Final Report: Iterative Retrieval Augmented Generation through Subclaim Generation for Efficient Multi-Hop Fact Verification

Saeyoung Choi
saeyoung.choi@snu.ac.kr

Jinwha Jang
gdejin@snu.ac.kr

Yan GuangJing
mysnuguangjing@snu.ac.kr

Abstract

Fact verification is an urgent problem which is even more crucial as the advent of generative language models. Retrieval augmented generation (RAG) approaches are combined with the large language models (LLM) to verify the factuality of claims with information retrieved from the knowledge base. However, the real-world sentences are often complex multi-hop claims, which are related to many different documents. To solve this problem, we propose the Iterative Retrieval Augmented Generation through Subclaim Generation model for efficient multi-hop fact verification. Instead of a single retrieval and generation, the proposed model conducts the iterative retrieval through generating subclaims. The model is evaluated with the fact verification datasets, using GPT-3.5-turbo as the base model. Although the proposed model shows higher performance than the simple retrieval model, it is inaccurate than the LLM-only model. In the further study, we should develop the method to process complicated relation between the different documents into the consistent context to verify the claim.

1 Introduction

As the cost of generating and delivering texts decreases, identifying misinformation and fact-checking becomes more important issues. Automatically generated misinformation is threatening to pollute the Internet. However, it is almost impossible to manually check the factuality of a massive number of texts generated by bots or language models. Hallucinations, plausible but incorrect text generation by language models, are also a difficult challenge to solve. Thus, the need for automated fact verification is urgent. With advances in natural language processing, attempts are being made to use various machine learning models for automatic fact verification [1, 2, 3].

Recently, retrieval-augmented generation (RAG) [4, 5] is drawing attention for its potential in improving factuality of the language models [6, 7]. It can not only improve factuality but also provide explainability of the generated results, by utilizing both LLM's prior knowledge (parameters) and external knowledge (retrieved document). However, it is difficult to verify complex multi-hop claims. To verify multi-hop claims, the model should find several different relevant documents.

In this study, we propose a generative approach to enhance the model performance and efficiency, by utilizing generative capability of the language model for not only claim verification, but also retrieval. This approach includes (1) subclaim generation to verify the complex factuality within multi-hop claims, (2) document retrieval to obtain relevant evidence for each claim, and (3) claim verification based on the retrieved context. We will integrate them into our baseline model, and evaluate the performance of the integrated model.

2 Related Work

Iterative retrieval-generation synergy (ITRG) [8] is the iterative retrieval approach based on the LLM. Unlike the simple RAG model, it queries again with the generated results. Through this iterative procedure, the complex question related to the many different documents can be answered step by step. It can be considered as generating relevant documents by simultaneously exploiting parametric and non-parametric knowledge.

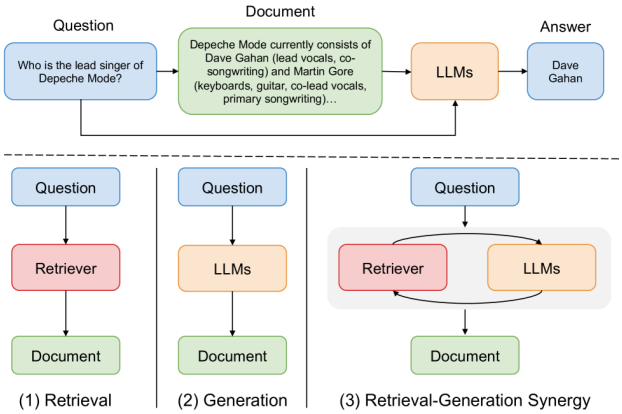

Figure 1: The Architecture of the ITRG System

ITRG consists of two parts: generation augmented retrieval and retrieval augmented generation. In generation augmented retrieval, it concatenates the original question and the context, and uses it as query to retrieve the related documents. Then, in retrieval augmented generation, it reinforces the context with the generated results. By iterating this retrieval-generation-reinforcement process, the model can generate the context that integrates information from different documents.

