Build Moses on Ubuntu (64-bit) in VirtualBox and 186-server: recorded by Aaron

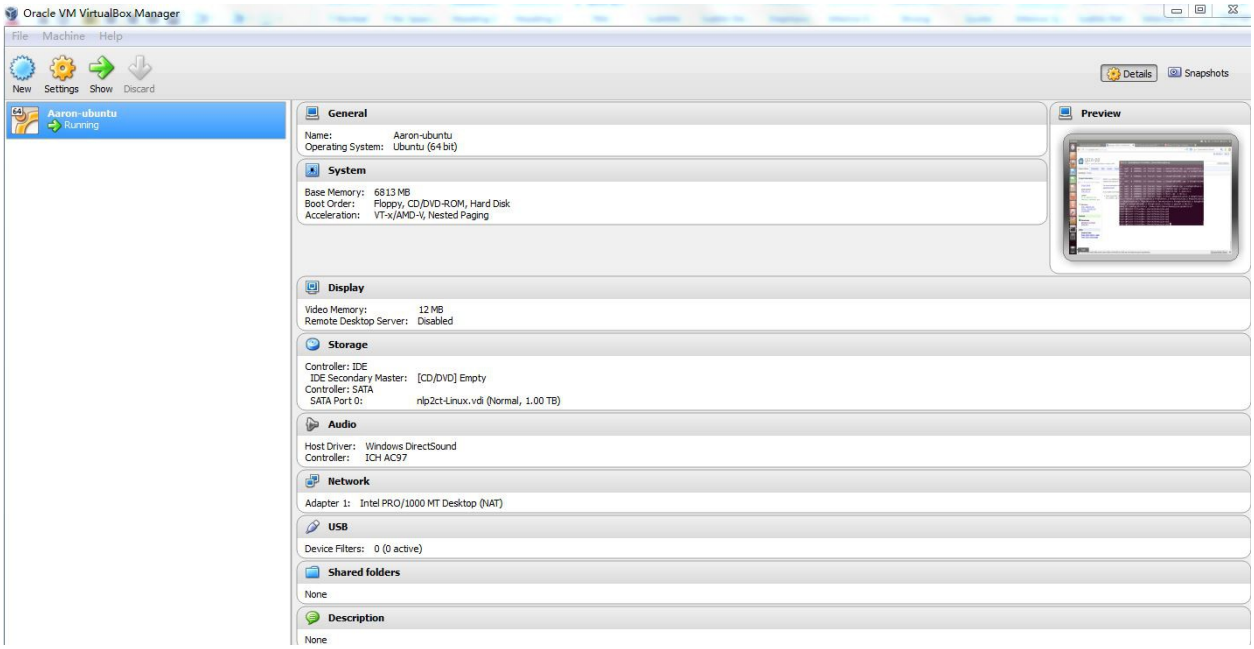
[http://www.linkedin.com/in/aaronhan]

This document introduces the record by Aaron when running Moses translation systems (http://www.statmt.org/moses/). If you want to get more information about the Moses please see the official website of Moses or Moses-manual.pdf (http://www.statmt.org/moses/manual/manual.pdf).

Contents

Install Moses MT system and run an example:

1. Download virtual box from https://www.virtualbox.org/wiki/Downloads
2. Install virtual box by double click the "VirtualBox-4.2.18-88781-Win.exe" file. [To make the Ubuntu fluent, take the following actions.
   A1.download and install the VirtualBox Extension Pack outside the VM:
   https://www.virtualbox.org/wiki/Downloads
   A2.Ubuntu->Devices -> Install guest additions…]
3. Press "New" button to guide your "driver" (I use nlp2ct-Linux.vdi) into the virtual box. During the selection, chose the Ubuntu64-bit. Setting the base memory around 7GB.



4. Press the "start" to lunch the Ubuntu system in the virtual box.
5. Download Giza++ (word alignment model) from https://code.google.com/p/giza-pp/downloads/list
6. Download IRSTLM (language model) from http://hlt.fbk.eu/en/irstlm
7. Download Moses (Statistical Machine Translation decoder) from http://www.statmt.org/moses/?n=Moses.Releases
8. Install the Giza++. If you store the Giza++ in the address Home/Aaron/Moses/giza-pp, open the "Terminal", get into this address (press "ls" to show the items under your current position; press "cd xx" to enter the xx file; press "cd .." to jump out current file), then press "make" command. If show "g++ command not found", type "g++", then type "sudo apt-get install g++" to install g++. After the install of g++, type "make" command to install Giza++ again.

```
g++ -Wall -W -DNDEBUG -O3 -funroll-loops -c KategProblemWBC.cpp -o KategProblemW
BC.o
g++ -Wall -W -DNDEBUG -O3 -funroll-loops -c KategProblem.cpp -o KategProblem.o
g++ -Wall -W -DNDEBUG -O3 -funroll-loops -c StatVar.cpp -o StatVar.o
g++ -Wall -W -DNDEBUG -O3 -funroll-loops -c general.cpp -o general.o
g++ -Wall -W -DNDEBUG -O3 -funroll-loops -c mkcls.cpp -o mkcls.o
g++ -Wall -W -DNDEBUG -O3 -funroll-loops -o mkcls GDAOptimization.o HCOptimizati
on.o Problem.o IterOptimization.o ProblemTest.o RRTOptimization.o MYOptimization
.o SAOptimization.o TAOptimization.o Optimization.o KategProblemTest.o KategProb
lemKBC.o KategProblemWBC.o KategProblem.o StatVar.o general.o mkcls.o
make[1]: Leaving directory `/home/nlp2ct/Aaron/Moses/giza-pp/mkcls-v2'
```

9. Install IRSTLM. Get into the location of the IRSTLM file. (install guideline http://sourceforge.net/apps/mediawiki/irstlm/index.php?title=Installation_Guidelines) Type “sh regenerate-makefiles.sh”. if shows “command not found aclocal failed” type “sudo apt-get install automake”, if show “libtoolize: no such file” type “sudo apt-get install libtool”. Then type “bash regenerate-makefiles.sh” again.
   Type “./configure –prefix=/Home/Aaron/Moses/irstlm-5.80.03” to generate and locate the “Makefile”

```
checking getopt.h usability... yes
checking getopt.h presence... yes
checking for getopt.h... yes
configure: creating ./config.status
config.status: creating Makefile
config.status: creating src/Makefile
config.status: creating scripts/Makefile
config.status: creating config.h
config.status: executing depfiles commands
config.status: executing libtool commands
configure: The software will be installed into /Home/Aaron/Moses/irstlm-5.80.03
nlp2ct@nlp2ct-VirtualBox:~/Aaron/Moses/irstlm-5.80.03$
```

   Type “make” for compilation. If show “zlib.h: No such file” type “sudo apt-get install zlib1g-dev” to install zlib. Type “make” again.

```
make[2]: Leaving directory `/home/nlp2ct/Aaron/Moses/irstlm-5.80.03/src'
Making all in scripts
make[2]: Entering directory `/home/nlp2ct/Aaron/Moses/irstlm-5.80.03/scripts'
make[2]: Nothing to be done for `all'.
make[2]: Leaving directory `/home/nlp2ct/Aaron/Moses/irstlm-5.80.03/scripts'
make[2]: Entering directory `/home/nlp2ct/Aaron/Moses/irstlm-5.80.03'
make[2]: Leaving directory `/home/nlp2ct/Aaron/Moses/irstlm-5.80.03'
make[1]: Leaving directory `/home/nlp2ct/Aaron/Moses/irstlm-5.80.03'
nlp2ct@nlp2ct-VirtualBox:~/Aaron/Moses/irstlm-5.80.03$
```

   Type “sudo make install” for installation.

```
make[2]: Leaving directory `/home/nlp2ct/Aaron/Moses/irstlm-5.80.03/scripts'
make[1]: Leaving directory `/home/nlp2ct/Aaron/Moses/irstlm-5.80.03/scripts'
make[1]: Entering directory `/home/nlp2ct/Aaron/Moses/irstlm-5.80.03'
make[2]: Entering directory `/home/nlp2ct/Aaron/Moses/irstlm-5.80.03'
make[2]: Nothing to be done for `install-exec-am'.
make[2]: Nothing to be done for `install-data-am'.
make[2]: Leaving directory `/home/nlp2ct/Aaron/Moses/irstlm-5.80.03'
make[1]: Leaving directory `/home/nlp2ct/Aaron/Moses/irstlm-5.80.03'
nlp2ct@nlp2ct-VirtualBox:~/Aaron/Moses/irstlm-5.80.03$
```

The IRSTLM library and commands are generated respectively under the address "/Home/Aaron/Moses/irstlm-5.80.03"

10. Install Boost (C++ libraries).
    Download it from (http://www.boost.org/users/history/version_1_52_0.html).
    cd boost_1_52_0
    ./bootstrap.sh –prefix=/Home/Aaron/Moses/boost_1_52_0

```
Building Boost.Build engine with toolset gcc... tools/build/v2/engine/bin.linuxx
86_64/b2
Detecting Python version... 2.7
Detecting Python root... /usr
Unicode/ICU support for Boost.Regex?... /usr
Generating Boost.Build configuration in project-config.jam...

Bootstrapping is done. To build, run:

    ./b2

To adjust configuration, edit 'project-config.jam'.
Further information:

   - Command line help:
     ./b2 --help

   - Getting started guide:
     http://www.boost.org/more/getting_started/unix-variants.html

   - Boost.Build documentation:
     http://www.boost.org/boost-build2/doc/html/index.html

nlp2ct@nlp2ct-VirtualBox:~/Aaron/Moses/boost_1_52_0$
```

Type "./b2" to install boost.

```
gcc.compile.c++ bin.v2/libs/wave/build/gcc-4.6/release/link-static/threading-multi/wave
_config_constant.o
common.mkdir bin.v2/libs/wave/build/gcc-4.6/release/link-static/threading-multi/cpplexe
r
common.mkdir bin.v2/libs/wave/build/gcc-4.6/release/link-static/threading-multi/cpplexe
r/re2clex
gcc.compile.c++ bin.v2/libs/wave/build/gcc-4.6/release/link-static/threading-multi/cppl
exer/re2clex/aq.o
gcc.compile.c++ bin.v2/libs/wave/build/gcc-4.6/release/link-static/threading-multi/cppl
exer/re2clex/cpp_re.o
gcc.archive bin.v2/libs/wave/build/gcc-4.6/release/link-static/threading-multi/libboost
_wave.a
common.copy stage/lib/libboost_wave.a
...failed updating 2 targets...
...skipped 6 targets...
...updated 971 targets...
nlp2ct@nlp2ct-VirtualBox:~/Aaron/Moses/boost_1_52_0$
```

11. Install other dependencies (gcc, zlib, bzip2).

    Type “sudo apt-get install build-essential libz-dev libbz2-dev” to install.

```
Setting up libtimedate-perl (1.2000-1) ...
Setting up libdpkg-perl (1.16.1.2ubuntu7.1) ...
Setting up dpkg-dev (1.16.1.2ubuntu7.1) ...
Setting up build-essential (11.5ubuntu2.1) ...
Setting up fakeroot (1.18.2-1) ...
update-alternatives: using /usr/bin/fakeroot-sysv to provide /usr/bin/fakeroot (fakeroo
t) in auto mode.
Setting up libalgorithm-diff-perl (1.19.02-2) ...
Setting up libalgorithm-diff-xs-perl (0.04-2build2) ...
Setting up libalgorithm-merge-perl (0.08-2) ...
Setting up libbz2-dev (1.0.6-1) ...
nlp2ct@nlp2ct-VirtualBox:~/Aaron/Moses$
```

12. Install Moses.

    Type “git clone git://github.com/moses-smt/mosesdecoder.git”.

```
Cloning into 'mosesdecoder'...
remote: Counting objects: 90550, done.
remote: Compressing objects: 100% (23416/23416), done.
Receiving objects: 100% (90550/90550), 111.65 MiB | 1.01 MiB/s, done.
remote: Total 90550 (delta 68741), reused 86391 (delta 64660)
Resolving deltas: 100% (68741/68741), done.
nlp2ct@nlp2ct-VirtualBox:~/Aaron/Moses$
```

    Compile Moses. To examine the options you want, type “cd ~/mosesdecoder”, “./bjam --help”. There will be automatic updates.

```
*** No errors detected
gcc.compile.c++ mert/bin/gcc-4.6/release/debug-symbols-on/link-static/threading-multi/VocabularyTest.o
gcc.link mert/bin/gcc-4.6/release/debug-symbols-on/link-static/threading-multi/vocabulary_test
testing.unit-test mert/bin/gcc-4.6/release/debug-symbols-on/link-static/threading-multi/vocabulary_test.passed
Running 2 test cases...

*** No errors detected
common.copy mert/mert
common.copy mert/extractor
common.copy mert/evaluator
common.copy mert/pro
common.copy mert/kbmira
common.copy mert/sentence-bleu
...updated 746 targets...
SUCCESS
nlp2ct@nlp2ct-VirtualBox:~/Aaron/Moses/mosesdecoder$
```

13. Run Moses with an example.
    Type the following commands to run the example:
    cd ~/mosesdecoder
    wget http://www.statmt.org/moses/download/sample-models.tgz
    tar xzzf sample-models.tgz
    cd sample-models
    ~/Aaron/Moses/mosesdecoder/bin/moses –f phrase-model/moses.ini < phrase-model/in > out

```
Defined parameters (per moses.ini or switch):
        config: phrase-model/moses.ini
        feature: KENLM name=LM factor=0 order=3 num-features=1 path=lm/europarl.srilm.gz Distortion WordPenalty UnknownWordPenalty PhraseDictionaryMemory input-
factor=0 output-factor=0 path=phrase-model/phrase-table num-features=1 table-limit=10
        input-factors: 0
        mapping: T 0
        n-best-list: nbest.txt 100
        weight: WordPenalty0= 0 LM= 1 Distortion0= 1 PhraseDictionaryMemory0= 1
/home/nlp2ct/Aaron/Moses/mosesdecoder/bin
line=KENLM name=LM factor=0 order=3 num-features=1 path=lm/europarl.srilm.gz
FeatureFunction: LM start: 0 end: 0
Loading the LM will be faster if you build a binary file.
Reading lm/europarl.srilm.gz
----5---10---15---20---25---30---35---40---45---50---55---60---65---70---75---80---85---90---95--100
**The ARPA file is missing <unk>.  Substituting log10 probability -100.000.
****************************************************************************************************
line=Distortion
FeatureFunction: Distortion0 start: 1 end: 1
line=WordPenalty
FeatureFunction: WordPenalty0 start: 2 end: 2
line=UnknownWordPenalty
FeatureFunction: UnknownWordPenalty0 start: 3 end: 3
line=PhraseDictionaryMemory input-factor=0 output-factor=0 path=phrase-model/phrase-table num-features=1 table-limit=10
FeatureFunction: PhraseDictionaryMemory0 start: 4 end: 4
Start loading text SCFG phrase table. Moses  format : [1.000] seconds
Reading phrase-model/phrase-table
----5---10---15---20---25---30---35---40---45---50---55---60---65---70---75---80---85---90---95--100
****************************************************************************************************
IO from STDOUT/STDIN
Created input-output object : [1.000] seconds
Translating line 0  in thread id 140710001190656
Translating: das ist ein kleines haus
Line 0: Collecting options took 0.000 seconds
Line 0: Search took 0.000 seconds
BEST TRANSLATION: this is a small house [11111]  [total=-28.923] core=(-27.091,0.000,-5.000,0.000,-1.833)
Line 0: Translation took 0.000 seconds total
Translating line 1  in thread id 140710001190656
Translating: das ist ein kleines haus
Line 1: Collecting options took 0.000 seconds
Line 1: Search took 0.000 seconds
BEST TRANSLATION: this is a small house [11111]  [total=-28.923] core=(-27.091,0.000,-5.000,0.000,-1.833)
Line 1: Translation took 0.000 seconds total
Name:moses      VmPeak:197672 kB        VmRSS:32300 kB  RSSMax:32544 kB user:0.648      sys:0.028       CPU:0.676       real:0.783
nlp2ct@nlp2ct-VirtualBox:~/Aaron/Moses/mosesdecoder/sample-models$
```

The translation results of source sentence “das ist ein kleines haus” will be shown in the file of out as “it is a small house”. Succeed!

To test the Chart decoder, type the following command, the translation result will be shown in out.stt file:

~/Aaron/Moses/mosesdecoder/bin/moses_chart –f string-to-tree/moses.ini < string-to-tree/in > out.stt

```
Reading string-to-tree/rule-table
----5---10---15---20---25---30---35---40---45---50---55---60---65---70---75---80
---85---90---95--100
********************************************************************************
********************
max-chart-span: 20
IO from STDOUT/STDIN
Created input-output object : [0.000] seconds
Translating: <s> das ist ein kleines haus </s>  ||| [0,0]=X (1) [0,1]=X (1) [0,2
]=X (1) [0,3]=X (1) [0,4]=X (1) [0,5]=X (1) [0,6]=X (1) [1,1]=X (1) [1,2]=X (1)
[1,3]=X (1) [1,4]=X (1) [1,5]=X (1) [1,6]=X (1) [2,2]=X (1) [2,3]=X (1) [2,4]=X
(1) [2,5]=X (1) [2,6]=X (1) [3,3]=X (1) [3,4]=X (1) [3,5]=X (1) [3,6]=X (1) [4,4
]=X (1) [4,5]=X (1) [4,6]=X (1) [5,5]=X (1) [5,6]=X (1) [6,6]=X (1)

  0   1   2   3   4   5   6
  0   3   2   2   2   1   0
    0   0   0   0   0   0
      0   0   0   3   0
        0   0   4   0
          0   4   0
            0   0
              1
BEST TRANSLATION: 41 TOP  -> <s> S </s> :1-1 : c=-3.206 core=(-6.413,-2.000,0.00
0,0.000)  [0..6] 20 [total=-15.501] core=(-27.091,-7.000,0.000,-3.912)
Translation took 0.000 seconds
End. : [0.000] seconds
Name:moses_chart        VmPeak:190436 kB        VmRSS:31856 kB  RSSMax:32312 kBu
ser:0.596       sys:0.048       CPU:0.644       real:0.795
nlp2ct@nlp2ct-VirtualBox:~/Aaron/Moses/mosesdecoder/sample-models$
```

To test the tree-to-tree demo, type the following command, the translation result will be shown in out.ttt file:
~/Aaron/Moses/mosesdecoder/bin/moses_chart –f tree-to-tree/moses.ini < tree-to-tree/in.xml > out.ttt

```
 [3,8]=X (1) [3,9]=X (1) [3,10]=X (1) [3,11]=X (1) [3,12]=X (1) [3,13]=X (1) [4,4]=X (
1) [4,5]=X (1) [4,6]=X (1) [4,7]=X (1) [4,8]=X (1) [4,9]=X (1) [4,10]=X (1) [4,11]=X (
1) [4,12]=X (1) [4,13]=X (1) [5,5]=X (1) [5,6]=X (1) [5,7]=X (1) [5,8]=X (1) [5,9]=X (
1) [5,10]=X (1) [5,11]=X (1) [5,12]=X (1) [5,13]=X (1) [6,6]=X (1) [6,7]=X (1) [6,7]=N
S (1) [6,8]=X (1) [6,9]=X (1) [6,10]=X (1) [6,11]=X (1) [6,12]=X (1) [6,13]=X (1) [7,7
]=X (1) [7,7]=NS (1) [7,8]=X (1) [7,9]=X (1) [7,10]=X (1) [7,11]=X (1) [7,12]=X (1) [7
,13]=X (1) [8,8]=X (1) [8,9]=X (1) [8,10]=X (1) [8,11]=X (1) [8,12]=X (1) [8,12]=OS (1
) [8,13]=X (1) [9,9]=X (1) [9,10]=X (1) [9,11]=X (1) [9,12]=X (1) [9,13]=X (1) [10,10]
=X (1) [10,11]=X (1) [10,12]=X (1) [10,13]=X (1) [11,11]=X (1) [11,12]=X (1) [11,13]=X
 (1) [12,12]=X (1) [12,13]=X (1) [13,13]=X (1)

  0   1   2   3   4   5   6   7   8   9  10  11  12  13
  1   1   1   1   1   1   1   1   1   1   1   1   1   0
    1   0   1   0   0   0   1   0   0   0   0   0   0
      1   0   0   0   0   0   0   0   0   0   0   0
        1   0   0   0   0   0   0   0   0   0   0
          1   0   0   1   0   0   0   0   1   0
            1   0   1   0   0   0   0   0   0
              1   0   0   0   0   0   0   0
                1   0   0   0   0   0   0
                  1   0   0   0   0   0
                    1   0   0   0   0
                      1   0   0   0
                        1   0   0
                          1   0
                            1
BEST TRANSLATION: 35 S  -> S </s> :0-0 : c=0.000 core=(-1.000,0.000,0.000,0.000,0.000,
0.000,0.000,0.000)  [0..13] 33 [total=-9.680] core=(-9.000,0.000,-5.839,-9.522,0.000,-
6.000,4.999,3.000)
Translation took 0.000 seconds
End. : [0.000] seconds
Name:moses_chart        VmPeak:160972 kB        VmRSS:2720 kB   RSSMax:3520 kB  user:0
.008    sys:0.008       CPU:0.016       real:0.075
nlp2ct@nlp2ct-VirtualBox:~/Aaron/Moses/mosesdecoder/sample-models$
```

==========

Another way to install GIZA++:

Type the following commands to install GIZA++.
wget http://giza-pp.googlecode.com/files/giza-pp-v1.0.7.tar.gz
tar xzvf giza-pp-v1.0.7.tar.gz
cd giza-pp
make

above commands will create the binaries ~/giza-pp/GIZA++-v2/GIZA++, ~/giza-pp/GIZA++-v2/snt2cooc.out and ~/giza-pp/mkcls-v2/mkcls. To automatically copy these files to somewhere that Moses can find, type the following commands:
cd ~/mosesdecoder
mkdir tools

cp ~/Aaron/Moses/giza-pp/GIZA++-v2/GIZA++ ~/Aaron/Moses/giza-pp/GIZA++-v2/snt2cooc.out ~/Aaron/Moses/giza-pp/mkcls-v2/mkcls tools

above commands will copy the three binaries “GIZA++, snt2cooc.out, and mkcls” into the directory “~/mosesdecoder/tools”.

When you come to run the training, you need to tell the training script where GIZA++ was installed using the –external-bin-dir argument as below:
train-model.perl –external-bin-dir $HOME/mosesdecoder/tools

==========

Another way to install IRSTLM:

Download the latest version of IRSTLM.
tar zxvf irstlm-5.80.03.tgz
cd irstlm-5.80.03
./regenerate-makefiles.sh
./configure –prefix=$Home/Aaron/Moses/irstlm-5.80.03
sudo make install

==========

Corpus preparation:

mkdir corpus
cd corpus
wget http://www.statmt.org/wmt13/training-parallel-nc-v8.tgz
tar zxvf training-parallel-nc-v8.tgz

corpus tokenization (add spaces between words and punctuations):
type the following command for corpus tokenization:
~/Aaron/Moses/mosesdecoder/scripts/tokenizer/tokenizer.perl -l en < ~/Aaron/corpus/training/news-commentary-v8.fr-en.en > ~/Aaron/corpus/news-commentary-v8.fr-en.tok.en

~/Aaron/Moses/mosesdecoder/scripts/tokenizer/tokenizer.perl -l fr < ~/Aaron/corpus/training/news-commentary-v8.fr-en.fr > ~/Aaron/corpus/news-commentary-v8.fr-en.tok.fr

The tokenized files will be generated in the rectory file “~/Aaron/corpus”

Truecasing (to reduce data sparsity, convert the initial words of the sentence into the most probable casing):

To get the truecasing training models that are the statistics data extracted from text, type the following commands:
~/Aaron/Moses/mosesdecoder/scripts/recaser/train-truecaser.perl --model ~/Aaron/corpus/trucase-model.en --corpus ~/Aaron/corpus/news-commentary-v8.fr-en.tok.en

~/Aaron/Moses/mosesdecoder/scripts/recaser/train-truecaser.perl --model ~/Aaron/corpus/trucase-model.fr --corpus ~/Aaron/corpus/news-commentary-v8.fr-en.tok.fr

Above commands will generate the model files “truecase-model.en” and “truecase-model.fr” under the directory “/Aaron/corpus”

Using the extracted truecasing training models to perform the truecase function as below commands:
~/Aaron/Moses/mosesdecoder/scripts/recaser/truecase.perl --model ~/Aaron/corpus/trucase-model.en < ~/Aaron/corpus/news-commentary-v8.fr-en.tok.en > ~/Aaron/corpus/news-commentary-v8.fr-en.true.en

~/Aaron/Moses/mosesdecoder/scripts/recaser/truecase.perl --model ~/Aaron/corpus/trucase-model.fr < ~/Aaron/corpus/news-commentary-v8.fr-en.tok.fr > ~/Aaron/corpus/news-commentary-v8.fr-en.true.fr

Above commands will generate the files “news-commentary-v8.fr-en.true.ed” and “news-commentary-v8.fr-en.true.fr” under directory “Aaron/corpus”

Cleaning (to remove the mis-aligned sentencs, long sentences and empty sentences, which may cause problems with the training pipeline)

Type the following command to delete the sentences whose length is larger than 80:
~/Aaron/Moses/mosesdecoder/scripts/training/clean-corpus-n.perl ~/Aaron/corpus/news-commentary-v8.fr-en.true fr en ~/Aaron/corpus/news-commentary-v8.fr-en.clean 1 80

```
nlp2ct@nlp2ct-VirtualBox:~/Aaron/Moses/mosesdecoder$ ~/Aaron/Moses/mosesdecoder/script
s/training/clean-corpus-n.perl ~/Aaron/corpus/news-commentary-v8.fr-en.true fr en ~/Aa
ron/corpus/news-commentary-v8.fr-en.clean 1 80
clean-corpus.perl: processing /home/nlp2ct/Aaron/corpus/news-commentary-v8.fr-en.true.
fr & .en to /home/nlp2ct/Aaron/corpus/news-commentary-v8.fr-en.clean, cutoff 1-80
.........(100000).....
Input sentences: 157168  Output sentences:  155362
nlp2ct@nlp2ct-VirtualBox:~/Aaron/Moses/mosesdecoder$
```

The files “news-commentary-v8.fr-en.clean.en” and “news-commentary-v8.fr-en.clean.fr” will be generated in the directory “~/Aaron/corpus”.

==========

Language model training (built on the target language to ensure fluent output translation):

cd ~/Aaron
mkdir lm
cd lm
~/Aaron/Moses/irstlm-5.80.03/scripts/add-start-end.sh < ~/Aaron/corpus/news-commentary-v8.fr-en.true.en > news-commentary-v8.fr-en.sb.en

The above commends will generate the file “news-commentary-v8.fr-en.sb.en” in the directory “Aaron/lm”. In this file, each sentence is added with the start and end symbol “<s> </s>”.

Type the following command to generate the language model of English.
export IRSTLM=$Home/Aaron/Moses/irstlm-5.80.03; ~/Aaron/Moses/irstlm-5.80.03/scripts/build-lm.sh –i news-commentary-v8.fr-en.sb.en –t ./tmp –p –s improved-kneser-ney –o news-commentary-v8.fr-en.lm.en

```
Temporary directory ./tmp does not exist
creating ./tmp
Extracting dictionary from training corpus
Splitting dictionary into 3 lists
Extracting n-gram statistics for each word list
Important: dictionary must be ordered according to order of appearance of words in dat
a
used to generate n-gram blocks,  so that sub language model blocks results ordered too
dict.000
dict.001
dict.002
$bin/ngt -i="$inpfile" -n=$order -gooout=y -o="$gzip -c > $tmpdir/ngram.${sdict}.gz" -
fd="$tmpdir/$sdict" $dictionary -iknstat="$tmpdir/ikn.stat.$sdict" >> $logfile 2>&1
Estimating language models for each word list
dict.000
dict.001
dict.002
$scr/build-sublm.pl $verbose $prune $smoothing "cat $tmpdir/ikn.stat.dict.*" --size $o
rder --ngrams "$gunzip -c $tmpdir/ngram.${sdict}.gz" -sublm $tmpdir/lm.$sdict >> $logf
ile 2>&1
Merging language models into news-commentary-v8.fr-en.lm.en
Cleaning temporary directory ./tmp
Removing temporary directory ./tmp
nlp2ct@nlp2ct-VirtualBox:~/Aaron/lm$
```

Above commend generate the file “news-commentary-v8.fr-en.lm.en.gz” in the directory “Aaron/lm”.

