Learning relation entailment graphs

Justin Huang, Mitchell Koch | CSE 515 Spring 2013

Definitions

Given textual binary relations:

grown-in(X, Y)
conquer(X, Y)

An entailment is an implied relation. For example:

capture(X, Y) → conquer(X, Y)
seize(X, Y) → conquer(X, Y)
produce-in(X, Y) → grown-in(X, Y)

A relation entailment graph has relation strings as nodes, with directed edges representing entailments. This can be useful for inference and predicting data.

Can learning relation entailment graphs based on WordNet or Bayes net structure learning improve relation query performance?

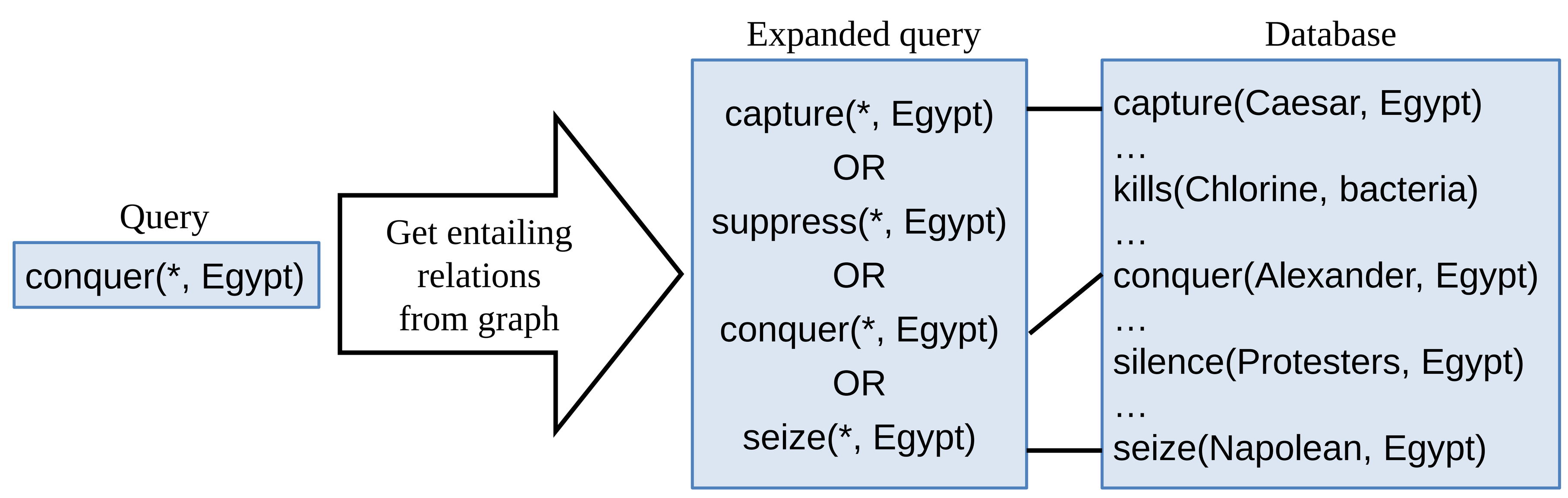
We try two methods for learning an entailment graph:

1. Learning constraints on WordNet entailments
2. Bayesian network structure learning

Database query task

Given a query for relation R with arguments X and Y, we expand the query to include all entailing relations:

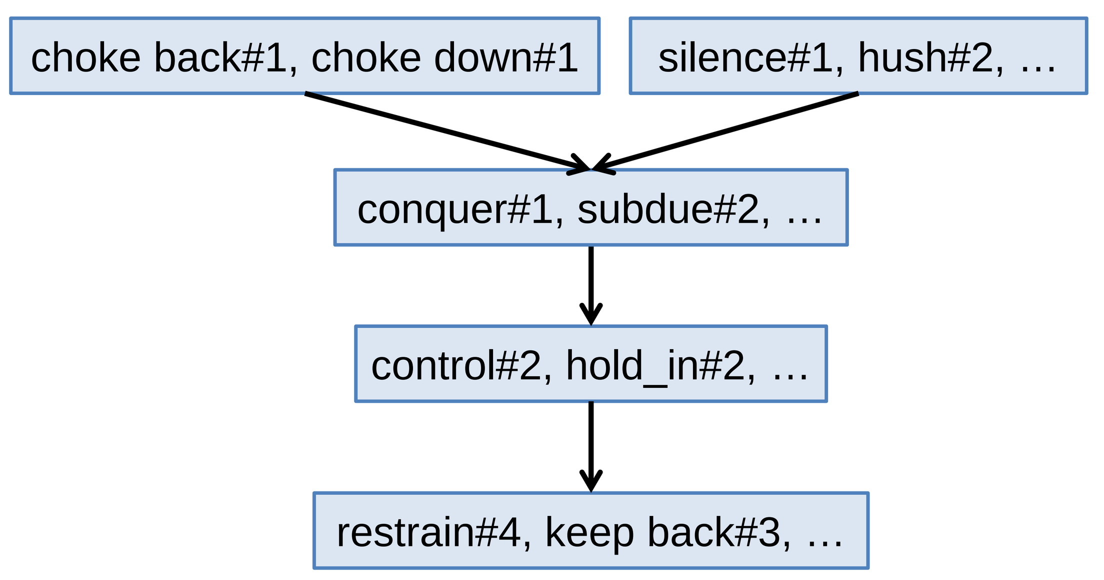
Original query: $R(X, Y)$
Expanded query: $R'(X, Y)$ for all R' such that $R' \to R$ in the entailment graph.



An example of a query being expanded.

Constraints on WordNet

WordNet (wordnet.princeton.edu) is a hand-crafted resource that distinguishes between different word senses, and provides synonyms and entailments between them.