QACheck [9] adopts a question-guided multi-hop reasoning framework, significantly enhancing the clarity and effectiveness of fact-checking processes. Unlike traditional models that rely on linear evidence assessment, QACheck dissects complex claims into simpler components through a series of contextually relevant questions. This methodology not only increases the interpretability of the verification process but also aligns closely with human cognitive patterns in assessing veracity.

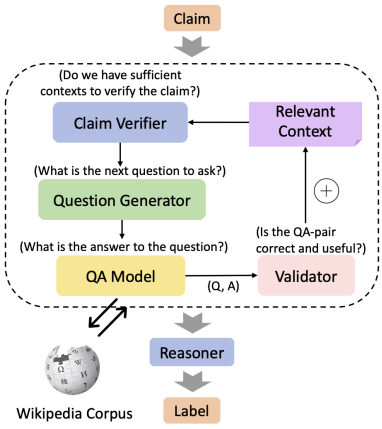

Figure 2: The Architecture of the QACheck System

The architecture of QACheck is a composite of five key modules: a claim verifier, a question generator, a question-answering module, a QA validator, and a reasoner. Each module plays a distinct yet interdependent role in the fact-checking pipeline. The claim verifier initially assesses the sufficiency of available context in substantiating a claim. Sequentially, the question generator and the QA module

collaboratively work to explore and fill informational gaps. This synergy ensures a comprehensive examination of claims, thereby enhancing the accuracy of verification.

3 Model Description

3.1 Model Overview

The fact verification task is to verify the factuality of the given claim. The proposed model utilizes the embedding model and the vector database for transforming, storing, and searching. LLM is used as the subclaim generator and the claim verifier. Figure 3 and Figure 4 shows the architecture of our baseline models and proposed model, respectively.

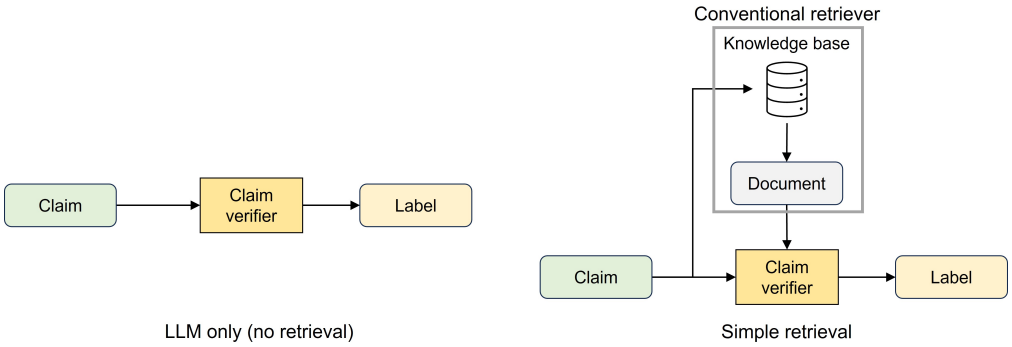

Figure 3: Baseline Model Architecture

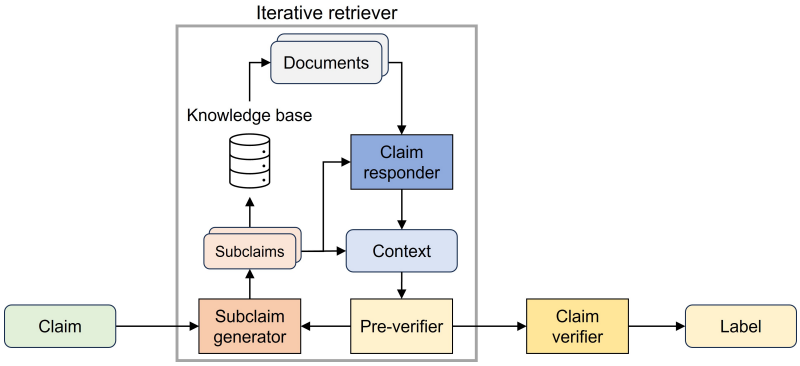

Figure 4: Proposed Model Architecture

3.2 Baseline model

Two baseline models are presetned: LLM only (no retreival) and simple retrieval. In LLM only model, the LLM simply takes the given claim and verify it with its prior knowledge. The simple retrival model consists of retriever and claim verifier. The retriever first retrieves the documents that are similar to the given claim, by querying to the knowledge base. Claim and retrieved documents are put into the claim verifier, which then returns the factuality label (true or false).

