Numbers Don't Lie: Hybrid Extraction and Validation of Quantitative Statements in Arguments with Semi-Structured Information*

Mirko Lenz[1,2], Lorik Dumani[2], Ralf Schenkel[2], and Ralph Bergmann[1,2]

[1] German Research Center for Artificial Intelligence (DFKI)
Behringstraße 21, 54296 Trier, Germany
{mirko.lenz,ralph.bergmann}@dfki.de
[2] Trier University, Universitätsring 15, 54296 Trier, Germany
info@mirko-lenz.de, {dumani,schenkel,bergmann}@uni-trier.de

Abstract. Evidence in arguments may be stated in various forms, including quantitative statements (i.e, numerical relations between entities). This measurable information can be validated against reliable sources like Wikipedia to combat the spread of misinformation. In this paper, we propose a four-step pipeline that combines rule-based techniques with prompting strategies for generative language models in a hybrid fashion. We use regular expressions to identify candidates in claim-premise structures, extract statements using GPT-4o, augment the data with tables from Wikipedia, and validate statements through retrieval-augmented generation (RAG). The pipeline is evaluated on two existing argumentation corpora and the generated dataset is manually annotated to assess the quality of our predictions, showing promising results for extraction and mixed results for validation. Our code and data are available to foster further research in this area.

Keywords: Argumentation · Quantitative Statements · Validation · Large Language Models · Retrieval-Augmented Generation

1 Introduction

Argumentation is ubiquitous in human communication—having the most convincing argument is a key factor in persuading others. In addition to its persuasiveness, the quality of an argument is influenced by a number of factors, including the evidence used to support it [21]. Such evidence may be stated in various forms—for instance, as a reference to a scientific study, an exemplary case, or a quantitative statement. The latter is particularly interesting, as it provides a precise and measurable piece of information between two entities. We use the widely spread definition of an argument that describes it as *claim* (i.e., a standpoint the arguer wants the reader to take or object) and a *premise* (i.e., evidence or a reason to support or oppose the claim). Take the following argument as an example: "*Global overpopulation is a myth, because humans can now*

* The Version of Record is available online: doi.org/10.1007/978-3-032-02813-6_6

produce 16 times as much wheat per acre as we could in the 1500s". Here, the claim "*Global overpopulation is a myth*" is supported by a premise that contains a quantitative statement, namely that the wheat yield per acre has increased by a factor of 16. More precisely, it contains the entities "1500s", "today" (not explicitly stated), the quantity "wheat yield per acre", and the factor "16". The provided evidence can be verified by consulting data in a reliable source such as Wikipedia tables, enabling the reader to assess the argument's credibility. A particular challenge in this purpose is the fact that information from natural language texts might need to be extracted and rephrased. For example, one or multiple tables would most likely contain (in the best case) numbers from today and the 1500s, where a system has to derive whether the data is sufficient to represent the entities today and the 1500s, as well as whether the difference is (approximately) 16 times higher.

In this work, we focus on the, to the best of our knowledge, new task of automatic extraction and validation of quantitative statements. To combat the spread of misinformation, we propose a hybrid approach that combines rule-based techniques with prompting strategies involving Large Language Models (LLMs) in a pipeline: (i) identify quantitative operators in the argument using Regex, (ii) send the matches to a generative language model to extract the quantitative statement, (iii) scrape relevant Wikipedia articles (especially tables) for validation, and (iv) apply Retrieval-Augmented Generation (RAG) [16] to validate the extracted statement against the articles. To evaluate our approach, we developed an open source application that can perform all steps of the pipeline and tested it on two different datasets. Thus, we seek to answer the following research question: *How reliable are LLM-based techniques for extracting and validating quantitative statements in arguments?* Compared to existing fact-checking techniques, our approach is unique in that it incorporates the claim-premise-structure of arguments—enabling the validation of reasoning structures rather than individual statements.

The remainder of this paper is structured as follows: After discussing related work concerning argumentation and validation in Section 2, we present our approach in Section 3. The evaluation is carried out in Section 4, and we conclude with a discussion and an outlook in Section 5.

