Numerical and Temporal Fact Checking

Alexandra Ioana Neagu
Delft University of Technology
A.I.Neagu@student.tudelft.nl

Emil Malmsten
Delft University of Technology
E.L.Malmsten@student.tudelft.nl

Teodor-Gabriel Oprescu
Delft University of Technology
T.Oprescu@student.tudelft.nl

Jan Warchocki
Delft University of Technology
J.Z.Warchocki-1@student.tudelft.nl

Abstract

Numerical claims are often perceived by humans as credible, even if the facts are partially inaccurate. This makes the problem of verifying such claims crucial to avoid disinformation spreading, especially on the internet. Although existing works investigate claim verification on synthetic claims, little work has been done in verifying real-world, numerical claims. In this work, we investigate methods for enhancing numerical and temporal fact-checking by evaluating different NLP models on the QuanTemp dataset. We find that modern NLI or generative models perform the best on numerical claims. Furthermore, rather surprisingly, we find that models designed specifically to handle numerical problems under-perform compared to NLI and generative models. We also find that claim decomposition methods do not yield an overall improvement in performance. Yet, tailoring decomposition prompts towards specific claim types can lead to an improvement for those types. Finally, we propose a novel evidence reranking method, although its results do not conclusively point towards improvement. The code implementation for this paper can be found at https://github.com/Jaswar/nlp-project.

1 Introduction

The rise of digital media and online information sharing has brought unprecedented connectivity and access to knowledge, but it has also been accompanied by a concerning rise in the spread of disinformation. False or misleading claims, especially those involving numerical data or temporal information, can have serious negative impacts on society (Vosoughi et al., 2018). People tend to perceive numerical claims as more credible due to the illusion of numerical truth effect, even when the underlying facts are inaccurate (Sagara and Peters, 2015). Verifying the truth of such quantitative claims poses unique challenges for automated fact-checking systems.

Existing automated fact verification approaches have made significant progress on synthetic claims from structured knowledge bases like Wikipedia, as well as real-world claims (Viswanathan et al., 2024). However, there remains a significant gap in handling complex numerical claims from the open domain that often lack precise information and require temporal reasoning. To address this limitation, we work on the recently released QuanTemp dataset, a diverse, multi-domain benchmark focused exclusively on verifying numerical claims encompassing temporal, statistical, and other quantitative aspects across different contexts (Viswanathan et al., 2024).

Numerical reasoning abilities like understanding numerical patterns, interpreting data, and performing arithmetic operations are crucial for evaluating the truthfulness of such claims. However, current natural language inference (NLI) models employed for fact verification often treat numbers as regular tokens without capturing their semantic significance (Ravichander et al., 2019). This limitation lowers their performance on claims that require quantitative analysis, data interpretation, or temporal reasoning.

In this work, we investigate techniques to enhance the numerical and temporal reasoning capabilities of fact-checking systems on the QuanTemp benchmark. Specifically, we aim to answer the following research questions:

RQ1: How do the existing state-of-the-art NLI models, including large language models, perform on different types of numerical claims in QuanTemp?

RQ2: Does specialized handling of numerical information improve the performance of NLI models (both fine-tuned and large language models) on numerical claim verification?

RQ3: Does decomposing complex claims into sub-questions help in better evidence retrieval and downstream veracity prediction?

By exploring approaches like specialized numerical handling within NLI models, claim decomposition into sub-questions, and incorporating temporal relevance signals, we seek to advance the state of automated fact-checking for complex numerical and temporal claims from the open domain. As the volume and complexity of online information continue to grow exponentially, reliable fact-verification techniques focused on quantitative claims will become increasingly valuable.

2 Related work

Most existing fact-checking datasets focus on textual claim verification using either structured or unstructured data (Zeng et al., 2021; Thorne et al., 2018). However, real-world data often involve claims that require numerical understanding for evidence retrieval and verification. Annotators of datasets like FEVEROUS (Aly et al., 2021) have noted the difficulty of verifying numerical claims, which require numerical reasoning; even just simple arithmetic.

For datasets that focus on numerical claims, (Vlachos and Riedel, 2015) propose a simple distant supervision approach using Freebase to verify statistical claims that reach 60% accuracy. (Thorne and Vlachos, 2017) improve upon this and reach 68% accuracy. These claims are synthetic and answerable with simple knowledge base facts. Similarly, (Cao et al., 2018) explore formula extraction for checking numerical consistency in financial statements using Wikidata. (Jandaghi and Pujara, 2023) investigate the identification of quantitative statements for fact-checking trend-based claims. However, these datasets do not represent real-world claims.

