Divide & Conquer for Entailment-aware Multi-hop Evidence Retrieval

Anonymous ACL submission

Abstract

Lexical and semantic matches are commonly used as relevance measurements for information retrieval. Together they estimate the semantic equivalence between the query and the candidates. However, semantic equivalence is not the only relevance signal that needs to be considered when retrieving evidences for multi-hop questions. In this work, we demonstrate that textual entailment relation is another important relevance dimension that should be considered. To retrieve evidences that are either semantically equivalent to or entailed by the question simultaneously, we divide the task of evidence retrieval for multi-hop question answering (QA) into two sub-tasks, i.e., semantic textual similarity and inference similarity retrieval. We propose two ensemble models, EAR and EARnest, which tackle each of the sub-tasks separately with off-the-shelf retrieval models, and jointly retrieve sentences with the consideration of the diverse relevance signals. Experimental results on HotpotQA verify that our models not only significantly outperform all the single retrieval models it is based on, but is also more effective than two intuitive ensemble baseline models.

Question: What nationality was James Henry Miller's wife?
Answer: American

Supporting Evidences

Ewan MacColl
(1) James Henry Miller (25 January 1915 – 22 October 1989), better known by James Henry Miller stage name Ewan MacColl, was an English folk singer, songwriter, communist, labour activist, actor, poet, playwright and record producer .

Peggy Seeger
(2) Margaret "Peggy" Seeger (born June 17 , 1935) is an American folksinger.
(3) She is also well known in Britain, where she has lived for more than 30 years, and was married to the singer and songwriter Ewan MacColl until his death in 1989.

Semantic Equality
James Henry Miller (Q) $\approx$ James Henry Miller (25 January 1915 – 22 October 1989), better known by James Henry Miller stage name Ewan MacColl (1)

Textual Entailment
American (2) $\vdash$ nationality (Q)
was married to (3)$\vdash$ wife (Q)

Figure 1: An example from the HotpotQA dataset (Yang et al., 2018) showing the two different dimensions of relevance between the question and its supporting evidences.

1. Introduction

Widely adopted QA approaches use a two-stage pipeline, i.e., a retriever module followed by a reader module (Chen et al., 2017). The retriever is responsible for collecting relevant contextual evidence fragments; then the reader module combines the relevant information from the retriever module to infer the answer. While it is common to utilize an inference model as a reader to infer the correct answer from the retrieved context, most existing retrievers only focus on lexical and/or semantic matches, ignoring the inference relations between question and context.

According to formal semantic notions, the semantic relationship between two text fragments includes semantic equivalence, referential equality, and textual entailment. While referential equality can be mostly solved by coreference resolution and entity linking, semantic similarity and textual entailment require deep semantic understanding between question and context. Most researches use semantic similarity as a shorthand to refer to the well-known Semantic Textual Similarity (STS) tasks, where semantic similarity is operationally defined by the annotation guidelines, which fall around the notion of semantic equivalence, i.e., “Do these two sentences mean the same thing?” (Lin et al., 2021b) While textual entailment is a framework that captures semantic inference. Textual Entailment (TE) of two text fragments can be defined as the task of deciding whether the meaning of one text fragment can be inferred from another text fragment. That is, a premise T entails a hypothesis H if, typically, a human reading T would infer that H is most likely true. For example, T: Jack sold the house to Peter. H: Peter owns the house. Here H can be inferred from T, so T entails H.

Most of the evidence retrieval works only measure the semantic textual similarity between the question and candidate corpus to determine the relevance. This works for single-hop questions, in which relevant information usually share the same entity mentions with the question. However, it is not sufficient for multi-hop questions. For multi-hop questions, the relevant relationship between question and evidence(s) is beyond the lexical or semantic similarity that is targeted by most retrievers, especially for secondary hops.

