Causality Detection for Causality-driven Adhoc Information Retrieval Task

Chuan-An Lin[a], Yi Zhang[a]

[a]IRKM Lab, Engineering Building 2, University of California, Santa Cruz, Santa Cruz, CA, 95064

Abstract

This is the working note of our participation in the 2020 Causality-driven Adhoc Information Retrieval Task. We aim to retrieve news documents that indicate possible causes of a query event. Our proposed method involves query expansion based on causality relation detection from a primarily retrieved collection. We compare our method with two baselines and analyze the performance of expansion terms on selected topics. Our analysis shows that although the naive causality detection method recognizes some critical terms, precision should be improved to outperform the baselines.

Keywords

query expansion, causality detection

1. Introduction

Automatic connecting events based on causality relation provides a promising application for information retrieval. In our participation in Causality-driven Adhoc Information Retrieval (CAIR) 2020 task, we aim to retrieve news documents that provide possible causes of a given event. A closely related area for this task is causality detection, on which many researches have been focusing, and different approaches have been discussed [1, 2]. Applications on news article have also been studied, focusing on predicting possible future event [3, 4]. In contrast, the application in CAIR is a quite new direction where we want to infer possible cause from a current event [5]. In this work, we proposed a naive method to perform query expansion based on causality detection to approach this task; We describe our method in section 2 and discuss the experiments in section 3; We then provide further analysis of our approach in section 4; Section 5 concludes our work.

2. Method

2.1. Pre-event Query Expansion

The core idea of our method is query expansion based on causality detection. We define a **post-event** as the event we want to know about its cause (e.g., Telecom minister A. Raja resignation), and a **pre-event** as one possible event that leads to the post-event. Thus, we treat the task as detecting possible pre-events given the post-event, which is the query, and retrieving documents

Forum for Information Retrieval Evaluation (FIRE)-2020, 16th-20th December, Hyderabad, India

clin134@ucsc.edu (C. Lin); yiz@soe.ucsc.edu (Y. Zhang)

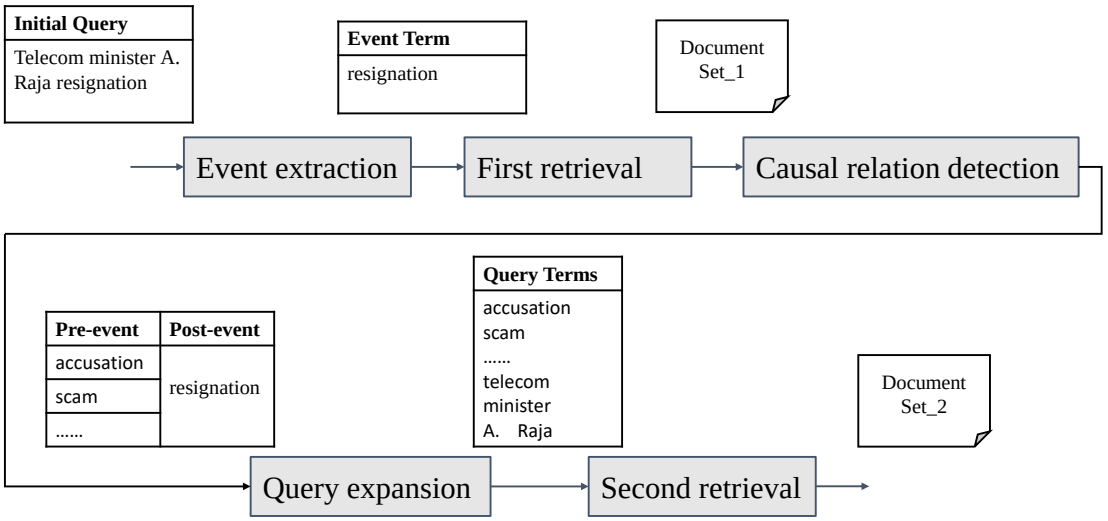

Figure 1: Diagram of our query expansion approach based on causal relation detection.

that describe the pre-events. To achieve this, we assume that potential pre-events can be found in the documents that describe the post-event. Once retrieve these kinds of documents, we can conduct causal relation detection to find potential pre-events and use them to do a second retrieval to get the final desired documents. We organize our approach's procedure to 5 steps, as shown in figure 2.1, and describe them in detail in the following paragraphs.