This work is heavily inspired by (Viswanathan et al., 2024), who collected and released a multi-domain dataset primarily composed of numerical claims and temporal expressions, including fine-grained metadata from fact-checkers and an evidence collection. This was the first natural numerical claims dataset. In this work, we use this dataset to assess the numerical reasoning capabilities of current LMs on numerical claims.

3 Method

3.1 Evidence extraction

Following (Viswanathan et al., 2024), for each claim in the QuanTemp dataset, we use the BM25 algorithm (Trotman et al., 2014) to rank evidence. The top $k = 100$ evidences are then selected, re-ranked using `paraphrase-MiniLM-L6-v2` (Reimers and Gurevych, 2019), and entailed with the claim. Although (Viswanathan et al., 2024) suggests using $k = 3$, in this work, we use $k = 100$ to ensure that all 256 tokens are used. It should be noted that using more evidence does not have an adverse effect, as the extra tokens will be truncated. We combine the evidence with the claim as shown in Figure 1.

```
[Claim]: ...[Evidence]: ...[Evidence]: ...
```

Figure 1: Evidence with claim entailment.

The resulting claim is processed using the tokenizer associated with the model being tested. The length of the processed sequence is then adjusted, either truncated or padded, to fit 256 units. To enable classification, the `[CLS]` token is appended to the sequence.

3.2 Model selection and setup

From the NLI models, we chose to evaluate BART-large-MNLI (Meta AI, 2023), Roberta-Large-MNLI (Liu et al., 2019), and deberta-v3-base-tasksource-nli (Sileo, 2024). The generative models we chose are Flan-T5 (Chung et al., 2022), GPT-2 (Radford et al., 2019), and BART (Lewis et al., 2020). Finally, we evaluate math models. With the term 'math model', we refer to any deep learning model trained to handle numerical tasks better. For this set of models, we selected NumT5 (Yang et al., 2021), MathRoberta (Novotnỳ and Stefánik, 2022), PASTA (Gu et al., 2022), and Elastic BERT (Liu et al., 2022). These models were selected for evaluation due to their ease of setup and availability on HuggingFace.

For all of the listed models, we use the pre-trained checkpoints and fine-tune on the claim classification task. The checkpoints were often missing the pooler layer; thus, we decided to use a custom pooler layer for all models. The architecture of the final model contains the main backbone (the transformer being tested), a custom pooler layer, and an MLP with the classification head. Given the final output of the backbone (e.g., the final output of the encoder), the class token is extracted in the pooler layer and passed to the MLP. The models are trained without freezing any components. Although we have attempted to follow this workflow, some models required additional care. The exact

implementation details can be found in the source code for this paper.

The MLP is a two-layer network with one hidden dimension of 500 neurons and an output dimension of 3, corresponding to the number of classes. ReLU was chosen to be the non-linearity. Dropout is used before the MLP with $p = 0.1$. The models are trained with the AdamW optimizer (Loshchilov and Hutter, 2018) with a learning rate of $2 \cdot 10^{-5}$ and $\epsilon = 10^{-8}$. Cross-entropy is used as the loss function, and the models are trained using a stopping criterion of 2 epochs. The batch size varies between models and is chosen so that the model can fit into the available memory.

3.3 Claim decomposition

To increase the effectiveness of evidence gathering and determining the veracity of claims, we use claim decomposition. We anticipate this will enhance our ability to gather evidence, as a decomposed claim such as "The waste of the Democrats is staggering. Obama spent 100 million dollars on a bench in Central Park, and Biden just bought a private jet for his son with the federal budget." can be broken down into two parts: about the bench and about the plane. It is likely easier to find evidence for each of these parts individually than to find an article mentioning both. Therefore, we use the decomposed claims as extra input to the BM25 evidence extraction mechanism from Section 3.1. This ensures that equal amounts of evidence are gathered for every sub-question.

Additionally, determining veracity should be simpler since both parts being false could lead the model to predict *False*, both true to *True*, and a mix would lead to *Conflicting*. To allow this, we use the generated programs or sub-questions as extra input to the language models (LMs), as shown in Figure 2.

```
[Claim]: ...[Program or Sub-question]:
          ...[Evidence]: ...
```

Figure 2: Evidence with claim and program or sub-question entailment.

