LLM-based Fact-Checking: A Pipeline for Studying Information Disorder*

Gabriele Fioretti[1], Lorenzo Goglia[1] and Eugenio Zimeo[1,*]

[1]University of Sannio, 1 Via Traiano, Benevento, 82100, Italy

Abstract

Today, the way of accessing and interpreting information has changed profoundly compared to the past. People are flooded with large amounts of content at very high speed, leaving limited time to verify its truthfulness, implicitly creating improper trust with content sources. Therefore, the impact of fake news, spread maliciously or otherwise, is significantly amplified, with potential moral, physical, or economic damages. Fact-checking, *i.e.*, the process of verifying the factual accuracy of a statement, represents a possible solution to address this problem. Due to its complexity, this task is usually performed by humans, but with the increasing amount of sources, it is becoming expensive in terms of both resources (*i.e.*, people involved) and amount of work required (*i.e.*, time spent to interpret a statement and find supporting evidence). For these reasons, automated fact-checking approaches have started to appear in the literature. Given the nature of the task, recently, attention has been paid to approaches that leverage Large Language Models. In this paper, we present a big data processing pipeline that aims to mark statements coming from different sources with an appropriate level of truthfulness. The pipeline includes an LLM-based fact checker whose goal is to automate the verification process by exploiting the ability of LLMs to analyze text by performing a comparison with their knowledge base to evaluate the truthfulness of the analyzed facts. According to other recent proposals, we show that a zero-shot adoption of an LLM could lead to limited precision and recall if compared with an oracle; however, by conducting a more in-depth analysis of the problem and considering the different truthfulness semantics, we identify room for further analyses to be conducted in a dedicated pipeline stage which can also exploit relationships among statements and with authors. The idea is to complete the pipeline with a graph-based analyzer that aims to improve the results obtained only with the LLM, especially when the confidence level is not high.

Keywords

Information disorder, Fact checking, Processing pipeline, Graphs, LLM

1. Introduction

Information disorder refers to the processes of sharing genuine information with bad intentions, *i.e.*, mal-information, or false information with or without bad intentions, *i.e.*, dis-information or misinformation, respectively [1]. In a society like ours, characterized by an extremely easy and pervasive access to information, the impact of such processes is significant: the diffusion of fake news can have more or less serious consequences, ranging from gossip that can harm individuals to political or economic facts that can be harmful to an entire community. Unfortunately, people tend to establish the truthfulness of content by relying on the feelings and beliefs they have with respect to its topic in the specific moment when they come into contact with it, without worrying about verifying the sources or finding evidence.

In the last few years, there has been a proliferation of fact-checking websites[1] that can be used to verify whether the news is reliable or not. However, such portals are maintained by people who are in charge of manually analyzing news. Consequently, this task is resource- and time-consuming and is not suitable to cope with the huge volume of news generated and the frequency and velocity with which they propagate [2]. Diverse approaches trying to automate fact-checking processes have been proposed in the literature (see Sec. 2). Among them, those based on Large Language Models (LLMs), specifically on Instruction-following Language Models (IFLMs), are gaining interest. Specifically, these

Joint National Conference on Cybersecurity (ITASEC & SERICS 2025), February 03-8, 2025, Bologna, IT

*This paper has been partially supported by the IDA project within the SERICS framework.

*Corresponding author.

g.fioretti@studenti.unisannio.it (G. Fioretti); logoglia@unisannio.it (L. Goglia); zimeo@unisannio.it (E. Zimeo)

[1]For a complete list of fact-checking websites, please refer to https://en.wikipedia.org/wiki/List_of_fact-checking_websites

solutions ask a model (*i.e.*, ChatGPT) for evaluating the level of truthfulness of a claim or a statement according to a dichotomous or multi-valued rating scale. It is known that the performance of these tools strongly depends, among other things, on the way the execution of a specific task is requested, the so-called prompt, and the amount and quality of contextual information included in it [3, 4].

In this paper, we present and discuss the results obtained by an automated fact-checker based on Gemma-2 (*gemma2.9b*[2]). Through their in-depth analysis, we identify room for further elaborations to be conducted in a dedicated pipeline stage, which can exploit the relationships among statements and with authors to improve the performance in discerning reliable content from unreliable ones. To this end, we propose a pipeline equipped with a graph-based analyzer that aims to improve the classification results obtained downstream of the interaction with the LLM, especially when the confidence level is not high.

The pipeline is able to ingest statements coming from different sources (*e.g.*, social networks, websites, files etc.) and classify them properly, thus enabling the study of different phenomena related to information disorder, such as network polarization, echo chamber and filter bubbles [5], or to analyze propagation patterns [6, 7] and diffusion processes [8].

