency of references to other individuals, with the least referenced still being relatively prominent compared to the general population.

Top Mentioned Context of Statements

We conduct a graph (figure 5) displays the various contexts in which statements were made, with 'news release' being the most common, followed by 'an interview' and 'a speech'. This suggests that formal releases and direct communication are the primary sources of statements in the dataset. Other contexts like 'a tweet', 'a radio interview', and 'a news conference' also appear frequently, reflecting a diverse range of platforms for statements dissemination.

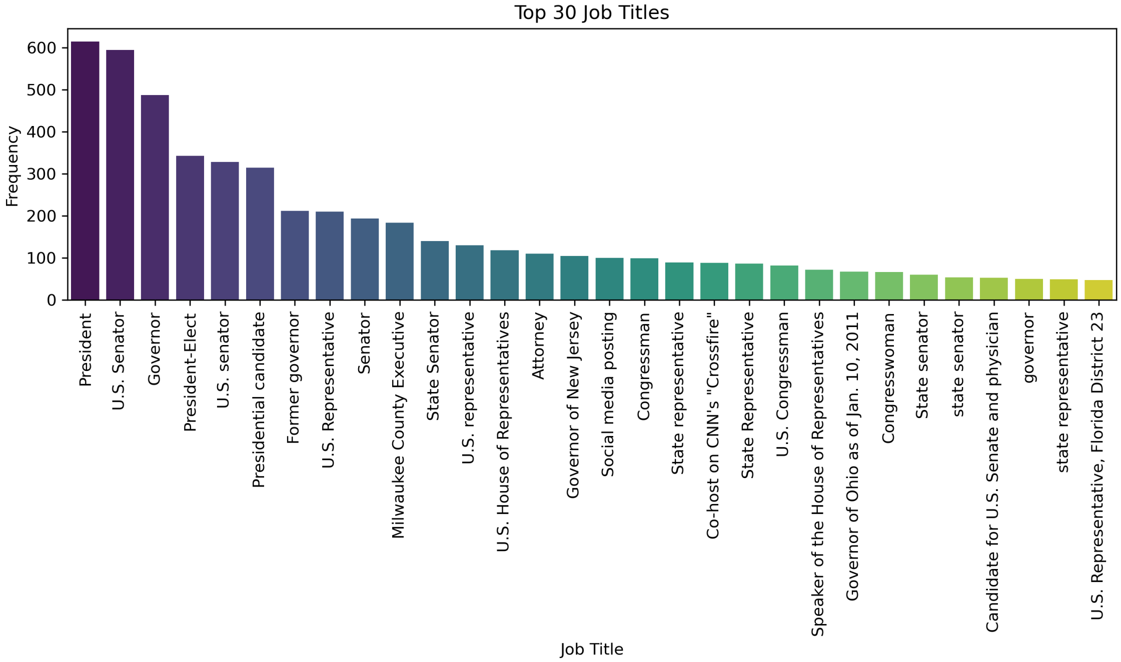


Figure 3: Distribution of the top job titles

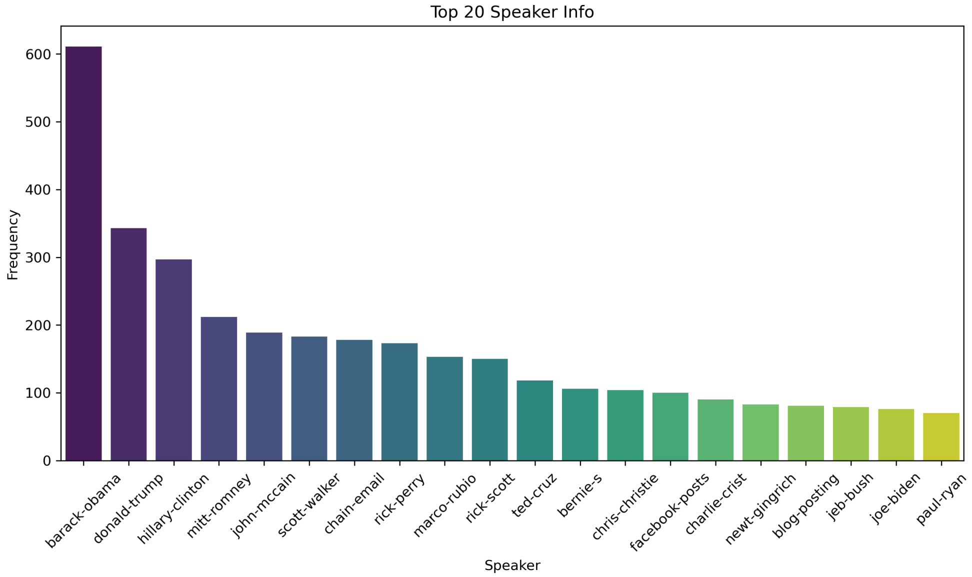


Figure 4: Distribution of the top speakers

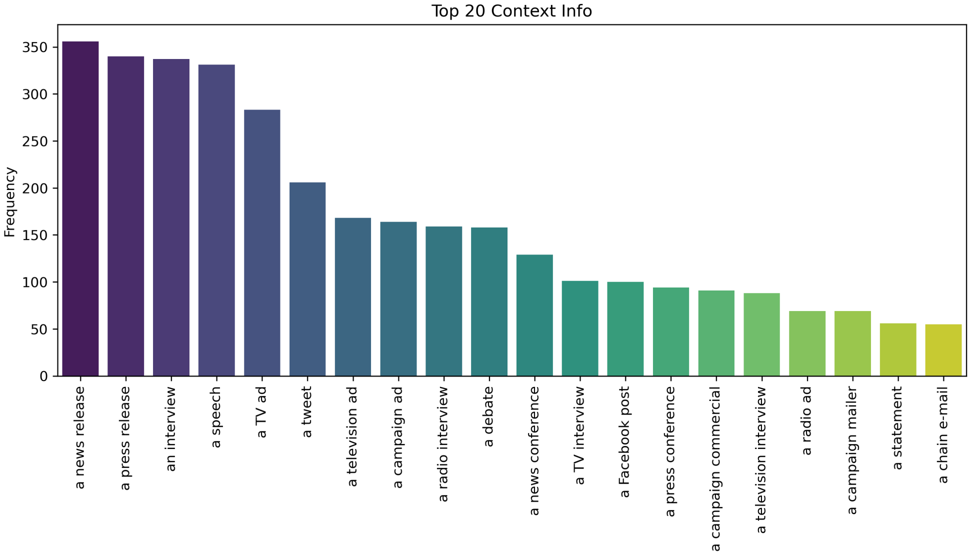


Figure 5: Distribution of the Context of Statements

Party Affiliations V.S. Label

Utilizing the top three party affiliations, which make up 94% of the data, we observe that the Republican Party is associated with the highest number of statements. Interestingly, the Democratic Party has a greater proportion of statements categorized as ”true” (mostly-true, true) compared to those classified as ”false” (pants-fire, FALSE). Conversely, the Republican Party exhibits a higher number of ”false” statements than ”true” ones. On the other hand, the group without any party affiliation appears to have a relatively even distribution of truthful statements. (Figure 6).

Job Title V.S. Label

The analysis focused on job titles appearing more than 200 times in the dataset, revealing notable patterns in label distributions across eight job titles (Figure 7). Among these, ”President” and ”U.S. Senator” emerge as the most frequent job titles, with similar distributions of labels: fewer false statements and a predominance of statements rated at least half-true. An interesting finding is that statements attributed to ”President-Elect” predominantly lean towards falsehood, with minimal occurrences of true statements compared to other job titles. Conversely, for Presidential candidates, the trend is reversed. For ”former governor” and ”U.S. Representative,” label distributions are akin, with a majority of statements being half-true, a small proportion being pants-fire, and a relatively even split among the remaining labels.

