Distilled Evidence Trees for Efficient QA
WIP - Not for distribution

Jeremiah Milbauer & Emma Strubell
Carnegie Mellon University
{jmilbaue|strubell}@cmu.edu

Abstract

The most efficient systems for question answering (QA) use pre-computed question/answer pairs to rapidly retrieve answers for user questions. But the most challenging QA tasks require powerful systems which can retrieve and synthesize multiple passages of evidence, referred to as "multi-hop reasoning". Existing approaches to multi-hop typically either traverse an existing graph of linked documents (such as `Wikipedia`), or rely on iterative query expansion using dense passage representations from an encoder such as BERT. But these approaches are either limited by the citation graph's (limitation to only coarse document-level relationships) (lack of passage-level relationships) (in the first case), or the high cost of re-building queries at each step (in the second). We propose a new strategy for multi-hop question answering in which both neural passage representations and structural links are distilled to a traversable *evidence tree*, and questions are used as keys to efficiently navigate the tree. This approach will unlock the benefits of iterative retrieval at a fraction of the runtime cost.

1 Introduction

Question answering is quickly becoming one of the most important interfaces for artificial intelligence technology. Human factors, such as a desire for rapid feedback, usability on mobile devices, and even environmental impact (Strubell et al., 2019) motivate a desire for deployable question answering systems which are both time- and memory-efficient at runtime. Recent success includes PAQ, (Lewis et al., 2021b) a system which generates millions of question/answer pairs which can be retrieved by fast nearest neighbors search. (Johnson et al., 2017; Malkov and Yashunin, 2018)

But complex questions often require the synthesis of information contained in multiple passages. For instance, the question "In what year was the worst baseball team in New York founded?"

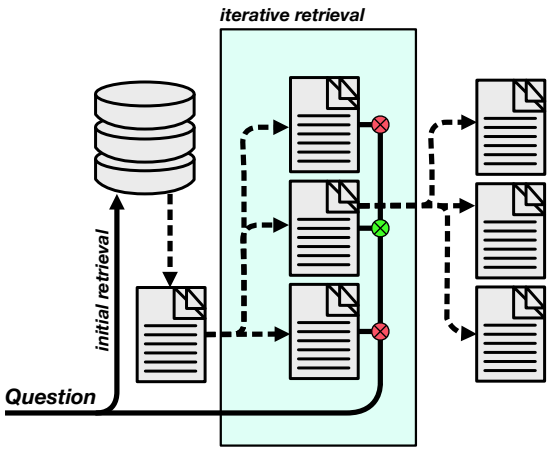

Figure 1: Iterative traversal of the evidence graph. After initial document retrieval, the question is used to guide traversal of the evidence graph, collecting passages to use for answer generation.

would require synthesis of "The Mets are the worst baseball team in New York," and "The Mets were founded in 1962." Because the information contained within one passage may provide the necessary link to another passage (we would not know to search for information on the Mets until we knew they were the worst team in the city), this requires *multi-hop reasoning*. There are a number of datasets which aim to test multi-hop reasoning (Yang et al., 2018; Welbl et al., 2018; Talmor and Berant, 2018).

Many recent QA systems for *multi-hop reasoning* decompose QA into two distinct parts: a *retriever*, which identifies passages likely to contain relevant information to the question, and a *reader*, which takes the question and retrieved passage as input to produce an answer.

A common strategy (Zhao et al., 2021b; Xiong et al., 2020) for retrieval in multi-hop scenarios is to iteratively expand a query to retrieve multiple passages. After an initial retrieval step, passages are used to expand the query and search for more

related information. But this approach is computationally expensive: it requires recomputing query representations, as well as document scores, after each iteration. Another approach (Asai et al., 2019) relies on existing chains between documents in a citation graph.

Retrieval in all of these systems is also highly dependent on document representations. Traditional work in information retrieval has frequently used sparse document representations built from a variety of text features (Salton et al., 1975; Robertson and Zaragoza, 2009) and document collection heuristics (Page et al., 1999), though more recent question answering research has demonstrated the efficacy of a pre-trained encoder, such as BERT, for generating dense document representations (Xiong et al., 2020; Zhao et al., 2021b). However this technique discards many of the structural features that have long been important in information retrieval. Promisingly, research from Cohan et al. (2020) has demonstrated that document representation fine-tuned using document relatedness signals from the links within a corpus perform better on downstream tasks.

Once passages are retrieved, systems typically pass both the query representation and the passage representations to an *answer generator* (Lewis et al., 2021a) trained to find answers within a passage. More retrieved passages may improve the quality of the generated answers, but at the price of increasing the computational costs of answer generation.