The rest of the paper is organized as follows. Sec. 2 positions the work with respect to literature. In Sec. 3 we present the experimentation conducted to evaluate the performance of the fact-checking module, with the results obtained. The latter are deeply analyzed and discussed in Sec. 4, whereas in Sec. 5 we describe the conceived pipeline. Finally, Sec. 6 concludes the work by highlighting possible future directions.

2. Related Works

Automated fact-checking is gaining more and more attention as a means to try to face phenomena related to information disorder. In general, a fact-checking pipeline is composed of the following steps: *i)* claim detection, *ii)* evidence retrieval, *iii)* verdict prediction, and *iv)* justification production [9]. The main ones are the second and the third, which are about retrieving possible relevant evidence supporting the claims and then using it to generate the labels that represent the level of truthfulness, respectively. Recently, these tasks have been performed with LLMs, specifically IFLMs.

In [10], authors evaluated GPT-3.5 and GPT-4 on a specialized dataset, *i.e.*, PolitiFact[3], and by giving the two models the possibility to retrieve contextual data via queries to Google Search engine. They found that *i)* incorporating contextual information significantly improves accuracy (above 80% for non-ambiguous verdicts), and *ii)* the performance depends on veracity, with false statements being better identified than true ones. In [11], authors evaluated GPT-3.5, GPT-4, Google's Bard/LaMDA, and Microsoft's Bing AI in discerning the truthfulness (by using a three-valued scale (*i.e.*, true, false, partially true/false)) of 100 news fact-checked items retrieved from independent fact-checking agencies. Although on a very small dataset, the results showed a good average accuracy ($\approx 65\%$), with GPT-4 being the best alternative. In [12], authors evaluated GPT-3.5 in classifying verified statements collected from PolitiFact. In particular, they used two different prompts instructing the model to categorize the statements according to binary and multi-valued scales. On average, across the two prompts, they observed an accuracy of 69%. In [13], authors evaluated the ability of diverse variants of GPT-based and BERT-based models to classify claims from ClaimBuster dataset [14] as check-worthy or not using various methodologies, including zero-shot and few-shot learning, along with fine-tuning techniques. Their findings indicate that fine-tuned BERT models can perform comparably to GPT models with accuracy values ranging from $\approx 75\%$ to $\approx 93\%$ and F1-score values ranging from $\approx 73\%$ to $\approx 90\%$. In [15], authors evaluated the fact-checking capabilities of LLaMA, with and without contextual information, on two datasets: RAWFC [16] and LIAR [17]. In particular, they observed F1-score values of $\approx 55\%$ on RAWFC and $\approx 30\%$ on LIAR.

[2]https://blog.google/technology/developers/google-gemma-2/
[3]https://www.politifact.com/

In all the aforementioned works, the rationale behind performance improvement attempts is *i)* to provide as much contextual information as possible in terms of evidence and/or *ii)* to avoid using zero-shot learning approaches by adopting fine-tuning strategies. In this work, instead, we exploit the results deriving from the evaluation of an LLM-based fact-checker to highlight the possibility of improving performances by exploiting the relationships between statements and with the authors synthesized in a property graph, *i.e.*, an attributed knowledge graph. This, in turn, leads to our proposal of a processing pipeline for leveraging such a property graph, which is built starting from the collected statements, to increase fact-checking capabilities.

3. LLM-based Fact-checking

LLM-based fact-checking is characterized by the use of an LLM, specifically an IFLM, to mark a given statement as truthful or not (with binary or fuzzy variants). In this work, we considered Gemma-2 (*gemma2:9b*), and we evaluated it on the LIAR dataset (see Sec. 3.1). As already pointed out, the prompt is one of the crucial elements when working with LLMs. For this reason, in our performance evaluation, we conducted different tests with diverse prompts, built by exploiting best practices known from the literature (see Sec. 3.2). Prompt engineering is indeed out of the scope of this paper. The evaluation we present in this section is preparatory to the pipeline we propose in Sec. 5.

In the following, we describe the whole experimentation, starting from the dataset used, going through the prompts employed, and ending with the results obtained.

3.1. Dataset

The selected model has been evaluated on the LIAR dataset, a popular dataset that contains manually fact-checked political statements that have been labeled based on their level of truthfulness according to a multi-valued scale. In particular, a statement can be: *i)* TRUE, *i.e.*, the statement is accurate and there's nothing significant missing; *ii)* MOSTLY TRUE, *i.e.*, the statement is accurate but needs clarification or additional information; *ii)* HALF TRUE, *i.e.*, the statement is partially accurate but leaves out important details or takes things out of context; *iv)* BARELY TRUE, *i.e.*, the statement contains an element of truth but ignores critical facts that would give a different impression; *v)* FALSE, *i.e.*, the statement is not accurate; *vi)* PANTS ON FIRE, *i.e.*, the statement is not accurate and makes a ridiculous claim.

