Information Fusion

Fake News Detection: Taxonomy and Comparative Study

--Manuscript Draft--

Manuscript Number:	INFFUS-D-23-00797R1
Article Type:	VSI: DisINFUS
Keywords:	disinformation; misinformation; Machine learning; Deep learning; Natural language processing; Fake news detection
Corresponding Author:	Faramarz Farhangian, M.D École de technologie supérieure Montreal, Quebec CANADA
First Author:	Faramarz Farhangian, M.D
Order of Authors:	Faramarz Farhangian, M.D
	Rafael M. O. Cruz, Assistant Professor
	George D. C. Cavalcanti, Full Professor
Abstract:	The proliferation of social networks has presented a significant challenge in combating the pervasive issue of fake news within modern societies. Due to the large amount of information and news produced daily in text, audio, and video, the validation and verification of this information have become crucial tasks. Leveraging advancements in artificial intelligence, distinguishing between fake news and factual information through automatic fake news detection systems has become more feasible. Automatic fake news detection has been explored from diverse perspectives, employing various feature extraction and classification models. Nonetheless, empirical evaluations, categorization, and comparisons of existing techniques for handling this problem remain limited. In this paper, we revisit the definitions and perspectives of fake news and propose an updated taxonomy for the field based on multiple criteria: 1) Type of features used in fake news detection; 2) Fake news detection perspectives; 3) Feature representation methods; and 4) Classification approaches. Moreover, we conduct an extensive empirical study to evaluate several feature representation techniques and classification approaches based on accuracy and computational cost. Our experimental results demonstrate that the optimal feature extraction techniques vary depending on the characteristics of the dataset. Notably, context-dependent models based on transformer models consistently exhibit superior performance. Additionally, employing transformer models as feature extraction methods, rather than solely fine-tuning the network for the downstream task, improves overall performance. Through extensive error analysis, we identify that a combination of feature representation methods and classification algorithms, including classical ones, offer complementary aspects and should be considered for achieving better generalization performance while maintaining a relatively low computational cost. For further details, including source codes, figures, and datasets, please refer to our project's GitHub repository: [https://github.com/FFarhangian/Fake-news-detection-Comparative-Study].
Suggested Reviewers:	Paolo Rosso prosso@dsic.upv.es
	David Camacho david.camacho@upm.es
Response to Reviewers:	

Monday 16th October, 2023

Manuscript Number: INFFUS-D-23-00797

Manuscript title: All about automatic fake news detection: A wide comparative study

Authors: Faramarz Farhangian, George D. C. Cavalcanti, Rafael M. O. Cruz

Dear Dr. Salvador Garcia,

As recommended in the received Decision Letter after the first round of reviews, we have fully revised our submitted paper to your journal, Information Fusion, according to the constructive advice provided by the anonymous reviewers, and we are resubmitting the new version of our manuscript to be considered for publication. We would like to thank the anonymous reviewers for their insightful high quality comments that certainly helped us to enhance our paper.

In the following pages, we respond to each reviewer's comment: we reproduce the original comments made by the reviewers (shown in **boldface**) and we introduce after "**Answer to Comment #X**", our remarks in blue color on how the specific comment #X of the reviewer has been addressed in the revised version of the paper. Moreover, specific changes to the paper are presented in quotes, and highlighted in italic.

Yours sincerely,

Faramarz Farhangian (faramarz.farhangian.1@ens.etsmtl.ca)

Corresponding author

COMMENTS FOR THE AUTHOR:

Reviewer #1

Comment #1: The paper provides an updated taxonomy for fake news detection models, covering feature extraction methods and perspectives. It conducts an empirical study comparing various techniques, including transformer models, and analyzes their performance. The paper suggests combining different classifiers and feature extractors for better predictions. It also considers cost-effectiveness and offers directions for future research. I think this work is meaningful and provides experience in the field of fake news detection.

Answer to Comment #1:
Thank you for your insightful comments and for recognizing the significance of our work in the field of fake news detection. We are pleased to know that you found the empirical study valuable and appreciate the acknowledgment of our suggestion to combine different classifiers and feature extractors for enhanced prediction. We sincerely appreciate your support as well as your feedback to improve the manuscript.

Comment #2: The baseline model used in the paper does not include the large language models like LLaMA. The authors should add some recent large language models. And some additional visualization experiments would be better.

Answer to Comment #2: Thank you for your insightful feedback regarding the inclusion of recent large language models in our study.

In response to your comment, we incorporated the **LLaMA** [Touvron et al., 2023] and **Falcon** [Penedo et al., 2023] models into our experiments. The Falcon model, renowned for its efficient text encoding capabilities, and the LLaMA model, known for its impressive multilingual support, have been examined thoroughly in the context of our research. These integrations provide a richer understanding and a more robust comparison in the realm of fake news detection.

To elaborate on the specific updates made to our paper:

1. **Figures Updates:** Figures 6 and 7 have been revised to incorporate the newly added models, providing visual representation and comparative metrics.

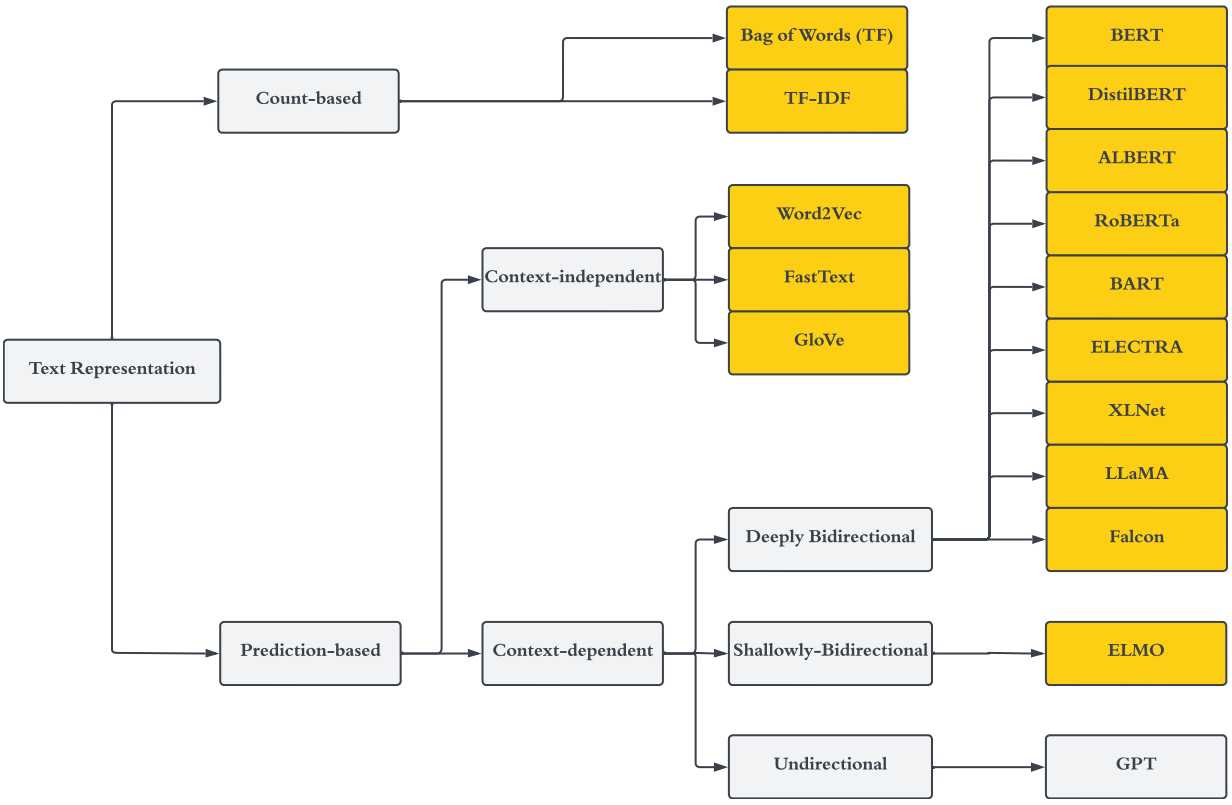


Figure 1: Taxonomy of text representation methods. The methods are highlighted in orange since it is the focus of this paper.

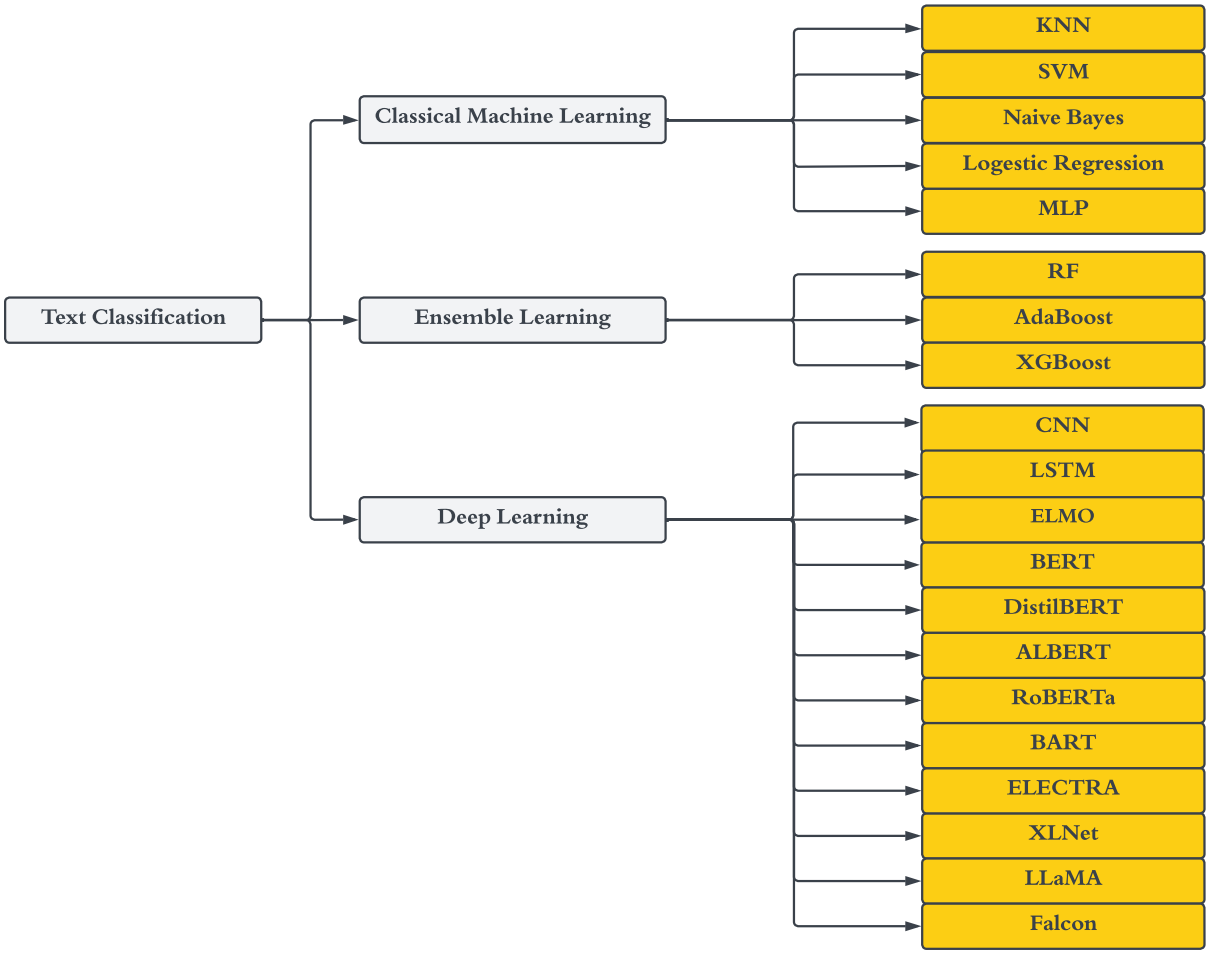


Figure 2: Taxonomy of text classification methods. The methods are highlighted in orange since it is the focus of this paper.

2. **Methodology Enhancement:** In Section 3.2.2.2, titled 'Context-dependent methods', we have included two new subsections detailing the architecture and unique features of both LLaMA and Falcon models.

   LLaMA *[Touvron et al., 2023] is a cutting-edge language model which is available in a range of sizes, ranging from 7 billion to 65 billion parameters, with each size trained on various token quantities, such as 1 trillion and 1.4 trillion tokens [Touvron et al., 2023]. LLaMA trained based on texts from the 20 most widely spoken languages, especially those written in the Latin and Cyrillic scripts. A few examples of its training data sources are public websites and forums like CommonCrawl, GitHub repositories, Wikipedia in many languages, Project Gutenberg's public domain books, scientific publications from ArXiv, and Q&A from Stack Exchange websites. LLaMA just like other large language models uses next-word prediction algorithms in the training phase. LLaMA is flexible, accommodating a wide range of use cases, and also shows enhanced performance in several natural language processing tasks. LLaMA model has been shown to outperform other open language models in multiple areas according to the Open LLM Leaderboard*[1] *.*

   Falcon *[Penedo et al., 2023] is a large language model that the Technology Innovation Institute (TII) in the UAE has developed. It consists of the Falcon-7B and Falcon-40B models, where the digits represent the number of parameters in each model. Notably, Falcon-40B has outperformed GPT and LLaMa based on the Open LLM Leaderboard. Falcon is trained by high-quality training data, which is mostly drawn from RefinedWeb, a sizable online dataset created from CommonCrawl [Penedo et al., 2023] . By using the multi-query attention method, it improves the model architecture for inference and paves the way for creative applications. Falcon is also open source and transparent.*

3. **Algorithm Updates:** The 'Classification Algorithm' section (Section 3.3) has been updated to incorporate the configuration specifics for the two newly integrated LLMs.

   There are several transformer models available, including ELMO, BERT, DistilBERT, ALBERT, RoBERTa, BART, ELECTRA, XLNet, LLaMA, and Falcon, which are commonly used in text classification tasks. According to recent literature, transformers have also been successfully used for fake news detection, which motivates their usage in this comparative analysis.

4. **Comparative Studies and Experimental Setup:** Both these sections now encompass discussions and setups involving the LLaMA and Falcon models, ensuring a holistic experimental approach.

   In this section, we address the research questions posed by this study through an empirical comparison of 20 state-of-the-art classification algorithms and 15 feature representation techniques. The classification models include 5 classical machine learning models, 3 ensemble models, and 12 deep learning models (including recent large language models such as Falcon and LLaMA). Also, the feature representation techniques include 2 count-based methods, 3 context-independent methods, and 10 context-dependent methods. It is worth highlighting that we used transformer models in both feature extraction and end-to-end classification models, and all models have been evaluated under the same experimental protocol over four benchmark fake news datasets. Figure 1, 2 show the details of the models used in this experiment.

   From context-dependent methods, we include BERT ('bert-base-uncased'), DistilBERT ('distilbert-base-uncased'), ALBERT ('albert-base-v2'), RoBERTa ('roberta-base'), BART ('facebook-bart-large'), ELECTRA ('google-electra-base-discriminator'), XLNet ('xlnet-base-cased'), Falcon ('Rocketknight1/falcon-rw-1b'), ELMO ('elmo2') and LLaMA that we obtained the weights by sending an email request, completing the necessary form, and subsequently converting them to the Hugging Face Transformers format. All the implementation of transformers was facilitated by the Hugging Face framework[2] *[Wolf et al.],*

[1] `https://llm-leaderboard.streamlit.app/`
[2] `https://github.com/huggingface/transformers`

Tensorflow Hub[3] and PyTorch [Paszke et al., 2019] version 1.11.0. All transformer models yield fixed feature vectors of 768 dimensions extracted from their respective pooling layers, with the exception of ELMO, LLaMA, and Falcon, which use feature vectors of 1024, 2048, and 4096 dimensions, respectively. In this experiment, we aggregate the hidden state of the last four layers and freeze the parameters of the initial layers.

5. **Results and Conclusion:** The addition of these models naturally impacted our results. Consequently, we revisited our "esults" and "Conclusion" sections to take into account for the addition of the LLMs. All tables and figures have been revised to provide updated and information.

Comment #3: For better presentation, the author should improve the figures and tables, e.g. Figure 3, 4.

Answer to Comment #3:

Thank you for pointing out the areas of improvement concerning the presentation of our figures and tables. We agree that some figures and tables needed improvements. As such we revisited Figures 3 and 4, as mentioned, to ensure they are clearer and presented in a manner that aids in better comprehension. In particular we fixed several alignment problems that occurred in the previous version, increase the font size for better readability and also changed the color scheme and highlighting for a better contranst and visibility.

The new Figures 3 and 4 are presented below

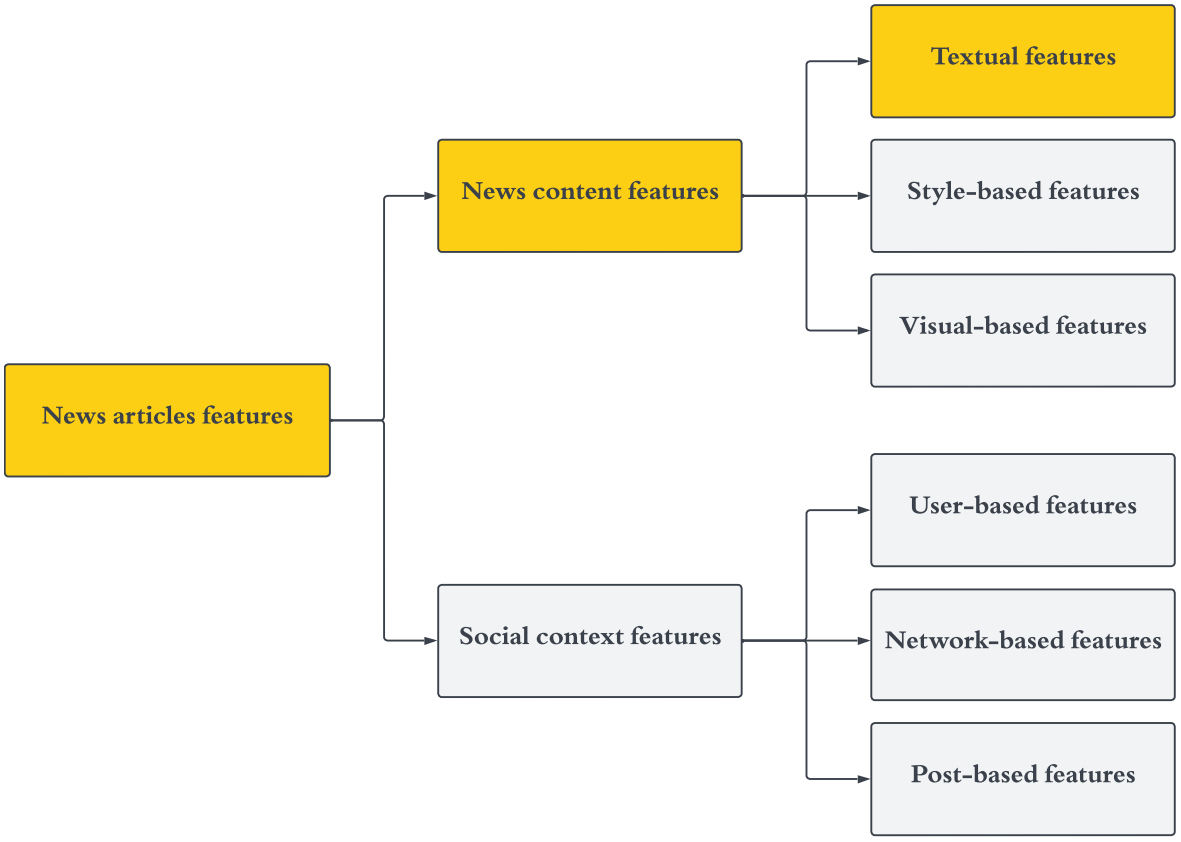


Figure 3: Type of features in fake news detection problem. The highlighted in orange shows the focus of this research.

[3] https://www.tensorflow.org/hub

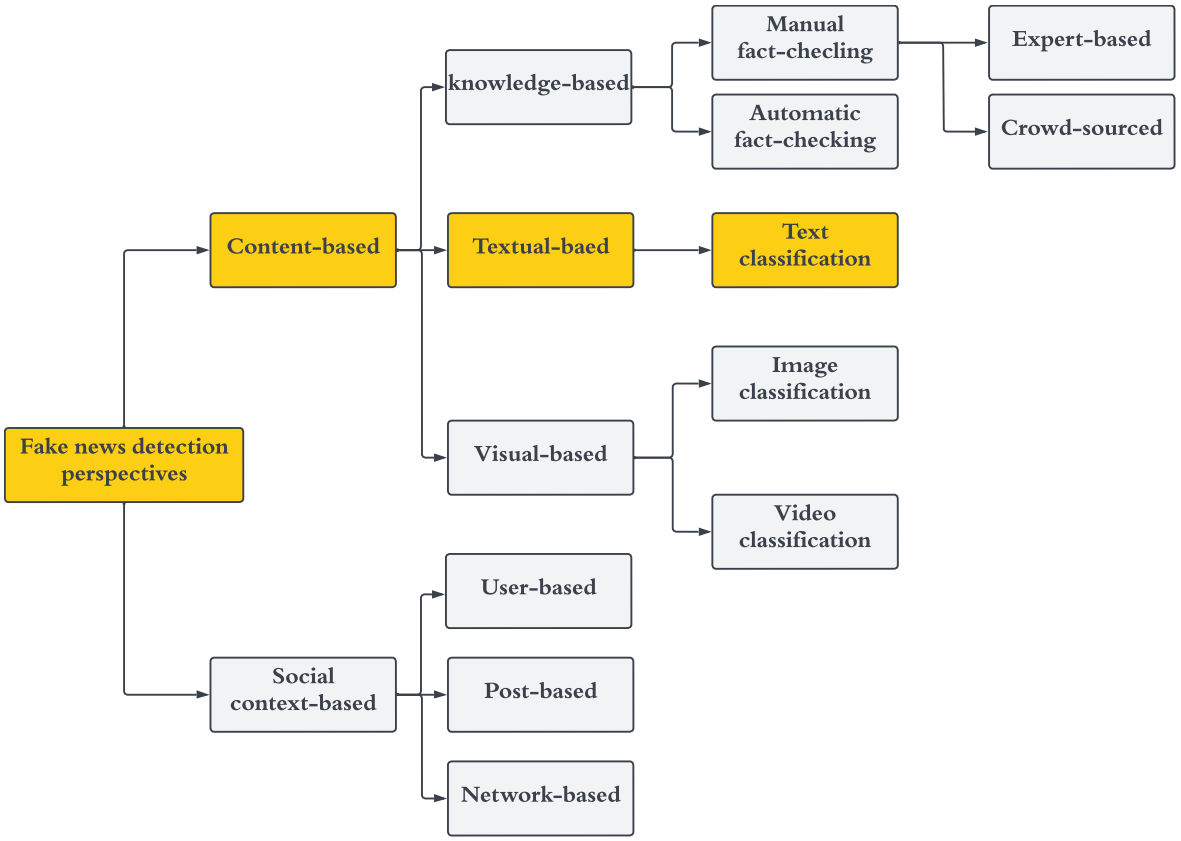


Figure 4: Fake news detection perspectives. The highlighted in orange shows the focus of this research.

In addition to changes to Figures 3 and 4, several other figures and Tables in the manuscript were modified or completely changes (As also pointed out by Reviewer #2 in comment #XXX). Thus, we believe the new figures and overall paper presentation are now aligned with the overall quality of the paper.

Reviewer #2

Comment #1: **The theoretical background is closer to a survey of the different methods (text-based and non-text-based) for fake news classification than to the analysis of all text-based classification techniques. The structure and explanation would have been perfect for that kind of paper, but given the scope of this attempted publication, the non-text-based parts should have been reduced to give more importance to the text-based focus. As a consequence, some weaknesses can be seen in the parts that do not cover the specialization of text-based methods. For example, in the image-based methods, no more explanation after the conventional use of CNNs is given, missing all the recent advances that represent the evolution of methods for visual features nowadays. Furthermore, this generic view of everything goes against a more detailed background that goes paper by paper showing the advances in terms of classifying fake news and justifying why the approach taken for this article is necessary.**

Answer to Comment #1: Thank you for your insightful observations regarding the structure and focus of our paper. We understand the importance of adhering strictly to the scope of our intended publication, especially concerning the specialized topic of text-based fake news detection techniques.

Based on your feedback, we have undertaken the following modifications to ensure a sharper focus on textual-based methods in the theoretical background:

- **Restructured the Background Section:** Recognizing the importance of a specialized focus, we have condensed the non-textual-based methods into a condensed text. This allows us to give an overview without diverting the readers' attention from the primary focus.

- **Enhanced Detailing for Text-Based Techniques:** To emphasize the significance and depth of our paper's primary focus, we have dedicated an extensive section to text-based fake news detection. This section elucidates on preprocessing, feature extraction methods, and models, among other essential aspects.
- **Removal of Unnecessary Details:** We have pruned certain sections, especially those pertaining to image-based methods using conventional CNNs. Instead, we've incorporated a concise overview, ensuring that our readers are informed but not sidetracked from the main content.
- **Streamlined Theoretical Background:** By concentrating primarily on textual-based methods, we've been able to delve deeper into the subject matter, discussing advances in a sequential manner. This approach not only brings clarity but also establishes the necessity of our chosen method for this article.

The more streamlined text, condensing the previous background section is presented below:

> *Fake news can be detected with the help of features extracted from the news articles, and these features can be categorized into two main approaches: content features and social context featuresShu et al. [2017]. Figure3 shows the summary of the available features in fake news detection applications. The goal of the content-based approach is to find signs or patterns through the news content features that are extracted from the body of the article like text, image, or video [Zhou and Zafarani, 2020, Bondielli and Marcelloni, 2019]. These types of features are divided into Textual features, Style-based features, and Visual-based features [Shu et al., 2017]. So, we can study content-based fake news detection approaches from four main perspectives based on the type of features mentioned above: Textual-based, Style-based, Visual-based, and Knowledge-based. Figure 4 summarizes these approaches.*
>
> *It can often be difficult to distinguish fake news based just on its content; but, any success in doing so may encourage fake news authors to adopt further precautions in the future in order to fool the detection system. So, social context features might be integrated into news analysis to improve the detection algorithm's resistance to such an attack [Shu et al., 2019a].The methods that use social context features in order to distinguish between fake news and factual news are social context-based approaches. Social context features imply the features that are extracted from marginal information such as user information, network information and/or propagation, and the reaction of other users [Bondielli and Marcelloni, 2019]. Social context-based approaches are divided into three main categories: user-based, post-based, and network-based [Shu et al., 2017].*
>
> *There are also a few studies that used the hybrid perspective for the detection of fake news. From a hybrid perspective, different characteristics of fake news have been combined to increase the power and accuracy of fake news detection models. To be more specific, they used multi-modal machine learning in order to develop models capable of handling and relating data from several modalities. In this regard, there are some studies that processed news content features and social context features at the same time [Raj and Meel, 2021, Qian et al., 2021, Li et al., 2021, Fung et al., 2021, Tuan and Minh, 2021, Sharma and Garg, 2021, Song et al., 2021, Wang et al., 2021, Wu et al., Zhou et al., 2019, Birunda and Devi, 2021, Shu et al., 2019b, Nikiforos et al., 2020, Zhang et al., 2019, Silva et al., 2021, Nguyen et al., 2019, Dong et al., 2018, Smith et al., 2020]. As this article specifically focuses on textual-based features, we will only elaborate on these features.*

We genuinely believe that these revisions have heightened the clarity, coherence, and depth of our paper, making it a more focused contribution to the literature on fake news detection.

Comment #2: The absence of the words "misinformation" and "disinformation" is notorious given that they appear as keywords of the paper and in the bibliography cited at the end. Regardless of the diverse views of the expression "fake news", the theory makes a correct explanation with references to why this word is used, but this is only motivated by the goal of this paper. The more complex reality of this problem of false information and, thus, the use of the words "misinformation" or "disinformation" should also be part of papers like this. This would also introduce the question of why the focus is on fake news rather than on disinformation in general when disinformation is also intentional but it is

not necessarily restricted to the shape of what it is conceived as news.

Answer to Comment #2:

Thank you for your thoughtful observations concerning the terminological focus of our paper. We recognize the broader landscape of information challenges, including both "misinformation" and "disinformation." Your comment provides a crucial opportunity to clarify our choice of focus and its pertinence to the broader discussion on information veracity.