3.3.1 Fine-tuning on the ClaimDecomp dataset

The first approach to decomposing claims is to fine-tune a BART model to perform the decomposition. We do this using the ClaimDecomp dataset composed by (Chen et al., 2022), which contains claims decomposed into sub-questions by annotators. In this approach, the input is a claim, and the target is the sub-questions as an array.

3.3.2 Prompting a larger model

As suggested by (Mohri and Hashimoto, 2024), our next approach is to prompt a more advanced LM. While the original paper uses GPT-4 for this task, we use GPT-3.5-Turbo due to financial constraints. We build upon the prompt used in (Mohri and Hashimoto, 2024) by providing several examples of what a good question decomposition looks like.

3.3.3 ProgramFC

Proposed in (Pan et al., 2023), ProgramFC decomposes complex claims into executable, Python-like programs. In this work, we follow this motion and generate the programs for each claim. It should be noted that the original work used the now deprecated `text-davinci-003` model for program generation. Following the official recommendation (OpenAI, 2024), we instead use `gpt-3.5-turbo-instruct`.

3.3.4 Tailored Decomp through custom claim breakdown

Our custom decomposition method consists of three main components: claim type classification, claim decomposition, and sub-question generation. We leverage various pre-trained models and fine-tune them as necessary to achieve optimal performance in each step.

To classify the type of claim, we trained a BART-large-MNLI model. This model is well-suited for NLI and helps identify each claim's taxonomy label. We fine-tuned the model on the QuanTemp dataset to learn the distinctions between different types of claims. During inference, the model predicts the type of each incoming claim, allowing it to be routed to the appropriate decomposition method. Once the claim type is determined, we use a Flan-T5-large model to decompose the claim into a set of sub-questions. The decomposition method varies based on the claim type, ensuring that the resulting sub-questions are relevant and informative. We utilize a prompt-based approach for few-shot learning to guide the model. The prompts are specifically designed for each claim type:

- **Comparison claims:** Prompts for these claims focus on breaking down the compara-

tive elements, such as identifying the entities being compared and the metrics used.

- **Interval claims:** These prompts guide the model to identify the range or intervals mentioned in the claim, breaking them down into specific sub-intervals or related quantities.
- **Temporal claims:** Temporal claims are decomposed by focusing on the time-related aspects, extracting sub-questions related to specific time points or periods.
- **Statistical claims:** For statistical claims, the prompts help to decompose the claim into sub-questions related to statistical measures, distributions, or data sources.

After the initial decomposition, each sub-question undergoes further decomposition using a Flan-T5-large model fine-tuned on the StrategyQA dataset (Geva et al., 2021). This dataset is specifically designed to generate multiple decomposition sub-questions from a single question, making it ideal for our purpose.

3.4 Temporal and numeric relevance

We further aimed to improve the performance for specific taxonomies of labels by conducting an additional re-ranking of evidence. Thus, for temporal claims inspired by (Gade and Jetcheva, 2024), we take the semantic scores generated by the `paraphrase-MiniLM-L6-v2` model and add a temporal relevance score. This temporal relevance score was computed by extracting the timestamps from the claim and each evidence, then calculating the difference between the dates, normalized by the number of days. Date extraction involved first checking whether the claim has an existing `crawled_date` field, followed by attempting to extract the first encountered date from the claim and the evidence. Due to the variety of date formats and the presence of month names in multiple languages, extraction was challenging. Therefore, we created a comprehensive regular expression to identify dates, and if no date was found, we used spaCy's (Honnibal et al., 2020) advanced entity recognition (NER) capabilities to detect any dates in the text.

For statistical claims, we extracted all numbers from the claim and the evidence and calculated the number of common occurrences. These counts were then normalized by dividing each count by the maximum count. This determined the numeric relevance, which was added to the `paraphrase-MiniLM-L6-v2` semantic scores. Comparison and interval claims were left unchanged, with no additional re-ranking implemented.

4 Results

4.1 Quantitative analysis

Following (Viswanathan et al., 2024), we measure the weighted F1 score per each category of claims and the overall F1 score. Furthermore, we choose to calculate the accuracy of each model for a more intuitive evaluation of performance. The results for NLI, generative, and math-specialized models on the test set are presented in Table 1. The results of the best models are denoted in bold font.

