

Sublexical Compositionality in Semantic Parsing

Diana Wan, Gunaa Arumugam Veerapandian

CS221 Project Progress Report,
Stanford University

jdwan@stanford.edu, avgunaa@stanford.edu

1. Introduction

2. Problem Description

3. Aligning Predicates from Question Answer Pairs

4. Data collection

The data used in the project is basically obtained from two sources, one portion comes from the freebase data and the other is in the form of text triples.

Freebase essentially contains a massive amount of data on various topics, we have gathered a subset of 14,873,062 data entries from coherent topics and categories like People, Business, Government and Organization. Each entry in this set is in the form of triples ($arg1 \rightarrow reln \rightarrow arg2$). For example, $fb : en.viswanathan_anand \rightarrow fb : people.person.profession \rightarrow fb : en.chess_master$.

The text data consists of triples with text instead of freebase predicates. It is a subset of the data used in the original paper and it is again of the form ($arg1 \rightarrow reln \rightarrow arg2$). An example of this database would be "*Vishwanathan Anand*" \rightarrow "*profession of*" \rightarrow "*chess master*". We have around 3,000,000 data entries of this kind.

For both the data sets, the arguments are referred to as entities and the relationship is referred to as the edge between two entities in later parts of the report.

The basic goal of alignment using question answer pairs is to align "profession of" with fb:people.person.profession using certain techniques. For every such alignment we have a set of features like text frequency, KB frequency, intersection size, etc.

Let us now look at the algorithm by which we can align possible formulas or predicates for different words.

5. Algorithm

Given two sets of triples, the knowledge base triples and the text based triples we learn the alignment by creating templates and looking for similar template pattern matches in the data.

The triples can be interpreted as a graph with entities as vertices and relationships as edges. Let us look at some templates which

6. Examples

7. Baseline

The algorithm we use

8. Oracle

9. Algorithm

10. Testing

11. Future work