Hierarchical Relation Extraction with Graph Techniques

CS 224U: Natural Language Understanding
Experimental Protocol

Aakash Pattabi
Dept. of Economics
{apattabi@stanford.edu}

Ben Barnett
Dept. of Computer Science
{ben.barnett@stanford.edu}

May 27, 2019

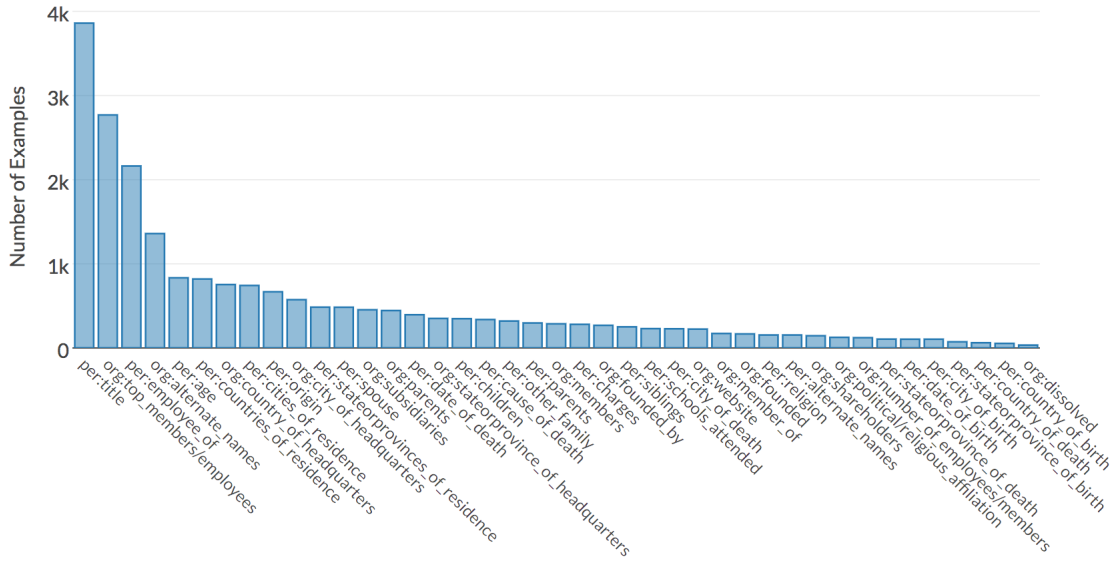
Experimental Protocol

I. Hypotheses & General Reasoning

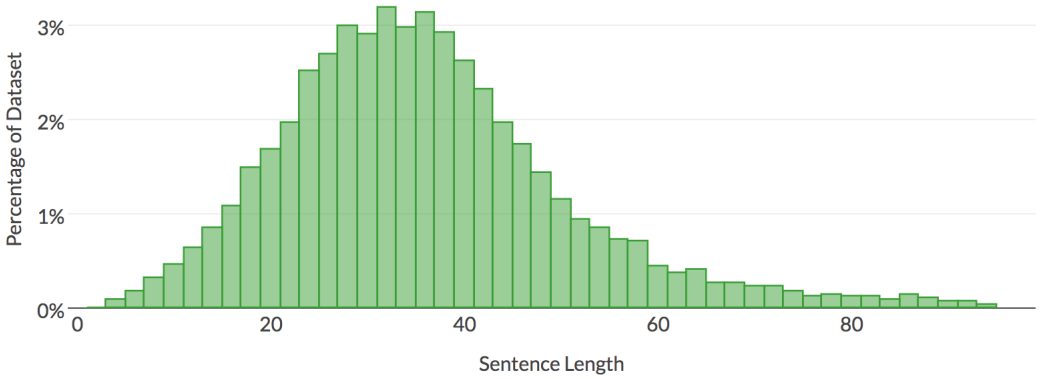
We hypothesize that the use of graph-based relational embeddings will outperform traditional models including pattern-based extractors and logistic regressions in accurately identifying sentence relations. We suspect that the graph features contain sentence parsing information that is otherwise difficult for models to incorporate. Therefore, when combined with node2vec, which embeds individual words, we anticipate that graph-based features will be especially successful. Although more successful models have been published that use deep neural networks, we are interested in exploring the benefits of graph-based relational embeddings and believe it will be easier to examine its impact within the context of a simpler model, as opposed to more advanced CNNS, for example.

