CS410 Technology Review - Suxiang Han

Google knowledge vault: a new probabilistic approach of knowledge base.

Abstract:

In this article, we will go through the basic concepts and structure of the knowledge base. Then, we will introduce one of them called Google Knowledge Base (KV) which presents a web-scale approach to probabilistic knowledge fusion [4]. We will walk through the detail/structure/methods of KV along with its evaluation. Lastly, we will discuss its characteristics and possible aspects of improvement/limitation.

Knowledge base:

Google knowledge vault is a new kind of knowledge base structure/application proposed by Google. In order to understand Google's knowledge vault, the details of the knowledge base needs to be clarified.

In short, Knowledge base is a collection of entities, facts, relationships that conforms with a certain data model [1]. In a more general approach, the knowledge base is just an approach to store complex organized or unorganized information. Compared to databases where information is stored in tabular format where we strictly define rows and columns [2]. Knowledge base store documents will focus more on “knowledge” compared to “fact”.

For instance, if a database stores the salary of every worker in Champaign along with their name, the user will be expected to enter precise queries to obtain precise facts. Specifically, the salary of a user named “Sam”, number of people (row) with a salary more than 50. However, the knowledge base will store/summarize such knowledge where the system will actively post such information to the user. For instance, in this example, the knowledge base will display some information like “People in champaign have a high salary”. “Female workers have higher salaries”, etc.

Meanwhile, a general approach of the knowledge base system might store information in the forms of RDF triples (subject, predicate, object), such as NELL [3]. The system might also contain various functionality including extraction, expansion, evolution and integration. These cover aspects of fetching and analyzing/retrieving knowledge from given general documents.

Google knowledge vault:

After introducing the concept of knowledge base, we can now dive into the details of KV. KV has a similar structure to what we have mentioned above in the previous section. KV also stores information in RDF triples (for instance: <Champaign, organization/university, UIUC>). Differ from other knowledge bases such as Freebase and NELL, KV aims to include as large as 1.6B triples [4]. As a result, KV has quite a different base construction which combines noisy distraction and prior knowledge.

Meanwhile, KV contains three major components: Extractors (extraction), Graph-based prior(expansion, evolution) and knowledge fusion(integration). Which we will specific in the following paragraph:

- **Extractors**

  The extraction of KV is built from a combination of three different methods which aim at three different scenarios: Text documents, HTML tree, HTML tables and Human annotation [4]. Text documents will generate the most RDF triples and its methods work as a "baseline"[4]. A standard nlp tool will be runned across document documents to provide background data (as in CS410), and a relations extractors is trained using a distant supervisor [7]. In specific, this will extract predicates and will find corresponding sets of entity pairs from existing KB, such as Freebase. Then by finding the example of entity pairs along with predicates from the documents, features(RDF pairs) will be obtained. Meanwhile, the other three scenarios apply similar methods but modify some detail to fit the situation. For instance, when processing DOM KV use a tree representation instead and use lexicalIzed path as a feature vector, while for human annotated page a smaller subset and manual mapping will be applied for better compromise the annotated information on the webpage

- **Graph-based prior**

  After extractor extracts various information from the web a further analysis should be applied as internet information might be incorrect and biased (For instance, spam website or outdated information). Therefore, by using existing prior in other KB, probability will be assigned to triples (even for those not extracted from extractors) to build up a prior, which converts this problem to something graphically similar to find other edges based on existing edges. Specifically, two different methods can match this requirement, which is Path ranking algorithm and Neural network model [8].

- **Graph-based prior**

  After the extractors and Graph-based prior is applied, an integration called knowledge fusion takes place. This step will use the prior to assign probability to the RDF triple obtained by extractors. Specifically, boosted decision stumps take place in order to create a non-linear decision boundary [4]. And while these methods can better integrate the four different extractors we used for different scenarios, this will generate a better threshold compared to those by linear logistic regression.

Conclusion and Future:

Overall, we can observe that Google Knowledge Vault maintains practices of a classical Knowledge Base but also has its unique advantage. By using various methods for different scenarios and prior training from existing knowledge base, KV can handle

heavier workload and can have a higher accuracy. However, we can still seeks for further development as the system still has some limitation: Some information might has time issue where only incorrect/correct in a specific time, RDF triples might not present some of the information, some information might not be text-based, how can we improve the prior base using the trainning result, and even how should we avoid ethical issue when the size of the system become too big and stores too much personal information. All of these questions should be worth considering and can provide an insight for future development.

Reference:

1. http://cidrdb.org/cidr2013/Talks/CIDR13_Gongshow15.pdf
2. https://www.ontotext.com/knowledgehub/fundamentals/what-is-a-knowledge-base/
3. http://ceur-ws.org/Vol-992/paper2.pdf
4. https://storage.googleapis.com/pub-tools-public-publication-data/pdf/45634.pdf
5. https://aclanthology.org/D11-1049.pdf
6. https://cs.nyu.edu/~grishman/tarragona.pdf

7. https://web.stanford.edu/~jurafsky/mintz.pdf
8. https://arxiv.org/pdf/1301.3781.pdf