Enhancing Fake News Classification Through an Argumentation-Based Pipeline

Mayla Kersten and Giovanni Sileno

University of Amsterdam, Amsterdam, the Netherlands
maylakersten@gmail.com, g.sileno@uva.nl

Abstract. Various methods have been proposed in the literature for classifying fake news. Yet, the dynamic and diverse nature of fake news (texts of varying lengths, topics concerning different domains, and continuously evolving content) remains a relevant challenge. This study investigates whether this problem can be tackled by introducing an argumentation-based pipeline. The intuition is that, by extracting the most relevant arguments in the text, the classifier will be less dependent on contingent aspects such as length, or less relevant content. We run several experiments to measure the performance of a fake news classifier trained by fine-tuning state-of-the-art pre-trained language models employing argumentation structures (claim, evidence) extracted from fake and real news texts. For the extraction of arguments, we apply two off-the-shelf solutions: MARGOT (an argument mining tool) and Dolly 2.0 (an instruction-following large language model). Our initial experiments show promising results with respect to our baseline pipeline, particularly with respect to long texts.

Keywords: Fake news detection · Text Classification · Argument mining · Argumentation · Large Language Models

1 Introduction

In recent years, the proliferation of false or misleading information, commonly called "fake news", has become a significant concern for individuals and society. The propagation of such misinformation has far-reaching consequences, including the polarization of public opinion [10, 16], the incitement of hate speech [20] and violence [31], and the erosion of trust in democratic institutions [47]. In line with the magnitude of the problem, there has been a growing interest in developing effective strategies to deal with fake news, resulting, amongst other applications, in various automated classification methods [12, 13, 29, 33, 34].

Khan et al.'s work [13] compares various existing techniques to classify fake news, including traditional machine learning models, deep learning algorithms, and pre-trained language models. By utilizing pre-trained BERT-based language models [7], Khan et al. [13] achieved the highest overall performance, positioning them as the current state-of-the-art. Khan et al. [13] also found that text lengths substantially impact the accuracy of the models; on the Liar dataset, using only

short texts, the highest accuracy achieved was 0.62, in stark contrast to the higher accuracy of 0.98 measured on longer texts [13].

This work stems from the hypothesis that argumentation knowledge may pose a valuable addition to the intricate task of distinguishing between fake and real news, as humans naturally employ argumentation to assess the validity of information [15, 36]. While considerable advancements have been made in automatically extracting argumentation from texts accurately [4, 14, 18], integrating argumentation into fake news classification remains understudied. To the best of our knowledge, the study by Yan et al. [43] is the only example of incorporating argument components into fake news detection. However, their approach relies on manual annotation of argument components, limiting its scalability and applicability to other data [43]. Given the dynamic nature of fake news, where new topics emerge daily, this limitation poses a significant concern and triggers questions on whether such a preparatory task could be performed automatically.

Our study aims therefore to assess the performance of current state-of-the-art pre-trained BERT-based language models for the task of fake news classification by comparing two pipelines: a baseline pipeline (taking as input preprocessed texts) and an argumentation-based pipeline (taking as input a text made of claims and evidence extracted from preprocessed texts). Motivated by the finding by Khan et al. [13] regarding the relationship between the models' performance and the length of texts, we will also incorporate text lengths as a factor in our performance evaluation.

The paper is organized as follows. Section 2 will give a more detailed overview of relevant contributions to fake news detection and argument extraction. Section 3 will present the proposed methodology and the experimental setup. Section 4 analyses the results, with a discussion in Section 5. A conclusion section ends the paper, with a note on limitations and on future works.

2 Related Work

2.1 Fake news detection

In recent years, researchers have proposed various machine-learning [11, 17, 37, 46] and deep-learning methods [13, 27, 32] devoted to counteract the spread of fake news. These models can be broadly categorized into context-based, content-based approaches, or a combination of the two. Recently, large language models [13] have also been deployed to classify fake news.

Context-based models leverage supplementary knowledge, such as user engagements, to distinguish between fake and real news. For example, Li et al. [17] propose a Graph Convolutional Network (GCN) to capture the political perspective of a news document. The experiments conducted on a dataset of 10,385 news articles and 1,604 Twitter users who frequently share news articles and follow politicians achieved an accuracy of up to 0.89 for multi-class classification. A comparable approach, by Yang et al. [44], leverages a Bayesian Network Model to identify users' opinions toward the authenticity of the news. The experiments

were conducted on two datasets: one comprising 1,627 news articles from Facebook and the second, known as the Liar dataset[1][42, 44]. For the users' opinions on the articles, tweets corresponding to the news articles were collected. Yang et al. [44] achieved an overall performance of up to 0.74 in the binary classification of fake news. While these context-based approaches demonstrate high performances, they are limited due to the reliability of user engagement.

Content-based models leverage linguistic and syntactic cues to differentiate between fake and real news. A study by Rashkin et al. [32] compares the language used in fake and real news. Their approach employs lexical resources derived from communication theory and computational linguistics to capture the distinctive characteristics of fake and real news [32]. These characteristics are fed into a long short-term memory network (LSTM) model, resulting in an accuracy of 0.58 against a dataset of 10.000 labeled quotes [32]. This study outperforms experiments using LSTM models alone, e.g., by Oshikawa et al. [27] — in turn outperforming Convolutional Neural Networks (CNN). The work by Hu et al. [11] focuses on capturing news similarity, proposing a multi-depth graph convolutional networks framework, achieving an accuracy of 0.49 on the aforementioned Liar [45] dataset. The relatively low performance of these approaches indicates that relying solely on stylistic differences is insufficient for effectively distinguishing fake news. This limitation is also emphasized by Zhou et al. [46], highlighting the vulnerability for potentially misclassifying fact-tampering fake news and incorrectly categorizing under-written real news.

