Expert Systems With Applications 42 (2015) 9086–9104

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

Knowledge-based question answering using the semantic embedding space

Min-Chul Yang[a], Do-Gil Lee[b], So-Young Park[c], Hae-Chang Rim[a,*]

[a] *Department of Computer & Radio Communications Engineering, Korea University, Seoul, Republic of Korea*
[b] *Research Institute of Korean Studies, Korea University, Seoul, Republic of Korea*
[c] *Department of Game Design and Development, Sangmyung University, Seoul, Republic of Korea*

ARTICLE INFO

Keywords:
Question answering
Knowledge base
Embedding model
Neural networks
Labeled-LDA
Distributional semantics

ABSTRACT

Semantic transformation of a natural language question into its corresponding logical form is crucial for knowledge-based question answering systems. Most previous methods have tried to achieve this goal by using syntax-based grammar formalisms and rule-based logical inference. However, these approaches are usually limited in terms of the coverage of the lexical trigger, which performs a mapping task from words to the logical properties of the knowledge base, and thus it is easy to ignore implicit and broken relations between properties by not interpreting the full knowledge base. In this study, our goal is to answer questions in any domains by using the semantic embedding space in which the embeddings encode the semantics of words and logical properties. In the latent space, the semantic associations between existing features can be exploited based on their embeddings without using a manually produced lexicon and rules. This embedding-based inference approach for question answering allows the mapping of factoid questions posed in a natural language onto logical representations of the correct answers guided by the knowledge base. In terms of the overall question answering performance, our experimental results and examples demonstrate that the proposed method outperforms previous knowledge-based question answering baseline methods with a publicly released question answering evaluation dataset: WebQuestions.

1. Introduction

Question answering (QA) is concerned with building systems that automatically answer questions posed by humans in a natural language by exploiting the techniques of natural language processing (NLP) and information retrieval (IR). In general, QA systems can retrieve and extract answers from natural language documents in unstructured Web-data by using a structured query that is semantically associated with a given question.

As an alternative form of QA implementation, knowledge-based question answering (KB-QA) requires a structured database called a knowledge base (KB) because KB-QA systems simply extract answers from the structured knowledge base instead of the unstructured Web-data. In other words, these systems can use information to resolve a query without having to navigate to other sites and assemble the information themselves. A KB is a technique used to store large volumes of factual information in a structured format, which is constructed based on well-written textual data, such as Wikipedia[1] and human contributions. In particular, Freebase,[2] one of the publicly available databases, allows users to add new information and to modify incorrect information even if the users are not experts. Freebase comprises a huge volume of facts as multi-relational data in a triple format: $<$ *subject entity, logical predicate(= relation), object entity* $>$. For KB-QA, the central problem is how to transform the input question into its corresponding structured query for KB as the logical form. This task involves the transformation of various natural language representations in an unstructured format into semantically similar KB-properties in the structured and canonical format.

The latest KB-QA systems [2–4,9,13,21,24,37] employ the semantic parsing technique to exploit the mappings between the lexical phrases and logical predicates in the KB. Semantic parsing is a learning task that maps natural language statements onto formal meaning representations of their underlying meanings. In KB-QA, this technique is used to transform given natural language questions into structured queries for KB. However, previously proposed methods have the following three limitations. (1) The meaning of a logical

* Corresponding author. Tel.: +82 2 3290 3195; fax: +82 2 929 7914.
E-mail addresses: mcyang@nlp.korea.ac.kr (M.-C. Yang), motdg@korea.ac.kr (D.-G. Lee), ssoya@smu.ac.kr (S.-Y. Park), rim@nlp.korea.ac.kr (H.-C. Rim).

[1] http://en.wikipedia.org.
[2] https://www.freebase.com.

http://dx.doi.org/10.1016/j.eswa.2015.07.009
0957-4174/© 2015 Elsevier Ltd. All rights reserved.

predicate often shares different natural language expression (NLE) forms, so the lexical representations linked with a predicate may be limited in size with respect to the user inputs. (2) Entities detected by the named entity recognition (NER) component are used to combine the logical forms with the logical predicates, and thus their types should also be consistent with the predicates. However, most of the NER components used in existing KB-QA systems are independent of the NLE-to-predicate mapping procedure. (3) Semantic parsing-based approaches have difficulty fully representing various properties of the KB because the appropriate knowledge information may be found for given lexical statements by exploring the lexicon rather than by providing comprehensive descriptions.

In general, embedding models in the NLP area, such as Word2Vec [22], are used to represent the meanings of words as low-dimensional vectors. These distributed representations of words can also be used to memorize many linguistic regularities and patterns. The basic idea of this approach is that the vector of a word has a similar weight to the vectors of its surrounding words because co-occurring words within a limited range are likely to share similar semantics and contexts. In the same manner, we propose a semantic embedding space that jointly encodes words and KB-properties based on their semantic relationships. Joint embeddings [34] have been used to learn various representations of items with different types, but we focus on building semantic mappings for KB-QA over the embedding space. The semantic embedding space has three roles, as follows. (1) Semantic embedding jointly encodes words and KB-properties into the same space based on their semantic associations. (2) We can simply compute the semantic similarities of two given features (a word or KB-property) using the dot product operation between two embedding vectors connected by the features. (3) The semantics of words or KB-properties can be represented as distributed values.