Type the following command to generate the file “news-commentary-v8.fr-en.arpa.en” in the directory “Aaron/lm”

~/Aaron/Moses/irstlm-5.80.03/src/compile-lm –text news-commentary-v8.fr-en.lm.en.gz news-commentary-v8.fr-en.arpa.en

```
inpfile: news-commentary-v8.fr-en.lm.en.gz
outfile: news-commentary-v8.fr-en.arpa.en
loading up to the LM level 1000 (if any)
dub: 10000000
Language Model Type of news-commentary-v8.fr-en.lm.en.gz is 1
Language Model Type is 1
iARPA
loadtxt_ram()
1-grams: reading 62504 entries
done level 1
2-grams: reading 906385 entries
done level 2
3-grams: reading 378679 entries
done level 3
done
OOV code is 62503
OOV code is 62503
Saving in txt format to news-commentary-v8.fr-en.arpa.en
savetxt: news-commentary-v8.fr-en.arpa.en
save: 62504 1-grams
save: 906385 2-grams
save: 378679 3-grams
done
nlp2ct@nlp2ct-VirtualBox:~/Aaron/lm$
```

To make the faster loading, type the following command to binary the arpa.en file "~/Aaron/Moses/mosesdecoder/bin/build_binary news-commentary-v8.fr-en.arpa.en news-commentary-v8.fr-en.blm.en"

To check the language model, type the following command: "$ echo "is this an English sentence ?" | ~/Aaron/Moses/mosesdecoder/bin/query news-commentary-v8.fr-en.arpa.en". it will show the following result.

```
Loading the LM will be faster if you build a binary file.
Reading news-commentary-v8.fr-en.arpa.en
----5---10---15---20---25---30---35---40---45---50---55---60---65---70---75---80---85--
-90---95--100
*********$: command not found
**********************************************************************************
****
Loading statistics:
Name:query      VmPeak:47816 kB VmRSS:31108 kB  RSSMax:31940 kB user:0.452028   sys:0.0
12      CPU:0.464028    real:0.597656
After queries:
Name:query      VmPeak:47816 kB VmRSS:31108 kB  RSSMax:31940 kB user:0.452028   sys:0.0
12      CPU:0.464028    real:0.601562
Total time including destruction:
Name:query      VmPeak:47816 kB VmRSS:1724 kB   RSSMax:31940 kB user:0.452028   sys:0.0
12      CPU:0.464028    real:0.601562
nlp2ct@nlp2ct-VirtualBox:~/Aaron/lm$
```

If using the following command: “$ echo “is this an English sentence ?” | ~/Aaron/Moses/mosesdecoder/bin/query news-commentary-v8.fr-en.blm.en”. it will show the following result:

==========

Training translation system (run word alignment, phrase extraction and scoring, create lexicalized reordering tables and create Moses configuration file):

Type the following command to train the translation system using language model:

cd ~/Aaron
mkdir working
cd working
~/Aaron/Moses/mosesdecoder/scripts/training/train-model.perl -root-dir train -corpus ~/Aaron/corpus/news-commentary-v8.fr-en.clean -f fr -e en -alignment grow-diag-final-and -reordering msd-bidirectional-fe -lm 0:3:$HOME/Aaron/lm/news-commentary-v8.fr-en.blm.en:8 -external-bin-dir ~/Aaron/Moses/mosesdecoder/tools >& training.out &

```
 ~/Aaron/Moses/mosesdecoder/scripts/training/train-model.perl -root-dir train -corpus ~/Aaron/
corpus/news-commentary-v8.fr-en.clean -f fr -e en -alignment grow-diag-final-and -reordering msd-
bidirectional-fe -lm 0:3:$HOME/Aaron/lm/news-commentary-v8.fr-en.blm.en:8 -external-bin-dir ~/
Aaron/Moses/mosesdecoder/tools >& training.out &
```

The running details are written into the ~/working/training.out file step by step as below:

```
Using SCRIPTS_ROOTDIR: /home/nlp2ct/Aaron/Moses/mosesdecoder/scripts
Using single-thread GIZA
(1) preparing corpus @ Fri Oct 25 19:30:19 CST 2013
Executing: mkdir -p /home/nlp2ct/Aaron/working/train/corpus
(1.0) selecting factors @ Fri Oct 25 19:30:19 CST 2013
(1.1) running mkcls  @ Fri Oct 25 19:30:19 CST 2013
/home/nlp2ct/Aaron/Moses/mosesdecoder/tools/mkcls -c50 -n2 -p/home/nlp2ct/Aaron/corpus/news-commentary-
v8.fr-en.clean.fr -V/home/nlp2ct/Aaron/working/train/corpus/fr.vcb.classes opt
Executing: /home/nlp2ct/Aaron/Moses/mosesdecoder/tools/mkcls -c50 -n2 -p/home/nlp2ct/Aaron/corpus/news-
commentary-v8.fr-en.clean.fr -V/home/nlp2ct/Aaron/working/train/corpus/fr.vcb.classes opt

***** 2 runs. (algorithm:TA)*****
;KategProblem:cats: 50   words: 71168

start-costs: MEAN: 7.50606e+07 (7.50496e+07-7.50716e+07)  SIGMA:10981
  end-costs: MEAN: 6.96206e+07 (6.9613e+07-6.96283e+07)  SIGMA:7614.14
   start-pp: MEAN: 656.723 (655.266-658.18)  SIGMA:1.45712
     end-pp: MEAN: 218.785 (218.449-219.122)  SIGMA:0.336596
 iterations: MEAN: 1.92174e+06 (1.88842e+06-1.95504e+06)  SIGMA:33310
       time: MEAN: 117.479 (114.747-120.212)  SIGMA:2.73217
(1.1) running mkcls  @ Fri Oct 25 19:34:46 CST 2013
/home/nlp2ct/Aaron/Moses/mosesdecoder/tools/mkcls -c50 -n2 -p/home/nlp2ct/Aaron/corpus/news-commentary-
v8.fr-en.clean.en -V/home/nlp2ct/Aaron/working/train/corpus/en.vcb.classes opt
```

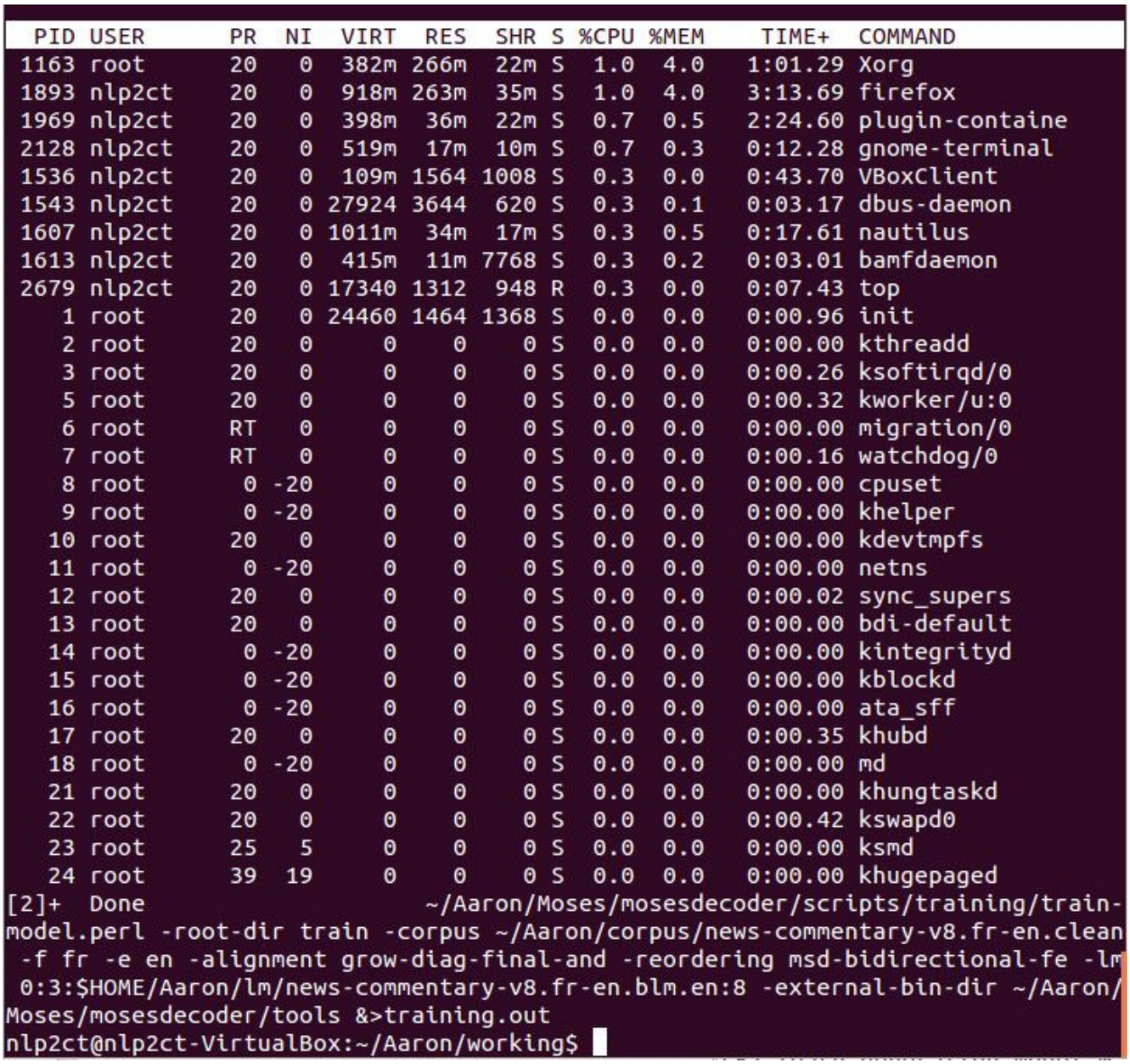
If you want to see the running thread, type command “top” to show the running details in the window as below:

```
  PID USER      PR  NI  VIRT  RES  SHR S %CPU %MEM    TIME+  COMMAND
 2610 nlp2ct    20   0  381m 368m 2028 R 94.1  5.6  28:22.37 GIZA++
 1893 nlp2ct    20   0  905m 280m  41m S  2.0  4.2   1:26.69 firefox
 1969 nlp2ct    20   0  398m  34m  22m S  1.7  0.5   0:47.25 plugin-containe
 1163 root      20   0  374m 258m  22m S  0.7  3.9   0:40.56 Xorg
 1536 nlp2ct    20   0  109m 1572 1016 S  0.7  0.0   0:14.23 VBoxClient
 2128 nlp2ct    20   0  519m  18m  11m S  0.3  0.3   0:06.06 gnome-terminal
    1 root      20   0 24460 2356 1368 S  0.0  0.0   0:00.96 init
    2 root      20   0     0    0    0 S  0.0  0.0   0:00.00 kthreadd
    3 root      20   0     0    0    0 S  0.0  0.0   0:00.12 ksoftirqd/0
```

Type command “ctrl+c” to shut down the current running (top) shown in the window as below [type "ctrl+z" to pause the run; type "jobs" show the running; type "bg + job-num" to put the detail background; type "fg + job-num" to show runing thread (bring to fore-ground)]:

```
   17 root      20   0     0    0    0 S  0.0  0.0   0:00.33 khubd
   18 root       0 -20     0    0    0 S  0.0  0.0   0:00.00 md
   21 root      20   0     0    0    0 S  0.0  0.0   0:00.00 khungtaskd
   22 root      20   0     0    0    0 S  0.0  0.0   0:00.00 kswapd0
   23 root      25   5     0    0    0 S  0.0  0.0   0:00.00 ksmd
   24 root      39  19     0    0    0 S  0.0  0.0   0:00.00 khugepaged
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$
```

Around 2 hours and 10 minutes later, the translation training stage will be finished. By command “ctrl+c”, the following figure show the cmd window content:



```
  PID USER      PR  NI  VIRT  RES  SHR S %CPU %MEM    TIME+  COMMAND
 1163 root      20   0  382m 266m  22m S  1.0  4.0   1:01.29 Xorg
 1893 nlp2ct    20   0  918m 263m  35m S  1.0  4.0   3:13.69 firefox
 1969 nlp2ct    20   0  398m  36m  22m S  0.7  0.5   2:24.60 plugin-containe
 2128 nlp2ct    20   0  519m  17m  10m S  0.7  0.3   0:12.28 gnome-terminal
 1536 nlp2ct    20   0  109m 1564 1008 S  0.3  0.0   0:43.70 VBoxClient
 1543 nlp2ct    20   0 27924 3644  620 S  0.3  0.1   0:03.17 dbus-daemon
 1607 nlp2ct    20   0 1011m  34m  17m S  0.3  0.5   0:17.61 nautilus
 1613 nlp2ct    20   0  415m  11m 7768 S  0.3  0.2   0:03.01 bamfdaemon
 2679 nlp2ct    20   0 17340 1312  948 R  0.3  0.0   0:07.43 top
    1 root      20   0 24460 1464 1368 S  0.0  0.0   0:00.96 init
    2 root      20   0     0    0    0 S  0.0  0.0   0:00.00 kthreadd
    3 root      20   0     0    0    0 S  0.0  0.0   0:00.26 ksoftirqd/0
    5 root      20   0     0    0    0 S  0.0  0.0   0:00.32 kworker/u:0
    6 root      RT   0     0    0    0 S  0.0  0.0   0:00.00 migration/0
    7 root      RT   0     0    0    0 S  0.0  0.0   0:00.16 watchdog/0
    8 root       0 -20     0    0    0 S  0.0  0.0   0:00.00 cpuset
    9 root       0 -20     0    0    0 S  0.0  0.0   0:00.00 khelper
   10 root      20   0     0    0    0 S  0.0  0.0   0:00.00 kdevtmpfs
   11 root       0 -20     0    0    0 S  0.0  0.0   0:00.00 netns
   12 root      20   0     0    0    0 S  0.0  0.0   0:00.02 sync_supers
   13 root      20   0     0    0    0 S  0.0  0.0   0:00.00 bdi-default
   14 root       0 -20     0    0    0 S  0.0  0.0   0:00.00 kintegrityd
   15 root       0 -20     0    0    0 S  0.0  0.0   0:00.00 kblockd
   16 root       0 -20     0    0    0 S  0.0  0.0   0:00.00 ata_sff
   17 root      20   0     0    0    0 S  0.0  0.0   0:00.35 khubd
   18 root       0 -20     0    0    0 S  0.0  0.0   0:00.00 md
   21 root      20   0     0    0    0 S  0.0  0.0   0:00.00 khungtaskd
   22 root      20   0     0    0    0 S  0.0  0.0   0:00.42 kswapd0
   23 root      25   5     0    0    0 S  0.0  0.0   0:00.00 ksmd
   24 root      39  19     0    0    0 S  0.0  0.0   0:00.00 khugepaged
[2]+  Done                    ~/Aaron/Moses/mosesdecoder/scripts/training/train-
model.perl -root-dir train -corpus ~/Aaron/corpus/news-commentary-v8.fr-en.clean
 -f fr -e en -alignment grow-diag-final-and -reordering msd-bidirectional-fe -lm
 0:3:$HOME/Aaron/lm/news-commentary-v8.fr-en.blm.en:8 -external-bin-dir ~/Aaron/
Moses/mosesdecoder/tools &>training.out
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$
```

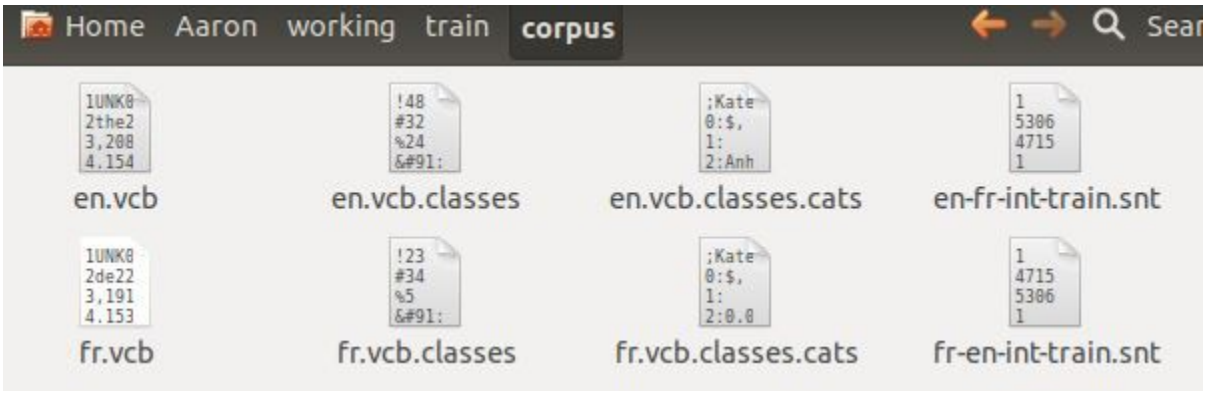
The following content is shown in the end of the file “~/working/training.out”:

```
Finished Fri Oct 25 21:39:52 2013
(6.6) consolidating the two halves @ Fri Oct 25 21:39:52 CST 2013
Executing: /home/nlp2ct/Aaron/Moses/mosesdecoder/scripts/../bin/consolidate /home/nlp2ct/Aaron/
working/train/model/phrase-table.half.f2e.gz /home/nlp2ct/Aaron/working/train/model/phrase-
table.half.e2f.gz /dev/stdout | gzip -c > /home/nlp2ct/Aaron/working/train/model/phrase-table.gz
Consolidate v2.0 written by Philipp Koehn
consolidating direct and indirect rule tables
.....
rm -f /home/nlp2ct/Aaron/working/train/model/phrase-table.half.*
(7) learn reordering model @ Fri Oct 25 21:42:17 CST 2013
(7.1) [no factors] learn reordering model @ Fri Oct 25 21:42:17 CST 2013
(7.2) building tables @ Fri Oct 25 21:42:17 CST 2013
Executing: /home/nlp2ct/Aaron/Moses/mosesdecoder/scripts/../bin/lexical-reordering-score /home/
nlp2ct/Aaron/working/train/model/extract.o.sorted.gz 0.5 /home/nlp2ct/Aaron/working/train/model/
reordering-table. --model "wbe msd wbe-msd-bidirectional-fe"
Lexical Reordering Scorer
scores lexical reordering models of several types (hierarchical, phrase-based and word-based-
extraction
(8) learn generation model @ Fri Oct 25 21:43:24 CST 2013
  no generation model requested, skipping step
(9) create moses.ini @ Fri Oct 25 21:43:24 CST 2013
```

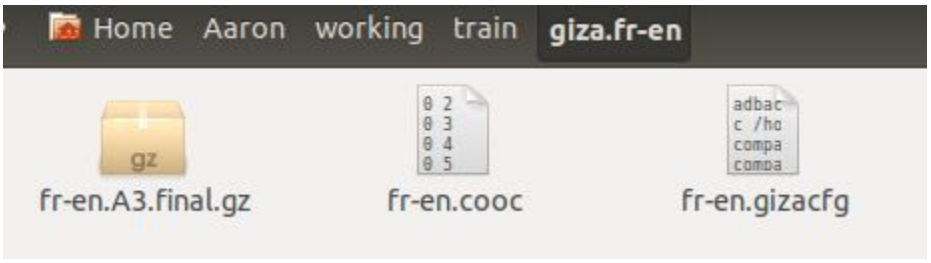
Type “fg 2”, it will show that the progress is finished:

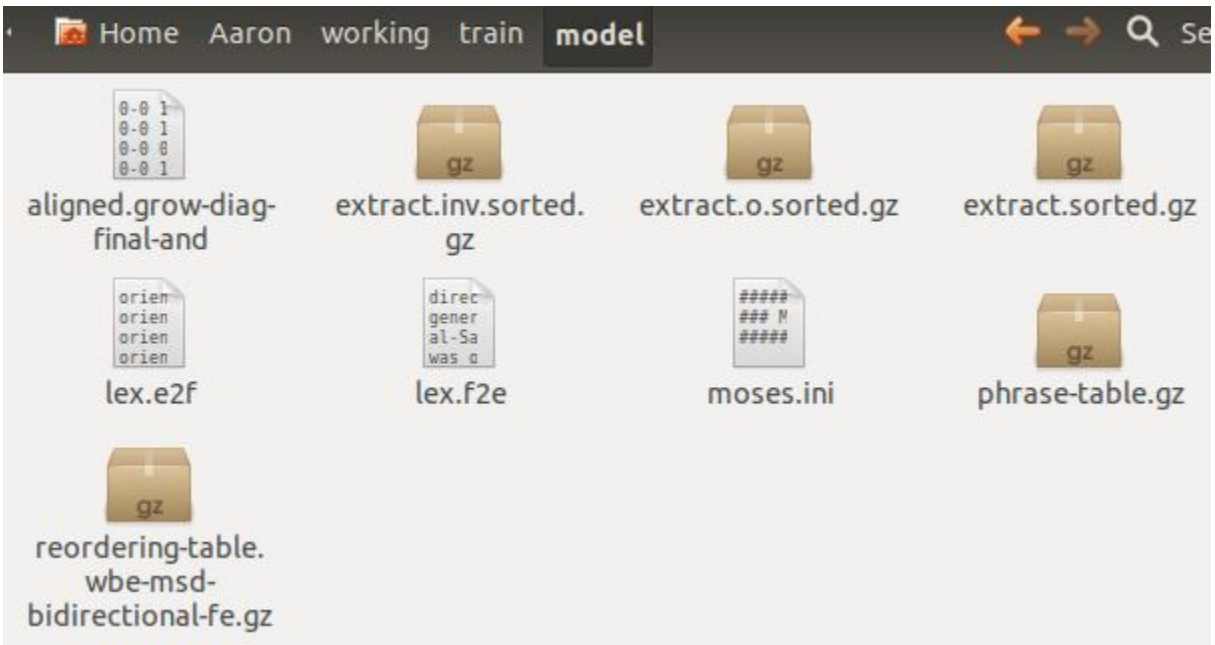
```
   24 root      39  19     0    0    0 S  0.0  0.0   0:00.00 khugepaged
[2]+  Done                    ~/Aaron/Moses/mosesdecoder/scripts/training/train-
model.perl -root-dir train -corpus ~/Aaron/corpus/news-commentary-v8.fr-en.clean
 -f fr -e en -alignment grow-diag-final-and -reordering msd-bidirectional-fe -lm
 0:3:$HOME/Aaron/lm/news-commentary-v8.fr-en.blm.en:8 -external-bin-dir ~/Aaron/
Moses/mosesdecoder/tools &>training.out
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$ fg 2
bash: fg: 2: no such job
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$
```

There will be a “train” file generated at “~/Aaron/working/train”. The “train” file contains 4 files including “corpus, giza.en-fr, giza.fr-en, and model”. The 4 files contain the below files respectively:


Home
Aaron
working
train
corpus
en.vcb
en.vcb.classes
en.vcb.classes.cats
en-fr-int-train.snt
fr.vcb
fr.vcb.classes
fr.vcb.classes.cats
fr-en-int-train.snt


Home
Aaron
working
train
giza.en-fr
en-fr.A3.final.gz
en-fr.cooc
en-fr.gizacfg


Home
Aaron
working
train
giza.fr-en
fr-en.A3.final.gz
fr-en.cooc
fr-en.gizacfg


Home
Aaron
working
train
model
aligned.grow-diag-final-and
extract.inv.sorted.gz
extract.o.sorted.gz
extract.sorted.gz
lex.e2f
lex.f2e
moses.ini
phrase-table.gz
reordering-table.wbe-msd-bidirectional-fe.gz

==========

Tuning translation system:

The weights used by Moses to weight the different models against each other are not optimized, as shown in the moses.ini file. To tune the parameters, it requires a small amount of parallel data, which is separated from the training data.

First, we download the WMT08 data, tokenise and truecase the corpus. The WMT08 corpus is used as development set in WMT12.
Make a mew file “tune” to store the tuning corpus, then

cd ~/corpus/tune
wget http://www.statmt.org/wmt12/dev.tgz
tar zxvf dev.tgz

~/Aaron/corpus/tune$ ~/Aaron/Moses/mosesdecoder/scripts/tokenizer/tokenizer.perl -l en < dev/news-test2008.en > news-test2008.tok.en

~/Aaron/corpus/tune$ ~/Aaron/Moses/mosesdecoder/scripts/tokenizer/tokenizer.perl -l fr < dev/news-test2008.en > news-test2008.tok.fr

~/Aaron/Moses/mosesdecoder/scripts/recaser/truecase.perl --model ~/Aaron/corpus/truecase-model.en < news-test2008.tok.en > news-test2008.true.en

~/Aaron/Moses/mosesdecoder/scripts/recaser/truecase.perl --model ~/Aaron/corpus/truecase-model.fr < news-test2008.tok.fr > news-test2008.true.fr

After above commands, there will be the following files generated for tuning:

Now, we begin the tuning stage using the MERT (minimum error rate training) method.

cd ~/Aaron/working
~/Aaron/Moses/mosesdecoder/scripts/training/mert-moses.pl ~/Aaron/corpus/tune/news-test2008.true.fr ~/Aaron/corpus/tune/news-test2008.true.en ~/Aaron/Moses/mosesdecoder/bin/moses train/model/moses.ini --mertdir ~/Aaron/Moses/mosesdecoder/bin/ &> mert.out &

top,

```
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$ ~/Aaron/Moses/mosesdecoder/scripts/tra
ining/mert-moses.pl ~/Aaron/corpus/tune/news-test2008.true.fr ~/Aaron/corpus/tun
e/news-test2008.true.en ~/Aaron/Moses/mosesdecoder/bin/moses train/model/moses.i
ni --mertdir ~/Aaron/Moses/mosesdecoder/bin/ &> mert.out &
[1] 12434
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$ top

top - 12:26:23 up  2:06,  2 users,  load average: 1.14, 0.72, 0.33
Tasks: 153 total,   1 running, 152 sleeping,   0 stopped,   0 zombie
Cpu(s): 98.3%us,  1.3%sy,  0.0%ni,  0.0%id,  0.0%wa,  0.0%hi,  0.3%si,  0.0%st
Mem:   6786756k total,  4410596k used,  2376160k free,   214852k buffers
Swap:  7336956k total,        0k used,  7336956k free,  2338496k cached

  PID USER      PR  NI  VIRT  RES  SHR S %CPU %MEM    TIME+  COMMAND
12470 nlp2ct    20   0  678m 595m  31m S 93.1  9.0   1:51.71 moses
23689 nlp2ct    20   0 1254m 401m  53m S  3.0  6.1   2:02.51 firefox
```

Type “Ctrl+c” to exit the showing, “jobs” to show the job number, “ctrl+z” to pause the job, “fg [num_job]” to show the running job foreground, “ps” to show running, the jobs are paused as below:

```
    8 root       0 -20     0    0    0 S  0.0  0.0   0:00.00 cpuset
    9 root       0 -20     0    0    0 S  0.0  0.0   0:00.00 khelper
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$ jobs
[1]+  Running                 ~/Aaron/Moses/mosesdecoder/scripts/training/mert-m
oses.pl ~/Aaron/corpus/tune/news-test2008.true.fr ~/Aaron/corpus/tune/news-test2
008.true.en ~/Aaron/Moses/mosesdecoder/bin/moses train/model/moses.ini --mertdir
 ~/Aaron/Moses/mosesdecoder/bin/ &>mert.out &
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$ fg 1
~/Aaron/Moses/mosesdecoder/scripts/training/mert-moses.pl ~/Aaron/corpus/tune/ne
ws-test2008.true.fr ~/Aaron/corpus/tune/news-test2008.true.en ~/Aaron/Moses/mose
sdecoder/bin/moses train/model/moses.ini --mertdir ~/Aaron/Moses/mosesdecoder/bi
n/ &>mert.out
C^Cnlp2ct@nlp2ct-VirtualBox:~/Aaron/working$ jobs
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$ ps
  PID TTY          TIME CMD
12162 pts/3    00:00:00 bash
12504 pts/3    00:00:00 ps
```

To save the time, type the following command to run the tuning with 6 threads,

~/Aaron/Moses/mosesdecoder/scripts/training/mert-moses.pl ~/Aaron/corpus/tune/news-test2008.true.fr ~/Aaron/corpus/tune/news-test2008.true.en ~/Aaron/Moses/mosesdecoder/bin/moses train/model/moses.ini --mertdir ~/Aaron/Moses/mosesdecoder/bin/ --decoder-flags="-threads 6" &> mert.out &

```
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$ ~/Aaron/Moses/mosesdecoder/scripts/tra
ining/mert-moses.pl ~/Aaron/corpus/tune/news-test2008.true.fr ~/Aaron/corpus/tun
e/news-test2008.true.en ~/Aaron/Moses/mosesdecoder/bin/moses train/model/moses.i
ni --mertdir ~/Aaron/Moses/mosesdecoder/bin/ --decoder-flags="-threads 6" &> mer
t.out &
[1] 12508
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$ top

top - 12:29:17 up  2:09,  2 users,  load average: 3.58, 1.41, 0.62
Tasks: 153 total,   1 running, 152 sleeping,   0 stopped,   0 zombie
Cpu(s): 95.7%us,  4.3%sy,  0.0%ni,  0.0%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Mem:   6786756k total,  4620088k used,  2166668k free,   215040k buffers
Swap:  7336956k total,        0k used,  7336956k free,  2335844k cached

  PID USER      PR  NI  VIRT  RES  SHR S %CPU %MEM    TIME+  COMMAND
12512 nlp2ct    20   0 1102m 800m  31m S 85.8 12.1   0:48.32 moses
23689 nlp2ct    20   0 1252m 403m  53m S  5.0  6.1   2:08.16 firefox
```

Type “ctrl+c”, “ps -aux” to show detailed running,

```
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$ ps -aux
Warning: bad ps syntax, perhaps a bogus '-'? See http://procps.sf.net/faq.html
USER       PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root         1  0.0  0.0  24456  2364 ?        Ss   10:19   0:00 /sbin/init
root         2  0.0  0.0      0     0 ?        S    10:19   0:00 [kthreadd]
root         3  0.0  0.0      0     0 ?        S    10:19   0:01 [ksoftirqd/0]
root         5  0.0  0.0      0     0 ?        S    10:19   0:00 [kworker/u:0]
root         6  0.0  0.0      0     0 ?        S    10:19   0:00 [migration/0]
root         7  0.0  0.0      0     0 ?        S    10:19   0:00 [watchdog/0]
```

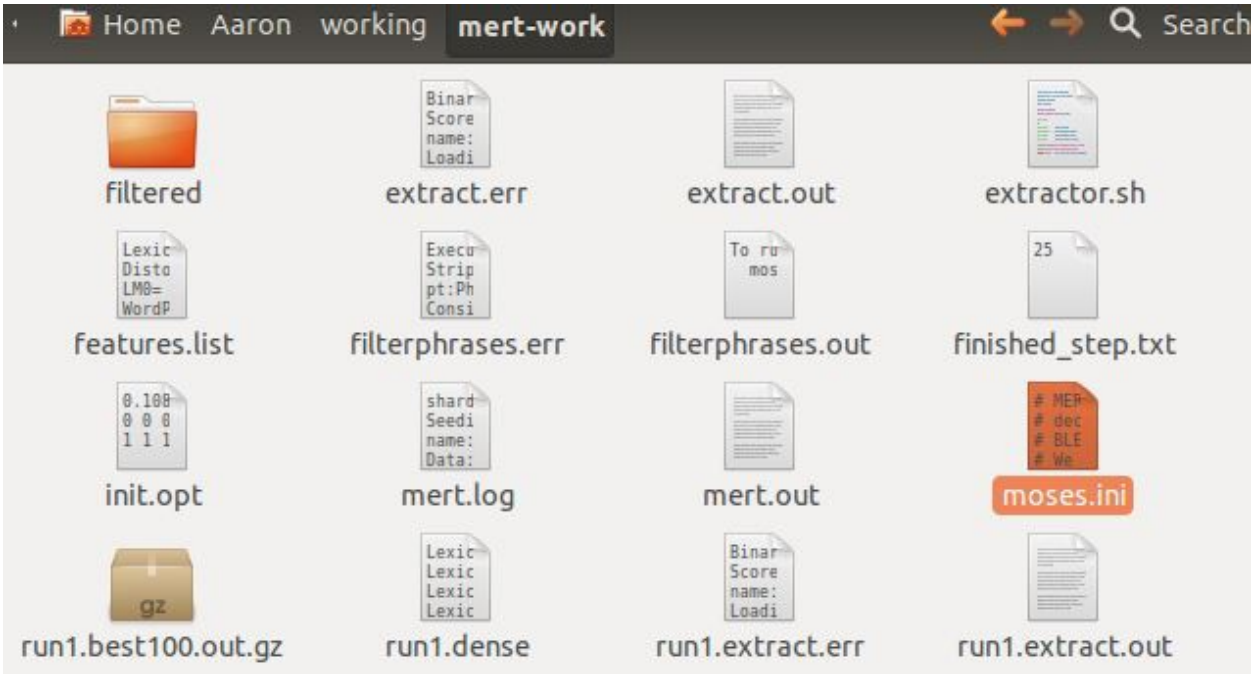
Type “ps –aux|grep moses” to show the running that only related to moses:

```
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$ ps -aux|grep moses
Warning: bad ps syntax, perhaps a bogus '-'? See http://procps.sf.net/faq.html
nlp2ct    2182  0.4  1.3 587756 93516 ?        Sl   10:34   0:33 gedit /home/nlp
2ct/Aaron/working/train/model/moses.ini
nlp2ct   12508  0.0  0.0  29720  5532 pts/3    S    12:28   0:00 /usr/bin/perl -
w /home/nlp2ct/Aaron/Moses/mosesdecoder/scripts/training/mert-moses.pl /home/nlp
2ct/Aaron/corpus/tune/news-test2008.true.fr /home/nlp2ct/Aaron/corpus/tune/news-
test2008.true.en /home/nlp2ct/Aaron/Moses/mosesdecoder/bin/moses train/model/mos
es.ini --mertdir /home/nlp2ct/Aaron/Moses/mosesdecoder/bin/ --decoder-flags=-thr
eads 6
nlp2ct   12511  0.0  0.0   4404   608 pts/3    S    12:28   0:00 sh -c /home/nlp
```

A long time later, type “top” showing there will be no running of “moses”, which means the tuning is finished:

```
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$ top

top - 10:06:37 up 23:47,  2 users,  load average: 3.15, 1.45, 1.44
Tasks: 152 total,   2 running, 150 sleeping,   0 stopped,   0 zombie
top - 10:53:09 up 1 day, 33 min,  2 users,  load average: 0.09, 0.10, 0.42
Tasks: 152 total,   3 running, 149 sleeping,   0 stopped,   0 zombie
Cpu(s):  1.0%us,  0.7%sy,  0.0%ni, 98.3%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Mem:   6786756k total,  6664780k used,   121976k free,     6220k buffers
Swap:  7336956k total,  2599052k used,  4737904k free,    77656k cached

  PID USER      PR  NI  VIRT  RES  SHR S %CPU %MEM    TIME+  COMMAND
12260 nlp2ct    20   0  396m  11m 2628 R  1.3  0.2  24:41.10 plugin-containe
 1142 root      20   0  369m 171m 9420 S  0.3  2.6   6:24.51 Xorg
16606 nlp2ct    20   0 17340  800  436 R  0.3  0.0   0:07.77 top
```

There will be one document “mert.out” generated under working file, at the same time, a “mert-work” file containing **the tuning results “moses.ini”** is generated under working file as below:



Begin the tuning at 2013-10-29-17:00, finish the tuning and generate the “moses.ini” file at 2013-10-30-09:44, it takes **around 16 hours** (using 7 threads). The “moses.ini” file contains the tuned parameters and other information as following:

```
# MERT optimized configuration
# decoder /home/nlp2ct/Aaron/Moses/mosesdecoder/bin/moses
# BLEU 0.186923 on dev /home/nlp2ct/Aaron/corpus/tune/news-test2008.true.fr
# We were before running iteration 25
# finished Wed Oct 30 09:44:37 CST 2013
### MOSES CONFIG FILE ###
#########################

# input factors
[input-factors]
0

# mapping steps
[mapping]
0 T 0

[distortion-limit]
6

# feature functions
[feature]
UnknownWordPenalty
WordPenalty
PhrasePenalty
PhraseDictionaryMemory name=TranslationModel0 table-limit=20 num-features=4 path=/home/nlp2ct/Aaron/
working/train/model/phrase-table.gz input-factor=0 output-factor=0
LexicalReordering name=LexicalReordering0 num-features=6 type=wbe-msd-bidirectional-fe-allff input-
factor=0 output-factor=0 path=/home/nlp2ct/Aaron/working/train/model/reordering-table.wbe-msd-
bidirectional-fe.gz
Distortion
KENLM lazyken=0 name=LM0 factor=0 path=/home/nlp2ct/Aaron/lm/news-commentary-v8.fr-en.blm.en order=3

# dense weights for feature functions

[threads]
7
[weight]

LexicalReordering0= 0.108647 0.0221523 0.0679283 0.139558 0.0377834 0.0857802
Distortion0= 0.0713619
LM0= 0.0763578
WordPenalty0= -0.0298535
PhrasePenalty0= 0.124991
TranslationModel0= 0.0275796 0.0760543 0.0925488 0.0394041
```

In the file “working/mert.out”, it shows:

```
(25) BEST at 25: 0.108647 0.0221523 0.0679283 0.139558 0.0377834 0.0857802 0.0713
Executing: \cp -f mert.log run25.mert.log
featlist: LexicalReordering0=0.108647
featlist: LexicalReordering0=0.0221523
featlist: LexicalReordering0=0.0679283
featlist: LexicalReordering0=0.139558
featlist: LexicalReordering0=0.0377834
featlist: LexicalReordering0=0.0857802
featlist: Distortion0=0.0713619
featlist: LM0=0.0763578
featlist: WordPenalty0=-0.0298535
featlist: PhrasePenalty0=0.124991
featlist: TranslationModel0=0.0275796
featlist: TranslationModel0=0.0760543
featlist: TranslationModel0=0.0925488
featlist: TranslationModel0=0.0394041
Parsing --decoder-flags: |-threads 7|
Saving new config to: ./moses.ini
Saved: ./moses.ini
Training finished at Wed Oct 30 09:44:37 CST 2013
```

==========

Testing the translation model:

This stage is to test the translation quality of the built translation model. The translation quality is measured by the automatic evaluation metric score, such as the metric BLEU(2002), METEOR(2005), and LEOPR(2012), etc.