The dataset contains 12791 statements. For each of them, it provides the following information: the statement ID, the truthfulness label, the statement text, the topics associated with the statement, the speaker's name, the speaker's professional title, state information, the speaker's political affiliation, the total counts of historical credibility ratings, and the context (location or venue of the speech or statement). Among them, we only consider the statement text, which is submitted to the LLM, and the label, which is used as an oracle for performance evaluation.

3.2. Prompt

As already mentioned, the way users ask an IFLM for executing a given task profoundly impacts performances. For the purposes of our analysis, we used two different prompts, one for asking Gemma-2 to classify the statements according to a multi-valued rating scale compliant with that used in the dataset (see Sec. 3.1), and another one for asking Gemma-2 to classify the statements according to a binary scale (true or false). Both have been designed to provide the model with contextual information about the semantics of each label. Basically, truthfulness is defined based on the availability of evidence supporting or contradicting a statement and the accuracy of the statement itself. Note that the model is always allowed to perform search queries for retrieving possible evidence to use for evaluation.

The first prompt appears as follows:

You're a fact-checking expert. Evaluate the truthfulness of this statement: '{statement}'. Use your information and explore the link that may be contained in the statement. Consider your sources and context while assessing. To answer, return ONLY a JSON format string containing "verdict", "reason" and "sources". The "verdict" must be one from TRUE (The statement is accurate, supported by evidence, and there's nothing significant missing), MOSTLY-TRUE (The statement is accurate, it is mostly supported by evidence, but there's something missing/needs clarification), HALF-TRUE (The statement is partially accurate, partially supported by evidence, but takes things out of context and needs clarification), BARELY-TRUE (The statement lacks evidence, is barely supported by evidence, contains an element of truth but ignores critical facts that would give a different interpretation), FALSE (The statement is not accurate and not supported by evidence) or PANTS-ON-FIRE (it makes claims that are strongly influenced by personal opinion and there is evidence that is against it). "reason" should contain the reasons that brought you to give that verdict. In "sources" you have to note the sources you used for your research.

The second prompt, instead, looks as follows:

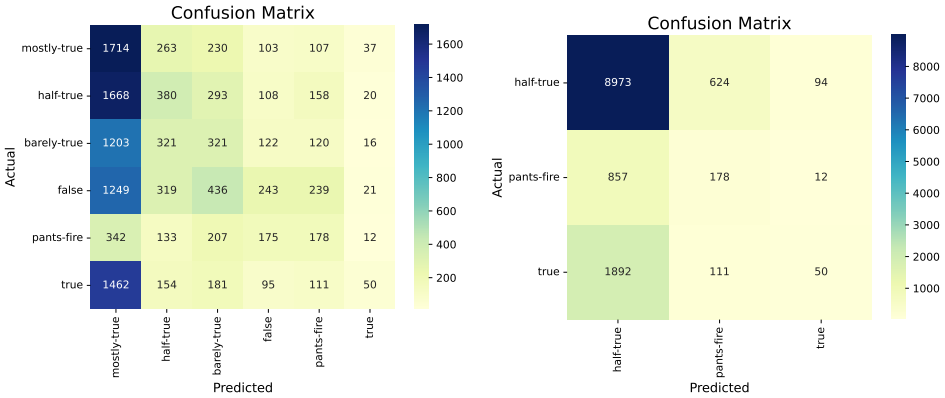
You're a fact-checking expert. Evaluate the truthfulness of this statement: '{statement}'. Use your information and explore the link that may be contained in the statement. Consider your sources and context while assessing. To answer, return ONLY a JSON format string containing "verdict", "reason" and "sources". The "verdict" must be one from TRUE (The statement is accurate, supported by evidence, and there's nothing significant missing) or FALSE (The statement is not accurate and not supported by evidence). "reason" should contain the reasons that brought you to give that verdict. In "sources" you have to note the sources you used for your research.

In both cases, *{statement}* is a placeholder which is replaced with the statement contained in the record to be verified. As one can note, we also ask the model to motivate its choice with a justification and the list of sources it used.

