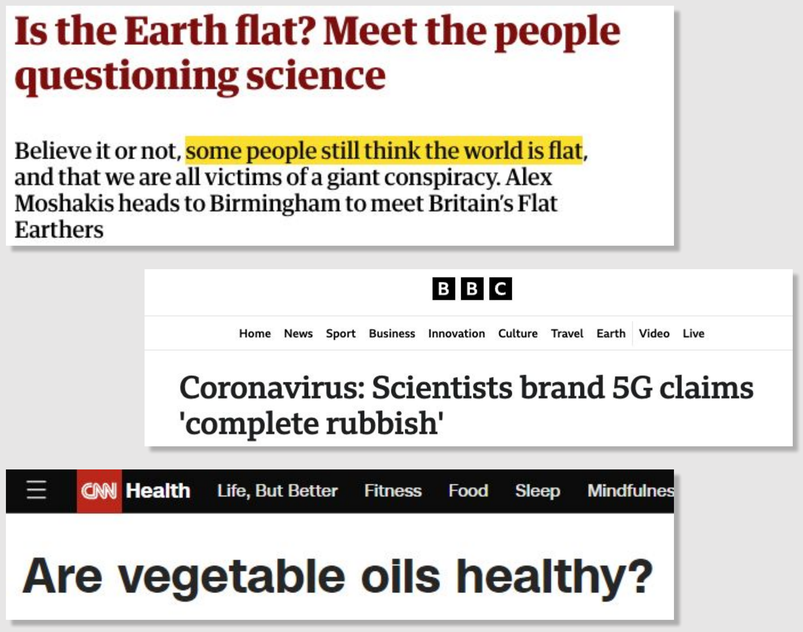
Disinformation on Social Media: Truthsayer

> The problem with quotes found on the internet is that they are often not true.
>
> — Abraham Lincoln

Ren Jeik Ong, Habib Bartu Gökalp, Niyazi Ulke, Fabian Greif, Bilgehan Emiral

Introduction

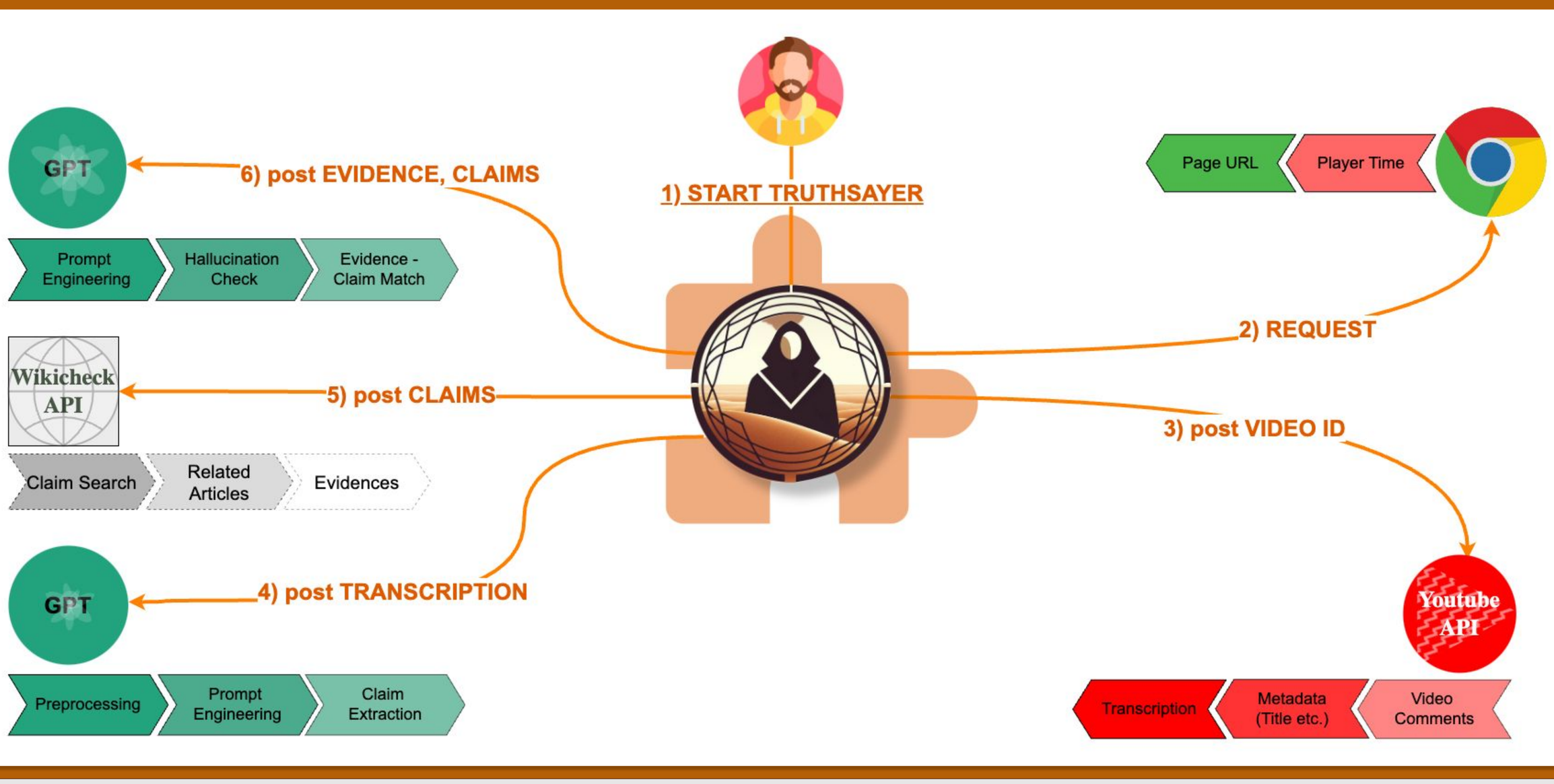
- The rise of digital media has led to an increase in the spread of disinformation, posing a threat to informed decision-making and societal trust.
- Our work explores advanced techniques in natural language processing to detect and inform users about potential disinformation.
- As part of this effort, we developed a web browser extension that automatically identifies misleading content in YouTube videos by using several tools.
- Truthsayer prioritizes transparency by allowing users to see the sources and rationale behind its outputs, enabling them to make informed decisions.



