Mehryar Mohri
Speech Recognition
Courant Institute of Mathematical Sciences
Project
Due: December 17, 2010

The objective of this project is to design a spelling correction system based on weighted finite-state transducers.

Description

The idea consists of modeling spelling correction in a way similar to speech recognition as a search through a composition of weighted transducers, as already discussed in class.

Let G be a statistical n-gram language model trained on a large text corpus (you can use the homework data), L a transducer mapping each word of the vocabulary to its spelling, thus a sequence of letters, T an edit-distance transducer, and I a sequence of letters typed by the user. Then, the most likely word transcription for I given these components is obtained by finding the best path of

$$I \circ T \circ L \circ G \tag{1}$$

and projecting on the output to obtain the sequence of words.

The language model G can be obtained as in a previous homework assignment. You could select the n-gram order based on efficiency and accuracy considerations. Creating the transducer L should be straightforward. Thus, the two main tasks of this project are: (1) the definition and learning of the *edit-distance transducer*; (2) the transducer optimizations and combination for a successful and fast search.

Edit-Distance Transducer

To design the edit-distance transducer, you could of course start from the simple one already presented in the class lectures. But, how to determine the cost of each edit operation? One way to do that is to learn the edit costs from data. The data could be a list of English words and their common misspellings, which can be found under

`http://en.wikipedia.org/wiki/Wikipedia:List_of_common_misspellings` .

Now, each edit cost can be modeled as the negative log of some probability. You could then use the maximum-likelihood principle and the EM algorithm to determine these costs based on your training data. The topology of your transducer T needs not be the one-state machine for the standard edit-distance. You may want to model more complex mistypings or misspellings, e.g., the confusion in English of *ie* and *ei*, and learn their costs. You can also directly model the misspelling for some words, for example those downloaded, by including them in T and assigning a very low weight to the corresponding mapping.

Search

Depending on the size of the components, the composition and search could be quite slow without any optimization. You could adopt optimization techniques similar to those described in our lectures for speech recognition to tackle this problem.

In particular, as in the case of phonemes in speech, the number of letters in English is quite small compared to the number of words. Thus, one can expect determinization to produce similar beneficial effects. Creating an optimized version of the full misspelling transducer $T \circ L \circ G$ could perhaps be done and speed up search. Additionally, pruning and a Viterbi search could be necessarily if the search through the full transducer $I \circ T \circ L \circ G$ is impractical. Different strategies could be explored and adopted to make the search as efficient as possible.

Organization

You will be working on the project as part of a team. Each team of about 6 students will be divided in two sub-groups, one working on the edit-transducer problem, the other on search. The sub-groups are then combining their work to produce the full misspelling system. It could be useful to select one person as the overall architect to make sure that the pieces will work together. Each team should turn in a 2-4 page description of their project and present it on December 20 in class, at the usual time and location.