Numerical and Temporal Fact Checking

May 3, 2024

1 Introduction

With the growth of digital tools for disseminating information, misinformation, and disinformation has also increased tremendously in recent years. The misinformation has serious consequences, such as initiating turmoil [13] in society. While numerous fact-checking efforts have been undertaken by websites like Politifact, snopes, Africa check, manual checking of growing misinformation is a cumbersome task.

Example: Claim from QuanTemp

Claim: `Under GOP plan, U.S. families making  $86k see avg tax increase of $794.`

[Evidence]: If enacted, the Republican tax reform proposal would saddle only 8 million households that earn up to $86,100 with an average tax increase of $794 .... Only a small percentage (6.5 percent) of the nearly 122 million households in the bottom three quintiles will actually face a tax increase.

[Verdict]: False

Figure 1: Example claim from QuanTemp

.

The case of real-world numerical claims Numerical claims are more important to verify as by the Illusion-of-Numeric-Truth effect [6], people tend to believe claims grounded in numbers more even if they are false. Individuals often manipulate numbers and add abstract logic to create a mirage of truth to make narratives in their favor. However, such claims can have various negative impacts on society. For instance, the opioid pandemic in America was believed to be caused by unsubstantiated, exaggerated claims made regarding the effectiveness of the drug [11]. In this project, we define numerical claims as those which *contain explicit or implicit numerical/temporal information and may require numerical operations for verification.*

Challenges in verifying numerical claims Verification of numerical claims requires numerical reasoning ability which involves understanding numeric patterns and mathematical concepts like arithmetic, numerical estimation, and data interpretation. While several works focus on automated verification of natural claims, only a handful of works[5, 8, 12, 9, 7, 4] focus on claims that require numerical reasoning. However, these techniques only focus on the detection of numerical claims[7][8] or are restricted to specific statistical properties[9][12]. Existing approaches handle numerical elements in natural claims as regular tokens without attempting to understand their significance.

A typical fact verification pipeline consists of three stages: claim detection, evidence retrieval, and reranking and veracity prediction.

The veracity predictions is usually performed using a fine-tuned model for Natural language Inference.

Numerical claims verification poses a unique challenge, where a fact-checking system must critically analyze and reason about the numerical data presented in both the claim and its evidence. For example, in verifying the claim shown in Figure 1 as ‘False’, the NLI model needs to identify that the evidence only mentions 8 million households with incomes up to $86k facing tax increases, contradicting the claim of tax increases for all families earning $86k.

In this project, the goal would be to improve the NLI model for veracity prediction in fact-checking pipeline with focus on numerical claims.

2 Research Questions

The following are the research questions we will aim to answer in this project:

RQ1: How does existing state of the art NLI models including LLM based models perform on different types of numerical claims in QuanTemp?

RQ2: Does special handling of numerical information improve performance for NLI models (fine-tuned and LLM based)?

RQ2: Does claim decomposition help in better evidence retrieval and downstream veracity prediction?

3 Experimental Setup

You will be working on the QuanTemp dataset collected for the purpose of numerical fact-checking. The dataset comprises 15K claims with train, val and test splits and an evidence collection with 420K snippets. The paper related to the dataset can be found at [10]. The data for this project can be found at https://anonymous.4open.science/r/NumTemp-E9C0. The repository also has a script that demonstrates off the shelf retrieval and veracity prediction. You can use the Table in NumTemp [10] as template for metrics and format in which results need to be reported.

You will perform the following experiments to answer the above research questions.

- Evaluate different NLI models while freezing the retrieval component. Some examples are BART-large-MNLI, Roberta-Large-MNLI, *sileod/deberta-v3-base-tasksource-nli*. Also consider generative models like FlanT5, GPT2, GPT3 (you can use API but evaluating on gpt3 is optional) and BART. Analyze the performance across different classifications of claims like temporal, statistical, comparison and interval claims given in the dataset. A reference for training NLI models can be found at https://colab.research.google.com/drive/1gZJCakmY28cKGMj8B7wd1GUM3r72pdbi?usp=sharing. While this script shows how you can use claim and justification document as evidence to form entailment, in your experiments you would retrieve evidence relevant to claim and train your classifier. You could choose to select top-k evidence and concatenate them to perform entailment with the claim. You can also choose alternate ways of performing inference where you perform entailment with claim and an evidence and aggregate predictions across evidences for a single claim for more fine-grained veracity prediction.

- Perform a qualitative analysis of randomly selected claims for which the predictions failed.

- Now evaluate using models which handle numbers well such as MathRoberta, NumT5, LUNA and PASTA [1] and Elastic Roberta [14] as backbone. Now analyze the performance across different categories of claims. Also perform a qualitative analysis of the same claims selected in the earlier step and check if their performance has improved.