Importantly, we first observe that all models outperform the naive baseline. We find that Flan-T5 and BART obtain the best results and are almost equivalent in performance, both being the only models to obtain an overall F1 score of above 66%. Furthermore, we observe that GPT-2 is noticeably worse than the other generative and NLI models. Being released in 2019, GPT-2 (Radford et al., 2019) is older compared to other models such as Flan-T5 (Chung et al., 2022) or DeBERTa (Sileo, 2024), which could explain this difference. Comparing the math models, we observe their performance to be far worse than that of generative or NLI models. Although this might be a counter-intuitive result, we provide potential reasons for this outcome in the following section.

4.2 Qualitative analysis

To perform the qualitative analysis, we randomly choose examples from the following three types of claims:

1. Claims classified incorrectly by all NLI and generative models but classified correctly by all math models.
2. Claims classified correctly by all NLI and generative models but incorrectly by all math models.
3. Claims classified incorrectly by all models.

For each of the three types of claims, we pick 5 examples at random. We then examine the examples and the top 10 evidence related to each claim. Based on these examples, we attempt to find cases where different models perform better than others. Example claims with top 1 evidence for each type are presented in Table 2.

Table 1: The results for the tested model per each claim category. We measure performance in terms of the F1 and accuracy scores. The best-performing models are denoted in bold text. Modern NLI and generative models, such as FlanT5 or DeBERTa can be observed to perform the best. Furthermore, these models can be seen to outperform math-specialized algorithms.

Method	Comparison		Interval		Statistical		Temporal		Overall	
	F1	Acc.	F1	Acc.	F1	Acc.	F1	Acc.	F1	Acc.
NLI models										
`BART-large-MNLI`	51.52	51.37	63.68	64.84	58.77	59.34	**78.56**	79.5	64.58	64.81
`Roberta-Large-MNLI`	48.55	49.41	60.53	59.08	59.68	59.01	76.39	77.45	63.54	63.09
`deberta-v3-base-tasksource-nli`	53.2	52.94	**67.61**	**66.57**	60.93	60.0	76.86	77.45	65.67	64.97
Generative models										
`Flan-T5`	**55.14**	**55.29**	66.26	64.84	61.72	60.58	77.92	78.77	**66.79**	65.61
`GPT2`	46.33	47.06	56.44	54.76	56.21	55.79	72.59	75.55	60.26	60.16
`BART`	51.34	51.37	66.21	65.71	**61.79**	**61.24**	78.54	**79.65**	66.47	**65.89**
Math models										
`NumT5`	47.44	47.06	63.25	64.55	55.11	55.12	70.34	76.28	60.53	61.4
`MathRoberta`	43.31	45.49	58.27	56.2	52.26	51.07	70.32	71.3	57.61	56.75
`PASTA`	46.8	46.67	52.95	48.99	49.54	47.11	65.45	65.59	54.9	52.38
`Elastic BERT`	50.6	52.16	62.37	61.67	54.56	54.88	74.79	74.52	61.54	60.92
Dummy classifier (majority class)	16.32	32.94	49.2	63.4	34.26	50.83	62.66	73.79	41.43	57.03

For the second type of claim, we have observed that math models often mistake if the top evidence is similar to the claim but contains a different numerical value. In Table 2, both claim and evidence mention a drop in firearm homicides, yet with a different percentage drop. It is likely that math models focus more on the numerical value than other models and ignore the inconsistency in location, classifying the claim as *Conflicting*. Other models observe this problem and classify this claim correctly as *True*.

Type 3 claims are claims that were incorrectly classified by all models. Examples here included claims seemingly annotated incorrectly in the dataset or claims in different languages. An example of an incorrectly annotated claim is presented in Table 2. Both claim and evidence are formulated identically, yet the claim is annotated as *False*. Manually inspecting the dataset suggests that, indeed, the claim should be annotated as *True* as it also appears 1:1 in the golden evidence string. All the models predict this claim as *True*.

Interestingly, we have not been able to find any relation between the claims from the 1st type. One could expect that math models would outperform other models in cases where, for example, only the numerical value between claim and evidence is changed. We have not, however, observed any correlation like this. The analysis for this type is also limited, as only 3 claims exist in this category. Table 2 contains one of the three examples.

4.3 Claim decomposition

To perform the performance analysis of claim decomposition methods, we select the best-performing model from each category in terms of the overall F1 score. Thus, we evaluate DeBERTa (Sileo, 2024), Flan-T5 (Chung et al., 2022), and Elastic BERT (Liu et al., 2022). The results of this comparison can be seen in Table 3.