2 Related Work

In this section, we examine work in the area of fact-checking. Guo et al. [5] provide a comprehensive overview of this field. Here, it is crucial to distinguish the term fact-checking from fake news detection. The latter involves assessing news articles and labeling items based on aspects not related to veracity (e.g., satire detection [17,22]). Actual fact-checking is the task of assessing whether the claims made are true, consisting of (i) claim detection, (ii) evidence retrieval, and (iii) claim verification (i.e., verdict prediction and justification production). Existing work typically covers individual components of the task, such as (i) identifying rumors on social media [23,9] (ii) detecting the stance of a given piece of evidence towards a claim [13,7], or (iii) producing explanations and justifications for fact checks [11]. In our work, we focus more on the premise than on the claim. However, in Computational Argumentation (CA) premises can also serve as claims for other premises [15]. We do not validate natural language statements as common, but rather quantities that are contained in them, specifically using tables from the Web. Naturally, not all sources on the Web are credible. Like most fact-checking approaches, we rely on credible sources, in our case Wikipedia [5].

As in Thorne et al. [20], most approaches use natural language sentences from Wikipedia to verify claims. To the best of our knowledge, we are only aware of the work of Gupta et al. [6] and Chen et al. [3], which utilize semi-structured data and use tables and info boxes from Wikipedia. The work of Gupta et al. [6] is only comparable to a limited extent to our work, because while they restrict their dataset InfoTabS to tables consisting of info boxes from Wikipedia—more precisely, 23,738 (premise, hypothesis) pairs from 2,540 info boxes—we scraped all tables from a Wikipedia page whose content is related to a quantitative statement. Furthermore, Chen et al. [3] present the dataset TabFact comprising 16k Wikipedia tables as evidence for 118k human-annotated natural language statements labeled as entailed or refuted. They also present the two methods Table-BERT and Latent Program Algorithm, which lag far behind human performance. Koleva et al. [10] investigate the entity linking task in tables, and together with a new dataset, they present an LLM prompting strategy for the recognition of named entities in table cells. Lastly, our approach has a certain similarity to subject-predicate-object triples, which can be double-checked in knowledge bases such as DBpedia. Guo et al. [5] criticize that it is difficult to extract these from texts, but with the advent of instruction-based LLMs, our work overcomes this limitation.

Our approach combines claim-premise arguments with quantitative statements, so we need corpora that contain such structures. The aforementioned TabFact or InfoTabS datasets do not contain this essential information, so we use two other datasets for this paper: (i) Args containing 382,545 arguments whose premises are divided into 6,146,646 sentences [1] and (ii) Kialo containing 208,969 arguments with 302,555 sentences [15]. The use of other corpora containing argument graphs or debates is also possible and an interesting direction for future work.

3 Pipeline Steps

In this section, we describe our approach consisting of four sequential phases: (i) *matching* potential quantitative statements in arguments through RegEx, (ii) *extracting* these statements using LLMs, (iii) *augmenting* the found statements with information from Wikipedia, and (iv) *validating* the extracted statements against the augmented data using RAG. Our approach is designed to be generic and applicable to any argumentation corpus. We use real example from our evaluation (see Figure 4a) to illustrate each step in this section: a *pro-*

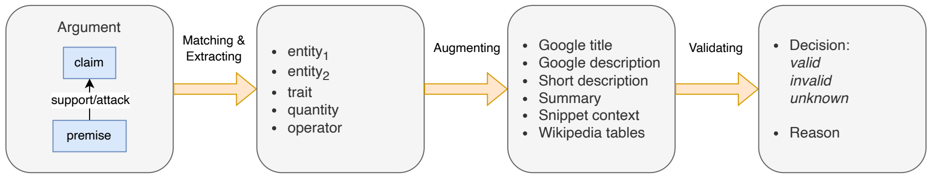

Fig. 1: Architectural overview of our proposed pipeline. Starting from claim-premise structures, quantitative statements are extracted and then augmented with Wikipedia data to validate their contents.

Table 1: The number of distinct claims matches for each RegEx and dataset.

pattern name	**Args**	**Kialo**	**example**
$\text{REGEX}_{\text{qualitative}}$	15,145	509	*is larger than*
$\text{REGEX}_{\text{as } attribute \text{ as}}$	4,137	134	*as strong as*
$\text{REGEX}^{\text{decimal}}_{\text{times}}$	2,546	118	*4 times higher*
$\text{REGEX}^{\text{number as word}}_{\text{times}}$	1,760	106	*four times higher*
$\text{REGEX}_{\text{percentage}}$	963	94	*4 % higher*
$\text{REGEX}_{\text{times as } attribute \text{ as}}$	115	13	*four times as fast as*
total distinct claims	20,195	590	

abortion premise stating “it takes $245,000 to raise a kid, that’s 490 times the amount of money it takes to have an abortion”.