Combining content- and context-based models Zhou et al. [46] address this concern. Their hybrid solution involves extracting causal relationships within the news articles, comparing them with a dynamically updated knowledge graph, and capturing linguistic characteristics [46]. They evaluate their method on the Fake or Real news dataset[2]. The hybrid approach yields an accuracy score of 0.62 [46]. While their approach outperforms the previous approaches and tackles an important challenge content-based models face: their approach does not take away the dependence on additional knowledge [46].

Large language models used for the classification task are the most recent advancement in the field of fake news detection. The benchmark study reported in [13] compares various approaches, including previously discussed approaches, to pre-trained BERT-based [7] models. They found that these pre-trained large language models significantly outperform other approaches, achieving an overall accuracy score of 0.62 on the Liar and 0.98 on the Fake or Real News dataset [13]. The performance is attributed to the bi-directional nature of these models, allowing them to capture contextual information, surpassing the capabilities of alternative approaches [13].

[1] The Liar dataset [42] contains 12,790 English statements collected from various sources and labeled fake or real by PolitiFact [41]. This dataset is considered a benchmark dataset in the field of fake news detection.

[2] The Fake or Real news dataset [21] includes 4,594 fake and real news articles surrounding the 2016 U.S. elections, and is a second benchmark dataset commonly used in the field.

2.2 Argumentation

Argumentation is crucial to humans, to evaluate the validity of ideas, to convince an addressee, or to solve a difference of opinion [8]. Following the literature [8, 9, 15], argumentation is here considered to consist of a statement to be a standpoint, henceforth called a *claim*, and a set of backing premises, henceforth referred to as *evidence*. For the argumentation to be valid, a logical connection between the presented pieces of information must be found, enabling the inference of the claim from the evidence [9]. Reasoning plays a crucial role in determining whether such a connection exists [23]. However, despite humans' innate capacity for reasoning, they are found to be highly susceptible to misinformation [23, 28]. The extent to which people engage in reasoning is heavily influenced by factors such as their personal partisanship and background knowledge [28]. Individuals are more inclined to challenge argumentation when they do not initially agree with it [22]. This, for example, explains why individuals who support gun rights and lack knowledge about the gun-related homicide rate in Europe are less likely to challenge a factually false claim like the following, extracted from the Liar [45] dataset: "America[']s gun-related homicide rate would be about the same as Belgium's if you left out California". Mercier and Spencer [23] also suggest that individuals are more inclined to critically examine and question arguments that contradict their existing beliefs or positions.

There has been little attention to integrating the argumentative dimension into fake news detection. To the best of our knowledge, the study by Yan et al. [43] is the only counterexample. They use an annotated news editorial of argumentation provided by Al Khatib et al. [1], including 300 annotated articles [43]. Feeding the manually annotated argument components from news articles into LSTM models, resulted in accuracy scores ranging to 0.94. Additionally, they explored using BERT-based language models as well, achieving accuracy scores up to 0.93 [43]. However, such high performance relies heavily on manually annotated argumentation, posing limitations in terms of scalability. Another observation from the study is that the results varied significantly across different topics, highlighting the topic-sensitivity of their approach.

Instead of relying on annotated data, a solution could be to utilize existing automated argumentation component detectors [4, 14, 18]. The automated detection method typically involves a two-step process. First, a prediction is made about whether or not a sentence is argumentative, and second, a boundary module is employed to detect the beginning and end of the argument in argumentative sentences [4, 14, 18]. Unfortunately, many of these relatively old tools are not openly accessible, with MARGOT [18] being the only exception in our findings. To deal with this limitation, we have considered leveraging generative language models such as Dolly 2.0 [24]. These models are specifically designed to comprehend and generate text based on given prompts and instructions, which makes them potentially suitable for extracting argument components from texts [24].

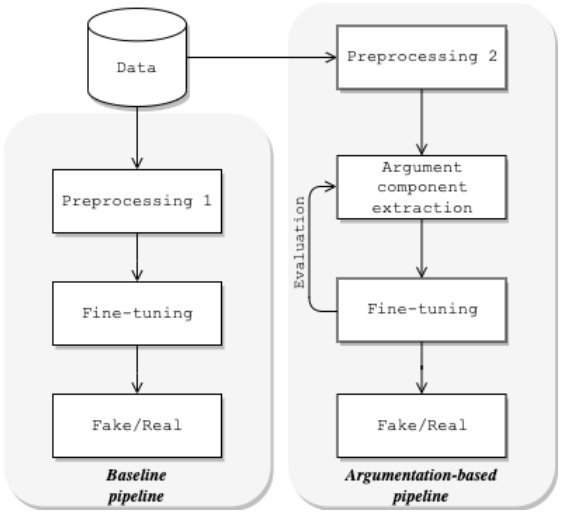


Fig. 1: Overview of baseline and argumentation-based pipelines.