3.3. Results

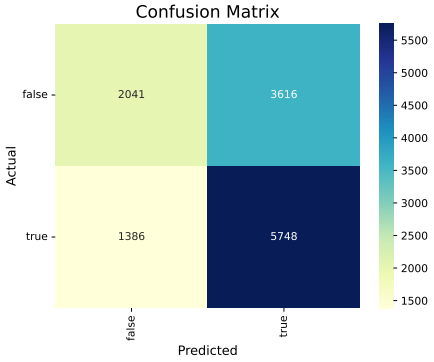
The performances of our LLM-based fact-checker are evaluated via standard metrics, such as accuracy, precision, recall, and F1-score. The latter are summarized in Tab. 1, while in Fig. 1 we report the corresponding confusion matrices. In particular, for an in-depth analysis, we considered four different settings: *i) multi-valued scale* (Fig. 1a and Tab. 1a), in which we check for an exact match between the label provided in the dataset and that generated by the fact-checker; *ii) three-valued binarization* (Fig. 1b and Tab. 1b), in which we group together all intermediate labels (*i.e.*, MOSTLY-TRUE, HALF-TRUE, BARELY-TRUE, and FALSE), while keeping the most extreme ones as in the previous case; *iii) two-valued binarization* (Fig. 1c and Tab. 1c), in which we put TRUE, MOSTLY-TRUE, and HALF-TRUE in one group, and BARELY-TRUE, FALSE, and PANTS-ON-FIRE in another group; *iv) dichotomous scale* (Fig. 1d and Tab. 1d), in which we ask the model for an exact match between the label provided in the dataset and that generated by the fact-checker, but only for TRUE and FALSE. In the latter case, the second prompt has been employed.

In the first configuration, performance does not seem to be satisfactory, with very low metric values, except for the recall in the case of MOSTLY-TRUE class. However, this may be due to the fact that MOSTLY-TRUE is by far the most predicted label and not to the ability of the model to identify that class correctly. What captures the attention is the presence of TRUE statements labeled as PANTS-ON-FIRE, and vice versa, with the first case being more frequent than the second. The manual analysis of some statements suggests that *i)* the model has not fully understood the semantics of the labels and *ii)* there are statements judged by humans as truthful or not because of some particular interpretation of the context that goes beyond the presence of evidence and facts. For example, the statement *"President Obama, Sen. Harry Reid and Rep. Nancy Pelosi passed a $1.2 trillion stimulus bill."* has been fact-checked as PAINT-ON-FIRE, but the model marks it as TRUE with the following justification: *"The statement is accurate. The American Recovery and Reinvestment Act of 2009, a $787 billion stimulus package, was signed into law by President Obama in February 2009. Key figures in its passage included Senate Majority Leader Harry Reid and House Speaker Nancy Pelosi. While the stated amount of $1.2 trillion is higher than the actual cost, it reflects common public perception and may be a result of rounding or inflation."*. The

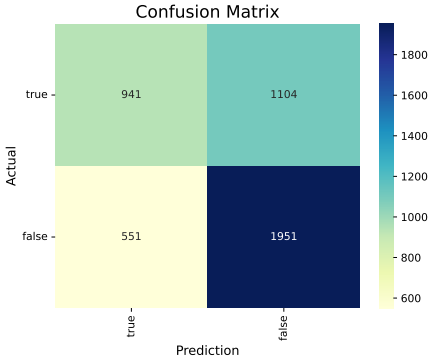


(a) Multi-valued scale.

(b) Three-valued Binarization.



(c) Two-valued Binarization.



(d) Dichotomous scale.

Figure 1: Confusion matrices.

statement *"Hes the only candidate whos balanced budgets and brought jobs to Providence."* has been fact-checked as MOSTLY-TRUE, but the model marks it as PAINTS-ON-FIRE, with the following justification: *"This statement makes a bold claim without providing any evidence or specifics. It's impossible to verify the truthfulness of such a general statement without concrete examples of balanced budgets and job creation attributed to this candidate. Furthermore, the use of vague language like 'balanced budgets' and 'brought jobs' leaves room for significant interpretation and potential exaggeration."*. Finally, the statement *"Iran President Hassan Rouhani has more Cabinet members with Ph.D.s from American universities than members of Barack Obamas Cabinet."* has been fact-checked as TRUE, but the model marks it as PAINTS-ON-FIRE, saying that: *"The statement is impossible to verify and relies on unsubstantiated claims. There's no reliable public information comparing the educational backgrounds of Iranian cabinet members under President Rouhani with those of Barack Obama's cabinet. It also makes a sweeping generalization about an entire group of people based on their nationality or alma mater, which is problematic."*. Sometimes, the model tries to interpret the statement (like in the first and last examples); sometimes, it assumes a more objective behaviour by giving much more importance to the presence/absence of evidence (like in the second case).