3.1 Matching Regular Expressions

Since prompting on datasets is very costly, we employed regular expressions as pre-filters to find potential quantitative statements in arguments. For this purpose, we designed six regular expressions which are available in our GitHub repository and an overview shown in Table 1. They were developed as follows: (i) Two people developed them sequentially by testing them on our datasets, with the second taking up and refining the work of the first. (ii) We ran 103 simple regular expressions such as “`\w+ one of`”, “`\w+ [(vs.)(vs)(versus)] \w+`”, or “`\d+ of`” for the datasets and evaluated a maximum of 99 of each (depending on how many were found) on a scale of $0, 1, 2$ to determine whether the statements found are quantitative (0: not quantitative, 1: partially quantitative, 2: completely quantitative). They contained 7,479 matches with a mean of 72 and a median of 99 per expression. (iii) The most promising expressions were manually merged into the six general ones shown in Table 1. In our example, the pattern $\text{REGEX}^{\text{decimal}}_{\text{times}}$ matches “490 times” in the premise.

3.2 Extracting Quantitative Statements

Through the pattern matching results, we have a set of candidate premises that contain potentially quantitative statements. Next, we use a prompting strategy

Fig. 2: System message for extracting quantitative statements.

You are an assistant that extracts quantitative statements from arguments.

Task Description: An argument consists of a claim (a statement that is being argued) and a premise (a statement that supports or attacks the claim). The stance indicates whether the premise supports or attacks the claim. The goal is to extract quantity statements from the premise that are relevant to the claim. A quantity statement consists of two entities (e.g., "computers" and "consoles"), a trait between them (e.g., "cost"), an operator (e.g., "greater"), and a quantity (e.g., "2"). The operator indicates the relationship between the two entities, and the quantity specifies the amount of the trait that one entity has compared to the other.

Input: You will be provided with a claim, its premise, and the stance between them. As a starting point, a pattern-based approach has been used to identify sentences in the premise that contain some free-form operator. The operator indicates the relationship between two currently unknown entities in the sentence. As additional context, you are provided the entire regex pattern that matched the sentence together with the operator.

Output: You shall extract all relevant information to call the function PredictStatements.

Constraints: If *quantity* == 1.0, the operator "equal" or "approx" must be used. If *quantity* is any other value, the operator must be one of the other four options. The premise id will later be used to match the extracted quantity statements with the provided premise, so make sure to keep it.

combined with an LLM to obtain the referenced *entities* (e.g., game console and computer), the *trait* that is compared (e.g., price), the *quantity* (e.g., 4), and the *operator* between the entities (e.g., greater). To make an informed prediction, we provide the following properties of the argument to the LLM: The claim, its stance, all connected premises, and the list of matched sentences (including the full regular expression that caused each match). With the extensive context length of recent LLMs, we are able to send *all* premises connected to the investigated claim to the model—this should help to deal with challenges like coreferences. Dealing with multiple matches at once brings certain benefits, such as improved context awareness, but due to the probabilistic nature of LLMs, makes it harder to enforce predictable output. We use *function calling* as supported by an increasing number LLMs, allowing the definition of the expected response format using JSON schemas. For models without this feature, libraries such as Instructor[3] can be used to enforce the response format. Consequently, a claim with multiple pattern matching results in different premises only needs one request to the model—significantly speeding up the process and decreasing the cost of inference. For our example, the LLM extracts: $\{\text{'raising a kid'} > \text{'abortion'} : 490.0{\times}\text{cost}\}$. It works by providing a JSON schema describing the response format—in our case, a list of objects with the following properties: ID of the premise, the names of the two entities to be compared, their trait, operator, and quantity. It is accompanied by the system message shown in Figure 2.

3.3 Augmenting Statements with Wikipedia Data

The statements have been extracted from an argument and, as such, may contain incorrect data—for instance, the premise may contain exaggerations concerning

[3] github.com/567-labs/instructor.

the quantity to make a stronger point. Although LLMs are capable text generators, they are also prone to hallucination and are not directly usable for validation purposes. By grounding their predictions in some kind of background knowledge, we aim to mitigate this issue. We chose regular Wikipedia articles as our source of truth instead of more structured resources such as DBpedia because this eliminates the need of encoding knowledge graphs for use with LLMs. For specialized domains, this could be extended to include other sources as well (e.g., scientific articles). We use Google instead of Wikipedia's built-in search to gain access to valuable metadata such as snippets for each result. We employ the following search string: *"{entity_1} {trait} {quantity} times {operator} than {entity_2} site:en.wikipedia.org"*. Our example leads to: *"raising a kid cost 490.0 times greater than abortion site:en.wikipedia.org"*.