3 Methodology

This Section presents the methodology employed in our study. As illustrated in Fig. 1, we performed experiments with two distinct pipelines. A *baseline* pipeline, constructed following state-of-the-art methods, and an *argumentation-based* pipeline, including an argument extraction tool. In Section 3.1 we describe the datasets we have used for the training in both pipelines. Section 3.2 describes the pre-processing steps, which vary depending on the pipeline. Section 3.3 lists the pre-trained models we selected for the experiments. Details on the experiments and the evaluation methods are provided respectively in sections 3.4 and 3.5. The code of all experiments is publicly available.[3]

3.1 Data

Datasets We compiled a dataset containing English texts labeled *FAKE* or *REAL* from three publicly available datasets: Fake and Real News [2], Fake or Real News [21], and Liar [45]. The texts in these datasets originate from various sources, including newspapers and social media platforms. We use two datasets that were also used in the work by Khan et al. [13]: Fake or Real News [21], consisting of 4,594 news articles gathered during the 2016 U.S elections, and Liar [45] containing 12,790 short statements collected from 2007 to 2016, manually labeled by PolitiFact [41]. As a new addition to our research, we include the Fake and Real News dataset [2], which contains 44,898 news articles related to politics collected from 2015 until 2018. Table 1 demonstrates statistics about

[3] `https://github.com/himayla/fake_news`

	Fake and Real News [2]	Fake or Real News [21]	Liar [45]
Total number of texts	4594	44898	12790
FAKE label (%)	50.0%	52.3%	44.2%
REAL label (%)	50.0%	47.7%	55.8%
Min. number of words	1	1	11
Max. number of words	115372	51794	3192
Average # of words	4861.67	2469.11	107.15

Table 1: Statistics of the three datasets used in this study

these three datasets. We see a balanced distribution of *FAKE* and *REAL* labels. Yet, the texts exhibit significant variation in terms of length.

Data wrangling We applied data wrangling steps so that the three datasets include only texts and corresponding binary labels. For the datasets Fake or Real News and Fake and Real News, we considered only the *text* and *label* columns, discarding any additional metadata. Regarding the Liar dataset, we transform the original six-point scale for labeling statements into binary classifications. Specifically, statements labeled as *true*, *half-true*, and *mostly-true* are labeled *REAL*, while those labeled as *barely-true*, *pants-fire*, and *false* are labeled *FAKE*. Additionally, for Fake and Real News, we remove duplicate texts. We did not do this for Fake or Real News and Liar to keep these datasets consistent with those deployed in the study by Khan et al. [13].

Subsampling and splitting Due to time constraints (see Limitations in Section 6), we subsampled the three datasets for conducting our experiments. In total, we collected 8,831 texts and their corresponding labels. After the subsampling, we checked whether the distribution of texts labeled *FAKE* and *REAL* and the average number of words remained balanced, As a final step, we separated the selected texts into a training, validation, and test set using an 80:15:5 ratio.

3.2 Preprocessing

Baseline pipeline For our baseline pipeline, we replicated the preprocessing steps (e.g., stemming) described in Khan et al. [13]. These techniques effectively reduce the dimensionality of the textual data and mitigate noise. We removed any URL addresses in the text using regular expressions. Next, we utilized the Natural Language Toolkit (NLTK) [26] to tokenize the text into word-level tokens and remove common stop words. Additionally, we employed a contextual spelling correction algorithm [38] to address any spelling errors in the text. To further simplify the text, we applied the NLTK SnowballStemmer [26] algorithm. Finally, we reassembled the word tokens by inserting white spaces between them. Analyzing the output distribution, we observe that most texts consist of less than 50 words. Yet, 128 texts in our subsample surpassed the 1,000-word threshold,

with some extending up to 7,481 words. We have chosen not to remove these outliers from our subsample for a better alignment between the two pipelines.

Argumentation-based pipeline Our focus in the argumentation-based pipeline is to conduct the analysis at a sentence and paragraph level, and therefore, in contrast to the baseline pipeline, we prioritized preserving the original words to capture the nuanced details necessary for comprehensive argument analysis. As a first step, we converted all text to lowercase. This passage facilitates faster processing and alleviates potential issues related to case sensitivity. We then created regular expressions specific to our study to remove non-argumentative parts, such as "Featured image by", to reduce noise. More importantly, we made changes to handle punctuation because MARGOT operates on the sentence level [18]. We expanded acronyms and abbreviations, such as "F.B.I" and "wouldn't", to their full forms ("Federal Bureau of Investigation", "would not"). Additionally, we ensured that all sentences in the texts had consistent punctuation. The majority of texts cleaned for the argumentation-based pipeline, including the abbreviation expansion, contain less than 100 words. However, it is noteworthy that 92 texts exceed 2,000 words, with the longest text spanning 13,739 words. Similarly to the baseline pipeline, we have decided not to exclude these outliers from our dataset.

3.3 Argument extraction

During our research, we identified several tools that may have been relevant for argument extraction [3, 4, 14, 18, 30]. Unfortunately, despite all our efforts, the only argument mining tool we managed to have access to was MARGOT. The use of Dolly 2.0 came out of the necessity to have an additional alternative.

MARGOT [18] was the first publicly available argument-mining tool with applications across different domains. The MARGOT system segments the input texts into sentences. By employing a tree-kernel-based classifier, it assigns to each sentence a score indicating the presence of an argumentative component: a *claim*, or an *evidence* [18]. It then utilizes a boundary module to identify the beginnings and endings of claims and evidence within the predicted sentences [18]. To have MARGOT running, we utilized Stanford CoreNLP [39] and SVM-Light 1.5 [25]. In our pipeline, to annotate each news article, we provided the clean texts of our dataset as input to MARGOT. Consider the following example, labeled as *FAKE*, extracted from the Liar dataset: "*Honduras bans citizens from owning guns and has the highest homicide rate in the entire world. Switzerland, with a similar population, requires citizens to own guns and has the lowest homicide rate in the entire world. Barack Obama and Hillary Clinton are negotiating with the United Nations about doing a treaty that will ban the use of firearms.*". MARGOT segments the text into three sentences and assign scores to each of them, respectively 0.29, 0.52, and -0.72 for claim scores, and -0.16, -0.30, 0.39 for evidence scores. This means that MARGOT classifies the first two sentences as a claim and the third as evidence. We stored the output of this process for each text in our dataset to capture the detected argumentative components.

