Summary 16: A Neural Probabilistic Langauge Model

Susan Cherry

This paper attempts to deal with the problem of the curse of dimensionality in statistical language modeling. The curse of dimensionality comes from the fact that a word sequence on which the model will be tested is probably different from all the word sequences seen during training. The proposed model simultaneously learns a distributed representation for each world and the probability function for word sequences. The authors then run experiments on two text corpa using neural networks.

Their proposed solution to fighting the curse of dimensionality with Distributed Representations can be summarized in three main points: 1) Associate a word feature vector with each word in the vocabulary, 2) express the joint probability function of word sequences in terms of the feature vectors of the words in the sequence, and 3) learn simultaneously the word feature vectors and the parameters of that probability function. In the proposed model, similar words are expected to have a similar feature vector and the probability function is a smooth function of the feature values so a small change in features leads to a small change in probability. So the presence of one sentence increases the probability of its "neighbors."

Next the authors describe the neural model. The objective is to learn a good model $f(w_t, ...w_{t-n+1}$ that gives high out-of-sample likelihood. $f(w_t, ...w_{t-n+1}$ is decomposed into two parts: 1) A mapping C from any element of V to a real vector $C(i) \in R^m$ and 2) The probability function over words. Training is achieved by looking for θ that maximizes the training corpus penalized log-likelihood. In most of the experiments, the neural network has one hidden layer beyond the word features mapping.

Next the paper describes parallel implementation. Running the model on a parallel computer reduces computation time. The authors use data-parallel processing in which each processor works on a different subset of the data and parameter parallel processing in which they parallelize across the parameters because that's where the vast majority of computation takes place. On page 1145 the give details of the computation for processor i on example t.

Next they move to the experimental results. They perform experiments on the Brown Corpus and on text from the Associated Press News. N-Gram models are used as a benchmark agains the neural network. Their main result is that the neural network performs significantly better than the best of the n-grams. The difference in test perplexity is about 24% on the Brown and 8% on the AP news text. The paper concludes by discussing extension and future work. A variant of the neural network here can be interpreted as an energy minimization model, for example. Methods to speed-up training and recognition are needed and more ways to generalize should be explored.

In summary, the authors have presented a model that yields much better perplexity than the traditional method. They believe the reason for improvement is that each training sentence informs the model about a combinatorial number of other sentences. This work should lead to improvements in statistical language models brought by replacing "tables of conditional probabilities" by more compact and smoother representations that allow far more conditioning variables.