The remainder of this paper is organized as follows. We examine previous research related to this area in Section 2 and we investigate KB-QA in Section 3. In Section 4, we introduce the setup of the proposed method and the three stages of our KB-QA system are described in Sections 5–Section 7. In Section 8, we present our experimental results and analysis. Finally, Section 9 gives our conclusions.

2. Related work

Semantic parsers can be used to transform natural language sentences into their machine interpretable corresponding logical forms in a fully formal language. Supervised semantic parsers [24,38,39] are highly reliant on < *sentence, semantic annotation* > pairs for lexical trigger extraction and model training to map natural language expressions onto formal meaning representations. Due to their requirement for data annotation, these methods are usually restricted to specific domains (such as GEO, ATIS, and JOBS) and they struggle with coverage issues caused by the limited size of lexical triggers. Thus, other studies have employed weakly supervised semantic parsers to reduce the amount of human supervision by using question–answer pairs [21] or *distant supervision* [19] instead of full semantic annotations. These approaches can automatically extract the < *sentence, semantic annotation* > pairs supported by the structured database, although some incorrect pairs may be obtained due to the redundant information in KBs.

As conventional QA approaches, when given a question statement, IR-based QA systems [14,18,30,32] try to retrieve, extract, and assemble answer information for a given question based on a large volume of unstructured data such as Wikipedia, before generating an answer statement. Similarly, community-based question answering (CQA) systems [1,15] aim to find answers from past question–answer pairs in online social sites such as Yahoo! Answers.[3]

Recently, researchers have developed open-domain systems based on large-scale KBs such as Freebase. Thus, semantic parsers for Open-QA may be learned using manually prepared schema [3,9], a comprehensive ontology [20], paraphrased questions that are semantically similar [4,12,13], a machine translation-perspective model [2], pairs of graph structures of the question statement and KBs with QA-pairs [37], a formalized knowledge representation [27] and ontology-based inferences of question's syntactic structure and context [26]. The semantic parsers employed are usually unified, formal, and scalable, where they allow a question statement to be mapped onto the appropriate logical form based on precise manually prepared lexicons or schema matching. These methods might provide correct answers, but some responses cannot be provided for complex questions because the size of the lexicon may be limited (low recall). Similarly, our method also aims to obtain similar logical forms but we only use low-dimensional embeddings of *n*-grams and the KB-properties are learned from a huge volume of texts and KBs. In previous studies of KB-QA, Cai and Yates and Berant, Chou, Frostig, and Liang produced evaluation datasets where the QA-pairs were annotated by humans based on Freebase, i.e., Free917 and WebQuestions, respectively. These two standard datasets have been utilized for QA evaluations in recent KB-QA systems, and we use WebQuestions for QA evaluations in the present study.

The pioneering studies of QA using embedding models [8] aimed to learn low-dimensional vector representations of words and KB-properties in the same space. This type of QA model obtains answers via candidate QA paths that are linked directly with each other in the KB by scoring the paths based on their similarities in the learnt embedding space. In our study, the QA paths are defined as being equal to the answer derivations. However, previous embedding models were restricted to capturing various QA paths, which could not handle various complex types of questions. The most similar KB-QA systems to our proposed method [7,36] focused on the semantic associations between words and the KB-properties of candidate answers, where the method proposed by Yang et al. was the initial version of our method. Similar to the method suggested by [8], Bordes et al. also aimed to learn embeddings of questions and answers based on question–answer pairs in ReVerb [11] and those of paraphrased questions in Yahoo Answers. However, these previous methods used highly sophisticated inferences to handle long QA paths, thereby obtaining rich representations of the training QA paths and the surrounding subgraphs of the KB, whereas our method obtains high-quality semantic links (QA paths) directly from a large-scale corpus instead of using ReVerb, which was constructed previously. In the method described in Yang et al., the aim is to construct the joint relational embeddings of words and KB-properties based on semantic links obtained directly from Wikipedia and Satori,[4] which is a KB produced by Microsoft Research. Unlike [36], we employ the following features: (1) we use question category features to identify expected answer types, (2) we filter out unassociated semantic links with distributional semantics during post-processing, and (3) feature-level representations are used to identify semantic links before training the embedding space. We consider that feature-level semantic links provide more robust semantic mappings for KB-QA than pattern-level versions.

3. Knowledge-based question answering

3.1. Challenges in KB-QA

Basically, KB-QA systems interpret the given question statement and then map it onto a logical representation of the correct answer in

[3] https://answers.yahoo.com/.

[4] http://searchengineland.com/library/bing/bing-satori.