Type the command:
cd ~/Aaron
~/Aaron/Moses/mosesdecoder/bin/moses -f ~/Aaron/working/mert-work/moses.ini

```
nlp2ct@nlp2ct-VirtualBox:~/Aaron$ ~/Aaron/Moses/mosesdecoder/bin/moses -f ~/Aaro
n/working/mert-work/moses.ini
Defined parameters (per moses.ini or switch):
        config: /home/nlp2ct/Aaron/working/mert-work/moses.ini
        distortion-limit: 6
        feature: UnknownWordPenalty WordPenalty PhrasePenalty PhraseDictionaryMe
```

It will take some minutes to show the finish the initializing LexicalReordering and reading phrase-table as following:

```
Initializing LexicalReordering..
line=Distortion
FeatureFunction: Distortion0 start: 13 end: 13
line=KENLM lazyken=0 name=LM0 factor=0 path=/home/nlp2ct/Aaron/lm/news-commentar
y-v8.fr-en.blm.en order=3
FeatureFunction: LM0 start: 14 end: 14
Loading table into memory...done.
Start loading text SCFG phrase table. Moses  format : [128.000] seconds
Reading /home/nlp2ct/Aaron/working/train/model/phrase-table.gz
----5---10---15---20---25---30---35---40---45---50---55---60---65---70---75---80
---85---90---95--100
********************************************************************************
********************
IO from STDOUT/STDIN
Created input-output object : [996.000] seconds
```

Type a French sentence “c'est une petite maison.(//this is a small house.)” and “enter”, it will translate it as below, which is not a fully translated sentence:

```
Created input-output object : [996.000] seconds
c'est une petite maison.
Translating line 0  in thread id 140224420796160
Translating: c'est une petite maison.
Line 0: Collecting options took 1.000 seconds
Line 0: Search took 4.000 seconds
c'est a small maison.
BEST TRANSLATION: c'est|UNK|UNK|UNK a small maison.|UNK|UNK|UNK [1111]  [total=-
202.004] core=(-200.000,-4.000,3.000,-1.392,-3.215,-0.705,-1.600,-0.403,0.000,0.
000,-0.181,0.000,0.000,0.000,-26.435)
Line 0: Translation took 5.000 seconds total
```

Type another French sentence “vous êtes beau (// you are handsome)”, it will translate it into “you are beautiful” as below:

```
vous êtes beau
Translating line 1  in thread id 140224404010752
Translating: vous êtes beau
Line 1: Collecting options took 1.000 seconds
Line 1: Search took 0.000 seconds
you are beautiful
BEST TRANSLATION: you are beautiful [111]  [total=-2.952] core=(0.000,-3.000,2.0
00,-3.737,-7.607,-2.750,-4.150,-0.538,0.000,0.000,-0.280,0.000,0.000,0.000,-27.4
34)
Line 1: Translation took 1.000 seconds total
```

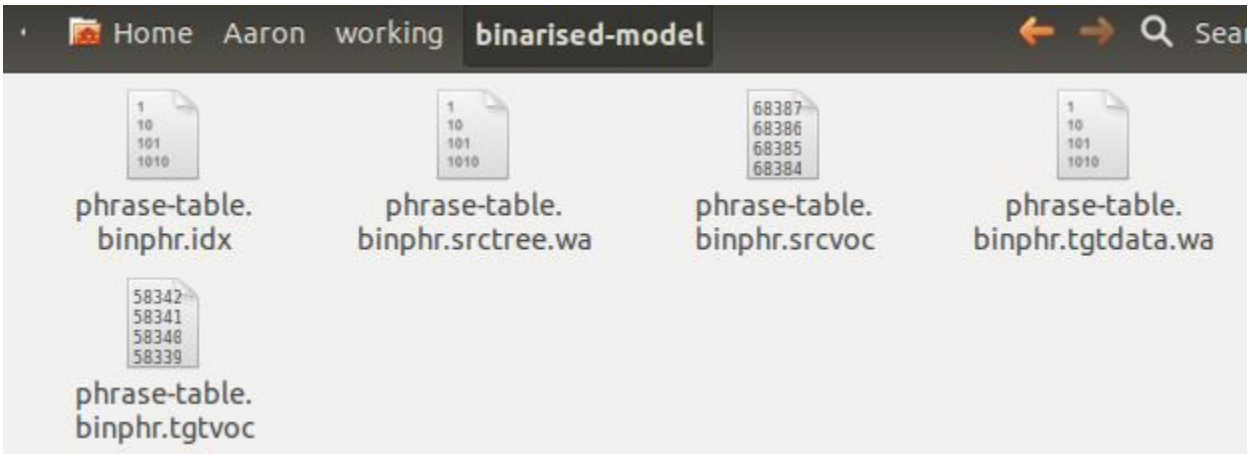
Type “ctrl+c” to exit the job.
To make the translation faster, we should binary the lexicalized reordering models and the phrase-table.

mkdir ~/Aaron/working/binarised-model

Type the following command to binaries the phrase-table:
~/Aaron/Moses/mosesdecoder/bin/processPhraseTable -ttable 0 0 train/model/phrase-table.gz -nscores 5 -out binarised-model/phrase-table

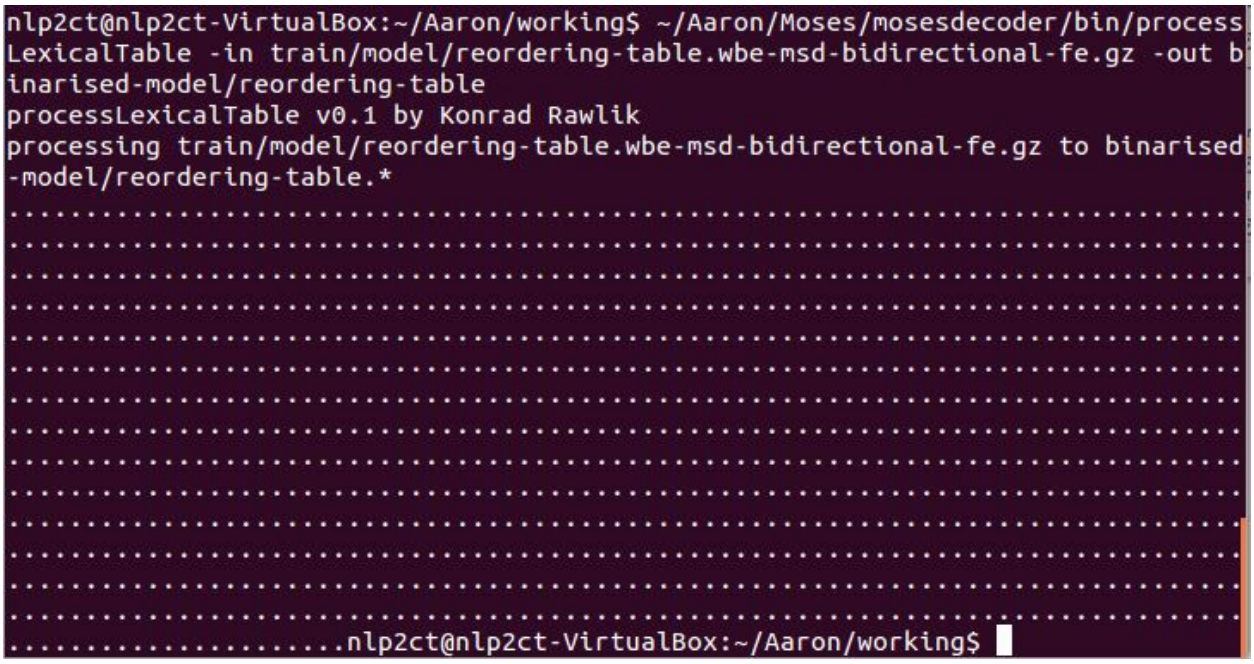
```
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$ ~/Aaron/Moses/mosesdecoder/bin/process
PhraseTable -ttable 0 0 train/model/phrase-table.gz -nscores 5 -out binarised-mo
del/phrase-table
processing ptree for train/model/phrase-table.gz
.................................................[phrase:500000]
.................................................[phrase:1000000]
.................................................[phrase:1500000]
.................................................[phrase:2000000]
.................................................[phrase:2500000]
.................................................[phrase:3000000]
.................................................[phrase:3500000]
.................................................[phrase:4000000]
.................................................[phrase:4500000]
.................................................[phrase:5000000]
.................................................[phrase:5500000]
.................................................[phrase:6000000]
.................................................[phrase:6500000]
.................................................[phrase:7000000]
.................................................[phrase:7500000]
...distinct source phrases: 7537247 distinct first words of source phrases: 6838
8 number of phrase pairs (line count): 11424565
Count of lines with missing alignments: 0/11424565
WARNING: there are src voc entries with no phrase translation: count 9684
There exists phrase translations for 58704 entries
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$
```

It will generate the following files:

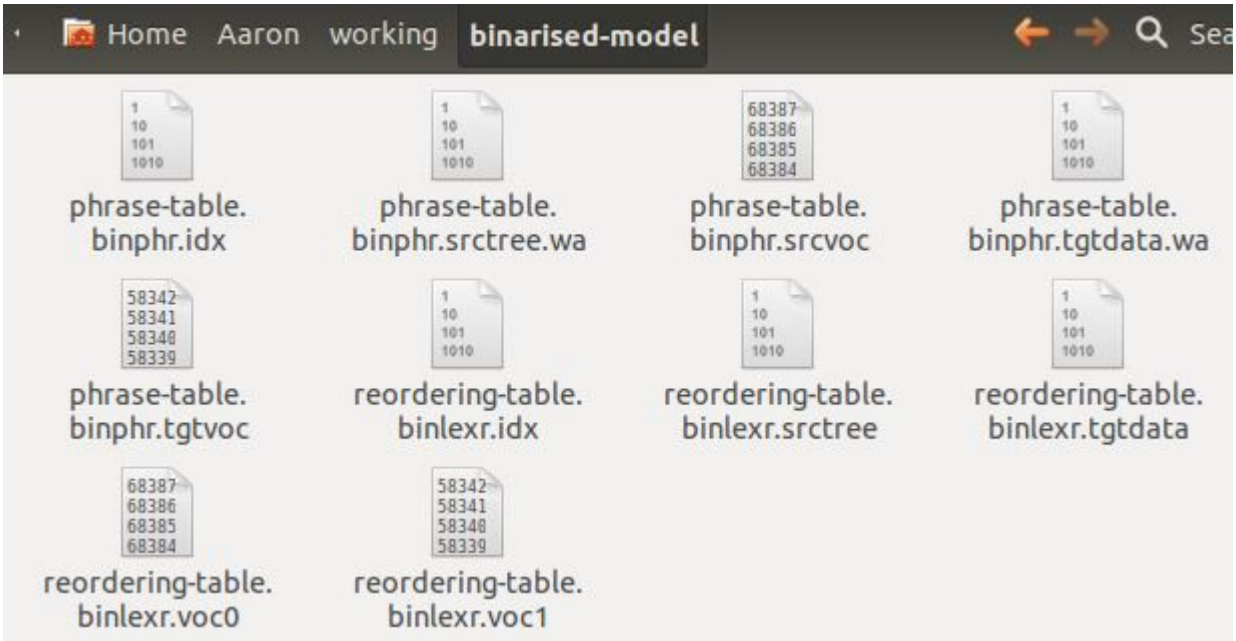


To binary the lexical reordering-table, type the following command:

~/Aaron/Moses/mosesdecoder/bin/processLexicalTable -in train/model/reordering-table.wbe-msd-bidirectional-fe.gz -out binarised-model/reordering-table



It will generate the files:



Copy the ~/working/mert-work/moses.ini document into the ~/binarised-model. Then change the moses.ini content as below to point to the binarised files:

```
# feature functions
[feature]
UnknownWordPenalty
WordPenalty
PhrasePenalty
PhraseDictionaryBinary name=TranslationModel0 table-limit=20 num-features=4 path=/home/nlp2ct/Aaron/
working/binarised-model/phrase-table input-factor=0 output-factor=0
LexicalReordering name=LexicalReordering0 num-features=6 type=wbe-msd-bidirectional-fe-allff input-
factor=0 output-factor=0 path=/home/nlp2ct/Aaron/working/binarised-model/reordering-table
Distortion
KENLM lazyken=0 name=LM0 factor=0 path=/home/nlp2ct/Aaron/lm/news-commentary-v8.fr-en.blm.en order=3
```

Using the binarised tables, the loading will be very fast using the following command:
~/Aaron/Moses/mosesdecoder/bin/moses -f ~/Aaron/working/binarised-model/moses.ini

```
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$ ~/Aaron/Moses/mosesdecoder/bin/moses -
f ~/Aaron/working/binarised-model/moses.ini
```

If you type the above two french sentences again it will generate the same translations as before.

Type the commands to prepare the testing data as below:
mkdir ~/Aaron/corpus/test
cd ~/corpus/test
~/Aaron/Moses/mosesdecoder/scripts/tokenizer/tokenizer.perl -l en
<~/Aaron/corpus/tune/dev/newstest2011.en > newstest2011.tok.en

~/Aaron/Moses/mosesdecoder/scripts/tokenizer/tokenizer.perl -l fr <
~/Aaron/corpus/tune/dev/newstest2011.fr > newstest2011.tok.fr

~/Aaron/Moses/mosesdecoder/scripts/recaser/truecase.perl --model ~/Aaron/corpus/truecase-model.en < newstest2011.tok.en > newstest2011.true.en

~/Aaron/Moses/mosesdecoder/scripts/recaser/truecase.perl --model ~/Aaron/corpus/truecase-model.fr < newstest2011.tok.fr > newstest2011.true.fr

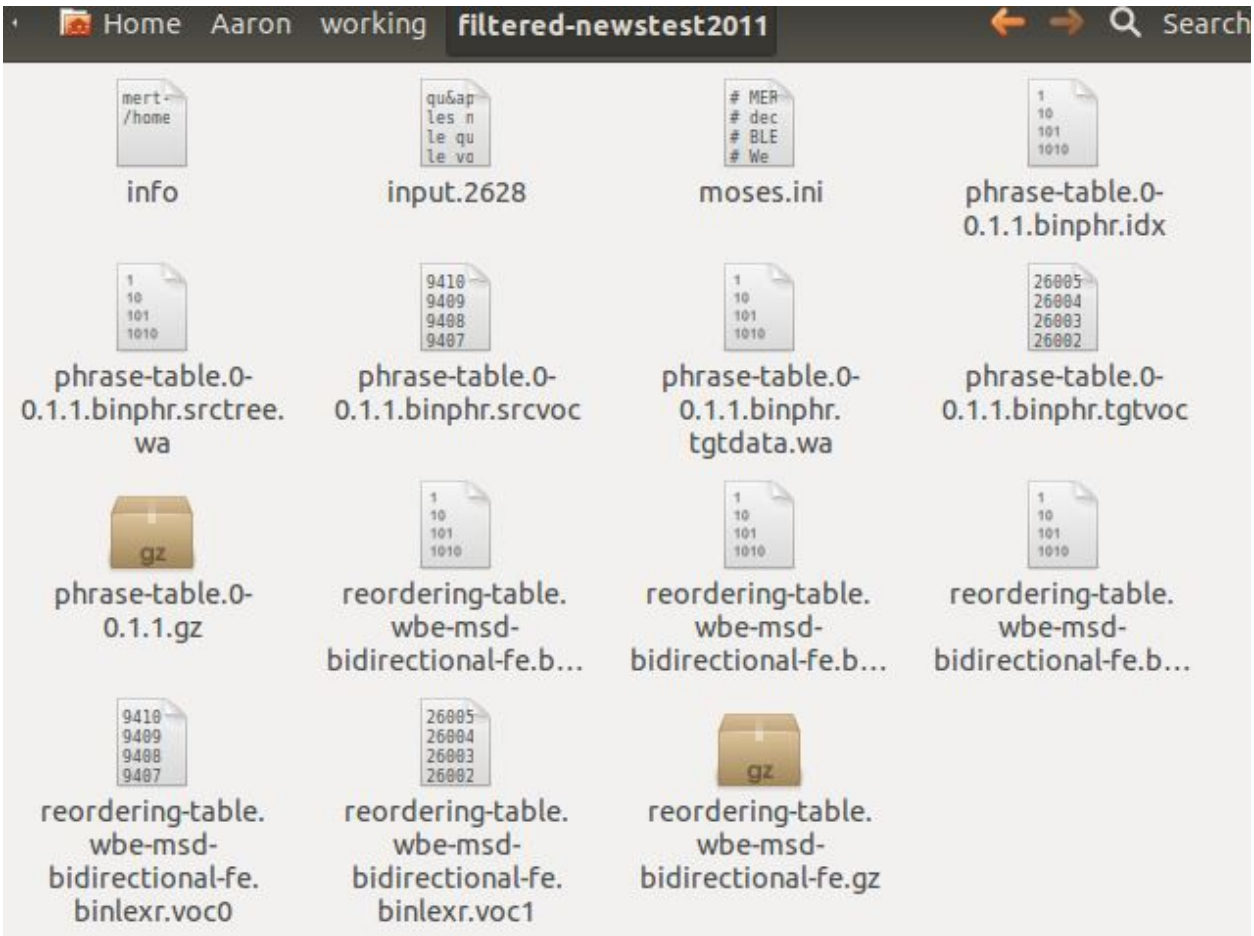
To make the translation faster, we can filter the trained translation model to retain the entries that are only needed to translate the offered test corpus:
VirtualBox:~/Aaron/working$ ~/Aaron/Moses/mosesdecoder/scripts/training/filter-model-given-input.pl filtered-newstest2011 mert-work/moses.ini
~/Aaron/corpus/test/newstest2011.true.fr -Binarizer
~/Aaron/Moses/mosesdecoder/bin/processPhraseTable

```
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$ ~/Aaron/Moses/mosesdecoder/scripts/tra
ining/filter-model-given-input.pl filtered-newstest2011 mert-work/moses.ini ~/Aa
ron/corpus/test/newstest2011.true.fr -Binarizer ~/Aaron/Moses/mosesdecoder/bin/p
rocessPhraseTable
```

The filtering and binary stage finished as below:

```
filtering /home/nlp2ct/Aaron/working/train/model/reordering-table.wbe-msd-bidire
ctional-fe.gz -> /home/nlp2ct/Aaron/working/filtered-newstest2011/reordering-tab
le.wbe-msd-bidirectional-fe...
1054277 of 11424565 phrases pairs used (9.23%) - note: max length 10
binarizing.../home/nlp2ct/Aaron/Moses/mosesdecoder/bin/processLexicalTable -in /
home/nlp2ct/Aaron/working/filtered-newstest2011/reordering-table.wbe-msd-bidirec
tional-fe.gz -out /home/nlp2ct/Aaron/working/filtered-newstest2011/reordering-ta
ble.wbe-msd-bidirectional-fe
processLexicalTable v0.1 by Konrad Rawlik
processing /home/nlp2ct/Aaron/working/filtered-newstest2011/reordering-table.wbe
-msd-bidirectional-fe.gz to /home/nlp2ct/Aaron/working/filtered-newstest2011/reo
rdering-table.wbe-msd-bidirectional-fe.*
.....
.....................To run the decoder, please call:
  moses -f /home/nlp2ct/Aaron/working/filtered-newstest2011/moses.ini -i /home/n
lp2ct/Aaron/working/filtered-newstest2011/input.2628
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$
```

Above command will generate a “~/Aaron/working/filtered-newstest2011” file that contains the following documents:



Then we translate the testing corpus and score the translation quality using BLEU metric:

nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$ ~/Aaron/Moses/mosesdecoder/bin/moses -f ~/Aaron/working/filtered-newstest2011/moses.ini < ~/Aaron/corpus/test/newstest2011.true.fr > ~/Aaron/working/newstest2011.translated.en 2> ~/Aaron/working/newstest2011.out

It takes abound 25 minutes to finish the translation. There will be two documents generated "~/working/newstest2011.out & ~/working/newstest2011.translated.en".
The document newstest2011.translated.en contains the translated output sentences; the document newstest2011.out contains the detailed translation procedure and its volume is larger:

newstest2011.translated.en ✖ newstest2011.out ✖

```
that is what will happen ? CSSD is neither Voldemort or pâtisserie in Prague .
the new CSSD elected to the mayor of the capital will apparently overcome a real linguistic barrier to
understand the level of city council with their old and new collegues of the ODS .
the newspaper Actualne.cz has &quot; tested &quot; the new members of the city council belonging to
CSSD , whether they know the argôt who was born over the years dernieres mayoral , when it were the
current coalition partners that ruled in Prague .
the vocabulary codé that the best policy of Prague put underway over the previous era of the mayor
Pavel Bem described a few people , situations , and causes of the mainland .
it is with surprise that he has been discovered that the new elected do not understand trés well the
terms known .
at least the they say .
&quot; that is Valdemort . &quot;
```

newstest2011.translated.en ✖ newstest2011.out ✖

```
Defined parameters (per moses.ini or switch):
        config: /home/nlp2ct/Aaron/working/filtered-newstest2011/moses.ini
        distortion-limit: 6
        feature: UnknownWordPenalty WordPenalty PhrasePenalty PhraseDictionaryBinary
name=TranslationModel0 table-limit=20 num-features=4 path=/home/nlp2ct/Aaron/working/filtered-
newstest2011/phrase-table.0-0.1.1 input-factor=0 output-factor=0 LexicalReordering
name=LexicalReordering0 num-features=6 type=wbe-msd-bidirectional-fe-allff input-factor=0 output-
factor=0 path=/home/nlp2ct/Aaron/working/filtered-newstest2011/reordering-table.wbe-msd-bidirectional-
fe Distortion KENLM lazyken=0 name=LM0 factor=0 path=/home/nlp2ct/Aaron/lm/news-commentary-v8.fr-
en.blm.en order=3
        input-factors: 0
        mapping: 0 T 0
        threads: 7
        weight: LexicalReordering0= 0.108647 0.0221523 0.0679283 0.139558 0.0377834 0.0857802
Distortion0= 0.0713619 LM0= 0.0763578 WordPenalty0= -0.0298535 PhrasePenalty0= 0.124991
TranslationModel0= 0.0275796 0.0760543 0.0925488 0.0394041
/home/nlp2ct/Aaron/Moses/mosesdecoder/bin
line=UnknownWordPenalty
FeatureFunction: UnknownWordPenalty0 start: 0 end: 0
line=WordPenalty
FeatureFunction: WordPenalty0 start: 1 end: 1
line=PhrasePenalty
FeatureFunction: PhrasePenalty0 start: 2 end: 2
line=PhraseDictionaryBinary name=TranslationModel0 table-limit=20 num-features=4 path=/home/nlp2ct/
```

Type the following command to test the BLEU score of the automatic translation, as compared with the reference translation:
VirtualBox:~/Aaron/working$ ~/Aaron/Moses/mosesdecoder/scripts/generic/multi-bleu.perl -lc ~/Aaron/corpus/test/newstest2011.true.en < ~/Aaron/working/newstest2011.translated.en

```
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$ ~/Aaron/Moses/mosesdecoder/scripts/gen
eric/multi-bleu.perl -lc ~/Aaron/corpus/test/newstest2011.true.en < ~/Aaron/work
ing/newstest2011.translated.en
BLEU = 23.41, 60.0/29.8/16.8/10.0 (BP=1.000, ratio=1.017, hyp_len=76012, ref_len
=74753)
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$
```

It shows that the BLEU score is 23.41.

==========

Trial of Tree to tree translation using Moses:

Install the chart decoder:

We already install the chart decoder as the executable file:
~/Aaron/Moses/mosesdecoder/bin/moses_chart

First use the following command to try the string to tree model:

```
nlp2ct@nlp2ct-VirtualBox:~/Aaron/Moses/sample-models$
```

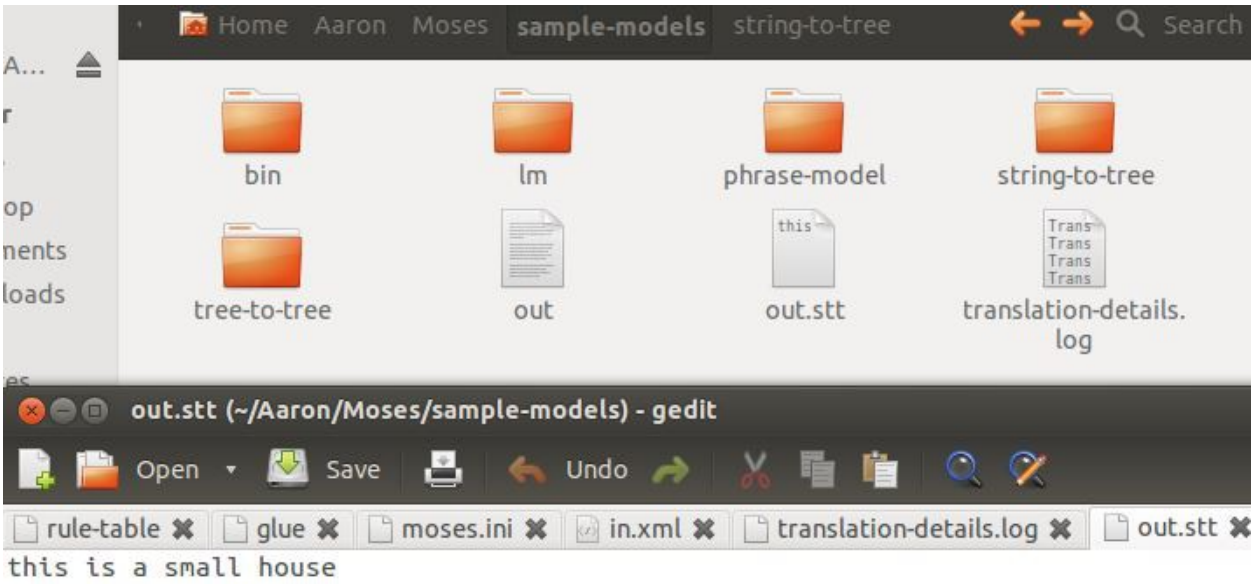
```
nlp2ct@nlp2ct-VirtualBox:~/Aaron/Moses/sample-models$ ~/Aaron/Moses/mosesdecoder
/bin/moses_chart -f string-to-tree/moses.ini < string-to-tree/in >out.stt
```

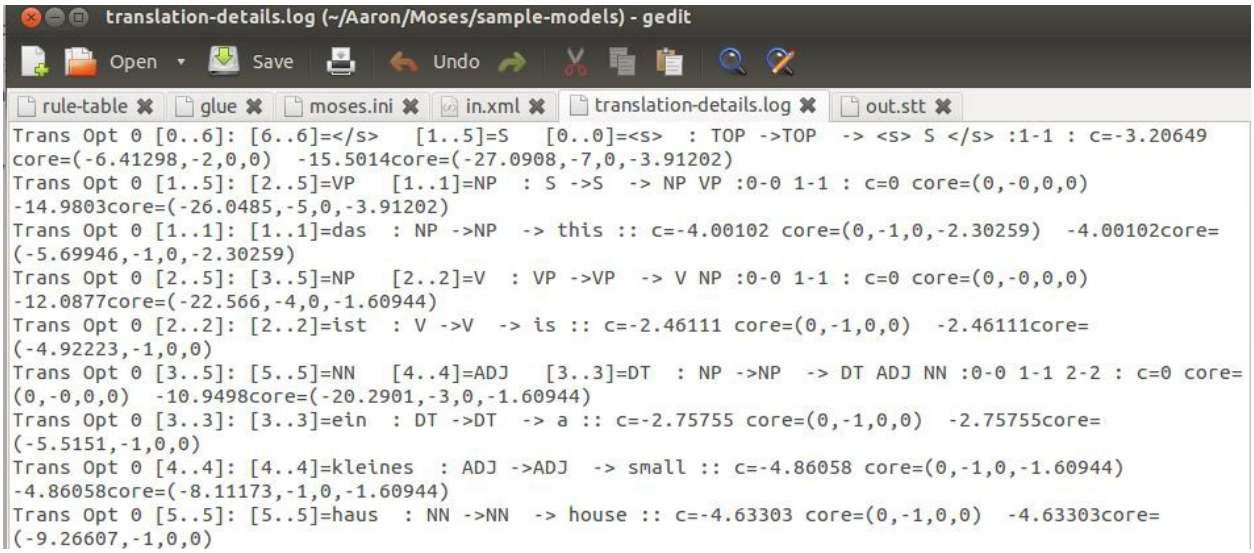
The running will call the language model: ~/lm/europarl.srilm.gz

```
Defined parameters (per moses.ini or switch):
        config: string-to-tree/moses.ini
        cube-pruning-pop-limit: 1000
        feature: KENLM name=LM factor=0 order=3 num-features=1 path=lm/europarl.
srilm.gz WordPenalty UnknownWordPenalty PhraseDictionaryMemory input-factor=0 ou
tput-factor=0 path=string-to-tree/rule-table num-features=1 table-limit=20
        input-factors: 0
        inputtype: 3
        mapping: 0 T 0
        max-chart-span: 20 1000
        non-terminals: X S
        search-algorithm: 3
        translation-details: translation-details.log
        weight: WordPenalty0= 0 LM= 0.5 PhraseDictionaryMemory0= 0.5
/home/nlp2ct/Aaron/Moses/mosesdecoder/bin
line=KENLM name=LM factor=0 order=3 num-features=1 path=lm/europarl.srilm.gz
FeatureFunction: LM start: 0 end: 0
Loading the LM will be faster if you build a binary file.
Reading lm/europarl.srilm.gz
----5---10---15---20---25---30---35---40---45---50---55---60---65---70---75---80
---85---90---95--100
**The ARPA file is missing <unk>.  Substituting log10 probability -100.000.
********************************************************************************
******************
line=WordPenalty
FeatureFunction: WordPenalty0 start: 1 end: 1
line=UnknownWordPenalty
FeatureFunction: UnknownWordPenalty0 start: 2 end: 2
line=PhraseDictionaryMemory input-factor=0 output-factor=0 path=string-to-tree/r
ule-table num-features=1 table-limit=20
FeatureFunction: PhraseDictionaryMemory0 start: 3 end: 3
Start loading text SCFG phrase table. Moses  format : [1.000] seconds
Reading string-to-tree/rule-table
----5---10---15---20---25---30---35---40---45---50---55---60---65---70---75---80
---85---90---95--100
********************************************************************************
```

```
********************
max-chart-span: 20
IO from STDOUT/STDIN
Created input-output object : [1.000] seconds
Translating: <s> das ist ein kleines haus </s>  ||| [0,0]=X (1) [0,1]=X (1) [0,2
]=X (1) [0,3]=X (1) [0,4]=X (1) [0,5]=X (1) [0,6]=X (1) [1,1]=X (1) [1,2]=X (1)
[1,3]=X (1) [1,4]=X (1) [1,5]=X (1) [1,6]=X (1) [2,2]=X (1) [2,3]=X (1) [2,4]=X
(1) [2,5]=X (1) [2,6]=X (1) [3,3]=X (1) [3,4]=X (1) [3,5]=X (1) [3,6]=X (1) [4,4
]=X (1) [4,5]=X (1) [4,6]=X (1) [5,5]=X (1) [5,6]=X (1) [6,6]=X (1)

  0   1   2   3   4   5   6
  0   3   2   2   2   1   0
    0   0   0   0   0   0
      0   0   0   3   0
        0   0   4   0
          0   4   0
            0   0
              1
BEST TRANSLATION: 41 TOP  -> <s> S </s> :1-1 : c=-3.206 core=(-6.413,-2.000,0.00
0,0.000)  [0..6] 20 [total=-15.501] core=(-27.091,-7.000,0.000,-3.912)
Translation took 0.000 seconds
End. : [1.000] seconds
Name:moses_chart        VmPeak:190440 kB        VmRSS:31860 kB  RSSMax:32428 kBu
ser:0.656       sys:0.092       CPU:0.748       real:0.840
nlp2ct@nlp2ct-VirtualBox:~/Aaron/Moses/sample-models$
```

It will generate the document “out.stt” and “translation-details.log” under the directory: ~/Aaron/Moses/sample-models



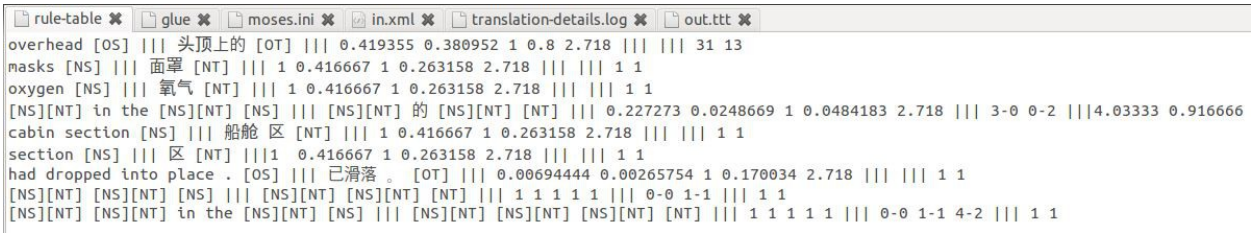


Secondly use the following command to try the tree-to-tree model:

```
nlp2ct@nlp2ct-VirtualBox:~/Aaron/Moses/sample-models$ ~/Aaron/Moses/mosesdecoder
/bin/moses_chart -f tree-to-tree/moses.ini < tree-to-tree/in.xml > out.ttt
```