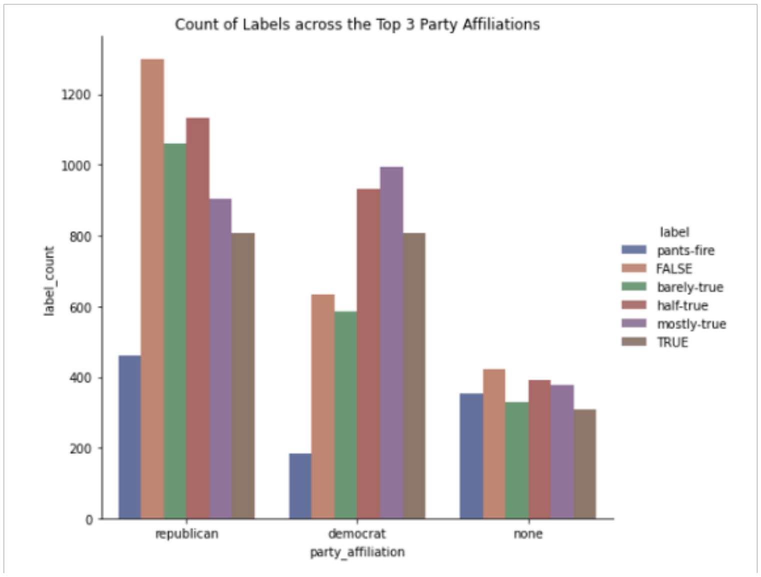


Figure 6: Party Affiliation and Label

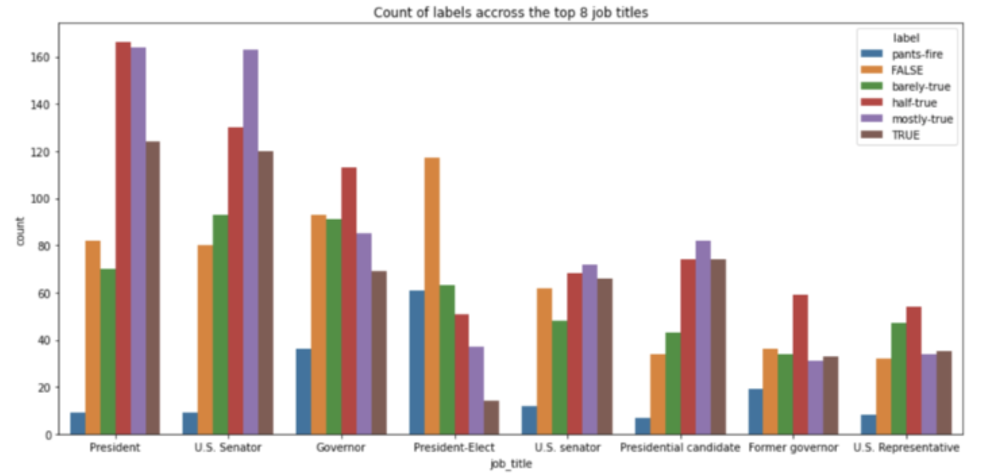


Figure 7: Job Title and Label

Correlation Analysis of Label Truthfulness

We convert labels from categorical to numerical, allowing for correlation calculations with non-textual features (Figure 8). Top 6 correlations are bar plotted. Democrats' statements show the highest positive correlation with truthfulness, suggesting relatively greater truth. In contrast, Republican statements rank 4th in negative correlation, with Donald Trump's statements ranking 2nd, suggesting their words decrease authenticity. This aligns with previous insights from label distributions across party affiliations, possibly indicating bias in statement evaluation or genuine differences in truthfulness. Statements in president election are also a big negative contributor to statement truthfulness, which makes sense.

