DeepTruth: Combining Large Language Models and Neural Networks Against Misinformation

Jakob Gürtler
gurtler.jakob@gmail.com
7032777

Ho-Hsuan Wang
hohsuann@gmail.com
7038925

Philipp Jonas Hawlitschek
phha00002@stud.uni-saarland.de
7043167

William LaCroix
williamplacroix@gmail.com
7038732

***Abstract*—This study explores the challenge of combating fake news, particularly within the political news landscapes of Taiwan and the United States. We present a prototype model that leverages Large Language Models (LLMs) combined with neural network classification heads to detect and classify misinformation. This model integrates sentiment scores, surprisal values from LLMs, and the final hidden state of the language model as feature embeddings. The model development focuses on three main areas: (1) Creating an Effective Neural Model, (2) Engaging Users through Crowdsourced Insights, and (3) Implementing Data Collection Mechanisms. We evaluate the performance of various LLMs, including LLaMa 2 7b, Gemma 2b, and BERT, across different neural network classification heads (SLP, MLP, LSTM, and CNN) using two datasets: LIAR and Cofacts. Simpler architectures like SLP and MLP are effective for straightforward datasets such as LIAR, while more complex inputs, like Cofacts, benefit from advanced architectures, particularly CNNs. The hypothesis is that CNNs excel by convolving the LLMs' rich hidden features to extract key information and filter out noise effectively. Beyond model development, the paper discusses the broader implications of fake news, highlighting its amplified impact in the digital age, posing risks to social cohesion and democratic values. Finally, the paper advocates for a holistic approach that merges technological solutions with public education to better understand fake news dynamics and support the global fight against misinformation.**

***Index Terms*—Political Misinformation, Fake News Detection, Neural Networks, Large Language Models (LLMs)**

github.com/NN-SoftP

I. Introduction - Ho-Hsuan

Fake news is not a modern-day construct but a longstanding element of human communication intricately woven into the fabric of society. Its historical roots trace back to ancient times when misinformation was strategically employed for various purposes, including political gain, warfare strategies, and social manipulation. In the contemporary world, the proliferation of digital platforms has exponentially amplified the reach and impact of fake news, making it a potent tool for many agendas, from influencing elections to swaying public opinion on critical social issues. Lazer et al. [1] define fake news as fabricated content that resembles the format of news media but lacks its editorial integrity and intent, serving as a subset of misinformation that particularly threatens the credibility of legitimate news sources.

The intricate relationship between truth and falsehood in news articles is intriguingly encapsulated in the quote from one of China's Four Great Classical Novels, *Dream of the Red Chamber*, written in the mid-18th century, "假作真時真亦假，無為有處有還無。 When false is taken for true, true becomes false; If non-being turns into being, being becomes non-being." This quote highlights the elusive boundaries between truth and deception that resonate with the essence of misinformation and offers a timeless reflection on the challenges of fake news in distorting our perception of reality in today's internet and information age. When truth can be mistaken for falsehood and vice versa, we risk destabilizing the shared values essential for a healthy and prosperous society. The erosion of these shared values can lead to a fragmented society where misinformation thrives. As Lazer et al. [1] have highlighted, this is particularly concerning as even if fake news does not change most people's beliefs, it can still reinforce pre-existing misconceptions, amplify their significance, or influence the media landscape, thereby exerting detrimental effects on societal well-being.

The impact and consequences of misinformation on our societies were particularly evident during the COVID-19 pandemic, when misinformation regarding the virus, treatments, and vaccines proliferated on social media, leading to widespread confusion, fear, and misguided actions. For instance, false claims about the efficacy of certain drugs or the dangers of vaccines have undermined public health responses and endangered lives [2] [3]. Fake news has also demonstrated its power to sway stock market movements and influence investment decisions, showcasing its capacity to cause significant economic disruptions [4]. This situation underscores the urgency for robust fact-checking mechanisms, especially in an era where advancements in Natural Language Processing (NLP) offer new frontiers in detecting and countering misinformation. However, the human element remains crucial, as individuals' discernment and critical thinking skills are essential in navigating the complex landscape of modern information. False news propagates more quickly and exten-

sively among people due to its usually unexpected, negative, and sensational nature, and this phenomenon is exaggerated for false political news more than other news topics [5]. Political fake news can profoundly impact society with its divisive nature and erode trust in democratic institutions [1].

Addressing the challenge of fake news is as crucial as enhancing technological security. In the battle against the tide of misinformation that threatens to distort our perception of reality, the combined efforts of technology, education, and critical engagement with information stand as our most effective defense. In light of these considerations, this paper presents a prototype of a model that can classify fake news using LLMs and Neural Networks with an interactive user interface. Our objective in developing this initial model and platform is to explore the possibility of combining technology with public collaboration to combat the spread of fake news. While the effectiveness of this approach remains to be assessed, our work lays a foundation for future research and development.

II. Social and Cultural Background - Ho-Hsuan

Misinformation is a major challenge globally, and it affects not only democratic nations but also those under totalitarian rule. In the latter, governing bodies increasingly use the term 'fake news' to legitimize censorship measures and suppress dissent in order to consolidate control over public discourse. At the same time, political parties in democratic societies are known to weaponize misinformation to manipulate public opinion in their favour and serve their strategic interests. This manipulation of information undermines the integrity of democratic processes, erodes trust in public institutions and media, and complicates the political landscape as it becomes increasingly difficult for people to discern truth from fabrication. This phenomenon influences electoral outcomes and public policy debates. This section of the paper examines misinformation in two countries that have struggled with fake news and its impact on their respective political climates. By reviewing these case studies, we aim to shed light on the multifaceted nature of misinformation and its consequences for democratic values and governance.

The spread of false information during important electoral periods is causing significant concerns worldwide, such as the recent presidential election held in Taiwan on January 13, 2024, and the upcoming US presidential election in the end of the same year. These elections are pivotal not only for these countries but also for their substantial influence on global geopolitics and the future course of international relations. Taiwan's election, in particular, has been closely observed due to its implications for cross-Strait relations, the U.S.-China dynamic, and broader regional peace and stability. Following the election, Taiwan elected Vice President Lai Ching-te from the Democratic Progressive Party (DPP), marking a continuation of the party's presidency but with a reduced majority compared to previous elections. This outcome reflects not only domestic political dynamics but also Taiwan's strategic geopolitical position amidst escalating tensions between the United States and China [1] [2].

The upcoming 2024 US presidential election is also taking place amidst various challenges related to misinformation and its possible impact on both domestic and international affairs. The result of this election will play a significant role in shaping global politics, including the US relationship with Europe, the strategic competition with China, and efforts to tackle climate change. The stance taken by the incoming administration, whether it is a unilateral or collaborative approach, could lead to a redefinition of alliances, with the possibility of a Republican presidency adopting more aggressive policies against China, while a Democratic administration may focus on engaging allies and promoting climate initiatives [3]. The emergence of AI-generated falsehoods in the US in the recent years also compounds the challenges with misinformation in the US political landscape. This adds a new layer of complexity that could make misinformation even more pervasive and persuasive. As Brookings highlights, integrating AI into the misinformation ecosystem could alter how falsehoods spread and erode trust in the democratic process [4].

1) Taiwan: Chang et al. [6] investigated the online misinformation landscape in Taiwan during the 2020 Presidential Election, specifically on Line, Twitter (now X), and PTT Bulletin Board System [5]. During the election, there was a high level of online engagement in misinformation, particularly targeting President Tsai Ing-wen. Analysis revealed strategic coordination among Chinese users, who selectively discussed certain topics (e.g., China, Hong Kong, President Tsai) while avoiding others (e.g., COVID-19), indicating a targeted misinformation campaign. The study also highlights political bias in Twitter's handling of misinformation, where efforts to combat misinformation might be biased or favor one political party over another. From the observations highlighted above, it's evident that digital platforms can easily shape political discourse, presenting challenges in navigating the spread of misinformation while striving to preserve democratic values and free speech in the digital era.

A study conducted by Tai-Li Wang [7] on the effect of fake news on Taiwan's 2018 local elections found that disinformation has had a significant impact on voter behavior. Over 50% of voters cast their votes based on incorrect campaign news. Politically neutral voters, who tend to support the China-friendly Kuomintang (KMT) party, and certain demographic groups such as younger voters, females, and lower-income individuals showed lower discernment in identifying fake news. The study points out that selective news exposure and the spread of fake news from Chinese sources or China-friendly

[1] https://www.csis.org/analysis/taiwans-2024-elections-results-and-implications

[2] https://www.brookings.edu/articles/the-impact-of-taiwans-election-in-2024-and-beyond/

[3] https://ecfr.eu/publication/brace-yourself-how-the-2024-us-presidential-election-could-affect-europe/

[4] https://www.brookings.edu/articles/misunderstood-mechanics-how-ai-tiktok-and-the-liars-dividend-might-affect-the-2024-elections/

[5] https://en.wikipedia.org/wiki/PTT_Bulletin_Board_System

media within Taiwan have contributed to societal divisions. These findings highlight fake news's challenges to democracy and call for urgent measures to improve media literacy and counter foreign influence campaigns.

2) The United States: Misinformation has become a major concern in American politics and media since the 2016 election. Allcott et al. [8] reported that Americans encountered an average of one to three false news stories in the month before the 2016 presidential election, and this number is likely underestimated. A 2019 Pew Research survey found that 50% of Americans consider misinformation to be a significant issue, and 67% believe it confuses the public about basic facts related to current events [6]. Furthermore, the term "misinformation" has been used in American TV news stories 6,071 times in 2019, which is a 323% increase from the 1,875 times it was used in 2015, according to the Internet TV news archive [9].

Among papers that discuss fake news in the US, Oehmichen et al. [10] focused their studies on the political misinformation spread through tweets related to the 2016 U.S. presidential election. The study revealed that accounts spreading misinformation displayed certain traits, such as having fewer followers, following more accounts, being newer, and favoring other tweets more frequently. According to the study, there are distinct differences between misinformation and trustworthy information. Misinformation tends to use more exclamations, capital letters, and digits. It also evokes less joy and trust but more surprise. These differences suggest that misinformation exploits human psychological biases such as reciprocity and confirmation bias. Reciprocity in this context refers to the tendency of social media users to engage with or share content from accounts that interact with them, a strategy used by misinformation spreaders to increase their content's reach. Confirmation bias is the inclination to favor and share information that aligns with one's preexisting beliefs. Misinformation creators exploit this tactic to ensure their content is readily accepted and disseminated within ideologically similar groups. Additionally, the emotional content of misinformation varied, indicating an attempt to polarize, and it was more frequently preferred, implying a potentially wider reach. This research highlights the sophisticated strategies used to spread misinformation and emphasizes the need to understand and counter these tactics to protect democratic integrity.

III. Theoretical Framework - Ho-Hsuan

The traditional method of manually fact-checking information to combat fake news is becoming increasingly inadequate due to the sheer volume of content generated daily. As a result, researchers are exploring deep learning as an alternative strategy for detecting fake news. This involves using NLP techniques, such as word tokenization, vectorization, and feature extraction, along with deep learning architectures like Feed Forward Networks (FFN) and Recurrent Neural Networks (RNN) utilizing Long Short-Term Memory (LSTM) cell layers and Convolutional Neural Networks (CNN). According to Mridha et al., advanced deep learning methods like Attention mechanisms, Generative Adversarial Networks (GANs), and Bidirectional Encoder Representations from Transformers (BERT) offer superior performance over conventional machine learning techniques in efficiently detecting fake news [11].

Kozik et al. [12] conducted a study to evaluate the effectiveness of state-of-the-art models for detecting fake news, using advanced natural language processing techniques on benchmark datasets with metrics such as precision and F1-score. The study found that LSTM and BERT-based models were the most effective due to their ability to understand complex language and sequences. However, classical machine-learning approaches were still relevant for simpler datasets. The research revealed challenges such as dataset bias and the ever-changing nature of fake news, which necessitates models that can adapt and generalize across various contexts. It also emphasized the importance of explainability, the integration of AI with human judgment, and ensemble methods for improving detection accuracy and adaptability in combating fake news.

Mahmud et al. [13] compared Graph Neural Networks (GNNs) with traditional machine learning algorithms (SVM, logistic regression, decision tree, and random forest) for identifying fake news on social media. They found that GNNs, particularly GraphSAGE and GCN, outperformed traditional algorithms using social data's graph-structured nature. By integrating textual content and graph-based data, these models achieved superior accuracy in identifying fake news. The study also highlighted the importance of including user engagement and social context in the analysis. They suggest that expanding datasets across languages and platforms and exploring real-time applications can further enhance the authenticity and reliability of news on social networks.

Alonso et al. [14] explore the use of sentiment analysis (SA) in detecting fake news and discuss its role as a central component or a supplementary feature in detection frameworks. It reviews the evolution of SA from lexicon-based methods to advanced deep learning techniques and how they are integrated into fake news detection strategies. They recognize the biases that are inherent in SA methods, which can come from data representation, algorithmic preferences, cultural, linguistic, or gender-based variations. These biases can compromise the fairness and accuracy of SA applications. While SA significantly contributes to enhancing detection efficacy, its effectiveness varies across datasets and methodologies.

Liu et al. [15] introduce TELLER, a framework that detects fake news by combining human expertise with large language models (LLMs). It introduces a cognition system to guide LLMs and a decision system to derive transparent and controllable logic rules. TELLER uses structured Yes/No questions and a neural-symbolic model for news authenticity evaluation, allowing human intervention to refine decision-making. It outperforms existing methods, offering transparent decision-making, strong generalization, and reliability through controllable rule adjustment. However, the complexity of

[6]https://www.pewresearch.org/journalism/2019/06/05/many-americans-say-made-up-news-is-a-critical-problem-that-needs-to-be-fixed/

TELLER's cognition system may pose usability challenges. Integrating human expertise and LLMs requires significant user understanding, and human intervention for logic rule adjustments may also introduce subjectivity.

AlEsawi et al. [16] suggest a deep-learning approach to detect Arabic misinformation. The approach uses AraBERT and BiLSTM with an attention mechanism to improve accuracy. The model uses AraBERT for feature extraction and a BiLSTM network enriched with an attention mechanism to better understand contextual nuances. The input text is tokenized via BERT Tokenizer, and the model processes it into contextual embeddings. The attention-enhanced BiLSTM model outperformed other models, achieving up to 96% accuracy for the ArCovid19-rumors dataset. The study demonstrates the potential of integrating BERT-based models with BiLSTM and attention mechanisms to detect non-English misinformation.

Liu et al. [17] have introduced a different approach for enhancing news credibility assessment called EKNet. It combines a knowledge graph and entity ontology to evaluate news articles comprehensively and accurately pinpoint inaccuracies. Tests show it outperforms traditional methods, offering a more accurate and reliable way to assess news credibility, especially in the context of fake news on social media. However, EKNet's reliance on complex entity ontologies and knowledge graphs may present usability challenges for users without technical expertise in these areas, potentially limiting its accessibility. Additionally, the intricate setup and maintenance of the entity ontology repository could pose difficulties for real-time news credibility assessment, affecting its practicality for immediate application in dynamic news environments.

Babar et al. [18] propose a real-time fake news detection model that utilizes big data analytics, a DNN framework, and a novel N-gram and LSTM hybrid approach. The system can efficiently identify true and false news items by leveraging datasets from Zenodo and Kaggle containing over 72,000 news items, achieving over 95% accuracy. The hybrid model uses Apache Spark for parallel and distributed computing, significantly improving the detection process's speed and accuracy. The results demonstrate significant improvements in accuracy, recall, and computational efficiency. However, the model's reliance on complex computational resources may limit its applicability in environments with limited technological infrastructure.