3.3 Proposed model

While the simple retrieval model has conventional retriever and claim verifier, our proposed model consists of the following modules: (1) subclaim generator, (2) claim responder, (3) pre-verifier, and (4) claim verifier.

First, the subclaim generator breaks down complex claims into contextually relevant subclaims, through iterative procedure. At each 'round', the subclaim generator makes new subclaim based on

the original claim and context. Then, the claim responder answers to the generated subclaim through retrieval augment generation. All generated subclaims and their responses are incorporated into the context. Then, pre-verifier determines wheter the claim is verifiable with the given context. If not, it passes the context to the subclaim generator to generate new subclaim. If the context is sufficient, claim verifier makes the final verification.

4 Experiment Setup

Our model is constructed based on LlamaIndex [10], a framework for LLM-based application. 5,416,537 Wikipedia documents, which is the knowledge base from FEVER [11] and HOVER [12] dataset, are stored in vector database (pgvector on PostgresSQL), after transformed to embedding vectors with BAAI's embedding model [13]. When a claim is given, it is transformed to a query embedding with the same sentence transformer. Then, the documents are retrieved from the knowledge base based on the similarity with the query embedding. The base model for subclaim generator, claim responder, pre-verifier, and claim verifier is GPT-3.5-turbo model [14].

4.1 Dataset

The proposed model is tested on claim samples from FEVER and HOVER datasets. They are open-domain fact verification dataset based on Wikipedia articles. While FEVER mostly has single-hop claims (related to a single document), HOVER has multi-hop claims (related to multiple documents). It should be noted that HOVER dataset has only two labels (Supported, Not supported) while FEVER dataset has three labels (Supports, Reputes, Not enough information) because of ambiguity between False and Not Verifiable in multi-hop claims. It is clearly shown that the conventional retrieval model fails to correctly verify the claims with multi-hop. Thus, in this study, we exclude the samples with Not enough information label in FEVER dataset for consistent comparison of the performance.

4.2 Evaluation

Overall fact verification performance of the model is evaluated through the F1 score. Retrieval performance is evaluated with the following retrieval score (RS).

$$\texttt{Retrieval score} = \frac{\sum_{i=1}^{N} s_i}{N}$$

where N is the number of samples and s_i is the score for the sample. In FEVER, if any of 'evidence' documents (documents when the sample claims are related) is retrieved successfully, $s_i = 1$. In HOVER, which consists of multi-hop claims, the number of evidence documents retrieved is divided by the number of hops. For example, if only 1 evidence documents are successfully retrieved for 4-hop claim, $s_i = 1/4$.

In previous literature, the reasoning (explanation) of the fact verification was evaluated manually by human experts [15, 16]. Some other literature [17] provide the potential of LLM as a human-like evaluator. However, in this study, reasoning sentences are not evaluated.

5 Result

5.1 Fact Verification Performance

Fact verification performance, evaluated with F1 score, is shown in Table 1. We compare three different models: LLM only (no retrieval), simple retrieval, and our proposed model. For 1-hop claims from FEVER dataset, proposed model achieve the highest F1 score. For multi-hop claims from HOVER dataset, the LLM only model has the best performance, while the proposed model shows higher performance than the simple retrieval model.

It can be interpreted that the LLM with the sufficient number of parameters have enough prior knowledge to verify the given claims. However, it still has the potential for improving the performance by the appropriate retrieval, as shown in 1-hop claims.

