Word Sense Disambiguation in Web Search Results through Word Co-occurrence Clustering

Amittai Aviram

General Idea

When you get results from a Google or MS Live search, they are ranked by importance, but not subdivided into word sense categories. A search for "sandbox play" (without quotation marks) will return results about

-- Buying a sandbox for kids to play in
-- Edward Albee's one-act play _The Sandbox_
-- Playing in a sandbox in a metaphorical sense, e.g., "playing in Google's sandbox."

I would like to group these results under semantic category headings. Here is a sketch of an application of clustering:

-- Find the union of all semantically-signficant word lists (excluding pronouns, articles, conjuctions, etc.) in all documents returned. This list has n words.
-- Get word counts for each respective document against the word list. This gives you an n-dimensional vector for each document.
-- Compute the cosine for each pair of document vectors. (Time in O(n^2).)
-- Find clusters of document vectors through a nearest-neighbor algorithm.
-- Classify each document by its cluster.

It is also possible to use an online thesaurus to provide ready-made semantic categories and then to find the smallest cosine between the word count vector of each document and the word count vectors of the respective thesaurus entries. An advantage here is that the results _seem_ likely to be more reliable. (This would have to be proven empirically.) A disadvantage is that this approach may be less versatile, since it relies on a relatively unchanging or slow-changing thesaurus as a set of reference points.

The computation of word lists, word count vectors, cosines, clusters, and cosines again is demanding, so I propose to distribute the work among several parallel processes. This part of the project would fulfill the requirements of a final project in Parallel Programming, which I am also taking, and where we have been encouraged to develop a project to use parallel programming to solve some problem in some outside area. So the implementation of a clustering algorithm for an NLP application and the parallel processing back end will constitute two distinct but connected projects, for Data Mining and Parallel Programming, respectively.