Performance slightly improves when we move to the second and third configurations, in which we relax the constraints related to the semantics of the labels. However, the meaning assigned to the labels grouped is not the same as that used by the model to mark the statements. This aspect was taken into account in the last configuration, in which we observed a significant improvement, although metric values are not very high in absolute terms.

Table 1
Confusion matrices

Class	Precision	Recall	F1-Score	Support
mostly-true	0.22	0.70	0.34	2454
half-true	0.24	0.14	0.18	2627
barely-true	0.19	0.15	0.17	2103
false	0.29	0.10	0.14	2507
pants-fire	0.19	0.17	0.18	1047
true	0.32	0.02	0.05	2053
Accuracy			0.23	12791
Macro Avg	0.24	0.21	0.18	12791
Weighted Avg	0.25	0.23	0.18	12791

(a) Metrics with Multi-valued scale

Class	Precision	Recall	F1-Score	Support
half-True	0.77	0.93	0.84	9691
pants-Fire	0.19	0.17	0.18	1047
true	0.32	0.02	0.05	2053
Accuracy			0.72	12791
Macro Avg	0.43	0.37	0.35	12791
Weighted Avg	0.65	0.72	0.66	12791

(b) Metrics with Three-valued Binarization

Class	Precision	Recall	F1-Score	Support
false	0.60	0.36	0.45	5657
true	0.61	0.81	0.70	7134
Accuracy			0.61	12791
Macro avg	0.60	0.58	0.57	12791
Weighted avg	0.61	0.61	0.59	12791

(c) Metrics with Two-valued Binarization

Class	Precision	Recall	F1-Score	Support
true	0.63	0.46	0.53	2045
false	0.64	0.78	0.70	2502
Accuracy			0.64	4547
Macro Avg	0.63	0.62	0.62	4547
Weighted Avg	0.64	0.64	0.63	4547

(d) Metrics with dichotomous scale

4. Discussion

From the analysis of the literature (see Sec. 2), it emerges that LLMs do not give satisfying results in accomplishing fact-checking tasks. However, the results obtained in the different works cannot be easily compared since the semantics assigned to the labels used to express the truthfulness level are different, as are the prompts employed. Therefore, this work was motivated by the need to better understand the motivations behind those poor results in order to leave to LLMs only decisions related to more accurate trustfulness semantics (e.g., meaning of labels). Our results, in fact, show that performances improve when focusing only on TRUE and FALSE statements (with a specific semantic) and querying the model to distinguish between these cases. This is likely attributable to the model's limited ability to clearly differentiate between similar truth labels, such as TRUE and MOSTLY-TRUE, while it finds it easier to classify statements with significantly different truth values, such as TRUE and FALSE. To summarize, we observed that *i)* performances depend on veracity, and *ii)* sometimes the model fails[4] because either it has evidence supporting a statement, but it is not able to capture subjective dimensions (*e.g.*, intentions), it has evidence, but it cannot express a confident judgment because it concludes that the statement is biased and non-objective, or it has no evidence. Despite this, it is important to keep in mind that classifying the truthfulness value of a statement is a task that can sometimes be difficult to perform, even for humans.

In light of what has just been said, in the next section, we propose a processing pipeline to ingest and store labeled statements in a graph-oriented data store with the aim of improving the fact-checking step performed via the LLM through the use of a different kind of contextual information, specifically that concerning the relationships between the statements and with their authors. For example, a statement labeled with a low confidence level may be compared with similar statements, *i.e.*, those related to similar topics, or the information about the tendency of its author to propagate content with specific values of truthfulness may be exploited.

