Volume 5, Issue 2, February 2015 ISSN: 2277 128X

International Journal of Advanced Research in Computer Science and Software Engineering

Research Paper

Available online at: www.ijarcsse.com

Knowledge Base Population: A Survey

A. V. Zadgaonkar
Department of Computer Science & Engg, RCOEM,
Nagpur, Maharashtra, India

Abstract--- KBP (Knowledge Base Population) is a process of discovering facts about entities from a large corpus and used it to augment a knowledge base (KB). It takes as an input an incomplete KB and a large corpus of text and tries to complete the incomplete elements of the KB. There are two major phases of this task. Entity linking i.e. linking names from the corpus in context to entities in the KB and Slot filling i.e. adding information about an entity to the KB. This paper tried to provide an overview of the various techniques used for good KBP systems and discussed the various issues need to be considered. Also this paper attempted to measure the relevance of the KBP system with traditional Information Extraction (IE) and Question Answering (QA) systems.

***Keywords---* Entity linking, Slot filling, Knowledge Base, Co-reference Resolution, Cross Linking**

I. INTRODUCTION

Information Extraction is the process of extracting structured information from unstructured or semi-structured machine readable documents. Structured data is semantically well-defined data from a chosen target domain, interpreted with respect to category and context. It` needs to process natural language text by NLP(Natural Language Processing) techniques. Due to related inherent problems, current IE approaches focuses on narrowly restricted domains. A specific IE goal should be to use logical reasoning for drawing inferences based on the logical contents of the input data.

Traditional IE systems can extract information from individual documents in isolation quite efficiently. But real life requirement is different. One may need to collect information, scattered among multiple document collection. This demands ability to identify relevant documents and integrate facts involving redundant, complementary or conflicting entities. Such systems needs to link entities and information about these entities mentioned in a document so that it can be mapped with the entries in the data/knowledge base. On the other hand, a traditional Question Answering (QA) system does limited efforts for entity disambiguation in queries and limited use of relation/event extraction for generating answer.

The Knowledge Base Population systems (KBP) aims to combine IE and QA research communities together to discover facts about entities and augment a knowledge base with these facts. KBP involves two separate sub-tasks, **Entity Linking** and **Slot Filling**. A variety of approaches have been proposed to address both these tasks. Still there are certain conflicting issues like like what are the fundamental techniques used to achieve reasonable performance?, what is the impact of each novel method?, what types of problems are represented in the current KBP paradigm compared to traditional IE and QA? etc. In this paper overview of challenges associated with the task and the possible ways to address these challenges is discussed in detail.

Entity Linking Task

The overall goal of KBP is to automatically identify salient and novel entities, link them to corresponding Knowledge Base (KB) entries (if the linkage exists), then discover attributes about the entities and finally expand the KB with any new attributes.

In selecting among the KB entries, a system could make use of the Wikipedia text associated with each entry as well as the structured fields of each entry. In addition, there was an optional task where the system could only make use of the structured fields. This was intended to be representative of applications where no backing text was available. Each site could submit up to three runs with different parameters. The Entity Linking task align textual mention of a named-entity (a person, organization, or geo-political entity etc.) to its appropriate entry in the knowledge base or correctly determine that the entity does not have an entry in the KB. Some inherent problems associated with the task are

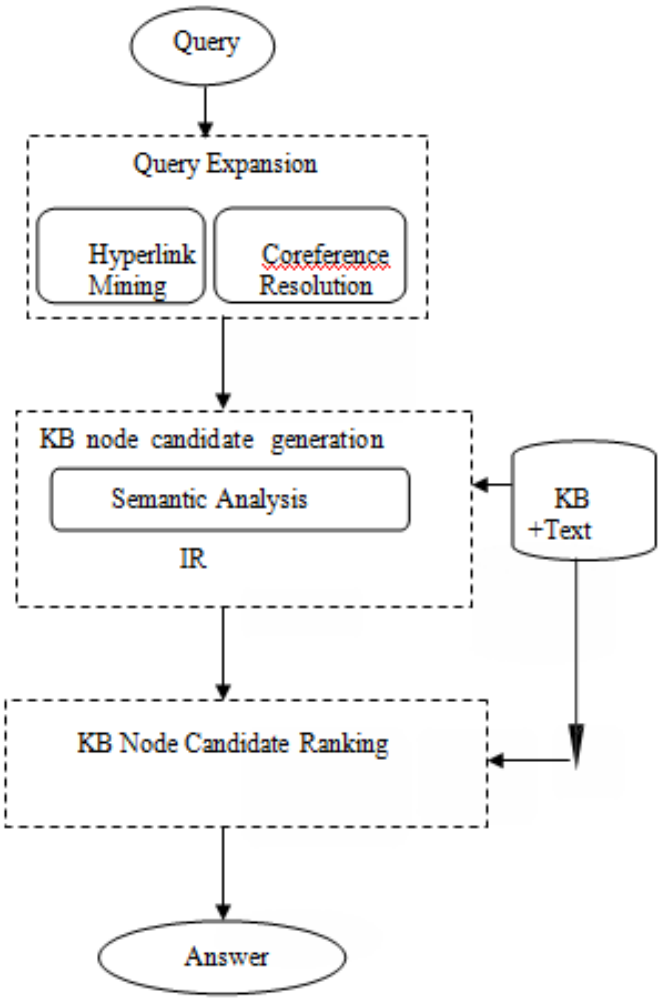
- Query entities can be referred by using multiple name variants (e.g., aliases, acronyms, misspellings)
- They may share one or more name variants with another distinct entity (e.g., Washington might refer to a person, city, state, or football team).