From the list of the ten most relevant results, we then extract several pieces of information: Starting from the Google search, we take the *Google snippet title*, which is identical to the Wikipedia title, and the *Google snippet description*, which contains text parts relevant to the search string. On the Wikipedia page itself, we extract several textual elements: (i) a *short description*, which is coded in the HTML code, (ii) the *summary* of the article, which is usually found in the text before the first heading, i.e., the text before the first h2 element in the HTML code, and (iii) the *snippet context*, i.e. the paragraph in the article that is most *similar* to the Google snippet description. In addition, we extract all *Wikipedia tables* in the article including their HTML code and ignore all attributes except *href* located in *a* tags. We use BM25 [18] to compute the similarity betweent the Google snippet and the paragraphs in the article to favor instances where the entity and trait are actually mentioned in the article. Compared to semantic-aware embedding approaches, BM25 is based on term frequency and inverse document frequency, which is more suitable for our use case.

Conflicting information may be extracted from the list of search results and the Wikipedia tables. We deliberately did not add a mechanism to resolve these conflicts as we wanted to evaluate the model's ability to handle such situations. In addition, it may even be the case that subjective elements are involved in the validation process, potentially mandating the inclusion of such conflicting information to make an informed decision.

3.4 Validating Quantitative Statements

After retrieving relevant Wikipedia articles and extracting tables, this information can be fed into the LLM to predict validations for the previously extracted quantitative statements. This RAG-based approach aims to reduce the risk of hallucinations and increase the reliability of the generated response [4]. With each table represented using HTML and each of the ten search results potentially containing multiple tables, passing all extracted information to the LLM has the potential of exceeding the model's maximum context size—for instance, the GPT-4o used in our evaluation is limited to 128,000 tokens. Requesting more tokens yields an error; thus, we prune the context by allocating a fixed number of tokens per article. We settled on a limit of 10,000 for GPT-4o so that even in the

Fig. 3: System message for validating quantitative statements.

You are an assistant that verifies quantitative statements via provided retrieval results.

Task Description: In the previous step, you extracted a quantitative statement from an argument. A quantity statement consists of two entities (e.g., "computers" and "consoles"), a trait between them (e.g., "cost"), an operator (e.g., "greater"), and a quantity (e.g., "2"). Via a web search, we identified relevant Wikipedia pages that contain additional context such as tables and summaries. The goal is to validate the extracted quantity statement based on the provided context.

Input: You will be provided with the extracted quantity statement, the claim, the premise, their stance, the web search string, and the Wikipedia search results. A quantity value of 1.0 acts as the reference point and the value 0.0 indicates that no meaningful quantity could be extracted. The tables have been extracted in their HTML representation. Only the given information shall be used to validate the quantity statement.

Output: You shall extract all relevant information to call the function PredictValidation.

Constraints: Do not use any external information beyond the provided context. If no data is available for the queried validation, respond with Unknown.

worst case scenario 28,000 tokens (or about 20% of the context size) are available for the remaining input data like the system prompt and the generated output. Each table is then sequentially added to the context if and only if the mentioned limit is not exceeded. In this process, we maintain the natural order of the tables as found on the page. Due to the generative nature of LLMs, the provided reason is not guaranteed to correspond to the predicted decision. Nevertheless, our tests showed a better decision quality when the reason was included in the prompt—similar to how chain-of-thought reasoning works better in certain cases. Similar to the statement extraction step, we use function calling—this time, to obtain a single object with the validation decision (valid/invalid/unknown) and the reasoning behind. The system message is shown in Figure 3. In our example, the model returns "Unknown" as Wikipedia lacks specific abortion cost data to verify the quantitative statement.

There is a limitation to this validation technique—temporal elements in arguments may introduce ambiguity. Consider the following statement used in Section 1: "*Global overpopulation is a myth, because humans can now produce 16 times as much wheat per acre as we could in the 1500s.*" The "now" in this statement is not explicitly defined, which could lead to different interpretations (e.g., due to the knowledge cutoff of the model) with the consequence that multiple answers may be correct.

4 Evaluation

Having presented our hybrid pipeline for extracting and validating quantitative statements in arguments, we now evaluate its effectiveness on the two argument corpora Args and Kialo presented in Section 3. After presenting our methodology, we analyze the generated dataset and discuss the results obtained from our manual annotation.