Methodology

- Youtube video content (transcript, title etc.) is retrieved through Youtube API.
- In addition, video transcript is extracted within a selected timestamp of 5-10 minutes context window.
- GPT 3.5 is used to extract claims from the content.
- Wikicheck API used to gathers related evidences from Wikipedia.
- The evidences and claims are fed to GPT 3.5 to receive a detailed misinformation evaluation.
- The browser extension outputs a clear output that is easy to interpret.

Architecture



Wikicheck API

Wikicheck is an open source api that automates fact-checking based on wikipedia. It uses Name Entity Recognition (NER) to find relevant wikipedia articles with their corresponding fragments that contain necessary information for factual evaluation. After browsing wikipedia, it predicts three scores to determine whether the gathered information supports, refutes, or not relevant to the target claim.

Future Work & Improvements

- Knowledge-graph representation for information that is out of vicinity of GPT or Google search query
- Integration of other fact checker tools (e.g. Google fact checker) instead of hardly relying on Wikicheck only
- Heuristic research for prompt-tuning/engineering
- Additional measures to handle hallucinations
- Evaluation on a dataset

Conclusion

We have a working product that requires a finalized architecture design. We aim to enhance its performance, particularly in terms of its correlation with real-world knowledge representation.

Prompt Engineering

- We included some custom prompt prefixes of instructions to GPT model. We specifically instructed GPT model to concatenate and classify Wikicheck claim results based on 3 labels of **SUPPORTS, REFUTES, NOT ENOUGH INFO** for each article claim in evidences.
- With the proper labelings generated by GPT, we can compute a simple score function.
- Besides prompt prefixes, we formalized a 1 short learning to give an example of input and output.
- We applied Chain-of-Thought to the prompt query such as "**Think step by step**".
- Furthermore, we added specific Temperature and Seeding to GPT query.

References

- Trokhymovych, M., & Saez-Trumper, D. (2021, October). Wikicheck: An end-to-end open source automatic fact-checking api based on wikipedia. In Proceedings of the 30th ACM International Conference on Information & Knowledge Management (pp. 4155-4164).
- Jurafsky, D., & Martin, J. H. (2024, February). Chapter 8: Sequence Labeling for Parts of Speech and Named Entities. In *Speech and Language Processing* (3rd ed.). Retrieved from https://web.stanford.edu/~jurafsky/slp3/8.pdf
- Jurafsky, D., & Martin, J. H. (2024, February). Chapter 10: Transformers and Large Language Models. In *Speech and Language Processing* (3rd ed.). Retrieved from https://web.stanford.edu/~jurafsky/slp3/10.pdf
- Brown, T., et al. (2020). Language models are few-shot learners. arXiv preprint arXiv:2005.14165.