Generally entities are ambiguous when described in text. For example, "**George Bush**" may refer to either *George Bush Sr*. or *George Bush Jr*. The acronym ACL may refer to the *Association for Computational Linguistics* or the *ACL music festival in Austin*. Entity linking aims to take these ambiguous mentions and "link" them with concrete entities in the knowledge base. This is very much similar to co-reference resolution but differs with the following issues.

- It is spanning across multiple documents
- Linking focuses on concrete entities rather than simply clustering them.

II. ENTITY LINKING ARCHITECTURE

A typical entity linking system architecture is presented in the following diagram. It includes three steps:



Query Expansion – Expand the query into a richer set of forms using data mining or co-reference resolution in the background document.

Candidate Generation – finding all possible KB entries that a query might link to

Candidate Ranking – rank the probabilities of all candidates and generate answer

III. ISSUES TO CONSIDER

A. Cross- document Co -reference Resolution

Entity linking task can be modeled as a cross-document entity resolution problem which includes two principal challenges: the same entity can be referred to by more than one name string and the same name string can refer to more than one entity. KBP requires linking an entity mention in a source document to a knowledge base. Therefore sometimes the linking decisions heavily rely on entity profile comparison.

B. Difficult Queries Require deep understanding of context entities

Many times it is possible that query entity is not a central topic of the document with the assumption that readers have enough background knowledge about these entities. For such queries, a system would need to interpret or extract attributes for their context entities.

C. Require external hyperlink analysis

Some queries require a system to conduct detailed analysis on the hyperlinks in the source document or the Wikipedia document. Such cases might require new query reformulation and cross-document aggregation techniques, which are both beyond the scope of traditional entity disambiguation systems.

D. Require Entity Salience Ranking

Some of the queries represent salient entities and in such cases correct answers can be generated by using web popularity rank. Since the web information is used as a black box (including query expansion and query log analysis) which changes over time, it's more difficult to duplicate research results. However, gazetteers with entities ranked by salience or major entities marked are worth encoding as additional features.

E. Paucity of training data

One particular challenge for KBP, in contrast with IE task, is the paucity of training data. The official training data, linked to specific text from specific documents, consists of responses to 100 queries and many more responses could be further generated. So traditional supervised learning, based directly on the training data, would provide limited coverage. Coverage could be improved by using the training data as seeds for a bootstrapping procedure.

IV. SLOT FILLING

The goal of Slot Filling is to collect from the cor- pus information regarding certain attributes of an entity, which may be a person or some type of organization. Each query in the Slot Filling task consists of the name of the entity, its type (person or organization), a background document containing the name (again, to disambiguate the query in case there are multiple entities with the same name), its node ID (if the entity appears in the knowledge base), and the attributes which need not be filled. Attributes are excluded if they are already filled in the reference data base and can only take on a single value. Along with each slot fill, the system must provide the ID of a document which supports the correctness of this fill. If the corpus does not provide any information for a given attribute, the system should generate a NIL response (and no document ID).

KBP2010 defined 26 types of attributes for persons (such as the age, birthplace, spouse, children, job title, and employing organization) and 16 types of attributes for organizations (such as the top employees, the founder, the year founded, the headquarters location, and subsidiaries). Some of these attributes are specified as only taking a single value (e.g., birthplace), while some can take multiple values (e.g., top employees).The reference KB includes hundreds of thousands of entities based on articles from an October 2008 dump of English Wikipedia which includes818,741 nodes. The source collection includes 1,286,609 newswire documents, 490,596 web documents and hundreds of transcribed spoken documents.

