CS- 410 Tech Review
Name: Uday Kanth Reddy Kakarla

Overview of Google Knowledge Vault and its difference from the existing Knowledge Base construction approaches

Introduction

Large-scale knowledge bases (KBs) such as Google Graph store millions of facts about the world, such as information about people, places, and things. This information or facts is stored in the form of relationships between entities in the form of RDF Triples (subject, predicate, object). Now, despite the seemingly large size of these knowledge bases, these repositories are still far from complete as they don't contain all the knowledge about the world. Further, standard methods for building these knowledge bases often produce very noisy, unreliable facts. Therefore, the authors of Knowledge Vault proposed a new way of automatically constructing a web-scale probabilistic knowledge base. Here, the Knowledge Vault is built by combining various extraction methods which collect information from various sources such as Text, Webpages, HTML tables etc. with the knowledge learned from the existing Knowledge Bases, thus leading to KV producing accurate results. Additionally, Google's Knowledge Vault uses supervised machine learning methods to verify the information extracted from the web. They use previously known information about an entity to make an informed guess.

Methods used for the construction of the Knowledge Vault

The authors of KV use relatively standard methods for relation extraction from text, but they do so on a much larger scale than previous systems. At first, a suite of standard NLP tools are run over each document extracted. These perform named entity recognition, part of speech tagging, dependency parsing, co-reference resolution, and entity linkage. Then, for each predicate of interest, they extract a seed set of entity pairs that have this predicate from an existing KB. Then, they find examples of sentences in which this pair is mentioned and extract features or patterns from all of these sentences. By using these features they tend to assign the probability to the particular RDF triple that has been mined. In order to conduct link prediction across entities connected by some predicate, Google's knowledge vault also employs a path ranking algorithm. These connections may be examined to establish whether two entities are related in any way and, if so, through which other entities. As discussed before, different priors can also be combined. These priors may be merged with the feature vector, which includes the confidence levels from each prior system's vector as well as whether or not the prior was successful in making correct predictions. Because they both compliment one another and each have their own advantages, combining the two earlier techniques can improve performance.
The authors also suggests an alternative approach to building the prior model by visualizing the link prediction problem as matrix completion. They have developed a state-of-the-art neural network model, and when they applied the Knowledge Vault data, they saw that the model learned to put semantically related (but not necessarily similar) predicates near each other.

Results

Using the methods described, the authors have extracted about 1.6 billion candidate triples, covering 4469 different types of relations and 1100 different types of entities. About 271 million of these facts have an estimated probability of being true above 90%; they are called "confident facts." This makes Google's knowledge vault 38 times larger than the next largest comparable system, which had a mere 7 million confident facts.

Discussion

Knowledge Vault is a large repository of useful knowledge, but there are still many ways in which it can be improved. For a functional relation such as born-in, there can only be one true value, so the (s, p, oi) triples representing different values oi for the same subject s and predicate p become correlated due to the mutual exclusion constraint. So, a simple way to handle this is to collect all candidate values together and force the distribution over them to sum to 1. This is similar to the notion of an X-tuple in probabilistic databases.

We might have a fact that Obama was born in Honolulu and another one stating he was born in Hawaii. These are not mutually exclusive, so the naive approach does not work.

Conclusion

In this paper, a description of how to build a web-scale probabilistic knowledge base, called Knowledge Vault, has been proposed. In contrast to previous work, the fusion of multiple extraction sources with prior knowledge derived from an existing KB has been implemented. The resulting knowledge base is about 38 times bigger than existing, automatically constructed KBs. The facts in KV have associated probabilities, which we show are well-calibrated so that we can distinguish what we know with high confidence from what we are uncertain about.

The authors hope to continue to scale KV, to store more knowledge about the world, and to use this resource to help various downstream applications, such as question answering, entity-based search, etc.

Google Knowledge Graph vs Google Knowledge Vault

The combining of several ways of extraction and validation to create a much more solid and trustworthy database entities is where the actual value of the methodology described in Knowledge Vault lies. In other words, Google has figured out how to expand and strengthen the Knowledge Graph. In essence, Google is triangulating the apps that make sense out of all the entity applications. The greater part of extractions originates from Google's crawler of websites' HTML trees, which is unstructured data that they transform into useful information. But the need for organized data remains.

Further, the fact that humans are involved in the Knowledge vault process is another noteworthy conclusion. This is mentioned in the report and adds some additional support to the notion that Google wants to have some form of author- or expertise-based ranking but wants to implement it via entities rather than explicit markup.

There is a component of link prediction and how Google is able to provide entities more context by knowing the kinds of information they should look for. They anticipate acquiring information about each individual such as "spouse," "date of birth," "place of birth," "gender," "parents," "children," etc. It is quite powerful to be able to produce these connection predictions and automatically search them out. The usage in this variety of scenarios is expected to increase as the Knowledge Vault develops, and Google's "trust" in it. This is moving Google closer to its ultimate goal of becoming a "ask me anything" Star-Trek computer. Although a substantial body of knowledge would be needed to accomplish this. If it were put together it would open up a world of new applications and computer possibilities.

References

Xin Luna Dong Evgeniy Gabrilovich Geremy Heitz Wilko Horn Ni Lao Kevin Murphy Thomas Strohmann Shaohua Sun Wei Zhang The 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '14, New York, NY, USA - August 24 - 27, 2014, pp. 601-610