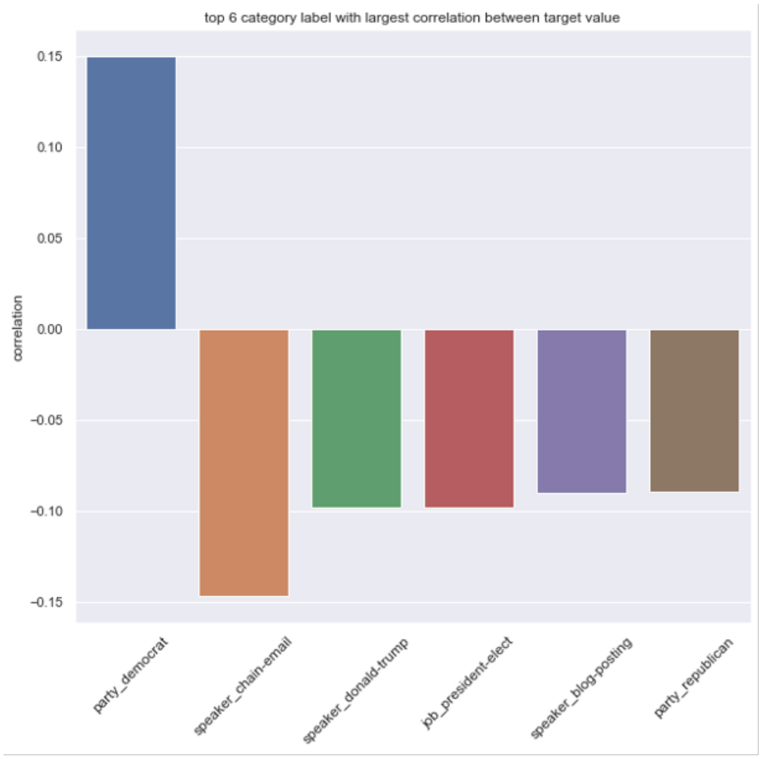


Figure 8: Label Correlation

Emotion Analysis of Statement

We investigated whether the emotional tone of statements influences the classification of news as true or false. To analyze the sentiment, we utilized the Valence Aware Dictionary and sEntiment Reasoner (VADER) from NLTK, a tool designed to assess sentiment in social media text. VADER calculates sentiment scores by summing the scores of each word in a statement and normalizes the result between -1 and 1. The more the score is close to 1, the higher the positivity the statement has. Vice versa. As illustrated in the boxplots of VADER scores across different truth labels, the results revealed no strong emotional trends differentiating false claims from true ones, except for a slight increase in neutrality in true claims. This was somewhat unexpected but understandable considering that most statements are formally written, using emotionally neutral language, and are typically brief, limiting the presence of many emotionally charged words.

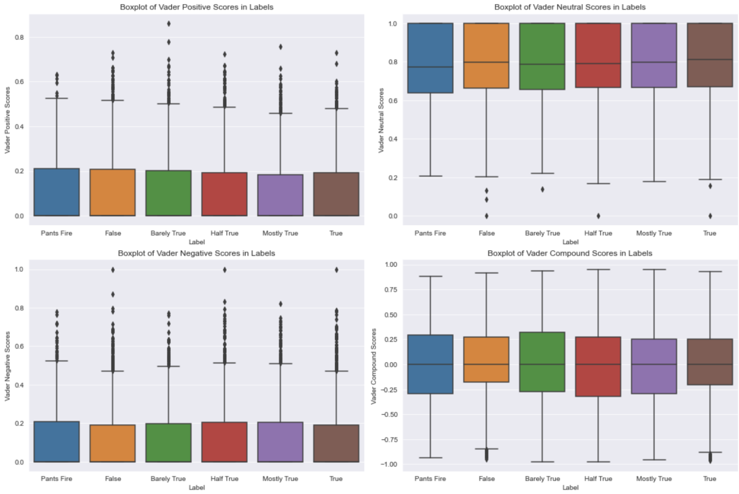


Figure 9: Label Correlation

Evaluation

In our collected datasets, each statement will be taken as input for each model and each model will be prompted to assign a truthfulness label to each claim. Specifically, the labels will be categorized into True-related, including ”True,” ”Mostly True,” and ”Half True,” and False-related, including ”Pants on Fire/False”, ”Mostly False”, and ”Barely True”. Hence, the model aims for a binary classification of fake news. The true labels provided by professionals within our dataset will serve as the gold standard against which the model's predictions will be compared. To evaluate the efficacy of the model's classification, we primarily applied the accuracy score, which provided a straightforward measure of the proportion of correctly classified claims out of the total. As the dataset is pretty balanced, where true-related labels count for 55% the accuracy score is an effective evaluation metric. Table 1 presents the results of the deep learning models.