The in.xml file contains the sentence:

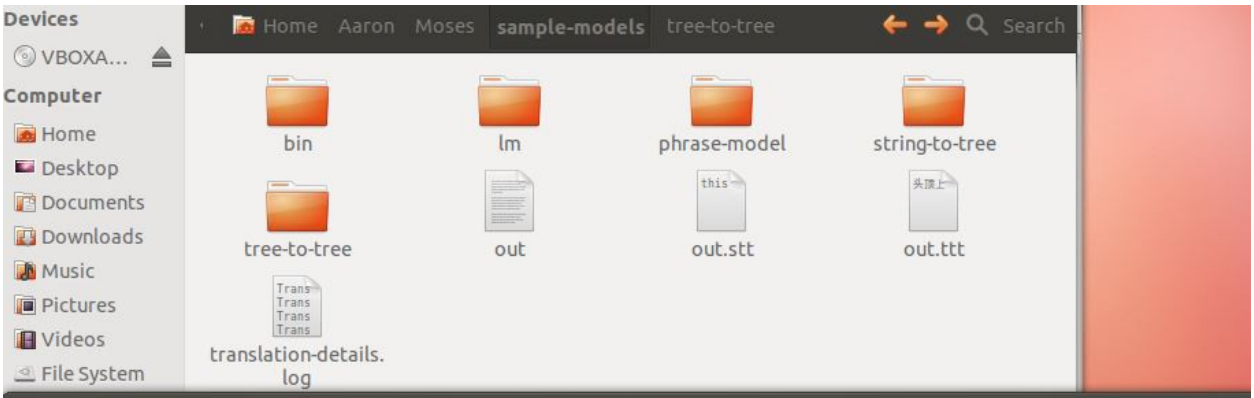
The running calls the rule table: ~/tree-to-tree/rule-table



```
Defined parameters (per moses.ini or switch):
        config: tree-to-tree/moses.ini
        cube-pruning-pop-limit: 1000
        feature: WordPenalty UnknownWordPenalty PhraseDictionaryMemory input-fac
tor=0 output-factor=0 path=tree-to-tree/rule-table num-features=5 table-limit=20
 PhraseDictionaryMemory input-factor=0 output-factor=0 path=tree-to-tree/glue nu
m-features=1 table-limit=20
        input-factors: 0
        inputtype: 3
        mapping: 0 T 0 1 T 1
        max-chart-span: 20 1000
        non-terminals: X S
        search-algorithm: 3
        translation-details: translation-details.log
        weight: WordPenalty0= 0 PhraseDictionaryMemory0= 0.5 0.5 0.5 0.5 0.5 Phr
aseDictionaryMemory1= -0.5
/home/nlp2ct/Aaron/Moses/mosesdecoder/bin
line=WordPenalty
FeatureFunction: WordPenalty0 start: 0 end: 0
line=UnknownWordPenalty
FeatureFunction: UnknownWordPenalty0 start: 1 end: 1
line=PhraseDictionaryMemory input-factor=0 output-factor=0 path=tree-to-tree/rul
e-table num-features=5 table-limit=20
FeatureFunction: PhraseDictionaryMemory0 start: 2 end: 6
line=PhraseDictionaryMemory input-factor=0 output-factor=0 path=tree-to-tree/glu
e num-features=1 table-limit=20
FeatureFunction: PhraseDictionaryMemory1 start: 7 end: 7
Start loading text SCFG phrase table. Moses  format : [0.000] seconds
Reading tree-to-tree/rule-table
----5---10---15---20---25---30---35---40---45---50---55---60---65---70---75---80
---85---90---95--100
********************************************************************************
********************
Start loading text SCFG phrase table. Moses  format : [0.000] seconds
Reading tree-to-tree/glue
----5---10---15---20---25---30---35---40---45---50---55---60---65---70---75---80
---85---90---95--100
********************************************************************************
```

```
********************
max-chart-span: 20
max-chart-span: 1000
IO from STDOUT/STDIN
Created input-output object : [0.000] seconds
Translating: <s> overhead oxygen masks in the cabin section had dropped into pla
ce . </s>  ||| [0,0]=X (1) [0,1]=X (1) [0,2]=X (1) [0,3]=X (1) [0,4]=X (1) [0,5]
=X (1) [0,6]=X (1) [0,7]=X (1) [0,8]=X (1) [0,9]=X (1) [0,10]=X (1) [0,11]=X (1)
 [0,12]=X (1) [0,13]=X (1) [1,1]=X (1) [1,1]=OS (1) [1,2]=X (1) [1,3]=X (1) [1,4
]=X (1) [1,5]=X (1) [1,6]=X (1) [1,7]=X (1) [1,8]=X (1) [1,9]=X (1) [1,10]=X (1)
 [1,11]=X (1) [1,12]=X (1) [1,12]=TOP (1) [1,13]=X (1) [2,2]=X (1) [2,2]=NS (1)
[2,3]=X (1) [2,4]=X (1) [2,5]=X (1) [2,6]=X (1) [2,7]=X (1) [2,8]=X (1) [2,9]=X
(1) [2,10]=X (1) [2,11]=X (1) [2,12]=X (1) [2,13]=X (1) [3,3]=X (1) [3,3]=NS (1)
 [3,4]=X (1) [3,5]=X (1) [3,6]=X (1) [3,7]=X (1) [3,8]=X (1) [3,9]=X (1) [3,10]=
X (1) [3,11]=X (1) [3,12]=X (1) [3,13]=X (1) [4,4]=X (1) [4,5]=X (1) [4,6]=X (1)
 [4,7]=X (1) [4,8]=X (1) [4,9]=X (1) [4,10]=X (1) [4,11]=X (1) [4,12]=X (1) [4,1
3]=X (1) [5,5]=X (1) [5,6]=X (1) [5,7]=X (1) [5,8]=X (1) [5,9]=X (1) [5,10]=X (1
```

When finishing it generates document “out.ttt” and update the document “translation-details.log”

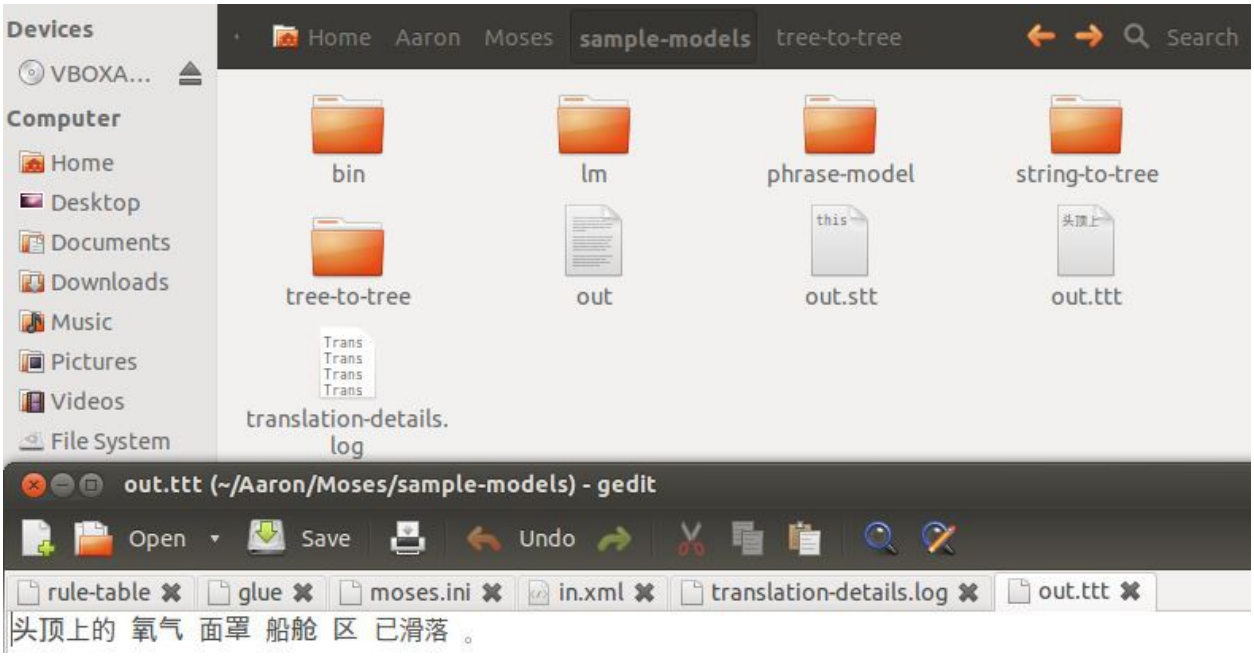


translation-details.log (~/Aaron/Moses/sample-models) - gedit

Open Save Undo

rule-table | glue | moses.ini | in.xml | translation-details.log | out.ttt

```
Trans Opt 0 [0..13]: [13..13]=</s>   [0..12]=S  : S ->S  -> S </s> :0-0 : c=0 core=(-1,0,0,0,0,0,0,0)
-9.6804core=(-9,0,-5.83885,-9.52184,0,-5.9999,4.99948,2.99969)
Trans Opt 0 [0..12]: [8..12]=OT   [0..7]=S  : S ->S  -> S OT :0-0 1-1 : c=-0.499948 core=
(-0,0,0,0,0,0,0,0.999896)  -9.6804core=(-8,0,-5.83885,-9.52184,0,-5.9999,4.99948,2.99969)
Trans Opt 0 [0..7]: [2..7]=NT   [0..1]=S  : S ->S  -> S NT :0-0 1-1 : c=-0.499948 core=
(-0,0,0,0,0,0,0,0.999896)  -3.34444core=(-6,0,-0.869037,-3.59149,0,-4.22815,3.99959,1.99979)
Trans Opt 0 [0..1]: [1..1]=OT   [0..0]=S  : S ->S  -> S OT :0-0 1-1 : c=-0.499948 core=
(-0,0,0,0,0,0,0,0.999896)  -1.02863core=(-2,0,-0.869037,-0.965082,0,-0.223144,0.999896,0.999896)
Trans Opt 0 [0..0]: [0..0]=<s>  : S ->S  -> <s> :: c=0 core=(-1,0,0,0,0,0,0,0)  0core=
(-1,0,0,0,0,0,0,0)
Trans Opt 0 [1..1]: [1..1]=overhead  : OT ->OT  -> 头顶上的 :: c=-0.528683 core=
(-1,0,-0.869037,-0.965082,0,-0.223144,0.999896,0)  -0.528683core=
(-1,0,-0.869037,-0.965082,0,-0.223144,0.999896,0)
Trans Opt 0 [2..7]: [6..7]=NT   [5..5]=the   [4..4]=in   [3..3]=NT   [2..2]=NT  : NT ->NT  -> NT NT
NT :0-0 1-1 4-2 : c=0 core=(-0,0,0,0,0,0,0,0)  -1.81586core=(-4,0,0,-2.6264,0,-4.005,2.99969,0)
Trans Opt 0 [2..2]: [2..2]=oxygen  : NT ->NT  -> 氧气 :: c=-0.605286 core=
(-1,0,0,-0.875468,0,-1.335,0.999896,0)  -0.605286core=(-1,0,0,-0.875468,0,-1.335,0.999896,0)
Trans Opt 0 [3..3]: [3..3]=masks  : NT ->NT  -> 面罩 :: c=-0.605286 core=
(-1,0,0,-0.875468,0,-1.335,0.999896,0)  -0.605286core=(-1,0,0,-0.875468,0,-1.335,0.999896,0)
Trans Opt 0 [6..7]: [7..7]=section   [6..6]=cabin  : NT ->NT  -> 船舱 区 :: c=-0.605286 core=
(-2,0,0,-0.875468,0,-1.335,0.999896,0)  -0.605286core=(-2,0,0,-0.875468,0,-1.335,0.999896,0)
Trans Opt 0 [8..12]: [12..12]=.   [11..11]=place   [10..10]=into   [9..9]=dropped   [8..8]=had  : OT
>OT  -> 已滑落 。 :: c=-5.83601 core=(-2,0,-4.96981,-5.93035,0,-1.77176,0.999896,0)  -5.83601core=
(-2,0,-4.96981,-5.93035,0,-1.77176,0.999896,0)
```



==========

Build ZH->EN Tree to tree translation using Moses:

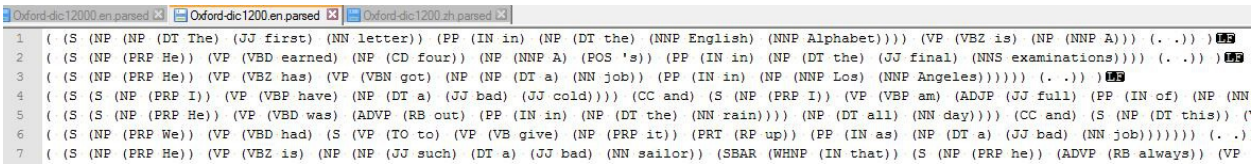
Prepare corpus:

The corpus “Oxford-dictionary” is extracted from Oxford Chinese-English dictionary (7th version), and segmented using NLPIR2013 segmentor (Zhang Huangping, CAS, accuracy 0.97).

Using the Oxford-dictionary.zh 12000 tokened simplified Chinese sentences, the GrammarTrained on CTB-7, BerkeleyParser-1.7.jar, the parsing information: 10 minutes.

Using the TrainedGramEng-WSJ, Berkeley-Parser-1.7.jar, Oxford-dictionary.zh 12000 tokened English sentences, the parsing: 11 minutes.

Firstly, we select the first 1200 sentences bilingual EN-ZH corpora for a trial. The parsed corpora as below:

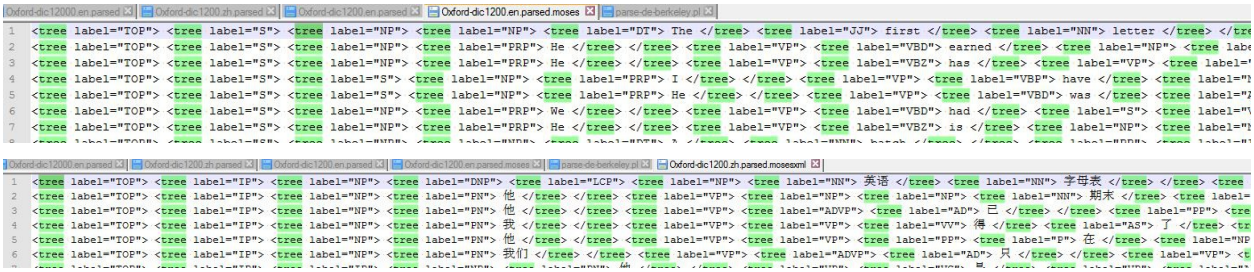


Convert the PENN parsed format into the Moses required tree format:

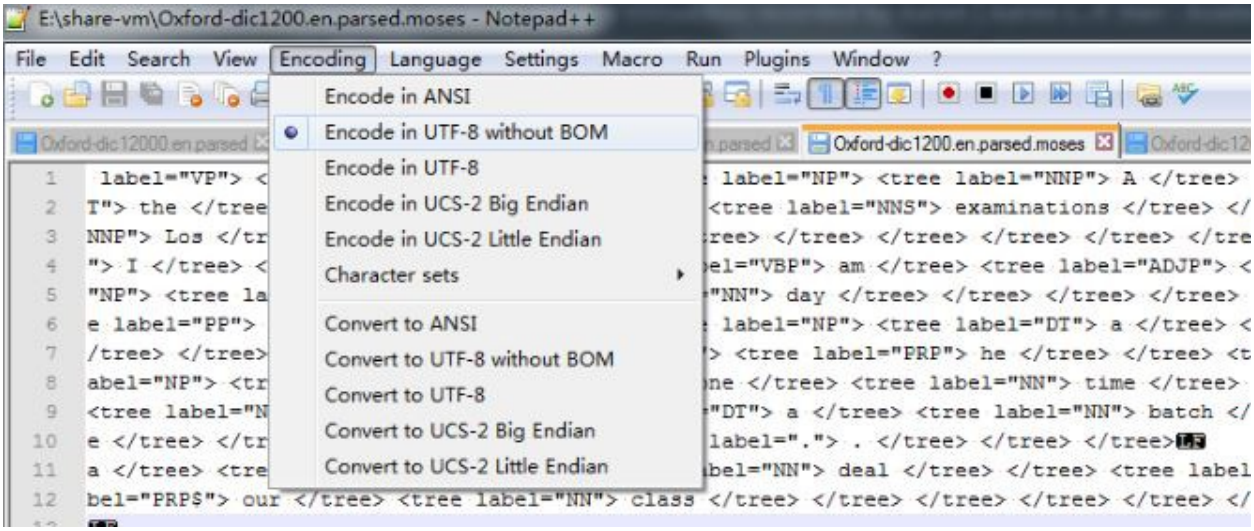
```
E:\share-vm>berkeleyparsed2mosesxml.pl < Oxford-dic1200.en.parsed > Oxford-dic12
00.en.parsed.mosesxml

E:\share-vm>
```

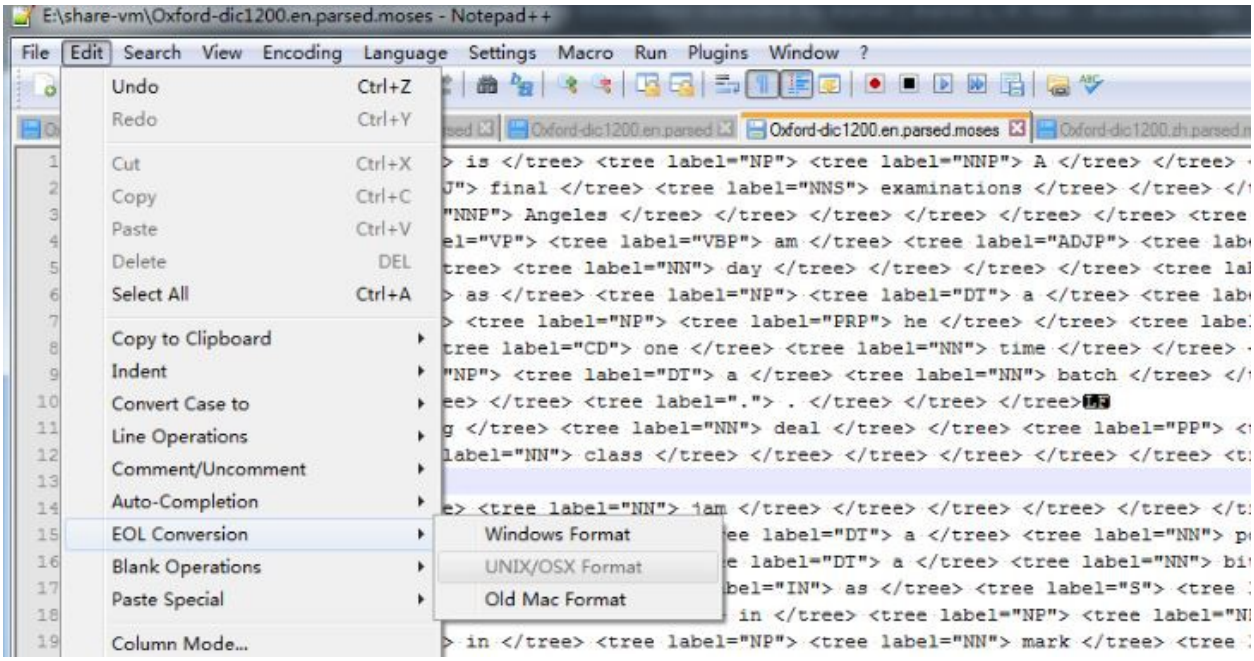
```
E:\share-vm>berkeleyparsed2mosesxml.pl < Oxford-dic1200.zh.parsed > Oxford-dic12
00.zh.parsed.mosesxml
```

After the converting, the format of the corpus becomes:



Encoding the content into TUF-8 format, edit content into UNIX format:





Training language model:

Put the converted documents “Oxford-dic1200.parsed.mosesxml.zh” and “Oxford-dic1200.parsed.mosesxml.en” into the VM directory: ~Aaron/corpus/training/

We first try to use the previously built language model “news-commentary-v8.fr-en.blm.en” from “news-commentary-v8.fr-en.tok.en”, WMT13 training corpus.

Training translation model:

Put the converted documents “Oxford-dic1200.parsed.mosesxml.zh” and “Oxford-dic1200.parsed.mosesxml.en” into the VM directory: ~Aaron/corpus/training/

Clear (empty) the directory ~Aaron/working/train/

Training command: (-f: source language; -e: target language)(without the reordering process in tree-to-tree translation training)

```
cd ~/Aaron
mkdir working
cd working
```

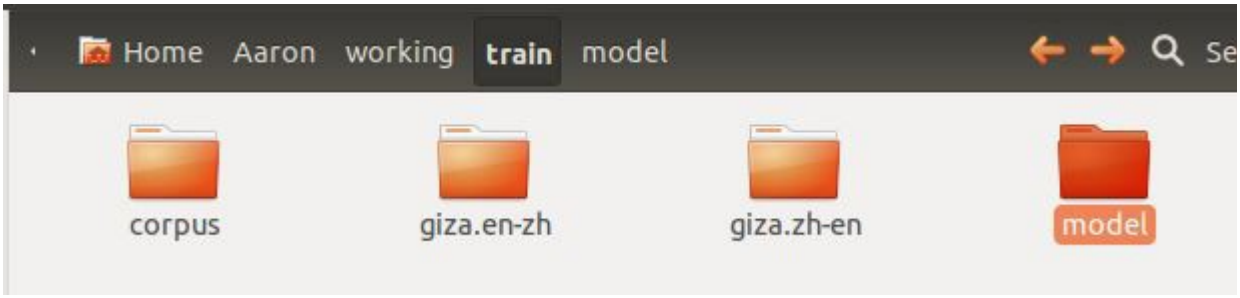
~/Aaron/Moses/mosesdecoder/scripts/training/train-model.perl -root-dir ./train -corpus ~/Aaron/corpus/training/Oxford-dic1200.parsed.mosesxml -f zh -e en -alignment grow-diag-final-and -hierarchical -glue-grammar --source-syntax --target-syntax -lm 0:3:$HOME/Aaron/lm/news-commentary-v8.fr-en.blm.en:8 -external-bin-dir ~/Aaron/Moses/mosesdecoder/tools >& training.out.Oxford.parse.zh-en1 &

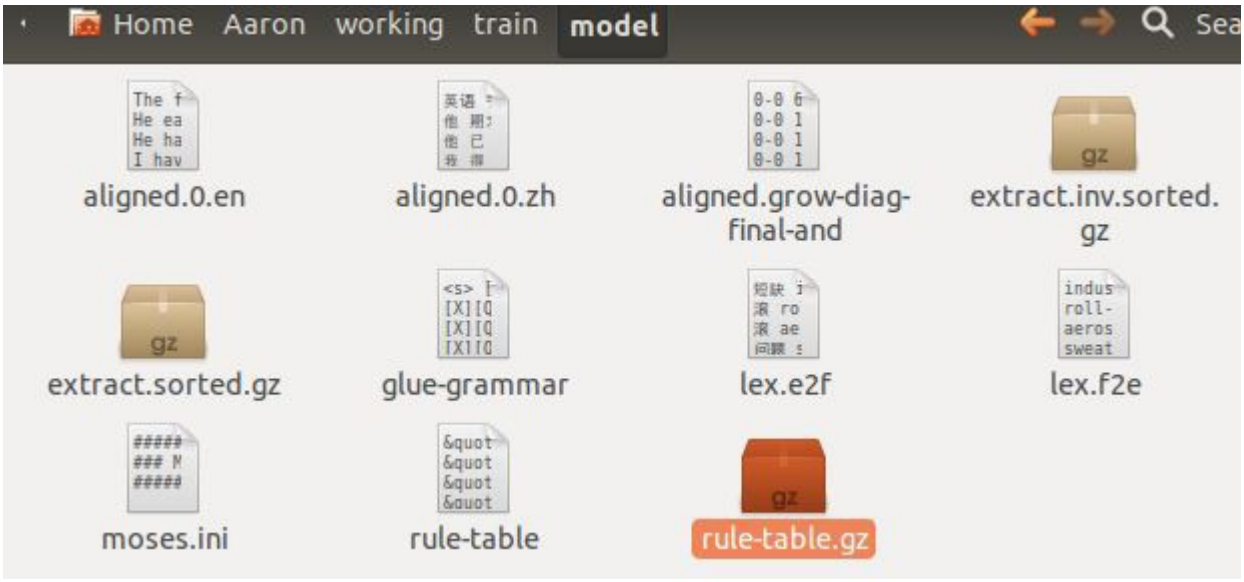
```
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$ ~/Aaron/Moses/mosesdecoder/scripts/tra
ining/train-model.perl -root-dir ./train -corpus ~/Aaron/corpus/training/Oxford-
dic1200.parsed.mosesxml -f zh -e en -alignment grow-diag-final-and -hierarchical
 -glue-grammar --source-syntax --target-syntax -lm 0:3:$HOME/Aaron/lm/news-comme
ntary-v8.fr-en.blm.en:8 -external-bin-dir ~/Aaron/Moses/mosesdecoder/tools >& tr
aining.out.Oxford.parse.zh-en1 &
```

The training will be finished from 2013.11.26-17:48 to 2013.11.26-17:49. Below files will be generated:

mosesxml2berkeleyparsed.perl | parse-en-collins.perl | rule-table | training.out.Oxford.parse.zh-en | training.out.Oxford.parse.zh-en1

```
/home/nlp2ct/Aaron/Moses/mosesdecoder/scripts/../bin/score /home/nlp2ct/Aaron/working/train/model/tmp.3774/extract.0.gz /home/nlp2ct/Aaron/working/
train/model/lex.f2e /home/nlp2ct/Aaron/working/train/model/tmp.3774/phrase-table.half.00000.gz --Hierarchical  2>> /dev/stderr
/home/nlp2ct/Aaron/working/train/model/tmp.3774/run.0.shmv /home/nlp2ct/Aaron/working/train/model/tmp.3774/phrase-table.half.00000.gz /home/nlp2ct/
Aaron/working/train/model/rule-table.half.f2e.gzrm -rf /home/nlp2ct/Aaron/working/train/model/tmp.3774
Finished Tue Nov 26 17:49:07 2013
(6.3)  creating table half /home/nlp2ct/Aaron/working/train/model/rule-table.half.e2f @ Tue Nov 26 17:49:07 CST 2013
/home/nlp2ct/Aaron/Moses/mosesdecoder/scripts/generic/score-parallel.perl 1 "sort    " /home/nlp2ct/Aaron/Moses/mosesdecoder/scripts/../bin/score /
home/nlp2ct/Aaron/working/train/model/extract.inv.sorted.gz /home/nlp2ct/Aaron/working/train/model/lex.e2f /home/nlp2ct/Aaron/working/train/model/
rule-table.half.e2f.gz --Inverse --Hierarchical 1
Executing: /home/nlp2ct/Aaron/Moses/mosesdecoder/scripts/generic/score-parallel.perl 1 "sort    " /home/nlp2ct/Aaron/Moses/mosesdecoder/scripts/../
bin/score /home/nlp2ct/Aaron/working/train/model/extract.inv.sorted.gz /home/nlp2ct/Aaron/working/train/model/lex.e2f /home/nlp2ct/Aaron/working/
train/model/rule-table.half.e2f.gz --Inverse --Hierarchical 1
Started Tue Nov 26 17:49:07 2013
ln -s /home/nlp2ct/Aaron/working/train/model/extract.inv.sorted.gz /home/nlp2ct/Aaron/working/train/model/tmp.3785/extract.0.gz
/home/nlp2ct/Aaron/Moses/mosesdecoder/scripts/../bin/score /home/nlp2ct/Aaron/working/train/model/tmp.3785/extract.0.gz /home/nlp2ct/Aaron/working/
train/model/lex.e2f /home/nlp2ct/Aaron/working/train/model/tmp.3785/phrase-table.half.00000.gz --Inverse --Hierarchical  2>> /dev/stderr
/home/nlp2ct/Aaron/working/train/model/tmp.3785/run.0.shgunzip -c /home/nlp2ct/Aaron/working/train/model/tmp.3785/phrase-table.half.*.gz 2>> /dev/
stderr| LC_ALL=C sort     -T /home/nlp2ct/Aaron/working/train/model/tmp.3785  | gzip -c > /home/nlp2ct/Aaron/working/train/model/rule-
table.half.e2f.gz  2>> /dev/stderr rm -rf /home/nlp2ct/Aaron/working/train/model/tmp.3785
Finished Tue Nov 26 17:49:07 2013
(6.6) consolidating the two halves @ Tue Nov 26 17:49:07 CST 2013
Executing: /home/nlp2ct/Aaron/Moses/mosesdecoder/scripts/../bin/consolidate /home/nlp2ct/Aaron/working/train/model/rule-table.half.f2e.gz /home/
nlp2ct/Aaron/working/train/model/rule-table.half.e2f.gz /dev/stdout --Hierarchical | gzip -c > /home/nlp2ct/Aaron/working/train/model/rule-table.gz
Consolidate v2.0 written by Philipp Koehn
consolidating direct and indirect rule tables
processing hierarchical rules
Executing: rm -f /home/nlp2ct/Aaron/working/train/model/rule-table.half.*
(7) learn reordering model @ Tue Nov 26 17:49:07 CST 2013
  ... skipping this step, reordering is not lexicalized ...
(8) learn generation model @ Tue Nov 26 17:49:07 CST 2013
  no generation model requested, skipping step
(9) create moses.ini @ Tue Nov 26 17:49:07 CST 2013
```





train-model.perl | parse-de-berkeley.perl | mosesxml2berkeleyparsed.perl | parse-en-collins.perl | rule-table

```
&quot; [NP][NP] &quot; [NP] ||| &quot; [NP][NP] &quot; [NP] ||| 0.42857 0.790123 1 0.790123 ||| 0-0 1-1 2-2 ||| 1.16667 0.5 0.5 |||
&quot; [PU] ||| &quot; [SYM] ||| 1 0.888889 0.666667 0.888889 ||| 0-0 ||| 4 6 4 |||
&quot; [PU] ||| &quot; [VB] ||| 1 0.888889 0.166667 0.888889 ||| 0-0 ||| 1 6 1 |||
&quot; [PU] ||| will [MD] ||| 0.0666667 0.0357143 0.166667 0.111111 ||| 0-0 ||| 15 6 1 |||
&quot; 我 [VP][VP] 。 [IP] ||| I will [VP][VP] . [S] ||| 0.5 0.0311261 0.5 0.0679012 ||| 0-1 1-0 2-2 3-3 ||| 0.5 0.5 0.25 |||
&quot; 我 [VP][VP] 。 [IP] ||| I will [VP][VP] . [TOP] ||| 0.5 0.0311261 0.5 0.0679012 ||| 0-1 1-0 2-2 3-3 ||| 0.5 0.5 0.25 |||
&quot; 我 [VP][VP] 。 [TOP] ||| I will [VP][VP] . [S] ||| 0.5 0.0311261 0.5 0.0679012 ||| 0-1 1-0 2-2 3-3 ||| 0.5 0.5 0.25 |||
&quot; 我 [VP][VP] 。 [TOP] ||| I will [VP][VP] . [TOP] ||| 0.5 0.0311261 0.5 0.0679012 ||| 0-1 1-0 2-2 3-3 ||| 0.5 0.5 0.25 |||
&quot; 玛蒂 小姐 &quot; [NP] ||| &quot; Miss Mattie &quot; [NP] ||| 0.6 0.790123 1 0.790123 ||| 0-0 1-1 2-2 3-3 ||| 0.833333 0.5 0.5 |||
-- [PU] ||| an afterthought [NP] ||| 1 0.257634 1 0.0816327 ||| 0-0 0-1 ||| 1 1 1 |||
. [PU] ||| . [.] ||| 0.00978648 0.0149385 1 0.586207 ||| 0-0 ||| 1124 11 11 |||
1000 [QP][CD] [QP] ||| up to $ 10 [QP][CD] [NP] ||| 1 0.513514 0.5 6.88481e-06 ||| 0-0 0-3 1-4 ||| 0.333333 0.666666 0.333333 |||
1000 [QP][CD] [QP] ||| up to $ 10 [QP][CD] [QP] ||| 1 0.513514 0.5 6.88481e-06 ||| 0-0 0-3 1-4 ||| 0.333333 0.666666 0.333333 |||
1000 万 元 [QP] ||| up to $ 10 million [QP] ||| 1 0.0962838 1 2.1515e-06 ||| 0-0 0-3 1-4 2-4 ||| 0.333333 0.333333 0.333333 |||
10% [CD] ||| I [PRP] ||| 0.00934579 0.0090909 1 1 ||| 0-0 ||| 107 1 1 |||
10% 的 [DNP] ||| I [PRP] ||| 0.00934579 0.00213274 1 1 ||| 0-0 ||| 107 1 1 |||
1988年 [NT] ||| 1988 [CD] ||| 1 1 1 1 ||| 0-0 ||| 1 1 1 |||
20820 人 [NP] ||| numbered 20820 [VP] ||| 1 0.375 1 0.127358 ||| 0-0 0-1 1-0 ||| 1 1 1 |||
980 [CD] ||| 703,980 [CD] ||| 1 1 1 1 ||| 0-0 ||| 1 1 1 |||
: [PU] ||| as [RB] ||| 0.2 0.0163934 1 0.5 ||| 0-0 ||| 5 1 1 |||
; [PU] ||| visit [NN] ||| 1 0.333333 1 1 ||| 0-0 ||| 1 1 1 |||
[ADVP][TO] 克服 [IP] ||| [ADVP][TO] crack [SBAR] ||| 0.181818 0.5 0.333333 1 ||| 0-0 1-1 ||| 0.785714 0.428571 0.142857 |||
[ADVP][TO] 克服 [IP] ||| [ADVP][TO] crack [S] ||| 0.181818 0.5 0.333333 1 ||| 0-0 1-1 ||| 0.785714 0.428571 0.142857 |||
[ADVP][TO] 克服 [IP] ||| [ADVP][TO] crack [VP] ||| 0.181818 0.5 0.333333 1 ||| 0-0 1-1 ||| 0.785714 0.428571 0.142857 |||
[ADVP][TO] 克服 [VP] ||| [ADVP][TO] crack [SBAR] ||| 0.181818 0.5 0.333333 1 ||| 0-0 1-1 ||| 0.785714 0.428571 0.142857 |||
```