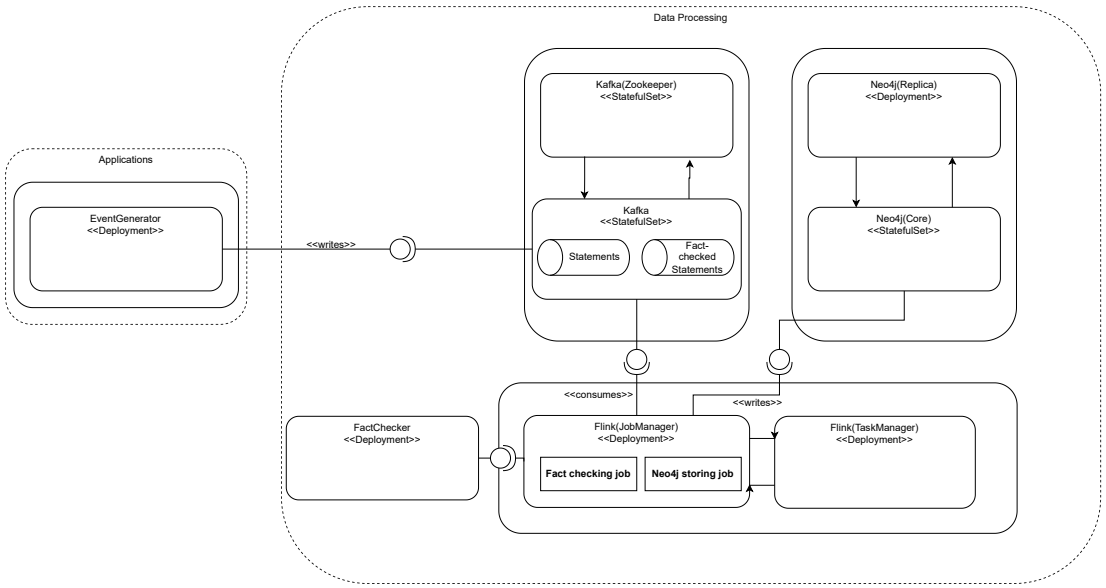


Figure 2: Pipeline architecture

5. Information disorder awareness pipeline architecture

To study information disorder with the aim of improving its awareness, we moved towards the integration of fact-checking in a processing pipeline that can be feed with statements by different sources. The statements flowing through the pipeline are marked by an LLM-based fact checker and propagated towards a graph DB, where they are connected to other statements and authors.

The pipeline (called IDA pipeline) has been designed to timely ingest and label large amounts of statements. We present this pipeline from both a structural (Sec. 5.1) and dynamic point of view (Sec. 5.2).

5.1. IDA pipeline components

The pipeline is composed of the following elements (see Fig. 2): *i)* a *Stream Broker* (SB), *ii)* a *Stream Processor* (SP), *iii)* a *Fact Checker* (FC), and *vi)* a *Graph-oriented DBMS* (GDB).

SB is a mediation component that enables asynchronous communication between all other components as the transitions among the diverse processing steps occur. It is implemented as a distributed asynchronous producer-consumer queue. In particular, the popular Apache Kafka[5] (and its discovery service Zookeeper) is used. SB receives raw statements from diverse sources (*e.g., social network APIs*), together with those marked by FC. Both are temporarily stored in two dedicated topics.

SP is responsible for ensuring low-latency processing of continuous high-throughput data streams. In this case, Apache Flink[6] is used. In particular, SP has two registered jobs: the former consumes messages from the Kafka topic hosting raw statements, marks them, and publishes new enriched messages on a different Kafka topic; the latter consumes the processed messages by performing ad-hoc queries to create entity (*e.g.*, statements, authors) and relationships (*e.g.*, creation, sharing), together with their properties (*e.g.*, label).

The first Flink job asks a fact-checking module, FC, for evaluating the level of truthfulness of statements in order to mark them with a pre-defined set of labels. In particular, FC exploits an LLM (*e.g.*, Gemma.2) provided via the Ollama platform [7].

[4]We assume that human-based fact-checking is free from errors and biases.
[5]https://kafka.apache.org/
[6]https://flink.apache.org/
[7]https://ollama.com/

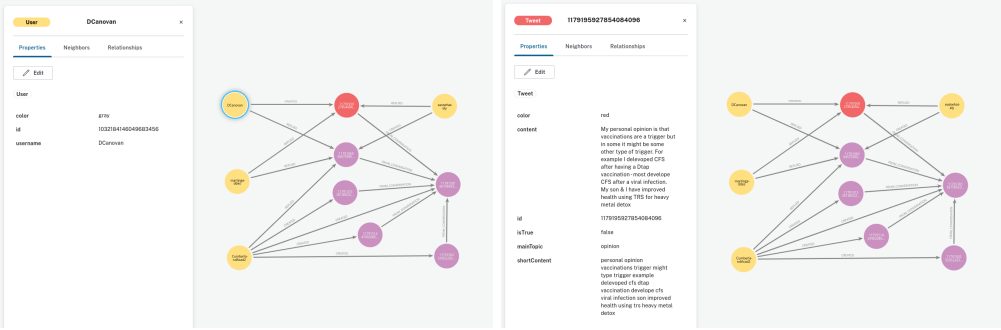


Figure 3: Example of a graph related to a set of marked Tweets. The red one is FALSE, whereas the other ones are UNKNOWN.

The pipeline ends with the GDB, which is implemented with Neo4j[8], a popular graph-oriented DBMS. In this way, information can be manipulated via an API based directly on concepts related to graphs.

In Fig. 2, there is an additional component, the *Event Generator* (EG), that is used to inject the statements into the pipeline, specifically to publish messages into the dedicated topic. It has been introduced due to the limitations of the majority of the current APIs that could be potentially used to gather statements and claims (*e.g.*, Twitter API[9]).