Despite numerous studies claiming near-perfect accuracies in detecting fake news, misinformation remains pervasive. This disconnect suggests that the academic pursuit of high accuracy, while important, does not fully address the complexities of misinformation in the real world. Recognizing this, we aim to bridge the gap between academic research and practical application by emphasizing usability and accessibility in our model development. Additionally, the common reliance on benchmark datasets for training and evaluation, while beneficial for standardizing results, may compromise the models' relevance due to the static and potentially outdated nature of these datasets.

In response, our strategy focuses on creating user-friendly fake news detection models and intuitive applications, broadening access to detection technologies. To enhance the sustainability and adaptability of our approach, we propose a straightforward method for dataset collection that combines user contributions into Excel sheets. This approach hinges on community engagement, where the active participation of informed users enriches our data repository, fostering ongoing model refinement and development. Our objective is to establish a framework that not only makes effective fake news detection tools accessible but also ensures their relevance and effectiveness over time through continuous data collection and user involvement.

IV. Model Development and Evaluation

A. Model Conceptualization - William

Our primary objective was to develop a robust model capable of accurately detecting false or misleading information within a given text. This involves not only identifying "pants-on-fire" falsehoods, but also learning the patterns that result from more subtly misleading information which may not be immediately apparent to the average reader. To achieve this, we leverage the knowledge base of several pretrained Large Language Models (LLM) to harness their ability to analyze textual content.

A successful implementation of these models has the potential to advance the domain of misinformation detection, and we anticipate that our models will demonstrate capabilities at or above current state of the art methods for classifying information as legitimate or false.

At the heart of this approach is the utilization of LLMs as the basis for classification. The rationale being that an LLM's vast "knowledge" is built on a wide foundation of publicly available information. This "knowledge" was acquired through extensive pre-training on massive amounts of textual data, granting these models a rich understanding of linguistic nuance, context, and patterns. It is this "knowledge" that we seek to tap into, as it may provide a layer of insight instrumental in distinguishing between legitimate and misleading information.

In this section, we dive into the integration of pre-trained LLMs within our training framework, highlighting the multiple ways our approach harnesses the vast knowledge buried deep within these models. Pre-trained models, such as BERT, offer a wide breadth of encoded linguistic information, gleaned from vast amounts of training data. We seek to build on that knowledge, and leverage this pre-training in 3 distinct ways:

1) Word feature embeddings from the language model's last hidden state.
2) Sub-word surprisal calculated via logits from the language model's output.
3) Data augmentation deriving statement sentiment scores from a separate pre-trained, fine-tuned BERT model for sentiment analysis.

We aim to enrich model training, enhance understanding, and improve accuracy of fake news detection by tapping into the underlying strengths of these pre-trained language models.

In order to capture the content of the input statement in a way that can be fed into the network, the words need to be first tokenized and then vectorized into unique numerical representations. Depending on the language model being used, the matching tokenizer splits the input into (sub-)word input IDs to serve as integers indexed to the model's vocabulary size. This serves as a point of distinction between language models, both in the manner of tokenization as well as the size of the model's vocabulary. While each of these models uses sub-word tokenization to represent complex language, they differ in the number of unique tokens they are able to encode. The BERT tokenizer has a vocabulary size of 30,522 unique tokens, Llama's uses 32,000, while Gemma uses the Gemini tokenizer, containing an incredible 256,128 unique tokens [19] [20] [21]. Even though the Gemma models are trained and designed to generate English text, this large multilingual vocabulary size from the base model allows Gemma models to uniquely identify many more tokens, potentially reducing the tokenized length of the input, this comes at a significant cost: in the case of Gemma-2B, a vocabulary size of over 250k, coupled with an embedding dimension of 2048, means that over 26% of Gemma-2B's 2 billion parameters are taken up by word embeddings alone, compared with 14.5% of BERT's 340 million parameters, or just 1.8% of the parameter space in Llama-2-7B model. Having such a high portion of the parameters taken up by word embeddings means that Gemma-2B can struggle with generalizability, since the embedding layer takes up a disproportionate portion of the model. This can lead to poor generalizability of the model, due to sacrificing attention layers and other calculations for embedding layer size. One potential upside of the Gemini vocabulary size is training speed: due to the increased vocabulary size, the input length of statements into the Gemma model will be shorter than that of models with smaller vocabulary sizes, resulting in fewer calculations performed. Whether this representational efficiency is beneficial for interpreting the subtleties of the statement texts remains to be seen.

Leveraging these transformative capabilities, we were able to use the final hidden state of the language model as word embeddings, getting language model-specific word representations. This approach marks a significant improvement over traditional static vectorization techniques, such as Word2Vec [22]. Unlike static embeddings, which provide singular, context-independent representations of the input sequence, a LLM's word embeddings capture the context and use of a given token dynamically. This means a single token will have different embedding representations depending on the surrounding context, and can vary significantly depending on the words that precede it, providing more accurate, contextualized, and relevant representations of a given input sequence. This inherent adaptability provides flexibility in a language model's ability to perform various downstream tasks, by alternately freezing or fine-tuning the hidden states of the underlying language model, in addition to the classification head. We will go into more detail on the specifics of this in section IV-C, *Model Architecture and Pipeline*. This dramatically expands our toolkit in terms of ability to optimize model performance and training, depending on the task. At their core, these word embeddings encapsulate what a language model "knows" about the content it processes. Moreover, given the dynamic and context-aware nature of these embeddings, the model is able to parse not only the literal text of an input statement, but also capture the semantic nuances that underlie the text. This provides a capacity for the language model to interpret and analyze input at a profound level, and signifies a paradigm shift in how we approach symbolic representation of language in computational models, moving away from the limits of traditional static vectorization schemes in favor of more dynamic, contextually informed representations of the text.

In the model pipeline, the tokenized input IDs serve as the primary input to the underlying base LLMs. In CausalLM mode, input to the base language model optionally returns the token logits at each step, as a probability distribution across the model's vocabulary. This is done internally by offsetting the input sequence by one against itself and calculating the probability distribution across the vocabulary for the next token, given the preceding context. Taking the negative log of the probability of the actual next token results in surprisal values for each token in the input sequence, which can be stacked on the word embeddings as additional input features. Our reason for including surprisal as an input feature for classification is the hypothesis that fake news will contain less likely sequences of words, resulting in higher surprisal values. Deceptive statements are then anticipated to display higher surprisal values, indicating their deviation from the predicted word sequence. This may manifest as distinct patterns in the surprisal values across input statements, leading to learnable indications of fake or deceptive content. The hope then is that either the magnitude and/or the distribution of surprisal values within input sequences will allow our networks to reliably predict false or misleading content.

B. Data Collection and Preprocessing - Jakob

Although the initial idea of this project was to develop a fake news detection model specifically to combat misinformation in Taiwan, we collected both English and Taiwanese data. This is largely owed to the fact that fake news detection research for English is much more prevalent. Thus, adequate benchmark datasets exist, and most established models have been thoroughly tested and compared. So, for a quick start in development, we used a well-known English dataset, namely the LIAR dataset [23], and later on applied our models to a newly comprised dataset for Taiwanese fake news, the Cofacts dataset as we call it.

1) LIAR Dataset: While many fake news detection benchmark datasets exist for the English language, one that has been widely used in previous research is the LIAR dataset [23]. About a fifth of the papers reviewed for this project used this dataset out of a total of ten different fake news detection datasets. So, LIAR was chosen for its wide range of comparisons. The dataset was collected in 2016 and contains

about 12.800 short statements, usually about one sentence, labeled for their truth value. The statements span a time frame of about one decade, from 2007 to 2016, and were collected together with their labels from PolitiFact.com[7] [23], which is a fact-checking service focused on English news from the United States. PolitiFact.com uses six labels for annotation, which are *true*, *mostly-true*, *half true*, *mostly false*, *false* and *pants on fire*. The labels roughly correspond to factual accurate, accurate but needs further clarification, partially accurate but out of context, contains some truth but omits important information, inaccurate and outright ridiculous [24]. Labels are assigned by editors on the platform according to general guidelines and then reviewed by two additional independent editors. All three editors finally vote together on a majority label [24]. In the LIAR dataset, the *mostly false* and *pants on fire* labels are replaced with *barely-true* and *pants-fire*, respectively. Statements in the LIAR dataset are roughly equally distributed across the six different categories, with the number of statements ranging from 2,063 to 2,638 except for the last category *pants-fire* with only 1,050 statements [23].

The LIAR dataset is focused on political news with many statements from politicians as well as social media. Included is a lot of meta-information besides the statements and their labels, such as the speaker's name, job title, political party affiliations, home state, as well as the subject matter, and a short description of the circumstances the statement was made in [23]. The dataset is also available through Hugging Face[8]. However, the version on Hugging Face splits one original column of the dataset into five separate columns, thereby omitting some information. The authors of the LIAR dataset originally added one column that contained information they called credit history, a vector of counts for each of the six categories of how many historical statements of the respective speaker were assigned to a certain category [23]. This vector was split into separate columns, but no column corresponds to the historical *true* counts. In our case, it is no problem that this column is missing since we set our focus on content-based fake news detection anyway and thus only require the statements and their labels.

To sanity check our models' capabilities, we resampled the LIAR dataset once by splitting off a completely different portion of the data to use as a test and validation set than the originally predefined splits. We also resampled a smaller test and validation split at about 25% of the original size, but preliminary experiments in both cases did not increase performance at all, so we used the LIAR dataset without any additional modifications. This at least preserves the functionality of the dataset as a benchmark.

2) Cofacts Dataset: While fake news data for English is readily available in the research community, this is not the case for fake news data from Taiwan. Especially since the problem of Taiwanese fake news is often lumped into one task with the detection of Chinese fake news. However, between these two different cultures, the perception of some news as true or fake may vary [25]. It is important to distinguish Taiwanese and Chinese fake news in particular, as they are in political opposition and tension is high between the two countries II-1. Therefore, we aimed to collect news and annotation data exclusively pertaining to Taiwan. Luckily, several services around the world provide fact-checking and review of news articles submitted by their users. One such service for Taiwan specifically is Cofacts[9], a completely open and crowd-sourced platform for information checking. It was first created as part of the g0v initiative[10], a community effort with the core values of collaboration, open-source, and improving the status quo [26]. Figure 3 shows an overview of Cofacts working structure [27]. Users interact with the service via a chatbot integrated into popular messaging services. A piece of news can be sent to the bot to receive fact-checking services. If the piece of news matches a previously checked item and thus is contained in the Cofacts database, the user can receive instant feedback. If the news is not known to Cofacts already, it is posted to their fact-checking space, where it can be picked up by an editor who compiles a reply with fact-checked information according to their general guidelines [27]. The reply is then sent to the user and saved in the database together with the original article. Additionally, editor replies are routinely reviewed by other editors, often junior editors, to ensure quality and correctness [28]. Since Cofacts also offers its users the opportunity to provide feedback on the initial editor reply, the process of quality assurance can efficiently be sped up by assigning reviewers to those replies that received unsatisfactory feedback first [27].

Cofacts hosts an up-to-date version of their database publicly available on Hugging Face[11]. Given one does not violate the terms of use, anybody can get access to these anonymized samples. We used this data to build our datasets for the fake news detection task.

The original dataset is available in the form of a group of large data frames that each represent different processing stages according to the Cofacts pipeline described above or that provide additional useful information. The different dataframes are linked by article IDs so that they can be joined together according to one's needs. The base dataframe is the *articles* data frame, which contains about 154,000 articles as they were submitted by Cofacts users. Articles are divided into text, audio, and video, with 82.2% being text-based articles. While video and audio-based articles only have a link to their sources stored as content, text-based articles are stored with their whole article body in the data frame. This is a major difference from the LIAR dataset as the average utterance length of the Cofacts data is much longer than the short statements from the LIAR dataset. Additional information includes whether the article is still available or has been blocked, as well as the date of submission, how

[7]https://www.politifact.com/

[8]https://huggingface.co/datasets/liar

[9]https://en.cofacts.tw/

[10]https://g0v.tw/intl/en/

[11]https://huggingface.co/datasets/Cofacts/line-msg-fact-check-tw

many times it has been submitted, and some anonymized meta-information about the user and channel of submission. All the publicly available articles were submitted fairly recently between 2020 and 2024, which is another difference from the older LIAR data. An extension of the basic data frame is the *article_categories* data frame that holds about 124,000 articles categorized by subject, which Cofacts distinguishes between a total of 21 different subjects. These subject categories are not exclusive, so one article can be assigned to multiple subjects at the same time. The categories include subjects such as medical, international, COVID-19, political, and many more. Many categories are only represented by IDs, but Cofacts also provides a table to map these IDs to legible category names. The last data frame of interest to us is the *article_replies* data frame, which contains editor replies to about 90,400 articles. Besides writing a textual reply corresponding to the article, editors also label the original article as one of four categories: RUMOR, NOT RUMOR, OPINIONATED, and NOT ARTICLE. The first two categories divide the articles into factually correct and incorrect according to the editor's research and judgment. The third category is for articles that might not be factually incorrect but exhibit strong biases that could cause the reader to understand the article in a potentially misleading way. The last category is assigned to articles that can not be processed or labeled by Cofacts for various reasons [27].

Preprocessing was applied after the three data frames above were joined together on the article IDs. Since the current work is focused on political fake news, the data was filtered for political news first, only including three categories from the *article_categories* data frame, namely 台灣政治、政黨, 中國政治宣傳and 跨國互動, which translate to Taiwan politics and political parties, Chinese political propaganda and international relations. These three categories account for about 22% or 27,280 items of the original *article_categories* data frame. All non-text-based news was also dropped, as we are only concerned with the language modality. Additionally, we excluded articles labeled as NOT ARTICLE or OPINIONATED. Articles in the first category often did not have an actual text body included as content, and while articles in the second category might be technically correct, their truthfulness is debatable, and they are certainly not an example of a gold standard truth. Although these opinionated articles might not state any outright false information, facts can still be presented only in parts or out of context so that they paint a picture not aligned with reality. From manual observation of the Cofacts data, editors have to provide further clarifying references in their replies for most of these opinionated articles so that the user has access to all the necessary information for the topic at hand. At the moment, it is unclear how to deal with such half-truths, but given malicious intent, they can also be considered as a tool of misinformation [29]. Following our judgment, the category of OPINIONATED articles represents some kind of a gray area between fake and not fake, and further, more detailed annotations would be required to disentangle this category. Unfortunately, since we do not have the manpower, we decided to exclude these articles for now, reducing the dataset size by about 12%. A quick analysis of the remaining articles revealed that an unexpectedly high proportion of them had a length of around 28 characters. Visual examination of these items revealed that they only contained URLs despite being labeled as text by Cofacts. These items are not fit as input to a context-based detection model, so they were also excluded. Furthermore, we removed URLs within texts to prevent the model from becoming sensitive to the article sources and instead solely relying on the textual contents for detection. Lastly, we set a minimum and maximum article length of 10 to 2000 characters to ensure that the model has at least some meaningful context in the input while also removing items with excessively large article bodies to keep down processing time. The length requirement ruled out about 400 additional items, which were less than 7% at this point. A large majority of the excluded items at that time were too short, mostly due to the previously applied text processing. In the final step, we sampled a train, validation, and test set with splits of 80%, 10%, and 10%, respectively. We made sure to keep the original label distribution across all three splits. The resulting dataset has a training split of about 5.000 items and a validation and test split of 629 items each, which is roughly half the size of the LIAR dataset.