Event extraction: The first step is event extraction, in which we extract the syntactic head word of the sentence (usually the verb) in the query title to represent the post-event. The intention is to get more information from similar events.

First retrieval: We use the resulting event term above to retrieve a document collection for extracting potential pre-events.

Causal relation detection: We implement a simplified version of [6]. Specifically, we use causal keywords ('because', 'lead to', 'after') to first recognize "causal sentence" with a causal relation from the retrieved document collection, then we extract verbs and nouns in the causal sentence after or before the occurring of the causal keyword depending on which keyword it is. To estimate the relevance score of the verbs and nouns as pre-events for query expansion, we simply calculate the frequency of their occurrence in among all the causal sentences.

Query expansion: We use the extracted pre-event terms above to expand the original query and remove the original event term. We presumably choose the top 5 sorted according to its frequency mentioned above. Further investigation of the number of terms to use can be explored in the future.

Second retrieval: The document set retrieved using the expanded query is our final result.

2.2. Implementation

Following the work of [5], we use Apache Lucene 5.5.5 version to build our retrieval system. We also use the EnglishAnalyzerWithSmartStopWord class to preprocess the documents as well

Table 1
An example from CAIR training topics.

query title	Telecom minister A. Raja resignation
query narratives	Documents retrieved should contain information about the accusations against A. Raja in the 2G Spectrum scam. Documents that talk about his criticisms for allocating 4.4MHz of spectrum a scarce resource at the 2001 price of Rs 1,651 crore through a first come, first served policy rather than auctions; by compromising on the market value of spectrum, the government is estimated to have lost revenues worth around Rs 50,000 crore; new licensees are signing equity deals with foreign players in which the value of spectrum has risen sevenfold, would be considered as relevant.

as the queries. For similarity function, we use LMJelineMercer similarity with the smoothing parameter set to 0.7. 5 top pre-event terms expanded by our method are used and removed if the expanded term is identical to the original event term. We retrieve up to $1,000$ documents for each topic as our final results.

3. Experiments

3.1. Dataset

The CAIR dataset is a collection of 303,291 Telegraph India news documents from 2001 to 2010 [5]. There are 20 query topics for testing in total. Each topic contains two parts: query title and query narratives. The former is a selected public event that we treat as the post-event, and the latter is a description of the criteria of relevant documents. Table 1 provides an example topic.

3.2. Baselines

We compare our method with two baselines that directly use the query title and query narratives, respectively, with the same setting. We use **title** to represent the baseline using query title and **narratives** to represent the one using query narratives. We use **title-expansion** to denote our method.

3.3. Results

Table 2 shows the experimental results. We can find that our method has not been competitive with the two baselines. **narratives** outperforms the other methods significantly, which is not surprising as the content of query narratives have already contained the pre-events desired. In Contrast, the results show that the expanded terms in **title-expansion** do not help improve the original query title in the retrieval task. We then provide further analysis in section 4.

Table 2
Experiment Results

Method	MAP	P@5
title	0.4066	0.5400
narratives	**0.4553**	**0.7000**
title-expansion	0.3885	0.5000