Component of the WordNet entailment graph with the first sense of conquer, conquer#1.
Boxes represent synonym sets, arrows represent entailments.

Challenge:

- Given the verb "take," is it take#21 (take by force) or take#2 (take time)?
- How do we use the entailment graph?

Our approach:

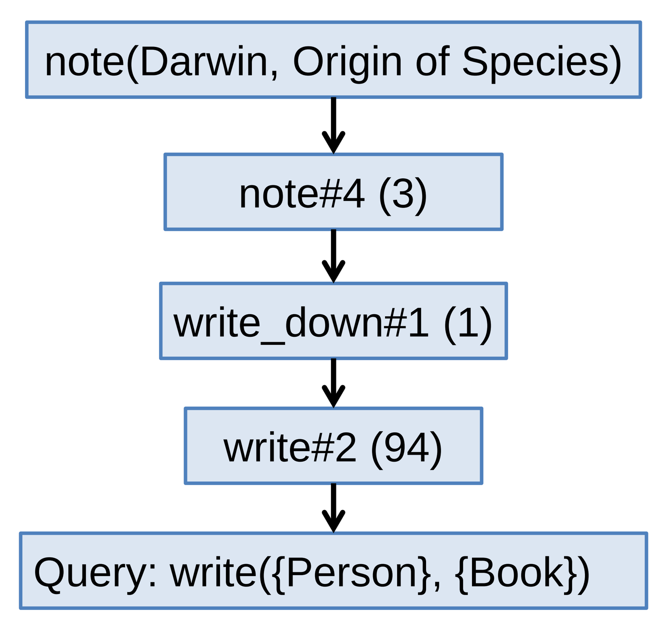
1. Assume a string can be any WordNet sense ("take" = take#1, take#2, …, take#42)
2. Attempt the database query task using all possible entailments.
3. Label results as correct/incorrect
4. Split results into train / test sets, gather features of the path taken through the entailment graph that connected the result to the query.
5. Train a model of the probability that the path is correct given features of the path using logistic regression.
6. Evaluate the model on the test set.

Each WordNet sense has

- A count, indicating how often that sense is used.
- A sense number, where the 1st sense of a word is the most common.
- A probability, which is the count divided by the sum of counts for all senses of the same word.

Our features for a path in the entailment graph are:

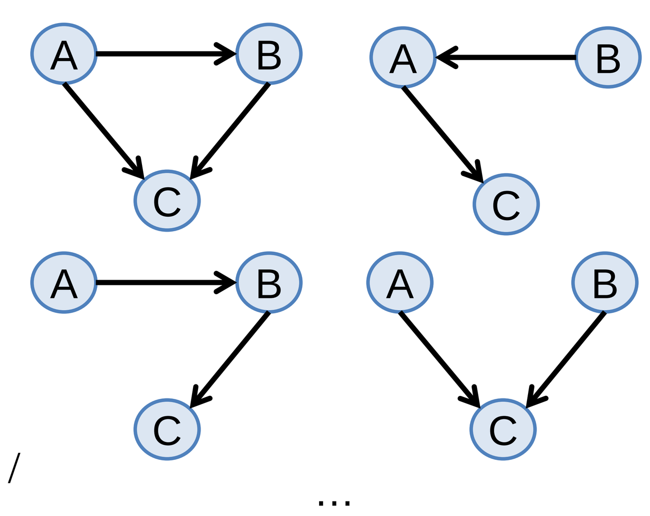
- Path length
- Average sense number
- Average WordNet probability
- Maximum sense number
- Minimum WordNet probability



Example path in entailment graph

Bayes net structure learning

- Structure learning maximizes the probability of the graph given the data, using a uniform prior: $\text{argmax}_G \log P(D|G) + \log P(G)$.
- We used the Bayes Net Toolbox for MATLAB (bnt.googlecode.com).
- The search done using 100 samples of Markov Chain Monte Carlo (MCMC), with a burn-in of 10.
- The input data was a matrix with the count of relation / entity pair occurrences. We also tried smoothing by adding 0.1 to all counts. Due to memory constraints, we had to limit each matrix to a sample of 150 relations.



Examples of possible network structures

Experiments

We evaluated our entailment graphs by querying ReVerb extractions from Open Information Extraction (openie.cs.washington.edu) for a certain set of queries. Then, we labeled each result returned as correct if it answered the query, or incorrect otherwise.

We tested four systems:

- **Baseline:** no query expansion
- **Structure learning:** expand query using entailments found from structure learning.
- **WordNet:** expand query using WordNet entailments
- **WordNet + logistic regression:** expand query using WordNet entailments, filtering results using our path correctness model, with a 75/25 train/test split.

Our experiment does not really demonstrate the utility of the structure learning entailments, for a couple reasons:

1. Limiting the input matrices to 150 relations leads to sparsity in the output
2. The set of queries we tested with was so small that none of the entailments we learned through structure learning were used by the queries.

System	# results returned	Precision
Baseline	115	83.48%
Structure learning	115	83.48%
WordNet	1103	86.85%
WordNet + logistic regression	981	95.01%

Summary of results

Queries

- grown-in(coffee, {Country})
- write({Person}, {Book})
- conquer(?, Egypt)
- play(Tom Hanks, ?)
- kill(?, Voldemort)

Conclusions

- WordNet entailments alone can provide a substantial increase in yield. Including logistic regression introduces a precision/yield tradeoff.
- In the future, we hope to get results using a much larger set of queries, which will show the effect of using the structure learning entailments.
- We expect that precision will be lower when using structure learning, since an edge between two relations indicates an influence between the parent and the child, but not necessarily an entailment.