- The term "fake news" has garnered significant attention over the past few years due to its pervasive impact on social media, public opinion, and even election outcomes. Given its profound societal implications, it is crucial to study and address the proliferation of fake news, which frequently manifests in a more tangible and recognizable format than the broader categories of misinformation or disinformation. The repercussions of unchecked fake news extend beyond individual beliefs and can have profound societal impact. It can influence societal behaviors, and political outcomes, and undermine trust in legitimate news sources. In our contemporary digital age, where news consumption largely happens through online platforms, addressing fake news is not just a matter of information accuracy but also of maintaining the integrity of our democratic institutions.

- Misinformation is "false information that is spread, regardless of whether there is intent to mislead." Disinformation is "deliberately misleading or biased information; manipulated narrative or facts; propaganda." Fake news is "purposefully crafted, sensational, emotionally charged, misleading or totally fabricated information that mimics the form of mainstream news". At its core, fake news is a subset of disinformation. While disinformation is the intentional spread of false information, regardless of the format or platform, fake news pertains specifically to fabricated content presented as news. Misinformation, on the other hand, may not always be spread with malicious intent, distinguishing it from the former terms. Thus, our focus on fake news narrows the scope to a specific, prevalent form of disinformation that has readily identifiable attributes and patterns, making it an ideal candidate for in-depth study.

- While we wholeheartedly acknowledge the importance of addressing the broader spectrums of misinformation and disinformation, our choice to focus on fake news was deliberate. As attested by comments from Reviewers #1 and #3, our contribution in this relevant area is valuable and timely due to its immense social impact. Furthermore, the depth of our analysis on fake news which, in this new version includes state-of-the-art Large Language Models (LLMs) such as Falcon and LLaMA, provides a foundational understanding that can, in turn, aid broader efforts to combat misinformation and disinformation at large.

  Hence, we update Section 2.1 by improving the fake news description, explaining how it differs from misinformation as well the distinctions among various concepts related to untruthful or deceptive information:

  > So, according to this definition, the *Satire*, *Rumors*, *Hoaxes*, *Misinformation*, and *Disinformation* concepts are distinct from fake news. Satire news is not intended to confuse or mislead users, and Rumors lack an unambiguous intent, may lack factual proof or verification, and are not always considered as news [Zhou and Zafarani, 2020]. Hoaxes that are just intended for entertainment or to defraud certain persons. Among all these concepts, misinformation, and disinformation are more close to fake news concepts. Misinformation in contrast with disinformation and fake news may not spread false information with malicious intent. Disinformation and fake news on the other hand intentionally circulate false information, however fake news is manufactured and presented as news, whereas disinformation is not necessary to be news.

We also, we added the difference between misinformation and disinformation with fake news to the introduction section.

Comment #3: The background unveils the lack of a proper structure and a focus on what matters for the scope of this journal: the effectiveness of mixing classifiers/feature extractors (the main goal) once it has been demonstrated that feature extraction with traditional methods also works for the proper combination of techniques (the steps to justify the main goal). Having followed the narrative of presenting an innovative review of all the existing

methods constitutes a mistake, since the Transformer models analyzed are closer to 2020 than to 2023 and the fine-tuning of the pre-trained models through the chosen datasets has not been deeply explored, and even more text-based implementations such as, for example, in data augmentation, could also be explored (rather than with just cross-validation). As a consequence, the title "All about automatic fake news detection: A wide comparative study" does not cover "all about" this issue, it is not multimodal and it is too generic to reveal any advance (and, thus, does not make justice to all the effort behind the paper). The lack of this journey towards a main goal (for example, the potential of the combination of techniques for this journal) in the attempt to cover all the domains of fake news text classification may constitute a disadvantage that would be emphasized in the Future Directions section, composed of many subsections that may demonstrate the absence of a focus.

Answer to Comment #3:

Thank you for your detailed feedback regarding the structure and focus of our background section.

- Firstly, the primary aim of our paper is to provide an updated taxonomy for fake news detection models, delving into various feature extraction methods and perspectives. By conducting an empirical study that compares different techniques, including transformer and recent LLMs, our intention is to provide a global view of the current landscape. Our findings indeed highlight the potential for enhancing fake news detection. Specifically, our results indicate the advantages of investigating models trained over diverse feature representations. This taps into their complementary power, building a robust ensemble of classifiers to significantly boost fake news detection accuracy. This approach is reinforced by our computational cost analysis, which underlines that certain classical techniques offer an optimal balance between accuracy and computational efficiency. They hold promise for ensemble models when trained over unique feature representations.

- Secondly, we acknowledge your concern regarding the relevance of transformer models from 2020. To ensure the up-to-date nature of our work, we incorporated two recent large language models, Falcon and LLaMA, into our study. Their addition will not only enrich our investigation but also align our study with the latest developments in the field. To the best of our knowledge, this is the first work to consider all these rence LLMs in a large experimental study.

- Regarding the title, we concur with your observation. The current title might be overly broad and not indicative of the depth and nuances of our study. We are considering a more fitting title, *"Fake News Detection: Taxonomy and Comparative Study"*, to better reflect the contents and contributions of our paper.

- We understand that a paper's focus is crucial. While our intention was to provide a thorough exploration, it is evident that certain areas may benefit from more depth rather than breadth. We refined our approach, particularly in the Future Directions section, as also requested by Reviewer #3 in comment #3, highlighting the main challenges, limitations of current methods.

Comment #4: There might also be concerns about the datasets used in the paper. As expressed in the article, although "caution is advised when evaluating the effectiveness of techniques based only on how well they perform on datasets like ISOT", ISOT has been also chosen as one of the datasets for the experiments.

Answer to Comment #4: Thank you for raising concerns regarding the datasets used in our research, especially with respect to ISOT.

Our decision to include the ISOT dataset was driven by our belief in comprehensive evaluation. While we did note that "caution is advised when evaluating the effectiveness of techniques based solely on how well they perform on datasets like ISOT", we also emphasized the importance of analyzing a technique's robustness, generalizability, and applicability across a variety of settings and datasets.

Using multiple datasets, including ISOT, serves to underscore this point. By evaluating feature extraction methods and classification models over different datasets under a consistent experimental protocol, we aim to offer a holistic view of their performance. This approach helps mitigate the potential pitfalls of over-relying on a single dataset and provides a more complete understanding of the model's strengths and limitations.

Furthermore, including ISOT ensures our evaluations are in line with other recent research that has also employed this dataset [Hakak et al., 2021, Goldani et al., 2021a, Jiang et al., 2021, Goldani et al.,

2021b, Rajalaxmi et al., 2022, Gravanis et al., 2019, Kaliyar et al., 2021, Nadeem et al., 2023], enabling more direct comparisons.

In conclusion, the inclusion of the ISOT dataset, when viewed within the broader context of our experimental approach, enhances the comprehensiveness of our analysis, fostering a deeper understanding of the techniques and their applicability in diverse scenarios.

Comment #5: The number of references should be coherent. For example, the weight that languages should have in this following sentence might vary for the ones given in the article through the number of citations: "Still, recent advances have also been made in many other languages like Slovak [62, 63, 62], Urdu [28, 64, 65, 66, 28, 67, 68, 69], Portuguese [70, 71, 72, 73], Korean [74, 75], Indian [76], German [77, 78], Spanish [79, 80], Bengali [81, 82], Chinese [83], Indonesian [84]". Moreover, it is recommended not to abuse multiple references together if they can be separated. For instance, the sentence "In this regard, several survey studies have focused on different aspects of fake news detection [1, 8, 16, 6]" claims for an enumeration of these aspects to include a reference in each of them, rather than all together. Furthermore, numerical order in the citations is expected, too ("[1, 6, 8, 16]").

Answer to Comment #5: Thank you for the insightful comment regarding the coherence and structuring of our references. We understand the importance of maintaining clarity and aiding the reader in understanding the weight and relevance of each citation. Below, we detail our approach to addressing the mentioned concerns.

- First of all, fix the redundancy in citations. Also, the disparity in the number of citations across different languages arises from our search strategy. As detailed in the response to Reviewer #3's comment #2, we employed a systematic search strategy with specific inclusion and exclusion criteria, one of which was focusing on papers written in English. This strategy may have resulted in some languages having a more significant representation than others in our reference list. Notably, the varied number of references can also emphasize the difference in the volume of research dedicated to fake news detection across different languages. To address the potential confusion this could cause for readers, we have added our search strategy in Appendix D.
- Our intention was to highlight that all referenced works focus on similar aspects, such as definitions, features, and techniques. We understand that clustering references together can make it unclear which specific contribution comes from each paper. In response, we have rephrased the sentence to provide clarity and have also corrected the order of the citations as recommended. And we've revised it to elucidate each study's particular focus in the Introduction section.

  > *In this regard, several survey studies have focused on fake news definition, data collection challenges, features, and techniques [Shu et al., 2017, Bondielli and Marcelloni, 2019, Zhou and Zafarani, 2020, Zhang and Ghorbani, 2020].*

- Regarding numerical order in citations, we have revised all the citations throughout the paper to ensure they appear in numerical order for ease of reference.

Comment #6: Visualizations are not refined for this paper. Table 8 is not easily comprehensible for the reader after having followed a structure of tables based on checking which column has the best score for each row and after having changed from f1 scores to ranks. Moreover, the main findings described after this are not visually understandable. More elaborated visualizations should be explored to make the reader keep the conclusions of each table (and thus, each dataset) and understand them after putting all the ranks together.

Answer to Comment #6: We sincerely appreciate the feedback regarding our visualizations. It is important that our readers can easily comprehend and retain the key takeaways from our visual data representations. To address the concerns, we implemented a series of modifications across the manuscript:

- Table 4,5,6,7, and 8 have been systematically revised to maintain a consistent structure. This will aid readers in quickly identifying patterns and key results across different tables.

- To provide a more intuitive understanding of which method outperforms others, we highlighted the best scores in each column. This will allow readers to quickly identify the top-performing metrics.
- We incorporated more descriptive labels, legends, and annotations to our visualizations to ensure they're interpretable even without extensive reference to the main text.
- We provided a separate bar chart to rank models and feature extraction methods to provide a more intuitive understanding of the models and feature representations ranking. Figure 5 shows them.

We believe that these modifications substantially improve the comprehensibility and effectiveness of our visual representations.

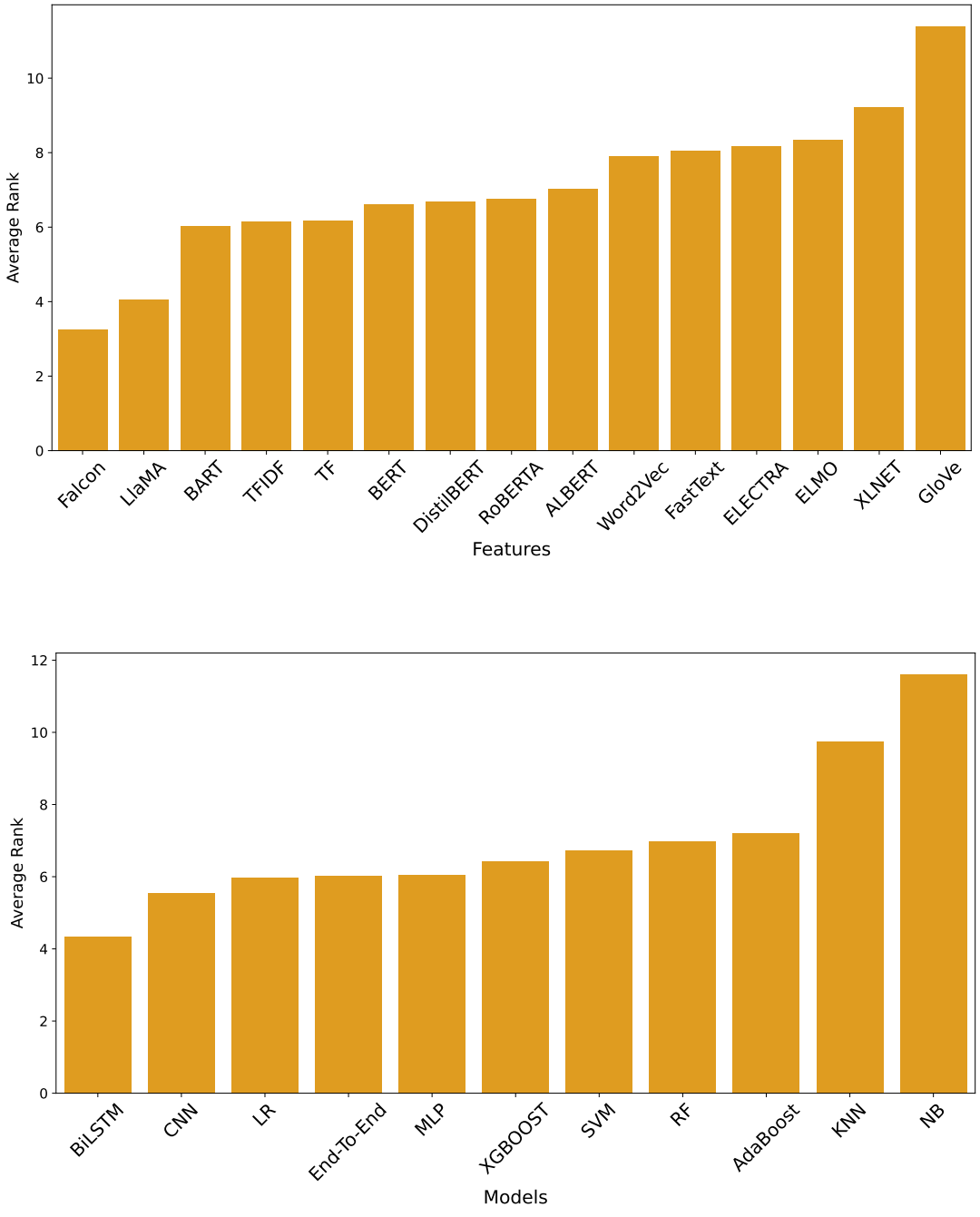


Figure 5: Average rank of methods based on different datasets. Graph (a) plots the average rank of different feature extraction methods over four datasets. Graph (b) plots the average rank of different classification models over four datasets. The bar charts have been sorted from smaller ranks to larger ranks.

Comment #7: On this matter, attention to tables and plots should be paid in general: on the one hand, the name 'Number of feature methods which the made same error" in some columns makes no sense; on the other hand, the grid of scatterplots for the cost-effectiveness (Figure 9) is not clear since every y-axis range from 0 to 1, so again there should be better ways to represent the information from this visualization instead of needing the text to understand the results.

Answer to Comment #7: Thank you for pointing out the clarity issues in our tables and visualizations. We recognize the importance of ensuring that figures and tables are intuitive and self-explanatory. Here's how we've addressed your concerns:

- We understand the confusion caused by the column name "Number of feature methods which the made same error". To address this, we've revised it to "Count of Feature Methods Sharing the Same Error". This rephrasing provides a clearer idea of what the column represents: counting how many feature extraction methods, when paired with various models, produced the same errors.
- Regarding Figure 9 (Cost-Effectiveness Scatterplots), The range between 0 and 1 on the y-axis for the scatterplots is intentional, given that the F1-score lies between this range. However, we understand that the uniform scaling might make the plots look visually dense and might obscure specific data distributions. To address this, we have adjusted the scales to fit the data distribution for each plot better. The updated figure is presented below:

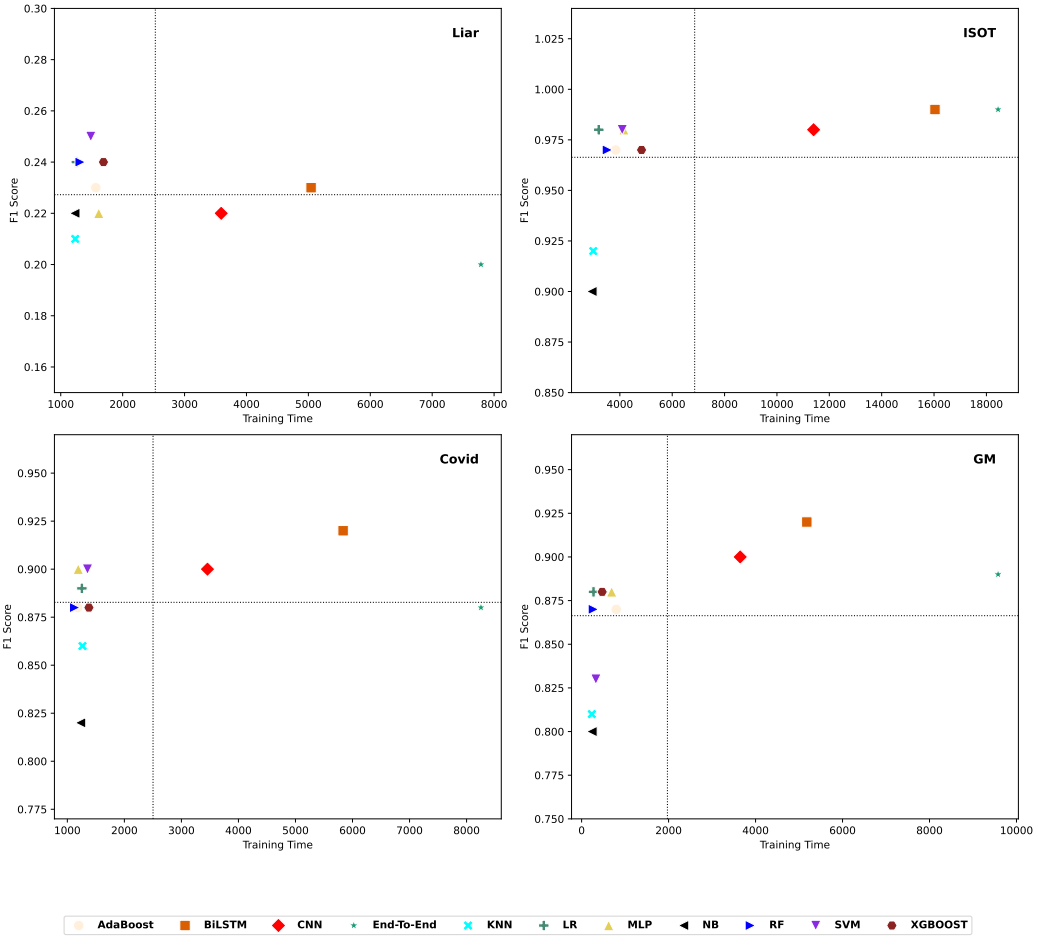


Figure 6: Models cost-effective visualization. This graph plots the average performance of models (F1-score) versus the average training time (second) of models. different colorful shapes show the models, and the dashed lines in the plots show the models' average performance and average training time in each dataset.

Comment #8: "Tables 4, 5, 6, and 7 The tables", "that numerous combinations", "miss classification" and both "Liar" and "LIAR" in the same text, among other examples, show

that grammar and consistency are not fully reviewed.

Answer to Comment #8:
Thank you for drawing our attention to these oversights in the manuscript, especially concerning grammar and consistency. We acknowledge the inconsistency in the use of the terms "Liar" and "LIAR". To maintain uniformity, we will ensure that the naming remains consistent throughout the document, So we just use "Liar". Furthermore, a full revision of the text was performed, improving language, and refining the text to be more technical and precise.

Overall, we sincerely value your feedback and believe that these corrections significantly enhance the quality of our paper.

Reviewer #3

Comment #1: The paper provides a broad and comprehensive overview of the current state of textual-based fake news detection, with empirical comparison, error analysis and effectiveness analysis. Overall, the paper is very good, and provide an interesting overview of fake news detection.

Answer to Comment #1: We deeply appreciate your constructive feedback on our paper. Your acknowledgment of our effort to present a broad and comprehensive overview of textual-based fake news detection. It has been our priority to provide an empirical comparison, error analysis, and effectiveness analysis in a cohesive manner. We are pleased to know that the paper resonates with its intended purpose and provides value in understanding fake news detection. We believe that the paper after this revision is much better than its original version.

Comment #2: Search Methodology: For a survey paper, the search methodology is crucial. The paper could benefit from a more transparent and detailed explanation of how the literature was sourced, the criteria for including or excluding certain works, and the databases or search engines used. Without a clear search methodology, there's a risk of unintentional bias in the selection of sources.

Answer to Comment #2: We would like to thank the reviewer for pointing out insightful suggestions to improve the literature reviews by adding search methodology. We added the search methodology, including search queries, including and excluding criteria and databases, into **Appendix D**. Our modifications are highlighted in bold.

> ***Search Methodology:***
>
> *We conducted a search methodology to find relevant primary and secondary studies related to fake news detection. In the first step, based on the research questions and objectives of our research, we used specific terms and keywords to create search queries including,* "Fake news detection", "Machine learning", "Deep learning", "Ensemble", "NLP", "Text classification". *In the second step, we searched the queries that were created based on the keywords in the most popular databases and digital libraries including Google Scholar, ACM digital library, and IEEE Explore. In the third step, we extract and classify the information of query results. Finally, we screened the results based on some inclusive and exclusive criteria in order to reduce the number of results.*
>
> - *Inclusive Criteria: English papers, papers related to fake news detection methods including machine learning, deep learning, and ensemble learning, peer-reviewed journal articles and conferences, papers with a publication date from 2014 to 2022.*
> - *Exclusive Criteria: Non-peer-reviewed journal articles, non-English papers, lake of clear methodology and results, low impact papers including low citation and impact factor, outdated papers with older than 10 years, duplicate records.*

Moreover, to enhance transparency, we compiled an Excel file detailing all the papers reviewed and shared it as supplementary material. Thus, allowing interested readers to delve into our research process and discover related resources.

Comment #3: Limited Discussion on Challenges: While the paper discusses the methods and techniques, there could be more emphasis on the challenges and limitations faced in the field of fake news detection.

Answer to Comment #3: We thank the reviewer for the good suggestion about adding the fake news detection Challenges. In this regard, we changed the conclusion and future works section to just the conclusion section and then created a new section as Challenges and Future Directions. Our modifications are highlighted in blue in manuscript Section 7.

Researchers face complex difficulties as a result of the growing incidence of fake news in modern media. It is challenging to distinguish fake news because of its complexity and dynamic nature. Therefore, fake news detection methods face challenges and limitations. One of the big challenges in fake news detection methods is content variability. In other words, fake news comes in a wide range of content formats, including text, photos, videos, and voice by its very nature. So this variation necessitates flexible detection methods. Rapid dissemination is another big challenge in detecting fake news. So the need for quick identification is obvious given how quickly fake news circulates, particularly on social media platforms. Additionally, creators of fake news who now use AI technologies are innovative and adaptable, which adds another level of difficulty. The difficulty is further exacerbated by the lack of sufficient training data, the complexity of context sensitivity in identifying fake news, language, and cultural diversity, and the biases built into algorithms. These difficulties not only highlight the complexity of the issue but also act as a foundation for further study. Although there are many studies related to fake news detection, this task still has a long way to go. Therefore, we present some perspectives to pave the way for other researchers in this field.

Then, we updated and added some sentences to the perspectives that we highlighted in blue in the text of manuscript Section 7.

Comment #4: Finally, there are some small improvements that can be made to the English language.

Answer to Comment #4: Thanks for highlighting this point. A full revision of the text was performed, improving language, and refining the text to be more technical and precise.

References

S Selva Birunda and R Kanniga Devi. A novel score-based multi-source fake news detection using gradient boosting algorithm. In *International Conference on Artificial Intelligence and Smart Systems (ICAIS)*, pages 406–414. IEEE, 2021.

Alessandro Bondielli and Francesco Marcelloni. A survey on fake news and rumour detection techniques. *Information Sciences*, 497:38–55, 2019.

Manqing Dong, Lina Yao, Xianzhi Wang, Boualem Benatallah, Quan Z Sheng, and Hao Huang. Dual: A deep unified attention model with latent relation representations for fake news detection. In *International Conference on Web Information Systems Engineering*, pages 199–209. Springer, 2018.

Yi Fung, Christopher Thomas, Revanth Gangi Reddy, Sandeep Polisetty, Heng Ji, Shih-Fu Chang, Kathleen McKeown, Mohit Bansal, and Avirup Sil. Infosurgeon: Cross-media fine-grained information consistency checking for fake news detection. In *Annual Meeting of the Association for Computational Linguistics and the International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1683–1698, 2021.

Mohammad Hadi Goldani, Saeedeh Momtazi, and Reza Safabakhsh. Detecting fake news with capsule neural networks. *Applied Soft Computing*, 101:106991, 2021a.

Mohammad Hadi Goldani, Reza Safabakhsh, and Saeedeh Momtazi. Convolutional neural network with margin loss for fake news detection. *Information Processing & Management*, 58(1):102418, 2021b.

Georgios Gravanis, Athena Vakali, Konstantinos Diamantaras, and Panagiotis Karadais. Behind the cues: A benchmarking study for fake news detection. *Expert Systems with Applications*, 128:201–213, 2019.

Saqib Hakak, Mamoun Alazab, Suleman Khan, Thippa Reddy Gadekallu, Praveen Kumar Reddy Maddikunta, and Wazir Zada Khan. An ensemble machine learning approach through effective feature extraction to classify fake news. *Future Generation Computer Systems*, 117:47–58, 2021.

TAO Jiang, Jian Ping Li, Amin Ul Haq, Abdus Saboor, and Amjad Ali. A novel stacking approach for accurate detection of fake news. *IEEE Access*, 9:22626–22639, 2021.

Rohit Kumar Kaliyar, Anurag Goswami, and Pratik Narang. Fakebert: Fake news detection in social media with a bert-based deep learning approach. *Multimedia tools and applications*, 80(8):11765–11788, 2021.

Peiguang Li, Xian Sun, Hongfeng Yu, Yu Tian, Fanglong Yao, and Guangluan Xu. Entity-oriented multi-modal alignment and fusion network for fake news detection. *IEEE Transactions on Multimedia*, 2021.

Muhammad Imran Nadeem, Syed Agha Hassnain Mohsan, Kanwal Ahmed, Dun Li, Zhiyun Zheng, Muhammad Shafiq, Faten Khalid Karim, and Samih M Mostafa. Hyprobert: A fake news detection model based on deep hypercontext. *Symmetry*, 15(2):296, 2023.

Duc Minh Nguyen, Tien Huu Do, Robert Calderbank, and Nikos Deligiannis. Fake news detection using deep markov random fields. In *Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1391–1400, 2019.

Maria Nefeli Nikiforos, Spiridon Vergis, Andreana Stylidou, Nikolaos Augoustis, Katia Lida Kermanidis, and Manolis Maragoudakis. Fake news detection regarding the hong kong events from tweets. In *IFIP international Conference on Artificial Intelligence Applications and Innovations*, pages 177–186. Springer, 2020.

Adam Paszke et al. Pytorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc., 2019.

Guilherme Penedo, Quentin Malartic, Daniel Hesslow, Ruxandra Cojocaru, Alessandro Cappelli, Hamza Alobeidli, Baptiste Pannier, Ebtesam Almazrouei, and Julien Launay. The RefinedWeb dataset for Falcon LLM: outperforming curated corpora with web data, and web data only. *arXiv preprint arXiv:2306.01116*, 2023.

Shengsheng Qian, Jinguang Wang, Jun Hu, Quan Fang, and Changsheng Xu. Hierarchical multi-modal contextual attention network for fake news detection. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 153–162, 2021.

Chahat Raj and Priyanka Meel. Convnet frameworks for multi-modal fake news detection. *Applied Intelligence*, 51(11):8132–8148, 2021.

RR Rajalaxmi, LV Narasimha Prasad, B Janakiramaiah, CS Pavankumar, N Neelima, and VE Sathishkumar. Optimizing hyperparameters and performance analysis of lstm model in detecting fake news on social media. *Transactions on Asian and Low-Resource Language Information Processing*, 2022.

Dilip Kumar Sharma and Sonal Garg. Ifnd: a benchmark dataset for fake news detection. *Complex & Intelligent Systems*, pages 1–21, 2021.

K. Shu, H. Liu, J. Han, L. Getoor, W. Wang, J. Gehrke, and R. Grossman. *Detecting Fake News on Social Media*. Synthesis Lectures on Data Mining and Knowledge Discovery. Morgan & Claypool Publishers, 2019a. ISBN 9781681735832.

Kai Shu, Amy Sliva, Suhang Wang, Jiliang Tang, and Huan Liu. Fake news detection on social media: A data mining perspective. *ACM SIGKDD Explorations Newsletter*, 19(1):22–36, 2017.

Kai Shu, Deepak Mahudeswaran, and Huan Liu. Fakenewstracker: a tool for fake news collection, detection, and visualization. *Computational and Mathematical Organization Theory*, 25(1):60–71, 2019b.