To score Entity Linking, one take each query and check whether the KB node ID (or NIL) returned by a system is correct or not. Then compute the Micro-averaged Accuracy, computed across all queries.

Slot Filling task is to complete all known information about a given query entity. E.g. given the query "Barack Obama", the system's goal is to collect Barack Obama's birthplace, birthdate, occupation, spouse, etc. This can be thought of as "filling" A key aspect of this is relation extraction is the classification of a sentence and two entities in the sentence to a relation of interest. For example, reading Barack Obama was born in Hawaii" and extracting the relation born_in(Barack Obama, Hawaii).The Slot Filling task required to automatically distill information from the document collection which fills missing KB attributes for focus entities. The slot-filling task is a hybrid of traditional IE (a fixed set of relations) and QA (responding to a query, generating a unified response from a large collection).

V. SLOT FILLING ARCHITECTURE

The basic Slot filling architecture consists of three phases:

A. **Document/passage retrieval:-** Retrieving passages involving the queried entity.
B. **Answer extraction:-** Getting specific answers from the retrieved passages.
C. **Answer combination:-** Merging and selecting among the answers extracted

VI. ISSUES TO CONSIDER

Cross-slot inference based on revertible queries, propagation links or world knowledge is required to capture some of the most challenging cases in KBP. In the KBP slot filling task, slots are often dependent on each other, so one can improve the results by maintaining consistency among all generated answers.

KBP Slot Filling is similar to Relation Extraction, which has been extensively studied for the past years. But as the amount of training data available is much training strategies need to be adjusted. Also, some of the constraints like both arguments should be present in the same sentence makes co reference and cross-sentence inference more critical. The increased number of diverse approaches proposed, provide new opportunities for both entity linking and slot filling tasks to benefit from system combination.

VII. CONCLUSION

Compared to traditional IE and QA tasks, KBP has raised some interesting and important research issues: It places more emphasis on cross-document entity resolution which received limited effort earlier. it forces systems to deal with redundant and conflicting answers across large corpora. it links the facts in text to a knowledge base so that NLP and data mining/database communities have a better chance to collaborate.

The increased number of diverse approaches proposed, provide new opportunities for both entity linking and slot filling tasks of KBP. In this paper, detailed analysis of the reasons which have made KBP a more challenging task is presented and suggested some possible research directions to address these challenges which may be helpful for current and new participants, or IE and QA researchers in general.

REFERENCES

[1] Heng Ji and Ralph Grishman. 2011. Knowledge Base Population: Successful Approaches and Challenges. *Proc. 49th Annual Meeting Assn. Computational Linguistics (ACL 2011).*

[2] Heng Ji, Ralph Grishman, and Hoa Trang Dang. 2011. Overview of the TAC2011 Knowledge Base Population Track. *Proc. Text Analysis Conference (TAC 2011)*

[3] Praveen Bysani, Kranthi Reddy, Vijay Bharath Reddy, Sudheer Kovelamudi, Prasad Pingali and Vasudeva Varma. 2010. IIIT Hyderabad in Guided Summariza- tion and Knowledge Base Population. *Proc. TAC2010 Workshop*.

[4] Zheng Chen, Suzanne Tamang, Adam Lee, Xiang Li, Wen-Pin Lin, Matthew Snover, Javier Artiles, Marissa Passantino and Heng Ji. 2010. CUNY- BLENDER TAC-KBP2010 Entity Linking and Slot Filling System Description. *Proc. TAC 2010 Work- shop*

[5] Mark Dredze, Paul McNamee, Delip Rao, Adam Gerber and Tim Finin. 2010. Entity Disambiguation for Knowledge Base Population. *Proc. COLING 2010*

[6] Ralph Grishman and Bonan Min. 2010. New York University KBP 2010 Slot-FillingSystem. *Proc. Text Analysis Conference (TAC 2010).*

[7] Vittorio Castelli, Radu Florian and Ding-jung Han. 2010. Slot Filling through Statistical Processing and Inference Rules. *Proc.Text Analysis Conference (TAC2010).*

[8] Ralph Grishman and Bonan Min. 2010. New York University KBP 2010 Slot-Filling System. *Proc. Text Analysis Conference (TAC 2010).*

[9] Mihai Surdeanu, Sonal Gupta, John Bauer, David McClosky, Angel X. Chang, Valentin I. Spitkovsky, Christopher D. Manning. 2010. Stanford's Distantly-Supervised Slot-Filling System. *Proc. Text Analysis Conference (TAC 2010).*