1.1 Contributions

Existing work in evidence-based retrieval has demonstrated the value of both dense passage-level representations and linked evidence chains. We propose a new method, called *distilled evidence graphs*. At a high level, *distilled evidence graphs* aim to encode high-quality passage representations into a directed evidence graph of connected passages, and then use questions as guides for an adaptive search over the graph. This will enable our system to weigh a number of additional factors, including the estimated quality of each passage representation and the computational cost to explore a new segment of the graph for evidence.

Our approach provides the following benefits over existing work:

1. **Efficiency.** By extracting link structure from either the training domain or the trained representations themselves and encoding this directly in the evidence graph, we significantly reduce run-time costs.

2. **Performance.** Dense representations often benefit from link-based filtering; link-based approaches benefit from passage-level representations. We offer both.

3. **Generalizability.** Our approach is generally applicable to any method for generating document and passage representations, and significantly simplifies computations

2 Proposed Method

In this section, we describe the two primary components of our proposed method: evidence graphs, and a trainable adaptive retriever.

2.1 Evidence Graphs

Existing methods for link-based retrieval rely on document abstracts to act as passages (Asai et al., 2019). But a significant amount of information is also contained within the passages of a document body. Dense search with query expansion over the entire passage space has been proposed to solve this (Zhao et al., 2021b,a; Xiong et al., 2020), but these approaches sacrifice the valuable citation signal between documents. Additionally, they are orders of magnitude more expensive than traversing links because each step of iteration requires both recomputing the query representation and finding nearest neighbors in the entire corpus.

We aim to use the additional (but so far understudied) supervision provided by *citation contexts* to solve this issue. Each citation in a document falls within some sentence of the document; by treating that sentence as a query itself, we can identify which passages within the target document are most relevant to the citation. This allows us to build a passage-level weighted graph which will act as a foundation for our evidence graphs. (This approach is highly dependant on the quality of our passage representations and their ability to predict citation structure in the corpus. To date, we have made significant progress on improving the quality of passage representations; this work is discussed later.)

Then, by using the passage-level representations as a prior, we can proceed to use any number of state-of-the-art question answering systems, entity

linking, or document representation systems to update the link weights between passages. This allows our evidence graph framework to distil knowledge from other work and grow as the field does – while still remaining efficient and powerful. We see evidence trees as a general framework which can be used to improve much of the existing work in question answering.

2.2 Adaptive Retrieval

The core idea of our adaptive retrieval framework is to build a system for traversing the evidence graph that responds to a number of considerations:

1. **Representation quality.** Whether the representation of the candidate passage is reliable. Exploring low-quality representations may not be worth the effort.

2. **Relevance.** Whether the candidate passage is relevant to the question.

3. **Exploration.** Some questions may require shallower reasoning while others require deep evidence chains.

4. **Computational cost.** Can we compress the size of the representations necessary to traverse the tree?

Figure 1 illustrates how a question is used for both initial retrieval, and evidence graph traversal once the evidence graph is assembled. Still, retrieval is additionally parameterized by beam search settings and by both question and document representations.

To date, we have explored methods for estimating the quality of passage representations in a state-of-the-art document representation system, SPECTER (Cohan et al., 2020). These experiments have enabled us to build a competence-based curriculum (Platanios et al., 2019) which outperforms state of the art on a variety of document representation metrics.

The next phase of our work will involve the parameterized exploration of adaptive retrieval. Currently, we are considering two approaches: a beam constructor approach, and a "passages as gateways" approach (we do not yet have a catchy name for this).

Beam Constructor We will train a model to automatically generate beam search parameters

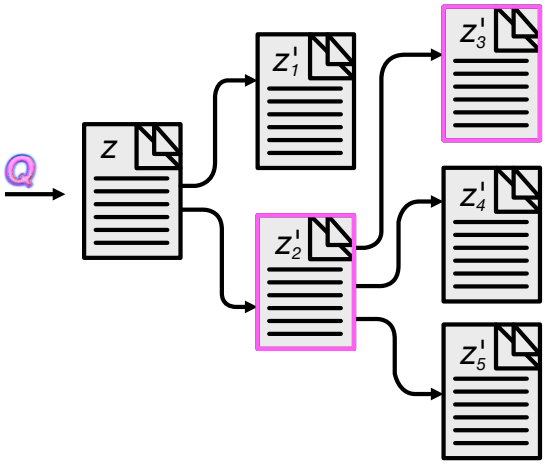

Figure 2: While an initial passage may not have the needed information, passages within its subtree - z'_2 and z'_3 – do have the required information. This should increase the weight given to passage z.