Amila Silva, Ling Luo, Shanika Karunasekera, and Christopher Leckie. Embracing domain differences in fake news: Cross-domain fake news detection using multi-modal data. In *The AAAI Conference on Artificial Intelligence*, volume 35, pages 557–565, 2021.

Marcellus Smith, Alexicia Richardson, Brandon Brown, Gerry Dozier, Michael King, and Joshua Morris. A study of the impact of evolutionary-based feature selection for fake news detection. In *Symposium Series on Computational Intelligence (SSCI)*, pages 1859–1865. IEEE, 2020.

Chenguang Song, Nianwen Ning, Yunlei Zhang, and Bin Wu. Knowledge augmented transformer for adversarial multidomain multiclassification multimodal fake news detection. *Neurocomputing*, 462: 88–100, 2021.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023.

Nguyen Manh Duc Tuan and Pham Quang Nhat Minh. Multimodal fusion with bert and attention mechanism for fake news detection. In *International Conference on Computing and Communication Technologies (RIVF)*, pages 1–6. IEEE, 2021.

Yaqing Wang, Fenglong Ma, Haoyu Wang, Kishlay Jha, and Jing Gao. Multimodal emergent fake news detection via meta neural process networks. In *ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, pages 3708–3716, 2021.

Thomas Wolf et al. Transformers: State-of-the-art natural language processing. In *Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Yang Wu, Pengwei Zhan, Yunjian Zhang, Liming Wang, and Zhen Xu. Multimodal fusion with co-attention networks for fake news detection. In *Findings of the Association for Computational Linguistics*, pages 2560–2569.

Jiawei Zhang, Bowen Dong, and S Yu Philip. Deep diffusive neural network based fake news detection from heterogeneous social networks. In *2019 IEEE International Conference on Big Data (Big Data)*, pages 1259–1266. IEEE, 2019.

Xichen Zhang and Ali A Ghorbani. An overview of online fake news: Characterization, detection, and discussion. *Information Processing & Management*, 57(2):102025, 2020.

Xinyi Zhou and Reza Zafarani. A survey of fake news: Fundamental theories, detection methods, and opportunities. *ACM Computing Surveys (CSUR)*, 53(5):1–40, 2020.

Xinyi Zhou, Atishay Jain, Vir V Phoha, and Reza Zafarani. Fake news early detection: An interdisciplinary study. *arXiv preprint arXiv:1904.11679*, 2019.

Highlights

- Provides an up-to-date taxonomy for textual-based fake news detection perspective.
- Presents an empirical comparison between several feature extraction techniques and classification algorithms.
- Provides an error analysis to analyze the potential of combining classifiers and/or feature extractors.
- Compares different methods in the way of cost-effectiveness.
- Proposes multiple perspectives for future research.

Fake News Detection: Taxonomy and Comparative Study

Faramarz Farhangian[a], Rafael M. O. Cruz[a], George D. C. Cavalcanti[b]

[a] *École de Technologie Supérieure, Université de Quebéc, , Montréal, , Québec, Canada*
[b] *Centro de Informática, Universidade Federal de Pernambuco , Recife, , Pernambuco, Brazil*

Abstract

The proliferation of social networks has presented a significant challenge in combating the pervasive issue of fake news within modern societies. Due to the large amount of information and news produced daily in text, audio, and video, the validation and verification of this information have become crucial tasks. Leveraging advancements in artificial intelligence, distinguishing between fake news and factual information through automatic fake news detection systems has become more feasible. Automatic fake news detection has been explored from diverse perspectives, employing various feature extraction and classification models. Nonetheless, empirical evaluations, categorization, and comparisons of existing techniques for handling this problem remain limited. In this paper, we revisit the definitions and perspectives of fake news and propose an updated taxonomy for the field based on multiple criteria: 1) Type of features used in fake news detection; 2) Fake news detection perspectives; 3) Feature representation methods; and 4) Classification approaches. Moreover, we conduct an extensive empirical study to evaluate several feature representation techniques and classification approaches based on accuracy and computational cost. Our experimental results demonstrate that the optimal feature extraction techniques vary depending on the characteristics of the dataset. Notably, context-dependent models based on transformer models consistently exhibit superior performance. Additionally, employing transformer models as feature extraction methods, rather than solely fine-tuning the network for the downstream task, improves overall performance. Through extensive error analysis, we identify that a combination of feature representation methods and classification algorithms, including classical ones, offer complementary aspects and should be considered for achieving better generalization performance while maintaining a relatively low computational cost. For further

details, including source codes, figures, and datasets, please refer to our project's GitHub repository: [https://github.com/FFarhangian/Fake-news-detection-Comparative-Study].

Keywords: Disinformation; Misinformation; Machine Learning; Deep Learning; Natural Language Processing; Fake news detection

1. Introduction

It is incontrovertible that we live in an age where social networks play a significant role. Thanks to social networks, many people in modern societies have rapid access to information and news. This easy and free access to social media has led many people to use this media as a source of news [1]. According to the survey of adults in the United States on social media news consumption in 2021[1], approximately half of the adults in the United States receive news on social media, especially on Twitter and Facebook. Social media prepare a space for individuals to share their opinions or send comments about news or events quickly and without physical contact.

On the one hand, social networks have led many news agencies to focus more on emerging social networks such as Twitter and Facebook than traditionally publishing news, such as newspapers and magazines [2, 3, 4, 5, 6, 7]. On the other hand, these media have paved the way for profiteers who seek to spread fake news with political and economic motives [1, 5, 6, 7, 8]. Fake news is most of the time related to influential social, political, and economic events such as the US presidential election [9, 10], the COVID-19 pandemic [11], Brexit [12], Syrian civil war [13], and Russia-Ukraine war [14].

An efficient mechanism to detect fake news and prevent the rapid spread of fake news in society has become one of the most critical challenges in advanced societies [1, 7, 8] because the dissemination of fake news can have devastating effects on society from social, political, and economic aspects. Statistics show that the human ability to distinguish fake news from

Email addresses: faramarz.farhangian.1@ens.etsmtl.ca (Faramarz Farhangian), rafael.menelau-cruz@etsmtl.ca (Rafael M. O. Cruz), gdcc@cin.ufpe.br (George D. C. Cavalcanti)

[1] www.pewresearch.org/journalism/2021/09/20/news-consumption-across-social-media-in-2021/

actual news is almost like tossing coins [8, 15]. Therefore, setting up an accurate and reliable automated system to detect fake news is very important these days [8, 1].

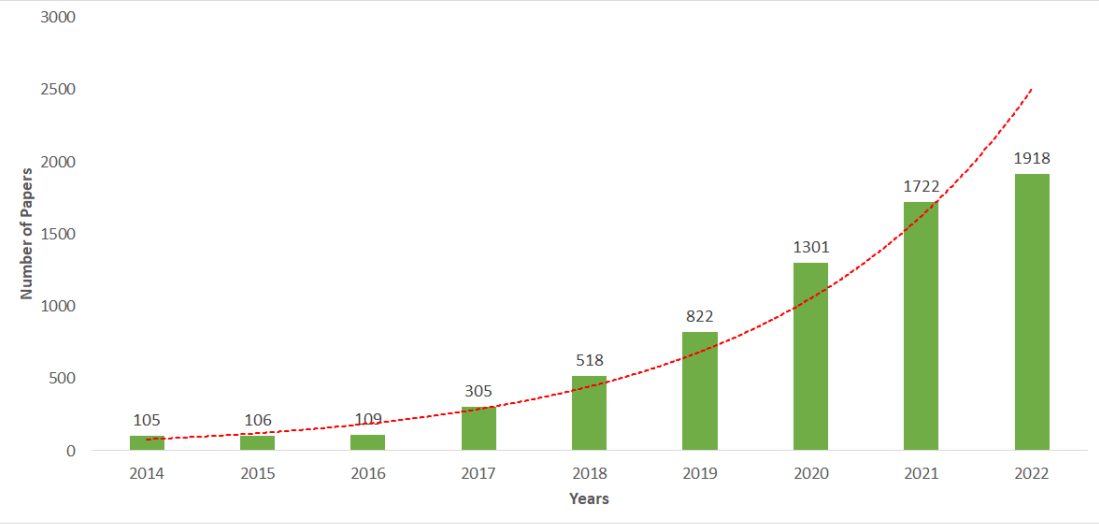


Figure 1: Frequency of published paper in fake news detection from 2014 to 2022.

Automatic Fake news detection is one of the emerging research areas in machine learning and artificial intelligence. The increased number of scientific articles published in the last years in this field compared to previous years is quite evident. In fact, after the 2016 US presidential election and because of its impact on the US election, many scholars turned their attention to fake news detection methods. Figure 1 shows the increase in Fake news detection publications across the years.

Due to the fact that one of the most attractive fake news detection approaches is linguistic-based fake news detection, in which fake news is detected based on news content features like textual body news articles or social media posts, most of the existing comparative studies focus on this perspective. In this regard, several survey studies have focused on fake news definition, data collection challenges, features, and techniques [1, 6, 8, 16]. However, these studies have been mostly limited to exploring the characteristics and available approaches and have not performed any experiments to evaluate and compare available methods.

To address this problem, in recent years, a few studies have focused on a comparative study to categorize and evaluate the available models in fake news detection [17, 18, 19, 20]. To be more specific, Gilda [17] and Gravanis et al. [19] conducted a comparative study on several datasets. However, they did not evaluate different deep learning models and state-of-the-art feature extraction techniques. Oshikawa et al. [18] evaluated several deep learning models and classical machine learning models over three datasets, but they did not analyze transformer models as feature extraction or end-to-end model. Khan et al. [20] evaluated several machine learning models and deep learning models on three datasets. However, they evaluated a few feature extraction methods without any statistical comparison analysis.

Despite significant progress in this field, a contemporary taxonomy of feature representation and classification models has yet to be presented. Moreover, no comprehensive comparative study has been conducted considering various feature extraction techniques, classification algorithms, and benchmark datasets in a consistent experimental framework. Hence, the discriminating power of different feature extraction and classification algorithms and how they differ in terms of prediction behavior and computational cost is still an open question that needs to be investigated. Exploring such points is crucial to have a comprehensive view of the current state-of-the-art in the field and pave the way for promising research directions. It can also indicate the complementary power of different approaches that can be leveraged for learning more robust systems. Hence, this work seeks to answer the following questions to address all of these issues:

RQ1: *What is the impact of feature extraction methods on model performance?*

RQ2: *Does the use of transformer models as representation learning have higher performance or as fine-tuning*[2]*?*

RQ3: *Would combining feature extraction methods improve performance in the fake news detection task?*

[2]Fine-tuning refers to training the weights of the pre-trained model on fresh data or domains, whereas representation learning employs the weights of the pre-trained model as a feature extractor or encoder for various tasks.

RQ4: *Which methods are more cost-effective?*

We first conduct a systematic literature review to identify existing feature extraction methods and classification models used in fake news detection. Then, we propose an up-to-date taxonomy of automatic fake news detection approaches based on three main aspects: 1) Pre-processing techniques for cleaning news articles; 2) news article features that consider whether content-based features or social context-based features are used to detect fake news articles; 3) The state-of-the-art methods used for fake news detection. Then we conduct a comparative study to compare the performance of several state-of-the-art classification models under the same experiment set up over four benchmark datasets used for fake news detection. This work considers a large set of feature representation techniques, including 15 from different perspectives (count-based, word embeddings, and transformer models) and 20 classification algorithms (transformers models employed in both feature extraction methods and end-to-end classifiers)This work considers a large set of feature representation techniques, including 15 from different perspectives (count-based, word embeddings, and transformer models) and 20 classification algorithms (transformers models employed in both feature extraction methods and end-to-end classifiers). We also compare the results of transformer models used to extract feature representations from a news article compared to its usage as an end-to-end model for classification. Experimental results show that contrary to their most common usage as an end-to-end classifier, improved performance is obtained when transformer models are used as feature extraction techniques to feed another classification algorithm.

Therefore, the contributions of this paper compared to other reviews on fake news detection are the following:

- It proposes an up-to-date taxonomy for textual-based fake news detection models and feature extraction methods.
- It presents an empirical comparison between several state-of-the-art feature extraction techniques and several classification algorithms, including transformer models used in

an end-to-end model (i.e., fine-tuning) vs. as feature extractors over multiple datasets under the same experimental environment.

- It provides an error analysis regarding the possibility of combining different classifiers and/or feature extractors to increase the models' effectiveness in predicting fake news.
- It compares different methods in the way of cost-effectiveness.
- It proposes multiple perspectives for future research based on the analysis conducted in this work.

The rest of the paper is organized as follows: Section 2 presents an overview of fake news characteristics and fake news detection perspectives, including definitions, available feature representation techniques, and different detection methods in fake news classification. Section 3 describes the fake news classification process, including text pre-processing methods, text representation methods, and state-of-the-art classification models, and proposes an updated taxonomy to the field. Section 4 presents the datasets and experimental protocol used in this study. Section 5 shows the results and analyzes them to answer the aforementioned research questions. Finally, Section 6 presents the conclusions and perspectives for future research on fake news detection.

2. Fake news characteristics and detection perspectives

Before dealing with different fake news detection perspectives, we need to know the definition of fake news and its main characteristics so that we can properly define different fake news detection strategies. Figure 2 demonstrates the example of fake news and factual news based on data that come from the Liar data set [21].

2.1. ***Fake news definition***

In the first place, knowing the exact definition of fake news is very important. Considering a single and accurate definition of fake news is essential in order to prevent confusing fake news with other related concepts that are sometimes used instead of fake news in the

Figure 2: Examples of fake news and factual news extracted from the Liar dataset [21]. The ID tag determines unique ids which come from the dataset.

research process. According to [1, 22] there are two accepted definitions of *"Fake News"* as follows:

Definition 2.1 (Broad definition). *"Fake news is false news"*

Definition 2.2 (Narrow definition). *"Fake news is a news article that is intentionally and verifiably false"*

Unlike the first definition, which only focuses on the validity of the news, the second definition focuses on both validity and intentions. More specifically, according to the second definition, fake news contains misleading information and is meant to deceive users. Many concepts related to fake news can overlap with it [8]. So the *narrow definition* has been adopted for the true definition of fake news in this research since this definition can prevent confusing fake news with other related concepts that are sometimes used instead of fake news in the research process. So, according to this definition, the *Satire*, *Rumors*, *Hoaxes*, *Misinformation*, and *Disinformation* concepts are distinct from fake news. Satire news is not intended to confuse or mislead users, and Rumors lack an unambiguous intent, may lack factual proof or verification, and are not always considered as news [8]. Hoaxes that are just intended for entertainment or to defraud certain persons. Among all these

concepts, misinformation, and disinformation are more close to fake news concepts. Misinformation [3] in contrast with disinformation and fake news may not spread false information with malicious intent [23]. Disinformation [4] and fake news on the other hand intentionally circulate false information, however fake news is manufactured and presented as news, whereas disinformation is not necessary to be news [24].

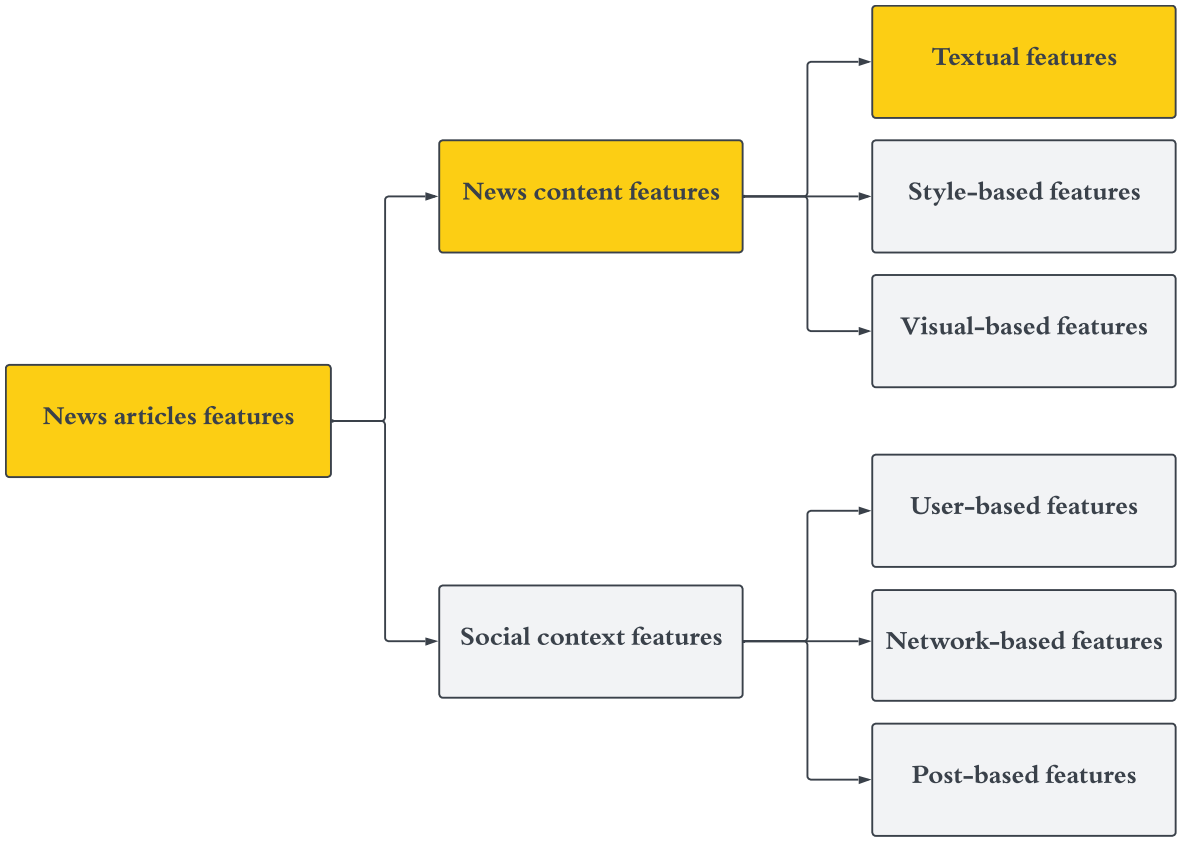


Figure 3: Type of features in fake news detection problem. The highlighted in orange shows the focus of this research.

*2.2. **Available features and fake news detection approaches***

Fake news can be detected with the help of features extracted from the news articles, and these features can be categorized into two main approaches: *content features and social context features* [1]. Figure 3 shows the summary of the available features in fake news

[3] https://www.dictionary.com/browse/misinformation
[4] https://www.dictionary.com/browse/disinformation

detection applications. The goal of the content-based approach is to find signs or patterns through the news content features that are extracted from the body of the article like text, image, or video [8, 6]. These types of features are divided into Textual features, Style-based features, and Visual-based features [1]. So, we can study content-based fake news detection approaches from four main perspectives based on the type of features mentioned above. Textual-based, Style-based, Visual-based, and Knowledge-based. Figure 4 summarizes these approaches.

It can often be difficult to distinguish fake news based just on its content; but, any success in doing so may encourage fake news authors to adopt further precautions in the future in order to fool the detection system. So, social context features might be integrated into news analysis to improve the detection algorithm's resistance to such an attack [25]. The methods that use social context features in order to distinguish between fake news and factual news are social context-based approaches. Social context features imply the features that are extracted from marginal information such as user information, network information and/or propagation, and the reaction of other users [6]. Social context-based approaches are divided into three main categories: *user-based, post-based, and network-based* [1].

There are also a few studies that used the hybrid perspective for the detection of fake news. From a hybrid perspective, different characteristics of fake news have been combined to increase the power and accuracy of fake news detection models. To be more specific, they used multi-modal machine learning in order to develop models capable of handling and relating data from several modalities. In this regard, there are some studies that processed news content features and social context features at the same time [26, 27, 28, 29, 30, 31, 32, 33, 34, 35, 36, 37, 38, 39, 40, 41, 42, 43]. As this article specifically focuses on textual-based features, we will only elaborate on these features.

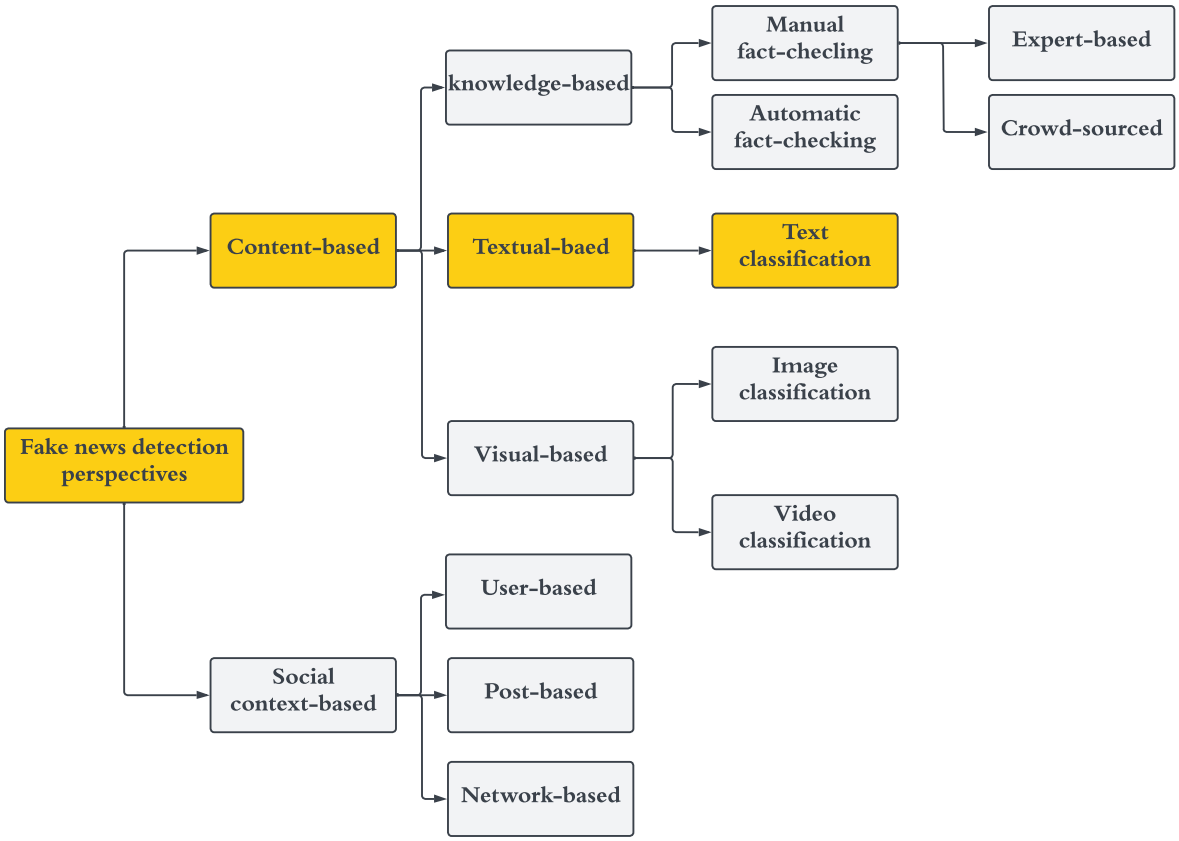


Figure 4: Fake news detection perspectives. The highlighted in orange shows the focus of this research.

2.2.1. ***Textual-based fake news detection***

In this method, we are looking for patterns through the textual features which are extracted from the news stories. Textual features refer to the features which are taken from the headline or body text of news articles [1]. There are three kinds of textual features, *Linguistic features, Low-rank textual features, and Neural textual features.* Linguistic features analyze text at multiple levels such as words, sentences, and phrases, and provide different characteristics of fake and factual news [1]. Examples of linguistic features include lexical features, syntactic features, and semantic features (e.g. TF, TF-IDF) [8, 1]. Low-rank models use tensor or matrix factorization to extract a small-scale text representation from a large-scale, noisy input feature matrix [1]. Neural textual features rely on dense vector representations such as embedding methods (Word2Vec, GloVe, fastText, etc), rather than high-dimensional and sparse features [8]. Therefore, we can consider textual-based fake news

Feature types	Feature sets	Techniques	References
Count-based	Frequency-based	Bag-of-words (BOW)	[44, 45, 46, 47, 48, 49, 50, 51, 52]
		TF-IDF	[44, 17, 53, 54, 45, 55, 46, 56, 47, 57, 58, 7, 50, 51, 52]
Prediction-based	Context-independent	Word2Vec	[21, 59, 60, 61, 19, 62, 63, 49, 50, 64, 65]
		FastText	[61, 51, 66, 65]
		GloVe	[67, 61, 68, 69, 49, 51, 66, 70, 65]
	Context-dependent	Bidirectional (i.e., BERT, ALBERT, ELECTRA)	[67]
		Unidirectional (i.e., ELMO)	[71]

Table 1: Feature extraction methods used in previous works

detection as a classification problem.

So despite most of the studies using supervised learning models to detect fake news, there are some studies that used unsupervised learning [72, 73, 74, 75], semi-supervised learning [69, 76, 77, 78, 79, 80, 81] and reinforcement learning [82]. In this study, we are focusing on the studies that used Linguistic features and Neural textual features for fake news detection. Not only because they have shown the ability to accurately discriminate between real and fake news, but they are also commonly used because of textual information since it is more abundant and present in all news. Moreover, there is a lack of datasets with other kinds of features.

There are myriad studies in textual-based fake news detection. We can classify the existing works based on definitions, related terms, language, fake news detection approaches, and feature extraction methods. In this regard, English is the most studied language in the context of fake news detection. Still, recent advances have also been made in many other languages like Slovak [83, 84], Urdu [49, 85, 86, 87, 88, 89, 90], Portuguese [91, 92, 93, 94], Korean [95, 96], Indian [97], Germany [98, 99], Spanish [100, 101], Bengali [102, 103], Chinese [104], Indonesian [105]. There are also some studies that connect fake news to terms like rumor [106, 77], satire news [107], hoax [108], Click-bait [109].

According to the fake news detection task, most of the works used supervised learning approaches. In this regard, on the one hand, there are some works that use monolithic classifiers in order to classify the news. On the other hand, there are some works that used ensemble classifiers for news classification tasks. Most recent papers use more word embed-

Classification methods	Models	References
Classical machine learning	K-nearest neighbors (KNN)	[44, 53, 56, 19, 7, 50, 110, 51]
	Support vector machine (SVM)	[44, 21, 17, 53, 54, 60, 55, 46, 56, 47, 19, 58, 7, 50, 110, 51, 71, 52]
	Naive Bayes (NB)	[21, 54, 45, 46, 56, 47, 19, 58, 7, 50, 51, 71]
	Logistic regression (LR)	[44, 21, 53, 60, 46, 47, 58, 110, 51, 52]
Ensemble models	Random forest (RF)	[17, 46, 56, 57, 48, 58, 7, 110, 51, 71, 52]
	Adaptive Boosting (AdaBoost)	[56, 19, 51]
	Extreme Gradient Boosting (XGBoost)	[56, 57, 58, 7, 51, 71, 52]
Deep learning models	Convolutional neural network (CNNs)	[21, 59, 60, 67, 61, 62, 63, 68, 69, 50, 70, 65]
	Long Short Term Memory (LSTM)	[21, 59, 60, 123, 67, 61, 62, 63, 68, 49, 50, 64, 66, 65]
	Multi-layer Perceptron (MLP)	[55, 56, 47, 49, 52]
	Transformers (End-To-End)	[71, 20]

Table 2: Classification methods used in previous works

ding methods and pre-trained methods than Bag-of-Words, N-gram, and Count Vectorizer because word embedding methods can give some information about the relation between words, and represent consequently, we can see an improvement in the performance. Table 1 shows the feature extraction methods used in previous works.

About the classifier models, different paradigms have been implemented; tree-based algorithms such as decision trees (DT) and random forest (RF) [7, 17, 19, 44, 46, 47, 48, 51, 52, 53, 55, 56, 57, 58, 71, 110], artificial neural networks such as multi-layer perceptron (MLP) [55, 56] and convolution neural networks (CNN) [21, 50, 59, 60, 61, 62, 65, 67, 68, 70], Bayesian as the naive Bayes (NB) [7, 46, 56, 47, 19, 58, 51, 71, 50, 54, 45], and Support Vector Machines (SVM) [7, 21, 44, 17, 53, 46, 47, 19, 58, 110, 51, 71, 52, 60, 50, 54]. The ensemble approaches can contain deep ensemble models and machine learning ensemble models. Most of the works used linguistics features to train the models; however, some works used visual features [111, 112, 113] or social features [114, 115, 116]. There are some studies that used multi-view learning approach [79, 112, 117, 118, 119, 41, 120, 121, 38, 122]. Table 2 shows classification methods used in previous works in fake news detection.