5.2. Data flow and graph-based analysis

The data flow in the pipeline begins with the *Event Generator*, which reads the source (*e.g.*, a CSV file) containing a selection of statements. For each record, it publishes a message on a Kafka topic, called *Statements Topic*. These messages are then processed by a Flink job, called *Fact-Checking Job*, which extracts the payload,*i.e.*, the statement, and invokes the API exposed by the *Fact Checker*. The latter returns a response containing the outcome of the fact-checking process performed by an IFLM. The Flink job then enriches the previously received message with this information and publishes it on a different Kafka topic, called *Fact-checked Statements Topic*.

Hence, the Flink job *Neo4j Storing Job* analyzes the messages written in this topic, converts them into entities that map the desired representation for the graph database, and executes insert/update queries (*i.e.*, MERGE queries) to store the statements, the users, and the relationships between them. Fig. 3 shows a sub-graph stored in Neo4j representing a group of Tweets related to each other and with their respective authors via "REPLIED", "FROM_CONVERSATION", and "CREATED" relationships. By clicking on a specific Tweet, it is possible to access its properties, including the output of the fact-checking phase. The same applies also to authors.

Formally, the final property graph is a multi-layer directed graph $G(V, E, D_V, D_E, \{\lambda_V^i\}_{i=0}^w, \{\lambda_E^j\}_{j=0}^z)$, where V is the set of nodes representing statements (*e.g.*, Tweets) and authors, E is the set of edges modeling the different relationships between them (REPLIED, CREATED, FROM_CONVERSATION), D_V is the set of node layers, each one referring to a specific type of node, D_E is the set of edge layers, each one referring to a specific type of edge, and $\{\lambda_V^i\}_{i=0}^w$ and $\{\lambda_E^j\}_{j=0}^z$ are two sets of functions that associate nodes and edges with properties and assign them specific values. For example, the following function assigns labels to tweets:

$$\lambda_V^0 : V \times \{p_V^0 = '' \, label''\} \rightarrow P_V^0 = \{''TRUE'',''FALSE'',''UNKNOWN''\},$$
$$\text{with} \quad V \in D_V = \{''TWEETS''\}$$

As already mentioned, such kind of graph can be exploited to improve the fact-checking step when the LLM outputs labels with a low level of confidence. For example, the label of an uncertain statement

[8] https://neo4j.com/

[9] https://developer.x.com/en/docs/x-api

may be inferred via similar statements, *i.e.*, those concerning the same topics. The propagation history of an author, *i.e.*, how many statements for each specific level of truthfulness and for a given topic have been published, may be used to build a kind of reputation about that author, which can be included in the contextual information submitted to the LLM.

6. Conclusions

The application of large language models (LLMs) to automated fact-checking presents a promising avenue for addressing the challenges of misinformation. These models offer scalability, adaptability, and the ability to process nuanced language, making them valuable tools for tackling the ever-increasing volume of claims requiring verification. However, the experimental results reveal that the performance of LLM-based approaches does not yet meet the desired levels of accuracy and reliability, especially when adopting zero-shot learning strategies. Issues such as context misinterpretation and difficulties with "reasoning" over complex or ambiguous claims significantly impact their effectiveness. Additionally, the lack of interpretability in their outputs poses challenges for transparent and trustful fact-checking. Despite this, possible improvements can derive from the exploitation of graph data, specifically that related to the diverse relationships among the stored entities. Such kind of information can be used to refine the prompts through which the fact-checking task is submitted to the LLM.

Currently, we are investigating on this aspect and we are also evaluating the possibility to extend the graph presented above with additional concepts related to the *evidence* retrieved by the LLM, along with the *sources* it used to express its judgment. In particular, as we hypothesized for the authors at the end of Sec. 5.2, we may build a kind of reputation system for sources as well. We plan to make the model include such knowledge through a properly designed training process.

Declaration on Generative AI

The author(s) have not employed any Generative AI tools.

References

[1] C. Wardle, H. Derakhshan, INFORMATION DISORDER : Toward an interdisciplinary framework for research and policy making Information Disorder Toward an interdisciplinary framework for research and policymaking, 2017.

[2] N. Hassan, B. Adair, J. Hamilton, C. Li, M. Tremayne, J. Yang, C. Yu, The quest to automate fact-checking, Proceedings of the 2015 Computation + Journalism Symposium (2015).

