Diction (Synonyms and Word-Sense Disambiguation)

*A Statistical Model for Near-Synonym Choice*

Many CR questions on the SAT involve finding the closest meaning for a given word in context (see Appendix). In *A Statistical Model for Near-Synonym Choice*, the author attempts to construct a model for an “intelligent thesaurus” that can select “the best near-synonym that should be used in a particular context”. The corpus of texts used in training this algorithm is drawn from the web, and the results are evaluated by comparing the machine’s answers to SAT-style questions with human-crowdsourced solutions.

Several versions of the algorithm were developed, using supervised and unsupervised learning techniques, and the author was eventually able to achieve an accuracy of 75%. The optimal algorithm used a combination of PMI rankings, word/document counts, and the notion of *anticollocation* – eliminating synonym choices by removing word combinations a native English writer (as defined by the corpora) would *not* use. The ideal context window size was found to be small – between 3 and 5 words, and word counts were found to yield higher performance than document counts. Interestingly, this paper claims that the highest-performing supervised method was slightly better than the unsupervised method, but the difference was not statistically significant.

The work undertaken in this paper is extremely helpful in obtaining a baseline methodology for our own project, and it will be interesting to perform research based on a corpus of texts not derived from the web. The authors also acknowledge but do not address the issue of nuance in synonym generation – in the phrase “a terrible slip”, how does the machine know that the dominant word is “terrible” (a strong word), and not “slip” (a weak word)? This idea of ranking individual words within synonymic expressions is something that may have a significant impact on our research, where almost every question requires an understanding of several words in context.

*Statistical Sense Disambiguation with Relatively Small Corpora Using Dictionary Definitions*

This paper tackles also tackles the problem of word-sense disambiguation, but instead focuses on efficiently finding synonyms by combining co-occurrence data from small corpora with dictionary definitions. This method, the author argues, alleviates the problem of data sparseness, which arises when using a larger, web-based corpus (see Inkpen’s approach, above).

The dictionary-based word definitions require a large amount of pre-processing, including the construction of a “conceptual expansion” for each word into its overarching concept, and the creation of a “co-occurrence matrix” for all pairs of concepts (see original paper, Figure 1). Word ambiguities are then given a score by comparing the target word concept with the co-occurrence entries in its context using log-based Maximum Likelihood Estimation. The author ran several tests using this approach on structured examples, and achieved 77% accuracy on identifying the correct ambiguity from a choice of three in several small passages.

This approach may be a useful strategy when dealing with SAT-style problems, which primarily deal with well formed, grammatically correct sentences with a vocabulary usually devoid of slang or colloquialisms. However, the author noted a major flaw in the system where several words were used in a conceptual sense that strayed outside of the standard dictionary definition. This indicates that a hybrid approach, borrowing elements from both large web-based corpora and standard dictionary definitions, may be the optimal solution.

*Personalizing PageRank for Word Sense Disambiguation*

This paper takes an unsupervised approach to the task of disambiguation and attempts to solve the problem using a graph built from an LKB based on WordNet. The algorithm used is as follows:

1. For each word, *w* in a context window, extract the relevant subgraph for that word a predetermined list of concepts.
2. Perform a BFS between *w* and every other word in the subgraph, keeping track of the shortest paths found.
3. The disambiguation graph is the union of the vertices and edges of the shortest paths over all concepts.
4. Now perform the PageRank [PR] or personalized PageRank [PPR] algorithm over this new graph. From the paper, “The intuition behind this step is that the vertices representing the correct concepts will be more relevant than the rest of the possible concepts of the context words.”

The results of this algorithm were computed over the Senseval-2 and -3 datasets and achieved a success rate of approximately 58%. The optimal results were gained using PPR and building the initial graph from WordNet 1.7.

This method (unlike the previous two) is built to handle the problem of sequence-labeling, which means that it can disambiguate all words in a context, regardless of order. It can also easily be extended to other languages with the introduction of a new word network. This may not be a good fit for our project, since we only ever need to disambiguate a single word and are already provided with a list of synonyms from which to choose. Furthermore, since the SAT is designed to test a student’s understanding of less commonly-used word ambiguities, a corpus- or dictionary-based approach (Inkpen, Luk) may yield the best results.