# Labeling Synonyms for Query Expansion Using Crowdsourcing and a Search Engine

**Bruce Smith, Chris Collins, and Pankaj Andhale**
Intuit, Inc.
Mountain View, CA
{Bruce_Smith, Christopher_Collins, Pankaj_Andhale}@intuit.com

### Abstract

Query expansion using synonyms can improve a search engine's recall, but using synonyms from the wrong domain can decrease precision. Our search application is customer self-help for Intuit products, so the relevant domains include small business accounting, personal income tax and consumer software. Lexical databases, such as WordNet, contain a large number of synsets, but many of these will not be appropriate for our domains. We describe how we use our search engine and crowdsourcing to label synonyms from WordNet with the domains where we can safely use them in query expansion. This procedure could be useful in other situations where it is necessary to give domain labels to synonyms.

## Query Expansion and Synonyms

Our team provides search services for customer self-help with Intuit products. For example, an Intuit customer filing his taxes might want to know "Can I deduct my grandmother's dance lessons?" or "How do I print my tax return?" In general, the relevant domains for our applications include small business accounting, personal income taxes, and consumer software.

Query expansion using synonyms can improve a search engine's recall (Witten, Moffat, and Bell 1999). An Intuit customer searching for help with a "lost bill" would benefit from results about a "lost invoice". We can automatically add 'invoice' to the query "lost bill", because in our domains the nouns 'bill' and 'invoice' are synonyms.

Query expansion using inappropriate synonyms, however, can decrease a search engine's precision. In other domains the verbs 'open' and 'afford' are synonyms, but the questions "How do I open an account?" and "How do I afford an account?" have different meanings. Adding 'afford' to the query "open account" would probably cause our search engine to return additional, irrelevant results

Lexical databases, such as WordNet, contain many sets of synonyms, known as *synsets* (Miller 1995). Whenever possible, we prefer to use public data sets rather than build our own. However, while many of the synonym pairs in WordNet's synsets are appropriate for our domains, many are not. For example, both the pair of nouns 'invoice' and 'bill' and the pair of verbs 'open' and 'afford' were taken from WordNet's synsets.

## Labeling Synonyms

Now we describe the process we use to label synonyms.

We define a domain implicitly by a corpus of help documents. These documents might be written by domain experts, or they might be user-generated content from online forums. If a corpus contains information about multiple domains, we can partition it via tags supplied by users or by automatic document classification. Rather than explaining a domain to the crowd workers, we show candidate synonyms in the context of sentences from documents in one of our corpora.

We use our search engine in several ways: determining the document frequency of a word in a corpus, finding documents in a corpus that contain a word, and finding sentences in those documents that contain a word.

### Preparing Data

Each unit of work for our crowdsourcing job is defined by a 4-tuple $(w_1, w_2, s_1, s_2)$ where $w_1$ and $w_2$ are words or phrases, and $s_1$ and $s_2$ are sentences. In particular, $w_1$ and $w_2$ are in a WordNet synset, $s_1$ is a sentence from a document in one of our corpora and contains $w_1$, and $s_2$ is derived from $s_1$ by replacing $w_1$ with $w_2$.

Of course, each $w_1$ and $w_2$ will occur in several 4-tuples, where each 4-tuple has a different $s_1$. We do not want to label synonyms based on the crowd's judgments about a single sentence.

### Synonym Pairs

After we choose a domain (and corpus), we find pairs $w_1$and $w_2$that meet the following criteria:

- $w_1$ and $w_2$ occur in a WordNet synset, with the same part of speech.
- $w_1$ occurs (or occurs often enough) in our logs. This makes sense for our search application, since we needn't expand words that never (or only rarely) occur in queries.
- $w_2$ has non-zero (or at least high enough) document frequency in our corpus. This makes sense for our search application, since we needn't expand queries to include words that never occur (or occur only rarely) in our corpus.

If a WordNet synset does not contain at least one $w_1$ and one $w_2$ that meet these criteria, we can ignore that synset.

These criteria allow us to reject synonym pairs such as 'stock' and 'broth' without depending on the crowd's judgments. While 'stock' occurs frequently in our logs and in our corpora, 'broth' occurs in neither. This pruning reduces the number of synonym pairs requiring judgments to thousands per domain.

### Sentences

Once we have pairs of words, we extend these to triples $w_1$, $w_2$, and $s_1$ that meet the following criteria:

- $s_1$ occurs in a document from our corpus and contains $w_1$.
- $s_1$ is no shorter than some minimum length. Sentences that are too short provide little context for the crowd, or they may be the result of errors in sentence boundary detection.
- $s_1$ is no longer than some maximum length. Sentences that are too long may be too hard for crowd workers to evaluate, or they may be the result of errors in sentence boundary detection.

If a pair of words has no sentences that meet these criteria, we drop that pair of words, and this provides additional pruning.

These sentences are occasionally ungrammatical or nonsensical. This might be the result of errors in sentence boundary detection. Or, in some of our corpora, it might be the sometimes erratic quality of user-generated content.

## Crowdsourcing

For each work unit $(w_1, w_2, s_1, s_2)$, we ask crowd workers whether replacing $w_1$ with $w_2$ in sentence $s_1$ changes the meaning of the sentence. In particular, we ask:

- Do the sentences have the same (or nearly the same) meaning?
- Do the sentences have different meanings?
- Are the sentences too hard to understand?
- Are one or both of the sentences nonsense?

The first answer counts as a yes vote for labeling the synonym pair with this domain, and the second counts as a no vote. The third and fourth answers indicate 4-tuples that are not useful in making the decision.

We label a pair of synonyms $w_1$ and $w_2$ with a domain if enough crowd workers gave enough yes votes for enough sentences containing $w_1$ and $w_2$. In our tests so far, enough has been 3 of 5 crowd workers and 3 of 5 sentences.

## Results

Early tests suggest that:

- Our procedure can generate work units for the crowdsourcing job.
- Crowd workers can perform the tasks with sufficient quality and speed for our needs.
- Examination of a sample of results shows strong agreement with our intuitions.

Because our focus is on search, our next steps are to measure the impact of query expansion on relevance.

## Future Work

We are currently testing the effects of query expansion on relevance. We do this this by computing the discounted cumulative gain (Jarvelin and Kekalainen 2002) for sample queries, chosen from our logs. (This, in turn, is based on crowdsourced relevance judgments.) Additional relevant results will tend to increase the discounted cumulative gain, whereas additional irrelevant results will tend to decrease it. The discounted cumulative gain on our sample queries will indicate which effect is greater.

## References

Jarvelin, K. and Kekalainen, J. 2002. Cumulated gain-based evaluation of IR techniques. *ACM Transactions on Information Systems* 20(4): 422–446.

Miller, G. A. 1995. WordNet: A Lexical Database for English. *Communications of the ACM* 38(11): 39-41.

Witten, I.H., Moffat, A., and Bell, T.C. 1999. *Managing Gigabytes: Compressing and Indexing Documents and Images*. San Francisco, CA: Morgan Kaufmann Publishers, Inc.