While removing URLs, we also experimented with other text processing, including removing emojis as well as characters and punctuation foreign to Taiwanese, as the data submitted by different Cofacts users varied widely in formatting as well as the inclusion of words or utterances borrowed from other languages such as Japanese, Korean or English. This did not come without considerable difficulties since the different punctuation and writing systems often mixed different Unicode blocks used to represent them. In order to preserve most of the information present in the data we would have to substitute and replace in some cases while removing punctuation in others to hopefully gain a marginal performance improvement. For this reason, we decided against extensive text processing in this manner since we use a large foundational language model with multilingual capabilities [20] that should well be able to process texts with intertwined languages. By removing those parts, we would instead be obfuscating important information and would thus most likely hurt performance rather than benefit it. Additionally, in the case of emojis, some large language models have demonstrated a very good understanding of the symbolic representations [30], which is why we decided to preserve them as well.

The final Cofacts dataset has a label distribution of 83.2% RUMOR or fake and only 16.8% NOT RUMOR or true. Already, the original *article_replies* dataset is highly imbalanced with 52.4% fake and 18.4& true. Considering the service Cofacts provides, it is much more likely that users submit news that turns out to be fake, as most users are probably more inclined to report news that appears fake but not news that they regard as obvious truths. Some initial filtering is already happening on the user side that we can not control. This assumes some competence already on the user's side to

distinguish fake from real news. On the other hand, we might assume that users are so naive that they can not distinguish fake from true news at all and just submit any article they come across. This would be the ideal scenario for us, but the relatively low proportion of true news in the dataset seems to suggest otherwise. In any case, the severe imbalance in our dataset is something to be aware of. One adaptation to deal with this is to use a weighted binary cross-entropy loss given by Equation 1, which adds a weight w_n for each training item n in the samples N depending on the target y_n to modulate the learning effect of a prediction x_n.

$$L_{bce} = \frac{1}{N}\sum_{n=1}^{N} -w_n[y_n \cdot \log \sigma(x_n) + (1-y_n) \cdot \log(1-\sigma(x_n))] \tag{1}$$

Since, in our case, the majority class is fake, we use the ratio of $\frac{N_{true}}{N_{fake}}$ as the first weight and 1 as the second weight. These weights are combined in a vector and broadcasted with the targets, such that the predictions for the majority class, i.e., fake news, are weighted down, while the predictions for the minority class, true news, are preserved:

$$w_n = \begin{cases} \frac{N_{true}}{N_{fake}} & y_n = fake \\ 1 & y_n = true \end{cases}$$

As an additional measure, we resampled different versions of the Cofacts data and reran our experiments on each of them to compare which dataset serves best to train a classification model. First, we resampled a dataset with a balanced distribution of an equal number of true and fake news. This dataset ended up relatively small at 1,694 trains and 212 validation and test items. Second, we augmented the dataset with additional news articles labeled as true, extracted from the MCFEND dataset [31]. This dataset is a very recent dataset for Chinese fake news. The authors collected news from many different fact-checking agencies, including agencies from China and Taiwan, as well as international agencies [31]. To select high-quality items, we only selected items with an article title and body as well as a label and an ID. We concatenated the article title and body to adapt to the format of the existing Cofacts dataset. After dropping duplicates, only 961 of the originally 6,074 true items [31] remained. Adding these to our original Cofacts dataset changed the ratio of true and fake news from 16.8% to 27.9% and 83.2% to 72.1%, respectively. The augmented dataset ended up at a size of 5,798 training and 725 validation as well as test items.

As discussed previously, the cultural and social pressures on Chinese and Taiwanese fake news might differ extensively, and it is not clear how the adaptation of news articles from a different domain than the locally sourced Cofacts articles will change the dataset and impact a model's capability to extract spurious relations. However, the MCFEND articles are fact-checked by many different agencies, including agencies from Taiwan [31], which we would assume to align closely with the data labeled by Cofacts. Unfortunately, the data was not annotated for sources, so we were unable to filter the articles for fact-checking agencies. Still, articles and labels in the MCFEND dataset were also cross-checked with the translated English versions of the articles and the corresponding labels assigned by international fact-checking agencies [31]. Additionally, factual truths are only susceptible to cultural and social pressures to a limited extent. We thus assume the additional ground truth labels to be rather beneficial to model performance than harmful.

Finally, we also combined Cofacts with the LIAR data into a bilingual dataset to test model robustness on mixed data. For this purpose, we remapped the six labels of the LIAR dataset in the following manner *true* and *mostly-true* as *true* and *half-true*, *barely-true*, *false* and *pants-fire* as *fake*. Following the category definitions according to PolitiFact.com [24], *true* and *mostly-true* were grouped as both of these categories are assigned to statements that present factually correct information. The remaining categories were grouped into the *fake* category since they represent articles that are only partially true, omit facts, or are outright wrong and thus can be considered misinformation in our eyes. After remapping, the label distribution is somewhat similar to the label distribution of the augmented Cofacts dataset with 35.6% *true* and 64.4% *false* labels. To not increase the already severe imbalance, we also undersampled the modified LIAR dataset so that it contained 3,649 items for both categories. This dataset was then split into train, validation, and test sets and added to our original Cofacts dataset, resulting in a train set of about 11,600 items and a validation and test set of about 1,460 items, which is the largest dataset in this project.

3) Sentiment Scores - William: Data augmentation plays a crucial role in enhancing the performance of both machine learning and neural models. This is especially true in tasks involving NLP, as the variety and quality of training data greatly affects the models' ability to generalize to unseen test data. Previous research on fake news detection identified sentiment analysis to offer a considerable performance increase [14], [32]. Therefore, we chose to add sentiment scores as additional features to our datasets. Sentiment analysis was performed on both the LIAR dataset as well as the Cofacts dataset using the lxyuan/distilbert-base-multilingual-cased-sentiments-student fine-tuned model [33] via the Hugging Face Transformers library [34]. The output from this step is a group of softmaxed sentiment values for each item in the datasets. An example output may be represented by the following dictionary: {positive: 0.2, negative: 0.1, neutral: 0.7}. This output is concatenated to the language model output at different points in the forward pass, depending on model architecture (detailed in Section IV-C). In the case of the Cofacts data, articles were truncated to a length of 512 tokens as some inputs exceeded the context length of the model. Sentiment is used to supplement the word embeddings and surprisal score as an additional input feature. The hypothesis is that high sentiment scores, when combined with high surprisal scores and the contextual word embeddings will allow the models to predict fake or misleading statements with greater

accuracy than models which rely solely on word embeddings.

Using sentiment for fake news classification is not in itself a novel technique, Bhutani et al. also utilize sentiment analysis for classifying the LIAR dataset using hybrid CNN models, and while their results reach a high of 0.274 accuracy on the test set, the models are also trained on the metadata from the dataset, and not simply the statement text by itself [35]. This is the main point of difference that sets our approach apart from other state of the art classifiers, that being the use of text alone without additional metadata. As will be shown later in Table I, many of our models outperform these hybrid CNN models without including additional metadata, using only text input in the form of word embeddings, sub-word surprisal, and statement sentiment as input features; all of which may be derived from the input text alone, without additional contextual metadata, such as the speaker, history, subject, etc. Our hypothesis is grounded in the idea that the sentiment expressed by a statement can help provide clues to the truthfulness of the content. For example, statements that are designed to mislead or manipulate the reader may attempt to elicit a strong emotional response in order to obfuscate the logical leaps or misrepresented facts. By incorporating sentiment scores into our models, we can utilize these sorts of patterns and relationships to improve the model's predictive capabilities. Furthermore, the combination of sentiment scores and sub-word surprisal scores give an extra layer of context sensitivity in addition to that of the word embeddings, enabling the model to better represent the subtleties expressed by these sophisticated forms of misinformation, where the linguistic features of the words themselves may not be enough to identify false or misleading statements. This approach attempts to move beyond simply using semantically encoded word embeddings, and extend the feature space to include other pragmatically relevant information that may be discernible by the LLMs, and this ability to harness the black-box knowledge space of LLMs for downstream tasks beyond the original training is an exciting new frontier in linguistic research.

C. Model Architecture and Pipeline - William

Each model in this experiment shared a general architecture, differing in the structure of the classification head. All models were trained with the same features described in Section IV-A. The hidden features from the language model output were concatenated with surprisal values and then fed into the feedforward or recurrent layers for transformation before being concatenated with sentiment scores and passed through a final linear layer for classification. While sentiment scores were computed during preprocessing, surprisal values were computed directly from the base model outputs during training. The result is an input of (batch size) x (LLM hidden features) x (sequence length), and an output of (batch size) x (number of class labels).

For testing and comparison, we developed four distinct model architectures:

1) "Naive" single linear layer with mean-pooling
2) Informed Multi-Layer Perceptron (MLP) with mean-pooling
3) Informed LSTM
4) Informed CNN

Diagrams of the model architectures can be found in the Appendix 2.

The SLP model was the first model developed for prototyping, and serves as our neural baseline, as it is the simplest possible classifier model. Word embeddings from the language model are condensed across the input sequence via mean-pooling, into a single vector the length of the language model's embeddings. This serves both practical and theoretical purposes: practically, mean-pooling of the statement input allows variable-length input to be translated into a vector of constant size, alleviating issues of sequence length difference across or between batches. Theoretically, mean-pooling across the entire input sequence provides an abstraction over the average semantic content of the statement. It is unclear whether this abstracted representation maintains much of the nuance provided by the word embeddings, but as we will show later, even these simple models performed competitively during testing, in spite of their heavy abstraction.

The MLP models make two primary improvements over the SLP model: adding layers, as the name suggests, and introducing the data augmentation detailed previously in section IV-A. Mean pooling is still used to reduce the input dimensions and transform the sequence to constant length, but the sub-word surprisal values are additionally concatenated to the word embeddings before pooling. The mean-pooled inputs are then passed through hidden linear layers of progressively smaller size, before being concatenated with the sentiment scores and finally passed through the final linear layer for classification. The sentiment scores were concatenated after reduction in the hidden layers so that the high-dimensional (between 768 and 2,048) word embeddings would not massively outweigh the sentiment scores in the final classification calculation. Many iterations of these models were tested with various numbers and sizes of hidden layers, where the final model consists of a single condenser layer bringing the input down from the word embedding size + 1 (1 for the subword surprisal value) to a hidden size of 50, before being brought down to a final output size of 6 for the LIAR dataset, and 2 for Cofacts classification. LeakyReLU was chosen as the activation function, due to the relative simplicity and efficiency of the ReLU function, without the limitations of "dying ReLU problem" where significant portions of the network are reduced to zero and contribute nothing to the model's learning [36].

The LSTM models were the next step in model complexity, which took as input the full sequence of word embeddings along with sub-word surprisal, and projected them to a hidden layer size of 100. In order to extract a constant size output from the LSTM layer(s), only the final cell's hidden state is taken as a sequence summary, which is then concatenated with the sentiment scores and sent through a final linear layer for classification. The LSTM model used a medium dropout rate

of 0.5 to achieve the best results (more on dropout in the following section, *Training and Optimization*).

The CNN models then represent the most sophisticated of our model architectures, by incorporating the data augmentation techniques of the previous models, while also utilizing the full feature space of the input data by opting for input unstacking as opposed to mean pooling. As with the LSTM models, the word embeddings were concatenated with the subword surprisal values as extracted from the base LLM. This matrix was padded and fed to a 1-dimensional convolutional layer with a kernel size of 5. Max pooling was performed also with a kernel size of 5. The output went through a dropout layer with a value of 0.9, before being "unstacked" – that is, the pooled output rows were concatenated into a single series per input statement. This series was then concatenated with the sentiment values (as above), and fed through a final fully connected layer for classification.

D. Training and Optimization - William

As previously discussed, data augmentation has been the cornerstone of our models' robustness: by leveraging sentiment analysis and surprisal from pre-trained language models, we were able to enrich our data and allow our models to grasp a more diverse range of expressions and context, improving their generalizability.

Another crucial aspect of model training and optimization is hyperparameter tuning, involving picking the optimal values for parameters such as batch size, learning rate, and model depth and breadth. Batch size was limited primarily by hardware memory constraints: due to the size of the language model word embeddings, batch sizes of 32 were used initially, but this was later reduced to 16 in order to increase the layer size for the recurrent networks. When designing neural models, careful consideration has to be taken with regards to the depth (number of layers) and width (number of nodes per layer) of the architecture, since the number and size of layers has great effect on both capacity and complexity. For model complexity, a large increase in the number of calculations performed slows down training, while the additional weight and bias parameters increase the memory requirements by a factor of the width of the layer. At the same time, depending on the nature of the task and the distribution of the dataset, a model with excessive capacity may over-fit the noisy training data and fail to generalize to unseen test data. In particular, Zhang et al. demonstrate how increased feed-forward depth may not be beneficial to model performance on tasks involving learning long-term dependencies [37]. The LSTM models examined during experimentation used the final hidden state of the output for classification, and while this does not model long-range dependencies in a traditional sense, the models instead learn to predict based on patterns that may or may not exist in the input statements. This applies as well to CNN models, if not to the same degree. Anecdotally, during development and testing, our initial CNN models used multiple linear layers after the final pooling layer in order to reduce the output size before the final classifier layer, to provide a sort of down-stepping of the dimensions, but the model performance increased dramatically when the structure of the network was simplified to include only a single linear layer as classifier.

At first blush, this seems to be another straightforward example of the classic machine learning principle of the bias-variance trade-off. In fact, in all cases the models were prone to over fitting, after a number of epochs (largely determined by the learning rate), the training accuracy would continue to improve as the validation loss began to increase. This occurred in both simple and complex networks, suggesting network depth is not the only issue. The issue was partially eliminated by introducing a high amount (0.9) of dropout after the final convolutional layer in our CNN models, while simultaneously lowering the learning rate from that used for the more simple linear models. This again suggests very chaotic and noisy data distribution with a high degree of variance. At the same time, this complexity trade-off only explains the tendencies for models to over-fit, but it does not explain the phenomenon of models failing to make any predictions beyond the majority class. In development, there was a tendency for models with excessively high capacity to make no predictions at all, defaulting to the majority class. This does not appear to be an issue of vanishing/exploding gradients, as including sigmoid or other non-linear activation functions did not alleviate the issue. Instead reducing the number of layers was the solution to this issue. This suggests that the increased model complexity can totally frustrate a model's ability to make predictions. Srivastava et al. present a problem similar to the issue of vanishing/exploding gradients; that of diminishing feature reuse, where deep neural networks develop inefficiencies due to the depth of the network and the later layer's inability to learn from feature inputs [38]. However, in that case the authors were dealing with plain feedforward networks with depths between 5 and 10 layers, which is beyond the complexity of any of the models experimented upon in this project. Traditional methods of incorporating dropout layers, non-linear activation functions, batch normalization, or class-weighted loss calculations did not alleviate the single category predictions of our deeper networks, leaving this issue an open mystery.