3. Textual-based fake news detection process

In this paper, we restrict our focus to text-based fake news detection, primarily because it is the most popular method employed by researchers [1], but also because benchmark

datasets for fake news detection typically include news content features, with a particular emphasis on text [6]. So considering having labeled textual datasets in automatically detecting fake news, this task can be formalized as a text classification task (Equation 1). Suppose that x refers to a news article, so we need a function to identify whether it is fake or real news. Therefore we can consider H to refer to the prediction function. So the fake news detection task can be identified as an indicator function:

$$H(x) = \begin{cases} 1 & \text{if } x \text{ is fake news} \\ 0 & \text{if } x \text{ is real news} \end{cases} \tag{1}$$

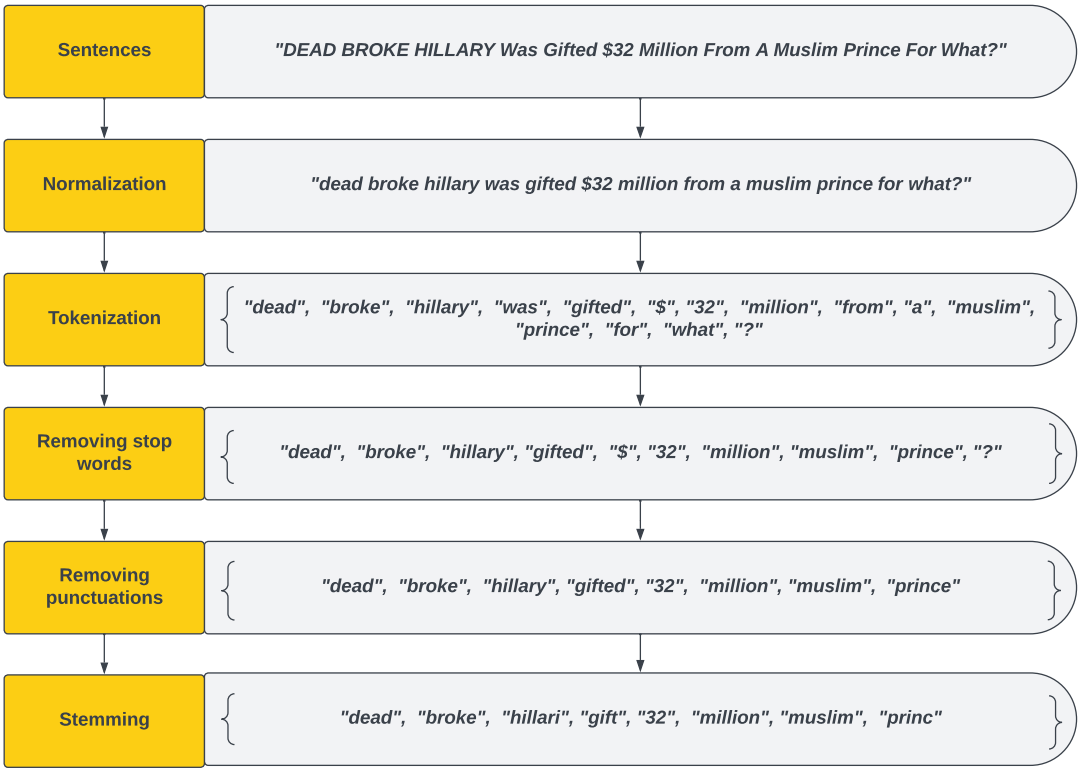


Figure 5: Overview of pre-processing phase. The example has been extracted from the Liar dataset [21].

3.1. ***Text pre-processing***

Text pre-processing is the initial step in preparing raw text for Natural Language Processing (NLP) which helps machines to understand human language. The purpose of data pre-processing is to produce "clean text" that computers can analyze without errors. Incorrect data can have long-term negative effects, which is why pre-processing tasks must be implemented before feature extraction and training steps. Text pre-processing consists of five primary steps: normalization, tokenization, stop words removal, punctuation removal, and stemming (as shown in Figure 5). Examples of these terms include:

- **Normalization:** The Process of converting terms in a text in a standard form called normalization. For example, "eaaaaaasy" converted to "easy" or ":(" converted to "sadness"
- **Tokenization:** It means the process of splitting textual data into meaningful units like words, sentences, and documents [124]. An N-gram is one of the famous methods of tokenization. N-gram means continuing the order of terms in a textual document [46]. N-gram is a probabilistic model that has been made based on the Markov chain theory.
- **Removing Stop Words:** Process of removing a useless word or terms in text classification process called removing stop words [124]. In the English language, there are some stop words such as "The", That, "a", "in" etc.
- **Removing Punctuation:** The process of removing some signs like "!", "()", "[]", "", "?", "@" and etc from textual data called as removing punctuation task [125].
- **Stemming:** Process of converting a word in its root form called stemming [124]. For example, Connected, Connection, Connects, and Connections convert to Connect.

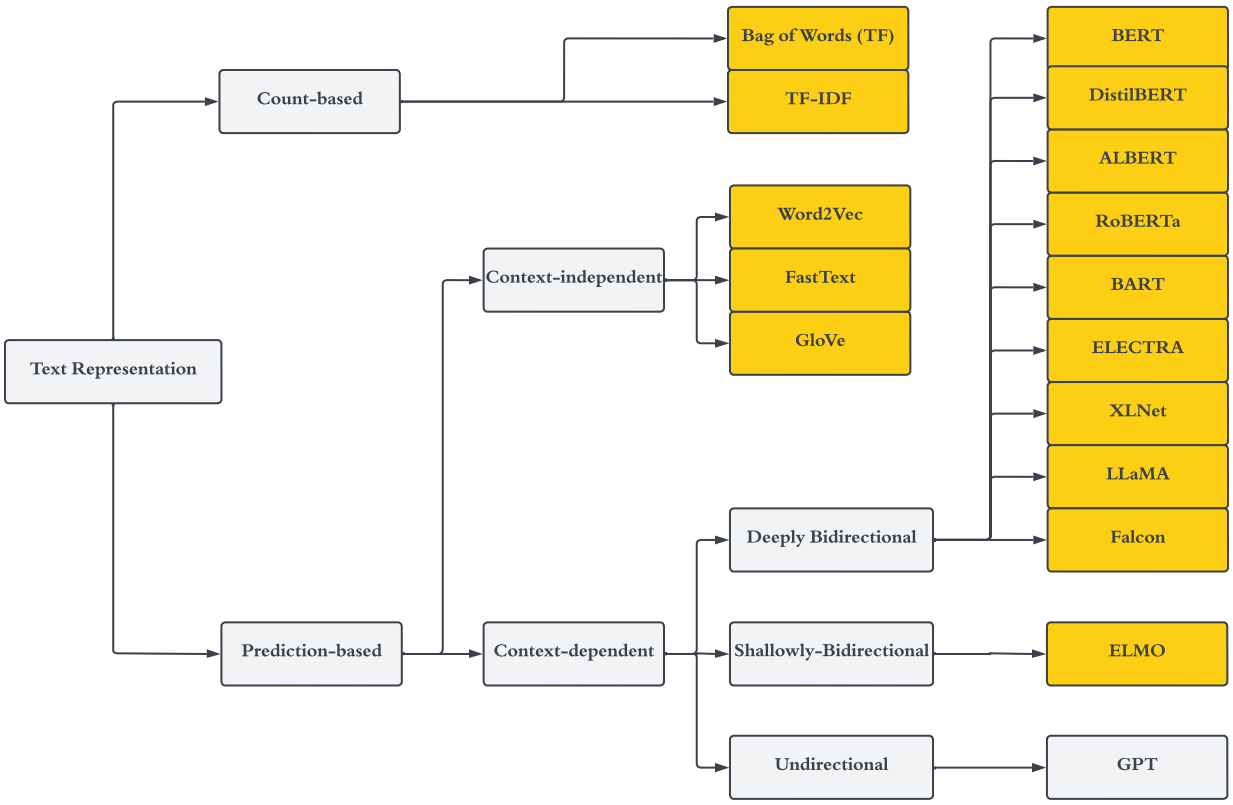


Figure 6: Taxonomy of text representation methods. The methods are highlighted in orange since it is the focus of this paper.

3.2. ***Text Representation***

Text representation, also known as feature extraction, is a key step in text classification. This process converts text data into numerical vectors, which can be processed by a machine. Different methods exist, from simple count-based approaches such as Bag-of-Words and TF-IDF to more complex prediction-based methods such as Word2Vec, fastText, ELMO, and BERT, and their choice can significantly impact the performance of fake news detection models. Figure 6 provides a taxonomy of text representation methods used in fake news detection, while Table 1 summarizes the most popular methods used in previous works. We review these methods, their advantages, and their drawbacks, to provide an overview of the various text representation methods which can be used in fake news detection. This will help researchers and practitioners select the most suitable method for their particular use case.

3.2.1. ***Count-based Text Representation***

One of the primary text representation methods is count-based representation, which only focuses on the number of times words occur or co-occurs in a document. To be more specific, count-based methods create a matrix of occurrence and co-occurrence, where each element indicates the frequency of a word within a certain document. Although this type of representation is simple and popular, it has a major drawback: it does not capture any meaningful information about the meaning and relationships between words. The most commonly used frequency-based methods are as follows:

Bag-of-Words (BOW) [126] is one of the most elementary text representation methods that have been used in a large number of fake news detection studies [46, 45, 47, 48, 49, 50, 51, 52]. Some technical paper knows it as a count vectorizer. This representation method just describes the occurrence and frequency of words (Term Frequency (TF)) in a document. To be more specific, it does not give anything about the position and structure of words in a document. Although this method is very simple, it does not give information about the relationships between words in documents. Also, in large textual datasets representation matrix gets too sparse and requires a high computational effort [127]. Equation 2 formulate it in a mathematical way. Suppose W is an occurrence matrix, d refers to a document, t refers to a term in a document, and V the vocabulary in a way that $t \in V$.

$$BOW : \{V \longrightarrow W | W \in R^{T \times D}, W_{ij} = TF(d,t)\} \tag{2}$$

Term Frequency-Inverse Document Frequency (TF-IDF) [128] is a statistical feature representation method in natural language processing to score and weight words in order to find the relevant and important words in a document [124]. This method has been used in many text classification problems, especially for fake news detection [44, 17, 53, 54, 45, 55, 46, 56, 47, 57, 58, 119, 52]. To put it in more formal mathematical terms, Equations 3, 4, 5, 6 formalize it as below:

$$TF(d,t) = Count\ (term\ t\ occurrence\ in\ document\ d) \tag{3}$$

$$IDF(t) = \log\left(\frac{Count(document)}{Count(number\ of\ document\ included\ term\ t)}\right) \tag{4}$$

$$TF - IDF(d,t) = TF(d,t) \times IDF(t) \tag{5}$$

$$TF - IDF : \{V \longrightarrow W | W \in R^{T \times D}, W_{ij} = TF - IDF(d,t)\} \tag{6}$$

Unlike Bag-of-Words which only provides a collection of vectors based on the count of word occurrences in the document, the TF-IDF model incorporates more than just the count. It adds information on the importance of certain terms. This gives TF-IDF greater discriminative power, as rare words which are indicative of fake news are given more weight while common words which are not indicative are given less. As a result, TF-IDF can improve the accuracy of fake news classification by identifying important features that might be missed by using TF alone.

3.2.2. ***Prediction-based Text Representation***

Deep learning methods have made significant progress in recent years, particularly in language modeling. One of the advances is the use of neural network models that create word embeddings from textual data by defining tasks such as next-word prediction. These methods extract semantic and syntactic features of words, addressing the limitations of Count-based methods. The first neural network language model using a feed-forward neural network was introduced by Bengio et al. [129]. As shown in Figure 6, Neural network text representation is divided into two main categories: context-independent and context-dependent methods. In context-independent methods, a word always has the same representation, regardless of its context. Word2Vec [130], Glove [131], and fastText [132] are the most popular context-independent text representation methods. Context-dependent methods represent words by considering the context, divided into three main classes: unidirectional, shallowly bidirectional, and deeply bidirectional text representation. Unidirectional representation contextualizes each word based on the words in the left or right directions, while bidirectional representation contextualizes each word based on both right and left context [133].

GPT [134] is the most popular unidirectional text representation, while BERT [133], DistilBERT [135], RoBERTa [136], BART [137], ELECTRA [138], XLNet [139], and GPT-2 [134] are the most popular bidirectional text representation methods.

3.2.2.1 Context-independent methods

Context-independent models or Word embeddings are a popular technique for representing textual data in a numerical format suitable for machine learning algorithms. Three popular word embedding methods are Word2Vec, GloVe, and fastText. This section provides an overview of these methods and highlights their main features and applications.

Word2Vec was the first context-free word embedding method proposed by Mikolov et al. [130] in 2013. Frequency-based methods, such as Count-based methods, suffer from two main problems: they are inefficient when the data size increases and do not account for word similarities [140]. To overcome these limitations, context-free methods like Word2Vec use a neural network model with a single hidden layer to learn the representation of words in textual data. Specifically, Word2Vec represents each word as a fixed-length numerical vector so that similar words have similar embedding vectors [140]. Two main prediction model architectures, Continues Bag of Word (CBOW) and Skip-gram, are used in Word2Vec to produce word representations from a large corpus of text. CBOW models consider both n words before and n words after a target word w_t to predict the target word, whereas Skip-gram models consider a target word like w_t to predict the surrounding n words of the target word [130]. Word2Vec has been used in many text classification problems, particularly for fake news detection [21, 59, 60, 61, 19, 62, 63, 49, 50, 64, 51, 65]. However, Word2Vec cannot handle out-of-vocabulary words due to the small local context. Nevertheless, the small context area makes this method less expensive regarding memory than the GloVe model.

GloVE, which stands for global vectors for word embedding, was published in 2014 by Pennington et al. [131]. This model directly extracts global statistical information from whole words, as the term "Global" suggests [131]. GloVe uses an unsupervised algorithm to

learn and represent the distribution of words in the form of numeric vectors. Specifically, this method uses matrix factorization to map words into space, where the distance between words in this space indicates the similarity between the words. Matrix factorization creates a word-word co-occurrence matrix to extract the relationship between words. GloVe has been used in many text classification problems, especially for fake news detection [67, 61, 68, 69, 49, 66, 70]. However, GloVe has a higher memory cost than other methods like Word2Vec or fastText.

FastText is one of the powerful word embedding methods that was published in 2017 by Bojanowski et al. [132]. Like Word2Vec, fastText is a prediction-based text representation that uses skip-gram with minor modifications in its architecture for representation learning. To be more specific, the fastText method skip-gram model does not consider the structure of each word. Instead, each word is represented as a set of characters by using N-gram tokenization [132]. So embedding vector of each word is calculated by the sum of the character N-gram vectors. This method has been used in many text classification problems, including for fake news detection [61, 51, 66, 65]. Due to the character N-gram level representation, fastText representation outperforms other word embedding techniques in morphologically rich languages like Arabic or German [132]. Also, compared to Word2Vec, it can better handle out-of-vocabulary words. However, it has a higher memory cost, and the optimal number of N-grams should be tuned. The author mentioned that the best number of N-grams is between three and six. However, it depends on the target task and language [132]. Another fastText advantage is that it can also be used to generate subword embeddings, which is useful for handling rare or unseen words.

3.2.2.2 Context-dependent methods

ELMo is an abbreviation of the word *Embedding from Language Models*, and it is a unidirectional contextualized word representation that was published in 2018 by Peters et al. [141]. As we mentioned in Section 3.2.2.1, one of the main problems of Context-independent text representations like Word2Vec, GloVe, or fastText is that they can not handle Polysemous

words. To be more specific, these models can not work efficiently when they face words whose meaning changes based on the context. To address this problem, Context-dependent models like ELMO were created. This model is trained based on a two-layer bidirectional LSTM model, which takes tokens at the character level and export embedding vectors at the word level. This method has been used in some text classification problems, including for fake news detection [71].

BERT is an abbreviation of *Bidirectional Encoder Representations from Transformers*, and it is a deep contextualized pre-trained language representation published by Devlin et al. [133] that has been able to achieve significant results in many natural language processing tasks. The BERT pre-train process consists of two main models, masked language modeling (MLM) and next sentence prediction (NSP). The masked language model predicts the percentage of random masked input tokens [140]. As a result, the model can capture the syntactic and semantic meaning of mask words. In the next sentence prediction, the goal is to train the relations between pairs of sentences by a binary classification task [140]. BERT takes word piece tokenization as input of the models, so it helps the models to handle out-of-vocabulary words or rare words in a dataset. BERT pre-trained based on a 12-24 layer transformer on huge unstructured data from Wikipedia and books. A shortage of labeled datasets, especially task-specific ones, is the biggest problem in natural language processing tasks. The most advantage of deep contextualized pre-trained models like BERT is that they are unsupervised bidirectional systems. This means that they are pre-trained based on a large number of unannotated data, and we can fine-tune them for any task in NLP. Despite the fact that transformers have solved many of the shortcomings of previous text representation methods and have had significant results on a variety of NLP tasks, they have a slow learning process and are not cost-effective due to a large number of trainable parameters. More specifically, there are over 110M trainable parameters in the BERT-base model.

RoBERTa is an abbreviation of *A Robustly Optimized BERT Pre-training Approach* that was published in 2019 by Liu et al. [136]. RoBERTa model shows the important role of hyper-parameters in the performance of pre-trained models. RoBERTa compared to BERT,

trained with a larger dataset (RoBERTa-base trained by 160GB text corpus and BERT-base trained by 16GB text corpus). This model is pre-trained based on five English corpora from various domains like book corpus, CC-news, OpenWebText, and others [136]. RoBERTa was developed based on BERT architecture with some minor modifications on key hyper-parameters like learning rates and the number of batches. Also, the next sentence prediction has been removed in this model. According to GLUE, benchmark [142], RoBERTa is one of the best pre-trained models in the leaderboard (it is currently in the 21^{st} place).

DistilBERT [135] is a distilled version of BERT that was published in 2019 by Sanh et al. To be more specific, DistilBERT is a smaller, faster, cheaper, and lighter version of BERT that is able to reduce BERT size up to 40% by preserving 97% of BERT model capability. These features can make DistilBERT 60% faster than BERT. They compacted the BERT model by using Knowledge distillation techniques [143]. According to [143], it is a compact version of the original model that can reproduce the behavior of the original model. In the literature, the large model is considered as the teacher, and the compact model is considered as the student. Also, according to the GLUE benchmark leaderboard, the BERT model (47^{th} place) ranks higher in the leaderboard table compared to DistilBERT (74^{th} place).

ALBERT is an abbreviation of *A Lite BERT* that was published in 2019 by lan et al. [144]. ALBERT is a light version of BERT self-supervised learning of text representation. Increasing the size of transformer models leads them to better performance in all downstream NLP tasks. However, this increase in the size of models slows down the training process and faces us with memory costs [144]. In order to solve this problem, they used BERT architecture with some key differences. First, word embedding parameters are factorized by splitting them into two smaller matrices [144]. Second, by cross-layer parameter sharing, layer parameters are shared for each comparable subsegment, so as a result of this modification is a considerable decrease in the number of parameters. So, exchanging parameters can do more than just lower the computational cost of training; it can also make training more effective [144]. Third, sentence-order prediction, often known as SOP, is used instead of the next-sentence prediction (NSP) loss to measure inter-sentence coherence [144].

Even though ALBERT has 70% fewer parameters than the BERT model, it often obtains higher performance. Furthermore, ALBERT models outperform BERT models in terms of data capacity and can be trained 1.7 times quicker [144].

BART [137] is an abbreviation of *Bidirectional Auto-Regressive Transformers* published in 2019 by Lewis et al. BART is a combination of BERT and GPT due to the fact that it is bidirectional like BERT and Auto-regressive like GPT pre-trained model. BART trained based on sequence-to-sequence model bidirectional encoder and auto-regressive decoder [137]. During the training process, firstly, the text is corrupted using a random noise generator, and then a model for reassembling the original text is learned. It employs a typical neural machine translation architecture based on Transformer [137].

ELECTRA [138] is one of the state-of-the-art text representation learning models which is outperforming the other models like RoBERTa or XLNet according to the GLUE benchmark leader board [142]. ELECTRA is an abbreviation of *Efficiently Learning an Encoder that Classifies Token Replacements Accurately*. In attention to BERT, ELECTRA by using replace token detection has made the training computing more efficient. In fact, replace token detection developed based on training generator (GAN) and discriminator models [138].

XLNet [139] is one of the unsupervised language representation models that has received a lot of attention these days. XLnet is an extension of the Transformer-XL model published in 2019 by Yang et al. [139]. Compared to BERT, XLNet was trained with a larger dataset. Consequently, it outperforms other models like BERT and ALBERT in various downstream NLP tasks. The increase in performance is the result of improvement in the model. XLNet was developed based on generalized auto-regressive pre-training Language models by avoiding the drawback of the BERT model. BERT faces some problems like discrepancies between the pre-training process and the fine-truing process, because of the mask language model (MLM). However, XLNet replaces this task with permutations over the factorization task [139].

LLaMA [145] is a cutting-edge language model which is available in a range of sizes, ranging from 7 billion to 65 billion parameters, with each size trained on various token quantities, such as 1 trillion and 1.4 trillion tokens [145]. LLaMA trained based on texts

from the 20 most widely spoken languages, especially those written in the Latin and Cyrillic scripts. A few examples of its training data sources are public websites and forums like CommonCrawl, GitHub repositories, Wikipedia in many languages, Project Gutenberg's public domain books, scientific publications from ArXiv, and Q&A from Stack Exchange websites. LLaMA just like other large language models uses next-word prediction algorithms in the training phase. LLaMA is flexible, accommodating a wide range of use cases, and also shows enhanced performance in several natural language processing tasks. LLaMA model has been shown to outperform other open language models in multiple areas according to the Open LLM Leaderboard[5].

Falcon [146] is a large language model that the Technology Innovation Institute (TII) in the UAE has developed. It consists of the Falcon-7B and Falcon-40B models, where the digits represent the number of parameters in each model. Notably, Falcon-40B has outperformed GPT and LLaMa based on the Open LLM Leaderboard. Falcon is trained by high-quality training data, which is mostly drawn from RefinedWeb, a sizable online dataset created from CommonCrawl [146]. By using the multi-query attention method, it improves the model architecture for inference and paves the way for creative applications. Falcon is also open source and transparent.

[5] `https://llm-leaderboard.streamlit.app/`

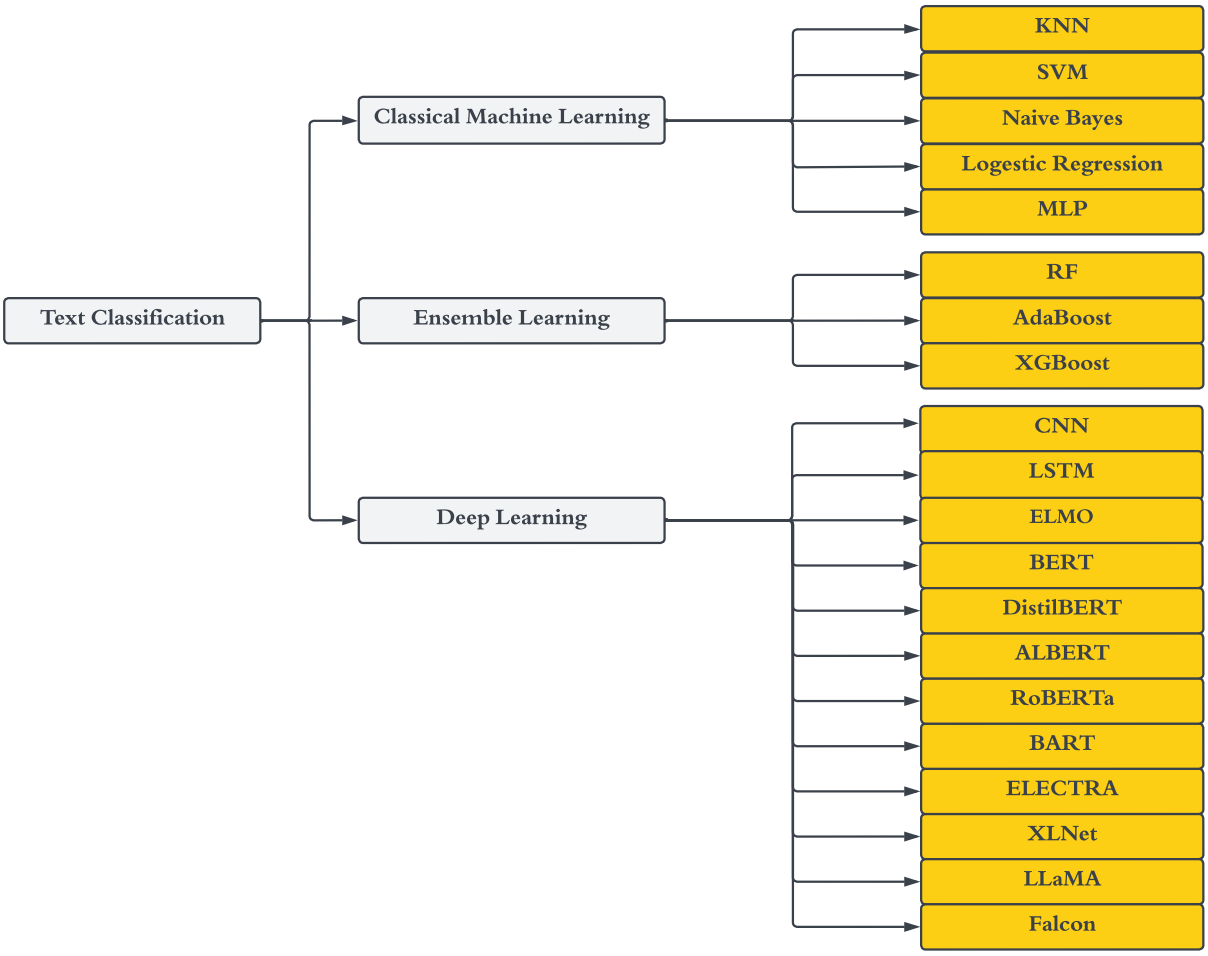


Figure 7: Taxonomy of text classification methods. The methods are highlighted in orange since it is the focus of this paper.

3.3. ***Classification Algorithms***

In this section, we present the most popular classification algorithm used in fake news detection. The algorithm was chosen based on its importance in previous works that are presented in Table 2. With that in mind, we categorized the classification algorithms into three main categories: Classical machine learning models, Ensemble learning models, and Deep learning models (Figure 7). We briefly describe these algorithms in the following sections.

3.3.1. ***Classical Machine Learning Models***

Classical machine learning algorithms cover a large range of algorithms consisting of probabilistic models (e.g., Naive Bayes, Logistic Regression), memory-based models (e.g., K-nearest neighbors, Support vector machine), and rule-based models (e.g., Decision Tree).

K-nearest neighbors (KNN) is a memory-based model that determines the class of a new instance based on the class of instances closest to it in the feature space. There are a few studies that used KNN for fake news detection because it is a lazy learner [56, 19, 119, 50, 110, 51]. KNN is very sensitive to both irrelevant characteristics and the size of datasets. Also, it might require a lot of computing resources and high memory because it stores all of the training.

Support vector machine (SVM) is an algorithm that tries to maximize the distances between instances of different classes by creating a hyperplane that maximizes the separation between them. This is achieved with the help of the support vectors, which are instances closer to the hyperplane that influences its position and orientation. classification is performed according to the position of an input instance to the hyperplane. In the case of non-linear data, the kernel trick is used to transform the data to a space where the points are linearly separable. The SVM model is usually a robust alternative for text classification since it can easily deal with a limited amount of data and high dimensional spaces. Also, it can obtain good generalization performance due to the maximum margin concept. As such, many studies used SVM for fake news detection [44, 21, 17, 53, 59, 54, 60, 55, 117, 46, 47, 19].

Naive Bayes (NB) is one of the probabilistic classifiers that developed based on Bayes theorem with features independence or "naive" assumption (i.e., the presence of a feature in a dataset is unrelated to the presence of other features). We can use the Naive Bayes classifier in three formats according to the probabilistic distribution we are modeling: Bernoulli Naive Bayes (BNB) for binary data, Multinomial Naive Bayes (MNB) for categorical data, and Gaussian Naive Bayes (GNB) for continuous data. Several studies used the Naive Bayes classifier for fake news detection [54, 45, 46, 64, 51, 71] since it obtains very good performance when used in conjunction with count-based feature extraction methods. However,

this method is limited by the naive assumption and, as such, cannot model the interaction between words in text data.