Table 1: Results of Deep Learning Models

Model	**Accuracy Score**
Statement Only	
Logistic Regression	0.6078
SVM	0.6216
GLoVe+RNN	0.6815
Word2Vec+RNN	0.6230
BERT	0.6147
Statement + Meta Data	
Random Forest	0.7168
Gradient Boosting	0.7239
XGBoost	0.7401

Logistic Regression/SVM(statement only)

We initiated our fake claim classification by employing CountVectorizer and TF-IDF for preprocessing textual data, as demonstrated in the Data cleaning and preprocessing section. As a baseline, we attempted a couple of simple but performant models, such as logistic regression and Support Vector Machine. Both models obtained an accuracy score of around 0.61.

RNN(statement only)

In this step, we experimented with more state-of-art techniques in the NLP word: combinations of word embedding models and recurrent neural network (RNN). In particular, we tried both GloVe and Word2Vect as word embedding models for our RNNs.

The GloVe (Global Vectors for Word Representation) is an unsupervised learning algorithm used to obtain vector representations for words. Developed by Stanford NLP, it includes models trained on a diverse set of corpora such as Wikipedia and Twitter. Specifically, we explored the GloVe6b model, trained on Wikipedia 2014 and Gigaword 5, and a Twitter-specific model trained on 2 billion tweets. Incorporating these embeddings with statements into a simple RNN with LSTM architecture improved our validation accuracy to approximately 0.68.

In addition, Word2Vec is another popular word embedding technique utilizing a shallow neural network. It comes in two flavors: Continuous Bag of Words (CBOW) and Skip-Gram. CBOW predicts a target word from a window of surrounding context words, whereas Skip-Gram does the opposite, using a word to predict its surrounding context. This method effectively reduces dimensionality, which enhances its training efficiency. We experimented

with pre-trained Word2Vec models on the statements and combined with RNN and LSTM architectures. In our tests, the Word2Vec combined with RNN outperformed the LSTM version, achieving an classification accuracy of 0.62 on our dataset of statements.
Overall, GLoVe with RNN, achieved the best accuracy score, when being trained on statement only.

BERT(statement only)

Bidirectional Encoder Representation Transformer(BERT) is designed to pre-train deep bidirectional representations from unlabeled texts by jointly conditioning on both left and right contexts in all layers. We tested 'bert-large-uncased' from HuggingFace, notable for its 24 layers and 336M parameters, which suited our preprocessed lowercase texts. We opted for a max sequence length of 100 to balance information retention and manageability. Utilizing AdamW as our optimizer and a learning rate of 2e-5, initial training across 10 epochs suggested optimal performance at the 5th epoch, after which we observed a decline in validation accuracy and an increase in loss.
In our evaluation, the BERT model achieved an accuracy of 0.6147 on the testing data. When classifying fake news, we focus more on the model's ability to distinguish false-related ones among all the claims so that we can effectively stop the spread of fake news. In this case, BERT obtained a 0.63 true positive rate for false-related claims.

Random Forest/Gradient Boosting/XGBoost (statement + meta data)

In our initial models, which solely relied on textual data from statements, we observed limited success in distinguishing false-related statements, with accuracies hovering around 0.62. This performance was only slightly better than random guessing. One potential reason for limited explainational power of statements in classification is the relatively brief length of statements in our dataset averaging just over 77 words post-cleanup. Therefore, they might not provide sufficient information for effectively distinguishing between true and false claims.
For the next step to enhance our model's efficacy, we decided to integrate additional meta variables, such as Job title, State info, and party affiliation, into our machine learning framework. Combining the preprocessed textual statements with encoded categorical and numerical features produced a spare training matrix. Therefore, we selected Random Forest, Gradient Boosting, and XGBoost, which are expert at training with sparse data. Random Forest classifier achieved an accuracy of 0.7168. Gradient Boosting reached 0.7239 and XGBoost obtained an accuracy of 0.7401. These results are achieved after tuning model's hyper-parameters.

