Machine Translation

Lab 4

We will experiment with the well-known alignment tool GIZA++. Download the archive, extract its content, and copy it to your account on the university's server. Enter from *putty* to your account (or from a similar program) and change directory until you are in the Giza directory. Run *make*, and then proceed with the steps below. The commands need to be run using *putty* in the directory Giza++-v2.

Part a

To familiarize yourselves with the tool, we will consider a very simple example. Make two files *eng* and *ro* with two lines each:

```
dog bites dog

dog barked
```

and

```
cainele musca cainele

cainele latra
```

To prepare the corpus for GIZA++ we have to compute some additional files.

Run first

```
./plain2snt.out ro eng
```

Take a look at the newly generated files and try to understand what they contain. An .vcb-file assigns a unique numerical identifier to each word. The second number, e.g., 3 in

```
2 dog 3
```

counts the number of occurrences of this word. In the .snt files, the words in the source files are replaced by identifiers.

We also need one more initial preparation

```
./snt2cooc.out eng.vcb ro.vcb eng_ro.snt > corp.cooc
```

We are not concerned with the content of this file but it is used in the next step.

Part b

We can then run the alignment program

```
./GIZA++ -S eng.vcb -T ro.vcb -C eng_ro.snt -CoocurrenceFile corp.cooc
```

This generates a series of files with a longish prefix. You may use a prefix of your choice by the option „-o", e.g.,

```
./GIZA++ -S eng.vcb -T ro.vcb -C eng_ro.snt -CoocurrenceFile corp.cooc -o experiment1
```

To see what is going on and what the files contain we may take a look at the screen output from the run (which we could dump to a file by)

```
./GIZA++ -S eng.vcb -T ro.vcb -C eng_ro.snt -CoocurrenceFile corp.cooc -o experiment1 > output1
```

We see that the program by default has run 5 iterations of IBM model1 followed by 5 iterations of model3 and 5 iterations of model4. The names of the produced files indicates

- whether it is an alignment file (a/A), a lexical probabilities file (t) etc.
- which IBM model has produced it
- and after how many iterations

For example „experiment1.t3.final" is the word probabilities after the program has finished model 3. Have a look at the file, and at „experiment1.a3.final" and try to understand it.

Part c

We may change the default values (which we see in output1). For example, to study the convergence behaviour of Model1 we may iterate it a hundred times by

```
./GIZA++ -S eng.vcb -T ro.vcb -C eng_ro.snt -CoocurrenceFile corp.cooc -o experiment2 -model1iterations 100
```

But this doesn't change which files are produced. How can we see the effect of the change? By asking the program to dump more intermediate results e.g.,

```
./GIZA++ -S eng.vcb -T ro.vcb -C eng_ro.snt -CoocurrenceFile corp.cooc -o experiment2 -model1iterations 100 -model1dumpfrequency 10
```

Repeat the experiment to see what results you get after 1, 2, 5, 25 and 100 iterations. (You may do this by running several experiments with different parameters.). Submit the values of the alignment table for the 5 situations, with your comments.

Compare the results to the output of model3, i.e. with the default settings (5 iterations of model1 followed by 5 iterations of model3).

Part d

Consider the A1 files after the first 5 runs of model1. The results are a little surprising, how? Then consider the result after 100 model1 iterations and after 5 model1 followed by 5 model 3 iterations. What do you see?

Part e

Use a slightly larger corpus (min. 3 sentences of at least 8-10 words each) to study the effects of alignments which are not one-to-one. First of all you should tokenize the texts and also lowercase them. Then run GIZA++ in both directions with the default settings. Consider the resulting alignments A3. Draw figures similar to the figure below for the three sentences.

	michael	a	dedus	că	va	rămâne	în	casă
michael	X							
assumes		X	X					
that				X				
he								
will					X			
stay						X		
in							X	
the								X
house								X

What to deliver?

Max 1 page, containing:

- Results from the experiments in part c.
- Answer the questions in part d.
- The figures in part e.