Learning rate: as mentioned above, the learning rate of the models played a crucial role in their ability to predict test data. For the LIAR dataset, when the learning rate was high (1e-2), model improvement was sporadic and jumpy, where both training and validation loss would alternately increase and decrease by the epoch. A learning rate of 1e-3 turned out to be suitable for feedforward models, whereas the recurrent models saw the most improvement from a learning rate of 1e-4. On Cofacts data lower learning rates generally performed the best with 1e-4 returning the best results even for the simpler linear models. Tuning this hyperparameter was a balancing act between step sizes being so large that model performance bounced around, failing to reach convergence, and being so small that no progress was made. This was exaggerated in the case of the CNN models when the high dropout rate was introduced, a high learning rate was akin to

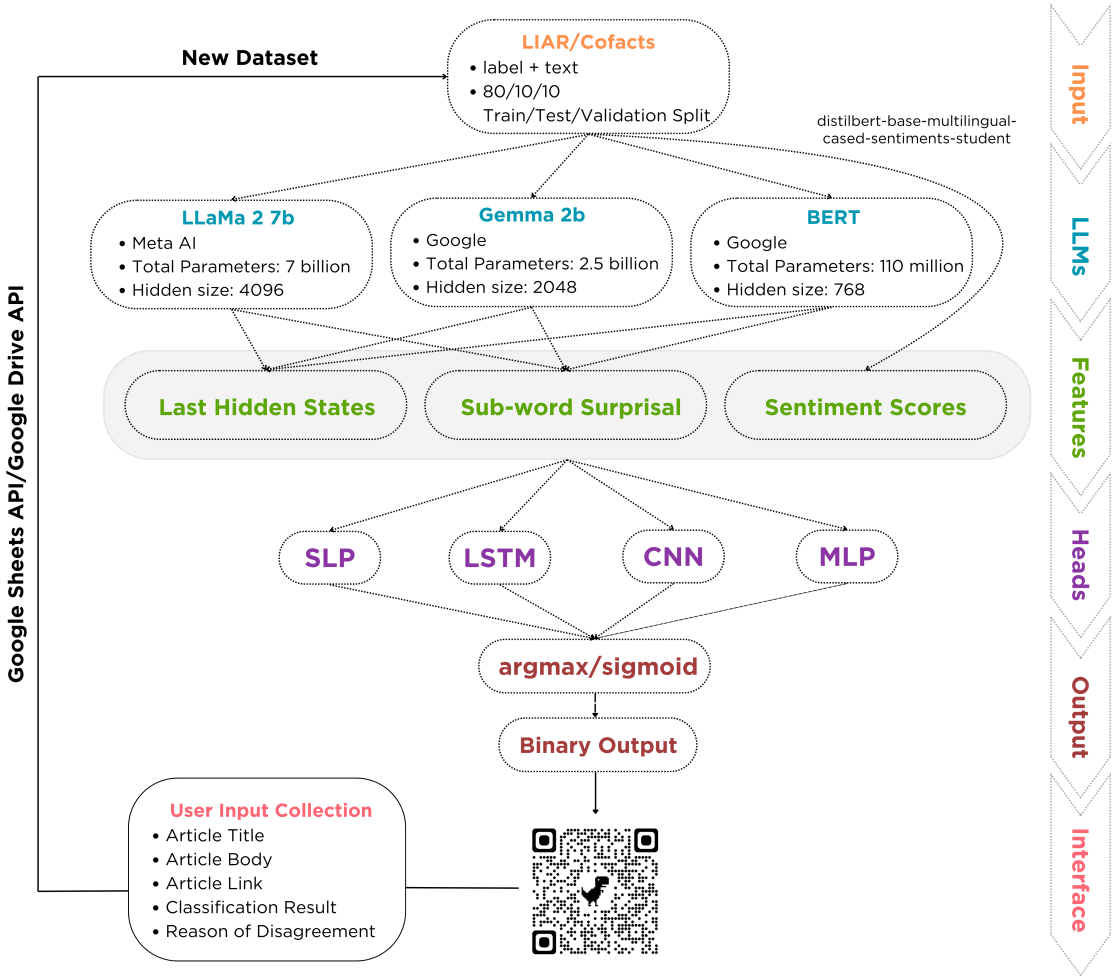

Fig. 1: Model Pipeline - Hidden features and surprisal values are extracted from one of three base LLMs (Llama 2, Gemma, and BERT), concatenated with sentiment scores, and fed through one of four classification heads (SLP, MLP, LSTM, and CNN), which assigns a label e.g. *true* or *fake*. Submitted articles, classification results, and user feedback are then collected to improve future models.

throwing the dice and hoping to land balanced on the edge, whereas a low learning rate of 1e-5 failed to take large enough steps that performance increased at all. In future work, these models could be extended to include adaptive learning rates via learning rate schedulers in order to benefit from larger early steps, and more subtle steps later in training.

E. Evaluation and Metrics - Philipp

1) Accuracy: The primary metric utilized for evaluating the model performance was accuracy, particularly in the case of models trained and tested on the 6-label LIAR dataset, which served as our initial dataset before expanding to different label distributions and the Cofacts dataset. Throughout the prototyping and hyperparameter tuning phases, our goal was to identify the most effective model within each architecture, assessing performance by accuracy on the validation set. Subsequently, the top-performing models underwent evaluation on the test set to obtain our final results.

Accuracy serves as a straightforward and accessible metric, offering a macroscopic estimation of a model's predictive performance by quantifying the ratio of correctly classified items versus the total number of classified items.

$$Accuracy = \frac{TP}{TP + FP + TN + FN} \tag{2}$$

Despite its utility, accuracy does not come without flaws, particularly in scenarios involving imbalanced datasets, where it may yield misleadingly high scores by favoring the majority class. To mitigate such issues and identify potential bugs or shortcomings, we conducted thorough analyses using confusion matrices, as depicted in the Appendix.

Additionally, we employed graphs showing the development of accuracy and loss during training. These visualizations enabled valuable insights into the progression of training over time, allowing us to identify for how many epochs the model demonstrated improvements in learning both the

training and validation datasets, as well as enabling us to detect notable oscillations, to gauge the point at which performance stabilized, and eventually when overfitting took place.

2) Precision: For the binary classifier, we also employed some more fine-grained metrics like precision, which measures the ratio of true positives versus all positive predictions. Optimizing for precision results in minimizing the number of false positives. In the context of fake news detection, this means that the majority of statements classified as fake are truly fake indeed.

$$Precision = \frac{TP}{TP + FP} \tag{3}$$

3) Recall: Measures the ratio of true positives versus all actual positive instances. Optimizing for recall results in minimizing the number of false negatives. For the fake news scenario, this means that as many statements as possible are correctly flagged as fake.

$$Recall = \frac{TP}{TP + FN} \tag{4}$$

4) F_1 Score: The F_1 score strikes a balance between precision and recall through the harmonic mean of both of these measures. Especially for the binary classifier, where this type of metric is easily applicable, it can be seen as a replacement for the traditional accuracy metric.

$$F_1 = 2 * \frac{Precision * Recall}{Precision + Recall} \tag{5}$$

5) Trade-Off between Precision and Recall: Achieving high precision and high recall simultaneously is essential for an accurate and well-performing classifier. However, in practical applications, attaining exceptional scores in both metrics simultaneously can be challenging. A guiding principle for determining the focus between precision and recall lies in evaluating the costs associated with false positives and false negatives. This assessment acknowledges the potential asymmetry between these costs, as they may have different implications and consequences in real-world scenarios.

In essence, high precision implies that the model must be highly confident when classifying an item as fake, maintaining credibility for both the model and the author of the statement in question. Conversely, high recall indicates that the model leans more towards freely classifying items as fake, thereby reducing the chance of misinformation spreading unchecked. This exhibits the possibility that an overabundance of "fake" labels may lower the trust in the model's judgments, especially if they contradict users' intuitions.

A possible solution for this dilemma in a real-world application might be to output the softmax of the predictions over the label distribution, effectively reflecting a measure of probability. This secondary and more graded output signals the model's level of uncertainty to the end-users, encouraging them to evaluate the model's judgments critically. Ultimately, this approach may enhance usability and trust by providing transparency regarding the model's confidence level for each item.

F. Results - Jakob

1) LIAR: We first tested the different model architectures devised in Section IV-C on the LIAR dataset described in detail in Section IV-B1. The four classification heads, SLP, MLP, LSTM, and CNN, were tested on three foundational LLMs: Llama 2 7B, Gemma 2B, and BERT. We tested each head with a frozen LLM to efficiently utilize the LLM's capabilities, but we also experimented with unfreezing the LLM weights on selected trials. The results are summarized in terms of accuracy in Table I. Amongst the different classification heads, the MLP appears to have the best performance across all three frozen base LLMs with an accuracy of 0.29, 0.27, and 0.27 for Llama, Gemma, and BERT. However, in the case of Llama 2 and BERT, the SLP and the LSTM, respectively, perform on par with the MLP head. Overall the CNN head appears to perform the worst out of the different classification heads when trained with a frozen base model. The unfrozen results do not seem to offer any performance benefit when compared to their frozen counterparts. However, these results should be taken with caution as the implementation of these models was not straightforward and was added mostly as an afterthought.

For a more detailed resolution than simple accuracy scores, we mostly relied on confusion matrices during development. Unfortunately, we can not include visualizations for all of them in this report, but a few selected examples from our trials with the Llama 2 base model are included in the Appendix A. Some interesting insights are that both the MLP as well as the CNN head assign most items to the four categories *false*, *half true*, *mostly true*, and *true*. Especially the MLP in Figure 5 is fairly confident on the labels for *false*, which indicates that this category is the most distinguishable from the rest. A similar case seems to hold for *mostly true* labels, although the proportion of false positives is also increased. In the case of the CNN, the overall number of false positives is slightly increased, but it is also most confident in identifying items labeled as *false* and fairly confident with *mostly true* labels. Other than the MLP, the CNN, however, assigns a disproportionately high number of items to the *half true* category, resulting in a high number of true positives for this category while at the same time severely adding to the number of false positives. Overall, the two confusion matrices hint at two problems. First, the training items assigned to *barely true* and *pants on fire* are not sufficiently coherent for any of our models to extract reliable patterns for classification. Both models are either just guessing by assigning items from these two categories fairly equally across the other four categories or incorrectly assigning most of those items as *false* or *half true*. One reason for this might be the considerably lower number of items in the *pants on fire* category, but this does not explain the effect for the *barely true* labels. More likely is that the observed effect hints at a superficially imposed boundary by the annotators on items that should have ideally been assigned to the same category.

Finally, a curious observation is that the LSTM has a

	LLaMa 2 7b		Gemma 2b		BERT	
	Frozen	Unfrozen	Frozen	Unfrozen	Frozen	Unfrozen
SLP	**0.29**	-	0.25	-	0.26	-
MLP	**0.29**	0.22	**0.27**	0.19*	**0.27**	0.22
LSTM	0.22	0.23	0.24	0.19*	**0.27**	0.24
CNN	0.27	0.21	0.22	0.25	0.26	0.26

TABLE I: Comparison of Classification Heads and Base LLMs - Values shown are accuracy score. All models are trained on LIAR and hyperparameters are kept constant for each head across the different base LLMs. Scores marked with (*) represent runs the resulted in NaN-losses and predicted only a single label at test time.

major problem identifying any meaningful patterns in the LLM features when trained on top of Llama 2. Figure 5 clearly shows that the model largely predicts only one label for every test item. However, when trained on top of BERT, the LSTM classification head can distinguish many labels reliably and perform as well as the MLP head as seen in Table I. Compared to the other two confusion matrices, however, this time, the model is even fairly confident with the *barely true* category, suggesting that the BERT features offer some kind of advantage in this case over the Llama 2 hidden features.

From the experiments in this section, we decided to continue with using the Llama 2 model as the base model for future experiments as it offered the best performance for three out of four classification heads and was also the base of the model that reached the overall highest accuracy score. Interestingly enough, the overall second-best model was not the next largest model, Gemma 2b, as we had originally thought, but BERT. Still, in the direct comparison of the SLP, MLP, and CNN heads, BERT falls short of the Llama 2 performance by a few percentage points.

2) Cofacts: The same four classification heads developed for the LIAR dataset were also tested on the dataset collected from Cofacts. Except for using only one base model, Llama 2. Experiments were conducted on four different versions of the dataset, which are the dataset as we originally sampled it, an undersampled version with the same number of true and fake news, an augmented version where we added additional articles labeled as true to the originally very imbalanced dataset and lastly a combined dataset of the Cofacts and LIAR data. Section IV-B2 describes these dataset augmentations in more detail. One important difference to remember when comparing the results presented in this section to the results obtained from the LIAR dataset in the previous section is that the task now is to classify binary labels rather than six labels.

The results of running our models on the differently sampled versions of the Cofacts data are shown in Table II. Table IIa shows the performance of different classifier heads on the original imbalanced dataset as we sampled it from Cofacts. As in the experiments on the LIAR dataset, the MLP head seems to outperform the alternatives in accuracy as well as in the f1-score metric. Although, precision is noticeably highest for the more complicated CNN head. All heads achieve quite high precision scores, but only the MLP shines with a recall score above the chance level, which also translates to the high f1-score. The MLP is the only head with scores above the chance level across all measures. However, at 0.78 f1-score and 0.65 accuracy, the improvement above the chance level is recognizable, but not by a lot. Surprisingly, the LSTM's performance is very low, with a recall of 0.01, which suggests that the model only learns to predict the majority class and thus fails to distinguish true positives from false negatives. The LSTM is affected the most by the imbalance in the dataset, and our measures to counteract this seem to have been rather ineffective.

In Table IIb, results on the undersampled Cofacts dataset with an equal proportion of true and fake news are shown. The best-performing heads are the MLP again as well as the CNN this time. Arguably, CNN's performance might be more robust in this case as it reaches the best f1 score with 0.88, the highest recall (0.97), and the highest accuracy (0.87). This seems to suggest that the more complicated architectures might be a better fit for the feature-rich texts in the Cofacts dataset, given that class imbalance is adequately dealt with. We assume that the automatic feature extraction of the CNN architecture is especially useful and enables the head to condense only relevant information from the texts, thus yielding a better classification result. Nevertheless, the MLP still performs very competitively, with only two percentage points less in the f1-score measure (0.86) and a four percentage points higher accuracy score (0.91). The MLP holds up as a rather simple architecture with high predictive power. Concerning accuracy, it is still the best-performing architecture. Nevertheless, even the LSTM shows very competitive performance despite having underperformed in the previous experiment, which again supports the hypothesis that complex heads, like the LSTM or CNN, are better suited to deal with feature-rich inputs than simpler architectures. The SLP, on the other hand, falls off completely, which hints at a correlation between architectural complexity and performance. Consistent with the previous experiment, our simplest baseline model does not appear to be able to learn anything meaningful from the data, thus we suspect a certain complexity in the architecture is required to successfully relate the foundational LLM's knowledge to the correct labels.

The third Table IIc summarizes the classification results on the original Cofacts dataset augmented with additional true

(a) Cofacts Original

	LLaMa 2 7b			
	Precision	Recall	F1-Score	Accuracy
SLP	0.84	0.47	0.60	0.48
MLP	0.84	**0.72**	**0.78**	**0.65**
LSTM	0.87	0.01	0.03	0.18
CNN	**0.92**	0.45	0.60	0.45

(b) Cofacts Undersampled

	LLaMa 2 7b			
	Precision	Recall	F1-Score	Accuracy
SLP	0.5	**1.0**	0.67	0.5
MLP	**0.81**	0.91	0.86	**0.91**
LSTM	**0.81**	0.92	0.86	0.85
CNN	**0.81**	0.97	**0.88**	0.87

(c) Cofacts Augmented

	LLaMa 2 7b			
	Precision	Recall	F1-Score	Accuracy
SLP	0.72	0.62	0.67	0.56
MLP	0.72	0.51	0.60	0.50
LSTM	0.87	0.21	0.34	0.41
CNN	**0.89**	**0.76**	**0.82**	**0.76**

(d) Combined Cofacts and LIAR

	LLaMa 2 7b			
	Precision	Recall	F1-Score	Accuracy
SLP	0.75	0.36	0.48	0.53
MLP	0.84	0.32	0.46	0.55
LSTM	0.82	**0.67**	**0.74**	**0.71**
CNN	**0.87**	0.54	0.66	0.67

TABLE II: Comparison of Classification Heads across Cofacts Datasets - Heads follow the same architecture as the heads tested on LIAR, hyperparameters were optimized individually.

articles from the MCFEND dataset [31]. In this experiment, we get a good performance for the CNN head with an accuracy score of 0.76 and an f1-score of 0.82, while all other architectures fall short of even reaching the chance level. Nevertheless, except for the MLP head, the additional true items seem to have helped performance across the board, and most so for the CNN head. Taken together with the results from the previous experiment, this seems to suggest that the CNN architecture is best equipped to deal with long feature-rich inputs and, second, that it might be the most robust architecture performing well even with inputs from two different domains. However, this only seems to hold as long as the label distribution is not severely imbalanced as in Table IIa.