Table 2: Results of Large Language Models

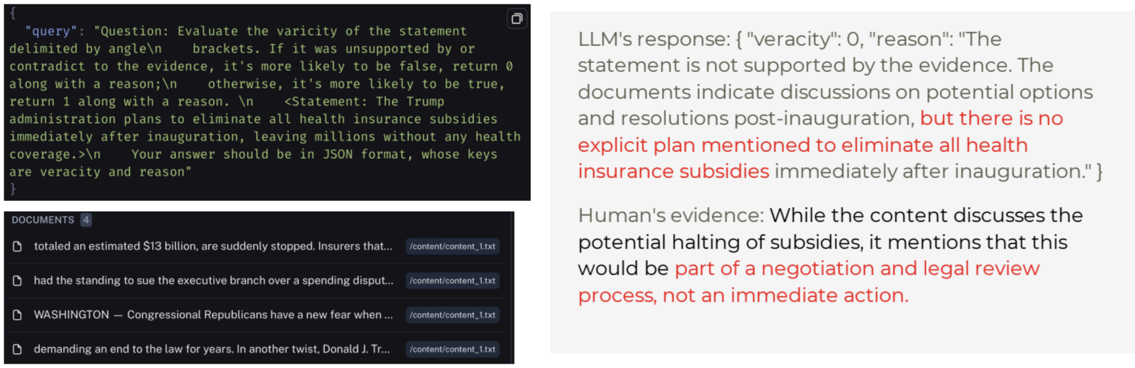
Model	**Accuracy Score**
Statement Only or **Statement Only + Justification**	
Zero-shot prompting	0.5369
Zero-shot Chain-of-Thought	0.5635
Zero-shot Chain-of-Thought from Textual Description Perspective	0.4765
Zero-shot Chain-of-Thought from Commonsense Perspective	0.4698
Statement + Justification + Meta Data	
Zero-shot prompting	0.5775
Zero-shot Chain-of-Thought	0.6127
Zero-shot Chain-of-Thought from Textual Description Perspective	0.5567
Zero-shot Chain-of-Thought from Commonsense Perspective	0.55

Large Language Model

We evaluate the potential of GPT-3.5-turbo as a fact-checker tools, using 3 prompting learning paradigms:

Zero-shot Prompting

Zero-shot prompting constructs prompt only containing task description and the given news, justification as professional's reasoning process and meta data, including the news' context, topics and the speakers' historical records. The Meta-data contain more information about the speakers' historical records of creating fake news of each level and the severity relative to the average number of fake news in the market. To make the response more proficient and decrease the refusal ratio, we optionally adopt the role-playing techniques when describing the task. Example prompts are as below:
System prompts: *As a talented fact-checker in journalism who will evaludate the veracity and truthfulness of the given news claim and answer the veraity score following users' instruction. Your judgments should be supported by evidence and logical reasoning. Please consider any potential biases or uncertainties in the provided information and address them in your evaluation*
User query: *Answer should be in JSON format whose keys are score and reason.Question: Given the following news claim, justification of other fact-checkers and other relevant information about the news claim and its speaker, you need to predict veracity of the news claim and tell me the reason. If it's more likely to be a real message,return 1; otherwise, return 0.*
NEWS-CLAIM: ...; Justification:...; Subject-News: ...; Context-News:...; Meta-data: ...
The highest accuracy of using zero-shot prompting to detect fake news is 58% with information from meta-data.



```
{
  "query": "Question: Evaluate the varicity of the statement
delimited by angle\n    brackets. If it was unsupported by or
contradict to the evidence, it's more likely to be false, return 0
along with a reason;\n    otherwise, it's more likely to be true,
return 1 along with a reason. \n    <Statement: The Trump
administration plans to eliminate all health insurance subsidies
immediately after inauguration, leaving millions without any health
coverage.>\n    Your answer should be in JSON format, whose keys
are veracity and reason"
}
```

