CS6390 PROJECT PROPOSAL: REDIRE

OLIVER RICHARDSON *MAKS CEGIELSKI-JOHNSON*

1. Description and Approach

We are ultimately interested in generating paraphrases for a given sentence, but we will begin our project by building a paraphrase recognition tool – so while we may end up focusing entirely on paraphrase recognition, we are not interested in training a purely discriminative model, but are instead looking to extend it to aid in paraphrase generation. We're not entirely certain what our approach will be (we will need to see what is in most dire need of attention after we build a baseline system) – but we have some ideas :

- Run a dependency parse, and align graphs; use a classifier to recognize if there exists a sufficiently good alignment to call the sentences paraphrases
- Use parallel monolingual corpora to induce a grammar for synonymous constructions
- Train classifiers to recognize essential words, and separately, extract a minimal grammar with maximum expresivity. Then, build a new sentence from this grammar and the essential words

These are just ideas off the top of our heads; we have not yet committed to a strategy.

2. Baseline Approach

There are several simple approaches we could use to get a recognition system running. We will train some simple machine learning classifier with features from PPDB, and re-implement the DIRT algorithm as a baseline for paraphrase or entailment recognition. The base system for generation will just do wordnet lookups to find synonyms, and replace noun phrases. We suspect that this is more or less what the popular-but-suspiciously-undocumented web sensation `spinbot` does.

3. Data Sources

3.1. **Penn Paraphrase Database (PPDB).**

A large set of parallel English phrases. This dataset appears to be very noisy, but is a large set of aligned incomplete phrases. Available at `http://www.cis.upenn.edu/~ccb/ppdb/`

3.2. **Microsoft Paraphrasing Corpus (MSRPC).**

Contains lists of pairs of full sentences, with a label corresponding to whether or not the two are paraphrases, and has already been split into training and test sets. Available at `http://research.microsoft.com/en-us/downloads/607d14d9-20cd-47e3-85bc-a2f65cd28042`

3.3. **SemEval Data for Paraphrasing (Textual Entailment).**

A set of question-response data which we will try to figure out how to use, and seems particularly promising where there are multiple responses given for a single question. Available at `https://www.cs.york.ac.uk/semeval-2013/task7/index.php%3Fid=data.html`

4. Evaluation

Evaluating the recognition tool should be fairly straightforeward; we have training and test data from MSRPC, with which we can measure accuracies and F-score. As we begin to focus on generation, the process of evaluation will become less straightforeward, but we have several evaluation metrics in mind:

(1) Finding a lower bound on accuracy by checking paraphrase datasets to see if we generated (more-or-less) an exact paraphrase. We can score reasonably close paraphrases by using wordnet lookups.

(2) Repurposing our Paraphrase Recognizer for testing – a tempting first pass, but suffers from two issues: it will probably not be accurate enough to test, and is also not independent.

(3) **BLEU** metric for evaluating the quality of machine translations

(4) **ROUGE** metrics for evaluating the quality of summarizations – this will require a lot of data, perhaps more than we can obtain reasonably, but we are still looking into it.

In addition to semantic entailment of some sort, we would also like require that the sentences are sufficiently dissimilar – a sentence hardly counts as a paraphrase of itself. To that end, we will also devise a scoring metric that awardds points based on the magnitude of change in dependency graph structure.

5. Project Name

Recognition
Entails
Discovering
Inference
Rules,
Ellen

Rédire means “to repeat” in french.