For DeBERTa, the best performance comes from not having any decomposition, with the second best being to decompose with GPT-3.5. For Flan-T5, decomposing with GPT-3.5 yields the best result, achieving 65.65% accuracy, which is also the highest score among all the models and decompositions. This is closely followed by having no decomposition on Flan-T5. Lastly, for Elastic BERT, fine-tuning from BART gives the best performance, followed by our custom decomposition.

Notably, our decomposition method significantly improves comparison and statistical claims. For comparison claims, the models improve between 4 to 9 percentage points in F1 score. Interestingly, this decomposition method also causes the models to perform worse on interval and temporal claims. As such, the overall result is worse than that with no decomposition.

Table 2: Qualitative predictions of 3 types of claims. Type 1 refers to claims incorrectly classified by all NLI and generative models but correctly by all math models, type 2 is vice-versa, and type 3 are claims that have been classified incorrectly by all models.

Type	Claim + Top 1 Evidence
1	**Claim:** "The president of the United States came in the Tuesday before the election in a ward that went 99% for him in the last election and they couldn't even (increase) the vote there." **Evidence:** this happened because president trump decided, even before election day 2020, that absentee ballots would be the source of fraud against him (although he and ...
2	**Claim:** Says since Australia passed tough gun laws, "homicides by firearm have declined almost 60 percent." **Evidence:** in 1995, connecticut tightened licensing laws, while in after tightening gun laws, firearm homicide rates dropped 40 percent in Connecticut.
3	**Claim:** "We welcome the decline in rhino poaching incidents since October 2015 which is for the first time in a decade." **Evidence:** we welcome the decline in rhino poaching incidents since october 2015, which is for the first time in a decade.

4.4 Temporal and numerical relevance

Similarly to the previous section, we selected for evaluation the best-performing model from each category: DeBERTa (NLI), Flan-T5 (Generative), and Elastic BERT (Math). We conducted two experiments to evaluate the effectiveness of these relevance measures by re-ranking evidence for specific types of claims:

- **Experiment 1:** *Temporal* claims (re-ranked using semantic and temporal relevance scores), *Statistical* claims (re-ranked using semantic and numerical relevance scores), *Comparison* & *Interval* claims (re-ranked using semantic scores only).
- **Experiment 2:** *Temporal* claims (re-ranked using semantic, temporal relevance, and numerical relevance scores), *Statistical* & *Interval* & *Comparison* claims (re-ranked using semantic scores only).

The results of adding additional temporal and numerical relevance measures are shown in Table 4. In general, we observe minor changes in accuracy in F1 score across the models, ranging from increases of 1% on average to extreme cases where the accuracy drops by over 8%. Notably, in the comparison and interval categories, both improvements and decreases are seen even though no additional relevance measures were applied. In contrast, slight decreases in performance are observed in the temporal and statistical categories, where additional relevance measures were implemented.

When we re-ranked evidence in Experiment 1, using both semantic and temporal relevance scores for temporal claims and using semantic and numerical relevance scores for statistical claims, the results showed slight improvements in certain models. However, these improvements were not consistent across all categories. For instance, the `deberta-v3-base-tasksource-nli` model showed slight increases in F1 and accuracy for temporal claims but mixed results for statistical claims. In Experiment 2, where we focused exclusively on temporal claims by re-ranking evidence using semantic, temporal, and numerical relevance scores, the improvements were minimal. The performance differences across models were generally small, indicating that the impact of these additional relevance measures is limited.

5 Discussion

5.1 Applicability of math models

Following the quantitative analysis from the previous section, we find that NLI and generative models outperform math-specialized models. Considering these models were explicitly designed to perform better on math-related problems, this might be a surprising result.

The qualitative analysis provides insight into potential reasons behind this phenomenon. We find that evidence, which is related to the claim contains an incorrect numerical value, it tricks the math models into believing the claim is false or conflicting. This explanation makes sense intuitively: models trained on mathematical data might tend to focus more on numerical values, ignoring facts that make the evidence invalid in the given case.

Table 3: Performance without claim decomposition and with various decomposition methods for the top 3 scoring models from Table 1. Bolded are the scores of the best-performing decomposition method for each model.