(namely, width and depth) appropriate to each question. Such a generator can be trained even with minimal domain-specific question/answer training data. Given just a question/answer pair, we can first identify passages most similar to the question as "sources" on the graph. Then, we can use an existing state-of-the-art QA system to identify passages which likely to contain the answer, as "sinks." We then train the *beam generator* (with the question as input) to produce parameters which will, from each "source," have a high-probability of reaching a "sink." From an efficiency perspective, our goal is to always select the minimal suitable beam parameters. Succeeding in this approach should also enable our system to handle single-hop questions efficiently.

Passages as Gateways When comparing a question to a passage in an iterative query-expansion retrieval system, we want both to understand whether the passage itself contains relevant information, as well as whether the passages *beyond* that passage in the graph contain relevant information. We propose a fine-tuning strategy which jointly optimizes question and document representations toward this goal:

$$\text{Loss}(q, z) = |P(z|q) - P(a|q, \text{TREE}(z))|$$
$$P(z|q) \sim \text{sim}(f_q(q), f_z(z))$$
$$P(a|q, \text{TREE}(z)) \sim \text{sim}(a, g(q, z, z'_1, ..., z'_n))$$

for a query q, a passage z, passages $z'_1, ..., z'_n$ below z in the graph to a depth of n, a query encoder f_q,

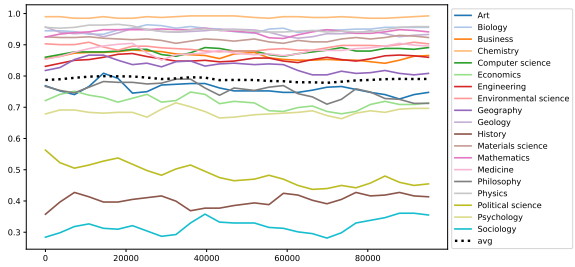

Figure 3: Representation quality (measured by classification F1 score on the validation set) for SPECTER-trained representations over training time.

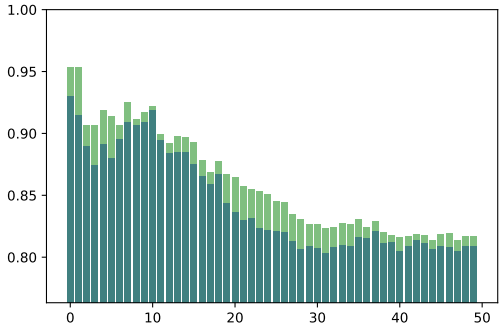

Figure 4: F1-score for each of 50 cumulative difficulty tranches. SPECTER results in dark green; our results in light green.

a passage encoder f_z, and an answer generator g. TREE(z) is a subtree rooted by z in the evidence graph.

Through this method, we train passage representations to not only represent whether they contain useful information, but also whether they are a suitable "next step" in the search. Figure 2 provides a visual representation of why subtrees matter. From an efficiency perspective, encoding this information into the passage representation itself requires significantly fewer computations than actually traversing the passage's subtree.

3 Project Updates

In this section, we describe our progress on a few key elements of this work.

3.1 Estimating document difficulty

We explored an existing state-of-the-art document representation system, SPECTER (Cohan et al., 2020), which was designed for scientific document representation. We found that there was a significant difference in the quality of representations related to two factors within the dataset: documents from humanistic and social-science disciplines, such as philosophy, political science, and sociology, are much harder for the system to represent (Figure 4). Initially we believed this might have been a consequence of a class imbalance within the training data, but when re-training SPECTER with class-balanced training data we found the results held. Further investigation revealed that the low performance was correlated with a high incidence of cross-disciplinary citation within the data. Documents which cite only within their discipline had better representations; documents which cited across disciplines had worse representations. Additionally, performance was highly correlated with the technicality of the language within the document.

These results suggest that factors, such as a documents position within the citation graph (either being in a densely connected region or a sparse region), may be a useful heuristic for the adaptive retriever – representations from sparse regions should be treated as less reliable, unless improved document representation methods are found.

3.2 Higher-quality document representations

Using what we discovered about estimating difficulty, we were able to build a *competence-based curriculum* fine-tuning approach which beats the state-of-the-art for scientific document representation on a variety of metrics (Figure **??**). Our system uses two different curricula: one based on the entropy of the citation distribution, the other on the rarity of words within a document abstract. Training using the entropy-based curriculum achieves better than state-of-the-art performance on classification tasks, in fewer epochs and with fewer training examples than the state-of-the-art fine-tuning approach. The cost to build the curriculum is minimal; taking less than 15 minutes on a CPU to build a curriculum of 500,000 training instances across 200,000 query papers.