The last Table IId shows results on the Cofacts dataset combined with LIAR data. The picture changes again, and the LSTM becomes the best-performing classification head with 0.74 f1-score and 0.71 accuracy. The other heads perform below chance level in terms of f1-score but in terms of accuracy, barely above chance in the case of the SLP and MLP heads and considerably above chance in the case of the CNN. Still, the observation that more complicated classification heads are more adaptable to utilizing features from the LLM seems to hold since both the LSTM, as well as the CNN, perform better than the SLP and MLP. Additionally, comparing the results to Table IIa, the best-performing head reaches much higher values in both f1-score and accuracy when trained on the combined dataset compared to training on the original dataset, which was also observed when training on the augmented dataset in Table IIc. Additionally, except for the MLP, performance is also higher for each individual head when trained in the combined dataset. Naturally, higher performance is expected when training on more data, but a major portion of this performance gain can presumably be attributed to the reduced class imbalance in the combined dataset, as we also observed in Table IIc. Interestingly, comparing Table IIc and Table IId, the highest accuracy is reached by a model trained on the data augmented with items from MCFEND rather than from the LIAR dataset, despite the former dataset being much smaller with only 5,798 compared to 11,400 items in the training split. This might be related to both the quality and domain of the data. While the MCFEND data stems from an arguably quite different domain than the Cofacts data, articles from the LIAR dataset are probably even more foreign. The strain of bridging this gap might result in a decreased performance. Additionally, as described in Section IV-B, the text data provided in LIAR and Cofacts varies widely in terms of utterance length. While LIAR provides only short quotes, Cofacts includes entire news articles. Since the MCFEND dataset also includes articles with their title and body, the quality of the data can be assumed to be much more similar to Cofacts. Each of the added true items might, therefore, be much more informative than the items from the LIAR dataset and thus improve performance despite being much less in numbers.

Taken together, the results on the Cofacts datasets seem to suggest that more complex classification heads are better suited to extract task-relevant information from feature-rich inputs and also more adept at dealing with data from different domains. To test this hypothesis, we trained a more complex version of our CNN classification head on the combined Cofacts and LIAR data. Compared to the simpler CNN head, the extended version has three convolution and pooling stages instead of only one, which, in theory, enables a more fine-grained feature extraction and, thus, should result in better performance. All other aspects of the architecture, as well as hyper-parameters, were held constant. The best results achieved by the extended head are shown in Table III. For comparison purposes, the original results from Table IId are repeated. As suspected, the extended CNN head overtakes the now simpler alternatives with an f1-score of 0.76 and an

accuracy of 0.74. This is better by a few percent points than the originally best-performing LSTM head as well as a large improvement over the performance of the simple CNN head. These results suggest that a more complex head is indeed a better fit for more feature-rich data, given that the data is not severely imbalanced. This effect is especially strong when mixing different domains of data. Therefore the addition of Cofacts data warrants a more complex classification head than required for training solely on the LIAR dataset (Table I). We chose to extend the CNN instead of the LSTM head for this test since both heads showed a similar tendency with arguably more robust results for the CNN across the different experiments. However, the main reason was purely practical, as the training time for the LSTM was almost three times longer than that of the CNN head. Therefore, the full training took about 48 hours on one A100 GPU. So, it was much more practical for us to test different extensions on the CNN head instead.

	LLaMa 2 7b			
	Precision	Recall	F1-Score	Accuracy
SLP	0.75	0.36	0.48	0.53
MLP	0.84	0.32	0.46	0.55
LSTM	0.82	0.67	0.74	0.71
CNN	**0.87**	0.54	0.66	0.67
CNN-extend	0.86	**0.68**	**0.76**	**0.74**

TABLE III: Extended CNN Head - New results are shown on the last line, other results are repeated from Table IId.

Considering overall performance and training efficiency, the CNN head seems to be the best choice in both monolingual cases as well as bilingual cases. A service for Taiwanese fake news detection would thus ideally use the model with a CNN head trained on the undersampled Cofacts dataset. This is a very different conclusion than initially expected from the results in Section IV-F1, where the CNN head was outperformed by simpler architectures in every scenario. Since the only change is the data used, we are very confident that this is a direct consequence of longer and, thus, more informative texts present in the Cofacts dataset. Such texts are, therefore, necessary to achieve high performance in content-based fake news detection tasks.

G. GPT4 Zero-Shot Prompting - Ho-Hsuan

As part of our research to assess the effectiveness of LLMs in detecting fake news, we used zero-shot prompting techniques with "gpt-4-0125-preview" on two datasets - the downsampled Cofacts Dataset and the initial 200 samples of the LIAR test set. For the Cofacts Dataset, we used binary true/false labels, while the LIAR dataset utilized its original six-label schema. The prompting tasks were carried out in Traditional Chinese for the Cofacts dataset and in English for the LIAR dataset to match their linguistic context. The prompts are presented in Figure 4.

Accuracy was determined by comparing the label generated by GPT-4's text generation mode against the correct dataset label. It is worth noting that GPT-4 sometimes produced invalid responses. For instance, when dealing with the Cofacts data, it would occasionally state that there was not enough information or it does not possess the ability to evaluate the news item's credibility and advise consulting reputable news sources instead. Additionally, for the LIAR dataset, GPT-4 introduced a new label, "mostly-false," which is not part of the original six-label system. This underscores a potential issue with the LIAR dataset's confusing labeling system. The invalid responses were all excluded from the accuracy calculation.

As reported in Table IV, the achieved accuracies were 66% for the Cofacts and 21% for the LIAR data. These figures fall short of the accuracies attained by our models that integrate both LLMs and neural classification heads, thereby addressing the question of whether SOTA LLMs alone surpass existing machine learning models in fake news detection. This small prompting experiment suggests that, although convenient, relying solely on LLMs in its text generation mode to classify fake news does not yet match that of models specifically engineered to counter misinformation, affirming the persistent relevance of incorporating classification heads in such models.

	Cofacts	Liar
ChatGPT-4	0.66	0.21
Llama 2 7B + MLP	**0.91**	**0.29**

TABLE IV: GPT 4 Prompting - Values refer to accuracy scores. Results are compared to the best-performing model from IV-F1 and IV-F2. Cofacts here refers to the undersampled version.

H. Ablation Study - Jakob

To test the contribution of the additional features added according to the conceptualization in Section IV-A, we conducted an ablation study with our most robust and well-performing model, the CNN head trained on the undersampled Cofacts data. Table V summarizes the results. The first line is the CNN from the original experiment taken from Table IIb. The lines after that represent versions of the same head with similar complexity but a decreasing number of additional features. The second line has the CNN head with added sentiment but no surprisal, while the third line has the classification head with added surprisal but without sentiment. The model on the last line is the simplest model with only hidden features extracted from the input text by the base Llama 2 model. For further clarification, the scores in this table should ideally include confidence intervals, but due to time constraints, we were unable to compute them. In this case however, the results are still reasonably well interpretable without them.

The accuracy and the f1-score show that the different models do not differ substantially in performance, suggesting that the additional features are not very useful for the model to identify fake news. This contradicts our initial expectations and refutes our theoretical contributions from Section IV-A. To some extent, we did expect the effect of adding explicit surprisal to have only a minor impact. The internal LLM representations of uncertainty seem to be much more intricate so that the addition of surprisal becomes superfluous. Although

	LLaMa 2 7b			
	Precision	Recall	F1-score	Accuracy
CNN	0.81	0.97	0.88	0.87
CNN -surprisal	0.83	0.94	0.88	0.87
CNN -sentiment	0.81	0.93	0.87	0.86
CNN -surprisal, -sentiment	0.82	0.96	0.89	0.88

TABLE V: Ablation - The CNN head is retrained on the undersampled Cofacts dataset with subsequently fewer additional features. The simplest model performs on the same level as the complex models.

sentiment had been identified as one of the features that offered the most increase in performance for content-based fake news detection in previous research [14], [32], we do not observe such an effect. The performance of the CNN head without sentiment is just as good as the performance of the head with all additional features, as well as the head with only sentiment added. We suspect that the LLM might have already developed an ability akin to sentiment analysis during pretraining as some pre-trained LLMs have successfully been adopted for sentiment analysis without many modifications [39]. From this point of view, the addition of explicit sentiment scores merely adds redundant information.

V. Usability and Public Accessibility

A. Existing Solutions - Ho-Hsuan

Given the ubiquitous false information nowadays, it is crucial to have easily available tools and platforms dedicated to detecting misinformation. Such resources are essential for keeping the public informed. Several user-friendly platforms are available that make it easier for a diverse audience to distinguish between truth and falsehood. This section will focus on the larger fact-check platforms available in the U.S. and Taiwan, each offering unique features that make them effective and easy to use.

The Taiwan FactCheck Center[12] is an independent organization accredited by the International Fact-checking Network (IFCN)[13]. It verifies facts and debunks misinformation in Taiwan's information sphere, following the IFCN's Code of Principles. It has undergone four successful accreditations by IFCN, highlighting its adherence to rigorous fact-checking standards and international best practices. Its process for classifying fake news does not utilize artificial intelligence; instead, verification is only conducted by human experts using openly available sources. *Cofacts*[14] is an open-source project that brings fact-checking to LINE, Taiwan's most popular messaging app. Its seamless integration into a widely used communication tool sets this platform apart, enabling users to verify information swiftly through a familiar interface. By harnessing the power of volunteer contributions, *Cofacts* has built a robust database of fact-checked information, making the verification process collaborative. The *Cofacts* dataset is also where the *Cofacts* dataset used in the current work is from, contributing valuable, up-to-date real-world data. However, this model presents certain challenges. The lack of AI involvement in the labeling process means that all assessments of truth or falsehood are conducted by human volunteers, who may vary widely in their expertise and biases. Since anyone can contribute to the fact-checking process, it is potentially vulnerable to individuals with malicious intentions who might seek to skew the information landscape. Moreover, the barrier to entry for contributors can be high; the requirement for users to find sources to support their claims and articulate reasons for their labeling decisions can be daunting. While ensuring a degree of reliability, this rigorous process might deter casual users from engaging more deeply with the platform, limiting the breadth of participation in the fact-checking process.

FactCheck.org, *PolitiFact*, and *The Washington Post Fact Checker* are among the most well-known fact-checking tools in the U.S. Their primary focus is all on political news. *FactCheck.org*[15] is a project run by the Annenberg Public Policy Center at the University of Pennsylvania. Its goal is to reduce confusion and deception in U.S. politics by checking the accuracy of statements made by politicians, political ads, and news releases. *FactCheck.org* has a reputation for conducting thorough and impartial analyses and providing detailed evidence to support its fact-checks, which adds to its credibility. However, the detailed nature of its reports can sometimes make them difficult for readers who want quick verifications. *PolitiFact*[16] is a nonprofit project with a website operated by the Poynter Institute that fact-checks claims made by politicians and public figures. It is known for its impartiality and broad coverage of political viewpoints. Reporters and editors from its newspaper and affiliated news media partners are responsible for verifying the truthfulness of statements made by elected officials, candidates, their staffs, lobbyists, interest groups, and others involved in U.S. politics. *The Washington Post Fact Checker*[17] is known for its Pinocchio rating system, which assigns more Pinocchios for political statements, reports, and campaigns that contain more falsehoods. *The Washington Post Fact Checker* offers comprehensive analyses and contextual information. However, readers must subscribe to The Washington Post and pay a

[12]https://tfc-taiwan.org.tw/en
[13]https://www.ifcncodeofprinciples.poynter.org/
[14]https://en.cofacts.tw/
[15]https://www.factcheck.org/
[16]https://www.politifact.com/
[17]https://www.washingtonpost.com/politics/fact-checker/

small fee to access the full content of their fact-checked articles and analysis.

Although multiple fact-checking websites are available, Wittenberg and Berinsky [40] indicate that misinformation remains resilient to correction due to cognitive biases such as the continued influence effect, where beliefs are influenced by misinformation even after it has been corrected. Attempts to correct misinformation can also backfire, reinforcing false beliefs in those with opposing ideologies. This research reminds us that understanding the psychological and contextual factors that govern fake news' persistence and impact is important, as well as how individual differences, such as political beliefs and personal traits, can affect one's susceptibility to misinformation and response to correction efforts. This study again illustrates the complex challenge of combating misinformation in today's digital media landscape.

B. Interface Design - Ho-Hsuan

Our News Classifier web app[18] [19] hosted on Streamlit is divided into four sections: the Main Page, Fact-Checking Links, Datasets, and Model Structure. The Main Page is where users can interact with the classification model. To initiate classification, users only need to provide the article's title, body, and source link, which are easily accessible for convenience. After users input a news article and click the "classify" button, the system tokenizes the article into subwords, each highlighted in varying shades of red. The intensity of the red correlates with the subword's contribution to the classification outcome, with darker shades indicating greater influence. This visualization is made possible by tracing the gradients in the neural network's classification head. Once the classification is complete, users can either agree or disagree with the result. If they disagree, they can select "no" and provide an explanation. By clicking the "save" button, the input is immediately archived in a Google Sheet by using Google Drive and Google Sheets APIs.

The Fact-Checking Links section compiles references to various fact-checking sites, as described in section A under "Existing Solutions." In the Datasets section, Visitors can see an overview of the first five rows from the Cofacts and LIAR datasets across the training, testing, and validation sets. This section also includes pie charts showing both datasets' label distributions and training, testing, and validation data splits. Lastly, the Model Structure segment displays the architecture of our model, as detailed in Fig. 1, providing an overview of the model running behind the web app.

C. Model and Visualization - Philipp

The model configuration implemented within the Streamlit application represents the simplest iteration within our repertoire, employing frozen BERT with the Single Layer Perceptron trained on the aggregated binary label LIAR dataset. This choice is partially attributed to the limited computational resources available under Streamlit because it does not provide access to a GPU. Additionally, prioritizing user experience in terms of speed and responsiveness led us to minimize processing waiting times, thus favoring the Single Layer Perceptron without any other separate inputs like sentiment because obtaining a sentiment classification would necessitate the need for another forward pass on a separate model.

The attribution of importance (saliency) is obtained using the gradient-based "Simple Gradient" interpretation method [41], enabling activation scores inside the neural network for each token, representing how much that token contributed to the classification output. The general recipe of computing the simple gradient prescribes performing a forward pass to receive an output, followed by a subsequent backward pass given a single output cell to back-propagate the signal with respect to the specific layer of interest.