4.1 Methodology

Our final pipeline is implemented in Python and is available as open-source software.[4] We use OpenAI's model gpt-4o-2024-05-13 for all experiments due to its favorable cost-performance ratio. The dataset KIALO contains 2,423 claim matches, while ARGS contains 36,542 matches, out of which we sampled 2,500 for our evaluation. These claims were then passed to GPT-4o to extract quantitative statements, resulting in 3,385 claim-premise pairs for ARGS and 2,671 for KIALO. To avoid rate limits imposed by Google when retrieving relevant Wikipedia articles, we created two variants of the datasets: (i) $\text{ARGS}_{\text{EVAL}}$ and $\text{KIALO}_{\text{EVAL}}$ containing the extracted quantitative statements and the validation decisions based on Wikipedia data for 50 randomly sampled claims (leading to 124 claim-premise pairs), and (ii) $\text{ARGS}_{\text{FULL}}$ and $\text{KIALO}_{\text{FULL}}$ containing the extracted quantitative statements for all 6,056 claim-premise pairs. In this section, we use variant (i) as it contains all the necessary information for our evaluation. The full set of extracted statements may still be a valuable resource for future research, so all four datasets are available on request via Zenodo.[5]

To assess the quality of the extracted quantitative statements and their validation, we performed a manual annotation study. We asked two student annotators that are familiar with argumentation to label the generated statements and validations as *correct*, *partly correct*, *incorrect*, or *unknown*. They were also tasked with manually fixing the quantitative statements if necessary and validating them against data from Wikipedia. The annotation task was performed using the tool *Label Studio*.[6] The code for setting up the annotation interface is included in our GitHub repository.

4.2 Analysis of the Dataset

In this section, we give an overview of the generated datasets. Pattern matching identified substantially more quantitative arguments in the ARGS dataset (ca. 7 %) compared to the KIALO dataset (ca. 0.3 %), with frequently occurring operators including phrases like "is better than" and "is more than". For the predicted statements, we found that entities appeared mostly once and concrete numerical comparisons were uncommon, whereas operators such as "greater", "equal", and "less" had overlap in both datasets. Data augmentation was largely successful, on average our approach found more than one table per extracted statement. The predicted validations were mostly "unknown". By analyzing the corresponding explanations, the main reason was insufficient context information.

4.3 Analysis of the Manual Annotation

Each annotator labeled the same 124 claim-premise pairs from the EVAL datasets as *correct*, *partly correct*, *incorrect*, or *unknown*, This allows us to calculate the

[4] github.com/recap-utr/quantigpt (MIT license).
[5] zenodo.org/records/15720817.
[6] github.com/HumanSignal/label-studio.

Table 2: Inter-annotator agreement with two perspectives of Krippendorff's α.

datset	**task**	α_{all}	α_{known}	C
Args	statements	.453	.519	.565
	validations	.103	.132	.580
Kialo	statements	.265	.339	.510
	validations	-.044	.016	.491

Table 3: Distribution of the merged labels.

dataset	**task**	**fully correct**	**partly correct**	**incorrect**	**unknown**
Args	statements	.434	.341	.174	.051
	validations	.732	.094	.138	.036
Kialo	statements	.509	.318	.127	.046
	validations	.700	.100	.127	.073

Inter-Annotator Agreement (IAA) to measure the reliability between annotators. We use Krippendorff's α [12]—it is suitable for multiple annotators, accounts for missing values, and handles categorical data. Our custom distance function $d(x, y)$ is shown in Section 4.3—it penalizes large differences by assigning a distance of 3 for all pairs that do not match.

$$d(x,y) = \begin{cases} 0 & \text{if } x = y, \\ 1 & \text{if } x \in \{\text{correct}, \text{incorrect}\} \wedge y = \text{partly correct}, \\ 3 & \text{otherwise.} \end{cases}$$

Table 2 shows the IAA results. In addition to the overall α_{all}, we also calculate the α_{known} for known labels only by excluding items where at least one annotator chose *unknown*. The percentage of concordant labels C shows that annotators agreed on the same label in only 50–60% of cases. Assessing the extraction of quantitative statements yielded a moderate agreement for the Args dataset and a fair agreement for the Kialo dataset [14]. The validation task showed poor agreement for both datasets (even being negative for Kialo). Multiple reasons could be the cause for this, such as the complexity of the task, the subjectivity of argumentation, vague instructions, and ambiguity in the data. The annotators noted that the Wikipedia sources were off-topic in some cases, possibly contributing to the low agreement.

Next, we analyze the label distribution. For the results in Table 3, we merged the individual labels from the two annotators into a single dataset with 248 labels regardless of agreement or disagreement. For the quantitative statements, the annotators assigned the label *incorrect* in less than 20 % of the cases and *unknown* in approximately 5 %. At the same time, 30 % of the statements were labeled as *partly correct* (i.e., one or multiple parts of the statement were wrongly extracted). For the validation task, the annotators agreed on the label *correct* in about 70 % of the cases, while the label *incorrect* was assigned in about 15 %. At the same time, the IAA for the validation task was lower than for the extraction task, limiting the significance of the otherwise promising results for the validation task.