The evidence retrieval for the multi-hop QA task broadly involves two different dimensions of relevance to measure: semantic equivalence and textual entailment. Thus, we divide the multi-hop QA evidence retrieval task into two separate retrieval tasks, where each aims to retrieve a subset of sentences that score highly on one of the relevance dimension respectively, and then combine them to output the final ranking with an ensemble model. To build a retrieval method that does not rely on a large training set with evidence annotations, we use both off-the-shelf statistical models and pre-trained language models as base models to capture the diverse relevance signals.

Our contributions in this work include: (1) We call attention to the fine-grained aspects of relevance for multi-hop QA evidence retrieval. In particular, textual entailment relation should be taken into consideration along with semantical equivalence in order to cover a more accurate relevant context. (2) We propose two ensemble models that combine diverse relevance signals captured by three off-the-shelf base models. Our experimental results demonstrate that not only are the individual base retrieval models necessary in evidence retrieval but cooperate advantageously to produce a better ranking for multi-hop QA evidence retrieval when used together. (3) We empirically show the effectiveness of the proposed ensemble retrieval models by evaluating on the HotpotQA dataset, and show they outperform all the base models and also several ensemble baselines.

2. Related Work

2.1. Text Retrieval

Traditional retrieval models such as TF-IDF and BM25 (Trotman et al., 2014) use sparse bag-of-words representations to collect lexical matching signals (e.g., term frequency). Such sparse retrieval models are mostly limited to exact matches. **Dense Retrieval models** move away from sparse signals to dense representations, which help address the vocabulary mismatch problem. These models can be categorized into two types according to their model architecture, *representation-based* (Huang et al., 2013; Shen et al., 2014) and *interaction-based* models (Pang et al., 2016; Lu and Li, 2013; Guo et al., 2016; Mitra et al., 2017). **Hybrid methods** (Lin et al., 2021a; Gao et al., 2020; Karpukhin et al., 2020; Shan et al., 2020) aggregate the dense retrieval with sparse retrieval methods to better model relevance. The entailment-aware retrieval models we propose are also hybrid methods that combine sparse and dense retrieval methods, but our method is unsupervised. Further, we combine a sparse model with multiple dense models in order to consider diverse relevance signals, i.e., textual entailment in addition to semantic equivalence.

2.2. Multi-hop Evidence Retrieval

Research works on multi-hop evidence retrieval can be broadly categorized into two directions: **(1) Question decomposition:** (Min et al., 2019; Jiang and Bansal, 2019; Fu et al., 2021; Perez et al., 2020; Talmor and Berant, 2018) decompose multi-hop questions into multiple single-hop sub-questions. Instead of training a decomposer, our entailment-aware retrieval models tackle this task in the opposite direction. That is, we assemble candidate sentences pairs that carry different relevance signals to match against the question. **(2) Iterative evidence retrieval:** Feldman and El-Yaniv (2019) proposed a method to iteratively retrieve supporting paragraphs, using the paragraphs retrieved in previous iteration to reformulate the search vector. (Asai et al., 2020) iteratively retrieve a subsequent passage in the reasoning chain with an RNN. Qi et al. (2019) trained a retriever to generate a query from the question and the available context at each step. While iterative retrieval considers evidence retrieval as a sequence process, so that the accuracy of subsequent retrieval steps highly depends on previous decisions, our method jointly consider high potential sentences pairs simultaneously.

3. Methodology

In this section, we introduce two ensemble models for entailment-aware multi-hop QA evidence retrieval. At a high-level, we model diverse relevance relationships with three base models, and combine the relevance signals they capture to jointly retrieve candidate evidences for multi-hop questions.

3.1. Task

Given a multi-hop question Q and a corpus $C = \{P_1, P_2, \ldots, P_m\}$ containing a set of documents or paragraphs, the evidence retrieval task is to rank the candidate text sentences (the “unit of indexing") from C and return a list of top N most relevant ones that provide sufficient and less distracting information for answering Q. Estimating the degree of relevance of each candidate sentence to the question is clearly an integral part of the task. To build a better retriever for multi-hop questions, two dimensions of relevance (i.e., semantic equivalence and entailment) need to be both considered in order to provide a more accurate estimation of relevance for each candidate sentence.