Logistic regression (LR) is one of the probabilistic classifiers. It is similar to the regression model except that the outcome variable should be categorical. We can use this model in three formats, Binomial Logistic Regression, Multinomial logistic regression, Ordinal logistic regression. Some studies used Logistic Regression for fake news detection [44, 21, 53, 60, 46, 47, 58, 51, 52].

Multi-layer Perceptron (MLP) is a type of artificial feed-forward neural network model that consists of one or more layers of neurons, including an input layer, one or more hidden layers, and an output layer. Each neuron in the input layer receives a vector of input values, which are then processed by the neurons in the hidden layers. The output layer produces the final output of the network. Each neuron in the network has a set of weights learned during training to produce the desired output. The neurons in the hidden layers use activation functions to transform the input into a non-linear space, allowing the network to extract more abstract representations and capture complex relationships between the input data. When used for text classification, MLP-based models are often used in conjunction with classical feature extraction techniques or word embedding techniques, which transform text data into numerical vectors that this model can process. The model then learns to identify patterns and relationships between the numerical representations of the text data and the corresponding labels (i.e., whether the news is real or fake). As such, MLP neural network has been successfully utilized in a few studies for fake news detection [60, 55, 56, 61, 47, 63, 68, 62, 49, 52].

3.3.2. ***Ensemble models***

An ensemble model is a learning system that uses multiple base classifiers to achieve better predictive performance [147]. The goal of an ensemble is to achieve better accuracy than any individual classifier, as the strengths of one model may compensate for the weakness of others. In other words, they complement each other and lead to more robust predictions. Several works used ensemble models for fake news detection [17, 59, 50, 56, 61, 19, 57, 48,

119, 110, 51, 71, 52]. The most common ensemble methods used in fake news detection are based on variants of the Bagging method (e.g., Random Forests [148]), and Boosting [149].

Random forest (RF) developed by Leo Breiman [148] based on the bagging algorithm. This algorithm selects a bootstrap of the training data to form the training set of each individual base model. As each base model is trained with a different bootstrap (i.e., sampling with replacement), they lead to a set of diverse models at the end and help reduce the model's variance [148]. The Random Forest model is a particular implementation of this algorithm where the features and samples are randomly sampled with replacement to generate a set of diverse trees. Then it ultimately makes a prediction based on the majority vote of the prediction results of all decision trees. The Random Forest model improves performance based on reducing variance. Several studies used the random forest for fake news detection [17, 56, 48, 58].

AdaBoost is a short form of Adaptive Boosting developed by Yoav Freund and Robert Schapire in 1995 [150] based on the boosting algorithm. Unlike bagging methods, where the training phase is parallel, in boosting methods, the training phase is sequential. AdaBoost is one of the prominent boosting algorithms. In this method, each classifier trains based on the result of previous classifiers in an iterative process. To be more specific, in the first iteration, each sample has equal weights, and after each training iteration, the weight of the sample is redistributed based on the previous training iteration. Samples that were classified incorrectly have their weights increased in order to give more focus on learning instances that were misclassified by previous models. Some studies used AdaBoost for fake news detection [19, 56, 51].

XGBoost is a short form of eXtreme Gradient Boosting which is an optimized implementation of gradient boosting algorithms for learning end-to-end scalable tree boosting systems. Unlike AdaBoost, which increased the weights of hard instances of gradient boosting, try to minimize a loss function that can be defined by the user. Several studies used XGBoost for fake news detection [17, 55, 56, 57, 58, 119, 51, 71, 52].

3.3.3. ***Deep learning models***

Deep learning models are a type of machine learning and artificial intelligence that does not require manual feature extraction. Instead, they automatically learn feature representations from raw data. Deep learning models have shown promising results in many tasks and are increasingly used in text classification. However, they also require more computational resources than traditional machine-learning models.

Several studies have investigated the use of deep learning models for fake news detection. Table 2 shows previous works that used deep learning models. In this section, we describe some of the most popular deep learning models that have been used for fake news detection.

Convolutional neural network (CNNs) was first introduced by Kunihiko Fukushima in 1980 and later developed by LeCun and his colleagues for various advanced computer vision tasks [151]. CNNs consist of three layers: a convolutional layer, a non-linear layer, and a pooling layer. When dealing with text classification, the convolutional layers are used to extract local features from a word representation matrix using a sliding kernel. In the non-linear layer, the values of local features are subjected to a non-linear activation function (e.g., ReLU). Then all the local features are pooled in the pooling layer to create universal features. Because CNNs can subject multiple sections of a sequence to convolutions defined by various kernels at the same time, they are commonly used for natural language processing tasks such as text classification. Although CNNs use fewer parameters for training due to the use of convolutional kernels, their design can be complex, and they can be slow if a large number of hidden layers are used [152]. Some studies have used CNNs for fake news detection [21, 59, 60, 67, 61, 62, 63, 68, 69, 50, 70].

Long Short Term Memory (LSTM) is a type of recurrent neural network (RNN) model that is commonly used for sequential data proposed. RNNs are designed to recall information from earlier neurons for future processes. However, one major disadvantage of RNNs is that they struggle to handle long-term dependencies in excessively long sequences due to gradient vanishing and exploding problems [151]. LSTM proposed by Hochreiter et al. [153] addresses this issue by creating a memory cell that can recall information from any

length of time. Its ability to capture long-term dependencies in sequential data makes it well-suited for tasks that involve analyzing text data, such as identifying the presence of misleading or fabricated information in news articles. LSTM-based models have been used to extract features from text data, and these features are then used as input to a classifier that can distinguish between real and fake news and have been used in several studies for fake news detection [21, 59, 60, 123, 67, 61, 62, 63, 68, 49, 50, 64, 66].

Transformers are a type of deep learning model proposed by Vaswani et al. [154] that uses self-attention mechanisms to weigh input data. They have proven to be highly effective for natural language processing (NLP) tasks. Traditional CNN and RNN models can be computationally expensive, especially when working with larger sentences. However, transformers can model the importance and relationships between words more efficiently by using parallel computing techniques to calculate the attention score of each word.

There are two main approaches for using transformers in NLP tasks: fine-tuning-based and feature-based approaches. In the fine-tuning-based approach, we can fine-tune transformer models for specific NLP tasks, such as text classification [151] and fake news detection [71]. In this regard, a pre-trained model that is trained over a large amount of text can be fine-tuned on a specific domain or dataset when its parameters get updated on that specific task. To be more specific, during the fine-tuning, the model is initialized by the weights that came from the pre-trained model and then trained over our desired domain [133]. It is important to note that the earlier layers of a pre-trained language model acquire broad language knowledge while the later layers learn task-specific patterns and representations [133]. So in the fine-tuning process, it is preferable only to update the parameters of the later layers and freeze the earlier layers to maintain the pre-trained representations [133].

In the feature-based approach, we can extract contextualized embeddings from the encoding layer of the transformer and then use the extracted embeddings in a downstream task like text classification. In the feature-based approaches, after updating the parameters (later layers) based on the domain-specific datasets, we can extract the output (hidden states) of the layers that are considered contextualized embeddings that collect knowledge from each layer. Then these hidden states are aggregated by applying layer-wise concatenation in order

to merge the hidden states into a fixed dimensional representation. After that, we can use this fixed dimensional representation as feature vectors for a downstream task like fake news detection.

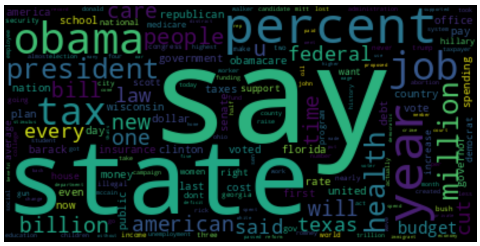
There are several transformer models available, including ELMO, BERT, DistilBERT, ALBERT, RoBERTa, BART, ELECTRA, XLNet, LLAMA, and Falcon which are commonly used in text classification tasks. According to recent literature, transformers have also been successfully used for fake news detection, which motivates their usage in this comparative analysis. We will briefly describe these specific models and discuss their main characteristics in Section 3.2.2.2.

4. Comparative Study

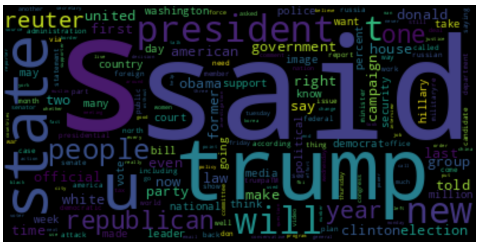
In this section, we address the research questions posed by this study through an empirical comparison of 20 state-of-the-art classification algorithms and 15 feature representation techniques. The classification models Include 5 classical machine learning models, 3 ensemble models, and 12 deep learning models. Also, the feature representation techniques include 2 count-based methods, 3 context-independent methods, and 10 context-dependent methods. It is worth highlighting that we used transformer models in both feature extraction and end-to-end classification models and all models have been evaluated under the same experimental protocol over four benchmark fake news datasets. Figure 7, 6 show the details of the models used in this experiment.

4.1. ***Datasets***

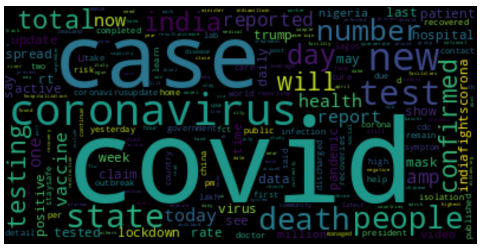
This comparative study is based on four benchmark fake news datasets covering different topics, such as politics, health, business, and technology, to avoid research bias and draw meaningful conclusions. Figure 8 presents the word clouds of each dataset. Furthermore, both binary and multi-class fake news detection datasets are represented in this set. We have chosen these datasets based on previous research on fake news detection since this study focuses on textual-based fake news detection. The main characteristics of each dataset are presented in Table 3.



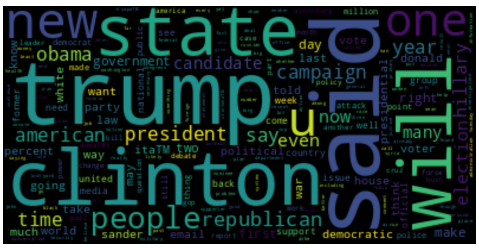
(a) Word cloud of LIAR dataset



(b) Word cloud of ISOT dataset



(c) Word cloud of COVID dataset



(d) Word cloud of George McIntire dataset

Figure 8: Word clouds of each dataset used in the study

Liar [21] is a manually labeled dataset that has been curated from PolitiFact.com (a fact-checking organization based in the United States) and consists of around 12.8k short statements. These statements span a range of contexts, such as news, television interviews, and radio interviews, collected from 2007 to 2016 [1, 21]. Each row of the dataset comprises a news statement, along with a label from one of six categories (Pants-fire, False, Barely-True, Half-True, Mostly-True, True), the subject of the text, the name of the speaker, the speaker's job title, the state information, the party affiliation, and the context (location of the speech or statement). However, in this study, we just used the news statement in our implementation since we are just concerned with textual-based fake news detection. In addition, there are two different formulations of this dataset, one consisting of the original six classes and another one transforming it into a binary classification problem by merging the classes *Pants-fire, False, Barely-True* as fake, and *Half-True, Mostly-True, True* as real news. However, we only considered this dataset's original six classes version in this study.

The **ISOT** dataset [44, 53] is comprised of two CSV files containing 21,417 real news articles and 23,481 fake news articles, respectively. The real news sources originated from Reuters.com, while the fake news sources were sourced from PolitiFact.com. The dataset covers various contexts, such as world news and political news. Each row in the dataset includes the article title, text, type, and publication date. However, in this study, we just used the text in our implementation since we are just concerned with textual-based fake news detection.

The **George McIntire (GM)** [155] dataset comprises 11,000 real news articles and 3,151 fake news articles from various mainstream media sources, including the New York Times, Wall Street Journal, Bloomberg, and the Guardian. The articles mostly relate to politics, business, technology, entertainment, and other topics. The news articles have been fact-checked by journalists to create the dataset labels.

The **COVID** dataset [156] was gathered for the purpose of detecting fake news related to the COVID-19 pandemic. Articles were sourced from Twitter and verified by authoritative sites such as politifact.com and snopes.com. Comprising 10,700 entries written in English, the dataset is composed of two categories: real and fake news.

Dataset	Domain	Media	Fact-checking	Size	No.Class	Class distribution
Liar [21]	*Politics*	*Mainstream media*	*Editors & journalists*	*12836*	*6*	(1050,2511,2108,2638,2466,2063)
ISOT [53, 44]	*Politics & World news*	*Mainstream media*	*Fact-checking websites*	*44866*	*2*	(23448,21417)
George McIntire [155]	*Business,Technology and etc.*	*Mainstream media*	*journalists*	*11000*	*2*	(3151,3159)
Covid [156]	*Covid-19 & Health*	*Twitter*	*Fact-checking websites*	*10700*	*2*	(5100,5600)

Table 3: Main characteristics of the benchmark dataset used in this study. "Class distribution" shows the (fake, real) instances.

4.2. ***Experimental Setup***

Before conducting our experiments, we pre-processed the data as outlined in Section 3.1, including converting all words to lowercase, removing numbers, punctuation, symbols, white spaces, and stop words in order to make the text clean and analyzable. We also applied normalization and stemming processes using the NLTK [157] library version 3.7. Finally, we used the feature representation methods described in Section 3.2.

Feature extraction. After obtaining the clean text, we move to the next step of extracting multiple representations from the text. In this step, we incorporate all 13 techniques described in Section 3.2. Our count-based methods include TF (Term Frequency) and TF-IDF (Term frequency-inverse document frequency). These techniques were implemented using Scikit-Learn [158] version 1.0.1 library.

Among context-independent methods, we employed Word2Vec, GloVe, and fastText using Gensim [159] version 3.6.0 and Zeugma[6] version 0.49. All of the word embedding models used in this study were trained on large online corpora. For Word2Vec, we used a model developed from a Google News corpus with 300-dimensional feature vectors. For GloVe, we used a model based on the WikiGigaword corpus with 300-dimensional feature vectors. For fastText, we employed a model developed from the Wikipedia corpus with 300-dimensional feature vectors for each sample. For all word embedding methods, we computed the average of the word feature vectors in each news to generate its feature representation for classical machine learning approaches.

From context-dependent methods, we include BERT ('bert-base-uncased'), DistilBERT

[6]https://zeugma.readthedocs.io/en/latest/

('distilbert-base-uncased'), ALBERT ('albert-base-v2'), RoBERTa ('roberta-base'), BART ('facebook-bart-large'), ELECTRA ('google-electra-base-discriminator'), XLNet ('xlnet-base-cased'), Falcon ('Rocketknight1/falcon-rw-1b'), ELMO ('elmo2') and LLaMA that we obtained the weights by sending an email request, completing the necessary form, and subsequently converting them to the Hugging Face Transformers format. All the implementation of transformers was facilitated by the Hugging Face framework[7] [160], Tensorflow Hub[8] and PyTorch [161] version 1.11.0. All transformer models yield fixed feature vectors of 768 dimensions extracted from their respective pooling layers, with the exception of ELMO, LLAMA, and Falcon, which use feature vectors of 1024, 2048, and 4096 dimensions, respectively. In this experiment, we aggregate the hidden state of the last four layers and freeze the parameters of the initial layers.

Classification algorithms. In this experiment, we choose the most popular classification algorithms in fake news detection, as described in Section 3. The classification models are divided into three main categories based on our proposed taxonomy: classical machine learning, ensemble models, and deep learning models.

Classical machine learning and ensemble models employed in this study include Support Vector Machine (SVM), Logistic Regression (LR), K-Nearest Neighbors (KNN), Naive Bayes (NB), Multi-layer Perceptron (MLP), Random Forest (RF), Adaptive Boosting (AdaBoost), and Extreme Gradient Boosting (XGBoost). These models were implemented using Scikit-learn version 1.0.1.

Deep learning models used in this study: Convolutional neural networks (CNNs), Bidirectional Long Short Term Memory (BiLSTM), and Transformers. The CNN and BiLSTM architectures used in this work were defined based on previous work in text classification [162, 151]. We implemented deep models using TensorFlow [163] version 2.8.2. In this experiment, we not only used transformer models to extract text representation but also used them in a fine-tuning way (End-to-End Classifiers). So similar to the feature-based approach, we freeze the parameter of the initial layers and just train the parameters of the

[7] https://github.com/huggingface/transformers

[8] https://www.tensorflow.org/hub

last four layers.

The hyperparameter tuning procedure was conducted using a 5-fold cross-validation procedure with stratified sampling to avoid any class bias in the analysis. The hyperparameter configuration for classical machine learning and deep learning models is shown in Appendix A (Table A.13) and Appendix B (Table B.14). The architecture of the deep learning models is also presented in Appendix C (Figure C.11). To determine the performance of the classifiers, we used the F1-score, which was found to be more robust compared to other metrics when dealing with datasets of varying class distributions [164]. The experiments were conducted on a CPU-based server with two Intel Xeon Silver 4310 processors and 512GB RAM; and a GPU-based server with one NVIDIA RTX A6000 processor and 256GB of RAM.

5. Results and discussion

In this section, we aim to answer the following research questions: *(**RQ1**) What is the impact of feature extraction methods on model performance? (**RQ2**) Does the use of transformer models as representation learning have higher performance or as fine-tuning? (**RQ3**) Would combining feature extraction methods improve performance in the fake news detection task? (**RQ4**) Which methods are more cost-effective?*

In order to answer RQ1 and RQ2 in Section 5.1, we compare the model's performance based on different feature representations in different datasets. In Section 5.2, we analyze how different feature representations and classifier models complement each other by conducting an error analysis and demonstrate to what extent their combination can be an interesting alternative to building robust fake news detection systems. In Section 5.3, we answer RQ4 by measuring the training time for each model and comparing it with their corresponding performance.

*5.1. **Comparison of feature extraction and classification techniques***

Tables 4, 5, 6, and 7 show the average performance across 15 feature representation approaches of 10 classification models for each dataset. Each row refers to a feature representation technique, and each column represents a particular classification model. Addition-

ally, the last column shows the performance of 10 end-to-end transformer models labeled "End-to-End". It is crucial to emphasize that in this experiment, we employ the transformer models as both feature representation techniques and end-to-end classifiers. The results report is based on the average F1-score and standard deviation over 5-fold cross-validation. The models marked with an (*) sign mean that they were excluded from the test for reasons such as lack of significance.

Features	SVM	LR	KNN	NB	RF	AdaBoost	XGBoost	MLP	CNN	BiLSTM	End-To-End
TF	***0.256 (0.011)***	*0.245 (0.002)*	*0.219 (0.006)*	*0.242 (0.007)*	*0.246 (0.008)*	*0.223 (0.011)*	*0.228 (0.006)*	*0.226 (0.005)*	*	*	*
TF-IDF	***0.246 (0.006)***	*0.242 (0.005)*	*0.228 (0.004)*	*0.239 (0.011)*	*0.236 (0.006)*	*0.218 (0.010)*	*0.220 (0.009)*	*0.222 (0.004)*	*	*	*
Word2Vec	***0.248 (0.006)***	***0.248 (0.006)***	*0.223 (0.006)*	*0.219 (0.010)*	*0.250 (0.012)*	*0.244 (0.006)*	*0.235 (0.008)*	*0.228 (0.008)*	*0.225 (0.003)*	*0.224 (0.002)*	*
GloVe	***0.242 (0.008)***	*0.240 (0.007)*	*0.212 (0.005)*	*0.225 (0.005)*	***0.242 (0.006)***	*0.237 (0.006)*	*0.228 (0.003)*	*0.235 (0.006)*	*0.218 (0.012)*	*0.229 (0.001)*	*
FastText	***0.245 (0.009)***	*0.242 (0.007)*	*0.222 (0.006)*	*0.218 (0.007)*	*0.243 (0.008)*	***0.245 (0.009)***	*0.239 (0.010)*	*0.233 (0.010)*	*0.226 (0.011)*	*0.242 (0.008)*	*
ELMO	***0.264 (0.006)***	*0.258 (0.010)*	*0.232 (0.012)*	*0.218 (0.017)*	*0.256 (0.008)*	*0.253 (0.004)*	*0.249 (0.005)*	*0.230 (0.006)*	*0.221 (0.014)*	*0.231 (0.016)*	*0.210 (0.001)*
BERT	***0.256 (0.001)***	*0.253 (0.003)*	*0.220 (0.003)*	*0.225 (0.003)*	*0.237 (0.005)*	*0.237 (0.005)*	*0.217 (0.005)*	*0.209 (0.011)*	*0.190 (0.003)*	*0.210 (0.000)*	*0.210 (0.003)*
DistilBERT	***0.252 (0.000)***	*0.224 (0.007)*	*0.210 (0.006)*	*0.223 (0.005)*	*0.248 (0.006)*	*0.240 (0.006)*	*0.222 (0.001)*	*0.216 (0.009)*	*0.190 (0.008)*	*0.210 (0.003)*	*0.220 (0.008)*
ALBERT	***0.247 (0.003)***	*0.228 (0.006)*	*0.205 (0.005)*	*0.221 (0.000)*	*0.230 (0.004)*	*0.234 (0.003)*	*0.224 (0.000)*	*0.218 (0.002)*	*0.230 (0.003)*	*0.240 (0.001)*	*0.180 (0.000)*
BART	***0.255 (0.007)***	*0.236 (0.003)*	*0.217 (0.002)*	*0.226 (0.003)*	*0.241 (0.004)*	*0.249 (0.001)*	*0.220 (0.005)*	*0.209 (0.006)*	*0.210 (0.000)*	*0.230 (0.005)*	*0.200 (0.001)*
RoBERTa	*0.238 (0.005)*	*0.238 (0.001)*	*0.203 (0.005)*	*0.226 (0.004)*	*0.232 (0.005)*	*0.241 (0.003)*	*0.221 (0.002)*	*0.218 (0.000)*	***0.250 (0.003)***	***0.250 (0.001)***	*0.190 (0.010)*
ELECTRA	***0.241 (0.000)***	*0.222 (0.003)*	*0.197 (0.008)*	*0.218 (0.002)*	*0.231 (0.002)*	*0.229 (0.003)*	*0.215 (0.000)*	*0.206 (0.003)*	*0.220 (0.007)*	*0.230 (0.006)*	*0.190 (0.002)*
XLNET	*0.217 (0.001)*	*0.221 (0.003)*	*0.196 (0.009)*	*0.218 (0.001)*	*0.228 (0.002)*	***0.237 (0.006)***	*0.216 (0.002)*	*0.205 (0.005)*	*0.210 (0.006)*	*0.220 (0.002)*	*0.210 (0.003)*
LLaMA	***0.266 (0.001)***	*0.246 (0.005)*	*0.226 (0.008)*	*0.220 (0.002)*	*0.252 (0.002)*	*0.244 (0.001)*	*0.242 (0.000)*	*0.244 (0.009)*	*0.252 (0.002)*	*0.258 (0.007)*	*0.261 (0.003)*
Falcon	***0.258 (0.000)***	*0.251 (0.003)*	*0.227 (0.007)*	*0.220 (0.002)*	*0.241 (0.000)*	*0.223 (0.003)*	*0.261 (0.001)*	*0.242 (0.003)*	*0.250 (0.005)*	*0.256 (0.000)*	***0.258 (0.001)***

Table 4: Average F1-score and standard deviation obtained on the Liar dataset. Each row represents a feature representation technique, while each column represents a classification model. The last column showcases the performance of end-to-end transformer models. The best results for each feature representation are highlighted in bold. The cells marked with an (*) indicate that the classification models and feature representation techniques in the corresponding rows and columns are incompatible.

As seen in Table 4, classical machine learning models perform relatively better than ensemble and deep learning models in the Liar dataset. The SVM model using LLaMA as a feature representation obtains the best result, with a score of 0.266. Furthermore, feature-based approaches attain significantly better performance than fine-tuning approaches among the transformer models in this dataset. The transformer models that obtained the highest F1-score with fine-tuning were LLaMA and Falcon, which obtained 0.261 and 0.258, respectively.

There are several reasons for the lack of accuracy between the different classifiers and feature extraction techniques in the multi-class Liar dataset. First of all, it is difficult for classifiers to distinguish between the classes in the Liar dataset since the classes are am-

biguous and tightly positioned in terms of their feature distributions. So, in this case, the feature extraction techniques did not have enough representation power to collect discriminative information to distinguish between the six classes. As a result, making it difficult for classifiers to produce distinctive predictions. Furthermore, the proximity of the classes causes substantial overlap in the class predictions made by several classifiers. These findings make it clear that a different modeling strategy is needed for the Liar dataset to consider its particular characteristics. These difficulties may not have been present in earlier studies that mostly employed the binary form of the dataset. Ordinal regression [165, 166] is a possible alternative modeling strategy in this case. Ordinal regression (also known as ordinal classification) methods are created primarily to deal with situations where class names have an intrinsic hierarchy or order. Thus, the relationships between the classes in the Liar dataset can be better modeled using an ordinal regression approach since their labels indicate the degree to which a news article is likely a fake, which should be considered in future approaches.

Features	SVM	LR	KNN	NB	RF	AdaBoost	XGBoost	MLP	CNN	BiLSTM	End-To-End
TF	*0.997 (0.001)*	*0.997 (0.000)*	*0.845 (0.005)*	*0.983 (0.001)*	*0.993 (0.001)*	***0.998 (0.001)***	***0.998 (0.000)***	*0.997 (0.001)*	*	*	*
TF-IDF	*0.995 (0.001)*	***0.997 (0.000)***	*0.706 (0.005)*	*0.983 (0.001)*	*0.993 (0.001)*	***0.997 (0.000)***	***0.997 (0.000)***	*0.994 (0.001)*	*	*	*
Word2Vec	***0.987 (0.002)***	*0.975 (0.001)*	*0.941 (0.003)*	*0.879 (0.003)*	*0.956 (0.002)*	*0.956 (0.001)*	*0.979 (0.001)*	*0.983 (0.001)*	*0.979 (0.007)*	*0.986 (0.003)*	*
GloVe	*0.917 (0.002)*	*0.886 (0.002)*	*0.897 (0.003)*	*0.796 (0.001)*	*0.916 (0.002)*	*0.880 (0.002)*	*0.928 (0.002)*	*0.909 (0.003)*	*0.967 (0.005)*	***0.995 (0.001)***	*
fastText	*0.982 (0.001)*	*0.979 (0.001)*	*0.947 (0.002)*	*0.880 (0.003)*	*0.963 (0.002)*	*0.964 (0.001)*	*0.983 (0.001)*	*0.985 (0.001)*	*0.983 (0.003)*	***0.991 (0.006)***	*
ELMO	*0.988 (0.001)*	*0.988 (0.001)*	*0.936 (0.003)*	*0.809 (0.003)*	*0.949 (0.001)*	*0.960 (0.001)*	*0.980 (0.001)*	***0.992 (0.001)***	*0.869 (0.010)*	*0.889 (0.011)*	*0.980 (0.003)*
BERT	*0.984 (0.001)*	*0.986 (0.001)*	*0.966 (0.002)*	*0.910 (0.000)*	*0.971 (0.002)*	*0.980 (0.004)*	*0.973 (0.000)*	*0.985 (0.003)*	***1.000 (0.002)***	***1.000 (0.004)***	*0.950 (0.001)*
DistilBERT	*0.982 (0.001)*	*0.985 (0.001)*	*0.962 (0.000)*	*0.915 (0.001)*	*0.971 (0.003)*	*0.976 (0.003)*	*0.968 (0.005)*	*0.986 (0.001)*	***1.000 (0.005)***	***1.000 (0.003)***	*0.990 (0.001)*
ALBERT	*0.982 (0.000)*	*0.990 (0.002)*	*0.973 (0.001)*	*0.941 (0.001)*	*0.973 (0.000)*	*0.981 (0.002)*	*0.975 (0.000)*	*0.988 (0.001)*	*0.990 (0.001)*	***1.000 (0.002)***	*0.990 (0.008)*
BART	*0.993 (0.001)*	*0.993 (0.001)*	*0.969 (0.001)*	*0.917 (0.000)*	*0.976 (0.002)*	*0.982 (0.001)*	*0.981 (0.001)*	*0.992 (0.001)*	***1.000 (0.002)***	***1.000 (0.008)***	***1.000 (0.004)***
RoBERTa	*0.970 (0.001)*	*0.992 (0.002)*	*0.966 (0.001)*	*0.918 (0.002)*	*0.978 (0.000)*	*0.983 (0.001)*	*0.979 (0.000)*	*0.991 (0.001)*	***1.000 (0.001)***	***1.000 (0.003)***	*0.990 (0.011)*
ELECTRA	*0.972 (0.001)*	*0.979 (0.001)*	*0.948 (0.003)*	*0.847 (0.001)*	*0.961 (0.002)*	*0.968 (0.001)*	*0.960 (0.001)*	*0.979 (0.001)*	*0.990 (0.003)*	*0.990 (0.001)*	***1.000 (0.002)***
XLNET	*0.963 (0.002)*	*0.986 (0.001)*	*0.919 (0.003)*	*0.926 (0.002)*	*0.963 (0.002)*	*0.970 (0.003)*	*0.965 (0.001)*	*0.982 (0.001)*	*0.980 (0.005)*	***0.990 (0.003)***	***0.990 (0.001)***
LLaMA	*0.999 (0.000)*	*0.999 (0.003)*	*0.990 (0.007)*	*0.950 (0.002)*	*0.991 (0.002)*	*0.990 (0.003)*	*0.998 (0.000)*	*0.999 (0.003)*	*0.998 (0.004)*	***1.000 (0.001)***	***1.000 (0.003)***
Falcon	*0.998 (0.000)*	*0.999 (0.003)*	*0.990 (0.008)*	*0.930 (0.002)*	*0.991 (0.002)*	*0.990 (0.003)*	*0.997 (0.000)*	*0.999 (0.003)*	*0.996 (0.009)*	***1.000 (0.000)***	***1.000 (0.002)***

Table 5: Average F1-score and standard deviation obtained on the ISOT dataset. Each row represents a feature representation technique, while each column represents a classification model. The last column showcases the performance of end-to-end transformer models. The best results for each feature representation are highlighted in bold. The cells marked with an (*) indicate that the classification models and feature representation techniques in the corresponding rows and columns are incompatible

According to Table 5, deep learning models perform significantly better in the ISOT dataset than the classical machine learning and ensemble learning models. We can see

this difference in deep learning models when using context-dependent feature representation techniques. For example, in the combination of BiLSTM or CNN with methods such as BERT, DistilBERT, BART, RoBERTa, LLaMA, and Falcon we can see up to 100% F1-score. Also, among ensemble models, AdaBoost and XGBoost perform better, especially when using count-based feature representation methods like TF or TF-IDF. In the ISOT dataset, among the transformer models, in BERT, DistilBERT, ALBERT, and RoBERTa, feature-based approaches outperform the fine-tuning approaches. But in BART, ELECTRA, XLNET, LLaMA, and Falcon, fine-tuning performs better than the feature-based approaches. Table 5 demonstrates that various combinations of feature extraction techniques and classifiers yield high performance on the ISOT dataset. It is important to stress that datasets like ISOT may have limited significance for finding strategies that have a good chance of addressing the fake news detection problem. This is due to the fact that attaining good performance on such datasets does not imply that one would always be able to generalize and correctly categorize fake news and factual news. As a result, caution is advised when evaluating the effectiveness of techniques based only on how well they perform on datasets like ISOT. It is critical to analyze a technique's robustness, generalizability, and applicability across many settings and datasets in order to accurately assess its potential for solving an issue.