When multiple claims are present, we join the elements of the list to form a coherent input for the text classifier. For instance, for the example provided, the extracted output, including cleaning, becomes: "*claims: "honduras bans citizens from owning guns and has the highest homicide rate in the entire world.", "switzerland with a similar population requires citizens to own guns and has the lowest homicide rate in the entire world.", evidence: "barack obama and hillary clinton are negotiating with the united nations about doing a treaty that will ban the use of firearms."*."

Dolly 2.0 [24] is a language model designed to generate text based on given prompts or instructions. We utilized in particular the 12B parameter language model version available on Hugging Face [6]. We instructed Dolly to articulate the claim and evidence by using the following prompt format: "*Please state the main claim made by the author in the following section of a news article:* ⟨*paragraph*⟩", inserting each input paragraph in ⟨*paragraph*⟩. We did this similarly for the evidence. We accommodated the maximum token length of 2048, including both instruction and text and the models' response, by segmenting texts into paragraphs of up to 400 words each, joining the responses to form a coherent input for the text classifier. From each paragraph, we temporarily save the output in a list. Once Dolly completes processing all paragraphs for a particular article, we join the claims and evidence extracted from each section. Using our above example Dolly's output is as follows: *"claims: "The author of the article stated that Honduras has banned citizens from owning guns." evidence: "The article states that Honduras banned citizens from owning guns stating that the ban was put into place due to the number of mass shootings that have occurred in the country. The article does not provide any evidence to support the claim that banning citizens from owning guns will reduce the number of mass shootings.""*

3.4 Pre-trained Language Models

We have considered four state-of-the-art pre-trained language models for the fine-tuning of the classifiers for the baseline and the argumentation-based pipelines: BERT [7], RoBERTa [19], DistilBERT [35], and ELECTRA [5]. We have chosen these because, following Khan et al. [13], they have proven to outperform other traditional and deep learning approaches with respect to robustness and accuracy. More precisely, we used BERT-base with 12 layers and 110 million parameters, RoBERTa-base with 6 layers and 125 million parameters, DistilBERT with 12 layers and 66 million parameters, ELECTRA-base with 12 layers and 100 million parameters.

3.5 Experiments

Pipelines For our baseline pipeline, we fine-tuned the four previously mentioned models using the preprocessed texts as input. To consider the relationship between the performance and text length, we categorized the texts in the test set into three categories: 0 up to 100 words, 100 up to 300 words, and 300 up to the

maximum length. We calculated the overall performance of the best performing model for each group.

For our argumentation-based pipeline, we prepared three distinct experiments for MARGOT and Dolly 2.0 individually. Each experiment focuses on different extracted components as input: the claims, the evidence, and the (argumentation) structure, combining claims and evidence, as illustrated in Section 3.3. For the text lengths, we utilize the similar three categories as in the baseline pipeline. We will evaluate the performance predictions utilizing the best model for every component for MARGOT and for Dolly 2.0 per category.

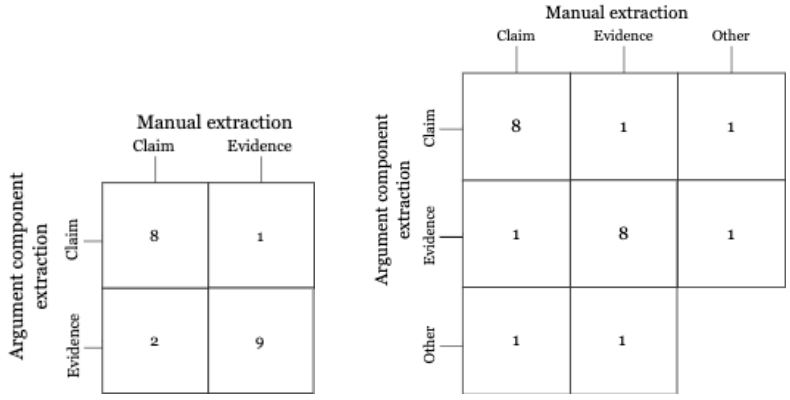
Experimental setup For training the language models, we adopted the parameter settings given by Khan et al. [13]. To tailor the classifiers for fake news detection, we fine-tuned for 10 epochs, utilizing a mini-batch size of 32. To prevent overfitting, we implemented early stopping using the validation loss with a delta value set to zero. For computing the loss, we employed binary cross-entropy. The maximum length for preprocessed texts was set to 300 to effectively accommodate these differences. Optimizing the models was facilitated by employing the AdamW optimizer with a learning rate of 4e-5. Additionally, we utilized B1 and B2 values of 0.9 and 0.99, respectively, while setting epsilon to 1e-8. Our experiments were conducted on the NVIDIA Snellius GPU provided by Surf [40].

3.6 Evaluations

Manual evaluation of argument extraction To have a rough assessment of the reliability of the argumentation extraction tools used in our study, we conducted a comparison between the claims and evidence extracted by MARGOT and Dolly 2.0 against manually extracted claims and premises for 10 news articles. The manual approach allows us also to overcome textual differences, present the claims extracted by MARGOT and by Dolly 2.0. For example, if a manually extracted claim states, *"Every time you buy an airline ticket, the federal government runs a background check on you"*, we can consider a claim such as *"The government is involved in air travel"* as a True Positive (TP) because it conveys the same key idea. If the system incorrectly labels this claim as evidence, it would be considered a False Negative (FN). We qualify following the same idea also True Negative (TN) and False Positive (FP) elements.