Refer below command is from Tianliang:

```
Kickstarted 12:40 28-Mar-2012
[ma96572@pearl ~]$ nohup nice /home/ma96572/research/moses/scripts/training/trai
n-model.perl -cores 5 --parallel -root-dir /home/ma96572/Michelle/Syntax-based2/
 -external-bin-dir /home/ma96572/research/giza-pp/bin -corpus /home/ma96572/Mich
elle/Corpus/train5 -f en -e zh -alignment grow-diag-final-and -first-step 1 -las
t-step 9 -hierarchical -glue-grammar -lm 0:5:/home/ma96572/Michelle/LM/train5.bl
m.zh:0 >& /home/ma96572/Michelle/trace/out3  &
```

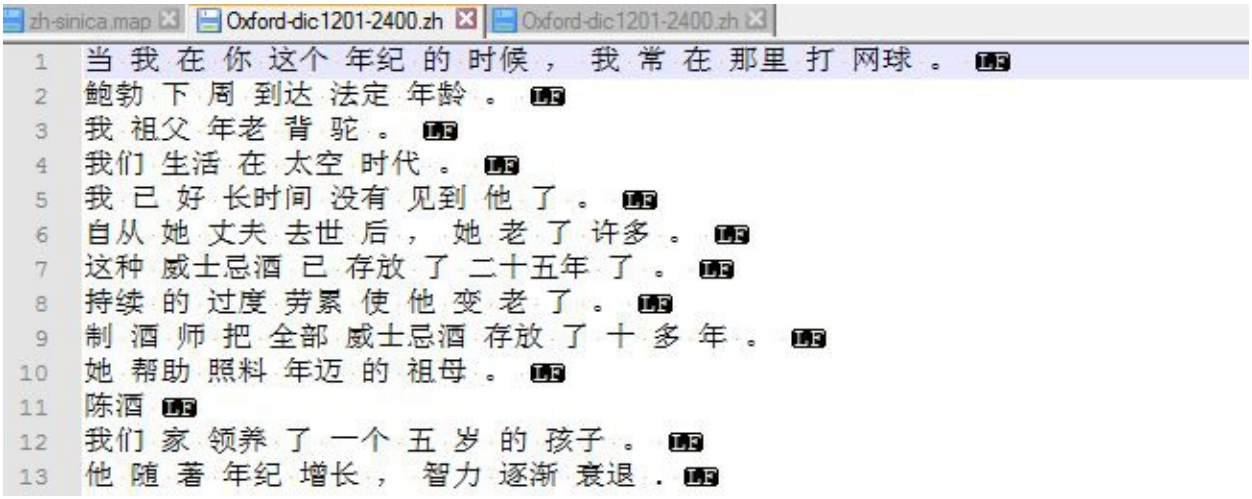
“/home/ma96572/research/moses/scripts/training/train-model.perl -cores 5 --parallel -root-dir /home/ma96572/Michelle/Syntax-based2/ -external-bin-dir /home/ma96572/research/giza-pp/bin -corpus /home/ma96572/Michelle/Corpus/train5 -f en -e zh -alignment grow-diag-final-and -first-step 1

-last-step 9 -hierarchical -glue-grammar -lm 0:5:/home/ma96572/Michelle/LM/train5.blm.zh:0 >& /home/ma96572/Michelle/trace/out3”

Testing:

We select the 1201th to 2400th Chinese sentences from the Oxford-dic120000.segmented EN-ZH bilingual corpus for testing.

Input testing document:



Using the testing command:

```
nlp2ct@nlp2ct-VirtualBox:~$ cd Aaron
nlp2ct@nlp2ct-VirtualBox:~/Aaron$ cd working/
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$ ~/Aaron/Moses/mosesdecoder/bin/moses_c
hart -f train/model/moses.ini < ~/Aaron/corpus/test/tree-to-tree/Oxford-dic1201-
2400.zh > Oxford-dic1201-2400.zh-en.ttt.out
```

The translation result:

moses.ini | in.xml | Oxford-dic1201-2400.zh-en.ttt.out | *record-moses-command.txt

```
when I in your The 年纪 of 时候 , I often had teases 网球 .
鲍勃 &apos;ll 到达 法定 年龄 .
I 祖父 年老 himself 驼 .
our life in 太空 时代 .
I .Brown good 长时间 no 见到 his abutting .
自从 adornments 丈夫 was after , adornments he abutting Many .
这种 威士忌酒 .Brown 存放 abutting 二十五年 abutting .
持续 of One 劳累 the his 变 he abutting .
制 a 师 to access 威士忌酒 存放 abutting 十 多 years .
her help 照料 年迈 of 祖母 .
陈酒
a 领养 abutting a 五 岁 of development .
his 随 著 年纪 增长 , 智力 逐渐 衰退 .
I ask 托尼 为什么 上班 迟到 . he let me his 碰到 abutting 倒霉 things , his mother The
```

Here, we change the testing format. Before the testing, we parse the input chinese sentences using the traindGrammarCTB-7, the input testing document:

The testing command:

```
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$ ~/Aaron/Moses/mosesdecoder/bin/moses_c
hart -f train/model/moses.ini < ~/Aaron/corpus/test/tree-to-tree/Oxford-dic1201-
2400.zh.parsed.mosesxml > Oxford-dic1201-2400.zh-en.parsed.ttt.out
```

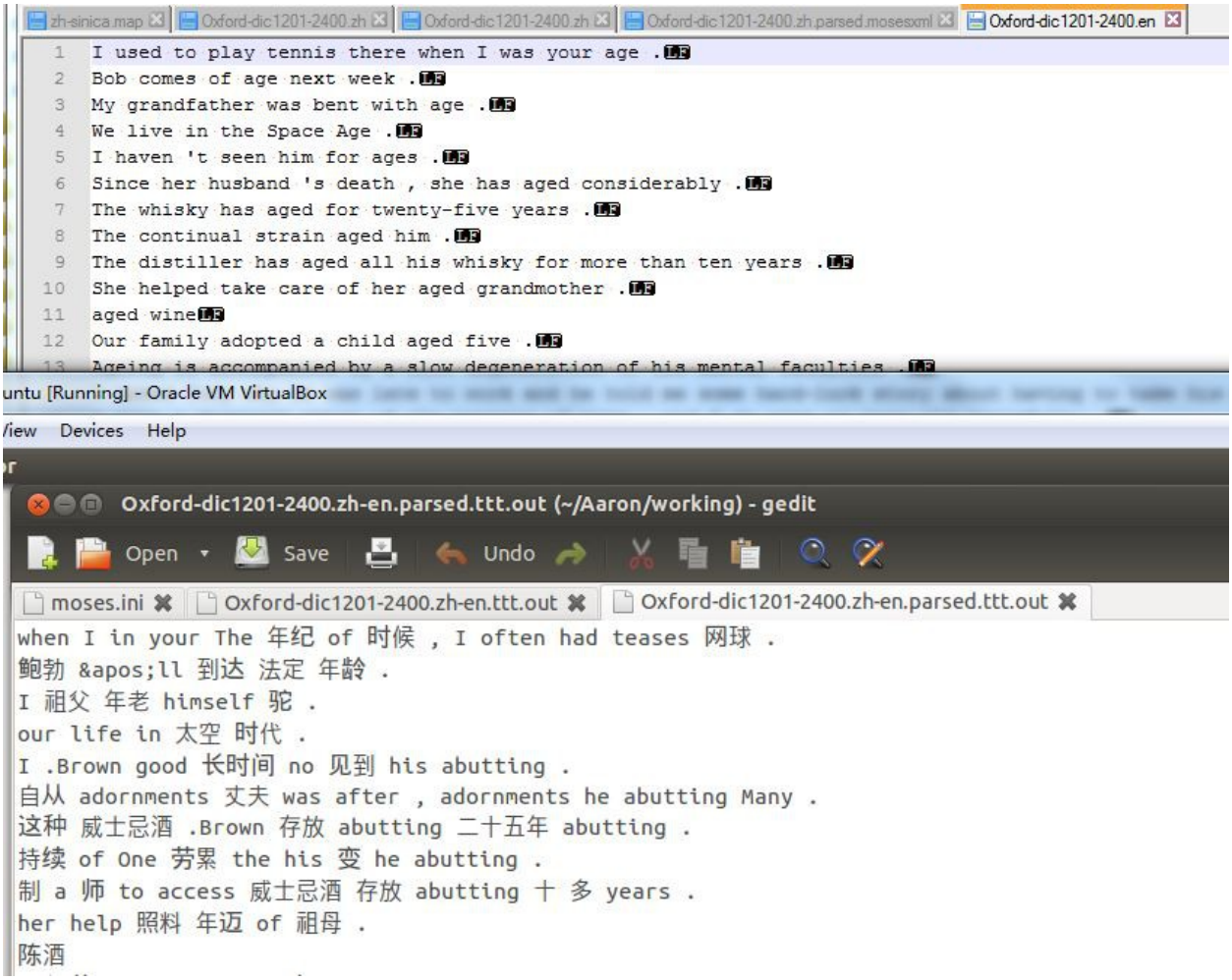
The testing translation result:

moses.ini | Oxford-dic1201-2400.zh-en.ttt.out | Oxford-dic1201-2400.zh-en.parsed.ttt.out

```
when I in your The 年纪 of 时候 , I often had teases 网球 .
鲍勃 &apos;ll 到达 法定 年龄 .
I 祖父 年老 himself 驼 .
our life in 太空 时代 .
I .Brown good 长时间 no 见到 his abutting .
自从 adornments 丈夫 was after , adornments he abutting Many .
这种 威士忌酒 .Brown 存放 abutting 二十五年 abutting .
持续 of One 劳累 the his 变 he abutting .
制 a 师 to access 威士忌酒 存放 abutting 十 多 years .
her help 照料 年迈 of 祖母 .
陈酒
a 领养 abutting a 五 岁 of development .
his 随 著 年纪 增长 , 智力 逐渐 衰退 .
I ask 托尼 为什么 上班 迟到 . he let me his 碰到 abutting 倒霉 things , his mother The 宝贝 of 长卷 毛
a gave microwave go to 兽医 .
```

The translation result using the parsed input sentence is the same with previous result (without parsing the input sentence).

The comparison of automatically translated sentences and references:



Preparing larger corpus-2

Using the Oxford-dictionary.zh 12000 tokened simplified Chinese sentences, the GrammarTrained on CTB-7, BerkeleyParser-1.7.jar, the parsing information: 10 minutes.

Using the TrainedGramEng-WSJ, Berkeley-Parser-1.7.jar, Oxford-dictionary.zh 12000 tokened English sentences, the parsing: 11 minutes.

10000 sentences (1st-10000th) for training, 1000 sentences (10001th-11000th) for developing, 1000 (11001th-12000th) sentences for testing. We skip the developing stage this time.

Put the files “Oxford-dic10000.parsed.mosesxml.en” and “Oxford-dic10000.parsed.mosesxml.zh” in the directory “~/Aaron/corpus/training/train10000Oxf”.

Training translation model-2:

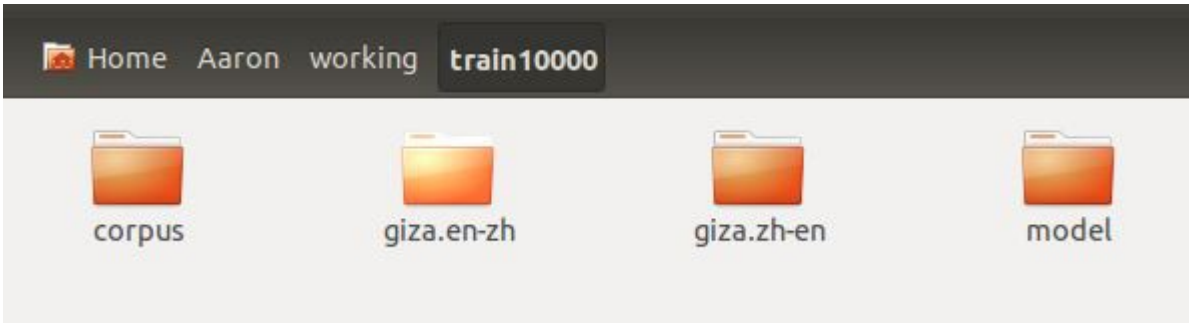
cd ~/Aaron
mkdir working
cd working
~/Aaron/Moses/mosesdecoder/scripts/training/train-model.perl -root-dir train10000 -corpus ~/Aaron/corpus/training/train10000Oxf/Oxford-dic10000.parsed.mosesxml -f zh -e en -alignment grow-diag-final-and -hierarchical -glue-grammar --source-syntax --target-syntax -lm 0:3:$HOME/Aaron/lm/news-commentary-v8.fr-en.blm.en:8 -external-bin-dir ~/Aaron/Moses/mosesdecoder/tools >& training.out.Oxford.parse.zh-en10000 &

```
nlp2ct@nlp2ct-VirtualBox:~$ cd Aaron
nlp2ct@nlp2ct-VirtualBox:~/Aaron$ cd working
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$ ~/Aaron/Moses/mosesdecoder/scripts/tra
ining/train-model.perl -root-dir train10000 -corpus ~/Aaron/corpus/training/trai
n10000Oxf/Oxford-dic10000.parsed.mosesxml -f zh -e en -alignment grow-diag-final
-and -hierarchical -glue-grammar --source-syntax --target-syntax -lm 0:3:$HOME/A
aron/lm/news-commentary-v8.fr-en.blm.en:8 -external-bin-dir ~/Aaron/Moses/mosesd
ecoder/tools >& training.out.Oxford.parse.zh-en10000 &
```

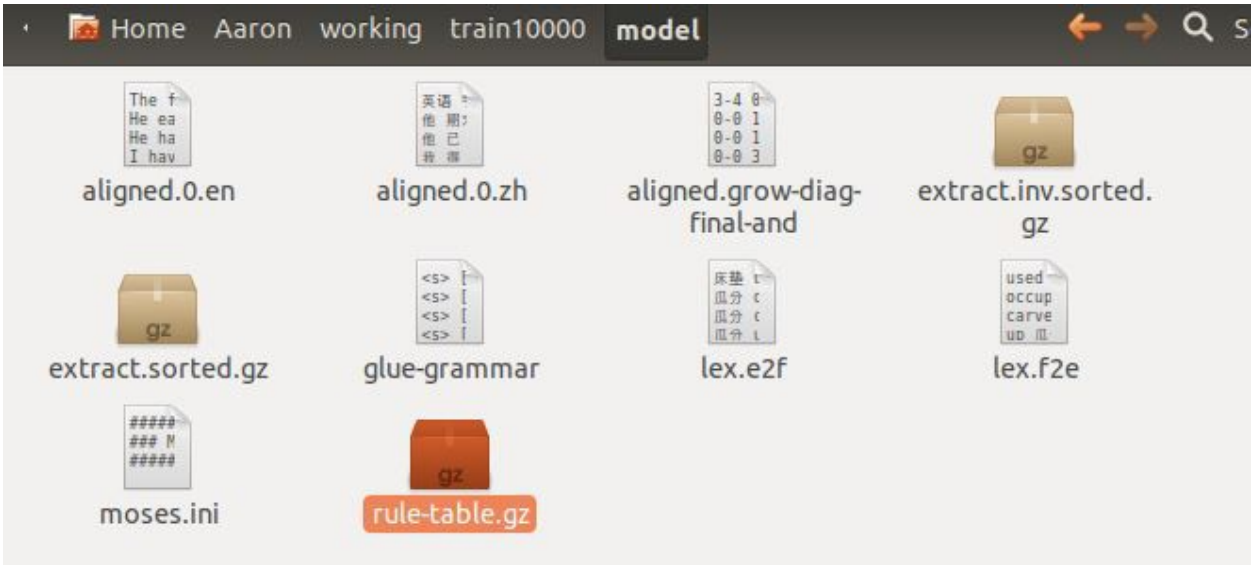
It runs as below:

```
nlp2ct@nlp2ct-VirtualBox: ~/Aaron/working
top - 13:42:28 up  2:54,  2 users,  load average: 0.36, 0.11, 0.07
Tasks: 153 total,   2 running, 151 sleeping,   0 stopped,   0 zombie
Cpu(s): 52.6%us,  3.1%sy,  0.0%ni, 44.3%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Mem:   6786596k total,  1138116k used,  5648480k free,    47352k buffers
Swap:  7336956k total,        0k used,  7336956k free,   450752k cached

  PID USER      PR  NI  VIRT  RES  SHR S %CPU %MEM    TIME+  COMMAND
 2370 nlp2ct    20   0 22356  10m 1396 R   49  0.2   0:01.47 mkcls
 1226 root      20   0  315m 138m  14m S    4  2.1   1:20.15 Xorg
 1656 nlp2ct    20   0 1484m  92m  36m S    2  1.4   1:30.53 compiz
 1679 nlp2ct     9 -11  354m 6228 3908 S    2  0.1   0:00.47 pulseaudio
 1690 nlp2ct    20   0 1227m  39m  20m S    1  0.6   0:11.99 nautilus
 1617 nlp2ct    20   0  109m 1568 1012 S    0  0.0   0:29.25 VBoxClient
 1783 nlp2ct    20   0  383m 9.8m 7336 S    0  0.1   0:00.66 bamfdaemon
 1792 nlp2ct    20   0  312m  10m 8408 S    0  0.2   0:00.96 gtk-window-deco
 1961 nlp2ct    20   0  407m 5952 4568 S    0  0.1   0:00.38 zeitgeist-datah
 2052 nlp2ct    20   0  307m   9m 7600 S    0  0.2   0:01.50 gnome-screensav
 2202 nlp2ct    20   0  511m  17m  11m S    0  0.3   0:01.29 gnome-terminal
 2329 nlp2ct    20   0  630m  25m  14m S    0  0.4   0:01.29 gedit
    1 root      20   0 24468 2412 1352 S    0  0.0   0:01.05 init
    2 root      20   0     0    0    0 S    0  0.0   0:00.00 kthreadd
    3 root      20   0     0    0    0 S    0  0.0   0:01.16 ksoftirqd/0
    5 root      20   0     0    0    0 S    0  0.0   0:00.80 kworker/u:0
    6 root      RT   0     0    0    0 S    0  0.0   0:00.01 migration/0
```

The training generates the files:



The rule-table.gz is 1.2MB large.



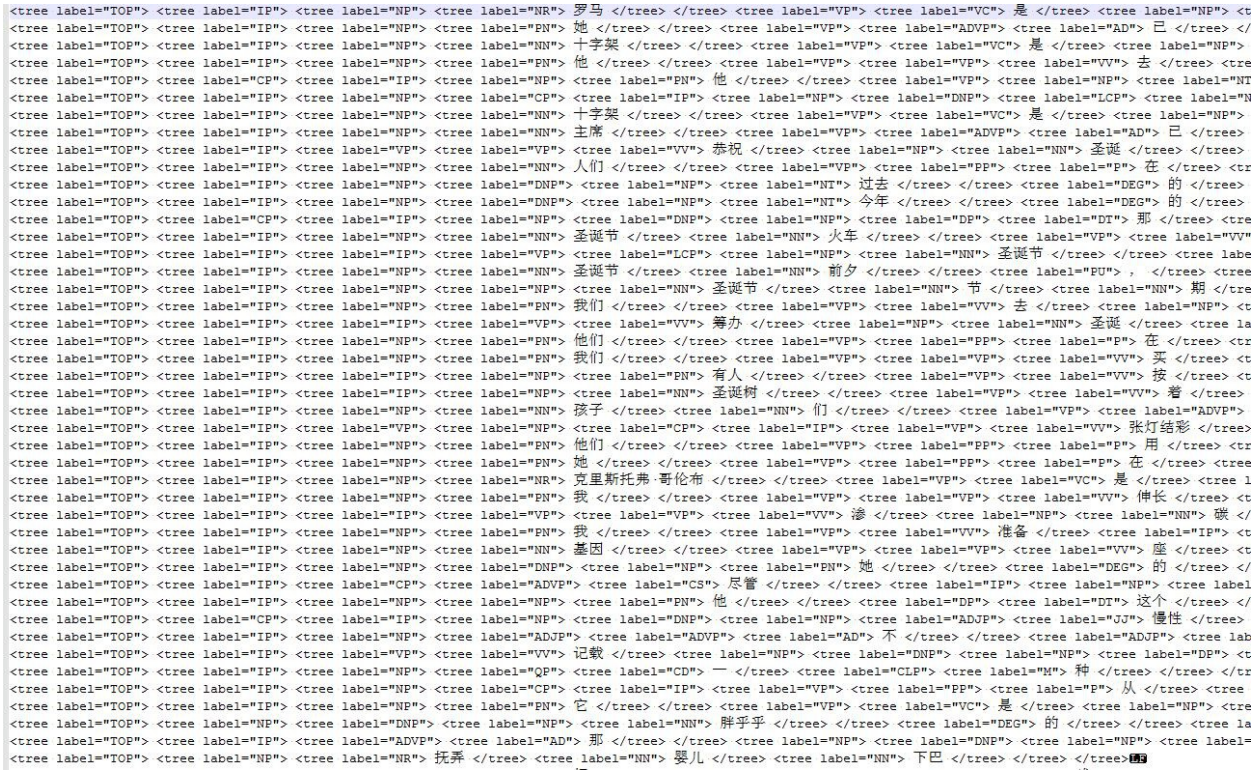
record-moses-commands02.txt ✖ rule-table ✖

```
! [PU] ||| ! [.] ||| 0.0375 0.198529 1 0.794118 ||| 0-0 ||| 80 3 3 |||
&amp;quot; A &amp;quot; [QP] ||| letter [NN] ||| 0.0833333 7.76749e-09 0.5 0.5 ||| 1-0 ||| 12 2 1 |||
&amp;quot; A &amp;quot; [QP] ||| the letter [NP] ||| 0.333333 7.76749e-09 0.5 0.0608267 ||| 1-1 ||| 3 2 1 |||
&amp;quot; [IP][TO] 即 付 &amp;quot; [IP] ||| [IP][TO] bearer on demand &amp;quot; [S] ||| 1 1.43341e-11 0.5 2.40744e-11
&amp;quot; [IP][TO] 即 付 &amp;quot; [IP] ||| [IP][TO] bearer on demand &amp;quot; [VP] ||| 1 1.43341e-11 0.5 2.40744e-11
&amp;quot; [PU] ||| &amp;quot; [NN] ||| 1 0.387755 0.0833333 0.44186 ||| 0-0 ||| 1 12 1 |||
&amp;quot; [PU] ||| &amp;quot; [SYM] ||| 0.470588 0.387755 0.666667 0.44186 ||| 0-0 ||| 17 12 8 |||
&amp;quot; [PU] ||| &amp;quot; [VB] ||| 1 0.387755 0.0833333 0.44186 ||| 0-0 ||| 1 12 1 |||
&amp;quot; [PU] ||| bearer on demand &amp;quot; [VP] ||| 1 0.387755 0.0833333 2.40744e-11 ||| 0-3 ||| 1 12 1 |||
&amp;quot; [PU] ||| notice [NN] ||| 1 0.166667 0.0833333 0.0232558 ||| 0-0 ||| 1 12 1 |||
&amp;quot; [VP][TO] 即 付 &amp;quot; [IP] ||| [VP][TO] bearer on demand &amp;quot; [S] ||| 1 1.43341e-11 0.5 2.40744e-11
&amp;quot; [VP][TO] 即 付 &amp;quot; [IP] ||| [VP][TO] bearer on demand &amp;quot; [VP] ||| 1 1.43341e-11 0.5 2.40744e-11
&amp;quot; p &amp;quot; [NP] ||| to [TO] ||| 0.00162075 1.12243e-10 1 1 ||| 1-0 ||| 617 1 1 |||
&amp;quot; 快 [IP] ||| Fast [IN] ||| 1 0.25 1 0.0229915 ||| 0-0 1-0 ||| 1 1 1 |||
&amp;quot; 我 [VP][NP] 。 [IP] ||| I will &amp;quot; is [VP][NP] . [S] ||| 0.5 0.320969 0.5 1.40163e-05 ||| 0-2 1-0 2
&amp;quot; 我 [VP][NP] 。 [IP] ||| I will &amp;quot; is [VP][NP] . [TOP] ||| 0.5 0.320969 0.5 1.40163e-05 ||| 0-2 1-0
```

Testing-2

We select the 11001 to 12000 sentences for testing.

Input testing document:



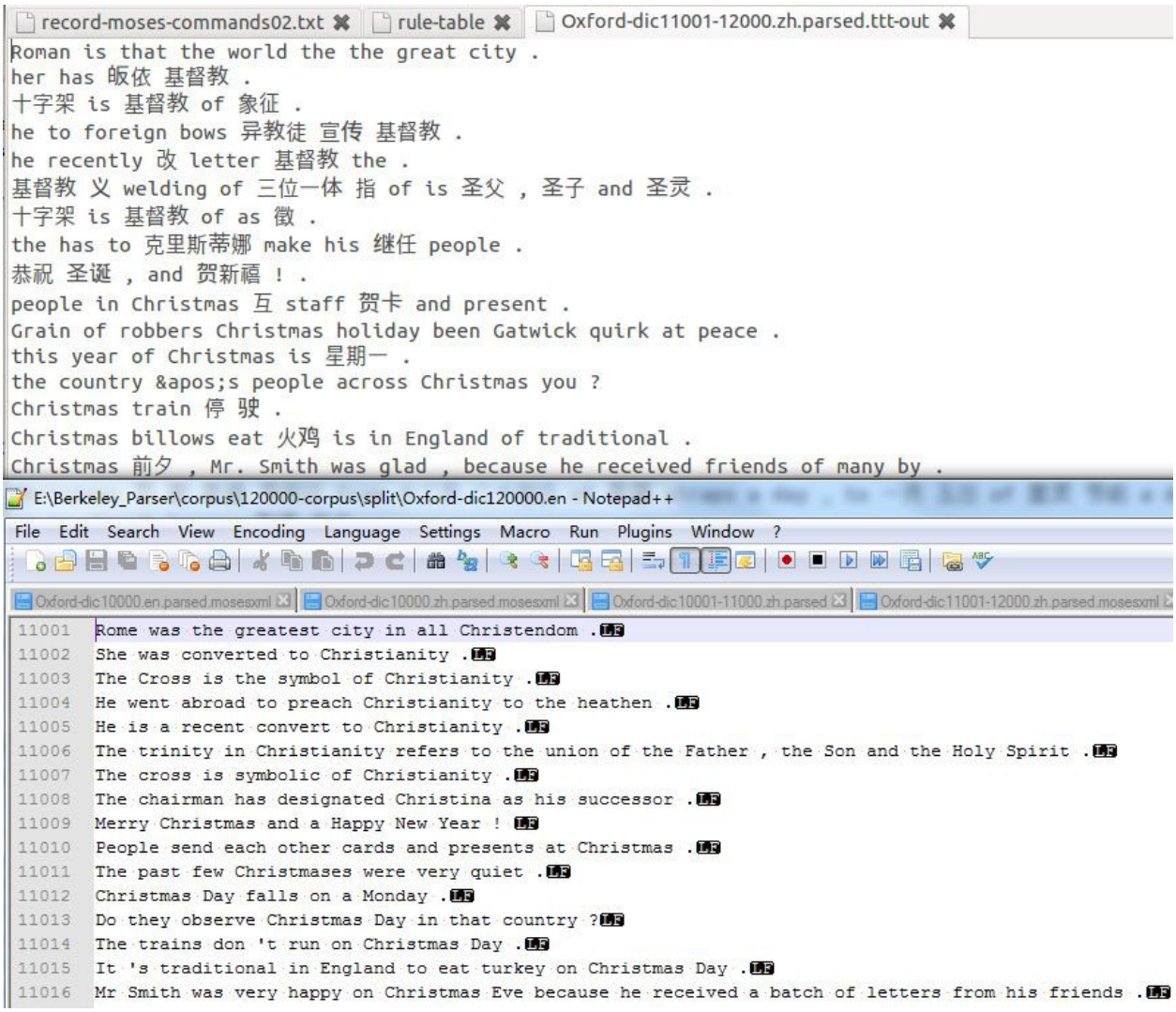
Using the testing command:

```
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$ ~/Aaron/Moses/mosesdecoder/bin/moses_c
hart -f train10000/model/moses.ini < ~/Aaron/corpus/test/tree-to-tree/Oxford-dic
11001-12000.zh.parsed.mosesxml > Oxford-dic11001-12000.zh.parsed.ttt-out
```

The translation result:

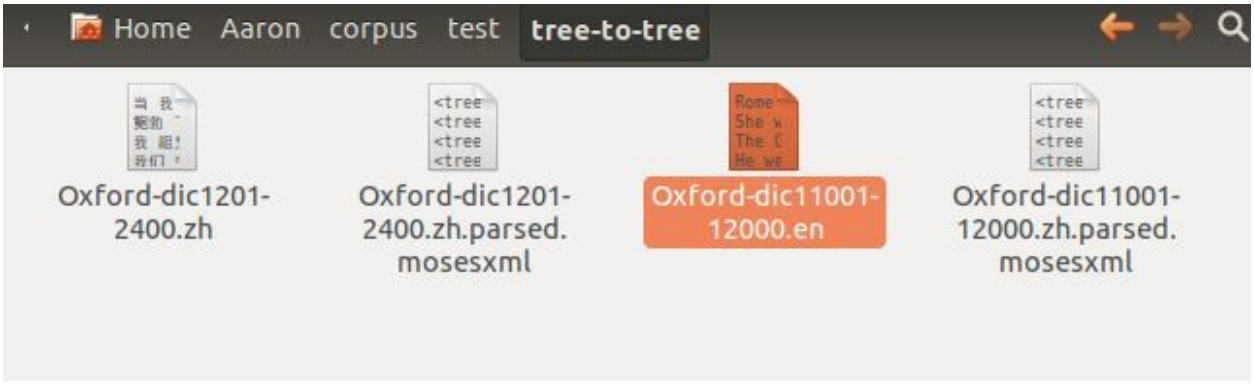
```
                                  1   0   0
                                    5   0
                                      1
BEST TRANSLATION: 170 Q  -> Q </s> :0-0 : c=-0.460 core=(0.000,-1.000,1.000,0.00
0,0.000,0.000,0.000,0.000,0.000)  [0..18] 168 [total=-322.267] core=(-300.000,-1
8.000,34.000,-20.515,-30.775,-8.093,-17.636,15.998,-95.322)
Translation took 0.030 seconds
End. : [19.000] seconds
Name:moses_chart        VmPeak:236388 kB        VmRSS:98156 kB  RSSMax:99492 kBu
ser:1.976       sys:8.405       CPU:10.381      real:19.320
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$
```

The comparison of automatical translation and reference sentences:



Calculate the testing score using BLEU:

Put the reference document:



Using the following command, we calculate the BLEU score of the 1000 translated sentences:

```
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$ ~/Aaron/Moses/mosesdecoder/scripts/gen
eric/multi-bleu.perl -lc ~/Aaron/corpus/test/tree-to-tree/Oxford-dic11001-12000.
en < Oxford-dic11001-12000.zh.parsed.ttt-out
```

```
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$ ~/Aaron/Moses/mosesdecoder/scripts/gen
eric/multi-bleu.perl -lc ~/Aaron/corpus/test/tree-to-tree/Oxford-dic11001-12000.
en < Oxford-dic11001-12000.zh.parsed.ttt-out
BLEU = 4.07, 36.2/6.7/1.7/0.6 (BP=1.000, ratio=1.005, hyp_len=9951, ref_len=9905
)
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$
```

Preparing larger corpus-2-Uni-phrase-tags

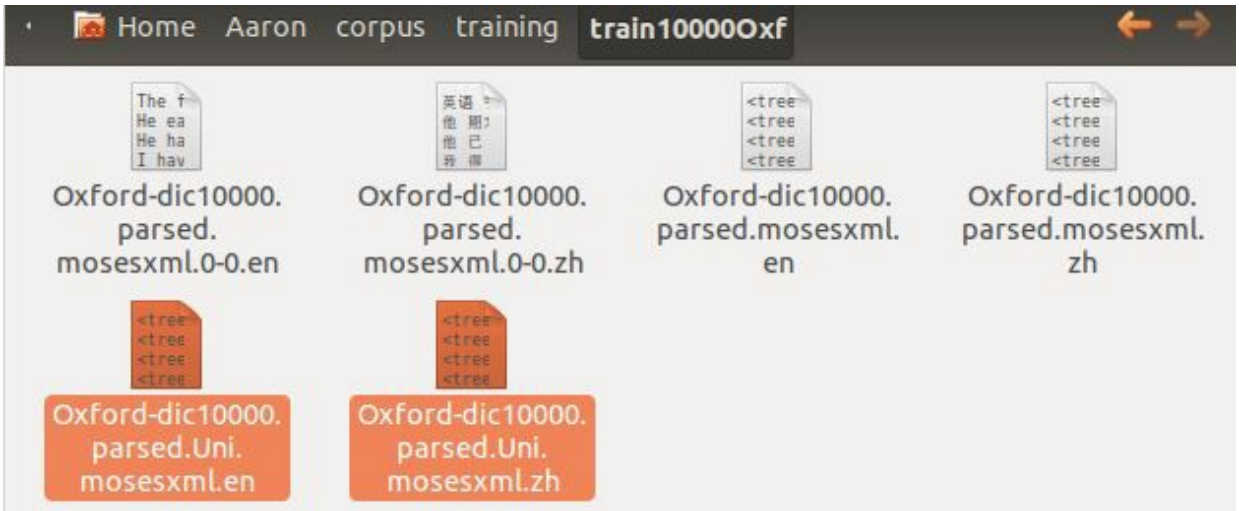
Using the Oxford-dictionary.zh 12000 tokened simplified Chinese sentences, the GrammarTrained on CTB-7, BerkeleyParser-1.7.jar, the parsing information: 10 minutes.