According to Table 6, in the COVID dataset, deep learning models have significantly better performance compared to the classical machine learning and ensemble learning models. We can see this difference in deep learning models when using context-dependent feature representation techniques. The BiLSTM model trained with BART, LLaMA, and Falcon as a feature representation achieves the best result in the COVID dataset (95%). Also, among classical machine learning models, SVM and Logistic regression perform better, especially when using count-based feature representation methods or context-independent methods like GloVe or Fasttext. In the COVID dataset, among the transformer models, just ELMO and DistilBERT fine-tuning approaches outperform the feature-based approaches.

According to Table 7, deep learning models exhibit significantly improved performance over classical machine learning and ensemble learning models when utilizing context-dependent

Features	SVM	LR	KNN	NB	RF	AdaBoost	XGBoost	MLP	CNN	BiLSTM	End-To-End
TF	***0.920 (0.004)***	***0.920 (0.002)***	*0.750 (0.007)*	*0.916 (0.004)*	*0.906 (0.004)*	*0.894 (0.008)*	*0.909 (0.005)*	*0.908 (0.009)*	*	*	*
TF-IDF	***0.928 (0.004)***	*0.924 (0.004)*	*0.889 (0.008)*	*0.916 (0.004)*	*0.903 (0.004)*	*0.892 (0.004)*	*0.904 (0.005)*	*0.919 (0.004)*	*	*	*
Word2Vec	***0.920 (0.007)***	*0.892 (0.005)*	*0.875 (0.005)*	*0.793 (0.008)*	*0.881 (0.007)*	*0.882 (0.008)*	*0.903 (0.003)*	*0.909 (0.007)*	*0.896 (0.003)*	*0.912 (0.006)*	*
GloVe	*0.858 (0.005)*	*0.799 (0.003)*	*0.832 (0.012)*	*0.719 (0.009)*	*0.855 (0.005)*	*0.820 (0.008)*	*0.853 (0.010)*	*0.846 (0.011)*	*0.883 (0.001)*	***0.914 (0.004)***	*
fastText	***0.906 (0.005)***	*0.888 (0.003)*	*0.831 (0.007)*	*0.824 (0.009)*	*0.877 (0.009)*	*0.888 (0.011)*	*0.904 (0.007)*	*0.904 (0.004)*	*0.892 (0.009)*	*0.902 (0.006)*	*
ELMO	*0.921 (0.008)*	*0.909 (0.002)*	*0.893 (0.008)*	*0.772 (0.013)*	*0.885 (0.003)*	*0.895 (0.005)*	*0.911 (0.006)*	*0.924 (0.003)*	*0.769 (0.002)*	*0.841 (0.001)*	***0.935 (0.001)***
BERT	*0.911 (0.004)*	*0.885 (0.004)*	*0.901 (0.001)*	*0.858 (0.001)*	*0.895 (0.006)*	*0.901 (0.001)*	*0.869 (0.006)*	*0.917 (0.003)*	***0.940 (0.003)***	*0.935 (0.002)*	*0.880 (0.002)*
DistilBERT	*0.909 (0.003)*	*0.893 (0.002)*	*0.903 (0.001)*	*0.862 (0.003)*	*0.895 (0.005)*	*0.899 (0.001)*	*0.870 (0.001)*	*0.917 (0.006)*	***0.940 (0.006)***	*0.930 (0.001)*	***0.940 (0.006)***
ALBERT	*0.892 (0.005)*	*0.886 (0.006)*	*0.881 (0.003)*	*0.821 (0.002)*	*0.884 (0.008)*	*0.893 (0.002)*	*0.860 (0.001)*	*0.904 (0.003)*	*0.929 (0.009)*	***0.930 (0.008)***	*0.840 (0.001)*
BART	*0.919 (0.002)*	*0.898 (0.003)*	*0.907 (0.005)*	*0.858 (0.002)*	*0.893 (0.010)*	*0.903 (0.006)*	*0.875 (0.000)*	*0.906 (0.001)*	*0.931 (0.001)*	***0.950 (0.001)***	*0.890 (0.002)*
RoBERTa	*0.881 (0.009)*	*0.899 (0.003)*	*0.886 (0.002)*	*0.844 (0.005)*	*0.887 (0.004)*	*0.901 (0.002)*	*0.869 (0.004)*	*0.917 (0.005)*	*0.930 (0.002)*	***0.941 (0.001)***	*0.850 (0.009)*
ELECTRA	*0.876 (0.007)*	*0.873 (0.002)*	*0.838 (0.008)*	*0.757 (0.001)*	*0.853 (0.006)*	*0.858 (0.004)*	*0.837 (0.002)*	*0.883 (0.002)*	*0.910 (0.005)*	***0.920 (0.003)***	*0.860 (0.004)*
XLNET	*0.833 (0.001)*	*0.872 (0.004)*	*0.788 (0.001)*	*0.760 (0.002)*	*0.836 (0.002)*	*0.866 (0.006)*	*0.826 (0.002)*	*0.796 (0.070)*	*0.902 (0.004)*	***0.912 (0.006)***	*0.860 (0.003)*
LLaMA	*0.931 (0.000)*	*0.941 (0.003)*	*0.920 (0.008)*	*0.850 (0.002)*	*0.910 (0.002)*	*0.883 (0.003)*	*0.931 (0.000)*	*0.950 (0.003)*	*0.931 (0.002)*	***0.952 (0.003)***	*0.941 (0.002)*
Falcon	*0.921 (0.000)*	*0.935 (0.003)*	*0.921 (0.008)*	*0.841 (0.002)*	*0.920 (0.002)*	*0.881 (0.003)*	*0.930 (0.000)*	*0.936 (0.003)*	*0.922 (0.008)*	***0.950 (0.000)***	*0.948 (0.001)*

Table 6: Average F1-score and standard deviation obtained on the COVID dataset. Each row represents a feature representation technique, while each column represents a classification model. The last column showcases the performance of end-to-end transformer models. The best results for each feature representation are highlighted in bold. The cells marked with an (*) indicate that the classification models and feature representation techniques in the corresponding rows and columns are incompatible

feature representation techniques. The BiLSTM model, which leverages Falcon as a feature representation, achieved the highest result in the GM dataset (99.3%), thus suggesting that feature-based approaches demonstrate greater efficacy than fine-tuning approaches in GM datasets.

In Table 8, we rank the methods based on their performance, with the method achieving the best performance (based on the f1 score) ranked 1, the second-best ranked 2, and so on. We then calculate the average rank over all datasets, with a lower average rank indicating better performance. Overall, deep learning models typically have lower ranks than other models, particularly when transformer methods are used for feature extraction. Among feature extraction methods, the Falcon model has the lowest rank (best), while LLaMA and BART are close behind, and frequency-based models follow them with only a small difference. Figure 9, displays the average rank of various feature extraction methods and models across four datasets. These rankings are efficiently summarized and organized for clarity in bar chart.

To summarize, the results in Tables 4, 5, 6, and 7 indicate that deep learning models, particularly BiLSTM with context-dependent feature representation techniques, outperform

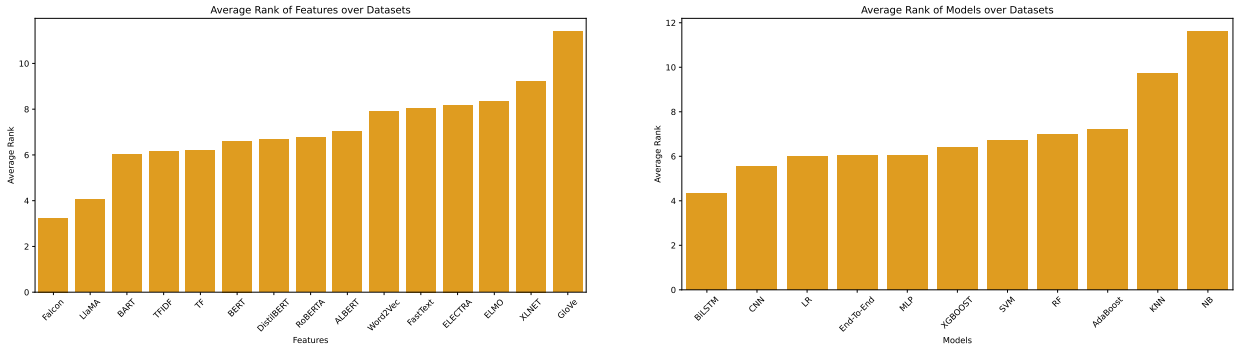
Features	SVM	LR	KNN	NB	RF	AdaBoost	XGBoost	MLP	CNN	BiLSTM	End-To-End
TF	*0.905 (0.009)*	*0.919 (0.007)*	*0.796 (0.011)*	*0.847 (0.019)*	*0.901 (0.006)*	***0.923 (0.007)***	*0.920 (0.006)*	*0.921 (0.006)*	*	*	*
TF-IDF	***0.939 (0.005)***	*0.936 (0.007)*	*0.594 (0.008)*	*0.847 (0.019)*	*0.906 (0.003)*	*0.924 (0.007)*	*0.924 (0.007)*	*0.937 (0.008)*	*	*	*
Word2Vec	***0.897 (0.007)***	*0.874 (0.003)*	*0.828 (0.014)*	*0.702 (0.009)*	*0.852 (0.013)*	*0.858 (0.008)*	*0.887 (0.005)*	*0.890 (0.007)*	*0.808 (0.000)*	*0.858 (0.000)*	*
GloVe	*0.834 (0.010)*	*0.787 (0.013)*	*0.777 (0.009)*	*0.662 (0.013)*	*0.812 (0.007)*	*0.795 (0.016)*	*0.824 (0.009)*	*0.817 (0.009)*	*0.819 (0.004)*	***0.859 (0.001)***	*
fastText	***0.885 (0.007)***	*0.871 (0.005)*	*0.795 (0.009)*	*0.712 (0.014)*	*0.854 (0.007)*	*0.866 (0.010)*	***0.885 (0.006)***	*0.878 (0.012)*	*0.838 (0.010)*	*0.842 (0.003)*	*
ELMO	*0.890 (0.007)*	*0.901 (0.006)*	*0.827 (0.012)*	*0.665 (0.014)*	*0.867 (0.005)*	*0.886 (0.004)*	*0.893 (0.005)*	*0.903 (0.006)*	*0.728 (0.016)*	*0.812 (0.008)*	***0.928 (0.005)***
BERT	*0.821 (0.009)*	*0.864 (0.001)*	*0.865 (0.013)*	*0.883 (0.009)*	*0.869 (0.009)*	*0.881 (0.005)*	*0.870 (0.011)*	*0.883 (0.012)*	*0.960 (0.002)*	***0.970 (0.001)***	*0.880 (0.006)*
DistilBERT	*0.791 (0.006)*	*0.857 (0.011)*	*0.853 (0.015)*	*0.868 (0.000)*	*0.866 (0.007)*	*0.873 (0.016)*	*0.856 (0.015)*	*0.872 (0.010)*	***0.960 (0.001)***	***0.960 (0.008)***	*0.890 (0.003)*
ALBERT	*0.784 (0.013)*	*0.880 (0.009)*	*0.864 (0.012)*	*0.869 (0.015)*	*0.887 (0.013)*	*0.887 (0.008)*	*0.877 (0.007)*	*0.888 (0.012)*	***0.950 (0.003)***	***0.950 (0.001)***	*0.870 (0.001)*
BART	*0.803 (0.011)*	*0.864 (0.001)*	*0.850 (0.012)*	*0.875 (0.020)*	*0.868 (0.008)*	*0.910 (0.007)*	*0.865 (0.000)*	*0.903 (0.005)*	*0.980 (0.005)*	***0.990 (0.001)***	*0.940 (0.001)*
RoBERTa	*0.718 (0.002)*	*0.899 (0.007)*	*0.853 (0.004)*	*0.888 (0.001)*	*0.861 (0.004)*	*0.883 (0.001)*	*0.869 (0.010)*	*0.897 (0.008)*	*0.970 (0.002)*	***0.980 (0.003)***	*0.850 (0.009)*
ELECTRA	*0.757 (0.000)*	*0.895 (0.006)*	*0.857 (0.016)*	*0.823 (0.004)*	*0.872 (0.012)*	*0.893 (0.010)*	*0.862 (0.011)*	*0.864 (0.009)*	*0.960 (0.004)*	***0.970 (0.002)***	*0.890 (0.001)*
XLNET	*0.719 (0.005)*	*0.868 (0.021)*	*0.783 (0.007)*	*0.783 (0.012)*	*0.842 (0.004)*	*0.881 (0.011)*	*0.838 (0.008)*	*0.830 (0.011)*	*0.940 (0.006)*	***0.960 (0.005)***	*0.890 (0.011)*
LLaMA	*0.936 (0.000)*	*0.961 (0.003)*	*0.910 (0.008)*	*0.774 (0.002)*	*0.914 (0.002)*	*0.915 (0.003)*	*0.932 (0.000)*	*0.964 (0.003)*	*0.981 (0.002)*	***0.992 (0.008)***	***0.992 (0.003)***
Falcon	*0.983 (0.000)*	*0.992 (0.003)*	*0.933 (0.008)*	*0.886 (0.002)*	*0.944 (0.002)*	*0.959 (0.003)*	*0.965 (0.000)*	*0.992 (0.003)*	*0.991 (0.006)*	***0.993 (0.001)***	***0.993 (0.000)***

Table 7: Average F1-score and standard deviation obtained on the GM dataset. Each row represents a feature representation technique, while each column represents a classification model. The last column showcases the performance of end-to-end transformer models. The best results for each feature representation are highlighted in bold. The cells marked with an (*) indicate that the classification models and feature representation techniques in the corresponding rows and columns are incompatible

other models in binary-class datasets such as ISOT, COVID, and GM. In the LIAR dataset, which is a multi-class dataset, classical machine learning models, particularly SVM, have the best performance across multiple feature representations. It is important to note that, compared to the other datasets in this study, the multi-class LIAR dataset acts differently. This completely different behavior compared to other datasets can be explained by the class ambiguity and overlapping class predictions. Additionally, the results collected from multiple datasets suggest that using transformer models as feature representation techniques produces higher performance than fine-tuning them. This is an important finding since it opens the doors for obtaining highly diverse and complementary ensemble models by training diverse classification algorithms over these feature representations.

Features	SVM	LR	KNN	NB	RF	AdaBoost	XGBoost	MLP	CNN	BiLSTM	End-To-End	Total
TF	*4.0 (3.559)*	*4.0 (2.944)*	*15.0 (6.377)*	*6.5 (5.686)*	*5.0 (3.559)*	*4.75 (2.872)*	*5.5 (3.109)*	*4.75 (2.872)*	*	*	*	***6.2 (4.993)***
TF-IDF	*3.25 (2.062)*	*3.75 (2.062)*	*15.25 (11.955)*	*6.5 (5.686)*	*5.25 (2.986)*	*5.25 (2.986)*	*5.5 (3.109)*	*4.5 (1.915)*	*	*	*	***6.2 (5.826)***
Word2Vec	*4.75 (3.594)*	*6.5 (4.726)*	*9.5 (5.066)*	*15.75 (9.179)*	*7.75 (5.252)*	*6.0 (3.559)*	*7.75 (4.5)*	*6.0 (3.464)*	*8.25 (7.274)*	*6.75 (5.123)*	*	***7.9 (5.615)***
GloVe	*10.0 (5.354)*	*13.0 (7.071)*	*13.25 (6.344)*	*18.0 (10.296)*	*10.5 (6.245)*	*10.5 (5.568)*	*12.75 (6.602)*	*10.75 (5.737)*	*9.0 (6.218)*	*6.25 (5.5)*	*	***11.4 (6.547)***
FastText	*5.5 (3.786)*	*6.75 (4.5)*	*11.25 (6.702)*	*15.0 (8.756)*	*8.0 (4.967)*	*6.0 (3.559)*	*7.0 (4.32)*	*6.25 (4.193)*	*7.75 (5.737)*	*7.0 (6.218)*	*	***8.05 (5.57)***
ELMO	*4.75 (4.272)*	*4.75 (3.775)*	*9.0 (5.416)*	*17.0 (9.416)*	*7.0 (4.546)*	*5.5 (3.786)*	*6.25 (3.403)*	*5.25 (3.403)*	*15.5 (7.853)*	*12.0 (5.715)*	*4.75 (2.63)*	***8.4 (6.361)***
BERT	*7.0 (7.439)*	*6.5 (5.447)*	*7.25 (3.948)*	*9.25 (2.986)*	*6.75 (4.272)*	*8.0 (3.916)*	*6.25 (4.031)*	*6.25 (4.349)*	*4.0 (3.559)*	*3.25 (2.63)*	*8.25 (2.63)*	***6.6 (4.138)***
DistilBERT	*8.0 (8.718)*	*7.25 (4.992)*	*8.25 (4.573)*	*9.5 (2.887)*	*6.5 (4.509)*	*8.25 (4.349)*	*6.5 (4.509)*	*6.25 (4.787)*	*4.0 (3.559)*	*3.75 (2.5)*	*5.25 (4.272)*	***6.7 (4.528)***
ALBERT	*8.75 (9.032)*	*6.5 (4.203)*	*8.25 (4.193)*	*10.0 (4.082)*	*7.0 (3.162)*	*7.75 (4.031)*	*6.5 (3.416)*	*6.25 (3.686)*	*3.75 (1.5)*	*3.25 (1.708)*	*9.25 (4.992)*	***7.02 (4.38)***
BART	*7.0 (8.718)*	*6.5 (5.26)*	*7.5 (5.066)*	*9.0 (2.944)*	*6.75 (4.5)*	*7.5 (4.203)*	*5.25 (2.872)*	*6.0 (3.367)*	*3.25 (2.63)*	*2.0 (2.0)*	*5.5 (3.109)*	***6.02 (4.332)***
RoBERTA	*10.5 (10.504)*	*5.5 (3.416)*	*8.5 (4.655)*	*9.25 (3.096)*	*7.25 (4.787)*	*7.75 (4.272)*	*6.25 (4.031)*	*5.5 (3.416)*	*2.5 (1.0)*	*2.0 (0.816)*	*9.25 (5.439)*	***6.75 (4.966)***
ELECTRA	*10.0 (9.522)*	*7.0 (3.162)*	*10.0 (3.651)*	*14.25 (5.679)*	*8.5 (4.123)*	*9.25 (4.425)*	*7.5 (3.512)*	*8.0 (4.546)*	*4.25 (1.708)*	*3.5 (1.291)*	*7.75 (4.573)*	***8.2 (4.966)***
XLNET	*12.5 (9.678)*	*7.5 (4.655)*	*13.75 (6.551)*	*13.5 (7.724)*	*9.5 (5.447)*	*9.75 (5.679)*	*7.25 (3.948)*	*10.5 (6.608)*	*5.5 (1.732)*	*4.25 (1.708)*	*7.5 (4.041)*	***9.2 (5.874)***
LLaMA	*2.75 (2.363)*	*2.5 (1.291)*	*5.0 (2.944)*	*11.5 (8.021)*	*4.75 (3.096)*	*6.75 (4.856)*	*3.75 (2.5)*	*2.5 (1.732)*	*2.25 (0.957)*	*1.25 (0.5)*	*1.5 (0.577)*	***4.1 (4.109)***
Falcon	*2.25 (1.258)*	*1.75 (0.957)*	*4.5 (2.082)*	*9.25 (2.754)*	*4.0 (1.633)*	*5.0 (2.582)*	*2.25 (0.957)*	*2.0 (1.414)*	*2.25 (1.5)*	*1.25 (0.5)*	*1.25 (0.5)*	***3.25 (2.695)***
Total	***6.7(6.643)***	***5.98 (4.478)***	***9.75 (5.979)***	***11.62 (6.735)***	***6.97 (4.174)***	***7.2 (4.008)***	***6.42 (4.014)***	***6.05 (4.16)***	***5.56 (5.181)***	***4.35 (4.191)***	***6.025 (4.264)***	***7.05 (5.357)***

Table 8: Average rank of methods based on different datasets. Each row represents a feature representation technique, while each column represents a classification model. The ranks are reported based on the average ranks and standard deviation over 4 different datasets. *Total* indicate average ranks of pair (features, classifiers) which are highlighted in bold. The cells marked with an (*) indicate that the classification models and feature representation techniques in the corresponding rows and columns are incompatible



(a) Features Average Rank

(b) Models Average Rank

Figure 9: Average rank of methods based on different datasets. Graph (a) plots the average rank of different feature extraction methods over four dataset. Graph (b) plots the average rank of different classification models over four datasets. The bar charts have been sorted from smaller ranks to larger ranks.

5.2. ***Error Analysis***

In this section, we answer ***(RQ3)*** *Would combining feature extraction methods improve performance in the fake news detection task?* We conducted an error analysis based on the results to answer this question. To be more specific, Tables 9, 10, 12 and 11 show the error distribution for all feature extraction methods based on three selective classifiers from

different families. In each row of the tables, the minimum number of instances that are miss-classified per number of feature extraction methods and specific classifiers is shown. The first column of the table shows the minimum number of miss-classified instances based on one feature extraction method (any method), column two shows the minimum number of miss-classified instances based on two methods (any two), and so on. For deep learning models, because we just use 13 feature extraction methods (TF and TF-IDF have been excluded), columns 14 and 15 are marked with an (*) sign. In this study, the classifiers that are less correlated to each other have been selected. We determine the correlation between classifiers by using Spearman's rank correlation coefficient, which evaluates the monotonic connection between variables based on their rankings rather than their actual values. We assess the level of correlation between classifiers by contrasting their predictions for a given collection of examples. Less similarity between predictions is shown by lower correlation values. The combination of feature extraction techniques and classifiers may be enhanced for better performance in fake news detection tasks by using fewer correlated classifiers, enabling us to pick models that give various independent insights.

For the error analysis in the Liar dataset, we select SVM, AdaBoost, and BiLSTM because their prediction results are less correlated to each other. The error distribution is presented in Table 9. As a result, except in BiLSTM, we can see that using different feature extraction methods can not decrease the misclassification rate, which indicates that the majority of the errors that occur in this dataset are commonly made by the classifiers trained over distinct feature representations. In particular, $1,045$ and 749 instances from this dataset are misclassified by all 15 feature methods in SVM and Adaboost classifiers, each using a different feature representation as input, respectively. The multi-class classification problem and the proximity of the classes of this dataset to each other have made the classification more difficult. This analysis can also indicate problems with ambiguity in modeling this dataset directly as a six-class classification problem. However, when BiLSTM is considered as the learning algorithm, it is evident that various feature extraction techniques have greater discriminating abilities. Additionally, it is important to note that no instance was incorrectly predicted by this model trained all of the feature sets. Hence, even in this difficult prediction

case, in which the data and classes are ambiguous, it is theoretically possible to obtain a 100% classification accuracy with a perfect fusion strategy.

Number of feature methods which made the same error	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**Total**
No.errors in SVM	*105*	*95*	*89*	*100*	*110*	*94*	*108*	*113*	*135*	*148*	*140*	*198*	*283*	*400*	*1045*	*3154*
No.errors in AdaBoost	*31*	*20*	*28*	*30*	*63*	*90*	*115*	*159*	*170*	*207*	*250*	*310*	*400*	*619*	*749*	*3206*
No.errors in BiLSTM	*0*	*0*	*12*	*41*	*173*	*387*	*698*	*939*	*715*	*246*	*0*	*0*	*0*	*	*	*3204*

Table 9: Error distribution across feature extraction methods based on three less correlated classifiers from different families over LIAR dataset. Each row represents the minimum number of instances that are miss-classified based on a specific classifier, while each column represents the number of feature extraction methods. columns 14 and 15 are marked with an (*) sign for deep learning models as only 13 feature extraction methods are employed (excluding TF and TF-IDF due to incompatibility).

Table 10 presents the error distribution for the ISOT dataset. For analyzing dataset logistic regression, XGBoost, and MLP have been selected based on Spearman's rank correlation coefficient. The results from this dataset present a completely different behavior compared to the LIAR one and strongly show that the majority of errors made by classifiers trained over distinct feature representations are made on different texts. Considering the MLP, LR, and XGBoost classifiers, no instances are systematically misclassified by models trained over five, six, and seven feature representations, respectively. This error analysis demonstrates that combining models trained over different feature representation techniques has a huge potential to increase fake news detection performance significantly. To be more specific, a perfect fusion of seven feature extraction techniques would yield a 100% recognition rate for this dataset.

The GM dataset, the SVM, AdaBoost, and BiLSTM have been selected as the less correlated models. We can observe the same phenomenon when analyzing the error distribution conducted for the GM dataset (Table 11). We observe that the majority of misclassified news is unique to a single model (i.e., one classifier and feature representation pair). The intersection of errors significantly decreases as more feature representation techniques are considered, confirming the hypothesis they capture distinct and complementary information from the input text. The results also highlight the crucial impact of using a perfect fusion

Number of feature methods which made the same error	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	Total
No.errors in LR	*1875*	*319*	*94*	*26*	*2*	*0*	*0*	*0*	*0*	*0*	*0*	*0*	*0*	*0*	*0*	*2318*
No.errors in XGBoost	*2031*	*314*	*74*	*39*	*1*	*1*	*0*	*0*	*0*	*0*	*0*	*0*	*0*	*0*	*0*	*2461*
No.errors in MLP	*1798*	*212*	*52*	*21*	*0*	*0*	*0*	*0*	*0*	*0*	*0*	*0*	*0*	*0*	*0*	*2083*

Table 10: Error distribution across feature extraction methods based on three less correlated classifiers from different families over the ISOT dataset. Each row represents the minimum number of instances that are miss-classified based on a specific classifier, while each column represents the number of feature extraction methods.

of multiple feature extraction methods on the discriminative power of selected classifiers in the GM dataset. To be more specific, in the SVM and AdaBoost model, a perfect fusion of at least nine feature extraction achieve a 100% recognition rate for the fake news detection task. Similarly, in the BiLSTM model, a perfect fusion of at least six feature extraction methods can achieve a 100% recognition rate for this dataset.