Model evaluation Following standard practice, we evaluated the model's performance with the validation set during the training phase; after training, in the evaluation phase, we used the test set. The performance of each model was captured in terms of accuracy, precision, recall, and F1 score.

In our classification task, we considered texts labeled as real the 'positive class', and texts labeled as fake the 'negative class'. Hence, True Positive (TP) represents the news that is actually real and correctly predicted as real, while False Positive (FP) refers to news that is false but incorrectly predicted as real. Similarly, True Negative (TN) and False Negative (FN) have their respective meanings in the context of the classification task.



(a) MARGOT (b) Dolly 2.0

Fig. 2: Evaluation of argument extraction tools via manual annotation.

4 Results

4.1 Manual evaluation of argument extraction

To have a rough estimation of the reliability of the argument extraction tools, we compared claims and evidence manually extracted with claims and evidence extracted by MARGOT and by Dolly 2.0 for 10 randomly selected news texts. The results are illustrated in Fig. 2. Overall, the results indicate that the argument component extraction performed by both MARGOT and Dolly 2.0 aligns well with manual extraction, with alignments ranging from 8/10 to 9/10 for both MARGOT and Dolly 2.0. However, during our evaluation, we encountered an important issue with Dolly 2.0: in one instance, the tool generated a complete new argument instead of extracting the claim and evidence from the given text.

4.2 Model performance

Baseline pipeline The performance of our baseline pipeline on the test set is presented in Table 2. RoBERTa achieves the highest F1 score (0.76). BERT attains the highest scores for precision (0.79), and accuracy (0.75) but lower scores for recall (0.66).

Argumentation-based pipeline with MARGOT Table 3 displays the performance results of fine-tuning the language models for text classification using as input the extracted *evidence*, *claim*, and *structure,* by MARGOT. Among the evaluated models, DistilBERT achieves the highest overall F1 scores. Specifically, DistilBERT reaches an F1 score of 0.84 for the *evidence* and *claim* components and 0.83 for using the *structure* as input. The highest performance was observed when using the *evidence* as input for the models. The F1 scores ranged from 0.74 for BERT to 0.84 for ELECTRA, RoBERTa, and DistilBERT. The extracted

	Accuracy	Precision	Recall	F1
BERT	**0.75**	**0.79**	0.66	0.72
RoBERTa	0.74	0.69	**0.86**	**0.76**
DistilBERT	0.73	0.70	0.79	0.74
ELECTRA	0.73	0.68	0.83	0.75

Table 2: Models' performance for the baseline pipeline

	Evidence				Claim				Structure			
	Acc.	Prec.	Rec.	F1	Acc.	Prec.	Rec.	F1	Acc.	Prec.	Rec.	F1
BERT	0.66	0.63	0.91	0.74	0.66	0.63	0.91	0.74	0.82	0.78	**0.95**	0.85
RoBERTa	**0.84**	0.90	0.79	**0.84**	0.66	0.62	**0.95**	0.75	0.64	0.62	0.90	0.73
DistilBERT	**0.84**	**0.92**	0.77	**0.84**	0.83	0.88	0.80	**0.84**	0.82	0.85	0.82	**0.83**
ELECTRA	0.80	0.75	**0.95**	**0.84**	0.82	**0.90**	0.76	0.82	**0.83**	**0.89**	0.79	0.84

Table 3: Models' performance on argumentation extracted by MARGOT

claim exhibited the lowest performance across the evaluated models, with F1 scores ranging from 0.72 to 0.84. However, the differences in performance between the argument components were minimal.

The highest-performing model for the *evidence* component is DistilBERT, yielding, in addition to its high F1 score, a precision score of 0.92. Regarding the *claim* component, while DistilBERT achieves the highest F1 score, ELECTRA demonstrates a precision score of 0.90, slightly higher than DistilBERT's precision of 0.88. Regarding the *structure* component, BERT achieves the highest overall F1 score of 0.85. However, DistilBERT surpasses BERT in precision, with a score of 0.85 compared to BERT's precision of 0.78.

Argumentation-based pipeline with Dolly 2.0 Table 4 presents the performance results of the argumentation extracted by Dolly 2.0. Similar to MARGOT, DistilBERT demonstrates the highest overall performance. Specifically, it achieves an F1 score of 0.76 for the *evidence* component, 0.75 for the *claim* component, and 0.72 for the *structure* component. The *evidence* argument component as input for the models yields the highest F1 scores, ranging from 0.72 for BERT to 0.76 for DistilBERT. The *structure* component consistently shows lower performance scores, ranging from 0.68 to 0.72.

When examining the *evidence* component, DistilBERT consistently achieves the highest performance across all metrics. For the *claim* component, RoBERTa exhibits a precision of 0.68, slightly surpassing DistilBERT with a precision of 0.64. Regarding the *structure* component, both ELECTRA and DistilBERT achieve the highest F1 score of 0.72. DistilBERT also achieves a slightly higher precision of 0.68, with a difference of only 0.01 compared to ELECTRA. However, DistilBERT has a lower recall by 0.02 compared to ELECTRA.