Returning to the research question, we conclude that the LLM-based techniques for extracting quantitative statements in arguments are not fully reliable. About half of the extracted statements were deemed correct by the annotators, but the validation task showed poor agreement between the annotators—requiring at least some manual intervention for the remaining half. Given the relatively low number of incorrect labels, our approach may be suitable for interactive fact-checking systems that combine human and machine intelligence. For our evaluation dataset, the two annotators manually fixed the incorrect and partly correct statements, which could be used to fine-tune the LLM in future work. By making available the annotated datasets, we hope to foster further research in this area.

4.4 Qualitative Analysis

Having obtained an overview of the dataset and the results of the manual annotation, we now analyze one successful and one failed outcome (see Figure 4) of our pipeline and investigate the annotators' decisions. The two examples are taken from ARGS—the largest of the two corpora. Together with the original argument, we show the predicted statement/validation and (if necessary) the corrected statement/validation from both annotators. After presenting the predictions and annotations, we continue with a discussion of the results. For the sake of brevity, we use the following notation for the quantitative statements:

'entity_1' operator 'entity_2' : quantity × trait

For the argument in Figure 4a, our pipeline was able to correctly extract the quantitative statement. One of the annotators agreed with that predicted validation *unknown*, the other however found additional evidence to mark this statement as *invalid*. This indicates that the augmentation step should be improved to inject more relevant contextual information into the validation prompt.

For the argument in Figure 4b, the pipeline failed to extract the correct quantitative statement. Notably, the two annotators extracted different statements from the argument, the first one focusing on the gravitational constant, the second one on Superman's speed. Consequently, the validation corrections differ as well—thus, both extraction and validation were found to be lacking. We also observe subjective elements w.r.t. the extraction of the quantitative statement. Validation suffers from a lack of contextual awareness of the LLM: It failed to deduce that the gravitational constant of the earth is compared to the one of Superman's origin planet and instead states that its value is constant anywhere on the earth. This may be caused by the incorrect extraction of the quantitative statement and the mentioned subjectivity—thus being a cascading error.

Claim: Abortion
Premise: On average, it takes \$245,000 to raise a kid, that's 490 times the amount of money it takes to have an abortion.
Stance: Pro
Predicted Statement: 'raising a kid' > 'abortion' : 490.0 × cost
Predicted Validation: Unknown—The extracted data provides the average cost of raising a child and some information on abortion but does not provide a specific cost for obtaining an abortion. Without an actual value for the cost of abortion, it is impossible to verify the quantitative relationship that the cost of raising a child is 490 times the cost of an abortion.
Corrected Validation 1: Invalid—In 2022, a medication abortion cost was about \$580 at Planned Parenthood, though it could be more, up to around \$800 (\$245000 / \$580 < 490) (Abortion in the United States and Cost of raising a child).
Corrected Validation 2: Unknown—No comparison context available in premise, without knowing the country/region/. . . the values can't be compared.

(a) Successful example.

Claim: I believe that Superman, without a doubt, would beat Goku in a fight.
Premise: Let's assume the radius of his hand is 3 in, or 0.0762 m. Then assuming the gravitational constant is equal to ours.
Stance: Con
Predicted Statement: 'gravitational constant' = 'ours' : 1.0 × constant
Predicted Validation: Valid—The data confirms that the gravitational constant, often denoted by G, is a universal constant that applies both in Earth's context and in broader physical laws such as Newton's law of universal gravitation and Einstein's general relativity. It does not change depending on the context or location, affirming that the claim of the gravitational constant being equal to our own is valid.
Corrected Statement 1: 'krypton' = 'earth' : 1.0 × gravitational constant
Corrected Validation 1: Invalid—Krypton has a much higher gravity than Earth which is the reason for Kryptonians' super powers (Krypton (comics)).
Corrected Statement 2: 'superman' = 'light' : 0.14 × speed
Corrected Validation 2: Unknown—Non decidable unless more specific context is available, depending on the source/variant of Superman talked about, the speeds vary wildly (Superman).

(b) Failed example.

Fig. 4: Two exemplary outcomes of our pipeline.