Number of feature methods which made the same error	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	Total
No.errors in SVM	*56*	*404*	*281*	*156*	*64*	*21*	*4*	*3*	*0*	*0*	*0*	*0*	*0*	*0*	*0*	*1389*
No.errors in AdaBoost	*545*	*323*	*166*	*103*	*34*	*7*	*1*	*1*	*0*	*0*	*0*	*0*	*0*	*0*	*0*	*1180*
No.errors in BiLSTM	*375*	*169*	*89*	*16*	*1*	*0*	*0*	*0*	*0*	*0*	*0*	*0*	*0*	*	*	*650*

Table 11: Error distribution across feature extraction methods based on three less correlated classifiers from different families over GM dataset. Each row represents the minimum number of instances that are miss-classified based on a specific classifier, while each column represents the number of feature extraction methods. columns 14 and 15 are marked with an (*) sign for deep learning models as only 13 feature extraction methods are employed (excluding TF and TF-IDF due to incompatibility).

Lastly, the SVM, XGBoost, and BiLSTM were the less correlated models according to the results obtained over the COVID dataset and were selected for analyzing the misclassification behavior made by classifiers trained over different feature representations. According to Table 12, we can see that for the SVM and XGBoost, the majority of misclassified instances are commonly misclassified by models trained over 7 out of 15 feature representations. How-

ever, this number significantly decreases as more techniques are added, with no single input being misclassified in any combination of 12 representations. Hence, it is theoretically possible to obtain a 100% performance for this dataset using those classification models with an ideal combination scheme. Another interesting observation is regarding the BiLSTM model, which is the one that presents the biggest error reduction as the number of feature extractions increases. With a total of 7 different feature representations, the error distribution is already zero.

Number of feature methods which made the same error	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**Total**
No.errors in SVM	*214*	*105*	*93*	*77*	*74*	*95*	*946*	*120*	*57*	*39*	*35*	*0*	*0*	*0*	*0*	*1755*
No.errors in XGBoost	*224*	*140*	*100*	*107*	*84*	*157*	*773*	*117*	*58*	*36*	*34*	*0*	*0*	*0*	*0*	*1830*
No.errors in BiLSTM	*850*	*259*	*122*	*53*	*9*	*2*	*0*	*0*	*0*	*0*	*0*	*0*	*0*	*	*	*1295*

Table 12: Error distribution across feature extraction methods based on three less correlated classifiers from different families over the COVID dataset. Each row represents the minimum number of instances that are miss-classified based on a specific classifier, while each column represents the number of feature extraction methods. columns 14 and 15 are marked with an (*) sign for deep learning models as only 13 feature extraction methods are employed (excluding TF and TF-IDF due to incompatibility).

As a conclusion to this section, we address RQ3. Tables 9, 10, 12 and 11 demonstrate that different feature representation methods have a strong influence on the performance of fake news detection models. These results demonstrate the complementarity of the 15 feature sets and how specific feature extraction techniques are better suited for certain news. Therefore, using an ensemble of these methods can significantly improve the results, particularly in the ISOT, COVID, and GM datasets. The next challenge is to identify the optimal combination scheme for this task. In contrast, in the LIAR dataset, the results illustrate that combination of different feature extraction methods does not help to decrease the misclassification rate in this dataset. Since in the Liar dataset, classes are systematically close to each other. So it forces the models to have the same mistakes and, as a result, makes the classification more difficult. Another important point is that this analysis demonstrates the BiLSTM is the model with the greatest potential for building diverse and complementary

ensemble models while changing the input representation. We observed the highest drop in terms of the intersection of errors as new feature sets were added for all datasets considered in this study.

5.3. ***Models cost-effectiveness evaluation***

In this Section, we answer ***(RQ4)*** *Which methods are more cost-effective?* Figure 10 illustrates the average performance of each model based on different feature extraction methods against their average time. In this analysis, we calculate the training time without considering the representation step to make the models comparable. The dashed lines in the figures show the models' average performance and average training time. The models in the first quadrant of the figures (top-left) are among the models with the most cost-effective models in this study. And the models that are in the second quadrant of the figures have good performance, but they have a high training cost. Also, the models in the third quadrant of the chart are models that are not optimal in terms of performance or training cost. And finally, the models in the fourth quarter of the figures have relatively lower performance than the rest of the models, but the cost of training these models is less expensive.

In summary, despite deep models' remarkable performance, most have a high computational cost. In contrast, the plots show that the ensemble models like AdaBoost, XGBoost, and Random Forest are the most cost-effective in this study. Among the classical machine learning models, SVM and Logistic regression are more effective in comparison to the K-nearest neighbors, Naive Bayes. Also, in some datasets like LIAR and COVID, we can see that the End-to-End models are too expensive with lower performance. In fact, Figure 10 shows that the behavior of the classifiers was stable for different datasets and for the average of the pairs (features, classifiers).

6. Conclusion

This paper presented an updated taxonomy of textual-based fake news detection. The critical element of textual-based fake news detection is explained. First, we describe fake news characterization and its different perspectives, such as content-based and social context

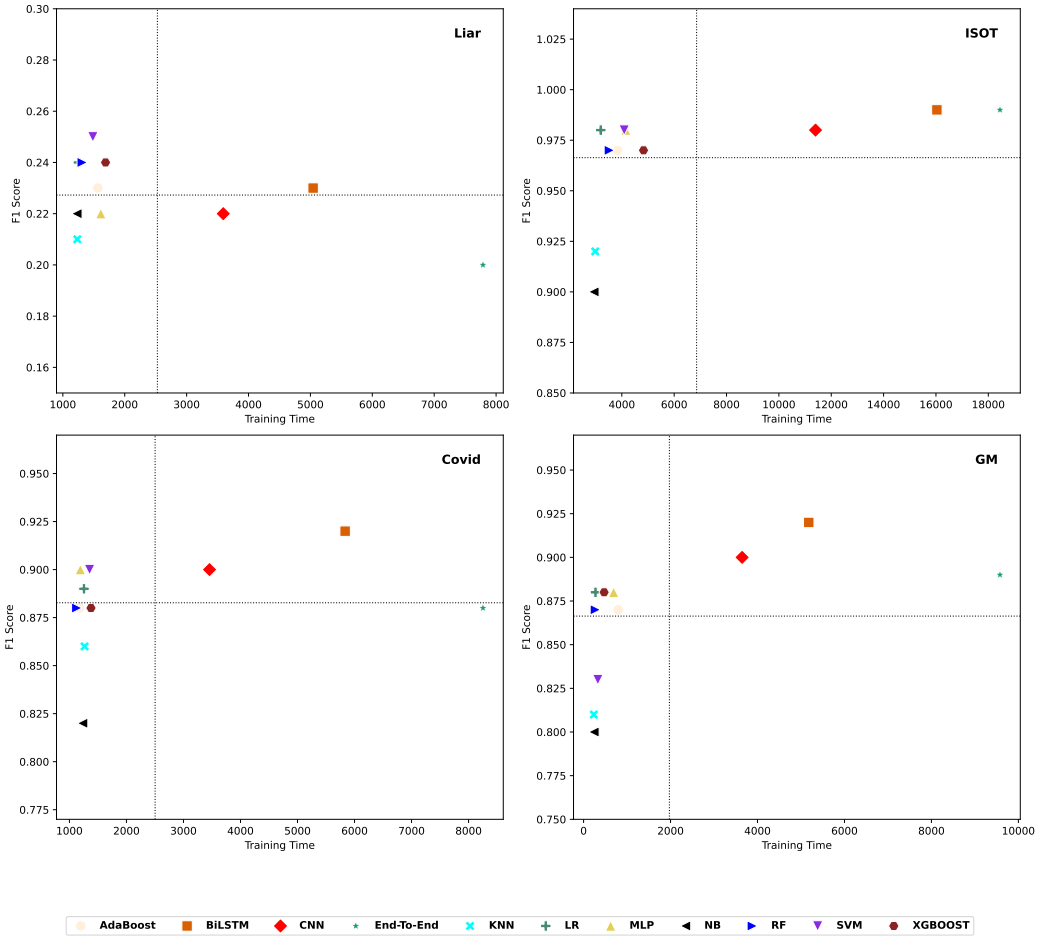


Figure 10: Models cost-effective visualization. This graph plots the average performance of models (F1-score) versus the average training time (second) of models. different colorful shapes show the models, and the dashed lines in the plots show the models' average performance and average training time in each dataset.

ones. Second, we analyze the state-of-the-art automatic fake news detection techniques according to multiple criteria. The review is used to propose an updated taxonomy for the field according to the main steps in building an automatic fake news detection system: preprocessing, feature extraction, and classification algorithms. We also review the main feature extraction and classification models, discuss their usage for fake news classification, and present their advantages and limitations.

An in-depth comparative study was carried out to analyze the main feature extraction techniques and multiple classification algorithms over a diverse set of fake news detection datasets. The study considers 15 feature extraction techniques coming from different families, including count-based ones, word embeddings, and context-based ones, as well as 20 classification algorithms from single models, ensemble learning techniques, and deep learning ones. Experimental results demonstrate that better classification performance is achieved when transformer models such as Falcon, LLaMA and BART are used as feature extractors instead of an end-to-end model fine-tuned to the downstream task. This is an interesting finding since it also allows the building of diverse multiple-classifier systems by training distinct classification models with complementary classification power over their features.

Subsequently, we investigated the correlation between classifiers trained using different feature representations and found that, in most cases, the misclassification errors made by the same classification algorithm using different feature representations were dissimilar. The analysis demonstrates that, in general, using between five to six distinct feature representation models while fixing the classification algorithm (e.g., SVM), it is possible to obtain a theoretical performance of 100% over several datasets as no news is commonly misclassified by models trained over a variety of feature spaces. These findings indicate the potential for improving fake news detection performance by investigating models trained over multiple feature representations, taking advantage of their complementary representation power to build an effective set of classifiers to significantly improve fake news detection accuracy. This strategy is further supported by the computational cost analysis, which suggests that certain classical techniques should be deemed worthy of consideration, as they provide a beneficial balance between accuracy and computational cost and have great potential for

forming strong ensemble models when trained over distinct feature representations.

7. Challenges and Future Directions

Researchers face complex difficulties as a result of the growing incidence of fake news in modern media. It is challenging to distinguish fake news because of its complexity and dynamic nature. Therefore, fake news detection methods face challenges and limitations. One of the big challenges in fake news detection methods is content variability. In other words, fake news comes in a wide range of content formats, including text, photos, videos, and voice by its very nature. So this variation necessitates flexible detection methods. Rapid dissemination is another big challenge in detecting fake news. So the need for quick identification is obvious given how quickly fake news circulates, particularly on social media platforms. Additionally, creators of fake news who now use AI technologies are innovative and adaptable, which adds another level of difficulty. The difficulty is further exacerbated by the lack of sufficient training data, the complexity of context sensitivity in identifying fake news, language, and cultural diversity, and the biases built into algorithms. Also, the challenge of multiclass fake news detection persists, especially when classes overlap. These difficulties not only highlight the complexity of the issue but also act as a foundation for further study. Although there are many studies related to fake news detection, this task still has a long way to go. Therefore, we present some perspectives to pave the way for other researchers in this field.

7.1. ***Fake news detection using multiple feature representations***

While significant progress has been made in utilizing individual feature representations, including large language models such as BERT, the results obtained in this paper demonstrate that leveraging the power of multiple feature representations holds great promise for enhancing the performance and effectiveness of detection systems. Context Sensitivity can be addressed by combining numerous feature representations, which keeps the system aware of various contexts and captures subtleties that a single representation could miss. This is

crucial when taking into account cultural and linguistic diversity because it's possible that a single viewpoint doesn't capture the whole picture.

According to the experimental study conducted in this article, we show that even when considering the same classification algorithm (e.g., Adaboost or SVM), when they are built using diverse feature representations, it is theoretically possible to significantly reduce the misclassification error using a suitable combination system as the majority of texts were just misclassified by a fraction of the techniques. Therefore, they complement each other by capturing different characteristics from the input text. Therefore, more effort into approaches for combining the complementary information from multiple feature representations should be made.

In addition, nowadays fake news publishers use AI-generating tools to publish fake news on social media. Therefore combining the complementary information from multiple feature representations can address this challenge. Moreover, it is well known that systems built with highly diverse models can better handle changes in distributions that may occur between training and test time [167, 168] as well as avoiding biases [169]. As mining fake news from social media is inherently a dynamic problem, in which the nature of news can change accordingly to major events happening in the world, it may be severely affected by distribution shifts and concept drift. As such, having a system considering multiple feature representations and classification algorithms trained over it can be an interesting avenue for building systems that can better handle the evolving nature of this problem in the real world and avoid algorithmic bias.

7.2. ***Fake news detection using multiple perspectives***

The integration of multiple detection perspectives holds immense potential for enhancing the accuracy and reliability of detection systems. While significant progress has been made in leveraging linguistic features for fake news identification, there is still a need to explore the inclusion of social context-based approaches such as news propagation and network-based analysis. In other words, due to the difficulty of Content Variability, which highlights the variety of fake news media, the fusion of different content formats becomes essential.

Hence, one promising avenue for future research lies in the fusion of linguistic and social context features. Furthermore, such an integrated approach would allow us to capture the correlations between linguistic cues and social network dynamics, which can provide a robust framework for detection as well as interpretability of how certain linguistic features are associated with different network propagation patterns. Besides text and network features, one could also consider information from different modalities, such as images and/or videos, which could further complement the analysis.

Nevertheless, in order to build such a system, the design and implementation of large-scale datasets that are representative of different types of fake news (e.g., politics, health, etc.) encompass diverse linguistic and social context dimensions are needed for training and evaluating these detection models taking multiple sources of information.

7.3. ***Dynamic ensemble models for fake news detection***

Although there are several techniques using ensemble models for fake news detection, the techniques proposed are based on static ensemble techniques such as Random Forests and XGBoost and also do not explicitly takes into consideration any information about the complementarity between the models and multiple feature representations in their conception.

To the best of our knowledge, dynamic ensemble models [170], which are a class of ensemble models that changes their topology on the fly according to each new input sample presented to the system, have yet to be explored in this context. Several works have demonstrated the benefits of such techniques in many problems, including ones associated with high overlap between classes. Furthermore, exploring dynamic ensemble selection methods in this context can be an interesting alternative for combining models trained over multiple feature representations as well as for handling more complex systems that take into account different perspectives for fake news detection since these methods can analyze the input data coming from different perspectives and select just the one(s) that are more reliable for classification.

*7.4. **Evaluation of fake news detection models using cross-dataset***

Given the dynamic nature of fake news detection from social media, news characteristics may change in response to significant events taking place around the world. Therefore, the task of detecting fake news may run into difficulties as a result of distribution changes and concept drift. Notably, most of the available datasets contain specific news domains (e.g., politics, health, etc.). To address this problem, cross-dataset analysis is essential for determining dataset biases and might offer insightful ideas for procedures with improved generalization power [171]. To the best of our knowledge, cross-dataset evaluation research is mainly unexplored, especially in the field of fake news detection. It must be considered in future approaches to evaluate the generalization ability and robustness of proposed fake news detection approaches to be deployed in real-world systems.

Acknowledgements

The authors would like to thank NSERC (Natural Sciences and Engineering Research Council of Canada), discovery grant program (RGPIN-2021-04130), and the Brazilian agencies CAPES (Coordenação de Aperfeiçoamento de Pessoal de Nível Superior), CNPq (Conselho Nacional de Desenvolvimento Científico e Tecnológico).

References

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

[2] N. Diakopoulos, M. De Choudhury, M. Naaman, Finding and assessing social media information sources in the context of journalism, in: Proceedings of the SIGCHI conference on human factors in computing systems, 2012, pp. 2451–2460.

[3] A. Hermida, Twittering the news: The emergence of ambient journalism, Journalism practice 4 (3) (2010) 297–308.

[4] P. Tolmie, R. Procter, D. W. Randall, M. Rouncefield, C. Burger, G. Wong Sak Hoi, A. Zubiaga, M. Liakata, Supporting the use of user generated content in journalistic practice, in: Proceedings of the 2017 chi conference on human factors in computing systems, 2017, pp. 3632–3644.

[5] A. Zubiaga, A. Aker, K. Bontcheva, M. Liakata, R. Procter, Detection and resolution of rumours in social media: A survey, ACM Computing Surveys (CSUR) 51 (2) (2018) 1–36.

[6] A. Bondielli, F. Marcelloni, A survey on fake news and rumour detection techniques, Information Sciences 497 (2019) 38–55.

[7] J. C. Reis, A. Correia, F. Murai, A. Veloso, F. Benevenuto, Supervised learning for fake news detection, IEEE Intelligent Systems 34 (2) (2019) 76–81.

[8] X. Zhou, R. Zafarani, A survey of fake news: Fundamental theories, detection methods, and opportunities, ACM Computing Surveys (CSUR) 53 (5) (2020) 1–40.

[9] A. Bovet, H. A. Makse, Influence of fake news in twitter during the 2016 us presidential election, Nature communications 10 (1) (2019) 1–14.

[10] N. Grinberg, K. Joseph, L. Friedland, B. Swire-Thompson, D. Lazer, Fake news on twitter during the 2016 us presidential election, Science 363 (6425) (2019) 374–378.

[11] S. van Der Linden, J. Roozenbeek, J. Compton, Inoculating against fake news about covid-19, Frontiers in psychology 11 (2020) 2928.

[12] C. M. Greene, R. A. Nash, G. Murphy, Misremembering brexit: Partisan bias and individual predictors of false memories for fake news stories among brexit voters, Memory (2021) 1–18.

[13] F. K. A. Salem, R. Al Feel, S. Elbassuoni, M. Jaber, M. Farah, Fa-kes: A fake news dataset around the syrian war, in: Proceedings of the international AAAI conference on web and social media, Vol. 13, 2019, pp. 573–582.

[14] Y. Shin, Y. Sojdehei, L. Zheng, B. Blanchard, Content-based unsupervised fake news detection on ukraine-russia war, SMU Data Science Review 7 (1) 3.

[15] V. L. Rubin, On deception and deception detection: Content analysis of computer-mediated stated beliefs, Proceedings of the American Society for Information Science and Technology 47 (1) (2010) 1–10.

[16] X. Zhang, A. A. Ghorbani, An overview of online fake news: Characterization, detection, and discussion, Information Processing & Management 57 (2) (2020) 102025.

[17] S. Gilda, Notice of violation of ieee publication principles: Evaluating machine learning algorithms for fake news detection, in: 2017 IEEE 15th student conference on research and development (SCOReD), IEEE, 2017, pp. 110–115.

[18] R. Oshikawa, J. Qian, W. Y. Wang, A survey on natural language processing for fake news detection, arXiv preprint arXiv:1811.00770 (2018).

[19] G. Gravanis, A. Vakali, K. Diamantaras, P. Karadais, Behind the cues: A benchmarking study for fake news detection, Expert Systems with Applications 128 (2019) 201–213.

[20] J. Y. Khan, M. T. I. Khondaker, S. Afroz, G. Uddin, A. Iqbal, A benchmark study of machine learning

models for online fake news detection, Machine Learning with Applications 4 (2021) 100032.

[21] W. Y. Wang, " liar, liar pants on fire": A new benchmark dataset for fake news detection, arXiv preprint arXiv:1705.00648 (2017).

[22] H. Allcott, M. Gentzkow, Social media and fake news in the 2016 election, Journal of economic perspectives 31 (2) (2017) 211–36.

[23] Misinformation, accessed: 2023-09-22.
URL https://www.dictionary.com/browse/misinformation

[24] Disinformation, accessed: 2023-09-22.
URL https://www.dictionary.com/browse/disinformation

[25] K. Shu, H. Liu, J. Han, L. Getoor, W. Wang, J. Gehrke, R. Grossman, Detecting Fake News on Social Media, Synthesis Lectures on Data Mining and Knowledge Discovery, Morgan & Claypool Publishers, 2019.
URL https://books.google.ca/books?id=y7GhDwAAQBAJ

[26] C. Raj, P. Meel, Convnet frameworks for multi-modal fake news detection, Applied Intelligence 51 (11) (2021) 8132–8148.

[27] S. Qian, J. Wang, J. Hu, Q. Fang, C. Xu, Hierarchical multi-modal contextual attention network for fake news detection, in: Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, 2021, pp. 153–162.

[28] P. Li, X. Sun, H. Yu, Y. Tian, F. Yao, G. Xu, Entity-oriented multi-modal alignment and fusion network for fake news detection, IEEE Transactions on Multimedia (2021).

[29] Y. Fung, C. Thomas, R. G. Reddy, S. Polisetty, H. Ji, S.-F. Chang, K. McKeown, M. Bansal, A. Sil, Infosurgeon: Cross-media fine-grained information consistency checking for fake news detection, in: Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), 2021, pp. 1683–1698.

[30] N. M. D. Tuan, P. Q. N. Minh, Multimodal fusion with bert and attention mechanism for fake news detection, in: 2021 RIVF International Conference on Computing and Communication Technologies (RIVF), IEEE, 2021, pp. 1–6.

[31] D. K. Sharma, S. Garg, Ifnd: a benchmark dataset for fake news detection, Complex & Intelligent Systems (2021) 1–21.

[32] C. Song, N. Ning, Y. Zhang, B. Wu, Knowledge augmented transformer for adversarial multidomain multiclassification multimodal fake news detection, Neurocomputing 462 (2021) 88–100.

[33] Y. Wang, F. Ma, H. Wang, K. Jha, J. Gao, Multimodal emergent fake news detection via meta neural process networks, in: Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery &

Data Mining, 2021, pp. 3708–3716.

[34] Y. Wu, P. Zhan, Y. Zhang, L. Wang, Z. Xu, Multimodal fusion with co-attention networks for fake news detection, in: Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021, 2021, pp. 2560–2569.

[35] X. Zhou, A. Jain, V. V. Phoha, R. Zafarani, Fake news early detection: An interdisciplinary study, arXiv preprint arXiv:1904.11679 (2019).

[36] S. S. Birunda, R. K. Devi, A novel score-based multi-source fake news detection using gradient boosting algorithm, in: 2021 International Conference on Artificial Intelligence and Smart Systems (ICAIS), IEEE, 2021, pp. 406–414.

[37] K. Shu, D. Mahudeswaran, H. Liu, Fakenewstracker: a tool for fake news collection, detection, and visualization, Computational and Mathematical Organization Theory 25 (1) (2019) 60–71.

[38] M. N. Nikiforos, S. Vergis, A. Stylidou, N. Augoustis, K. L. Kermanidis, M. Maragoudakis, Fake news detection regarding the hong kong events from tweets, in: IFIP international Conference on Artificial Intelligence Applications and Innovations, Springer, 2020, pp. 177–186.

[39] J. Zhang, B. Dong, S. Y. Philip, Deep diffusive neural network based fake news detection from heterogeneous social networks, in: 2019 IEEE International Conference on Big Data (Big Data), IEEE, 2019, pp. 1259–1266.

[40] A. Silva, L. Luo, S. Karunasekera, C. Leckie, Embracing domain differences in fake news: Cross-domain fake news detection using multi-modal data, in: Proceedings of the AAAI conference on artificial intelligence, Vol. 35, 2021, pp. 557–565.

[41] D. M. Nguyen, T. H. Do, R. Calderbank, N. Deligiannis, Fake news detection using deep markov random fields, in: Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), 2019, pp. 1391–1400.

[42] M. Dong, L. Yao, X. Wang, B. Benatallah, Q. Z. Sheng, H. Huang, Dual: A deep unified attention model with latent relation representations for fake news detection, in: International conference on web information systems engineering, Springer, 2018, pp. 199–209.

[43] M. Smith, A. Richardson, B. Brown, G. Dozier, M. King, J. Morris, A study of the impact of evolutionary-based feature selection for fake news detection, in: 2020 IEEE Symposium Series on Computational Intelligence (SSCI), IEEE, 2020, pp. 1859–1865.

[44] H. Ahmed, I. Traore, S. Saad, Detection of online fake news using n-gram analysis and machine learning techniques, in: International conference on intelligent, secure, and dependable systems in distributed and cloud environments, Springer, 2017, pp. 127–138.

[45] G. E. R. Agudelo, O. J. S. Parra, J. B. Velandia, Raising a model for fake news detection using

machine learning in python, in: Conference on e-Business, e-Services and e-Society, Springer, 2018, pp. 596–604.

[46] V. Agarwal, H. P. Sultana, S. Malhotra, A. Sarkar, Analysis of classifiers for fake news detection, Procedia Computer Science 165 (2019) 377–383.

[47] K. Poddar, K. Umadevi, et al., Comparison of various machine learning models for accurate detection of fake news, in: 2019 Innovations in Power and Advanced Computing Technologies (i-PACT), Vol. 1, IEEE, 2019, pp. 1–5.

[48] P. Ksieniewicz, M. Choraś, R. Kozik, M. Woźniak, Machine learning methods for fake news classification, in: International Conference on Intelligent Data Engineering and Automated Learning, Springer, 2019, pp. 332–339.

[49] A. Kumar, S. Saumya, J. P. Singh, Nitp-ai-nlp@ urdufake-fire2020: Multi-layer dense neural network for fake news detection in urdu news articles., in: FIRE (Working Notes), 2020, pp. 458–463.

[50] A. Agarwal, A. Dixit, Fake news detection: an ensemble learning approach, in: 2020 4th International Conference on Intelligent Computing and Control Systems (ICICCS), IEEE, 2020, pp. 1178–1183.

[51] H. Reddy, N. Raj, M. Gala, A. Basava, Text-mining-based fake news detection using ensemble methods, International Journal of Automation and Computing 17 (2) (2020) 210–221.

[52] N. Smitha, R. Bharath, Performance comparison of machine learning classifiers for fake news detection, in: 2020 Second International Conference on Inventive Research in Computing Applications (ICIRCA), IEEE, 2020, pp. 696–700.

[53] H. Ahmed, I. Traore, S. Saad, Detecting opinion spams and fake news using text classification, Security and Privacy 1 (1) (2018) e9.

[54] B. Al Asaad, M. Erascu, A tool for fake news detection, in: 2018 20th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing (SYNASC), IEEE, 2018, pp. 379–386.

[55] C. M. M. Kotteti, X. Dong, N. Li, L. Qian, Fake news detection enhancement with data imputation, in: 2018 IEEE 16th Intl Conf on Dependable, Autonomic and Secure Computing, 16th Intl Conf on Pervasive Intelligence and Computing, 4th Intl Conf on Big Data Intelligence and Computing and Cyber Science and Technology Congress (DASC/PiCom/DataCom/CyberSciTech), IEEE, 2018, pp. 187–192.

[56] A. P. S. Bali, M. Fernandes, S. Choubey, M. Goel, Comparative performance of machine learning algorithms for fake news detection, in: International conference on advances in computing and data sciences, Springer, 2019, pp. 420–430.

[57] H. E. Wynne, Z. Z. Wint, Content based fake news detection using n-gram models, in: Proceedings of the 21st International Conference on Information Integration and Web-based Applications & Services, 2019, pp. 669–673.

[58] R. K. Kaliyar, A. Goswami, P. Narang, Multiclass fake news detection using ensemble machine learning, in: 2019 IEEE 9th International Conference on Advanced Computing (IACC), IEEE, 2019, pp. 103–107.

[59] A. Roy, K. Basak, A. Ekbal, P. Bhattacharyya, A deep ensemble framework for fake news detection and classification, arXiv preprint arXiv:1811.04670 (2018).

[60] S. Girgis, E. Amer, M. Gadallah, Deep learning algorithms for detecting fake news in online text, in: 2018 13th International Conference on Computer Engineering and Systems (ICCES), IEEE, 2018, pp. 93–97.

[61] A. M. Braşoveanu, R. Andonie, Semantic fake news detection: a machine learning perspective, in: International Work-Conference on Artificial Neural Networks, Springer, 2019, pp. 656–667.

[62] V. M. Krešňáková, M. Sarnovskỳ, P. Butka, Deep learning methods for fake news detection, in: 2019 IEEE 19th International Symposium on Computational Intelligence and Informatics and 7th IEEE International Conference on Recent Achievements in Mechatronics, Automation, Computer Sciences and Robotics (CINTI-MACRo), IEEE, 2019, pp. 000143–000148.