	Evidence				Claim				Structure			
	Acc.	Prec.	Rec.	F1	Acc.	Prec.	Rec.	F1	Acc.	Prec.	Rec.	F1
BERT	0.70	0.65	0.79	0.72	0.69	0.63	0.84	0.72	**0.72**	**0.71**	0.69	0.70
RoBERTa	**0.72**	**0.66**	0.85	0.74	**0.72**	**0.68**	0.80	0.74	0.71	**0.71**	0.65	0.68
DistilBERT	**0.72**	**0.66**	0.88	**0.76**	0.71	0.64	**0.91**	**0.75**	0.71	0.68	0.77	**0.72**
ELECTRA	0.68	0.62	**0.90**	0.73	0.69	0.63	0.86	0.73	0.71	0.67	**0.79**	**0.72**

Table 4: Models' performance on argumentation extracted by Dolly 2.0

4.3 Influence of text length

To study the effect of the text length on performance, we divided the texts in the test set into the categories: short (0 up to 100 words), medium (100 up to 300 words), and long (300 up to the maximum number of words in the dataset). We used the predictions of the best-performing model. For the baseline pipeline, we use RoBERTa. For MARGOT and Dolly 2.0, we use DistilBERT fine-tuned on the respective component from the tool.

Figure 3 displays the F1 scores of the best-performing baseline pipeline model and the best-performing models for each argument component extracted by MARGOT and Dolly 2.0, considering the lengths of texts in the test set. For MARGOT, the results indicate that the baseline pipeline performs similarly as fine-tuning on *claim* or *structure* for short texts. However, when it comes to medium and long texts, both the *evidence* and *structure* components exhibit significantly higher performance compared to the baseline and the *claim* component. Nevertheless, the model with the *claim* component as input appears more consistent for short and long texts. For Dolly 2.0, it is evident that the baseline outperforms both short, medium, and long texts. Regarding claims, there is a slight decline in performance for medium texts compared to short and long texts. The overall trend suggests that classification performance tends to improve for longer texts in both MARGOT, Dolly 2.0, and the baseline pipeline.

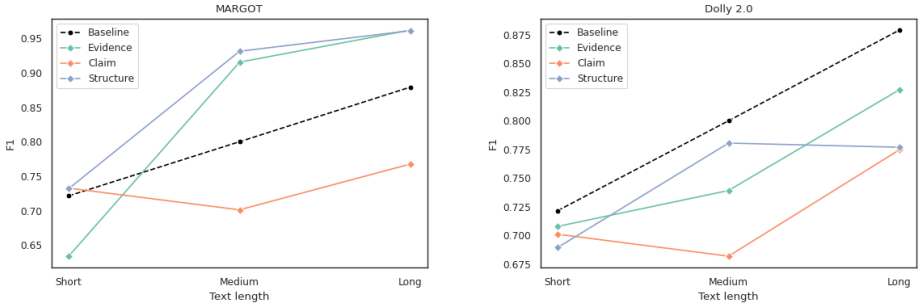


Fig. 3: Relation between best-performing model performance and text lengths

5 Discussion

Our original objective was to compare the performance of our baseline pipeline with the results from the benchmark study conducted by Khan et al. [13]. However, due to the difference in the dataset we eventually used, a direct comparison between the two studies is not possible. Nevertheless, an important observation from the results by Khan et al. [13] concerns the variation in performance across the different datasets. For instance, RoBERTa achieved an F1 score of 0.98 for the Fake or Real News dataset, but the performance was notably lower for the Liar dataset, with both BERT and RoBERTa achieving an F1 score of 0.62. The reason for lower performance was deemed to be the composition of the Liar dataset, consisting of short texts. We confirmed this during our preprocessing stage, prompting us to consider the lengths of texts as a factor in our analysis of the two distinct pipelines and with hindsight, our intuition was a correct one.

The performance of the models in the argumentation-based pipeline varied significantly between the two methods of argumentation extraction. When using the argument components extracted by MARGOT as input, the F1 scores were relatively high (Table 3), ranging from 0.73 to 0.84. Among all the models, DistilBERT consistently achieved the highest performance. Additionally, when considering the individual components, DistilBERT also performed the best, with the highest F1 score observed for the *evidence* component. For Dolly 2.0, the results were notably lower (Table 4), with F1 scores ranging from 0.68 to 0.76. Like with MARGOT, DistilBERT demonstrates the highest overall performance among all the models. DistilBERT also outperforms with the *evidence* component compared to the other components. The notably lower score for Dolly 2.0 may be due to errors like the one we observed during the manual evaluation, when the system generated a completely unrelated argument component; further investigation is however needed to confirm this hypothesis.

The reason why both MARGOT and Dolly 2.0 perform the highest for the *evidence* component could be attributed to the fact that the extracted evidence tends to consist of longer texts compared to the extracted claims. Yet, there may be content-related aspects involved; as the evidence acts as premise that enables the reader to infer the conclusion, their "quality" may be a good proxy of the soundness of the overall text.

Interestingly, even with this small subset of the original datasets, the performance of the models on all the argument components extracted by MARGOT consistently exceeds the F1 performance of the baseline models. Considering the lengths of texts as a factor, the most significant performance improvement occurs when fine-tuning DistilBERT for the *evidence* argument component extracted by MARGOT, as illustrated in Fig. 3. Vice-versa, the performance of Dolly 2.0 is always either comparable or slightly lower than our baseline.