We divide this multi-criterial task into two separate ranking subtasks, which include semantic equivalence as well as textual entailment. [1] Both tasks require comparing information between the question and candidates, but the objectives of the comparison are different. In this work, we propose to capture the entailment relations in parallel with the semantic equivalence with separate models, which produce different and potentially conflicting rankings. The goal is to combine them to figure out an aggregated ranking that promote gold evidence sentences to the top of the list.

3.2. Base Models

We chose three off-the-shelf base models to capture diverse relevance patterns. To better estimate semantic equivalence, we use both a sparse model (i.e., BM25) and a dense model (i.e., transformers pre-trained for semantic search) to examine exact match and semantic match respectively. In addition, we utilized another dense model pre-trained on QNLI dataset for capturing entailment relation. For the dense models, we choose two pre-trained cross-encoders (CE)[2] (Reimers and Gurevych, 2019), which are trained by taking concatenated question and candidate sentence as a single input sequence and generate an estimate of relevance score directly. The two pre-trained cross-encoders we choose are:

MSMARCO Passage Cross-Encoder[3] is trained on the MS Marco Passage Ranking dataset (Bajaj et al., 2016) for information retrieval. MS MARCO (Microsoft Machine Reading Comprehension) is a large scale corpus consists of about 500k real search queries from the Bing search engine with 1000 most relevant passages. The model is trained to rank the most relevant passage that answers the query labeled by human as high as possible.

QNLI Cross-Encoder[4] is a pre-trained model obtained using the Question Natural Language Inference (QNLI) dataset introduced by GLUE Benchmark (Wang et al., 2018). QNLI was automatically derived from SQuAD, with the processing target of question-answer entailment.

BM25	MSMARCO CE	QNLI CE	% Ques (k=3)	% Ques (k=5)
✓	✗	✗	14	10
✗	✗	✓	25	22
✗	✓	✗	20	16
✗	✓		33	30
✗		✓	38	35
	✓	✗	64	62
	✗	✓	35	33
✗	✗	✗	44	29

Table 1: Each line shows the percentage of questions that have at least one evidence ranked within top-k by the model marked with a ‘✓’ but beyond top-k by model(s) marked with a ‘✗’. For example, there are 14% questions with at least one evidence sentence is ranked within top-3 by BM25[5] , but ranked beyond top-3 by MSMARCO CE and QNLI CE; 35% questions with at least one evidence sentence ranked within top-3 by QNLI CE but ranked beyond top-3 by MSMARCO CE.

3.3. Ensemble Models

Since the three base models aforementioned independently capture diverse relevance signals and are complementing each other as shown in table 3.2,

[1] For the focus of this work, we conduct coreference resolution within each paragraph before the retrieval task.

[2] While usually much faster, bi-encoders are less effective than cross-encoder models because the latter can exploit relevance signals derived from attention between the query and candidate sentence at each transformer encoder layer.

[3] https://www.sbert.net/docs/pretrained-models/ce-msmarco.html

[4] https://www.sbert.net/docs/pretrained_cross-encoders.html#squad-qnli

[5] https://pypi.org/project/rank-bm25

an ensemble model should potentially improve the final retrieval performance if an appropriate aggregation strategy designed to combine them.

3.3.1. Ensemble Baselines

Average ranking (AR) is a simple ensemble ranking model which combines the ranking outputs from the multiple base models that ranks all candidate sentences independently. A rank of a candidate sentence obtained by each base model is with respect to the relevance signal the base model targets on. Thus, each sentence has M ranks (where M is the number of base models). The final ranking is obtained by sorting the *sum* of all M rankings that each sentence received.