Although compared to other interpretation methods, the simple gradient method does not come without its own limitations, most notably the saturation problem, which may lead to underestimating the importance of inputs [42]. The saturation problem arises from the derivative of non-linear activation functions, resulting in a distortion of the signal. For instance, commonly used ReLU layers assign 0 importance to negative inputs and 1 to positive inputs (through their derivative), thereby ignoring the relative scale of positive contributions. However, despite these drawbacks, we opted for the simple gradient method to demonstrate our system's capabilities. To additionally mitigate the saturation problem, we only back-propagated through our classification head with respect to the output hidden states of the underlying pre-trained language model.

To ensure meaningful token-level scores with the simple gradient method, a slight adjustment is made to the initial architecture, replacing the mean pooling layer with max pooling. This modification emerges from the behavior of mean pooling, which assigns averaged values to every token, thereby hiding individual contributions.

We found that the architecture using max pooling showed comparable results to our other models.

Providing a glimpse into the inner workings of our model offers great value compared to only showing a single output label because it enhances the overall user experience by offering insight into the typically non-transparent nature of neural networks. Additionally, it has been shown that, to some extent, neural networks do attend to the same words as humans do in certain tasks [43]. This increases the interpretability of our model's behavior since interpreting the raw hidden states may be infeasible. Moreover, neural word embeddings were shown to be cognitively plausible, as they correlate with human brain signals and eye-tracking data, indicating that they are not just mere artifacts of computational processing [44].

The color scheme employed for visualization utilizes a heat map in the red spectrum, where lower values are more transparent and tend towards white, while higher values are more saturated and tend towards a darker red, with orange representing middle values. To scale the gradient scores, we adopt a strategy whereby the lowest value is assigned 0%

[18]https://fakenewshq.streamlit.app/

[19]https://github.com/rozariwang/nn_softp_interface

saturation, and the highest value is assigned 100% saturation, with all other values scaled accordingly. In hindsight, this approach to scaling the gradient values turned out to be suboptimal because darker colors might be interpreted oppositely as being less important. Additionally, setting the lowest (most negative) value to 0% saturation might lead to other higher but still negative values being set to a saturation higher than 0%, indicating a positive contribution. A clearer approach might have been to completely ignore negatively contributing tokens, leaving them blank within the visualization output and applying the scaled saturation to positive values only.

D. Public Engagement and Education - Ho-Hsuan

Although Machine Learning methods are helpful in detecting fake news, the problem of misinformation is complex and cannot be solved by computational solutions alone. To address this challenge, some research has been focused on involving public engagement in the fight against fake news. This approach involves not only relying on complex ML models but also developing educational tools and promoting public awareness and participation. By engaging the public, individuals are empowered to evaluate information critically, which complements technological efforts to create a more informed and resilient digital ecosystem. This multidimensional strategy is indispensable in the fight against misinformation.

Regarding public engagement from the point of view of social media platforms, Avram et al. [45] conducted a study to examine the impact of social engagement metrics, such as likes and shares, on users' susceptibility to misinformation on social media platforms. The study used a news literacy game called Fakey, which simulates a social media feed with both mainstream and low-credibility news articles accompanied by randomly generated social engagement metrics. The research analyzed over 8,500 unique players' interactions with approximately 120,000 articles. The results of the study reveal that high social engagement metrics significantly increase the likelihood of users engaging with low-credibility content through likes and shares while decreasing the likelihood of fact-checking. These findings suggest that social engagement metrics amplify the susceptibility to misinformation by implying content validity based on its popularity. Therefore, the study calls for social media platforms to reconsider the display of engagement metrics to mitigate the spread of misinformation. It urges further research into design alternatives that balance visibility with the prevention of misinformation dissemination.

Lewandowsky et al. [46] presents a compelling argument for the effectiveness of inoculation theory as a proactive strategy against the spread of misinformation. By exposing individuals to weakened forms of misinformation along with refutations (prebunking), the authors have demonstrated that people can be "immunized" against false information, thus enhancing their resilience in various contexts, such as political misinformation and health myths. This approach, similar to biological inoculation, not only shows potential in countering specific misleading claims but also suggests a broader application capable of generating a "broad-spectrum" resistance to misinformation techniques. Additionally, the paper introduces the concept of attitudinal "herd immunity," indicating that widespread inoculation within a community can protect against the proliferation of misinformation. The authors also advocate for a layered strategy in misinformation countermeasures, incorporating both proactive and reactive elements, to foster societal resilience.

Roozenbeek and van der Linden [47] explore the efficacy of an online browser game, "Bad News," that helps combat online misinformation by teaching players how to recognize and use six core misinformation strategies. This approach is based on inoculation theory, suggesting that exposing individuals to weakened versions of misinformation can build cognitive resistance. The large-scale study involved 15,000 participants and showed that the game improved their abilities to identify and resist misinformation, regardless of their demographic differences. This highlights the potential of interactive, experiential learning through gameplay as a "broad-spectrum vaccine" against various misinformation strategies.

Combining advanced machine learning with better social platform designs and educational interventions is an all-around approach to combating misinformation. By detecting false information with algorithms and equipping individuals with critical thinking skills through interactive games and public awareness campaigns, we can target the technological aspects of misinformation, foster a more informed and resilient public, and ensure a healthier digital ecosystem.

VI. Limitations

A. Datasets

1) LIAR - Ho-Hsuan: The LIAR dataset, one of the most frequently utilized benchmarks in fake news detection studies, comprising approximately 18% of such research, presents limitations highlighted by its consistently low classification accuracies [11]. Since its introduction in 2017, various studies, irrespective of model architecture, report a maximum accuracy of only around 27% for its six-label classification system, including the original paper where it was first introduced [23]. This observation raises concerns regarding the adequacy of the dataset and its effectiveness in training models to detect fake news. We have identified three primary issues with the dataset: (1) A confusing and overly detailed labeling system that lacks clarity; (2) Outdated content and labels that do not match the LLMs' current real-world knowledge; (3) Statements within the dataset are too brief and lack sufficient information for classification.

Labeling system: According to PolitiFact[20], where the LIAR dataset was obtained from, the labels are defined as follows:

1) **MOSTLY TRUE** – The statement is accurate but needs clarification or additional information.
2) **TRUE** – The statement is accurate and there's nothing significant missing.

[20]https://www.politifact.com/article/2018/feb/12/principles-truth-o-meter-politifacts-methodology-i/

3) **HALF TRUE** – The statement is partially accurate but leaves out important details or takes things out of context.
4) **MOSTLY FALSE (BARELY-TRUE)** – The statement contains an element of truth but ignores critical facts that would give a different impression.
5) **FALSE** – The statement is not accurate.
6) **PANTS ON FIRE** – The statement is not accurate and makes a ridiculous claim.

The LIAR dataset's six labels introduce a level of complexity that challenges even human labelers. For instance, the statement, "Says Barack Obama's comments indicate he believes in redistribution of wealth," is labeled as "mostly-true" in the LIAR test set. However, the reason that it is labeled as "mostly-true" instead of "half-true" or "barely-true" remains unclear, as decision-making is based on majority voting from the fact-checkers without further explanation. This ambiguity is exemplified across numerous instances like this within the dataset, highlighting the cognitive blurred lines between categories. Moreover, analysis of confusion matrices from MLP and CNN (5) models reveal specific patterns: these models are adept at identifying false statements but rarely assign labels like "barely true" or "pants on fire," suggesting these categories may not be that meaningful. Furthermore, the highest error rates occur among the "half true," "mostly true," and "true" categories, indicating the model's difficulty in differentiating these labels. This could point to a need to consolidate these three categories into a single, more definitive category.

Outdatedness: The LIAR dataset, established in 2017, has become increasingly outdated as societal and factual contexts evolve. Statements that were once considered true may no longer hold true today, and conversely, assertions previously deemed inaccurate might now be validated by new developments. This temporal dynamic is particularly evident in statements involving quantitative data or forecasts, such as "Rhode Island's exports have increased by 53 percent in the last two years." and "Wisconsin is on pace to double the number of layoffs this year." These examples, where no anchor of time is included, illustrate how the passage of time can shift the factual accuracy of statements, causing their categorization within the dataset to potentially vary over time. This inherent temporal limitation in the LIAR dataset suggests a need for continual updates and revisions to maintain a dataset's relevance and accuracy. Only by doing so can one ensure that the datasets in use adapt to changing real-world conditions.

Uninformative Statements: The LIAR dataset contains mostly brief statements such as "On running a civil and polite campaign." and "Marijuana is less toxic than alcohol." Although the metadata columns provide additional details, they primarily focus on politically related information and thus do not enhance the overall informativeness of these statements. For instance, the statement "Marijuana is less toxic than alcohol." is classified as "mostly true." This classification acknowledges the statement's general accuracy but suggests the need for further clarification. Indeed, marijuana is generally regarded as less harmful than alcohol in terms of lethal overdose potential, long-term health effects, and its lesser association with violence and substantial societal costs. Yet, the conciseness of most entries in the LIAR dataset inherently demands more extensive description and supplementary information. Consequently, the combination of such a fine-grained labeling system with short, often vague statements renders the distinctions between labels such as "true" and "mostly true", as in the example above, very ambiguous. Additionally, the brevity of many statements in the dataset often precludes them from fitting the definition of fake news. As defined, fake news involves fabricated content that mimics legitimate news media but lacks journalistic integrity and intent, undermining authentic news sources' credibility. Nonetheless, numerous entries in the LIAR dataset labeled as false do not align with this definition. For instance, the statement, "What the facts say is ...the best scenario for kids is a loving mom and dad," categorized as 'false' in the test dataset, seems to reflect subjective values rather than objective falsity. Therefore, statements like this do not constitute fabricated news.

2) Cofacts - Jakob & Ho-Hsuan: One notable advantage of the Cofacts dataset compared to the LIAR dataset lies in its clarity. From the perspective of native speakers of Traditional Chinese from Taiwan, distinguishing between true and false articles in this dataset is relatively straightforward. The language employed and the formulation of claims facilitate intuitive assessments of their integrity. Conversely, the LIAR dataset often presents a challenge; its sample statements frequently require a deeper understanding of the context or additional research for a judgment on their truthfulness. This complexity can hinder the training and learning process in the neural network classification heads.

Yet, Cofacts does not come without its own challenges, as described in Section IV-B2, the distribution of fake and true news is severely imbalanced, and repercussions of this were observed in Section IV-F2. Although we implemented a weighted loss function, the imbalance effect seemed to prevail through many of our experiments only being alleviated by augmenting the data with additional items. This technique might not be the best option in every case as the quality of the additional data might drastically change the composition of the dataset and thus needs to be controlled very carefully as the introduction of potentially adversarial biases might substantially influence a model's capabilities. For the research at hand, we are confident that the MCFEND data [31] is sufficiently similar to the original Cofacts data that it improves performance as we observed in IIc. On the other hand, we are convinced that the LIAR dataset is less aligned with the Cofacts data and, thus, is not a good fit for augmentation. Although the results in Table IId indicate an improvement in some parts, they also suggest that the resulting dataset is very noisy. At this moment we are uncertain if this noise is due to the prevailing imbalance of the Cofacts data or new noise introduced by the LAIR data. A more effective technique to combat imbalance than what was presented here would go a long way to resolve this issue.

While the feature-rich texts are a major benefit of Cofacts, they can also be very noisy. Some text, for instance, might only be 10 characters long, while others have about 10,000 characters. Additionally, the formatting can vary widely as the texts submitted by different users can have vastly different sources. Cautious preprocessing of the data is, therefore, very important. Since LLMs are generally very adept at dealing with any kind of text, we decided to keep the preprocessing to a minimum for this project. However, this might not be the optimal solution in every case. For the combined Cofacts and LIAR dataset, for example, additional preprocessing that further aligns the Cofacts items to the LIAR format might reduce some potential noise effects caused by differences in common punctuation and sentence marking. We also deliberately included emoticons in the Cofacts data as discussed in Section IV-B2. However, since there are no emoticons in the LIAR dataset, this might be a potential point of tension between the two datasets.

B. LLMs' Inherent Biases - Philipp

The performance of any classifier rises and falls with the quality of the training data. For the kind of architectures we used, there are two kinds of training data: the vast amounts of general textual data used for the pre-trained LLM (not under our control), and then a smaller, but highly specific dataset for the classification head.

The first major limitation of our approach is the fact that the LLMs we used are mostly focused on the English language, even though they do possess some multilingual capabilities, but BERT was trained on 100% English text only [19]. Gemma's training data includes other languages, but the model was not intended for state-of-the-art multilingual performance [21], and for Llama 2, the original authors state that "[m]ost data is in English" [20].

LLMs appear to be able to fact-check. But this impression originates from their principle of operation in general, dealing with the probability of words co-appearing, therefore factuality is a challenge for language models overall, as they lack an internal knowledge base. However, solely relying on this mechanic of probability enables modern LLMs to approximate the sense of factuality surprisingly well, which in return can lead to catastrophic inconsistencies known as hallucinations [48], as well as the flawed characteristic of being manipulatable.

Training Cutoff When utilizing a pre-trained language model for fake news classification and fact-checking, it is necessary to acknowledge that the underlying LLM is exclusively trained on historical text data. Consequently, in dynamic environments where the state of information changes over time, there exists a risk of misclassifying newly emerged facts or information as fake. This discrepancy arises due to the LLM's reliance on past data, resulting in a lower probability assigned to the occurrence of recently surfaced information. Consequently, such information may appear misleading or false to the model despite its factual accuracy in the current context. To mitigate this issue, continual refinement and updating of the LLM's parameters with real-time news data is necessary.

Moreover, it is conceivable that certain historical instances of fake news articles are included in the LLM's training dataset. Additionally, there may be instances where original fake news was exposed, and their falsity was explained. This phenomenon could be interpreted as a bias towards known fake news because the system relies more on innate knowledge rather than identifying certain patterns. In consequence, the robustness of the classifier is impacted, as it is influenced by the frequency of encountering such data during training and the amount of contextual information provided.

Input Modality A broader limitation of all LLMs we used is the fact that they are exclusively trained on the input modality of text, rendering them susceptible to misinterpretations when processing written language. For instance, many statements included within the LIAR dataset are transcribed from public speeches or live events, omitting (para-)linguistic nuances and suprasegmental features present in spoken language like pauses, rhythm, timing, pitch, intonation, stress, or volume. Additionally, extralinguistic cues, like gestures or facial expressions, are missing. Modern fake news found on the internet is also often accompanied by images, where the visual content itself may accurately depict reality, and the accompanying textual content may remain legitimately factual. However, when both image and text are combined, the original meaning of both separate elements may be intentionally distorted. Consequently, the nature of our models that solely rely on textual input modality represents a substantial limitation, particularly concerning contemporary multi-modal fake news articles encountered in real-world scenarios.

C. Technical and Operational Challenges - Philipp

One limitation is the evaluation of the 6-label dataset since we primarily relied on the classical accuracy metric to assess the performance. However, it is important to acknowledge that other metrics do exist, such as the macro F1 score [49], and that these more fine-grained metrics could have provided additional insights for a more comprehensive evaluation and comparison. The macro F1 score considers the precision and recall for each class and calculates the unweighted average across all classes, offering a more balanced assessment, particularly in our scenario facing imbalanced class distributions.

Additionally, there are other limitations regarding training time and computing resources. Our custom-designed classification heads might have been small in size and rather simple. Expecting that the numerous parameters of the pre-trained LLMs would sufficiently address understanding the character of the fake news, we did not explore the implementation of significantly larger or more complex classification heads. The comparably poor performance of our models utilizing the unfrozen foundational LLMs suggests that insufficient resources were allocated toward training these models to their full potential. Training our classification heads and, thereby, running the LLMs for inference have already taken a substantial amount of time. The constraints regarding time and

computational resources also prohibited properly leveraging the very big models consisting of up to 70 billion parameters. To address these constraints about time and computational resources, we improved the computational efficiency by almost always making use of quantization [50], a technique that reduces computational precision. However, this approach may have limited the predictive performance of our models. A comparative examination of alternative strategies for mitigating these constraints, such as fine-tuning via LoRA (Low-Rank Adaptation) [51], would have provided valuable insights.