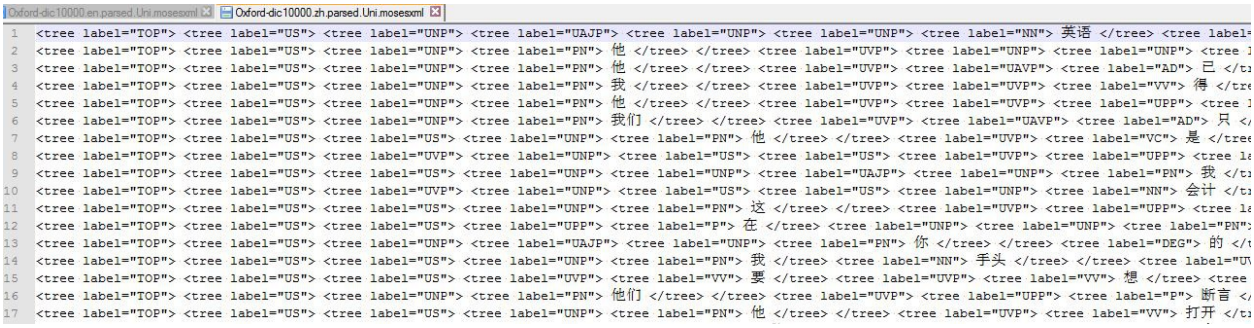
Using the TrainedGramEng-WSJ, Berkeley-Parser-1.7.jar, Oxford-dictionary.zh 12000 tokened English sentences, the parsing: 11 minutes.

Replace the phrase tagset in parsed 12000 bilingual EN-ZH sentences using the universal phrase tagset.

10000 sentences (1st-10000th) for training, 1000 sentences (10001th-11000th) for developing, 1000 (11001th-12000th) sentences for testing. We skip the developing stage this time.

Put the files “Oxford-dic10000.parsed.Uni.mosesxml.zh” and “Oxford-dic10000.parsed.Uni.mosesxml.en” in the directory “~/Aaron/corpus/training/train10000Oxf”.





Training translation model-2-Uni-phrase-tags

Using the following command to train the translation model:

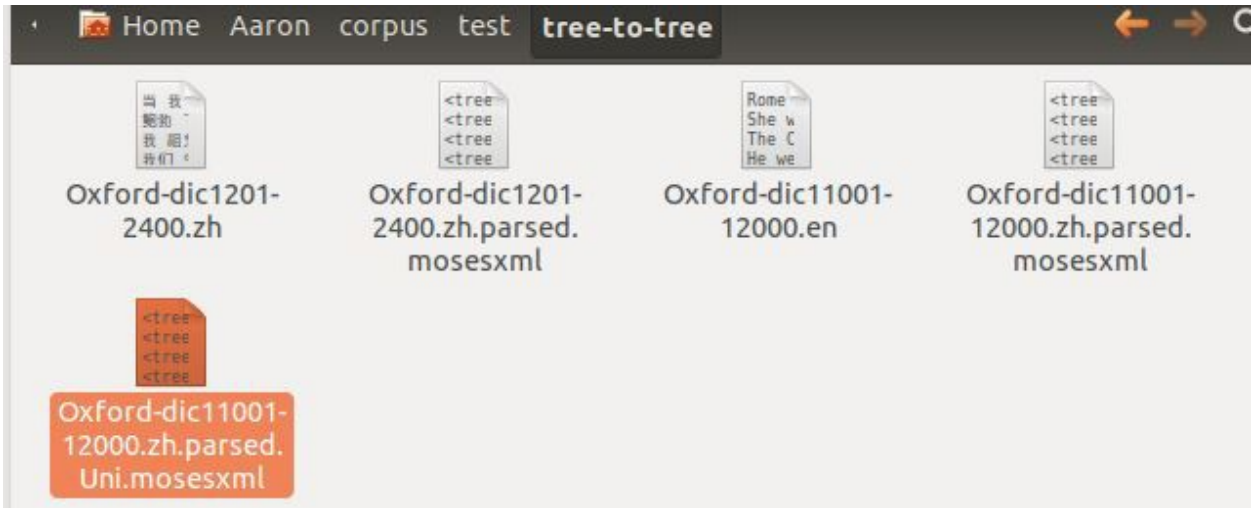
```
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$ ~/Aaron/Moses/mosesdecoder/scripts/tra
ining/train-model.perl -root-dir train10000Uni -corpus ~/Aaron/corpus/training/t
rain10000Oxf/Oxford-dic10000.parsed.Uni.mosesxml -f zh -e en -alignment grow-dia
g-final-and -hierarchical -glue-grammar --source-syntax --target-syntax -lm 0:3:
$HOME/Aaron/lm/news-commentary-v8.fr-en.blm.en:8 -external-bin-dir ~/Aaron/Moses
/mosesdecoder/tools >& training.out.Oxford.parse.Uni.zh-en10000 &
[1] 2628
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$ top
top - 15:59:01 up  5:11,  2 users,  load average: 0.45, 0.13, 0.08
Tasks: 153 total,   2 running, 151 sleeping,   0 stopped,   0 zombie
Cpu(s): 51.3%us,  0.2%sy,  0.0%ni, 48.6%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Mem:   6786596k total,  1309460k used,  5477136k free,    49668k buffers
Swap:  7336956k total,        0k used,  7336956k free,   559804k cached

  PID USER      PR  NI  VIRT  RES  SHR S %CPU %MEM    TIME+  COMMAND
 2660 nlp2ct    20   0 49208  36m 1968 R  100  0.5   0:08.94 GIZA++
 1226 root      20   0  319m 141m  14m S    1  2.1   2:23.45 Xorg
 1617 nlp2ct    20   0  109m 1568 1012 S    1  0.0   0:51.01 VBoxClient
 2202 nlp2ct    20   0  528m  23m  15m S    1  0.4   0:19.54 gnome-terminal
```

It finished as:

```
   15 root       0 -20     0    0    0 S    0  0.0   0:00.00 netns
[1]+  Done                    ~/Aaron/Moses/mosesdecoder/scripts/training/train-
model.perl -root-dir train10000Uni -corpus ~/Aaron/corpus/training/train10000Oxf
/Oxford-dic10000.parsed.Uni.mosesxml -f zh -e en -alignment grow-diag-final-and
-hierarchical -glue-grammar --source-syntax --target-syntax -lm 0:3:$HOME/Aaron/
lm/news-commentary-v8.fr-en.blm.en:8 -external-bin-dir ~/Aaron/Moses/mosesdecode
r/tools &>training.out.Oxford.parse.Uni.zh-en10000
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$
```

Testing-2-Uni-phrase-tags

Put the testing document:

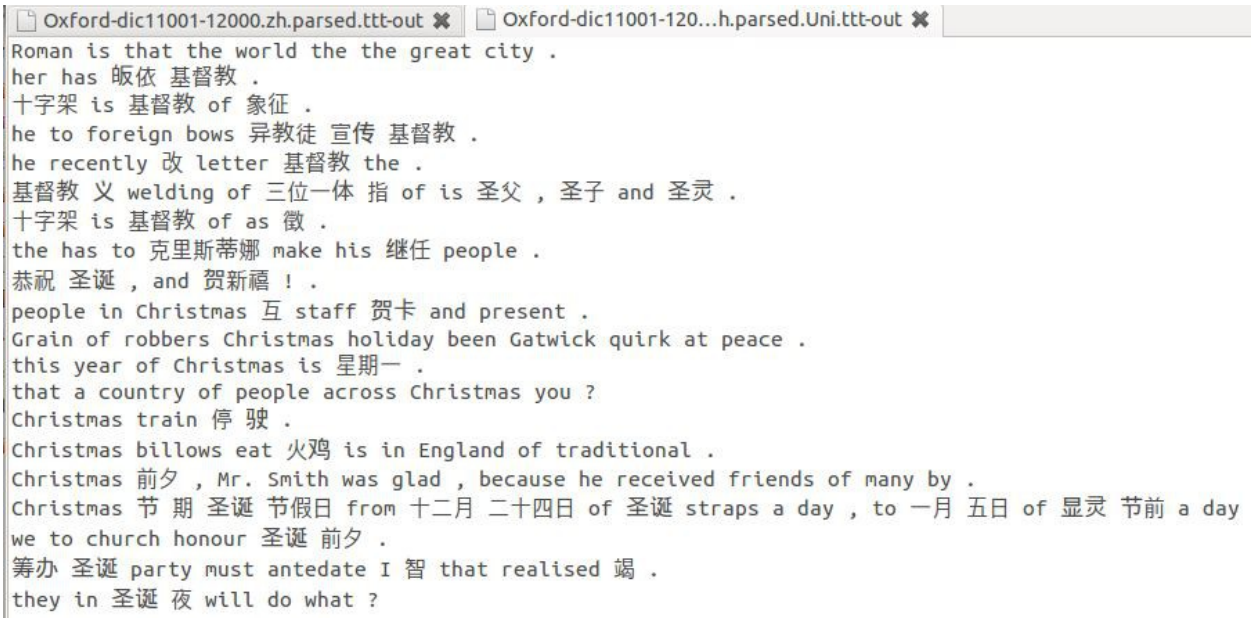


Using the following command for testing:

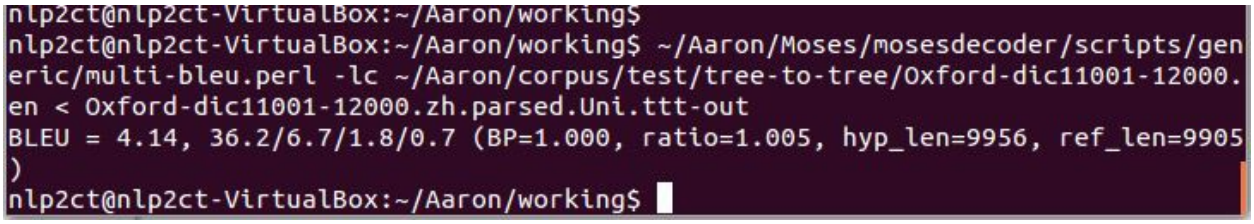
```
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$ ~/Aaron/Moses/mosesdecoder/bin/moses_c
hart -f train10000Uni/model/moses.ini < ~/Aaron/corpus/test/tree-to-tree/Oxford-
dic11001-12000.zh.parsed.Uni.mosesxml > Oxford-dic11001-12000.zh.parsed.Uni.ttt-
out
```

It finished as:

```
   15 root       0 -20     0    0    0 S    0  0.0   0:00.00 netns
[1]+  Done                    ~/Aaron/Moses/mosesdecoder/scripts/training/train-
model.perl -root-dir train10000Uni -corpus ~/Aaron/corpus/training/train10000Oxf
/Oxford-dic10000.parsed.Uni.mosesxml -f zh -e en -alignment grow-diag-final-and
-hierarchical -glue-grammar --source-syntax --target-syntax -lm 0:3:$HOME/Aaron/
lm/news-commentary-v8.fr-en.blm.en:8 -external-bin-dir ~/Aaron/Moses/mosesdecode
r/tools &>training.out.Oxford.parse.Uni.zh-en10000
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$
```



Calculate the testing score using BLEU:



Preparing larger corpus-3

Using the Oxford-dictionary.zh 120000 to-kened simplified Chinese sentences, the GrammarTrained on CTB-7, BerkeleyParser-1.7.jar, the parsing information: 3h+12 minutes.

Using the TrainedGramEng-WSJ, Berkeley-Parser-1.7.jar, Oxford-dictionary.en 120000 tokened English sentences, the parsing: 58 minutes.

100000 sentences (1st-100000th) for training, 10000 sentences (100001th-110000th) for developing, 10000 (110001th-120000th) sentences for testing. We skip the developing stage this time.

Put the files “Oxford-dic100000.parsed.mosesxml.en” and “Oxford-dic100000.parsed.mosesxml.zh” in the VM directory “~/Aaron/corpus/training/train100000Oxf”.

Training translation model-3

Using the following command the training began at 2013.12.01-19:12:04 and finished at

```
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$ ~/Aaron/Moses/mosesdecoder/scripts/tra
ining/train-model.perl -root-dir train100000 -corpus ~/Aaron/corpus/training/tra
in100000Oxf/Oxford-dic100000.parsed.mosesxml -f zh -e en -alignment grow-diag-fi
nal-and -hierarchical -glue-grammar --source-syntax --target-syntax -lm 0:3:$HOM
E/Aaron/lm/news-commentary-v8.fr-en.blm.en:8 -external-bin-dir ~/Aaron/Moses/mos
esdecoder/tools >& training.out.Oxford.parse.zh-en100000 &
[1] 2493
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$ top

top - 19:13:06 up 22 min,  2 users,  load average: 0.74, 0.31, 0.15
Tasks: 154 total,   2 running, 152 sleeping,   0 stopped,   0 zombie
Cpu(s): 50.2%us,  0.2%sy,  0.0%ni, 49.7%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Mem:   6786596k total,  1320200k used,  5466396k free,   125504k buffers
Swap:  7336956k total,        0k used,  7336956k free,   611940k cached

  PID USER      PR  NI  VIRT  RES  SHR S %CPU %MEM    TIME+  COMMAND
 2514 nlp2ct    20   0 69124  56m 1464 R  100  0.8   0:53.61 mkcls
  590 syslog    20   0  243m 1512 1128 S    0  0.0   0:00.11 rsyslogd
 2511 nlp2ct    20   0 17340 1340  976 R    0  0.0   0:00.06 top
    1 root      20   0 24468 2368 1352 S    0  0.0   0:01.26 init
```

Exit with the error report:

```
training.out.Oxford.parse.zh-en100000 ✖  Oxford-dic100000.parsed.mosesxml.0-0.zh ✖
Model1: (2) TRAIN CROSS-ENTROPY 6.93947 PERPLEXITY 122.741
Model1: (2) VITERBI TRAIN CROSS-ENTROPY 8.43846 PERPLEXITY 346.92
Model 1 Iteration: 2 took: 3 seconds
-----------
Model1: Iteration 3
Reading more sentence pairs into memory ...
ERROR: Forbidden zero sentence length 0
WARNING: The following sentence pair has source/target sentence length ration more than
the maximum allowed limit for a source word fertility
 source length = 0 target length = 8 ratio inf ferility limit : 9
Shortening sentence
Sent No: 31150 , No. Occurrences: 1
0
13 84 285 427 113 92 877 26
ERROR: Forbidden zero sentence length 0
WARNING: The following sentence pair has source/target sentence length ration more than
the maximum allowed limit for a source word fertility
 source length = 0 target length = 5 ratio inf ferility limit : 9
Shortening sentence
Sent No: 31174 , No. Occurrences: 1
0
14 72 53 6 26
ERROR: Forbidden zero sentence length 0
WARNING: The following sentence pair has source/target sentence length ration more than
the maximum allowed limit for a source word fertility
 source length = 0 target length = 8 ratio inf ferility limit : 9
Shortening sentence
Sent No: 43164 , No. Occurrences: 1
```

So, we should first do the “cleaning of the corpus” task this time to delete too long sentences and empty sentences, etc. due to the larger corpus.

After checking, we find the problem:

The original segmented 31174th sentence pair have no problem:

31174th – EN: How old are you ?

31174th – CN: 你 多 大 了 ？

After the parsing there is problem:

31174th – EN: “empty”

31174th – CN: <tree label="TOP"> <tree label="CP"> <tree label="IP"> <tree label="NP"> <tree label="PN"> 你 </tree> </tree> <tree label="VP"> <tree label="ADVP"> <tree label="AD"> 多 </tree> </tree> <tree label="VP"> <tree label="VA"> 大 </tree> </tree> </tree> </tree> <tree label="SP"> 了 </tree> <tree label="PU"> ？ </tree> </tree> </tree>

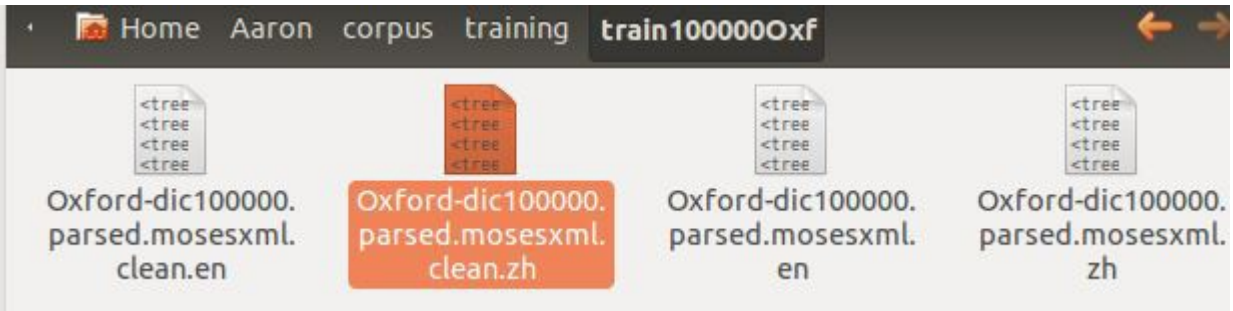
The 31173rd and 31175th parsed sentence pairs have no problems.

Cleaning (to remove the mis-aligned sentencs, long sentences and empty sentences, which may cause problems with the training pipeline)

Type the following command to delete the sentences whose length is larger than 80 (too long) or less than 1 (empty):

```
nlp2ct@nlp2ct-VirtualBox:~$
nlp2ct@nlp2ct-VirtualBox:~$ ~/Aaron/Moses/mosesdecoder/scripts/training/clean-co
rpus-n.perl ~/Aaron/corpus/training/train100000Oxf/Oxford-dic100000.parsed.moses
xml zh en ~/Aaron/corpus/training/train100000Oxf/Oxford-dic100000.parsed.mosesxm
l.clean 1 80
clean-corpus.perl: processing /home/nlp2ct/Aaron/corpus/training/train100000Oxf/
Oxford-dic100000.parsed.mosesxml.zh & .en to /home/nlp2ct/Aaron/corpus/training/
train100000Oxf/Oxford-dic100000.parsed.mosesxml.clean, cutoff 1-80
.........(100000)
Input sentences: 100000  Output sentences:  58202
nlp2ct@nlp2ct-VirtualBox:~$
```

The generated cleaned files:



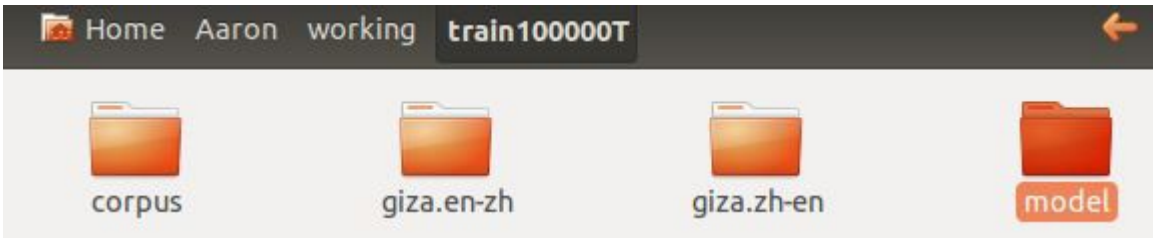
Using the following command the training once more began at 2013.12.02-10:28:49 and finished at 10:33:00.

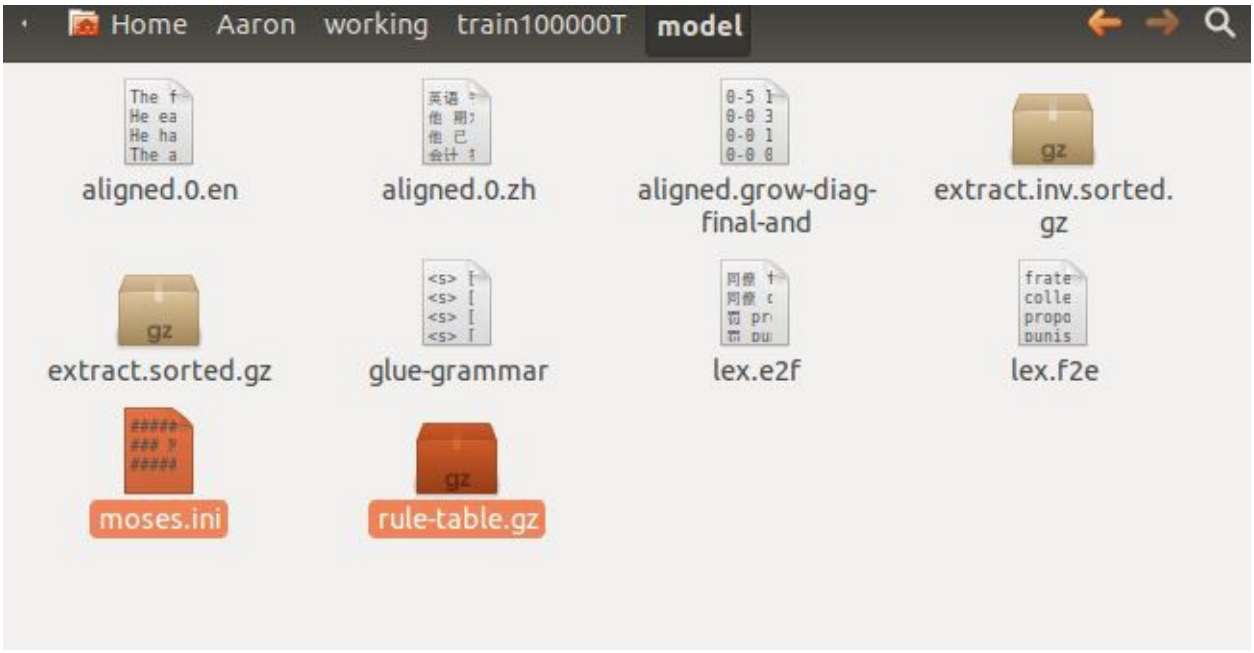
```
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$ ~/Aaron/Moses/mosesdecoder/scripts/tra
ining/train-model.perl -root-dir train100000T -corpus ~/Aaron/corpus/training/tr
ain100000Oxf/Oxford-dic100000.parsed.mosesxml.clean -f zh -e en -alignment grow-
diag-final-and -hierarchical -glue-grammar --source-syntax --target-syntax -lm 0
:3:$HOME/Aaron/lm/news-commentary-v8.fr-en.blm.en:8 -external-bin-dir ~/Aaron/Mo
ses/mosesdecoder/tools >& training.out.Oxford.parse.zh-en100000T &
[1] 2303
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$ top

top - 10:29:10 up 25 min,  2 users,  load average: 0.29, 0.09, 0.07
Tasks: 157 total,   3 running, 154 sleeping,   0 stopped,   0 zombie
Cpu(s): 51.2%us,  1.8%sy,  0.0%ni, 47.0%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Mem:   6786596k total,  1215172k used,  5571424k free,    45392k buffers
Swap:  7336956k total,        0k used,  7336956k free,   586404k cached

  PID USER      PR  NI  VIRT  RES  SHR S %CPU %MEM    TIME+  COMMAND
 2323 nlp2ct    20   0 39376  27m 1396 R  100  0.4   0:16.95 mkcls
 1160 root      20   0  293m 115m  14m S    2  1.7   0:24.88 Xorg
 1677 nlp2ct    20   0 1199m  90m  35m S    1  1.4   0:17.88 compiz
 1701 nlp2ct    20   0 1035m  38m  20m S    1  0.6   0:07.02 nautilus
```

```
Swap:  7336956k total,        0k used,  7336956k free,   647560k cached

  PID USER      PR  NI  VIRT  RES  SHR S %CPU %MEM    TIME+  COMMAND
 2399 nlp2ct    20   0 19352 2508 1548 R  100  0.0   0:10.24 extract-rules
 1160 root      20   0  293m 115m  14m S    2  1.7   0:25.96 Xorg
 2097 nlp2ct    20   0  511m  16m  10m S    1  0.3   0:03.36 gnome-terminal
 1640 nlp2ct    20   0  109m 1564 1012 S    0  0.0   0:04.27 VBoxClient
```

Within several minutes, training finish as:

```
   12 root       0 -20     0     0     0 S     0  0.0   0:00.00 cpuset
[1]+  Done                    ~/Aaron/Moses/mosesdecoder/scripts/training/train-
model.perl -root-dir train100000T -corpus ~/Aaron/corpus/training/train100000Oxf
/Oxford-dic100000.parsed.mosesxml.clean -f zh -e en -alignment grow-diag-final-a
nd -hierarchical -glue-grammar --source-syntax --target-syntax -lm 0:3:$HOME/Aar
on/lm/news-commentary-v8.fr-en.blm.en:8 -external-bin-dir ~/Aaron/Moses/mosesdec
oder/tools &>training.out.Oxford.parse.zh-en100000T
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$
```

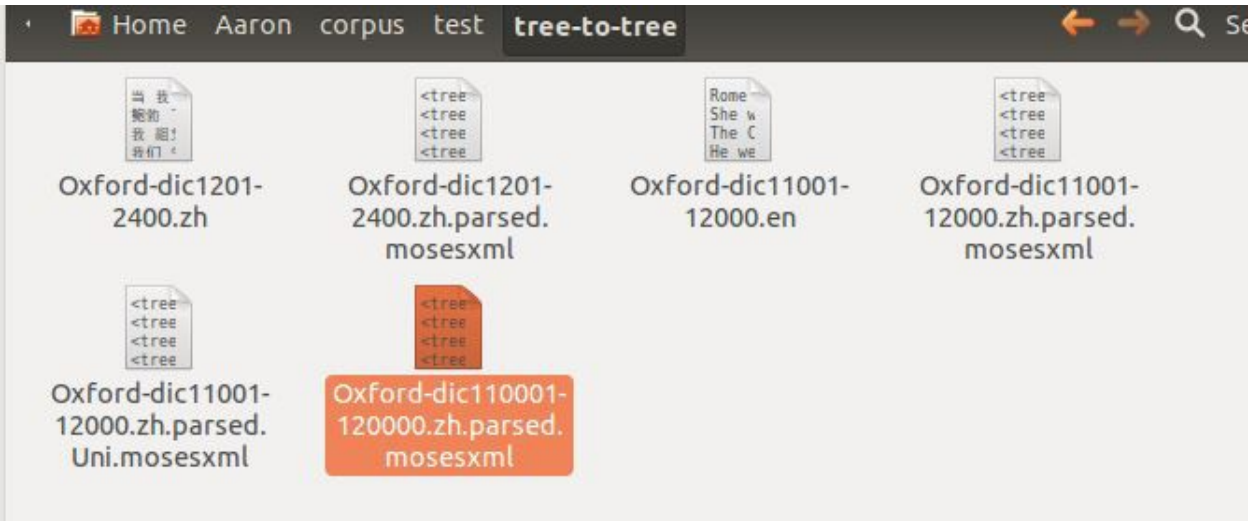




The generated rule-table.gz is 5.2MB volume.