Low F1 score on the HOVER dataset can be explained with 'not enough information'. When the complex claim is given, LLM often reasons that the claim cannot be verified because of insufficient

information (then answering as 'false'). The proportion of such cases (where reasoning includes 'cannot be verified', 'not possible', 'no evidence', or 'no mention') are shown in Table 2.

	FEVER	HOVER		
	1-hop	2-hop	3-hop	4-hop
LLM only (no retrieval)	0.836	0.727	0.609	0.424
Simple retrieval	0.824	0.533	0.457	0.182
Proposed model	0.914	0.719	0.588	0.286

Table 1: Fact verification performance (F1 score) of different models on FEVER and HOVER datasets

	HOVER		
	2-hop	3-hop	4-hop
Simple retrieval	0.694	0.786	0.953
Proposed model	0.404	0.608	0.765

Table 2: The ratio of 'not enough information' reasoning in HOVER dataset

5.2 Retrieval Performance

The retrieval performance is evaluated to investigate the influence of hyperparameters. The retrieval score, which indicates how well 'evidence' documents are retrieved, is analyzed as shown in Figure 5 and Figure 6. k is hyperparameter that determines how many documents are retrieved in each retrieval (top-k). From Figure 5, retrieval score increases gradually with increasing k. Regardless of hops, the number of documents should be decided carefully. Note that there may be trade-off due to an increase in the number of input context tokens. Figure 6 shows that the relationship between the maximum number of rounds and the retrieval score. It implies that the required number of rounds increases as the claim becomes complex.

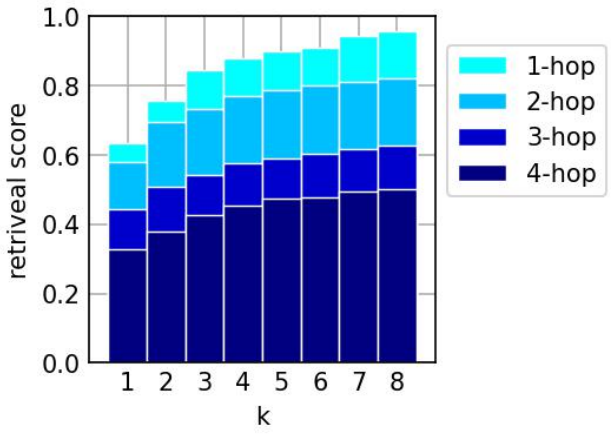

Figure 5: The retrieval score with the different number of retried documents (when number of rounds is 5)

6 Conclusion

In this study, we proposed the Iterative Retrieval Augmented Generation through Subclaim Generation model for complex fact verification problems. The fact verification performance and sensitivity to the hyperparameters were evaluated through the experiments. Although the proposed method was

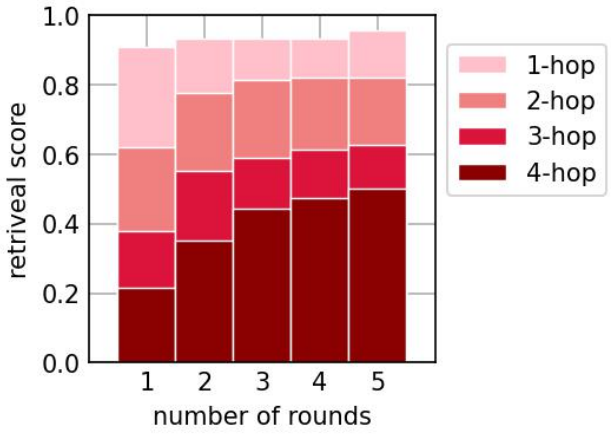

Figure 6: The retrieval score with the maximum number of rounds (when $k = 8$)

always more accurate than the simple retrieval model, it performed worse than LLM only model in multi-hop claims. Thus, we need to further investigate the followings:

- How to generate better subclaims for complex claims
- How to process multiple retrieved documents into the consistent context

To generate better subclaims, structured methods like entity recognition for claim decomposition could be effective. Using the chain-of-thought reasoning as subclaim may be helpful, but it sometimes reinforce the hallucination through iterative process.