Method	Comparison		Interval		Statistical		Temporal		Overall	
	F1	Acc.	F1	Acc.	F1	Acc.	F1	Acc.	F1	Acc.
Original results										
deberta-v3-base-tasksource-nli	53.2	52.94	**67.61**	**66.57**	60.93	60.0	76.86	77.45	**65.67**	**64.97**
Flan-T5	55.14	55.29	66.26	64.84	61.72	60.58	77.92	78.77	**66.79**	65.61
Elastic BERT	50.6	52.16	62.37	61.67	54.56	54.88	74.79	74.52	61.54	60.92
ProgramFC										
deberta-v3-base-tasksource-nli	52.53	52.94	65.68	64.84	58.65	57.77	75.97	76.57	64.04	63.41
Flan-T5	55.94	56.47	64.2	63.98	59.86	59.01	77.48	78.48	65.13	64.77
Elastic BERT	45.2	46.27	60.67	59.65	57.25	56.28	**75.07**	74.96	61.72	60.84
Finetuning BART										
deberta-v3-base-tasksource-nli	47.73	48.63	65.72	65.13	59.41	59.17	**77.07**	**78.62**	64.17	64.25
Flan-T5	51.39	51.76	64.24	63.4	59.83	58.84	**78.45**	**79.06**	64.99	64.29
Elastic BERT	51.4	50.98	**63.48**	62.54	58.39	57.52	73.72	74.52	**62.9**	**62.2**
Prompting GPT-3.5										
deberta-v3-base-tasksource-nli	52.13	52.55	66.24	65.13	60.34	59.5	76.92	78.33	65.21	64.73
Flan-T5	58.19	**58.43**	**67.39**	**67.15**	61.19	60.74	75.7	76.28	66.05	**65.65**
Elastic BERT	52.01	52.16	59.88	60.23	54.96	54.79	74.94	**76.13**	61.18	61.12
Tailored Decomp (ours)										
deberta-v3-base-tasksource-nli	**61.0**	**61.81**	62.42	63.4	**61.37**	**61.32**	68.19	68.96	63.39	63.75
Flan-T5	**59.07**	57.09	63.86	62.54	**62.74**	**61.49**	68.1	67.06	63.98	62.71
Elastic BERT	**59.85**	**60.24**	62.18	**62.82**	**59.75**	**60.08**	64.93	65.15	61.52	61.87

Hence, overall, we find that verification of numerical claims given evidence should not be performed using math-specialized models. Instead, we find that modern NLI or generative models such as Flan-T5, BART, or DeBERTa should be used.

5.2 Claim Decomposition

Claim decomposition was expected to improve the overall model's performance but, ultimately, did not. We believe there are several reasons for this. Firstly, due to the token limit of 256, including both the claim and its decomposition takes tokens away from potentially important evidence. For future work, if access to stronger GPUs is available, it could be worth exploring a higher token length when decomposing. Secondly, some decompositions led to sub-questions that were not relevant to the claim. For example, with the fine-tuned BART model, let us consider the claim: "UK government banned Covid vaccine for children age 5-11" with the sub-questions:

- Did the UK government ban Covid vaccine for children age 5-11?
- Did the CDC ban the vaccine for kids age 5-11?

In this case, the second sub-question is irrelevant to the claim since the CDC is American. This not only leads to the gathering of irrelevant evidence but also, if the truthfulness of the two sub-questions differs, the overall prediction becomes incorrect. This issue was not seen with the GPT-3.5 decompositions, which might be why it had a performance comparable to not having any decomposition.

Additionally, our novel decomposition method, Tailored Decomp, was designed to enhance the accuracy and relevance of sub-questions generated from different types of claims but achieves a mixed performance across various claim types in practice. Interestingly, our method's superior performance on comparison claims might be partly due to the under-representation of these claims in the dataset. With comparison claims making up only 10.60% of the dataset (Viswanathan et al., 2024), traditional models and methods might not be as finely tuned to handle them. In contrast, Tailored Decomp's specialized prompts and decomposition strategy address the specific challenges of comparison claims more precisely than other methods.

Despite the success with comparison claims, our method shows similar or slightly worse results

Table 4: Comparison of performance for the top 3 scoring models from Table 1 without and with additional relevance measures. Bolded numbers indicate increased scores compared to original results. The second part of the table shows results where temporal claims' evidence was reranked using semantic and temporal relevance scores, while statistical claims' evidence was reranked using semantic and numerical relevance scores. The third part of the table presents results where only temporal claims' evidence was reranked using semantic, temporal, and numerical relevance scores.