Testing-3:

Put the testing file:

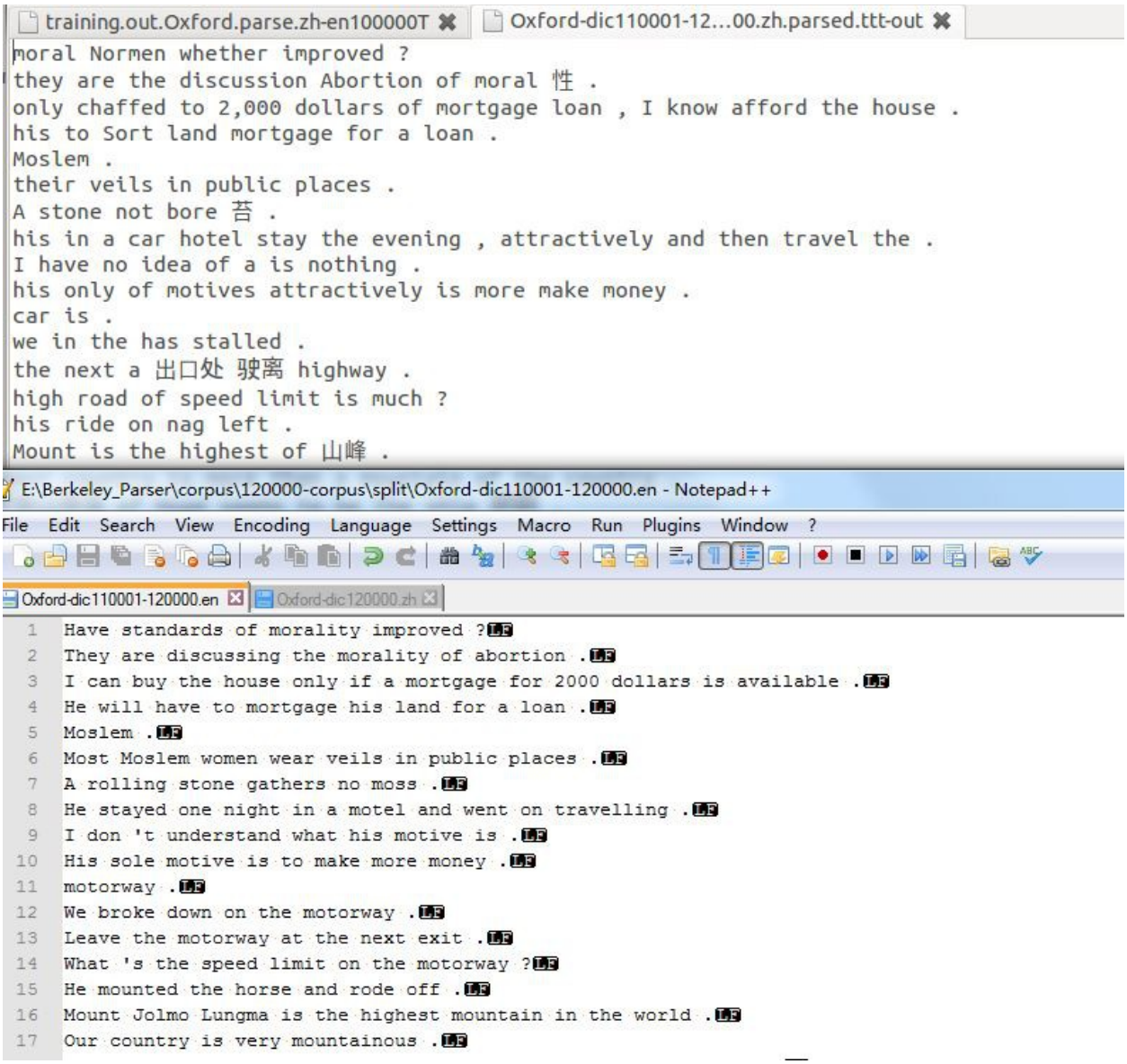


Using the testing command:

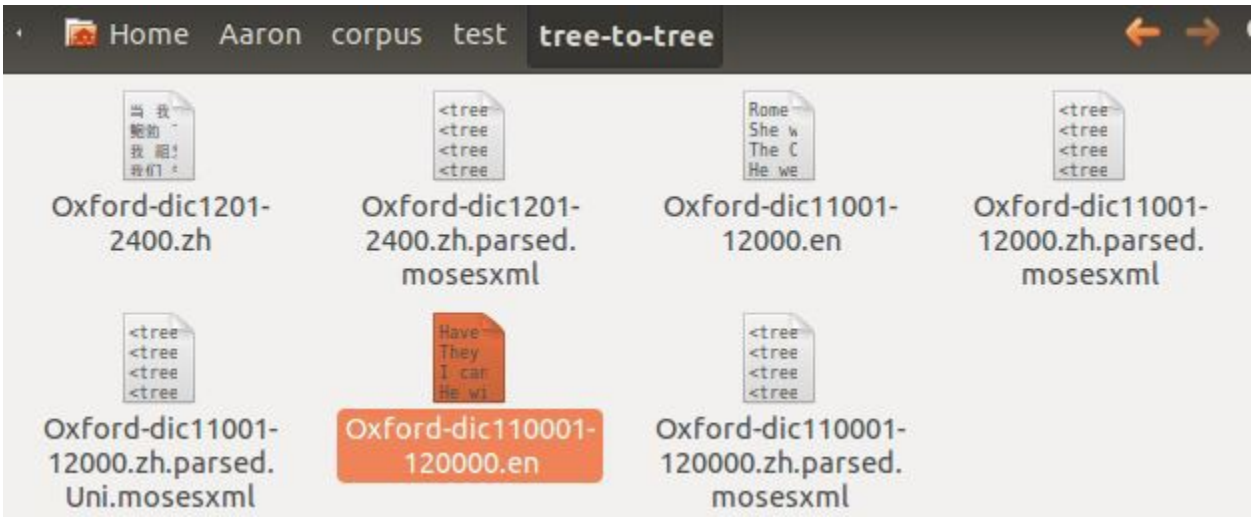
```
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$ ~/Aaron/Moses/mosesdecoder/bin/moses_c
hart -f train100000T/model/moses.ini < ~/Aaron/corpus/test/tree-to-tree/Oxford-d
ic110001-120000.zh.parsed.mosesxml > Oxford-dic110001-120000.zh.parsed.ttt-out
```

The translation result:

```
                                                    1   0   0
                                                   11   0
                                                    1
BEST TRANSLATION: 1304 Q  -> Q </s> :0-0 : c=-0.460 core=(0.000,-1.000,1.000,0.0
00,0.000,0.000,0.000,0.000,0.000)  [0..29] 1294 [total=-446.249] core=(-400.000,
-31.000,58.000,-22.687,-40.289,-33.279,-58.914,27.997,-171.625)
Translation took 0.040 seconds
End. : [228.000] seconds
Name:moses_chart        VmPeak:559000 kB        VmRSS:460288 kB RSSMax:472472 kB
user:23.949     sys:99.658      CPU:123.608     real:228.447
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$
```



Put the reference document:



Calculate the testing score using BLEU:

```
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$ ~/Aaron/Moses/mosesdecoder/scripts/gen
eric/multi-bleu.perl -lc ~/Aaron/corpus/test/tree-to-tree/Oxford-dic110001-12000
0.en < Oxford-dic110001-120000.zh.parsed.ttt-out
BLEU = 10.44, 41.8/12.1/6.1/4.0 (BP=0.987, ratio=0.987, hyp_len=115099, ref_len=
116607)
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$
```

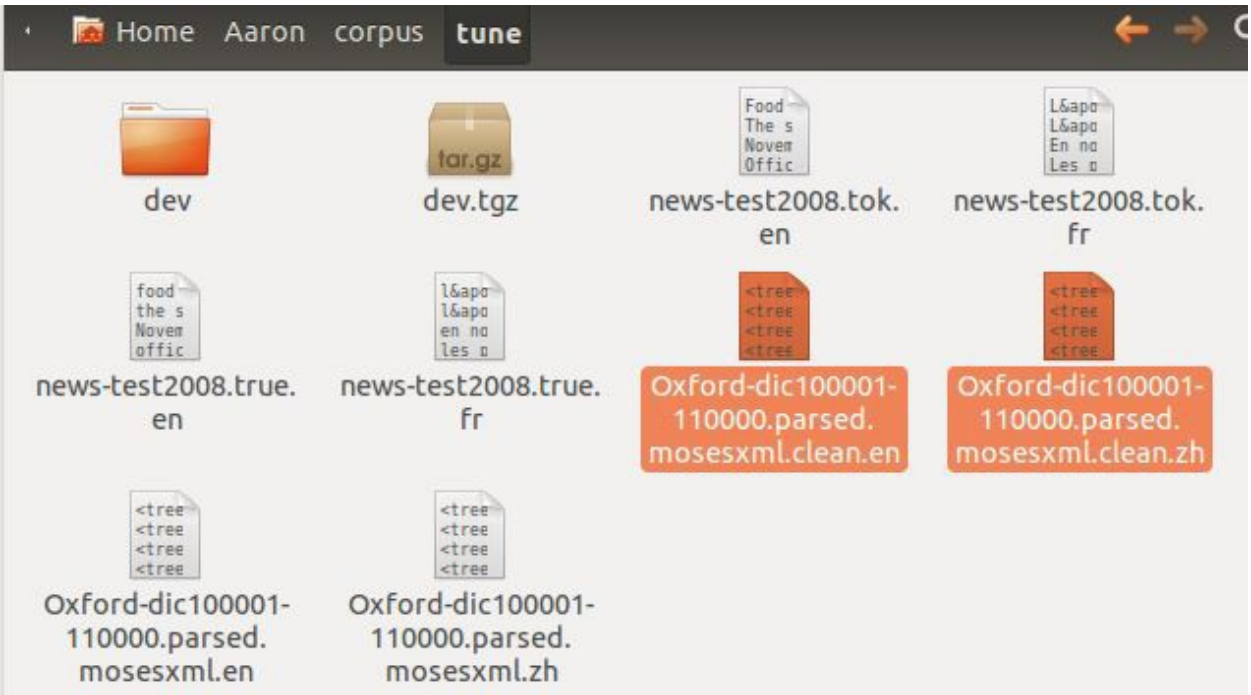
Tuning using 100001st -110000th sentences-3

Prepare the tuning document:

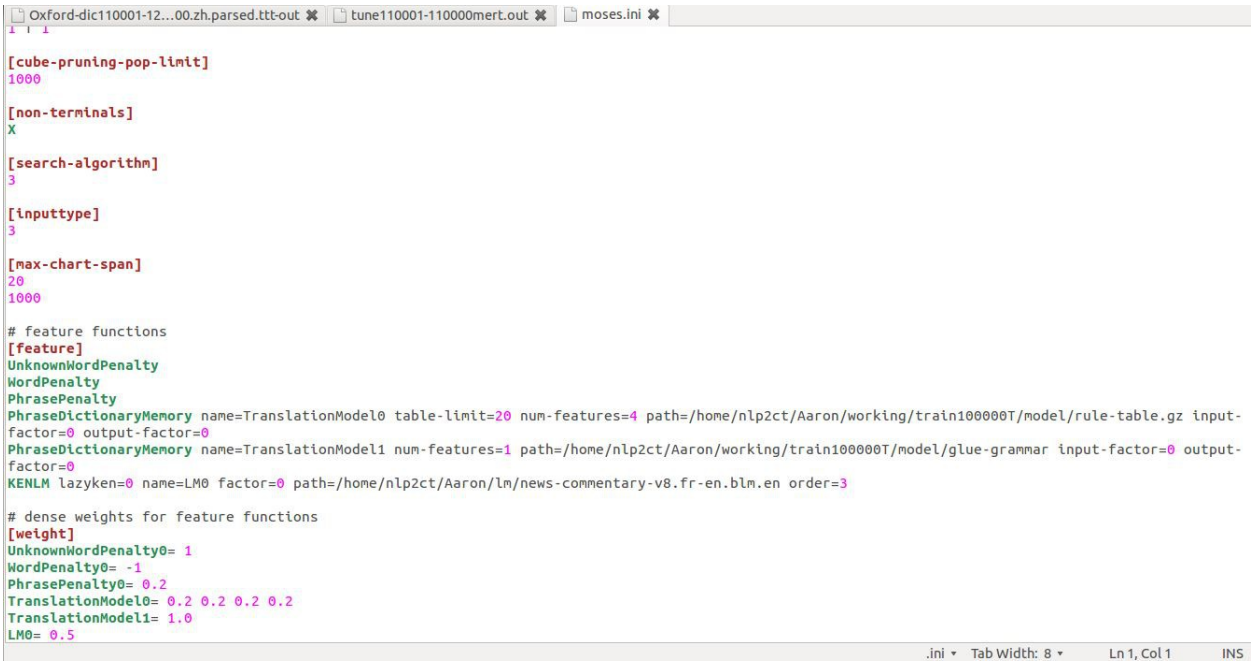
Clean the bilingual tuning files:

```
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$ ~/Aaron/Moses/mosesdecoder/scripts/train
ing/clean-corpus-n.perl ~/Aaron/corpus/tune/Oxford-dic100001-110000.parsed.mosesxm
l zh en ~/Aaron/corpus/tune/Oxford-dic100001-110000.parsed.mosesxml.clean 1 80
clean-corpus.perl: processing /home/nlp2ct/Aaron/corpus/tune/Oxford-dic100001-1100
00.parsed.mosesxml.zh & .en to /home/nlp2ct/Aaron/corpus/tune/Oxford-dic100001-110
000.parsed.mosesxml.clean, cutoff 1-80
.
Input sentences: 10000  Output sentences:  6388
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$
```

6388 sentences left among the 10,000 bilingual sentences.



The original parameters in moses.ini file before tuning:



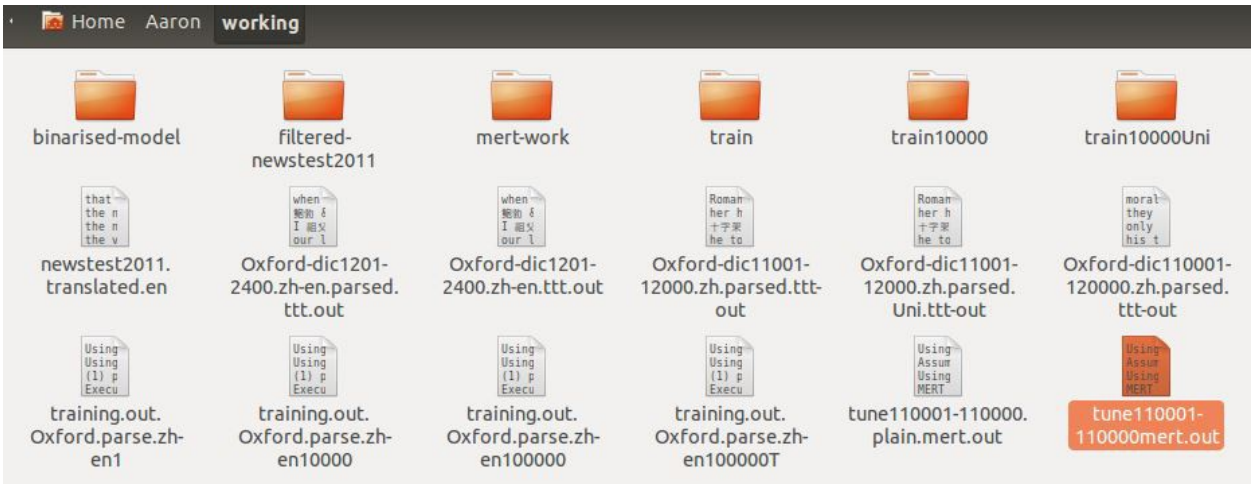
If use the following command for tuning:

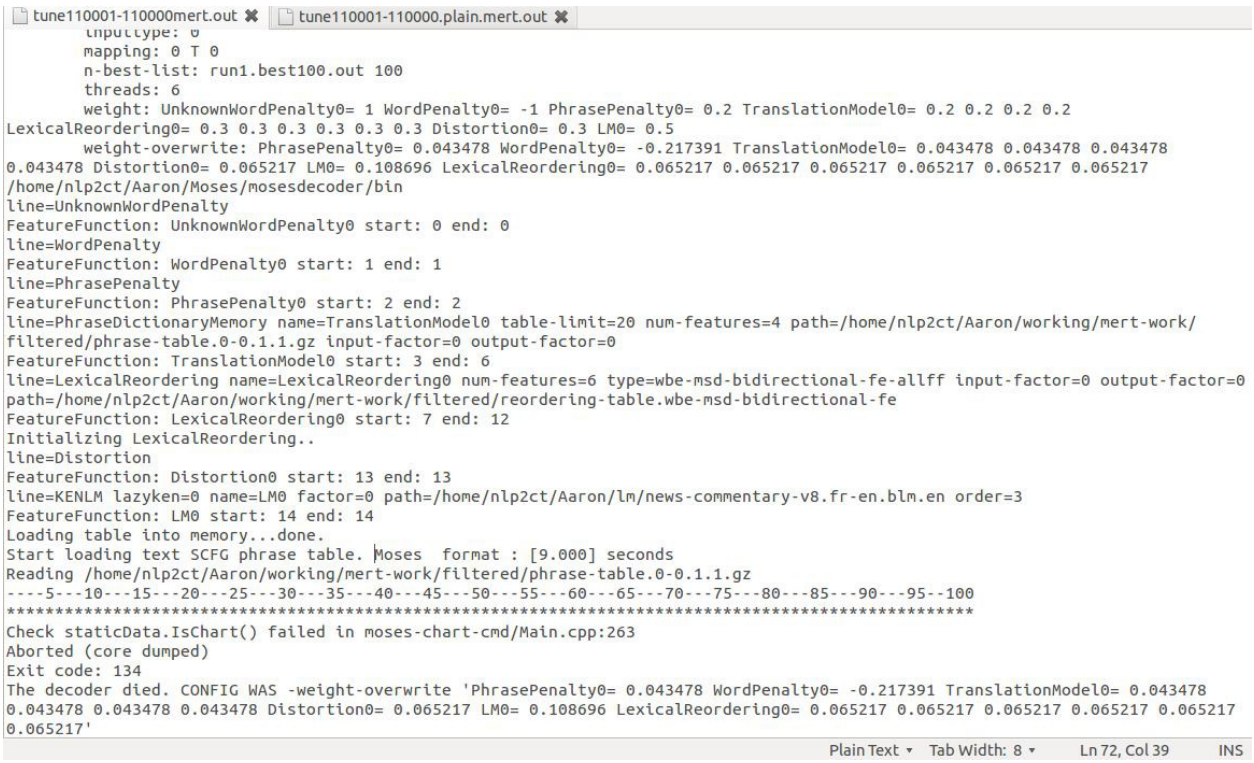
```
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$ ~/Aaron/Moses/mosesdecoder/scripts/training/mert-moses.pl ~/Aaron/corpus/tune/Oxford-dic100001-110000.parsed.mosesxml.clean.zh ~/Aaron/corpus/tune/Oxford-dic100001-110000.parsed.mosesxml.clean.en ~/Aaron/Moses/mosesdecoder/bin/moses_chart train100000T/model/moses.ini --mertdir ~/Aaron/Moses/mosesdecoder/bin/ --decoder-flags="-threads 6" &> tune110001-110000mert.out &
```

```
   13 root       0 -20     0    0    0 S    0  0.0   0:00.00 khelper
   14 root      20   0     0    0    0 S    0  0.0   0:00.00 kdevtmpfs
[1]+  Exit 2                  ~/Aaron/Moses/mosesdecoder/scripts/training/mert-moses.pl ~/Aaron/corpus/tune/Oxford-dic100001-110000.parsed.mosesxml.clean.zh ~/Aaron/corpus/tune/Oxford-dic100001-110000.parsed.mosesxml.clean.en ~/Aaron/Moses/mosesdecoder/bin/moses_chart train100000T/model/moses.ini --mertdir ~/Aaron/Moses/mosesdecoder/bin/ --decoder-flags="-threads 6" &>tune110001-110000mert.out
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$
```

~/Aaron/Moses/mosesdecoder/scripts/training/mert-moses.pl ~/Aaron/corpus/tune/Oxford-dic100001-11000.parsed.mosesxml.clean.zh ~/Aaron/corpus/tune/Oxford-dic100001-11000.parsed.mosesxml.clean.en ~/Aaron/Moses/mosesdecoder/bin/moses_chart train100000T/model/moses.ini --mertdir ~/Aaron/Moses/mosesdecoder/bin/ &> tune110001-110000mert.out &

As shown above, the tuning exits unsuccessfully. However, the “tune110001-110000mert.out” generated:





The record shows “Aborted (core dumped), Exit”, which means this exit is due to the un-enough memory of the computer. So let’s move it into the nlp2ct-186-server.

Tuning on nlp2ct-186-sever

The moses system has already installed on the sever as below directory with the root /smt:

```
wangyiming@lobo:~$
wangyiming@lobo:~$ ls
Aaron  corpus  CWMT2013  download  james  lm  moses  process  work
wangyiming@lobo:~$
wangyiming@lobo:~$
wangyiming@lobo:~$
wangyiming@lobo:~$ cd ..
wangyiming@lobo:/home$ cd ..
wangyiming@lobo:/$ smt/
-bash: smt/: Is a directory
wangyiming@lobo:/$ cd smt/
wangyiming@lobo:/smt$ ls
boost_1_55_0  boost-bin-dir  giza-bin-dir  giza-pp  irstlm-5.80.03  mosesdecoder
wangyiming@lobo:/smt$
```

Using the following command to try 186-sever tuning:

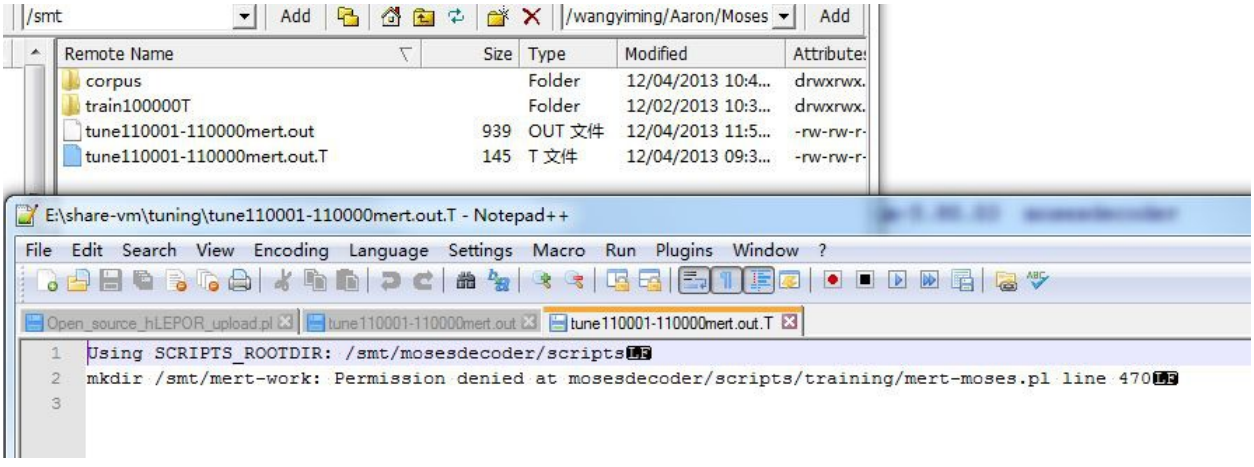
```
wangyiming@lobo:/smt/mosesdecoder$ ls
biconcor  BUILD-INSTRUCTIONS.txt  defer      lib   mira   moses-chart-cmd  OnDiskPt        regression-testing  symal
bin       contrib                 jam-files  lm    misc   moses-cmd        phrase-extract  scripts             util
bjam      cruise-control          Jamroot    mert  moses  NOTICE           previous.sh     search
wangyiming@lobo:/smt/mosesdecoder$ cd ..
wangyiming@lobo:/smt$
wangyiming@lobo:/smt$ mosesdecoder/scripts/training/mert-moses.pl ~/Aaron/Moses/corpus/tuning/Oxford-dic100001-110000.parsed.mosesxml.clean.zh ~/Aar
on/Moses/corpus/tuning/Oxford-dic100001-110000.parsed.mosesxml.clean.en mosesdecoder/bin/moses_chart ~/Aaron/Moses/train100000T/model/moses.ini --me
rtdir mosesdecoder/bin/ --decoder-flags="-threads 6" &> ~/Aaron/Moses/tune110001-110000mert.out
```

Without right mouse click for refresh, the generated file will not show in the 186-sever-directory.

If using the following command for tuning:

```
wangyiming@lobo:/smt$
wangyiming@lobo:/smt$ mosesdecoder/scripts/training/mert-moses.pl ~/Aaron/Moses/corpus/tuning/Oxford-dic100001-110000.parsed.mosesxml.clean.zh ~/Aar
on/Moses/corpus/tuning/Oxford-dic100001-110000.parsed.mosesxml.clean.en mosesdecoder/bin/moses_chart ~/Aaron/Moses/train100000T/model/moses.ini --me
rtdir mosesdecoder/bin/ --decoder-flags="-threads 6" &> ~/Aaron/Moses/tune110001-110000mert.out.T
wangyiming@lobo:/smt$
```

It exits with the error report in the generated record file:



If using the following command to try tuning:

```
wangyiming@lobo:~$ /smt/mosesdecoder/scripts/training/mert-moses.pl ~/Aaron/Moses/corpus/tuning/Oxford-dic100001-110000.parsed.mosesxml.clean.zh ~/A
aron/Moses/corpus/tuning/Oxford-dic100001-110000.parsed.mosesxml.clean.en /smt/mosesdecoder/bin/moses_chart ~/Aaron/Moses/train100000T/model/moses.i
ni --mertdir /smt/mosesdecoder/bin/ &> ~/Aaron/Moses/tune110001-110000mert.out
wangyiming@lobo:~$
```

It generate the record file with error report:

```
Using SCRIPTS_ROOTDIR: /smt/mosesdecoder/scripts
filtering the phrase tables... Wed Dec  4 11:52:15 CST 2013
exec: /smt/mosesdecoder/scripts/training/filter-model-given-input.pl ./filtered /home/wangyiming/Aaron/Moses/train100000T/model/moses.ini
/home/wangyiming/Aaron/Moses/corpus/tuning/Oxford-dic100001-110000.parsed.mosesxml.clean.zh
Executing: /smt/mosesdecoder/scripts/training/filter-model-given-input.pl ./filtered /home/wangyiming/Aaron/Moses/train100000T/model/moses.ini
/home/wangyiming/Aaron/Moses/corpus/tuning/Oxford-dic100001-110000.parsed.mosesxml.clean.zh > filterphrases.out 2> filterphrases.err
Exit code: 1
ERROR: Failed to run '/smt/mosesdecoder/scripts/training/filter-model-given-input.pl ./filtered /home/wangyiming/Aaron/Moses/train100000T/model/moses.ini
/home/wangyiming/Aaron/Moses/corpus/tuning/Oxford-dic100001-110000.parsed.mosesxml.clean.zh'. at /smt/mosesdecoder/scripts/training/mert-moses.pl line 1539.
```

Try the following command for tuning:

```
nlp2ct@nlp2ct-VirtualBox:~$
nlp2ct@nlp2ct-VirtualBox:~$ ~/Aaron/Moses/mosesdecoder/scripts/training/mert-mos
es.pl ~/Aaron/corpus/tune/Oxford-dic100001-110000.parsed.mosesxml.clean.zh ~/Aar
on/corpus/tune/Oxford-dic100001-110000.parsed.mosesxml.clean.en ~/Aaron/Moses/mo
sesdecoder/bin/moses_chart train100000T/model/moses.ini --mertdir ~/Aaron/Moses/
mosesdecoder/bin/ --decoder-flags="-threads 6" -glue-grammar --source-syntax --t
arget-syntax &> tune110001-110000.parsedmosesxml.mert.out &
```

It also exits as bellow. And there is no record file generated:

```
[1]+  Exit 1                  ~/Aaron/Moses/mosesdecoder/scripts/training/mert-m
oses.pl ~/Aaron/corpus/tune/Oxford-dic100001-110000.parsed.mosesxml.clean.zh ~/A
aron/corpus/tune/Oxford-dic100001-110000.parsed.mosesxml.clean.en ~/Aaron/Moses/
mosesdecoder/bin/moses_chart train100000T/model/moses.ini --mertdir ~/Aaron/Mose
s/mosesdecoder/bin/ --decoder-flags="-threads 6" -glue-grammar --source-syntax -
-target-syntax &>tune110001-110000.parsedmosesxml.mert.out
nlp2ct@nlp2ct-VirtualBox:~$
```

Try to use the clean corpus for tuning:

First, clean the plain corpus:

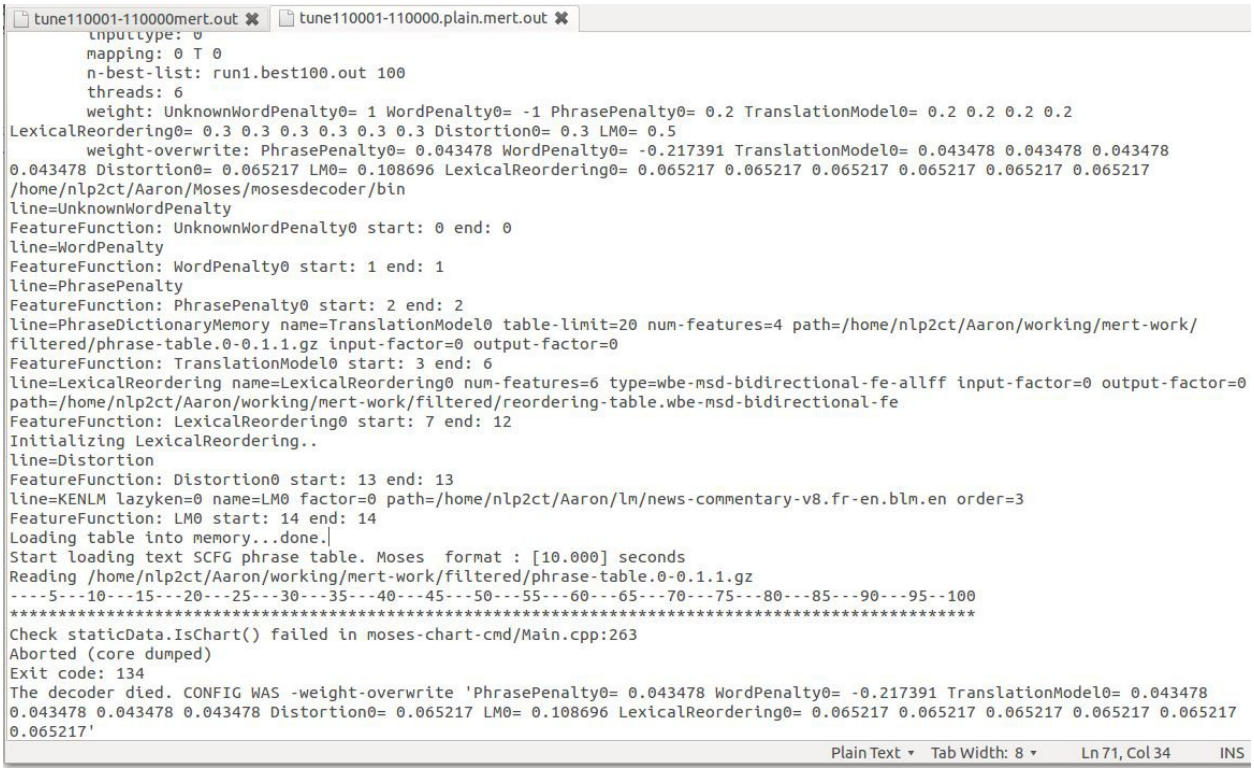
```
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$ ~/Aaron/Moses/mosesdecoder/scripts/train
ing/clean-corpus-n.perl ~/Aaron/corpus/tune/Oxford-dic100001-110000 zh en ~/Aaron/
corpus/tune/Oxford-dic100001-110000.clean 1 80
clean-corpus.perl: processing /home/nlp2ct/Aaron/corpus/tune/Oxford-dic100001-1100
00.zh & .en to /home/nlp2ct/Aaron/corpus/tune/Oxford-dic100001-110000.clean, cutof
f 1-80
.
Input sentences: 10000  Output sentences:  10000
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$
```

Using the following command to tune:

```
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$ ~/Aaron/Moses/mosesdecoder/scripts/train
ing/mert-moses.pl ~/Aaron/corpus/tune/Oxford-dic100001-110000.clean.zh ~/Aaron/cor
pus/tune/Oxford-dic100001-110000.clean.en ~/Aaron/Moses/mosesdecoder/bin/moses_cha
rt train100000T/model/moses.ini --mertdir ~/Aaron/Moses/mosesdecoder/bin/ --decode
r-flags="-threads 6" &> tune110001-110000.plain.mert.out &
```

```
   13 root       0 -20     0    0    0 S    0  0.0   0:00.00 khelper
   14 root      20   0     0    0    0 S    0  0.0   0:00.00 kdevtmpfs
[1]+  Exit 2                  ~/Aaron/Moses/mosesdecoder/scripts/training/mert-mos
es.pl ~/Aaron/corpus/tune/Oxford-dic100001-110000.clean.zh ~/Aaron/corpus/tune/Oxf
ord-dic100001-110000.clean.en ~/Aaron/Moses/mosesdecoder/bin/moses_chart train1000
00T/model/moses.ini --mertdir ~/Aaron/Moses/mosesdecoder/bin/ --decoder-flags="-th
reads 6" &>tune110001-110000.plain.mert.out
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$
```

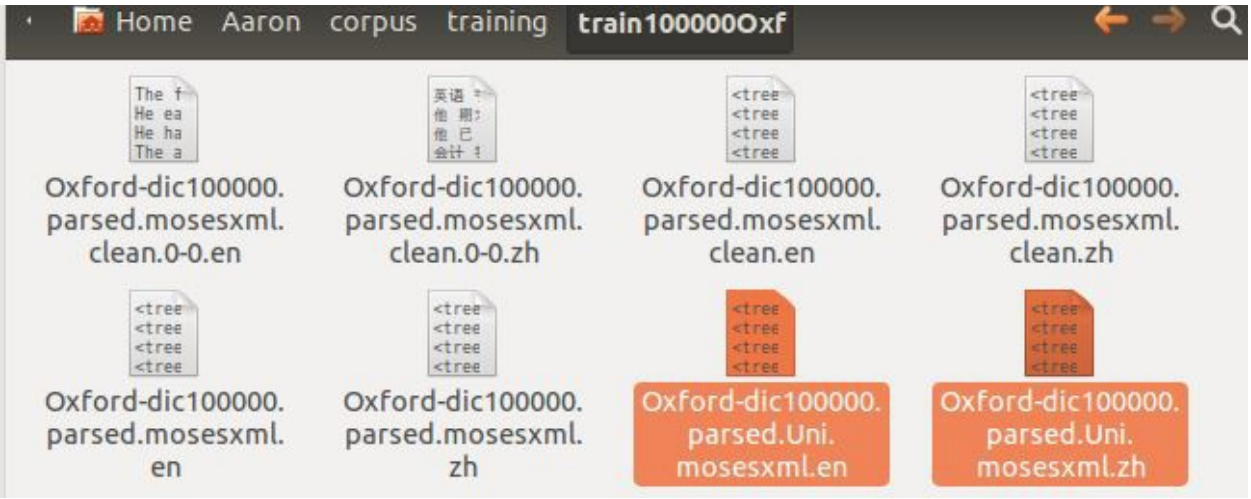
As shown above picture, the tuning also exits. Not successful. However, it generated the record file “tune110001-110000.plain.mert.out”:



This shows the exit reason “Aborted (core dumped)”, un-enough memory.