3.3 A dataset for citation contexts

Using the S2ORC (Lo et al., 2020) dataset, we were able to extract citations spans and the documents to which they point. An additional source of data, Microsoft Academic Graph (MAG) (Sinha et al., 2015) has been considered. Microsoft's support for MAG will be discontinued 12/31/2021, so we intend to download it in its entirety before then.

However, access to MAG may be reduced in the long term and it may not be suitable for conducting widely reproducible research.

3.4 Citation contexts as queries

We performed initial experiments to explore the difference between abstract-based document representations and reference-based document representations. Our initial findings, using a DOC2VEC (Le and Mikolov, 2014) approach, demonstrate that citation contexts are a powerful indicator of document content. Further experiments will show whether citation contexts are appropriate to use as queries to identify relevant passages of a target document.

References

Akari Asai, Kazuma Hashimoto, Hannaneh Hajishirzi, Richard Socher, and Caiming Xiong. 2019. Learning to retrieve reasoning paths over wikipedia graph for question answering. In *International Conference on Learning Representations*.

Arman Cohan, Sergey Feldman, Iz Beltagy, Doug Downey, and Daniel S Weld. 2020. Specter: Document-level representation learning using citation-informed transformers. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2270–2282.

Jeff Johnson, Matthijs Douze, and Hervé Jégou. 2017. Billion-scale similarity search with gpus.

Quoc Le and Tomas Mikolov. 2014. Distributed representations of sentences and documents. In *International conference on machine learning*, pages 1188–1196. PMLR.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. 2021a. Retrieval-augmented generation for knowledge-intensive nlp tasks.

Patrick Lewis, Yuxiang Wu, Linqing Liu, Pasquale Minervini, Heinrich Küttler, Aleksandra Piktus, Pontus Stenetorp, and Sebastian Riedel. 2021b. Paq: 65 million probably-asked questions and what you can do with them.

Kyle Lo, Lucy Lu Wang, Mark Neumann, Rodney Kinney, and Daniel S Weld. 2020. S2orc: The semantic scholar open research corpus. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4969–4983.

Yu. A. Malkov and D. A. Yashunin. 2018. Efficient and robust approximate nearest neighbor search using hierarchical navigable small world graphs.

Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. 1999. The pagerank citation ranking: Bringing order to the web. Technical report, Stanford InfoLab.

Emmanouil Antonios Platanios, Otilia Stretcu, Graham Neubig, Barnabas Poczos, and Tom M Mitchell. 2019. Competence-based curriculum learning for neural machine translation. In *Proceedings of NAACL-HLT*, pages 1162–1172.

Stephen Robertson and Hugo Zaragoza. 2009. *The probabilistic relevance framework: BM25 and beyond*. Now Publishers Inc.

Gerard Salton, A. Wong, and Chung-Shu Yang. 1975. A vector space model for automatic indexing. *Commun. ACM*, 18:613–620.

Arnab Sinha, Zhihong Shen, Yang Song, Hao Ma, Darrin Eide, Bo-June Hsu, and Kuansan Wang. 2015. An overview of microsoft academic service (mas) and applications. In *Proceedings of the 24th international conference on world wide web*, pages 243–246.

Emma Strubell, Ananya Ganesh, and Andrew McCallum. 2019. Energy and policy considerations for deep learning in nlp. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3645–3650.

Alon Talmor and Jonathan Berant. 2018. The web as a knowledge-base for answering complex questions. 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 641–651.

Johannes Welbl, Pontus Stenetorp, and Sebastian Riedel. 2018. Constructing datasets for multi-hop reading comprehension across documents. *Transactions of the Association for Computational Linguistics*, 6:287–302.

Wenhan Xiong, Xiang Li, Srini Iyer, Jingfei Du, Patrick Lewis, William Yang Wang, Yashar Mehdad, Scott Yih, Sebastian Riedel, Douwe Kiela, et al. 2020. Answering complex open-domain questions with multi-hop dense retrieval. In *International Conference on Learning Representations*.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William Cohen, Ruslan Salakhutdinov, and Christopher D Manning. 2018. Hotpotqa: A dataset for diverse, explainable multi-hop question answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2369–2380.

Chen Zhao, Chenyan Xiong, Jordan Boyd-Graber, and Hal Daumé III. 2021a. Distantly-supervised dense retrieval enables open-domain question answering without evidence annotation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 9612–9622.

Chen Zhao, Chenyan Xiong, Jordan Boyd-Graber, and Hal Daumé III. 2021b. Multi-step reasoning over unstructured text with beam dense retrieval. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4635–4641.