Method	Comparison		Interval		Statistical		Temporal		Overall	
	F1	Acc.	F1	Acc.	F1	Acc.	F1	Acc.	F1	Acc.
Original results										
`deberta-v3-base-tasksource-nli`	53.2	52.94	67.61	66.57	60.93	60.0	76.86	77.45	65.67	64.97
`FlanT5`	55.14	55.29	66.26	64.84	61.72	60.58	77.92	78.77	66.79	65.61
`Elastic BERT`	50.6	52.16	62.37	61.67	54.56	54.88	74.79	74.52	61.54	60.92
Experiment 1										
`deberta-v3-base-tasksource-nli`	**53.92**	**53.73**	64.26	62.54	60.13	59.26	**78.23**	**78.33**	65.14	64.37
`FlanT5`	**57.07**	**56.86**	65.28	64.27	59.56	58.68	74.4	76.13	64.67	64.05
`Elastic BERT`	**50.73**	50.98	59.61	57.35	**55.85**	54.55	72.23	71.74	60.76	59.28
Experiment 2										
`deberta-v3-base-tasksource-nli`	49.68	50.59	63.69	61.96	59.27	59.01	**77.87**	**79.36**	64.23	64.13
`FlanT5`	47.12	47.45	64.75	62.82	60.45	59.34	76.09	76.87	64.35	63.41
`Elastic BERT`	48.36	48.24	55.6	53.03	**56.64**	**55.29**	73.71	73.35	60.6	59.2

compared to existing methods for the other claim types. Several factors may contribute to this outcome. While our prompt-based approach for few-shot learning is effective for comparison claims, it may require further refinement for other claim types. The specificity and structure of the prompts are crucial for generating relevant sub-questions, and slight misalignments can lead to less effective decompositions. Interval, temporal, and statistical claims often involve complex relationships and detailed contextual information. Our current decomposition method might not fully account for these complexities, leading to sub-optimal performance.

5.3 Limitations

The evaluation presented in this paper is limited and could be extended. Not all LMs have been evaluated on the claim verification task. We have attempted to set up other models, such as LUNA (Han et al., 2022), but faced difficulties in their evaluation. In the case of LUNA, the problem was the missing pre-trained weights, making fine-tuning the model impossible.

Additionally, the experiments could be repeated multiple times, allowing for significance tests. Certain results, such as the numerical and temporal relevance scoring, differ only slightly from the original results. Those differences could stem from noise, and performing significance tests would allow us to draw conclusions with confidence.

6 Conclusion and Future Work

In this work, we investigated techniques that could help fact-check numerical claims. To this end, we evaluated NLI, generative, and math models on the QuanTemp dataset. Surprisingly, we find that models specifically designed to handle numerical problems should not be used for numerical fact-checking. Instead, we suggest the use of generic LMs, such as Flan-T5 or BART, that were found to perform better on the QuanTemp dataset. We have furthermore found that although claim decomposition methods do not improve overall performance, custom decomposition rules can help in improving models for certain claim types. Finally, we also describe how additional metrics could be used to improve evidence re-ranking, although the results are deemed inconclusive.

Future work on this topic could involve evaluating more models, enabling a wider model comparison. Similarly, the current models could be rerun multiple times to obtain confidence bounds. Finally, other decomposition methods could be designed to tailor different claim types even better.

References

Rami Aly, Zhijiang Guo, Michael Sejr Schlichtkrull, James Thorne, Andreas Vlachos, Christos Christodoulopoulos, Oana Cocarascu, and Arpit Mittal. 2021. Feverous: Fact extraction and verifi-

cation over unstructured and structured information. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 1)*.

Yixuan Cao, Hongwei Li, Ping Luo, and Jiaquan Yao. 2018. Towards automatic numerical cross-checking: Extracting formulas from text. In *Proceedings of the 2018 World Wide Web Conference*, WWW '18, page 1795–1804, Republic and Canton of Geneva, CHE. International World Wide Web Conferences Steering Committee.

Jifan Chen, Aniruddh Sriram, Eunsol Choi, and Greg Durrett. 2022. Generating literal and implied subquestions to fact-check complex claims. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 3495–3516.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al. 2022. Scaling instruction-finetuned language models. *Journal of Machine Learning Research*, 25(70):1–53.

Anoushka Gade and Jorjeta Jetcheva. 2024. It's about time: Incorporating temporality in retrieval augmented language models. *arXiv preprint arXiv:2401.13222*.

Mor Geva, Daniel Khashabi, Elad Segal, Tushar Khot, Dan Roth, and Jonathan Berant. 2021. Did aristotle use a laptop? a question answering benchmark with implicit reasoning strategies. *Transactions of the Association for Computational Linguistics*, 9:346–361.