4.5 Limitations

Our evaluation provides an overview of the performance of our pipeline, but it is not without limitations. Fact-checking is an immensely time-consuming task, which can take several days for a single statement, even if several trained and professional fact-checkers are working to find evidence for it. In our study, we evaluated a small dataset that still required substantial effort. We commissioned two annotators because four eyes see more than two. But, as usual when dealing with natural language, one actually needs more annotators. For reasons of efficiency, we initially identified the statements to be verified using regular expressions from the texts, which we refined over the period of more than a year by testing them on these datasets. The recall could be increased by adding more regular expressions. Naturally, other datasets may require other regular expressions. Still, we opted for a precision-oriented approach, meaning that even more regular expressions would increase the recall, but not necessarily decrease the precision.

The use of instruction-based LLMs such as ChatGPT to solve Natural Language Processing (NLP) problems is controversial due to the fact that their power and, above all, their limitations are not completely transparent. Results are rarely identical due to random seeds and are also highly dependent on the prompts sent to the system. However, we believe that the power of generative models far exceeds that of older fine-tuning-based approaches and have therefore opted for this approach. In our evaluation, we relied solely on a single LLM model, GPT-4o by OpenAI. We did not compare it to other models, especially open-weight models, and left this for future work.

5 Conclusion and Future Work

In this paper, we present a hybrid pipeline for extracting and validating quantitative statements in arguments. We used a combination of pattern matching, LLMs, and RAG to extract and validate the statements. Our approach is evaluated on two argument corpora, Args and Kialo, and the results are checked by two student annotators. We provide a working implementation and two datasets containing the extracted statements and their validation decisions for further research. The evaluation shows promising results for the extraction task with less than 20 % being rated as incorrect, but mixed results for the validation task—meaning that our approach is mostly suitable for interactive scenarios and conclude that LLM-based techniques for extracting quantitative statements in arguments are not fully reliable.

In future work, we plan to improve the extraction of quantitative statements by refining the regular expressions in an attempt to improve the context provided to the LLM. In addition to solving the task via pattern matching—which could oversimplify the task—we plan to investigate more complex methods such as template filling [8]. We will also evaluate the use of additional sources for data augmentation, such as DBpedia [2] and ConceptNet [19]. With recent advances in multimodal LLMs, another possible approach is to extract images from the matched Wikipedia pages and use them as additional context for the validation task. We plan to test open-source LLMs where the training data is known s.t. we can make more informed decisions about potential sources of background knowledge. Lastly, we aim to increase the IAA for the validation task by providing more detailed instructions to the annotators and possibly using more annotators to reduce the impact of individual biases.

Acknowledgments. We thank Mircea-Luchian Pojoni, Martin Hoppe, and Kilian Bartz for their invaluable help in doing the annotations.

References

1. Ajjour, Y., Wachsmuth, H., Kiesel, J., Potthast, M., Hagen, M., Stein, B.: Data Acquisition for Argument Search: The args.me Corpus. In: KI 2019: Advances in Artificial Intelligence. pp. 48–59. Springer International Publishing (2019)