[3] J. Wei, Y. Tay, R. Bommasani, C. Raffel, B. Zoph, S. Borgeaud, D. Yogatama, M. Bosma, D. Zhou, D. Metzler, E. H. Chi, T. Hashimoto, O. Vinyals, P. Liang, J. Dean, W. Fedus, Emergent abilities of large language models, 2022. URL: https://arxiv.org/abs/2206.07682. arXiv:2206.07682.

[4] G. Marvin, N. Hellen, D. Jjingo, J. Nakatumba-Nabende, Prompt engineering in large language models, in: I. J. Jacob, S. Piramuthu, P. Falkowski-Gilski (Eds.), Data Intelligence and Cognitive Informatics, Springer Nature Singapore, Singapore, 2024, pp. 387–402.

[5] R. Interian, R. G. Marzo, I. Mendoza, C. C. Ribeiro, Network polarization, filter bubbles, and echo chambers: an annotated review of measures and reduction methods, International Transactions in Operational Research 30 (2023) 3122–3158. URL: https://onlinelibrary.wiley.com/doi/abs/10.1111/itor.13224. doi:https://doi.org/10.1111/itor.13224. arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1111/itor.13224.

[6] Z. Zhao, J. Zhao, Y. Sano, O. Levy, H. Takayasu, M. Takayasu, L. Daqing, S. Havlin, Fake news propagate differently from real news even at early stages of spreading, EPJ Data Science 9 (2018). doi:10.1140/epjds/s13688-020-00224-z.

[7] M. Mendoza, S. Valenzuela, E. Núñez-Mussa, F. Padilla, E. Providel, S. Campos, R. Bassi, A. Riquelme, V. Aldana, C. López, A study on information disorders on social networks during the chilean

social outbreak and covid-19 pandemic, Applied Sciences 13 (2023). URL: https://www.mdpi.com/2076-3417/13/9/5347. doi:10.3390/app13095347.

[8] D. Zhang, Y. Wang, Z. Zhang, Identifying and quantifying potential super-spreaders in social networks, Scientific Reports 9 (2019) 1–11. doi:10.1038/s41598-019-51153-5.

[9] Z. Guo, M. Schlichtkrull, A. Vlachos, A survey on automated fact-checking, Transactions of the Association for Computational Linguistics 10 (2022) 178–206. URL: https://doi.org/10.1162/tacl_a_00454. doi:10.1162/tacl_a_00454. arXiv:https://direct.mit.edu/tacl/article-pdf/doi/10.1162/tacl_a_00454/1987018/tacl_a

[10] D. Quelle, A. Bovet, The perils and promises of fact-checking with large language models, Frontiers in Artificial Intelligence 7 (2024). URL: http://dx.doi.org/10.3389/frai.2024.1341697. doi:10.3389/frai.2024.1341697.

[11] K. M. Caramancion, News verifiers showdown: A comparative performance evaluation of chatgpt 3.5, chatgpt 4.0, bing ai, and bard in news fact-checking, 2023. URL: https://arxiv.org/abs/2306.17176. arXiv:2306.17176.

[12] E. Hoes, S. Altay, J. Bermeo, Leveraging chatgpt for efficient fact-checking, PsyArXiv. April 3 (2023).

[13] M. Sawinski, K. Węcel, E. Księżniak, M. Stróżyna, W. Lewoniewski, P. Stolarski, W. Abramowicz, Openfact at checkthat! 2023: Head-to-head gpt vs. bert - a comparative study of transformers language models for the detection of check-worthy claims, in: Conference and Labs of the Evaluation Forum, 2023.

[14] F. Arslan, N. Hassan, C. Li, M. Tremayne, A benchmark dataset of check-worthy factual claims, 2020. URL: https://arxiv.org/abs/2004.14425. arXiv:2004.14425.

[15] T.-H. Cheung, K.-M. Lam, Factllama: Optimizing instruction-following language models with external knowledge for automated fact-checking, in: 2023 Asia Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC), 2023, pp. 846–853. doi:10.1109/APSIPAASC58517.2023.10317251.

[16] Z. Yang, J. Ma, H. Chen, H. Lin, Z. Luo, Y. Chang, A coarse-to-fine cascaded evidence-distillation neural network for explainable fake news detection, in: Proceedings of the 29th International Conference on Computational Linguistics, International Committee on Computational Linguistics, Gyeongju, Republic of Korea, 2022, pp. 2608–2621. URL: https://aclanthology.org/2022.coling-1.230.

[17] W. Y. Wang, "liar, liar pants on fire": A new benchmark dataset for fake news detection, in: Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), Association for Computational Linguistics, Vancouver, Canada, 2017, pp. 422–426. URL: https://aclanthology.org/P17-2067. doi:10.18653/v1/P17-2067.