Zihui Gu, Ju Fan, Nan Tang, Preslav Nakov, Xiaoman Zhao, and Xiaoyong Du. 2022. Pasta: Table-operations aware fact verification via sentence-table cloze pre-training. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 4971–4983.

Hongwei Han, Jialiang Xu, Mengyu Zhou, Yijia Shao, Shi Han, and Dongmei Zhang. 2022. Luna: language understanding with number augmentations on transformers via number plugins and pre-training. *arXiv preprint arXiv:2212.02691*.

Matthew Honnibal, Ines Montani, Sofie Van Landeghem, and Adriane Boyd. 2020. spacy: Industrial-strength natural language processing in python.

Pegah Jandaghi and Jay Pujara. 2023. Identifying quantifiably verifiable statements from text. In *Proceedings of the First Workshop on Matching From Unstructured and Structured Data (MATCHING 2023)*, pages 14–22, Toronto, ON, Canada. Association for Computational Linguistics.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880.

Xiangyang Liu, Tianxiang Sun, Junliang He, Jiawen Wu, Lingling Wu, Xinyu Zhang, Hao Jiang, Zhao Cao, Xuanjing Huang, and Xipeng Qiu. 2022. Towards efficient NLP: A standard evaluation and a strong baseline. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3288–3303, Seattle, United States. Association for Computational Linguistics.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Ilya Loshchilov and Frank Hutter. 2018. Decoupled weight decay regularization. In *International Conference on Learning Representations*.

Meta AI. 2023. bart-large-mnli.

Christopher Mohri and Tatsunori Hashimoto. 2024. Language models with conformal factuality guarantees. *arXiv preprint arXiv:2402.10978*.

Vít Novotnỳ and Michal Stefánik. 2022. Combining sparse and dense information retrieval. In *Conference and Labs of the Evaluation Forum*, pages 104–118.

OpenAI. 2024. Deprecations.

Liangming Pan, Xiaobao Wu, Xinyuan Lu, Anh Tuan Luu, William Yang Wang, Min-Yen Kan, and Preslav Nakov. 2023. Fact-checking complex claims with program-guided reasoning. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6981–7004.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Abhilasha Ravichander, Aakanksha Naik, Carolyn Rose, and Eduard Hovy. 2019. EQUATE: A benchmark evaluation framework for quantitative reasoning in natural language inference. In *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*, pages 349–361, Hong Kong, China. Association for Computational Linguistics.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Namika Sagara and Ellen Peters. 2015. *Consumer Understanding and Use of Numeric Information in Product Claims*, pages 245–245. Springer.

Damien Sileo. 2024. tasksource: A large collection of NLP tasks with a structured dataset preprocessing framework. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 15655–15684, Torino, Italia. ELRA and ICCL.

James Thorne and Andreas Vlachos. 2017. An extensible framework for verification of numerical claims. In *Proceedings of the Software Demonstrations of the 15th Conference of the European Chapter of the Association for Computational Linguistics*, pages 37–40, Valencia, Spain. Association for Computational Linguistics.

James Thorne, Andreas Vlachos, Christos Christodoulopoulos, and Arpit Mittal. 2018. FEVER: a large-scale dataset for fact extraction and VERification. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 809–819, New Orleans, Louisiana. Association for Computational Linguistics.

Andrew Trotman, Antti Puurula, and Blake Burgess. 2014. Improvements to bm25 and language models examined. In *Proceedings of the 19th Australasian Document Computing Symposium*, pages 58–65.

Venktesh Viswanathan, Abhijit Anand, Avishek Anand, and Vinay Setty. 2024. Quantemp: A real-world open-domain benchmark for fact-checking numerical claims. *arXiv preprint arXiv:2403.17169*.

Andreas Vlachos and Sebastian Riedel. 2015. Identification and verification of simple claims about statistical properties. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 2596–2601, Lisbon, Portugal. Association for Computational Linguistics.

Soroush Vosoughi, Deb Roy, and Sinan Aral. 2018. The spread of true and false news online. *Science*, 359(6380):1146–1151.

Peng-Jian Yang, Ying Ting Chen, Yuechan Chen, and Daniel Cer. 2021. Nt5?! training t5 to perform numerical reasoning. *arXiv preprint arXiv:2104.07307*.

Xia Zeng, Amani S Abumansour, and Arkaitz Zubiaga. 2021. Automated fact-checking: A survey. *Language and Linguistics Compass*, 15(10):e12438.