General Requirements - please read this before you start.
Homework is due back on the date listed above. No late homeworks! You should return:

- written answers to the questions in this HW
- the code you used to generate the answers.

Code: can be in C, C++, Perl, java, shell script ... but must run under Unix. DO NOT SEND ME CODE THAT RUNS UNDER WINDOWS ONLY! For each question, one program conforming to a specific command line and generating specific output should be handed in (described below). Your program should perform all necessary steps internally, no additional user intervention should be needed (e.g. you can use a shell script to connect several different program calls).

Submitting: When submitting your HW, please include all pieces of code and other files needed to re-generate your results. Tar everything up into an archive, gzip it, and name it <yourlastname>.hw1.tar.gz. Submit your HW by email to me (katrin@ee.washington.edu) no later than 11:59pm on the due date. Don’t forget to include a file (simple ascii is fine) with answers to the questions and name it answers.txt.

For this HW you can utilize the GIZA++ toolkit, located on Pongo in /usr/src/november05/giza++. Reading the documentation and developing an understanding of how these tools work is part of your homework!

Help/Questions: You are supposed to do your HW yourselves, i.e. do not use any help from other students, researchers, etc., do not use pre-existing tools/code other than the ones mentioned above. This is an honour policy. You can email me if you have questions about the tasks below, but please refrain from sending emails asking how to run a tool when the answer is in the manpage (this will not help your grade).

1 Introduction

This homework is about word alignment. You are supposed to train word alignment systems for English-French and Romanian-English. For English-French,
you are given a large training set of parallel data. For Romanian-English, you
will have only have a small development set of about 150 sentences. For both
English-French and Romanian-English you will need to run your aligners on
the test data. After you have submitted your homework I will compute
the alignment error rates for all systems against the answers on the test data.

2 Data

The data you need is in /home/katrin/Ling575/hw2 on the Pongo cluster.
French-English: directory with French-English data
- cleantext.en,fr.gz: gzipped parallel text (en = English, fr = French)
- dev.en,fr: parallel data for development (tuning parameters)
- dev.answers: the ground-truth word alignment for the development set
- test.en,fr: parallel test data

wa,eval,align.pl: program to compute alignment error rate

Romanian-English: directory with Romanian-English data
- test.[1-5].*: parallel development data
- test1-5.answers: ground-truth for development data
- test.[6-10].*: parallel test data

3 Tasks

1) Train IBM model 1 and perform word alignment. Report your alignment
error rate on the French development data. Pay some attention to preprocessing!
The training and development/test corpora are not necessarily in the same
format!

2) Train IBM model 4 and perform word alignment. Report your alignment
error rate on the French development data.

3) Try to improve on your model using anything you can find out about
French (e.g. morphology, frequent function words, multiword expressions, word
order...). You can either preprocess the French/English texts to improve your
alignment or change the alignment model itself. Re-align the development data
with your best model and report the alignment error rate.

4) For all three models, produce alignments on the test data in the following
format:
< sentencenumber > < wordindexL1 > < wordindexL2 >
where $L_1$ and $L_2$ are the first and second language, respectively. Word indices should start at 1, with the 0 index reserved for the empty word.

5) (Try to) produce alignments for the Romanian-English test data. You can use any information you can think of: cognates, information from French (e.g., using a French-Romanian dictionary if you can find one), word context statistics, etc. Produce an alignment in the above format.

As before, send me both the answers to the questions, as well as the code you used to produce your answers. Provide all files necessary to regenerate the results.