6 Conclusion

The performance of the argumentation-based pipeline, particularly when considering the text length, shows promise in enhancing the accuracy of fake news clas-

sification compared to the baseline pipeline. However, the choice of the specific argument component extraction method, such as MARGOT or Dolly 2.0, significantly influences the performance outcomes. While MARGOT demonstrates higher overall performance in our study, it also requires more extensive cleaning of the texts, compared to Dolly 2.0. This aspect may limit its effectiveness when applied to for example social media texts, often characterized by unstructured and poorly formatted sentences. In addition to the observations regarding MARGOT and Dolly 2.0, it is worth considering other factors that can impact the overall performance enhancement, such as the dataset size. Additionally, the generalizability of the argumentation-based pipeline to other domains or topics beyond politics could be explored, as this study primarily focused on political news datasets.

Limitations Our original intention to compare the models' performance for our baseline pipeline to the results from the benchmark study by Khan et al. [13] was not feasible due to time constraints. The down-sampling we applied was necessary due to the computational power and time involved in the claims and evidence extraction process using MARGOT and Dolly 2.0. On average, using a single thread, the extraction process took approximately 100 texts per hour for MARGOT and 150 texts per hour for Dolly 2.0, with the processing time varying depending on the length of the texts.

A second limitation of our study also relates to the generalizability of our results. As discussed in Section 2, previous studies have highlighted the topic sensitivity of fake news detection in general and argument component extraction in particular. However, due to the unavailability of labeled datasets for non-political topics, we could not assess the generalizability of our findings to other subject areas. Therefore, it is important to acknowledge that our results may be specific to the political domain and may not necessarily extend to other topics.

A third limitation was the great difficulty of finding available (public and usable with current computational infrastructures) tools for extracting argument components. We hope our study to be an additional support for further advancement in the development of such tools.

Future research We believe there are several promising directions for future research to explore. First, our work could be replicated on the full dataset instead of a sample, allowing more generalizable results. In addition, while the argument components extracted by Dolly 2.0 were erroneous, generative models remain potentially valuable for argumentation extraction. MARGOT's sentence-level classification is vulnerable to more implicit argument components and limits its effectiveness when applied to social media texts, often characterized by unstructured and poorly formatted texts. An alternative may be a combination of detecting on a sentence level, with, for example, MARGOT, and using a generative model, like Dolly, to extract from the paragraph level. Another option would be to apply Dolly to summarize and correct the sentences provided by MARGOT. Potentially resulting in a more robust method for long texts and higher results for short texts.

References

1. Al-Khatib, K., Wachsmuth, H., Hagen, M., Stein, B.: Patterns of Argumentation Strategies across Topics. EMNLP 2017 - Conference on Empirical Methods in Natural Language Processing, Proceedings pp. 1351–1357 (2017)
2. Bisallion, C.: Fake and real news dataset, `https://www.kaggle.com/datasets/clmentbisaillon/fake-and-real-news-dataset`
3. Cabrio, E., Villata, S.: Generating Abstract Arguments: a Natural Language Approach. Frontiers in Artificial Intelligence and Applications **245**(1), 454–461 (2012)
4. Chakrabarty, T., Hidey, C., Muresan, S., Mckeown, K., Hwang, A.: AMPERSAND: Argument Mining for PERSuAsive oNline Discussions (2020)
5. Clark, K., Luong, M.T., Brain, G., Le Google Brain, Q.V., Manning, C.D.: Electra: pre-training text encoders as discriminantors rather than generators (2020), `https://github.com/google-research/`
6. Databricks Inc.: Dolly 2.0 12b, `https://huggingface.co/databricks/dolly-v2-12b`
7. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. NAACL HLT 2019 - 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies - Proceedings of the Conference **1**, 4171–4186 (2018)
8. van Eemeren, F.H., Garssen, B., Krabbe, E.C., Snoeck Henkemans, A.F., Verheij, B., Wagemans, J.H.: Handbook of argumentation theory. Springer Netherlands (2014)
9. van Eemeren, F.H., Verheij, B.: Argumentation Theory in Formal and Computational Perspective. IFCoLog Journal of Logics and Their Applications **4** (2017)
10. Flamino, J., Galeazzi, A., Feldman, S., Macy, M.W., Cross, B., Zhou, Z., Serafino, M., Bovet, A., Makse, H.A., Szymanski, B.K.: Political polarization of news media and influencers on Twitter in the 2016 and 2020 US presidential elections. Nature Human Behaviour **7**(6), 904–916 (2023)
11. Hu, G., Ding, Y., Qi, S., Wang, X., Liao, Q.: Multi-depth Graph Convolutional Networks for Fake News Detection. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics) **11838 LNAI**, 698–710 (2019)
12. Kaliyar, R.K., Goswami, A., Narang, P.: FakeBERT: Fake news detection in social media with a BERT-based deep learning approach. Multimedia tools and applications **80**(8), 11765–11788 (2021)
13. Khan, J.Y., Khondaker, M.T.I., Afroz, S., Uddin, G., Iqbal, A.: A benchmark study of machine learning models for online fake news detection. Machine Learning with Applications **4**, 100032 (2021)
14. Lawrence, J., Reed, C.: University of Dundee Argument Mining Using Argumentation Scheme Structures Argument Mining Using Argumentation Scheme Structures. Frontiers in Artificial Intelligence and Applications **287**, 379–390 (2016)
15. Lawrence, J., Reed, C.: Argument Mining: A Survey. Computational Linguistics **45**(4), 765–818 (2019)
16. Lee, T.: The global rise of "fake news" and the threat to democratic elections in the USA. Public Administration and Policy **22**(1), 15–24 (2019)
17. Li, C., Goldwasser, D.: Encoding Social Information with Graph Convolutional Networks forPolitical Perspective Detection in News Media. ACL 2019 - 57th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference pp. 2594–2604 (2019)