- Additionally, you can also handle certain categories of claims such as temporal claims by using timestamp information in claims or by matching numerical information in claims and corresponding evidence. Some works that model temporality can be used as reference to implement appropriate relevance measures [3]. You can also incorporate these relevance scores in the loss sued to train the NLI model as a regularizing factor to cross entropy loss to penalize scenarios where the model

- Implement several claim decomposition methods like ClaimDecomp [2], ProgramFC [5] and use the decomposed questions to retrieve top-k evidence instead of original claim. train your NLI models on the combination of claim+questions to predict entailment with corresponding evidence. Analyze the impact of decomposed claims on overall Fact checking performance. Fine tune smaller models like BART or Flan-t5-large for decomposing the original claim using ClaimDecomp train set.

- Implement your own decomposition method with different style of decomposition based on category of claims. You could train a classifier on a training set of QuanTemp to infer type of claim (comparison, interval, temporal or statistical) during inference. Then the claim could be routed to the appropriate decomposition model. To train the decomposition models you can leverage training data from QA. For instance to train a decomposition model for comparison based claims, you could train a decomposition model on strategy QA dataset to generate subquestions from the original question. Or you could prompt models like LLama or Mistral to generate the questions or a training dataset. An example for training generative models like FlanT5 can be found at https://colab.research.google.com/drive/1wGr_iBmjFa7rP3KYw2vb6Vk3_4aVPWHP?usp=sharing. Report your findings and if the performance is better than using plain claim for evidence retrieval and veracity prediction.

In your report detail the intuition behind the performance changes and also qualitative analysis of few prominent examples. Also detail how the experiments answer the various research questions.

References

[1] M. Akhtar, A. Shankarampeta, V. Gupta, A. Patil, O. Cocarascu, and E. Simperl, "Exploring the numerical reasoning capabilities of language models: A comprehensive analysis on tabular data," in *Findings of the Association for Computational Linguistics: EMNLP 2023*, H. Bouamor, J. Pino, and K. Bali, Eds. Singapore: Association for Computational Linguistics, Dec. 2023, pp. 15 391–15 405. [Online]. Available: https://aclanthology.org/2023.findings-emnlp.1028

[2] J. Chen, A. Sriram, E. Choi, and G. Durrett, "Generating literal and implied subquestions to fact-check complex claims," 2022.

[3] A. Gade and J. Jetcheva, "It's about time: Incorporating temporality in retrieval augmented language models," 2024.

[4] P. Jandaghi and J. Pujara, "Identifying quantifiably verifiable statements from text," in *Proceedings of the First Workshop on Matching From Unstructured and Structured Data (MATCHING 2023)*, E. Hruschka, T. Mitchell, S. Rahman, D. Mladenić, and M. Grobelnik, Eds. Toronto, ON, Canada: Association for Computational Linguistics, Jul. 2023, pp. 14–22. [Online]. Available: https://aclanthology.org/2023.matching-1.2

[5] L. Pan, X. Wu, X. Lu, A. T. Luu, W. Y. Wang, M.-Y. Kan, and P. Nakov, "Fact-checking complex claims with program-guided reasoning," *arXiv preprint arXiv:2305.12744*, 2023.

[6] N. Sagara, *Consumer understanding and use of numeric information in product claims.* University of Oregon, 2009.

[7] D. Shah, K. Shah, M. Jagani, A. Shah, and B. Chaudhury, "Concord: Numerical claims extracted from the covid-19 literature using a weak supervision approach," *Available at SSRN 4222185*, 2023.

[8] P. Shah, A. Banerjee, A. Shah, B. Chaudhury, and S. Chava, "Numerical claim detection in finance: A weak-supervision approach," *TechRxiv preprint*, vol. 21288087, 2022.

[9] J. Thorne and A. Vlachos, "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*, A. Martins and A. Peñas, Eds. Valencia, Spain: Association for Computational Linguistics, Apr. 2017, pp. 37–40. [Online]. Available: https://aclanthology.org/E17-3010

[10] V. V, A. Anand, A. Anand, and V. Setty, "Numtemp: A real-world benchmark to verify claims with statistical and temporal expressions," 2024.

[11] A. Van Zee, "The promotion and marketing of oxycontin: commercial triumph, public health tragedy," *American journal of public health*, vol. 99, no. 2, pp. 221–227, 2009.

[12] A. Vlachos and S. Riedel, "Identification and verification of simple claims about statistical properties," in *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, L. Màrquez, C. Callison-Burch, and J. Su, Eds. Lisbon, Portugal: Association for Computational Linguistics, Sep. 2015, pp. 2596–2601. [Online]. Available: https://aclanthology.org/D15-1312

[13] S. Vosoughi, D. Roy, and S. Aral, "The spread of true and false news online," *science*, vol. 359, no. 6380, pp. 1146–1151, 2018.

[14] J. Zhang and Y. Moshfeghi, "Elastic: Numerical reasoning with adaptive symbolic compiler," 2022.