There are several limitations in this study. The results are dependent on the specific model (GPT-3.5-turbo). We should check whether the result (performance improvement) is rigid with different embedding models, database, and LLM. Trade-off between the LLM size (the number of parameters) and the performance must be investigated. Also, the proposed approach assumes that the documents retrieved from the knowledge base is always true. However, it is very costly to construct and maintain 'clean and truthful' knowledge base. Nowadays, search-based approach is becoming popular. Thus, the factuality of the retrieved documents (and logical consistency) should be also verified, using other documents and prior knowledge.

References

[1] Sahil Chopra, Saachi Jain, and John Merriman Sholar. Towards automatic identification of fake news: Headline-article stance detection with lstm attention models. *Proc. Stanford CS224d Deep Learn. NLP Final Project*, pages 1–15, 2017.

[2] Wanjun Zhong, Jingjing Xu, Duyu Tang, Zenan Xu, Nan Duan, Ming Zhou, Jiahai Wang, and Jian Yin. Reasoning over semantic-level graph for fact checking. *arXiv preprint arXiv:1909.03745*, 2019.

[3] Yixin Nie, Haonan Chen, and Mohit Bansal. Combining fact extraction and verification with neural semantic matching networks. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pages 6859–6866, 2019.

[4] Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems*, 33:9459–9474, 2020.

[5] Gautier Izacard and Edouard Grave. Leveraging passage retrieval with generative models for open domain question answering. *arXiv preprint arXiv:2007.01282*, 2020.

[6] Kurt Shuster, Spencer Poff, Moya Chen, Douwe Kiela, and Jason Weston. Retrieval augmentation reduces hallucination in conversation. *arXiv preprint arXiv:2104.07567*, 2021.

[7] Ruiyang Ren, Yuhao Wang, Yingqi Qu, Wayne Xin Zhao, Jing Liu, Hao Tian, Hua Wu, Ji-Rong Wen, and Haifeng Wang. Investigating the factual knowledge boundary of large language models with retrieval augmentation. *arXiv preprint arXiv:2307.11019*, 2023.

[8] Zhangyin Feng, Xiaocheng Feng, Dezhi Zhao, Maojin Yang, and Bing Qin. Retrieval-generation synergy augmented large language models. *arXiv preprint arXiv:2310.05149*, 2023.

[9] Liangming Pan, Xinyuan Lu, Min-Yen Kan, and Preslav Nakov. Qacheck: A demonstration system for question-guided multi-hop fact-checking. *arXiv preprint arXiv:2310.07609*, 2023.

[10] LlamaIndex. https://www.llamaindex.ai/.

[11] James Thorne, Andreas Vlachos, Christos Christodoulopoulos, and Arpit Mittal. Fever: a large-scale dataset for fact extraction and verification. *arXiv preprint arXiv:1803.05355*, 2018.

[12] Yichen Jiang, Shikha Bordia, Zheng Zhong, Charles Dognin, Maneesh Singh, and Mohit Bansal. Hover: A dataset for many-hop fact extraction and claim verification. *arXiv preprint arXiv:2011.03088*, 2020.

[13] Shitao Xiao, Zheng Liu, Peitian Zhang, and Niklas Muennighoff. C-pack: Packaged resources to advance general chinese embedding, 2023.

[14] OpenAI GPT-3 API [gpt 3.5-turbo]. https://platform.openai.com/docs/models/gpt-3-5.

[15] Xuan Zhang and Wei Gao. Towards llm-based fact verification on news claims with a hierarchical step-by-step prompting method. *arXiv preprint arXiv:2310.00305*, 2023.

[16] Haoran Wang and Kai Shu. Explainable claim verification via knowledge-grounded reasoning with large language models. *arXiv preprint arXiv:2310.05253*, 2023.

[17] Tom Kocmi and Christian Federmann. Large language models are state-of-the-art evaluators of translation quality. *arXiv preprint arXiv:2302.14520*, 2023.