18. Lippi, M., Torroni, P.: Context-Independent Claim Detection for Argument Mining. Twenty-Fourth International Joint Conference on Artificial Intelligence (2015)
19. Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., Stoyanov, V., Allen, P.G.: RoBERTa: A Robustly Optimized BERT Pretraining Approach (2019)
20. Mathew, B., Dutt, R., Goyal, P., Mukherjee, A.: Spread of Hate Speech in Online Social Media. WebSci 2019 - Proceedings of the 11th ACM Conference on Web Science pp. 173–182 (2019)
21. McIntire, G.: Fake or Real News (2017), `https://github.com/GeorgeMcIntire/fake\_real\_news\_dataset`
22. Mercier, H.: The Argumentative Theory: Predictions and Empirical Evidence. Trends in Cognitive Sciences **20**(9), 689–700 (2016)
23. Mercier, H., Sperber, D.: Why do humans reason? Arguments for an argumentative theory. The Behavioral and brain sciences **34**(2), 57–74 (2011)
24. Mike Conover, Matt Hayes, Ankit Mathur, Xiangrui Meng, Jianwei Xie, Jun Wan, Sam Shah, Ali Ghodsi, Patrick Wendell, Matei Zaharia, Reynold Xin: Free Dolly: Introducing the World's First Open and Commercially Viable Instruction-Tuned LLM (2023)
25. Moschitti, A.: Tree Kernels in SVM-light, `http://disi.unitn.it/moschitti/Tree-Kernel.htm`
26. NLTK: Natural Language Toolkit, `https://www.nltk.org/`
27. Oshikawa, R., Qian, J., Wang, W.Y.: A Survey on Natural Language Processing for Fake News Detection. LREC 2020 - 12th International Conference on Language Resources and Evaluation, Conference Proceedings pp. 6086–6093 (2018)
28. Pennycook, G., Rand, D.G.: The Psychology of Fake News. Trends in Cognitive Sciences **25**(5), 388–402 (2021)
29. Pérez-Rosas, V., Kleinberg, B., Lefevre, A., Mihalcea, R.: Automatic Detection of Fake News. COLING 2018 - 27th International Conference on Computational Linguistics, Proceedings pp. 3391–3401 (2017)
30. Petasis, G., Karkaletsis, V.: Identifying Argument Components through TextRank. Proceedings of the 3rd Workshop on Argument Mining pp. 94–102 (2017)
31. Piazza, J.A.: Fake news: the effects of social media disinformation on domestic terrorism. Dynamics of Asymmetric Conflict: Pathways toward Terrorism and Genocide **15**(1), 55–77 (2021)
32. Rashkin, H., Choi, E., Jang, J.Y., Volkova, S., Choi, Y.: Truth of Varying Shades: Analyzing Language in Fake News and Political Fact-Checking. EMNLP 2017 - Conference on Empirical Methods in Natural Language Processing, Proceedings pp. 2931–2937 (2017)
33. Raza, S., Ding, C.: Fake news detection based on news content and social contexts: a transformer-based approach. International Journal of Data Science and Analytics **13**(4), 335–362 (2022)
34. Ruchansky, N., Seo, S., Liu, Y.: CSI: A Hybrid Deep Model for Fake News Detection. Proceedings of the 2017 ACM on Conference on Information and Knowledge Management pp. 797–806 (2017)
35. Sanh, V., Debut, L., Chaumond, J., Wolf, T.: DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter
36. Sap, M., Shwartz, V., Bosselut, A., Choi, Y., Roth, D., Allen, P.G.: Commonsense Reasoning for Natural Language Processing pp. 27–33 (2020)
37. Shu, K., Cui, L., Wang, S., Lee, D., Liu, H.: dEFEND: Explainable Fake News Detection. The 25th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD '19) **4**(8) (2019)

38. SpaCy: Contextual Spell Check, spaCy Universe, https://spacy.io/universe/project/contextualSpellCheck
39. Stanford NLP Group: CoreNLP, https://stanfordnlp.github.io/CoreNLP/
40. Surf: Dutch National Supercomputer Snellius vert SURF.nl, https://www.surf.nl/en/dutch-national-supercomputer-snellius
41. The Poyner Institute: PolitiFact, https://www.politifact.com/
42. Wang, W.Y.: "Liar, liar pants on fire": A new benchmark dataset for fake news detection. ACL 2017 - 55th Annual Meeting of the Association for Computational Linguistics, Proceedings of the Conference (Long Papers) **2**, 422–426 (2017)
43. Yan, M., Lin, Y.R., Litman, D.J.: Argumentatively Phony? Detecting Misinformation via Argument Mining; Argumentatively Phony? Detecting Misinformation via Argument Mining. ACM Transactions on Graphics **37**(4) (2018)
44. Yang, S., Shu, K., Wang, S., Gu, R., Wu, F., Liu, H.: Unsupervised Fake News Detection on Social Media: A Generative Approach. Proceedings of the AAAI Conference on Artificial Intelligence **33**(01), 5644–5651 (2019)
45. Yang Wang, W.: Liar: a benchmark dataset for fake news detection, https://github.com/tfs4/liar_dataset
46. Zhou, Z., Guan, H., Bhat, M., Hsu, J.: Fake news detection via NLP is vulnerable to adversarial attacks. Proceedings of the 11th International Conference on Agents }and Artificial Intelligence (2019)
47. Zuckerman, E.: Mistrust, efficacy and the new civics: understanding the deep roots of the crisis of faith in journalism. Knight Commission Workshop on Trust, Media and American Democracy (2017)