II. Data

For this experiment we'll be using TACRED, the (supervised) TAC Relation Extraction Dataset. Created at Stanford in 2017, TACRED contains over 100,000 sentences with labels for 41 relation types. The dataset also contains the spans of subject and object mentions, as well as 23 types of mentions (e.g. "Person" or "City"). The following is a distribution of relations in the dataset:



TACRED contains sentences slightly longer than many NLP datasets, which the creators attribute to "the complexity of contexts in which relations occur in real-world text" [4].



III. Metrics

Given that several baselines models (and their metrics) are already associated with TACRED, we will maintain consistency and use the same scores to measure success: precision, recall, and (macro) F1. In addition to consistency with past experiments being desirable in order to make meaningful comparisons, F1-scores are helpful metrics for statistical models because they incorporate precision *and* recall, as neither precision nor recall are individually as helpful for interpreting a model's success; in fact, F1 is defined as the harmonic mean of precision and recall:

$$\text{F1} = 2 * \frac{\text{precision} * \text{recall}}{\text{precision} + \text{recall}}$$

IV. Models

As a baseline, we will use Stanford's top performing system on the TAC KBP 2015 cold start slot filling task [1]. (Note that this is the same baseline used as in [4].) This model effectively combines a pattern-based extractor with a logistic regression classifier. Similar to models we've explored in class, the baseline extracts relations using n-grams, POS tags, entity positional tags, etc. In contrast, our model will incorporate features produced from graph representations of parsed sentences in place of pattern recognition techniques.

V. Progress So Far

We have downloaded the TACRED sample dataset (though have not yet requested access to the full dataset, which the website suggests we can use as Stanford students). Based on the data format in the sample dataset, we have started working on the following graph-based feature extractors:

i. A feature extractor that converts a parse tree to a graph by adding an additional "vantage point" node outside the parse tree graph and running `node2vec` from this vantage point. This method intuitively performs a probabilistic version of the graph-based sentence construction achieved by [3] and [2], namely the "document graph" method tested in that research.

ii. A feature extractor that uses the sentence document graphs from [3] and converts each to a feature vector using `graph2vec`.

Note that we do plan to use GloVe vector information as well, as previous research on TACRED has found that using GloVe vectors can have meaningfully positive effects on models' F1 scores. Incorporating GloVe information by e.g. taking the average or element-wise max of all GloVe vectors of the words in each sentence (or each *bridge* between the relation object and subject) will be tried as well. Finally, we are in the process of adapting the relation extraction test harness used in class to work on the TACRED dataset so that we can perform direct comparisons of our feature extractors to the pattern- and neural-based models in the TACRED benchmarks described above.

References

[1] Angeli Gabor et al. *Bootstrapped self training for knowledge base population. In Text Analysis Conference (TAC) Proceedings 2015*. 2015.

[2] Nanyun Peng et al. "Cross-Sentence N-ary Relation Extraction with Graph LSTMs". In: *Transactions of the Association for Computational Linguistics* 5 (2017). URL: https://www.cs.jhu.edu/~npeng/papers/TACL_17_RelationExtraction.pdf.

[3] Yuhao Zhang, Peng Qi, and Christopher D. Manning. "Graph Convolution over Pruned Dependency Trees Improves Relation Extraction". In: *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing* (2018), pp. 2205–2215. URL: https://aclweb.org/anthology/D18-12443.

[4] Yuhao Zhang et al. *Position-aware Attention and Supervised Data Improve Slot Filling*. 2017. URL: https://nlp.stanford.edu/pubs/zhang2017tacred.pdf.