D. Ethical and Social Considerations - Philipp

In a psychometric study about LLMs, it was shown that those models absorb the moral norms present in the training corpora [52]. As the models used in our study are primarily trained on (US-American) English text, the model inevitably embodies the moral values of the (US-American) English society, reflecting a greater affinity with what is deemed socially desirable within this cultural context, as opposed to other spectra of moral values. A prominent example of this phenomenon pertains to the discourse surrounding freedom of speech: in the United States, the right to freely express diverse opinions is highly esteemed and constitutionally protected. Conversely, in countries like Germany, freedom of speech is constrained by a commitment to truthfulness, thereby prohibiting the spread of blatantly false averments. Consequently, any fake news classifier fashioned similarly to ours may prove unsuitable for deployment in communities characterized by divergent debates and distinct social norms. Alternatively, its deployment could inadvertently propagate these societal norms and moral values to other cultures, disrupting the original equilibrium.

Moreover, a parallel argument applies to the society from which the training corpus originated, as pre-trained LLMs possess the potential to reinforce prevailing beliefs and viewpoints, thereby impeding the emergence and advancements of novel legitimate opinions. This phenomenon, grounded in confirmation bias, may lead to a disproportionate focus on opinions anticipated to evade being classified as fake. Another consideration addresses the phenomenon of survivorship bias, wherein only statements flagged as non-fake are deemed credible, potentially fostering the dominance of established narratives and perspectives.

Another possible risk concerning the misuse of a fake news classifier is the potential for defamation against individuals, in case the model erroneously classifies information as fake. Alternatively, the classifier may unintentionally promote the spread of genuine misinformation if it incorrectly assigns the true label to false information, thereby serving as a validating testimonial.

In conclusion, when releasing such a classifier to the public, it is necessary to raise awareness of this model not exhibiting absolute truth. To support this awareness of the model's limitations, further work needs to be done on trustworthy AI, increasing explainability and interpretability. This initiative is suitable to enable researchers to better understand and refine the model's predictive capabilities but also provide end-users with a sense of traceable reasoning for the model's behavior. Consequently, this benefits all stakeholders, as it facilitates critical reflection of the model's outputs.

We must also consider the origins of the foundational LLMs, as they may be gated or entirely closed source, indicating that entities with potentially differing intentions control them. This raises concerns regarding the potential for misuse when a society is subject to the influence of such models. In particular, given that the training process for modern LLMs demands substantial computing resources, the creation of powerful modern LLMs is exclusive to a select group of powerful actors, opening up the risk of whole societies becoming reliant on external technology.

VII. Future Directions - William

The development and deployment of a misinformation detection model leveraging the capabilities of LLMs represents a step toward actively combating online misinformation. However, the landscape of misinformation propagation is always changing, and this necessitates ongoing efforts to push the boundaries of these models' capabilities. Future work, with a special emphasis on developing user-friendly and publicly accessible solutions, can be directed toward the following tasks:

Instruction fine-tuning: also referred to as "prompt engineering", instruction fine-tuning tailors the model's responses via specific instructions or queries related to the task at hand, enhancing both precision and relevance.

Incorporating short-term memory into the prompt template. This method leverages recent or context-specific information to allow the model to make more informed predictions based on the immediate context. This dynamic approach could potentially enhance the model's ability to detect nuanced or emerging sources of misinformation.

Further refining our models by enhancing their long-term memory through comprehensive fine-tuning. This would focus on the model's ability to retain and utilize its extensive knowledge acquired across its training lifecycle, enriching its understanding of misinformation patterns across time. This would allow the model to stay effective against both current and future sources of misinformation by drawing on the totality of its learned representations.

Expanding model diversity through the integration of newer, larger, or additional LLMs, especially those trained on multilingual data. The ability to detect misinformation in a single setting is one thing, but being able to extend this and effectively detect misinformation across languages and cultures would constitute a degree of generalizability that is not currently attainable by our models. This could involve training on multilingual datasets, or incorporating additional multilingual language models for feature extraction.

Real-time detection and scalability by incrementally training the model on new data. One avenue for sourcing new data is by user input into our web app. When a user is looking for a prediction, the input data is fed to our model and a prediction

is given. If this data is saved, then further feedback in the form of user assessment and agreement could supplement the original datasets with new and up-to-date knowledge. Over time, this will enhance the models' capabilities and ensure the knowledge base does not fall out of date.

Ethical considerations and bias mitigation. The infamous example of *Tay, the Racist Chatbot* showcases how an LLM's knowledge base can be influenced by the kind of language it is trained on, to terrible social detriment [53]. In order to prevent misclassification and perpetuation of biases presented in the data, human assessment and monitoring would be necessary, in an ongoing effort to ensure the model's fairness and reduce the risk of bias and harm.

Trustworthy news source identification. One limitation we placed on ourselves when developing the misinformation detection models was to train them using exclusively input derived from the statement text. Other models have been developed incorporating significant amounts of metadata as input features, and this has been shown to provide significant performance increases [54]. By incorporating extra-textual information, such as the ability to discern and prioritize information from sources with a history of accuracy and reliability, the model can help identify and guide users toward more credible sources of information, educating the user and building misinformation literacy skills.

VIII. Conclusion - Ho-Hsuan

This paper explores the issue of political misinformation, examining its evolution over time and its impact on Taiwan and the United States. We reviewed various academic studies employing different methodologies and models to combat the spreading of fake news. In our research, we utilized datasets such as LIAR, which, while common and relatively easy to train, offers limited insights. Additionally, we used the Cofacts dataset, which provides richer information but suffers from an unbalanced label distribution. To enhance the utility of the Cofacts dataset, we incorporated additional sources, achieving satisfactory results.

Our prototype model integrates sentiment scores, surprisal values from LLMs, and the final hidden state of the language model to classify misinformation. Through our analysis, we demonstrate that LLMs with neural network classification heads, including simpler architectures like SLP and MLP for LIAR and more complex ones like CNNs for Cofacts, can address the sophisticated nature of misinformation. The performance of our models under various combinations of LLMs and classification heads helps us understand their predictive capabilities and lays the groundwork for future research. Our experiment also included deploying GPT-4 for zero-shot prompting and performing an ablation study. Our findings reveal that SOTA LLMs fall short of specialized models explicitly designed for fake news detection when used solely in their text-generation capacity. Moreover, the ablation study shows that including supplementary features such as sentiment scores and sub-word surprisal values did not significantly improve model performance, indicating that such information may already be encapsulated within the LLMs' last layer embeddings. As part of our research, we have also developed a user interface that helps collect user feedback and input for future model training. This interface includes a result visualization tool that identifies words that significantly influence classification outcomes.

As we navigate the ever-changing landscape of misinformation, our study highlights the complexity and persistence of fake news, underscoring the need for a multifaceted approach to detect and mitigate its spread effectively. Future model refinement must also navigate misinformation in diverse languages and cultures, prioritize ethical considerations, and address potential biases. These challenges require ongoing attention and innovation.

Just as the *Dream of the Red Chamber* warns of the fluidity between truth and deception, our efforts aim to discern fact from falsehood and fortify the foundations of trust upon which a cohesive society rests. Our commitment to truth, transparency, and collaborative innovation guides us toward a future where the integrity of information prevails, and the societal fabric remains unblemished by the shadows of deceit.

References

[1] D. Lazer, M. Baum, Y. Benkler, A. J. Berinsky, K. M. Greenhill, F. Menczer, M. J. Metzger, B. Nyhan, G. Pennycook, D. M. Rothschild, M. Schudson, S. Sloman, C. Sunstein, E. A. Thorson, D. Watts, and J. Zittrain, "The science of fake news," *Science*, vol. 359, pp. 1094 – 1096, 2018.

[2] Y. M. Rocha, G. A. de Moura, G. A. Desidério, C. H. de Oliveira, F. D. Lourenço, and L. D. de Figueiredo Nicolete, "The impact of fake news on social media and its influence on health during the covid-19 pandemic: a systematic review," *Journal of Public Health*, vol. 31, no. 7, pp. 1007–1016, 2023. [Online]. Available: https://doi.org/10.1007/s10389-021-01658-z

[3] O. D. Apuke and B. Omar, "Fake news and covid-19: modelling the predictors of fake news sharing among social media users," *Telematics and Informatics*, vol. 56, p. 101475, Jan 2021. [Online]. Available: https://doi.org/10.1016/j.tele.2020.101475

[4] C.-O. Cepoi, "Asymmetric dependence between stock market returns and news during covid-19 financial turmoil," *Finance Research Letters*, vol. 36, p. 101658, 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1544612320305912

[5] S. Vosoughi, D. Roy, and S. Aral, "The spread of true and false news online," *Science*, vol. 359, no. 6380, pp. 1146–1151, 2018.

[6] H.-C. H. Chang, S. Haider, and E. Ferrara, "Digital civic participation and misinformation during the 2020 taiwanese presidential election," *Media and Communication*, 2021.

[7] T.-L. Wang, "Does fake news matter to election outcomes?: The case study of taiwan's 2018 local elections," vol. 8, pp. 67–104, 2020.

[8] H. Allcott and M. Gentzkow, "Social media and fake news in the 2016 election," *Journal of Economic Perspectives*, vol. 31, pp. 211–236, 05 2017.

[9] E. Nisbet, C. Mortenson, and Q. Li, "The presumed influence of election misinformation on others reduces our own satisfaction with democracy," 2021.

[10] A. Oehmichen, K. Hua, J. A. D. López, M. Molina-Solana, J. Gómez-Romero, and Y.-K. Guo, "Not all lies are equal. a study into the engineering of political misinformation in the 2016 us presidential election," *IEEE Access*, vol. 7, pp. 126 305–126 314, 2019. [Online]. Available: https://api.semanticscholar.org/CorpusID:202688875

[11] M. F. Mridha, A. J. Keya, M. A. Hamid, M. M. Monowar, and M. S. Rahman, "A comprehensive review on fake news detection with deep learning," *IEEE Access*, vol. 9, pp. 156 151–156 170, 2021.

[12] R. Kozik, A. Pawlicka, M. Pawlicki, M. Choraś, W. Mazurczyk, and K. Cabaj, "A meta-analysis of state-of-the-art automated fake news detection methods," *IEEE Transactions on Computational Social Systems*, pp. 1–11, 2023.

[13] F. Mahmud, M. M. S. Rayhan, M. H. Shuvo, I. Sadia, and M. K. Morol, "A comparative analysis of graph neural networks and commonly used machine learning algorithms on fake news detection," *2022 7th International Conference on Data Science and Machine Learning Applications (CDMA)*, pp. 97–102, 2022.

[14] M. A. Alonso, D. Vilares, C. Gómez-Rodríguez, and J. Vilares, "Sentiment analysis for fake news detection," *Electronics*, vol. 10, no. 11, 2021. [Online]. Available: https://www.mdpi.com/2079-9292/10/11/1348

[15] H. Liu, W. Wang, H. Li, and H. Li, "Teller: A trustworthy framework for explainable, generalizable and controllable fake news detection," 2024.

[16] B. AlEsawi and M. Al-Tai, "Detecting arabic misinformation using an attention mechanism-based model," *Iraqi Journal For Computer Science and Mathematics*, vol. 5, pp. 285–298, 02 2024.

[17] Q. Liu, Y. Jin, X. Cao, X. Liu, X. Zhou, Y. Zhang, X. Xu, and L. Qi, "An entity ontology-based knowledge graph embedding approach to news credibility assessment," *IEEE Transactions on Computational Social Systems*, pp. 1–11, 2024.

[18] M. Babar, A. Ahmad, M. U. Tariq, and S. Kaleem, "Real-time fake news detection using big data analytics and deep neural network," *IEEE Transactions on Computational Social Systems*, pp. 1–10, 2023.

[19] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[20] H. Touvron, L. Martin, K. Stone, P. Albert, A. Almahairi, Y. Babaei, N. Bashlykov, S. Batra, P. Bhargava, S. Bhosale, D. Bikel, L. Blecher, C. C. Ferrer, M. Chen, G. Cucurull, D. Esiobu, J. Fernandes, J. Fu, W. Fu, B. Fuller, C. Gao, V. Goswami, N. Goyal, A. Hartshorn, S. Hosseini, R. Hou, H. Inan, M. Kardas, V. Kerkez, M. Khabsa, I. Kloumann, A. Korenev, P. S. Koura, M.-A. Lachaux, T. Lavril, J. Lee, D. Liskovich, Y. Lu, Y. Mao, X. Martinet, T. Mihaylov, P. Mishra, I. Molybog, Y. Nie, A. Poulton, J. Reizenstein, R. Rungta, K. Saladi, A. Schelten, R. Silva, E. M. Smith, R. Subramanian, X. E. Tan, B. Tang, R. Taylor, A. Williams, J. X. Kuan, P. Xu, Z. Yan, I. Zarov, Y. Zhang, A. Fan, M. Kambadur, S. Narang, A. Rodriguez, R. Stojnic, S. Edunov, and T. Scialom, "Llama 2: Open foundation and fine-tuned chat models," 2023.

[21] G. Team, T. Mesnard, C. Hardin, R. Dadashi, S. Bhupatiraju, S. Pathak, L. Sifre, M. Rivière, M. S. Kale, J. Love, P. Tafti, L. Hussenot, A. Chowdhery, A. Roberts, A. Barua, A. Botev, A. Castro-Ros, A. Slone, A. Héliou, A. Tacchetti, A. Bulanova, A. Paterson, B. Tsai, B. Shahriari, C. L. Lan, C. A. Choquette-Choo, C. Crepy, D. Cer, D. Ippolito, D. Reid, E. Buchatskaya, E. Ni, E. Noland, G. Yan, G. Tucker, G.-C. Muraru, G. Rozhdestvenskiy, H. Michalewski, I. Tenney, I. Grishchenko, J. Austin, J. Keeling, J. Labanowski, J.-B. Lespiau, J. Stanway, J. Brennan, J. Chen, J. Ferret, J. Chiu, J. Mao-Jones, K. Lee, K. Yu, K. Millican, L. L. Sjoesund, L. Lee, L. Dixon, M. Reid, M. Mikuła, M. Wirth, M. Sharman, N. Chinaev, N. Thain, O. Bachem, O. Chang, O. Wahltinez, P. Bailey, P. Michel, P. Yotov, P. G. Sessa, R. Chaabouni, R. Comanescu, R. Jana, R. Anil, R. McIlroy, R. Liu, R. Mullins, S. L. Smith, S. Borgeaud, S. Girgin, S. Douglas, S. Pandya, S. Shakeri, S. De, T. Klimenko, T. Hennigan, V. Feinberg, W. Stokowiec, Y. hui Chen, Z. Ahmed, Z. Gong, T. Warkentin, L. Peran, M. Giang, C. Farabet, O. Vinyals, J. Dean, K. Kavukcuoglu, D. Hassabis, Z. Ghahramani, D. Eck, J. Barral, F. Pereira, E. Collins, A. Joulin, N. Fiedel, E. Senter, A. Andreev, and K. Kenealy, "Gemma: Open models based on gemini research and technology," 2024.

[22] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," 2013.