4. Analysis

We choose 3 topics from the testing set to conduct the analysis. The top terms as potential pre-events extracted by our model are listed in table 3. For the event term 'assassination,' our model effectively extracted terms like 'attack' and 'deathkilled,' which is of quite a small portion. Most extracted terms are rather irrelevant. For the event term 'accused,' more terms for possible pre-events are detected ('court', 'police', 'case', 'arrest', 'murder', etc.) We notice that the accuracy of our extraction method is affected by the topic property. For the event 'assassination,' proper nouns are identified ('Gandhis,' 'Musharraf') as well as other terms like 'President,' 'Minister,' and 'government' that might be too specific for some particular events that involve assassination. The extracted terms for the post-event 'resignation' follow a similar pattern. In contrast, the extracted terms for the 'accused' event are rather general and can be applied to different events that involve accusations. We infer that the difference comes from the frequency of some particular events being reported on the news. For example, If there is a lot of news that reports the assassination of a particular person, the terms specifically related to this single event are more likely to be extracted by our model. Overall, the analysis shows that the extracted terms of our current method contain some useful information yet not accurate enough and highly depends on the topic.

5. Conclusion

In this work, we implement a retrieval model based on query expansion and naive causality detection to approach the CAIR task. We also compare our model with baselines that only use query title and query narratives. Experimental results show that our method has not been able to surpass the baselines. Further analysis of selected topics indicates that the causality detection method needs to be improved to enhance the extracted terms' precision. While our naive approach has not distinguished the causal sentences that are actually relevant to the topic from the non-relevant ones, possible approaches like embedding method to jointly model the topic and the causal sentences may be considered in future works to improve the performance.

Acknowledgments

This work is supported by the IRKM lab in University of California, Santa Cruz.

Table 3
Analysis of potential pre-event terms extracted by our model

Query Title	Event Term	Top 20 Retrieved Pre-event Terms
Assassination of Osama-bin-laden	assassination	assassination, said, Prime government, would, years, attack Gandhis, deathkilled, President Minister, Gandhi, leader, attempt security, Indira, minister, Musharraf people
Accused Ajmal Kasab	accused	accused, court, police, said, case trial, bail, evidence, years, Court murder, could, would, incident Delhi, CBI, arrest, death, failed told
Maharashtra CM ashok chavan resignation	resignation	resignation, said, minister, party meeting, government, would, Prime BJP, yearstoday, Minister, state chief, World, Cup, Pakistan President, quit, members

References

[1] N. Asghar, Automatic extraction of causal relations from natural language texts: A comprehensive survey, CoRR abs/1605.07895 (2016). URL: http://arxiv.org/abs/1605.07895. arXiv:1605.07895.

[2] R. Pradeep Kumar, P. Aswathi, Extraction of causality and related events using text analysis, in: 2019 2nd International Conference on Intelligent Computing, Instrumentation and Control Technologies (ICICICT), volume 1, 2019, pp. 1448–1453. doi:10.1109/ICICICT46008.2019.8993302.

[3] E. Kıcıman, Causal inference over longitudinal data to support expectation exploration, in: The 41st International ACM SIGIR Conference on Research Development in Information Retrieval, SIGIR '18, Association for Computing Machinery, New York, NY, USA, 2018, p. 1345. URL: https://doi.org/10.1145/3209978.3210215. doi:10.1145/3209978.3210215.

[4] K. Radinsky, S. Davidovich, S. Markovitch, Learning causality for news events prediction, in: Proceedings of the 21st International Conference on World Wide Web, WWW '12, Association for Computing Machinery, New York, NY, USA, 2012, p. 909–918. URL: https://doi.org/10.1145/2187836.2187958. doi:10.1145/2187836.2187958.

[5] S. Datta, D. Ganguly, D. Roy, F. Bonin, C. Jochim, M. Mitra, Retrieving potential causes from a query event, in: Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR '20, Association for Computing Machinery, New York, NY, USA, 2020, p. 1689–1692. URL: https://doi.org/10.1145/3397271.3401207. doi:10.1145/3397271.3401207.

[6] S. Zhao, Q. Wang, S. Massung, B. Qin, T. Liu, B. Wang, C. Zhai, Constructing and embedding

abstract event causality networks from text snippets, in: Proceedings of the Tenth ACM International Conference on Web Search and Data Mining, WSDM '17, Association for Computing Machinery, New York, NY, USA, 2017, p. 335–344. URL: https://doi.org/10.1145/3018661.3018707. doi:10.1145/3018661.3018707.