Similarity Combination (SimCom) calculates hybrid retrieval scores through a linear combination of sparse and dense scores. For a given question, we aggregate the scores produced by the base models through a *weighted average* for each candidate sentence, called Question Evidence Relevance (QER) (see the equation in Appendix A). QER are then used to rank the candidate evidence sentences, so that candidate sentences with high relevance to the question are promoted to the top of the list.

3.3.2. Entailment-Aware Retrieval

In this work, we propose an entailment-aware retrieval (EAR) method to jointly consider pairs of candidate sentences that potentially contain complementary relevance signals. We form such sentence pairs using the Cartesian product of two sets of top ranked candidate sentences with respect to semantic equivalence and textual entailment correspondingly. While BM25 and MSMARCO Cross-Encoder capture exact and semantic matches, respectively, they both aim for estimating STS. Thus, we take the union of top ranked sentences by BM25 and MSMARCO Cross-Encoder as a unified set $\mathcal{A} = \{S_{a_1}, S_{a_2}, S_{a_3}, S_{a_4}\}$, and top ranked sentences by QNLI Cross-Encoder as another set $\mathcal{B} = \{S_{b_1}, S_{b_2}, S_{b_3}\}$. The pairs we consider are $\mathcal{P} = \mathcal{A} \times \mathcal{B} = \{(\mathbf{a}, \mathbf{b}) \,|\, \mathbf{a} \in \mathcal{A} \wedge \mathbf{b} \in \mathcal{B}\}$.

We then concatenate the two sentences of each pair as a sequence to score against the question with a reranker[6], such that the top scored sentence pair (S_{a_i}, S_{b_j}) is most likely to form a compositional relevant context covering both semantic equivalence and entailment relevance signals. When S_{a_i} and S_{b_j} are examined individually, there is high chance that S_{a_i} receives a low IS score from the QNLI cross-encoder and ranked down to the list, S_{b_j} can be scored and ranked low by BM25 and MSMARCO cross-encoder. Thus, either using individual base models or aggregating ranking or scores with the ensemble baseline models, S_{a_i} and S_{b_j} are unlikely to be both promoted to the top of the ranking list. Finding the best combination from the top-ranked subsets with respect to both semantic equivalence and textual entailment efficiently takes the compositional requirement into consideration. We further concatenate the question q with the pair S_{a_i} and S_{b_j} as a new query to re-rank the rest of the candidate sentences.

3.3.3. EARnest

Evidences for a multi-hop question should be intuitively related, and often logically connected via a shared named entity that would allow a human reader to connect the information they contain. The presence of a shared named entity between two candidate sentences often indicates the likelihood that the sentence pairs relate to each other and, thus, they can be connected to form a coherent context for the question.

To leverage such connection as an additional cue, we add a named entity similarity term (NEST) to the scoring function of the reranker in EAR when estimate the top scored sentence pairs as

$$QER_{Earnest} = (1 + NEST) * Sim(q,\ s_i \,\|\, s_j)$$

where $Sim()$ is the scoring function of the reranker, which scores the concatenation of sentence pair S_i and S_j against the question. $NEST$ is a binary switch, that is, if the two sentences share one or more named entity, the promotion mechanism is activated; otherwise it is deactivated.

Named Entity Similarity Term Besides using SpaCy (Honnibal et al., 2020) to recognize named entities with common entity types (such as names of people, places, organizations), we also consider titles of documents and phrases between a pair of single or double quotes. When comparing whether two sentence share an entity, we apply basic normalization (i.e., lower case, removing articles and special punctuations) and fuzzy match to tolerate typo, variations, and inclusive match.

[6] We use the MSMARCO Cross-Encoder as the reranker since it is the best performing base model.