2. Bizer, C., Lehmann, J., Kobilarov, G., Auer, S., Becker, C., Cyganiak, R., Hellmann, S.: DBpedia - A crystallization point for the Web of Data. Journal of Web Semantics pp. 154–165 (2009)
3. Chen, W., Wang, H., Chen, J., Zhang, Y., Wang, H., Li, S., Zhou, X., Wang, W.Y.: Tabfact: A large-scale dataset for table-based fact verification. In: 8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020. OpenReview.net (2020), https://openreview.net/forum?id=rkeJRhNYDH
4. Gao, Y., Xiong, Y., Gao, X., Jia, K., Pan, J., Bi, Y., Dai, Y., Sun, J., Wang, M., Wang, H.: Retrieval-Augmented Generation for Large Language Models: A Survey (2024)
5. Guo, Z., Schlichtkrull, M.S., Vlachos, A.: A survey on automated fact-checking. Trans. Assoc. Comput. Linguistics **10**, 178–206 (2022). https://doi.org/10.1162/TACL_A_00454, https://doi.org/10.1162/tacl_a_00454
6. Gupta, V., Mehta, M., Nokhiz, P., Srikumar, V.: INFOTABS: inference on tables as semi-structured data. In: Jurafsky, D., Chai, J., Schluter, N., Tetreault, J.R. (eds.) Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020. pp. 2309–2324. Association for Computational Linguistics (2020). https://doi.org/10.18653/V1/2020.ACL-MAIN.210, https://doi.org/10.18653/v1/2020.acl-main.210
7. Hardalov, M., Arora, A., Nakov, P., Augenstein, I.: A survey on stance detection for mis- and disinformation identification. In: Carpuat, M., de Marneffe, M., Ruíz, I.V.M. (eds.) Findings of the Association for Computational Linguistics: NAACL 2022, Seattle, WA, United States, July 10-15, 2022. pp. 1259–1277. Association for Computational Linguistics (2022). https://doi.org/10.18653/V1/2022.FINDINGS-NAACL.94, https://doi.org/10.18653/v1/2022.findings-naacl.94
8. Huang, K.H., Tang, S., Peng, N.: Document-level Entity-based Extraction as Template Generation. In: Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing. pp. 5257–5269. Association for Computational Linguistics (2021)
9. Islam, M.R., Liu, S., Wang, X., Xu, G.: Deep learning for misinformation detection on online social networks: a survey and new perspectives. Soc. Netw. Anal. Min. **10**(1), 82 (2020). https://doi.org/10.1007/S13278-020-00696-X, https://doi.org/10.1007/s13278-020-00696-x
10. Koleva, A., Ringsquandl, M., Hatem, A., Runkler, T., Tresp, V.: Wiki-TabNER: Advancing Table Interpretation Through Named Entity Recognition (2024)
11. Kotonya, N., Toni, F.: Explainable automated fact-checking: A survey. In: Scott, D., Bel, N., Zong, C. (eds.) Proceedings of the 28th International Conference on Computational Linguistics, COLING 2020, Barcelona, Spain (Online), December 8-13, 2020. pp. 5430–5443. International Committee on Computational Linguistics (2020). https://doi.org/10.18653/V1/2020.COLING-MAIN.474, https://doi.org/10.18653/v1/2020.coling-main.474
12. Krippendorff, K.: Reliability in Content Analysis: Some Common Misconceptions and Recommendations. Human Communication Research pp. 411–433 (2004)
13. Küçük, D., Can, F.: Stance detection: A survey. ACM Comput. Surv. **53**(1), 12:1–12:37 (2021). https://doi.org/10.1145/3369026, https://doi.org/10.1145/3369026
14. Landis, J.R., Koch, G.G.: The Measurement of Observer Agreement for Categorical Data. Biometrics pp. 159–174 (1977)

15. Lenz, M., Sahitaj, P., Kallenberg, S., Coors, C., Dumani, L., Schenkel, R., Bergmann, R.: Towards an Argument Mining Pipeline Transforming Texts to Argument Graphs. In: Computational Models of Argument. pp. 263–270. IOS Press (2020)
16. Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., Yih, W.t., Rocktäschel, T., et al.: Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks. In: Advances in Neural Information Processing Systems. pp. 9459–9474. Curran Associates, Inc. (2020)
17. Oshikawa, R., Qian, J., Wang, W.Y.: A survey on natural language processing for fake news detection. In: Calzolari, N., Béchet, F., Blache, P., Choukri, K., Cieri, C., Declerck, T., Goggi, S., Isahara, H., Maegaard, B., Mariani, J., Mazo, H., Moreno, A., Odijk, J., Piperidis, S. (eds.) Proceedings of The 12th Language Resources and Evaluation Conference, LREC 2020, Marseille, France, May 11-16, 2020. pp. 6086–6093. European Language Resources Association (2020), https://aclanthology.org/2020.lrec-1.747/
18. Robertson, S.E.: Okapi at TREC-3 pp. 109–126 (1994)
19. Speer, R., Chin, J., Havasi, C.: ConceptNet 5.5: An open multilingual graph of general knowledge. In: Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence. pp. 4444–4451. AAAI Press (2017)
20. Thorne, J., Vlachos, A., Christodoulopoulos, C., Mittal, A.: FEVER: a large-scale dataset for fact extraction and verification. In: Walker, M.A., Ji, H., Stent, A. (eds.) Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2018, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 1 (Long Papers). pp. 809–819. Association for Computational Linguistics (2018). https://doi.org/10.18653/V1/N18-1074, https://doi.org/10.18653/v1/n18-1074
21. Wachsmuth, H., Naderi, N., Hou, Y., Bilu, Y., Prabhakaran, V., Thijm, T.A., Hirst, G., Stein, B.: Computational Argumentation Quality Assessment in Natural Language. In: Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers. pp. 176–187. Association for Computational Linguistics (2017)
22. Zhou, X., Zafarani, R.: A survey of fake news: Fundamental theories, detection methods, and opportunities. ACM Comput. Surv. **53**(5), 109:1–109:40 (2021). https://doi.org/10.1145/3395046, https://doi.org/10.1145/3395046
23. Zubiaga, A., Aker, A., Bontcheva, K., Liakata, M., Procter, R.: Detection and resolution of rumours in social media: A survey. ACM Comput. Surv. **51**(2), 32:1–32:36 (2018). https://doi.org/10.1145/3161603, https://doi.org/10.1145/3161603