[63] F. B. Gereme, W. Zhu, Early detection of fake news" before it flies high", in: Proceedings of the 2nd International Conference on Big Data Technologies, 2019, pp. 142–148.

[64] A. Uppal, V. Sachdeva, S. Sharma, Fake news detection using discourse segment structure analysis, in: 2020 10th International Conference on Cloud Computing, Data Science & Engineering (Confluence), IEEE, 2020, pp. 751–756.

[65] F. Bogale Gereme, W. Zhu, Fighting fake news using deep learning: Pre-trained word embeddings and the embedding layer investigated, in: 2020 The 3rd International Conference on Computational Intelligence and Intelligent Systems, 2020, pp. 24–29.

[66] S. Kula, M. Choraś, R. Kozik, P. Ksieniewicz, M. Woźniak, Sentiment analysis for fake news detection by means of neural networks, in: International Conference on Computational Science, Springer, 2020, pp. 653–666.

[67] A. Abedalla, A. Al-Sadi, M. Abdullah, A closer look at fake news detection: A deep learning perspective, in: Proceedings of the 2019 3rd International Conference on Advances in Artificial Intelligence, 2019, pp. 24–28.

[68] P. Bahad, P. Saxena, R. Kamal, Fake news detection using bi-directional lstm-recurrent neural network, Procedia Computer Science 165 (2019) 74–82.

[69] A. Benamira, B. Devillers, E. Lesot, A. K. Ray, M. Saadi, F. D. Malliaros, Semi-supervised learning and graph neural networks for fake news detection, in: 2019 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM), IEEE, 2019, pp. 568–569.

[70] R. K. Kaliyar, A. Goswami, P. Narang, S. Sinha, Fndnet–a deep convolutional neural network for fake

news detection, Cognitive Systems Research 61 (2020) 32–44.

[71] W. Antoun, F. Baly, R. Achour, A. Hussein, H. Hajj, State of the art models for fake news detection tasks, in: 2020 IEEE International Conference on Informatics, IoT, and Enabling Technologies (ICIoT), IEEE, 2020, pp. 519–524.

[72] S. Yang, K. Shu, S. Wang, R. Gu, F. Wu, H. Liu, Unsupervised fake news detection on social media: A generative approach, in: Proceedings of the AAAI conference on artificial intelligence, Vol. 33, 2019, pp. 5644–5651.

[73] S. C. R. Gangireddy, C. Long, T. Chakraborty, Unsupervised fake news detection: A graph-based approach, in: Proceedings of the 31st ACM conference on hypertext and social media, 2020, pp. 75–83.

[74] J. Gaglani, Y. Gandhi, S. Gogate, A. Halbe, Unsupervised whatsapp fake news detection using semantic search, in: 2020 4th International Conference on Intelligent Computing and Control Systems (ICICCS), IEEE, 2020, pp. 285–289.

[75] D. Li, H. Guo, Z. Wang, Z. Zheng, Unsupervised fake news detection based on autoencoder, IEEE Access 9 (2021) 29356–29365.

[76] W. S. Paka, R. Bansal, A. Kaushik, S. Sengupta, T. Chakraborty, Cross-sean: A cross-stitch semi-supervised neural attention model for covid-19 fake news detection, Applied Soft Computing 107 (2021) 107393.

[77] X. Dong, U. Victor, L. Qian, Two-path deep semisupervised learning for timely fake news detection, IEEE Transactions on Computational Social Systems 7 (6) (2020) 1386–1398.

[78] P. Meel, D. K. Vishwakarma, A temporal ensembling based semi-supervised convnet for the detection of fake news articles, Expert Systems with Applications 177 (2021) 115002.

[79] R. Mansouri, M. Naderan-Tahan, M. J. Rashti, A semi-supervised learning method for fake news detection in social media, in: 2020 28th Iranian Conference on Electrical Engineering (ICEE), IEEE, 2020, pp. 1–5.

[80] U. Victor, Robust semi-supervised learning for fake news detection, Ph.D. thesis, Prairie View A&M University Prairie View, TX, USA (2020).

[81] M. S. Hasan, R. Alam, M. A. Adnan, Truth or lie: Pre-emptive detection of fake news in different languages through entropy-based active learning and multi-model neural ensemble, in: 2020 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM), IEEE, 2020, pp. 55–59.

[82] Y. Wang, W. Yang, F. Ma, J. Xu, B. Zhong, Q. Deng, J. Gao, Weak supervision for fake news detection via reinforcement learning, in: Proceedings of the AAAI conference on artificial intelligence, Vol. 34, 2020, pp. 516–523.

[83] M. Sarnovskỳ, V. Maslej-Krešňáková, N. Hrabovská, Annotated dataset for the fake news classification in slovak language, in: 2020 18th International Conference on Emerging eLearning Technologies and Applications (ICETA), IEEE, 2020, pp. 574–579.

[84] P. Přibáň, T. Hercig, J. Steinberger, Machine learning approach to fact-checking in west slavic languages, in: Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2019), 2019, pp. 973–979.

[85] M. Amjad, G. Sidorov, A. Zhila, Data augmentation using machine translation for fake news detection in the urdu language, in: Proceedings of The 12th Language Resources and Evaluation Conference, 2020, pp. 2537–2542.

[86] N. Lina, S. Fua, S. Jianga, Fake news detection in the urdu language using charcnn-roberta, Health 100 (2020) 100.

[87] F. Balouchzahi, H. Shashirekha, Learning models for urdu fake news detection., in: FIRE (Working Notes), 2020, pp. 474–479.

[88] M. Amjad, G. Sidorov, A. Zhila, A. F. Gelbukh, P. Rosso, Overview of the shared task on fake news detection in urdu at fire 2020., in: FIRE (Working Notes), 2020, pp. 434–446.

[89] A. F. U. R. Khiljia, S. R. Laskara, P. Pakraya, S. Bandyopadhyaya, Urdu fake news detection using generalized autoregressors (2020).

[90] M. Amjad, G. Sidorov, A. Zhila, A. Gelbukh, P. Rosso, Urdufake@ fire2020: Shared track on fake news identification in urdu, in: Forum for Information Retrieval Evaluation, 2020, pp. 37–40.

[91] J. L. Alves, L. Weitzel, P. Quaresma, C. E. Cardoso, L. Cunha, Brazilian presidential elections in the era of misinformation: A machine learning approach to analyse fake news, in: Iberoamerican Congress on Pattern Recognition, Springer, 2019, pp. 72–84.

[92] M. Paixao, R. Lima, B. Espinasse, Fake news classification and topic modeling in brazilian portuguese, in: 2020 IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT), IEEE, 2020, pp. 427–432.

[93] H. Q. Abonizio, J. I. de Morais, G. M. Tavares, S. Barbon Junior, Language-independent fake news detection: English, portuguese, and spanish mutual features, Future Internet 12 (5) (2020) 87.

[94] R. M. Silva, R. L. Santos, T. A. Almeida, T. A. Pardo, Towards automatically filtering fake news in portuguese, Expert Systems with Applications 146 (2020) 113199.

[95] D.-H. Lee, Y.-R. Kim, H.-J. Kim, S.-M. Park, Y.-J. Yang, Fake news detection using deep learning, Journal of Information Processing Systems 15 (5) (2019) 1119–1130.

[96] Y.-C. Ahn, C.-S. Jeong, Natural language contents evaluation system for detecting fake news using deep learning, in: 2019 16th International Joint Conference on Computer Science and Software Engineering (JCSSE), IEEE, 2019, pp. 289–292.

[97] A. Verma, V. Mittal, S. Dawn, Find: Fake information and news detections using deep learning, in: 2019 Twelfth International Conference on Contemporary Computing (IC3), IEEE, 2019, pp. 1–7.

[98] I. Vogel, P. Jiang, Fake news detection with the new german dataset germanfakenc, in: International Conference on Theory and Practice of Digital Libraries, Springer, 2019, pp. 288–295.

[99] P. H. A. Faustini, T. F. Covoes, Fake news detection in multiple platforms and languages, Expert Systems with Applications 158 (2020) 113503.

[100] I. Vogel, M. Meghana, Detecting fake news spreaders on twitter from a multilingual perspective, in: 2020 IEEE 7th International Conference on Data Science and Advanced Analytics (DSAA), IEEE, 2020, pp. 599–606.

[101] D. Zaizar-Gutiérrez, D. Fajardo-Delgado, M. Á. Á. Carmona, Itcg's participation at mex-a3t 2020: Aggressive identification and fake news detection based on textual features for mexican spanish., in: IberLEF@ SEPLN, 2020, pp. 258–264.

[102] M. Gulzar Hussain, M. Rashidul Hasan, M. Rahman, J. Protim, S. Al Hasan, Detection of bangla fake news using mnb and svm classifier, arXiv e-prints (2020) arXiv–2005.

[103] S. B. S. Mugdha, S. M. Ferdous, A. Fahmin, Evaluating machine learning algorithms for bengali fake news detection, in: 2020 23rd International Conference on Computer and Information Technology (ICCIT), IEEE, 2020, pp. 1–6.

[104] A. Zervopoulos, A. G. Alvanou, K. Bezas, A. Papamichail, M. Maragoudakis, K. Kermanidis, Hong kong protests: using natural language processing for fake news detection on twitter, in: IFIP International Conference on Artificial Intelligence Applications and Innovations, Springer, 2020, pp. 408–419.

[105] A. Rusli, J. C. Young, N. M. S. Iswari, Identifying fake news in indonesian via supervised binary text classification, in: 2020 IEEE International Conference on Industry 4.0, Artificial Intelligence, and Communications Technology (IAICT), IEEE, 2020, pp. 86–90.

[106] A. Zubiaga, E. Kochkina, M. Liakata, R. Procter, M. Lukasik, K. Bontcheva, T. Cohn, I. Augenstein, Discourse-aware rumour stance classification in social media using sequential classifiers, Information Processing & Management 54 (2) (2018) 273–290.

[107] S. Vosoughi, D. Roy, S. Aral, The spread of true and false news online, Science 359 (6380) (2018) 1146–1151.

[108] N. Vo, K. Lee, Learning from fact-checkers: Analysis and generation of fact-checking language, in: Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval, 2019, pp. 335–344.

[109] X. Zhou, A. Jain, V. V. Phoha, R. Zafarani, Fake news early detection: A theory-driven model, Digital Threats: Research and Practice 1 (2) (2020) 1–25.

[110] I. Ahmad, M. Yousaf, S. Yousaf, M. O. Ahmad, Fake news detection using machine learning ensemble methods, Complexity 2020 (2020).

[111] J. Xue, Y. Wang, S. Xu, L. Shi, L. Wei, H. Song, Mvfnn: Multi-vision fusion neural network for fake news picture detection, in: International Conference on Computer Animation and Social Agents, Springer, 2020, pp. 112–119.

[112] S. Singhal, R. R. Shah, T. Chakraborty, P. Kumaraguru, S. Satoh, Spotfake: A multi-modal framework for fake news detection, in: 2019 IEEE fifth international conference on multimedia big data (BigMM), IEEE, 2019, pp. 39–47.

[113] E. Masciari, V. Moscato, A. Picariello, G. Sperlí, Detecting fake news by image analysis, in: Proceedings of the 24th Symposium on International Database Engineering & Applications, 2020, pp. 1–5.

[114] A. Bani-Hani, O. Adedugbe, E. Benkhelifa, M. Majdalawieh, F. Al-Obeidat, A semantic model for context-based fake news detection on social media, in: 2020 IEEE/ACS 17th International Conference on Computer Systems and Applications (AICCSA), IEEE, 2020, pp. 1–7.

[115] M. M. M. Hlaing, N. S. M. Kham, Defining news authenticity on social media using machine learning approach, in: 2020 IEEE Conference on Computer Applications (ICCA), IEEE, 2020, pp. 1–6.

[116] M. Meyers, G. Weiss, G. Spanakis, Fake news detection on twitter using propagation structures, in: Multidisciplinary International Symposium on Disinformation in Open Online Media, Springer, 2020, pp. 138–158.

[117] M. K. Balwant, Bidirectional lstm based on pos tags and cnn architecture for fake news detection, in: 2019 10th International Conference on Computing, Communication and Networking Technologies (ICCCNT), IEEE, 2019, pp. 1–6.

[118] L. Cui, K. Shu, S. Wang, D. Lee, H. Liu, defend: A system for explainable fake news detection, in: Proceedings of the 28th ACM international conference on information and knowledge management, 2019, pp. 2961–2964.

[119] J. C. Reis, A. Correia, F. Murai, A. Veloso, F. Benevenuto, Explainable machine learning for fake news detection, in: Proceedings of the 10th ACM conference on web science, 2019, pp. 17–26.

[120] Y. Wang, H. Han, Y. Ding, X. Wang, Q. Liao, Learning contextual features with multi-head self-attention for fake news detection, in: International Conference on Cognitive Computing, Springer, 2019, pp. 132–142.

[121] M. Choraś, et al., Advanced machine learning techniques for fake news (online disinformation) detection: A systematic mapping study, Applied Soft Computing 101 (2021) 107050.

[122] K. Nakamura, S. Levy, W. Y. Wang, r/fakeddit: A new multimodal benchmark dataset for fine-grained fake news detection, arXiv preprint arXiv:1911.03854 (2019).

[123] Y. Seo, C.-S. Jeong, Fagon: Fake news detection model using grammatic transformation on neural network, in: 2018 Thirteenth International Conference on Knowledge, Information and Creativity Support Systems (KICSS), IEEE, 2018, pp. 1–5.

[124] G. Shmueli, P. C. Bruce, I. Yahav, N. R. Patel, K. C. Lichtendahl Jr, Data mining for business analytics: concepts, techniques, and applications in R, John Wiley & Sons, 2017.

[125] All you need to know about text preprocessing for nlp and machine learning.
URL https://www.kdnuggets.com/2019/04/text-preprocessing-nlp-machine-learning.html

[126] Z. S. Harris, Distributional structure, Word 10 (2-3) (1954) 146–162.

[127] J. Brownlee, A gentle introduction to the bag-of-words model (Aug 2019).
URL https://machinelearningmastery.com/gentle-introduction-bag-words-model/

[128] K. S. Jones, A statistical interpretation of term specificity and its application in retrieval, Journal of documentation (1972).

[129] Y. Bengio, R. Ducharme, P. Vincent, A neural probabilistic language model, Advances in Neural Information Processing Systems 13 (2000).

[130] T. Mikolov, K. Chen, G. Corrado, J. Dean, Efficient estimation of word representations in vector space, arXiv preprint arXiv:1301.3781 (2013).

[131] J. Pennington, R. Socher, C. D. Manning, Glove: Global vectors for word representation, in: Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP), 2014, pp. 1532–1543.

[132] P. Bojanowski, E. Grave, A. Joulin, T. Mikolov, Enriching word vectors with subword information, Transactions of the Association for Computational Linguistics 5 (2017) 135–146.

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

[134] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, I. Sutskever, et al., Language models are unsupervised multitask learners, OpenAI blog 1 (8) (2019) 9.

[135] V. Sanh, L. Debut, J. Chaumond, T. Wolf, Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter, arXiv preprint arXiv:1910.01108 (2019).

[136] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, V. Stoyanov, Roberta: A robustly optimized bert pretraining approach, arXiv preprint arXiv:1907.11692 (2019).

[137] M. Lewis, Y. Liu, N. Goyal, M. Ghazvininejad, A. Mohamed, O. Levy, V. Stoyanov, L. Zettlemoyer, Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension, arXiv preprint arXiv:1910.13461 (2019).

[138] K. Clark, M.-T. Luong, Q. V. Le, C. D. Manning, Electra: Pre-training text encoders as discriminators rather than generators, arXiv preprint arXiv:2003.10555 (2020).

[139] Z. Yang, Z. Dai, Y. Yang, J. Carbonell, R. R. Salakhutdinov, Q. V. Le, Xlnet: Generalized autoregressive pretraining for language understanding, Advances in neural information processing systems 32 (2019).

[140] A. Louis, Master's Thesis :NetBERT: A Pre-trained Language Representation Model for Computer Networking 95.

[141] M. E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, L. Zettlemoyer, Deep contextualized word representations (2018). doi:10.48550/ARXIV.1802.05365.
URL https://arxiv.org/abs/1802.05365

[142] A. Wang, A. Singh, J. Michael, F. Hill, O. Levy, S. R. Bowman, Glue: A multi-task benchmark and analysis platform for natural language understanding, arXiv preprint arXiv:1804.07461 (2018).

[143] G. Hinton, O. Vinyals, J. Dean, et al., Distilling the knowledge in a neural network, arXiv preprint arXiv:1503.02531 2 (7) (2015).

[144] Z. Lan, M. Chen, S. Goodman, K. Gimpel, P. Sharma, R. Soricut, Albert: A lite bert for self-supervised learning of language representations, arXiv preprint arXiv:1909.11942 (2019).

[145] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar, et al., Llama: Open and efficient foundation language models, arXiv preprint arXiv:2302.13971 (2023).

[146] G. Penedo, Q. Malartic, D. Hesslow, R. Cojocaru, A. Cappelli, H. Alobeidli, B. Pannier, E. Almazrouei, J. Launay, The RefinedWeb dataset for Falcon LLM: outperforming curated corpora with web data, and web data only, arXiv preprint arXiv:2306.01116 (2023). arXiv:2306.01116.

[147] L. I. Kuncheva, Combining pattern classifiers: methods and algorithms, John Wiley & Sons, 2014.

[148] L. Breiman, Random forests, Machine learning 45 (1) (2001) 5–32.

[149] Y. Freund, R. E. Schapire, et al., Experiments with a new boosting algorithm, in: icml, Vol. 96, Citeseer, 1996, pp. 148–156.

[150] Y. Freund, R. E. Schapire, A decision-theoretic generalization of on-line learning and an application to boosting, in: Computational Learning Theory: Second European Conference, EuroCOLT'95 Barcelona, Spain, March 13–15, 1995 Proceedings 2, Springer, 1995, pp. 23–37.

[151] S. Minaee, N. Kalchbrenner, E. Cambria, N. Nikzad, M. Chenaghlu, J. Gao, Deep learning–based text classification: a comprehensive review, ACM Computing Surveys (CSUR) 54 (3) (2021) 1–40.

[152] Q. Li, H. Peng, J. Li, C. Xia, R. Yang, L. Sun, P. S. Yu, L. He, A survey on text classification: From traditional to deep learning, ACM Transactions on Intelligent Systems and Technology (TIST) 13 (2) (2022) 1–41.

[153] S. Hochreiter, J. Schmidhuber, Long short-term memory, Neural computation 9 (8) (1997) 1735–1780.

[154] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, I. Polosukhin,

Attention is all you need, Advances in neural information processing systems 30 (2017).

[155] G. McIntire, Mcintire fake news dataset, available online at: https://github.com/lutzhamel/fake-news, last accessed on 06.06.2023 (2017).

[156] P. Patwa, S. Sharma, S. PYKL, V. Guptha, G. Kumari, M. S. Akhtar, A. Ekbal, A. Das, T. Chakraborty, Fighting an infodemic: Covid-19 fake news dataset (2020). arXiv:2011.03327.

[157] S. Bird, E. Klein, E. Loper, Natural language processing with Python: analyzing text with the natural language toolkit, " O'Reilly Media, Inc.", 2009.

[158] F. Pedregosa, et al., Scikit-learn: Machine learning in Python, Journal of Machine Learning Research 12 (2011) 2825–2830.

[159] R. Rehurek, P. Sojka, Gensim–python framework for vector space modelling, NLP Centre, Faculty of Informatics, Masaryk University, Brno, Czech Republic 3 (2) (2011).

[160] T. Wolf, et al., Transformers: State-of-the-art natural language processing, in: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, Association for Computational Linguistics, Online, 2020, pp. 38–45.
URL https://www.aclweb.org/anthology/2020.emnlp-demos.6

[161] A. Paszke, et al., Pytorch: An imperative style, high-performance deep learning library, in: H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, R. Garnett (Eds.), Advances in Neural Information Processing Systems 32, Curran Associates, Inc., 2019, pp. 8024–8035.
URL http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learn
pdf

[162] R. M. Cruz, W. V. de Sousa, G. D. Cavalcanti, Selecting and combining complementary feature representations and classifiers for hate speech detection, Online Social Networks and Media 28 (2022) 100194.

[163] M. Abadi, et al., TensorFlow: Large-scale machine learning on heterogeneous systems, software available from tensorflow.org (2015).
URL https://www.tensorflow.org/

[164] C. Ferri, J. Hernández-Orallo, R. Modroiu, An experimental comparison of performance measures for classification, Pattern recognition letters 30 (1) (2009) 27–38.

[165] P. McCullagh, Regression models for ordinal data, Journal of the Royal Statistical Society: Series B (Methodological) 42 (2) (1980) 109–127.

[166] P. A. Gutiérrez, M. Perez-Ortiz, J. Sanchez-Monedero, F. Fernandez-Navarro, C. Hervas-Martinez, Ordinal regression methods: survey and experimental study, IEEE Transactions on Knowledge and Data Engineering 28 (1) (2015) 127–146.

[167] L. L. Minku, A. P. White, X. Yao, The impact of diversity on online ensemble learning in the presence

of concept drift, IEEE Transactions on knowledge and Data Engineering 22 (5) (2009) 730–742.

[168] A. Ross, W. Pan, L. Celi, F. Doshi-Velez, Ensembles of locally independent prediction models, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 34, 2020, pp. 5527–5536.

[169] D. Teney, E. Abbasnejad, S. Lucey, A. Van den Hengel, Evading the simplicity bias: Training a diverse set of models discovers solutions with superior ood generalization, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 16761–16772.

[170] R. M. Cruz, R. Sabourin, G. D. Cavalcanti, Dynamic classifier selection: Recent advances and perspectives, Information Fusion 41 (2018) 195–216.

[171] T. Tommasi, T. Tuytelaars, A testbed for cross-dataset analysis, in: Computer Vision-ECCV 2014 Workshops: Zurich, Switzerland, September 6-7 and 12, 2014, Proceedings, Part III 13, Springer, 2015, pp. 18–31.

Appendix A. Hyperparameters for classical and ensemble models

Method	Hyperparameters
SVM	Kernel:['rbf']
	Gamma: [1, 0.1, 0.01, 0.001, 0.0001]
	C: [0.1, 1, 10, 100, 1000]
LR	solver: ['liblinear']
	penalty: ['none', 'l1', 'l2', 'elasticnet']
	C: [0.01, 0.1, 1, 10, 100]
NB	alpha: [0.1, 0.5, 1]
	fit_prior: [False, True]
KNN	n_neighbors: [1 - 20]
RF	bootstrap: [True, False]
	max_depth: [5,10, 20, 30, 40, 50]
	max_features: ['auto', 'sqrt', 'log2']
	min_samples_leaf: [1, 2, 4]
	min_samples_split: [2, 5, 10]
	n_estimators: [200, 400, 600, 800, 1000]
	criterion: ['gini', 'entropy']
AdaBoost	n_estimators: [10, 50, 100, 200, 300, 400, 500, 1000]
	learning_rate: [0.001, 0.01, 0.1, 0.2, 0.5]
XGBoost	n_estimators: [200,300,400,500]
	max_features: ['sqrt', 'log2']
	max_depth: [4,5,6,7,8]
	criterion: ['gini', 'entropy']
	random_state: [18]
MLP	Activation function: [ReLU, logistic]
	solver: [Adam, lbfgs]

Table A.13: Hyper-parameters considered for machine learning and ensemble models.

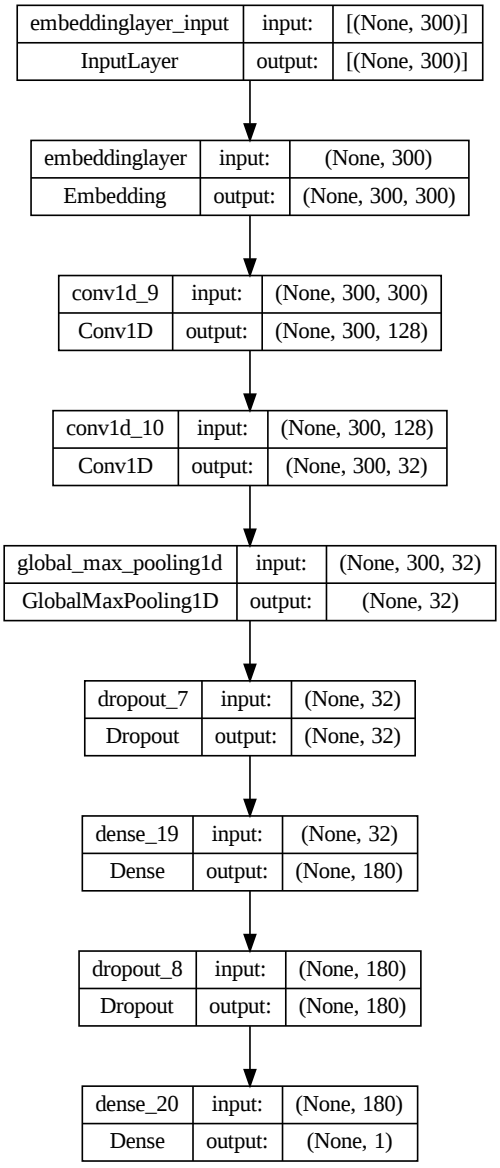
Appendix B. Hyperparameters for deep learning models

Method	Setup
CNN	Activation function: [sigmoid, ReLU]
	Batch size: [64, 128, 512]
	Number of epochs: [5,20,100]
	Optimizer: [Adam]
	Learning rate: 0.001
	Dropout: 0.2
BiLSTM	Activation function: [sigmoid, ReLU]
	Batch size: [64, 128, 512]
	Number of epochs: [5,20,100]
	Optimizer: [Adam]
	Learning rate: 0.001
	Hidden size: 128
	Dropout: 0.2

Table B.14: Hyper-parameters considered for all deep learning models.

Appendix C. Deep learning model architectures

The CNN architecture comprises two convolutional layers with 128 and 32 filters, an embedding layer, and global max pooling. A dropout rate of 0.2 was introduced between two dense layers, with ReLU activation and 180 units in each. For binary classification, a sigmoid activation function was added in the last layer, whereas for multi-class classification (LIAR dataset), a softmax activation function was used. The model was trained using the Adam optimizer and cross-entropy loss. The BiLSTM architecture consists of an embedding layer with a dense layer with 128 units, with ReLU activation. The sequences were processed bidirectionally by an LSTM layer with 128 units. A dropout rate of 0.2 was added between two dense layers, each with 32 units and a ReLU activation function. In the case of binary classification, a sigmoid activation function was employed in the last layer, and for multi-class classification, the softmax activation function was utilized. The model was trained using the Adam optimizer and cross-entropy loss. It is important to note that the other deep learning models used the same architecture; the sole difference is their embedding dimensions.



(a) CNN Model Architecture

(b) BiLSTM Model Architecture

Figure C.11: Deep Models Architecture

Appendix D. Search Methodology

We conducted a search methodology to find relevant primary and secondary studies related to fake news detection. In the first step, based on the research questions and objectives

of our research, we used specific terms and keywords to create search queries including, *Fake news detection*, *Machine learning*, *Deep learning*, *Ensemble*, *NLP*, *Text classification*. In the second step, we searched the queries that were created based on the keywords in the most popular databases and digital libraries including Google Scholar, ACM digital library, IEEE Explore, Springer, and Engineering Village database. In the third step, we extract and classify the information of query results. Finally, we screened the results based on some inclusive and exclusive criteria in order to reduce the number of results.

- **Inclusive Criteria:** English papers, papers related to fake news detection methods including machine learning, deep learning, and ensemble learning, peer-reviewed journal articles and conferences, Papers with a publication date from 2014 to 2022.
- **Exclusive Criteria:** Non-peer-reviewed journal articles, non-English papers, Lake of clear methodology and results, Low impact papers including low citation and impact factor, outdated papers with older than 10 years, Duplicate records.

Click here to access/download

LaTeX Source File

(Revised version) Comparative study article.zip

Declaration of interests

☒The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

☐The authors declare the following financial interests/personal relationships which may be considered as potential competing interests:

Faramarz Farhangian: Conceptualization, Methodology, Software, Validation, Formal analysis, Investigation, Writing – original draft.
Rafael M.O. Cruz: Supervision, Conceptualization, Methodology, Resources, Formal analysis, Writing – review & editing.
George D. C. Cavalcanti: Supervision, Conceptualization, Methodology, Formal analysis, Review & editing.