[23] W. Y. Wang, ""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 (Volume 2: Short Papers)*, R. Barzilay and M.-Y. Kan, Eds. Vancouver, Canada: Association for Computational Linguistics, Jul. 2017, pp. 422–426. [Online]. Available: https://aclanthology.org/P17-2067

[24] PolitiFact. (2020) The principles of the truth-o-meter: Politifact's methodology for independent fact-checking. [Online]. Available: https://www.politifact.com/article/2018/feb/12/principles-truth-o-meter-politifacts-methodology-i/

[25] M. Gupta, D. Dennehy, C. M. Parra, M. Mäntymäki, and Y. K. Dwivedi, "Fake news believability: The effects of political beliefs and espoused cultural values," *Information & Management*, vol. 60, no. 2, p. 103745, 2023.

[26] g0v. (2019) g0v manifesto. [Online]. Available: https://g0v.tw/intl/en/manifesto/en/

[27] g0v Cofacts. (2018) Cofacts . [Online]. Available: https://beta.hackfoldr.org/1yXwRJwFNFHNJibKENnLCAV5xB8jnUvEwY_oUq-KcETU

[28] ——. (2018) Cofacts tutorial. [Online]. Available: https://en.cofacts.tw/tutorial?tab=bust-hoaxes

[29] A. Estornell, S. Das, and Y. Vorobeychik, "Deception through half-truths," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 06, 2020, pp. 10 110–10 117.

[30] Y. Zhou, P. Xu, X. Wang, X. Lu, G. Gao, and W. Ai, "Emojis decoded: Leveraging chatgpt for enhanced understanding in social media communications," *arXiv preprint arXiv:2402.01681*, 2024.

[31] Y. Li, H. He, J. Bai, and D. Wen, "Mcfend: A multi-source benchmark dataset for chinese fake news detection," *arXiv preprint arXiv:2403.09092*, 2024.

[32] N. Capuano, G. Fenza, V. Loia, and F. D. Nota, "Content-based fake news detection with machine and deep learning: a systematic review," *Neurocomputing*, vol. 530, pp. 91–103, 2023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0925231223001376

[33] Lik Xun Yuan, "distilbert-base-multilingual-cased-sentiments-student (revision 2e33845)," 2023. [Online]. Available: https://huggingface.co/lxyuan/distilbert-base-multilingual-cased-sentiments-student

[34] T. Wolf, L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi, P. Cistac, T. Rault, R. Louf, M. Funtowicz, J. Davison, S. Shleifer, P. von Platen, C. Ma, Y. Jernite, J. Plu, C. Xu, T. L. Scao, S. Gugger, M. Drame, Q. Lhoest, and A. M. Rush, "Transformers: State-of-the-art natural language processing," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*. Online: Association for Computational Linguistics, Oct. 2020, pp. 38–45. [Online]. Available: https://www.aclweb.org/anthology/2020.emnlp-demos.6

[35] B. Bhutani, N. Rastogi, P. Sehgal, and A. Purwar, "Fake news detection using sentiment analysis," in *2019 Twelfth International Conference on Contemporary Computing (IC3)*, 2019, pp. 1–5.

[36] L. Lu, "Dying relu and initialization: Theory and numerical examples," *Communications in Computational Physics*, vol. 28, no. 5, p. 1671–1706, Jun. 2020. [Online]. Available: http://dx.doi.org/10.4208/cicp.OA-2020-0165

[37] S. Zhang, Y. Wu, T. Che, Z. Lin, R. Memisevic, R. R. Salakhutdinov, and Y. Bengio, "Architectural complexity measures of recurrent neural networks," in *Advances in Neural Information Processing Systems*, D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett, Eds., vol. 29. Curran Associates, Inc., 2016. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2016/file/860320be12a1c050cd7731794e231bd3-Paper.pdf

[38] R. K. Srivastava, K. Greff, and J. Schmidhuber, "Highway networks," *CoRR*, vol. abs/1505.00387, 2015. [Online]. Available: http://arxiv.org/abs/1505.00387

[39] W. Zhang, Y. Deng, B. Liu, S. J. Pan, and L. Bing, "Sentiment analysis in the era of large language models: A reality check," *arXiv preprint arXiv:2305.15005*, 2023.

[40] C. Wittenberg and A. J. Berinsky, *Misinformation and Its Correction*, ser. SSRC Anxieties of Democracy. Cambridge University Press, 2020, p. 163–198.

[41] K. Simonyan, A. Vedaldi, and A. Zisserman, "Deep inside convolutional networks: Visualising image classification models and saliency maps," 2014.

[42] M. Sundararajan, A. Taly, and Q. Yan, "Axiomatic attribution for deep networks," 2017.

[43] F. Ikhwantri, J. W. G. Putra, H. Yamada, and T. Tokunaga, "Looking deep in the eyes: Investigating interpretation methods for neural models on reading tasks using human eye-movement behaviour," *Information Processing Management*, vol. 60, no. 2, p. 103195, 2023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0306457322002965

[44] N. Hollenstein and L. Beinborn, "Relative importance in sentence processing," 2021.

[45] M. Avram, N. Micallef, S. Patil, and F. Menczer, "Exposure to social engagement metrics increases vulnerability to misinformation," *ArXiv*, vol. abs/2005.04682, 2020.

[46] S. Lewandowsky and S. van der Linden, "Countering misinformation and fake news through inoculation and prebunking," *European Review of Social Psychology*, vol. 32, pp. 348 – 384, 2021.

[47] J. Roozenbeek and S. Linden, "Fake news game confers psychological resistance against online misinformation," *Palgrave Communications*, vol. 5, pp. 1–10, 2019.
[48] L. Huang, W. Yu, W. Ma, W. Zhong, Z. Feng, H. Wang, Q. Chen, W. Peng, X. Feng, B. Qin, and T. Liu, "A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions," 2023.
[49] J. Opitz, "From bias and prevalence to macro f1, kappa, and mcc: A structured overview of metrics for multi-class evaluation," 2022. [Online]. Available: https://api.semanticscholar.org/CorpusID:253270558
[50] M. Nagel, M. Fournarakis, R. A. Amjad, Y. Bondarenko, M. van Baalen, and T. Blankevoort, "A white paper on neural network quantization," 2021.
[51] E. J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, and W. Chen, "Lora: Low-rank adaptation of large language models," 2021.
[52] M. Pellert, C. M. Lechner, C. Wagner, B. Rammstedt, and M. Strohmaier, "Ai psychometrics: Assessing the psychological profiles of large language models through psychometric inventories," *Perspectives on Psychological Science*, vol. 0, no. 0, p. 17456916231214460, 0, pMID: 38165766. [Online]. Available: https://doi.org/10.1177/17456916231214460
[53] Wikipedia, "Tay (chatbot) — Wikipedia, the free encyclopedia," http://en.wikipedia.org/w/index.php?title=Tay%20(chatbot)&oldid=1212008146, 2024, [Online; accessed 22-March-2024].
[54] R. R. Mandical, N. Mamatha, N. Shivakumar, R. Monica, and A. N. Krishna, "Identification of fake news using machine learning," in *2020 IEEE International Conference on Electronics, Computing and Communication Technologies (CONECCT)*, 2020, pp. 1–6.

APPENDIX

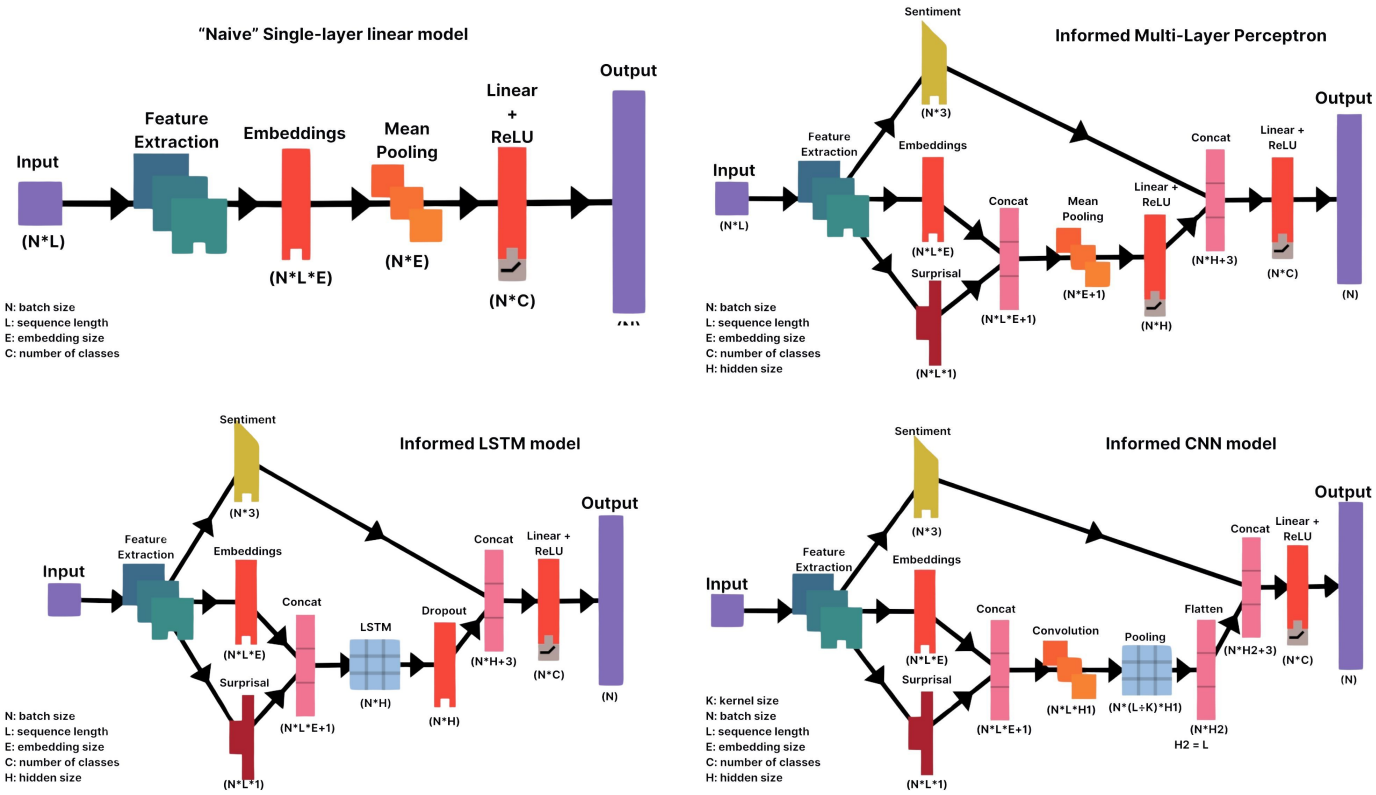

Fig. 2: Overview of the Classifiaction Head Architectures

Fig. 3: Cofacts Working Structure - Users submit articles, if they were already reviewed the user receives instant feedback. Otherwise, an editor reviews the article and uploads it to the database together with his reply.

Cofacts Prompt:

請根據以下新聞內容判斷新聞的真偽，並僅以數字0與1作答。回答1代表「假新聞」，0代表「真新聞」。新聞內容：{news_text}

English Translation: Please judge the truthfulness of the news according to the following news content, and answer only with the numbers 0 and 1. Answer 1 for "fake news" and 0 for "real news". News content: {news_text}

LIAR Prompt:

Please classify the following content into one of these six categories (and answer in only one of the six labels): true, mostly-true, half-true, barely-true, false, pants-on-fire.

Content: {content_text}

Fig. 4: Instructions for Prompting ChatGPT-4 on Cofacts and LIAR

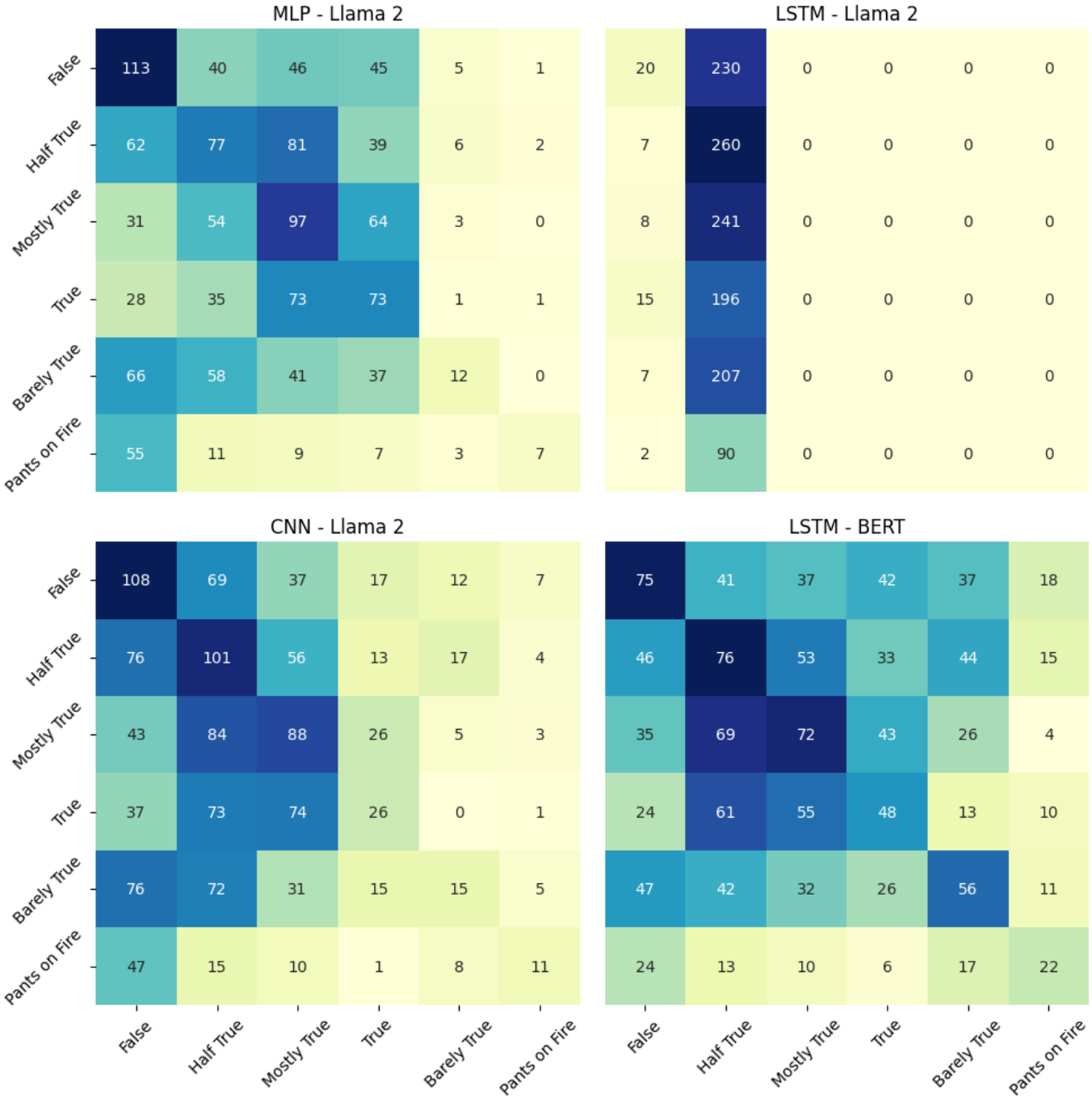

Fig. 5: Confusion Matrices - Results are shown for the MLP, LSTM, and CNN heads trained on the LIAR dataset with Llama 2 as the base model. We also include the results of the LSTM with BERT as a base model for comparison.