DOCUMENTS 4

totaled an estimated $13 billion, are suddenly stopped. Insurers that... /content/content_1.txt

had the standing to sue the executive branch over a spending disput... /content/content_1.txt

WASHINGTON — Congressional Republicans have a new fear when ... /content/content_1.txt

demanding an end to the law for years. In another twist, Donald J. Tr... /content/content_1.txt

LLM's response: { "veracity": 0, "reason": "The statement is not supported by the evidence. The documents indicate discussions on potential options and resolutions post-inauguration, but there is no explicit plan mentioned to eliminate all health insurance subsidies immediately after inauguration." }

Human's evidence: While the content discusses the potential halting of subsidies, it mentions that this would be part of a negotiation and legal review process, not an immediate action.

Figure 10: Enter Caption

Chain-of-Thought Prompting

Chain-of-Thought Prompting only adds eliciting sentence ”Let’s think step by step” in the end of user query, to encourage LLMs to decompose the task and to reason. Mirrored the perspective of real news fact-checker, we also try variants of CoT as well, for instance ”Let’s think step by step from textual description perspective.” and ”Let’s think step by step from Commonsense perspective.” From Table 2, LLM with CoT gain the highest accuracy, as 61%, using meta-data as well.

RAG augmentation

After comparing the detection performance of LLM with the those of previous model, we attribute the ineffectiveness of Chain-of-Thought to the complexity of fake news formation, intention, and the mechanism of generation. We only focused on fake news due to lack of evidence or contradict to the evidence from citation materials. We self-generated 10 fake news only due to unsupported evidence and 5 real statement for each piece of news in a news summary dataset [7] , and recorded the citation text chunks that were generated or contradicted. Then, we used an LLM augmented with RAG to detect fake news. The procedures of fake news detection using LLM augmented with RAG include retrival of relevant text chunks and generation via LLM’s reasoning. When we use Langsmith to track the retrieval situation of the LLM, the total repository contains 10 news summaries, but the LLM always manages to retrieve 3-4 text chunks from a single document related to a statement, which always includes a text chunk where the statement is generated from or contradicted. After combining the context, the LLM’s response and the logic used by humans to generate fake news are roughly the same. The overall accuracy is 94.56%. This shows that when we understand the formation mechanism of fake news, have the material behind generating news statements, and can instruct the LLM with clear definitions, the LLM with RAG becomes a very promising tool for fake news detection

Conclusion

The spread of fake news in today's media is a serious problem that threatens the reliability of information. With LLMs' vast knowledge base and reasoning capabilities, our project will build up LLMs to detect fake news and compare its performance with traditional deep learning and machine learning-based algorithms.

In conclusion, detecting fake news requires more than just analyzing the statements themselves. Fake news often aims to mimic the truth, making its detection a complex task. However, by incorporating metadata, particularly characteristics of the users who generate the content, significant progress can be made. The RAG method emerges as a promising approach in this endeavor. Moreover, linking with external knowledge bases significantly enhances classification accuracy, surpassing the 90% mark. This comprehensive approach underscores the necessity of multifaceted strategies in combating the dissemination of misinformation.

References

[1] Azizah, S.F.N., Cahyono, H.D., Sihwi, S.W., Widiarto, W.: Performance analysis of transformer based models (bert, albert and roberta) in fake news detection (2023)

[2] Brown, T.B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D.M., Wu, J., Winter, C., Hesse, C., Chen, M., Sigler, E., Litwin, M., Gray, S., Chess, B., Clark, J., Berner, C., McCandlish, S., Radford, A., Sutskever, I., Amodei, D.: Language models are few-shot learners (2020)

[3] Castelo, S., Almeida, T., Elghafari, A., Santos, A., Pham, K., Nakamura, E., Freire, J.: A topic-agnostic approach for identifying fake news pages. In: Companion Proceedings of The 2019 World Wide Web Conference. WWW '19, ACM (May 2019). https://doi.org/10.1145/3308560.3316739, http://dx.doi.org/10.1145/3308560.3316739