4. Evaluation and Results

4.1. Dataset

We conduct our evaluation using HotpotQA dataset (Yang et al., 2018). HotpotQA contains two question categories: *bridge-type questions*, in which an intermediate entity is needed to be retrieved before inferring the answer; and *comparison-type questions*, which compare two provided entities. Given the focus of this work, we use solely the bridge-type questions in our evaluation.[7] We conduct the evaluation of our proposed methods on the 5918 bridge-type questions out of the 7,405 examples from the development partition of HotpotQA dataset in the distractor setting. Each question in HotpotQA is supported by two documents, and provided with ground-truth supporting sentences, which enables us to evaluate the evidence retrieval performance of the various models.

Models	P@3	P@5	MAP	R@3	R@5	R@10
Base models						
BM25	0.43	0.31	0.59	0.54	0.65	0.78
MSmarco	0.47	0.33	0.64	0.59	0.69	0.81
QNLI	0.33	0.25	0.46	0.43	0.52	0.65
Ensemble Baselines						
AR	0.43	0.31	0.61	0.55	0.66	0.83
SimCom	0.5	0.36	0.68	0.63	0.74	0.86
Our Approach						
EAR	0.53	0.36	0.71	0.66	0.76	0.86
EARnest	**0.55**	**0.38**	**0.74**	**0.7**	**0.78**	**0.87**

Table 2: Evidence retrieval results of base models, baseline ensembles, and our methods on HotpotQA. As can be seen, the performance of our proposed ensemble methods (EAR and EARnest) are effective for improving the retrieval performance in terms of all the metrics. Our best model EARnest achieves the highest MAP performance, outperforming all the base models and ensemble baselines.

4.2. Results

Table 2 reports the evidence retrieval performance of all models discussed. All three base models that target either semantic equivalence or inference do not yield optimal performance. As expected, the MSMARCO CE achieves the highest performance among the base models, as it is a strong baseline that is commonly used for retrieval tasks. However, it only considers the semantic matching between question and individual candidate sentences, ignoring the other important relevance matching characteristics such as exact matching signals, textual entailment, and relatedness between candidate evidence sentences.

For the baseline ensemble models, AR performs worse than the MSMARCO CE, while being slightly better than BM25 and the QNLI CE. Its retrieval performance is essentially a compromise among the performances of the three base models, because it directly averages the individual ranking results. In contrast, SimCom[8] does take advantage of complementary relevance signals from the base models, so to perform better than any of the individual base model. However, it fails to deliver the best overall performance because it simply combines the final output scores from the base models without exploiting the interactions between the relevance signals behind.

Lastly, our approaches (i.e., EAR and EARnest) not only outperform the base models, but also exceed the order-based and score-based ensemble models on all metrics. They both jointly consider diverse relevance signals simultaneously, and therefore achieve greater improvements on the retrieval performances. EARnest further considers the relatedness between evidence sentences, becoming our best model. It achieves the highest MAP, and higher than the MSMARCO CE by 10%.

5. Conclusion

In this work, we showed that successful relevance matching for evidence retrieval in multi-hop QA requires considering diverse signals including exact matching, semantic textual similarity, and textual entailment between question and candidate sentences, and relatedness between candidate evidence sentences. We applied off-the-shelf statistical models and transformers to capture different dimensions of relevance and effectively combined them to jointly retrieve candidate evidences that cover diverse and most relevant information for the question when concatenated. Experimental results on HotpotQA reveal that our models are effective for improving the retrieval performance for multi-hop questions, comparing to all the single retrieval models they based on, also the order-based and score-based ensemble baseline models.

[7] On average, comparison-type questions are easier to answer because the necessary information (i.e., the two entities to be compared) tends to be present in the question.

[8] The result of the SimCom model uses $\alpha = 3$ and $\beta = 1$, which achieves the highest performance according to the grid search results on 10% of the full dataset.

References

Akari Asai, Kazuma Hashimoto, Hannaneh Hajishirzi, Richard Socher, and Caiming Xiong. 2020. Learning to retrieve reasoning paths over wikipedia graph for question answering.

Payal Bajaj, Daniel Campos, Nick Craswell, Li Deng, Jianfeng Gao, Xiaodong Liu, Rangan Majumder, Andrew McNamara, Bhaskar Mitra, Tri Nguyen, et al. 2016. Ms marco: A human generated machine reading comprehension dataset. *arXiv preprint arXiv:1611.09268*.

Danqi Chen, Adam Fisch, Jason Weston, and Antoine Bordes. 2017. Reading wikipedia to answer open-domain questions.

Yair Feldman and Ran El-Yaniv. 2019. Multi-hop paragraph retrieval for open-domain question answering.

Ruiliu Fu, Han Wang, Xuejun Zhang, Jun Zhou, and Yonghong Yan. 2021. Decomposing complex questions makes multi-hop qa easier and more interpretable. *arXiv preprint arXiv:2110.13472*.

Luyu Gao, Zhuyun Dai, Zhen Fan, and Jamie Callan. 2020. Complementing lexical retrieval with semantic residual embedding. corr abs/2004.13969 (2020). *arXiv preprint arXiv:2004.13969*.

Jiafeng Guo, Yixing Fan, Qingyao Ai, and W Bruce Croft. 2016. A deep relevance matching model for ad-hoc retrieval. In *Proceedings of the 25th ACM international on conference on information and knowledge management*, pages 55–64.

Matthew Honnibal, Ines Montani, Sofie Van Landeghem, and Adriane Boyd. 2020. spaCy: Industrial-strength Natural Language Processing in Python.

Po-Sen Huang, Xiaodong He, Jianfeng Gao, Li Deng, Alex Acero, and Larry Heck. 2013. Learning deep structured semantic models for web search using clickthrough data. In *Proceedings of the 22nd ACM international conference on Information & Knowledge Management*, pages 2333–2338.

Yichen Jiang and Mohit Bansal. 2019. Self-assembling modular networks for interpretable multi-hop reasoning. *arXiv preprint arXiv:1909.05803*.

Vladimir Karpukhin, Barlas Oğuz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense passage retrieval for open-domain question answering. *arXiv preprint arXiv:2004.04906*.

Jimmy Lin, Xueguang Ma, Sheng-Chieh Lin, Jheng-Hong Yang, Ronak Pradeep, and Rodrigo Nogueira. 2021a. Pyserini: An easy-to-use python toolkit to support replicable ir research with sparse and dense representations. *arXiv preprint arXiv:2102.10073*.

Jimmy Lin, Rodrigo Nogueira, and Andrew Yates. 2021b. Pretrained transformers for text ranking: Bert and beyond. *Synthesis Lectures on Human Language Technologies*, 14(4):1–325.

Zhengdong Lu and Hang Li. 2013. A deep architecture for matching short texts. *Advances in neural information processing systems*, 26.

Sewon Min, Victor Zhong, Luke Zettlemoyer, and Hannaneh Hajishirzi. 2019. Multi-hop reading comprehension through question decomposition and rescoring.

Bhaskar Mitra, Fernando Diaz, and Nick Craswell. 2017. Learning to match using local and distributed representations of text for web search. In *Proceedings of the 26th International Conference on World Wide Web*, pages 1291–1299.

Liang Pang, Yanyan Lan, Jiafeng Guo, Jun Xu, and Xueqi Cheng. 2016. A study of matchpyramid models on ad-hoc retrieval. *arXiv preprint arXiv:1606.04648*.

Ethan Perez, Patrick Lewis, Wen-tau Yih, Kyunghyun Cho, and Douwe Kiela. 2020. Unsupervised question decomposition for question answering. *arXiv preprint arXiv:2002.09758*.

Peng Qi, Xiaowen Lin, Leo Mehr, Zijian Wang, and Christopher D Manning. 2019. Answering complex open-domain questions through iterative query generation. *arXiv preprint arXiv:1910.07000*.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084*.