[4] Chen, C., Shu, K.: Combating misinformation in the age of llms: Opportunities and challenges (2023)

[5] Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: Bert: Pre-training of deep bidirectional transformers for language understanding (2019)

[6] Gao, Y., Xiong, Y., Gao, X., Jia, K., Pan, J., Bi, Y., Dai, Y., Sun, J., Guo, Q., Wang, M., Wang, H.: Retrieval-augmented generation for large language models: A survey (2024)

[7] Harinatha, S.R.K., Tasara, B.T., Qomariyah, N.N.: Evaluating extractive summarization techniques on news articles. In: 2021 International Seminar on Intelligent Technology and Its Applications (ISITIA). pp. 88–94. IEEE (2021)

[8] Hu, B., Sheng, Q., Cao, J., Shi, Y., Li, Y., Wang, D., Qi, P.: Bad actor, good advisor: Exploring the role of large language models in fake news detection (2024)

[9] Hu, E.J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., Chen, W.: Lora: Low-rank adaptation of large language models (2021)

[10] Kim B, Xiong A, L.D.H.K.: A systematic review on fake news research through the lens of news creation and consumption: Research efforts, challenges, and future directions. (2021)

[11] Lan, Z., Chen, M., Goodman, S., Gimpel, K., Sharma, P., Soricut, R.: Albert: A lite bert for self-supervised learning of language representations (2020)

[12] Liu, J., Jin, J., Wang, Z., Cheng, J., Dou, Z., Wen, J.R.: Reta-llm: A retrieval-augmented large language model toolkit (2023)

[13] Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., Stoyanov, V.: Roberta: A robustly optimized bert pretraining approach (2019)

[14] Mridha, M., Keya, A., Hamid, M., Monowar, M., Rahman, M.: A comprehensive review on fake news detection with deep learning. IEEE Access (2021). https://doi.org/https://doi.org/10.1109/ACCESS.2021.3129329, publisher Copyright: Author

[15] OpenAI, Achiam, J., Steven Adler, e.a.: Gpt-4 technical report (2023)

[16] Ouyang, L., Wu, J., Jiang, X., Almeida, D., Wainwright, C.L., Mishkin, P., Zhang, C., Agarwal, S., Slama, K., Ray, A., Schulman, J., Hilton, J., Kelton, F., Miller, L., Simens, M., Askell, A., Welinder, P., Christiano, P., Leike, J., Lowe, R.: Training language models to follow instructions with human feedback (2022)

[17] Press, O., Zhang, M., Min, S., Schmidt, L., Smith, N.A., Lewis, M.: Measuring and narrowing the compositionality gap in language models (2023)

[18] Schick, T., Dwivedi-Yu, J., Dessì, R., Raileanu, R., Lomeli, M., Zettlemoyer, L., Cancedda, N., Scialom, T.: Toolformer: Language models can teach themselves to use tools (2023)

[19] Wei, J., Tay, Y., Bommasani, R., Raffel, C., Zoph, B., Borgeaud, S., Yogatama, D., Bosma, M., Zhou, D., Metzler, D., Chi, E.H., Hashimoto, T., Vinyals, O., Liang, P., Dean, J., Fedus, W.: Emergent abilities of large language models (2022)

[20] Wei, J., Wang, X., Schuurmans, D., Bosma, M., Ichter, B., Xia, F., Chi, E., Le, Q., Zhou, D.: Chain-of-thought prompting elicits reasoning in large language models (2023)

[21] Yang, R., Song, L., Li, Y., Zhao, S., Ge, Y., Li, X., Shan, Y.: Gpt4tools: Teaching large language model to use tools via self-instruction (2023)

[22] Yao, S., Zhao, J., Yu, D., Du, N., Shafran, I., Narasimhan, K., Cao, Y.: React: Synergizing reasoning and acting in language models (2023)

[23] Zhou, X., Jain, A., Phoha, V.V., Zafarani, R.: Fake news early detection: An interdisciplinary study (2020)