Xuan Shan, Chuanjie Liu, Yiqian Xia, Qi Chen, Yusi Zhang, Angen Luo, and Yuxiang Luo. 2020. Bison: Bm25-weighted selfattention framework for multi-fields document search. *arXiv preprint arXiv:2007.05186*.

Yelong Shen, Xiaodong He, Jianfeng Gao, Li Deng, and Grégoire Mesnil. 2014. Learning semantic representations using convolutional neural networks for web search. In *Proceedings of the 23rd international conference on world wide web*, pages 373–374.

Alon Talmor and Jonathan Berant. 2018. The web as a knowledge-base for answering complex questions. *arXiv preprint arXiv:1803.06643*.

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

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R Bowman. 2018. Glue: A multi-task benchmark and analysis platform for natural language understanding. *arXiv preprint arXiv:1804.07461*.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William W. Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. 2018. Hotpotqa: A dataset for diverse, explainable multi-hop question answering.

A. QER of Similarity Combination

$$QER_{q,s_j} = \begin{cases} \frac{\eta(\mathcal{BM}25_{q,s_j}) + \alpha \cdot \eta(STS_{q,s_j}) + \beta \cdot \eta(IS_{q,s_j})}{3} & \text{if } \mathcal{BM}25_{q,s_j} > 0 \\ \frac{\alpha \cdot \eta(STS_{q,s_j}) + \beta \cdot \eta(IS_{q,s_j})}{2} & \text{Otherwise} \end{cases}$$

where semantic textual similarity (STS) and inference similarity (IS) are scores from MSMARCO CE and QNLI CE. It first normalizes the scores with η[9], and then combines the normalized scores using the weights α and β.

B. Impact of K

EAR and EARnest both jointly consider pairs of candidate sentences top ranked by the base models. The cut-off parameter K is used to partition sentences considered as top-ranked by individual base model or not. The larger K is, the more exhaustive combination of candidate sentence pairs would be considered. However, the number of pairs is quadratic in the number of K, so it becomes much more computational costly when K is too large. Thus, we test on 600 randomly sampled questions (about 10% of full dataset) to compare the impact when changing the value of K. The resulting retrieval performance is exact same when changing K from 3 to 5, while the number of pairs compared increases from 12 to 33.5 on average. This is expected, because we only consider the top pair to scored against the question, and sentences in the pairs are often more likely to be ranked closer to the top of lists by base models respectfully since they contain stronger relevance signals.

C. Necessity of Inference model

To further demonstrate the benefits brought by the inference model, we conduct an ablation experiment by replacing the QNLI CE in EARnest with MSMARCO CE while keep everything else the same. We also compare the difference on retrieval performance with the randomly sampled 600 questions. The result is shown in table C. Without the QNLI CE capturing the entailment relation to promote evidences that are can be inferred by the questions to the top, BM25 and MSMARCO CE might miss them according to lexical and semantic matches. Therefore, the result is significantly lower than the full EARnest model, which confirms that textual entailment is a very important relevance signal to the multi-hop QA evidence retrieval task and should be considered along with the semantic equivalence.

Models	P@3	P@5	MAP	R@3	R@5	R@10
BM25	0.44	0.32	0.6	0.55	0.66	0.79
MSmarco	0.47	0.34	0.65	0.59	0.71	0.82
QNLI	0.34	0.24	0.45	0.43	0.52	0.64
EARnest	**0.56**	**0.38**	**0.75**	**0.71**	**0.8**	**0.88**
EARnest - QNLI	0.52	0.37	0.7	0.66	0.77	0.86

Table 3: With the EARnest ensemble model framework, we replace QNLI CE with Ms Marco CE, and the retrieval performance significantly decreased. Comparing to the full EARnest model, MAP drops 5% without exploting the QNLI CE model to capture the textual entailment relevance signal.

[8] https://pypi.org/project/rank-bm25

[9] η performs normalization to scale inputs to unit norms with Scikit-learn's normalizer:https://scikit-learn.org/stable/modules/generated/sklearn.preprocessing.Normalizer.html