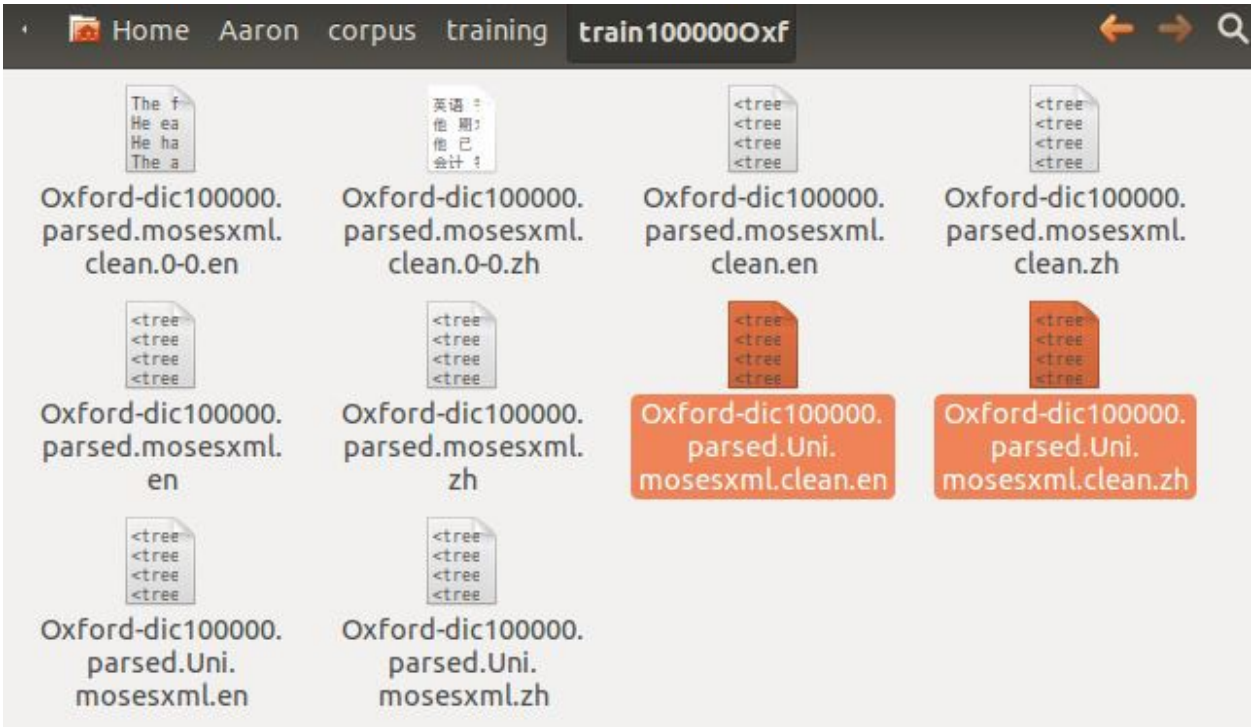
Preparing larger corpus-3-uni-phrse-tags

Put the prepared Uni-training documents:



Clean the document:

```
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$ ~/Aaron/Moses/mosesdecoder/scripts/tra
ining/clean-corpus-n.perl ~/Aaron/corpus/training/train100000Oxf/Oxford-dic10000
0.parsed.Uni.mosesxml zh en ~/Aaron/corpus/training/train100000Oxf/Oxford-dic100
000.parsed.Uni.mosesxml.clean 1 80
clean-corpus.perl: processing /home/nlp2ct/Aaron/corpus/training/train100000Oxf/
Oxford-dic100000.parsed.Uni.mosesxml.zh & .en to /home/nlp2ct/Aaron/corpus/train
ing/train100000Oxf/Oxford-dic100000.parsed.Uni.mosesxml.clean, cutoff 1-80
..........(100000)
Input sentences: 100000  Output sentences:  58202
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$
```



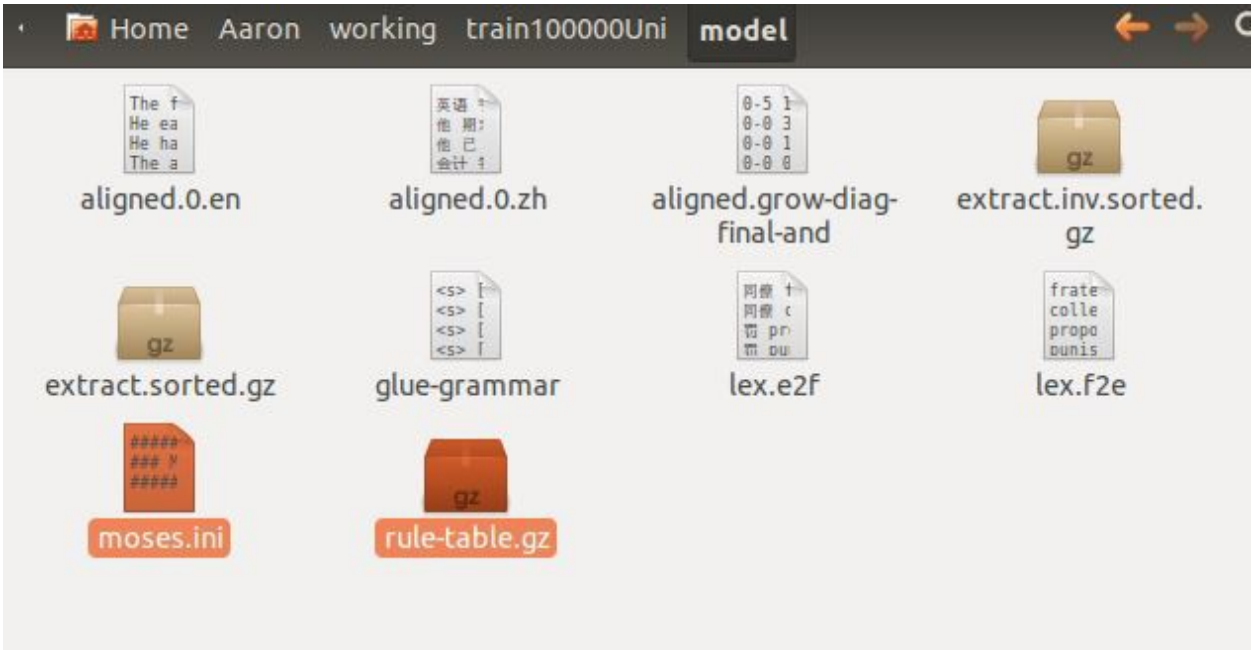
Training translation model-3-uni-phrase-tags

Using the following command for the training translation model:

```
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$ ~/Aaron/Moses/mosesdecoder/scripts/tra
ining/train-model.perl -root-dir train100000Uni -corpus ~/Aaron/corpus/training/
train100000Oxf/Oxford-dic100000.parsed.Uni.mosesxml.clean -f zh -e en -alignment
 grow-diag-final-and -hierarchical -glue-grammar --source-syntax --target-syntax
 -lm 0:3:$HOME/Aaron/lm/news-commentary-v8.fr-en.blm.en:8 -external-bin-dir ~/Aa
ron/Moses/mosesdecoder/tools >& training.out.Oxford.parse.Uni.zh-en100000 &
[1] 2927
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$ top

top - 15:49:24 up  5:45,  2 users,  load average: 0.22, 0.08, 0.06
Tasks: 154 total,   3 running, 151 sleeping,   0 stopped,   0 zombie
Cpu(s): 51.2%us,  1.2%sy,  0.0%ni, 47.6%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Mem:   6786596k total,  2001488k used,  4785108k free,   129348k buffers
Swap:  7336956k total,        0k used,  7336956k free,  1148156k cached

  PID USER      PR  NI  VIRT  RES  SHR S %CPU %MEM    TIME+  COMMAND
 2946 nlp2ct    20   0 39376  27m 1396 R  100  0.4   0:12.43 mkcls
 1160 root      20   0  322m 138m  14m S    2  2.1   2:51.32 Xorg
 1677 nlp2ct    20   0 1199m  90m  36m R    1  1.4   2:55.28 compiz
```

the training finished as:

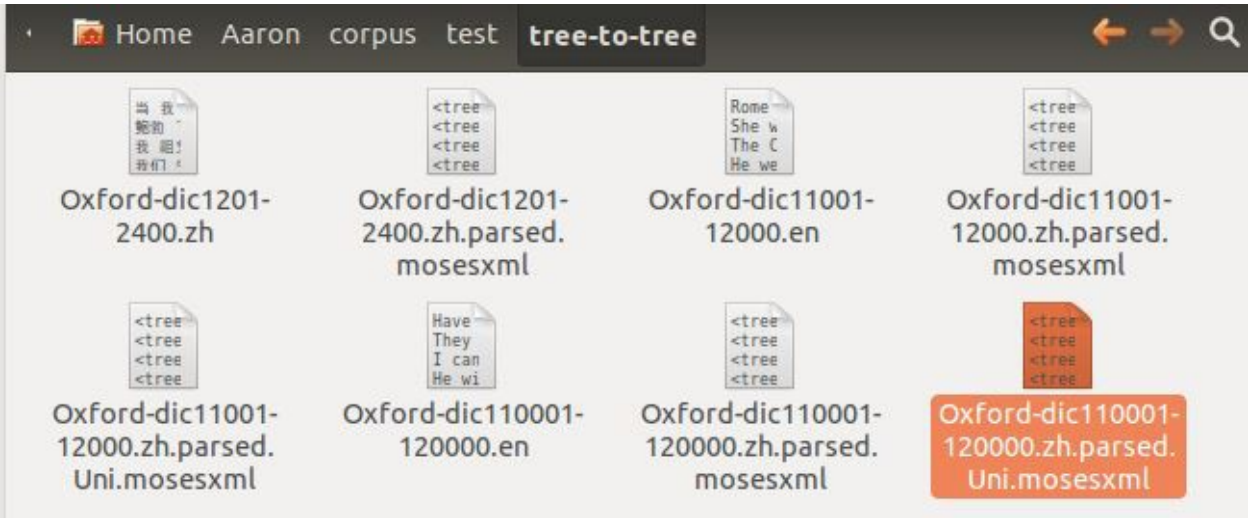
```
   13 root       0 -20     0    0    0 S    0  0.0   0:00.00 khelper
   14 root      20   0     0    0    0 S    0  0.0   0:00.00 kdevtmpfs
[1]+  Done                    ~/Aaron/Moses/mosesdecoder/scripts/training/train-
model.perl -root-dir train100000Uni -corpus ~/Aaron/corpus/training/train100000
xf/Oxford-dic100000.parsed.Uni.mosesxml.clean -f zh -e en -alignment grow-diag-f
inal-and -hierarchical -glue-grammar --source-syntax --target-syntax -lm 0:3:$HO
ME/Aaron/lm/news-commentary-v8.fr-en.blm.en:8 -external-bin-dir ~/Aaron/Moses/mo
sesdecoder/tools &>training.out.Oxford.parse.Uni.zh-en100000
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$
```



The rule-table.gz is 5.2MB.

testing-3-uni-phrase-tags:

prepare the testing corpus:



The testing command:

```
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$ ~/Aaron/Moses/mosesdecoder/bin/moses_c
hart -f train100000Uni/model/moses.ini < ~/Aaron/corpus/test/tree-to-tree/Oxford
-dic110001-120000.zh.parsed.Uni.mosesxml > Oxford-dic110001-120000.zh.parsed.Uni
.ttt-out
```

The testing finished as:

```
                                                    1   0   0   0
                                                      1   0   0
                                                       11   0
                                                         1
BEST TRANSLATION: 1304 Q  -> Q </s> :0-0 : c=-0.460 core=(0.000,-1.000,1.000,0.0
00,0.000,0.000,0.000,0.000,0.000)  [0..29] 1294 [total=-446.249] core=(-400.000,
-31.000,58.000,-22.687,-40.289,-33.279,-58.914,27.997,-171.625)
Translation took 0.040 seconds
End. : [229.000] seconds
Name:moses_chart        VmPeak:557012 kB        VmRSS:450864 kB RSSMax:470448 kB
user:24.278     sys:98.934      CPU:123.212     real:228.910
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$
```

Oxford-dic110001-12...00.zh.parsed.ttt-out | Oxford-dic110001-12...h.parsed.Uni.ttt-out

```
moral Normen whether improved ?
they are the discussion Abortion of moral 性 .
only chaffed to 2,000 dollars of mortgage loan , I know afford the house .
his to Sort land mortgage for a loan .
Moslem .
their veils in public places .
A stone not bore 苔 .
his in a car hotel stay the evening , attractively and then travel the .
I have no idea of a is nothing .
his only of motives attractively is more make money .
car is .
we in the has stalled .
the next a 出口处 驶离 highway .
high road of speed limit is much ?
his ride on nag left .
Mount is the highest of 山峰 .
our country is more than a mountain of the country .
排山倒海 of Huge seems to be the ship 掀翻 .
this 芭蕾舞 actress of every action dampness is a .
he was 困 in against bad of Cars Volumes , two legs 动弹 Delay .
public opinion is strongly in favor of disarmament .
I Sons six years of time to Spoken multiplication .
students of increase the school seemed Heavy .
large of people get the exhibition hall of the door .
normally people may will jibed his Music , but our more of it .
I in a 市立 at school .
her in a 市立 of the library work .
murmur that your friends must die .
his pay the extra cost murmur .
```

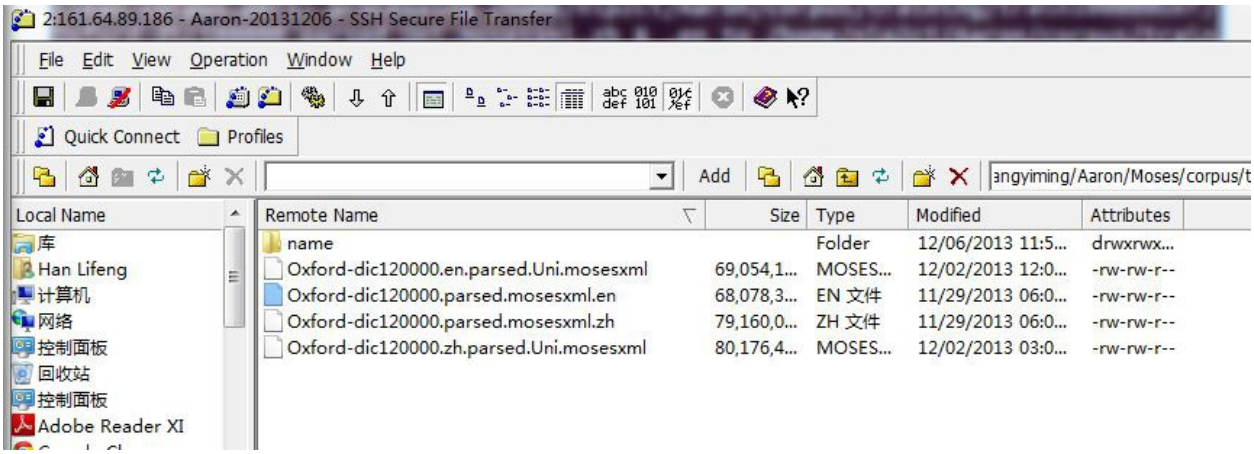
Calculate the testing score using BLEU:

```
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$ ~/Aaron/Moses/mosesdecoder/scripts/gen
eric/multi-bleu.perl -lc ~/Aaron/corpus/test/tree-to-tree/Oxford-dic110001-12000
0.en < Oxford-dic110001-120000.zh.parsed.Uni.ttt-out
BLEU = 10.43, 41.8/12.1/6.1/4.0 (BP=0.987, ratio=0.987, hyp_len=115124, ref_len=
116607)
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$
```

The testing result using original phrase tagset is BLEU=10.44, which means there is almost no difference.

Preparing larger corpus-4-186sever

Using the Oxford-120000 total sentences for training. The parsing use WSJ-trained-en and CTB-7-trained-cn, cleaning,



Clean the document:

```
wangyiming@lobo:~/Aaron/Moses$
wangyiming@lobo:~/Aaron/Moses$
wangyiming@lobo:~/Aaron/Moses$ /smt/mosesdecoder/scripts/training/clean-corpus-n
.perl corpus/training-120000/Oxford-dic120000.parsed.mosesxml zh en corpus/train
ing-120000/Oxford-dic120000.parsed.mosesxml.clean 1 80
clean-corpus.perl: processing corpus/training-120000/Oxford-dic120000.parsed.mos
esxml.zh & .en to corpus/training-120000/Oxford-dic120000.parsed.mosesxml.clean,
 cutoff 1-80
..........(100000)..
Input sentences: 120000  Output sentences:  68971
wangyiming@lobo:~/Aaron/Moses$
Connected to 161.64.89.186   SSH2 - aes128-cbc - hmac-md5   80x24
```

It shows that 68971 sentences are left after cleaning among the 120000 sentences.

Training translation model-4-186sever

Using the following command for the training translation model:

```
wangyiming@lobo:~/Aaron/Moses$
wangyiming@lobo:~/Aaron/Moses$ /smt/mosesdecoder/scripts/training/train-model.pe
rl -root-dir ./train120000 -corpus ./corpus/training-120000/Oxford-dic120000.par
sed.mosesxml.clean -f zh -e en -alignment grow-diag-final-and -hierarchical -glu
e-grammar --source-syntax --target-syntax -lm 0:3:$HOME/Aaron/Moses/lm/news-comm
entary-v8.fr-en.blm.en:8 -external-bin-dir /smt/giza-bin-dir/ >& train.out.Oxfor
d.parse.zh-en120000 &
[1] 2588
wangyiming@lobo:~/Aaron/Moses$ top
top - 12:21:08 up 199 days, 19:47,  2 users,  load average: 25.67, 25.24, 25.23
Tasks: 260 total,   2 running, 258 sleeping,   0 stopped,   0 zombie
Cpu(s):  6.8%us,  0.0%sy, 93.2%ni,  0.0%id,  0.0%wa,  0.0%hi,  0.0%si,  0.0%st
Mem:  148540056k total, 47119720k used, 101420336k free,   542200k buffers
Swap: 50319356k total,  1117744k used, 49201612k free, 28598256k cached

  PID USER      PR  NI  VIRT  RES  SHR S %CPU %MEM    TIME+  COMMAND
 1100 luyi      39  19 1959m 7672 1212 S 2236  0.0 360375:54 java
 2609 wangyimi  20   0 42016  29m 1460 R   92  0.0   1:23.61 mkcls
 7197 luyi      20   0 33.2g  11g 4980 S   70  8.4  67366:50 java
 2606 wangyimi  20   0 17476 1432  960 R    0  0.0   0:00.26 top
 4011 luyi      20   0 12.5g 8872 1360 S    0  0.0   2107:31 python
    1 root      20   0 24340 1556  756 S    0  0.0   0:01.59 init
    2 root      20   0     0    0    0 S    0  0.0   0:05.00 kthreadd
    3 root      20   0     0    0    0 S    0  0.0  14:43.81 ksoftirqd/0
    5 root      20   0     0    0    0 S    0  0.0   6:12.74 kworker/u:0
    6 root      RT   0     0    0    0 S    0  0.0   0:18.52 migration/0
    7 root      RT   0     0    0    0 S    0  0.0   0:50.89 watchdog/0
    8 root      RT   0     0    0    0 S    0  0.0   0:33.95 migration/1
   10 root      20   0     0    0    0 S    0  0.0  10:23.73 ksoftirqd/1
```

The training finished as :

```
   27 root      RT   0     0    0    0 S    0  0.0   0:41.99 watchdog/5
   28 root      RT   0     0    0    0 S    0  0.0   0:07.66 migration/6
   30 root      20   0     0    0    0 S    0  0.0   6:21.05 ksoftirqd/6
   31 root      RT   0     0    0    0 S    0  0.0   0:44.20 watchdog/6
[1]+  Done                    /smt/mosesdecoder/scripts/training/train-model.per
l -root-dir ./train120000 -corpus ./corpus/training-120000/Oxford-dic120000.pars
ed.mosesxml.clean -f zh -e en -alignment grow-diag-final-and -hierarchical -glue
-grammar --source-syntax --target-syntax -lm 0:3:$HOME/Aaron/Moses/lm/news-comme
ntary-v8.fr-en.blm.en:8 -external-bin-dir /smt/giza-bin-dir/ &>train.out.Oxford.
parse.zh-en120000
wangyiming@lobo:~/Aaron/Moses$ jobs
wangyiming@lobo:~/Aaron/Moses$
wangyiming@lobo:~/Aaron/Moses$
```

Using SCRIPTS_ROOTDIR: /smt/mosesdecoder/scripts

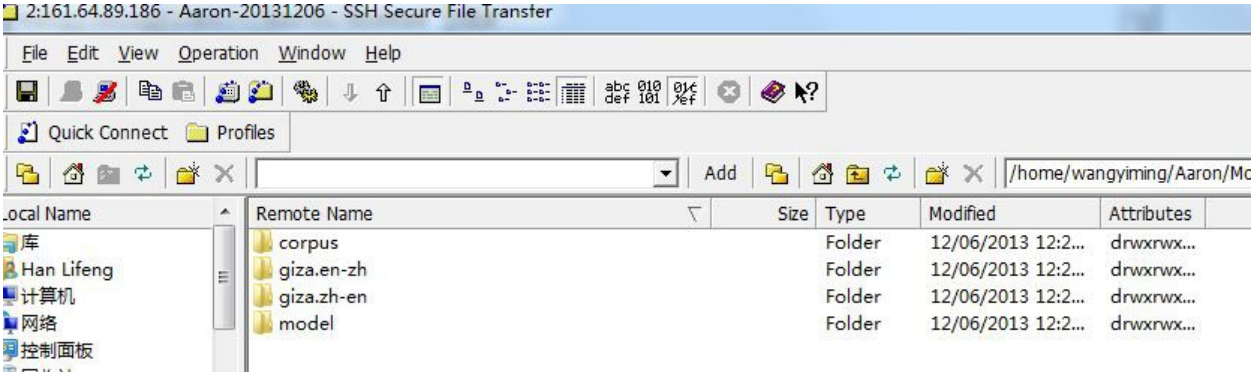
Using single-thread GIZA

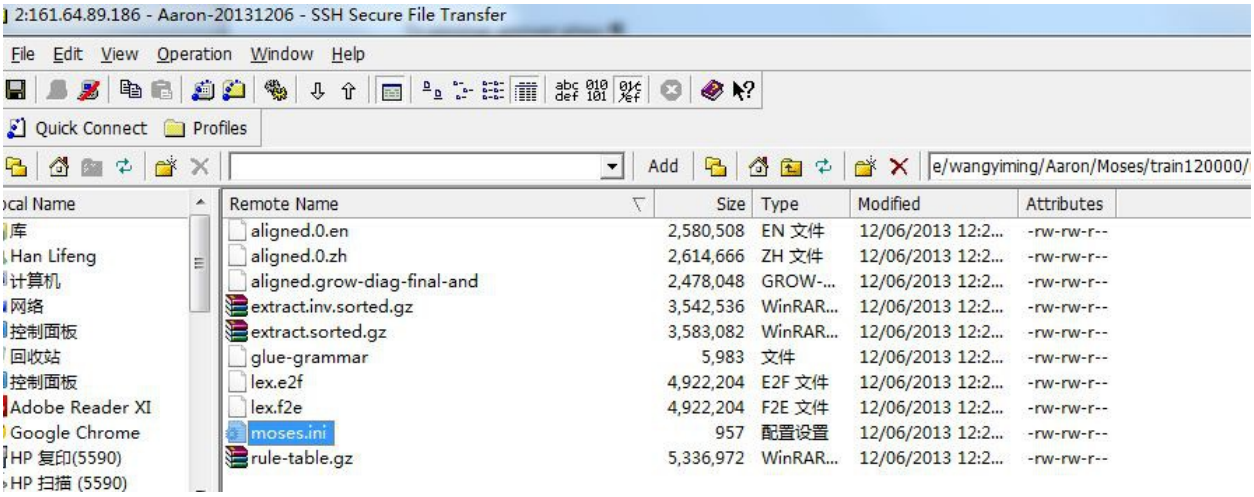
(1) preparing corpus @ Fri Dec 6 12:19:27 CST 2013

…

(9) create moses.ini @ Fri Dec 6 12:28:49 CST 2013

Training generates:

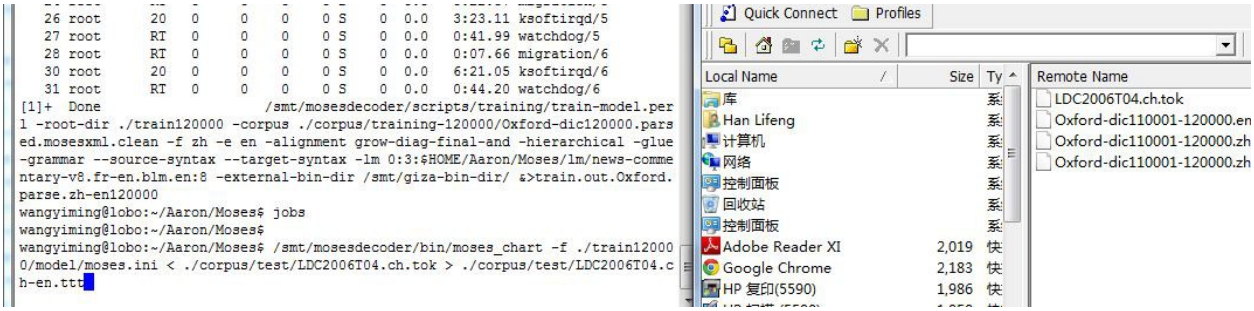




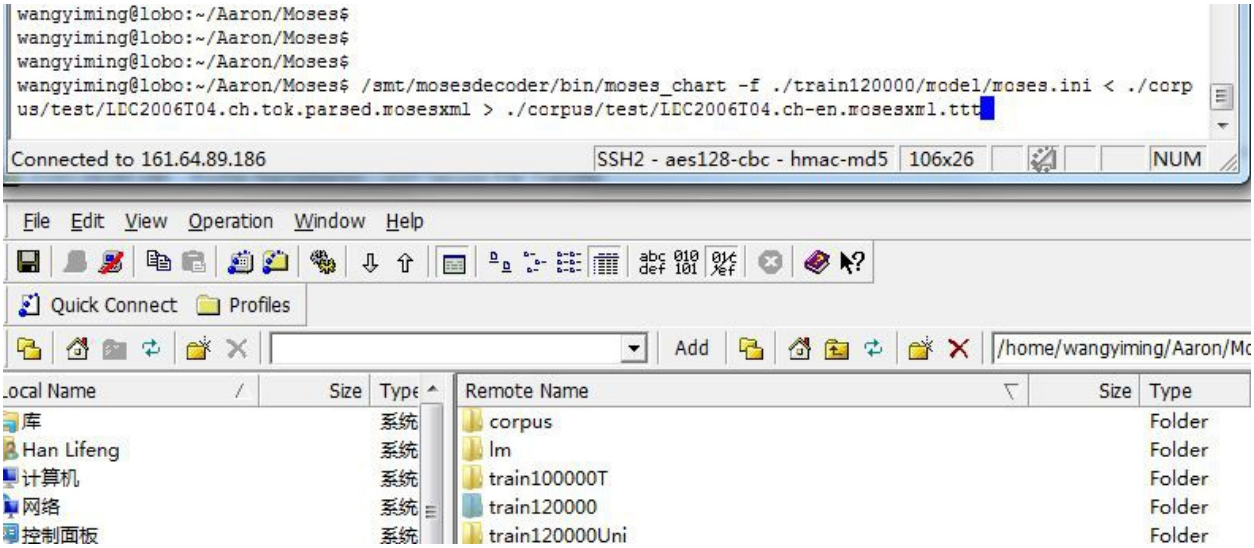
Testing-4-186sever:

Firstly, we used the NIST-MT03 testing CN-.EN data

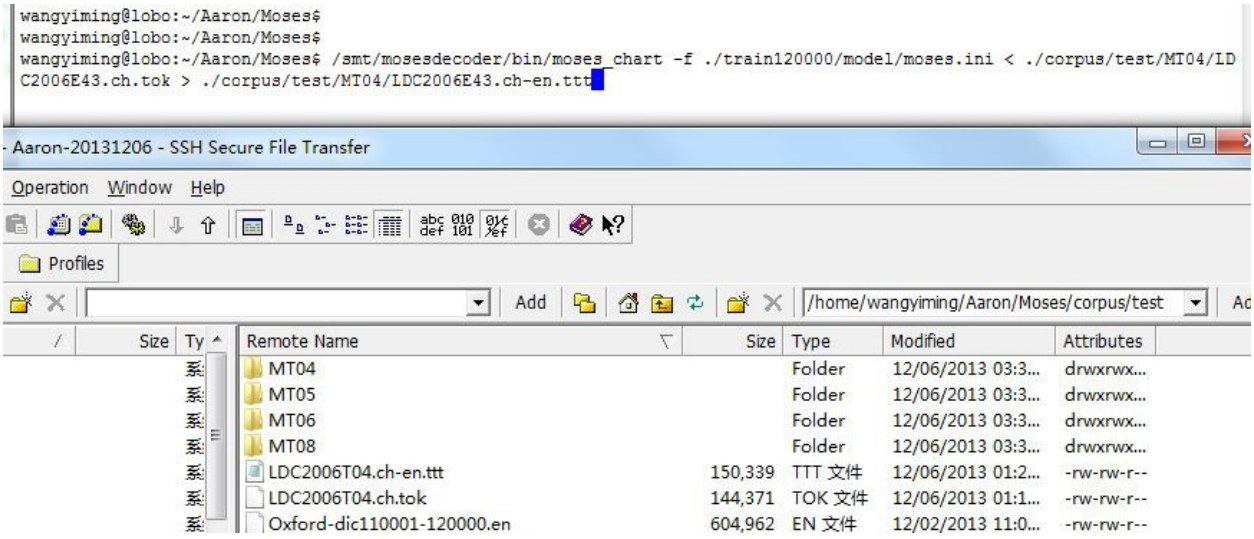
The testing command:



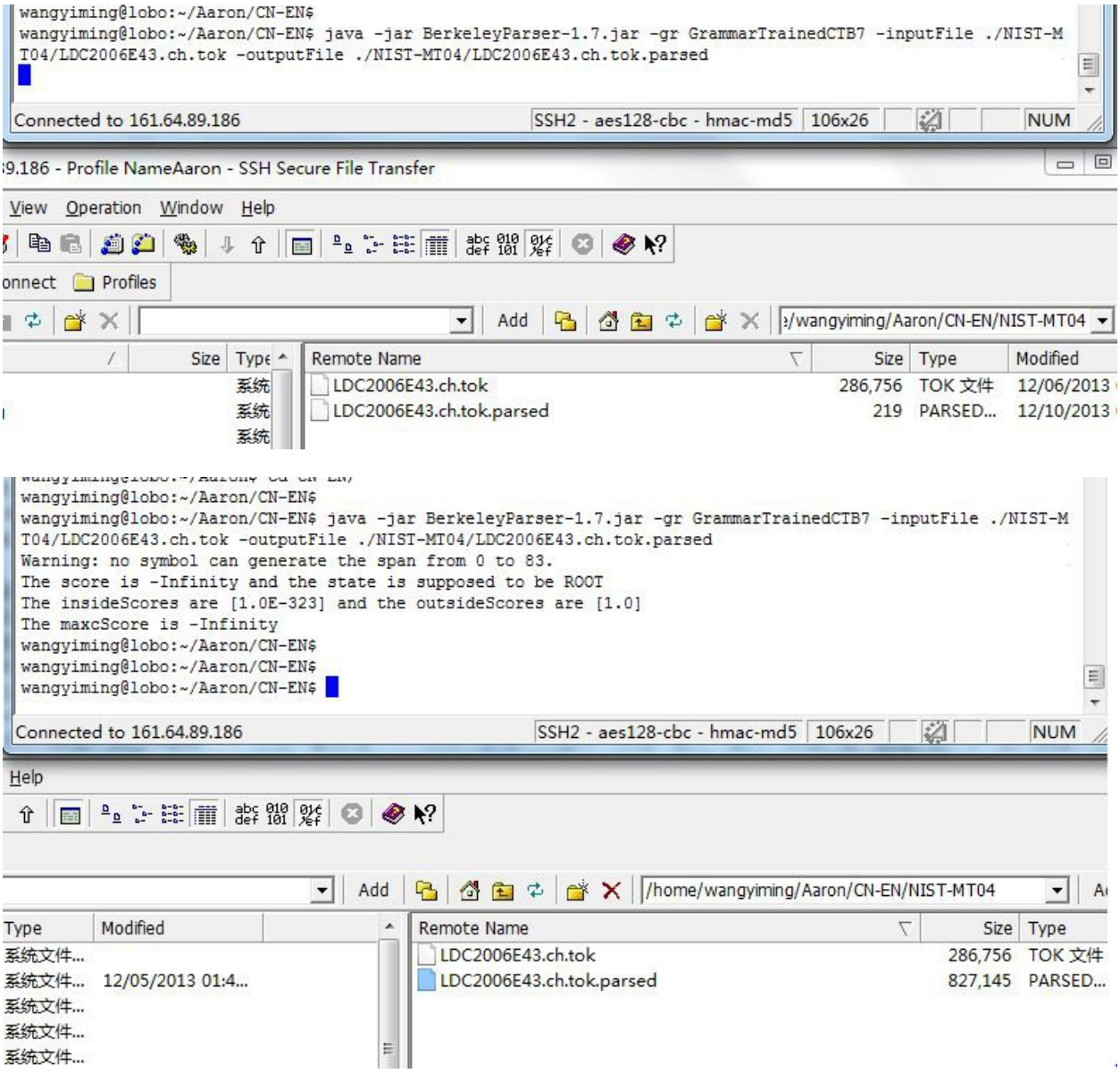
Testing once more using the parsed (use the grammar trained on CTB7) and converted mosesxml format:



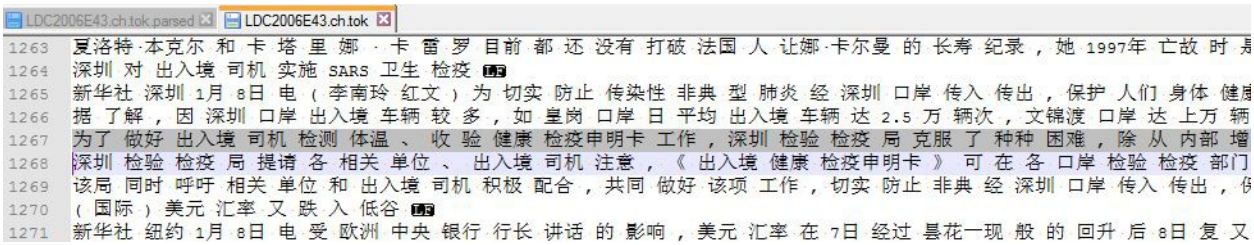
Secondly, we used the NIST-MT04 CN-EN test data:



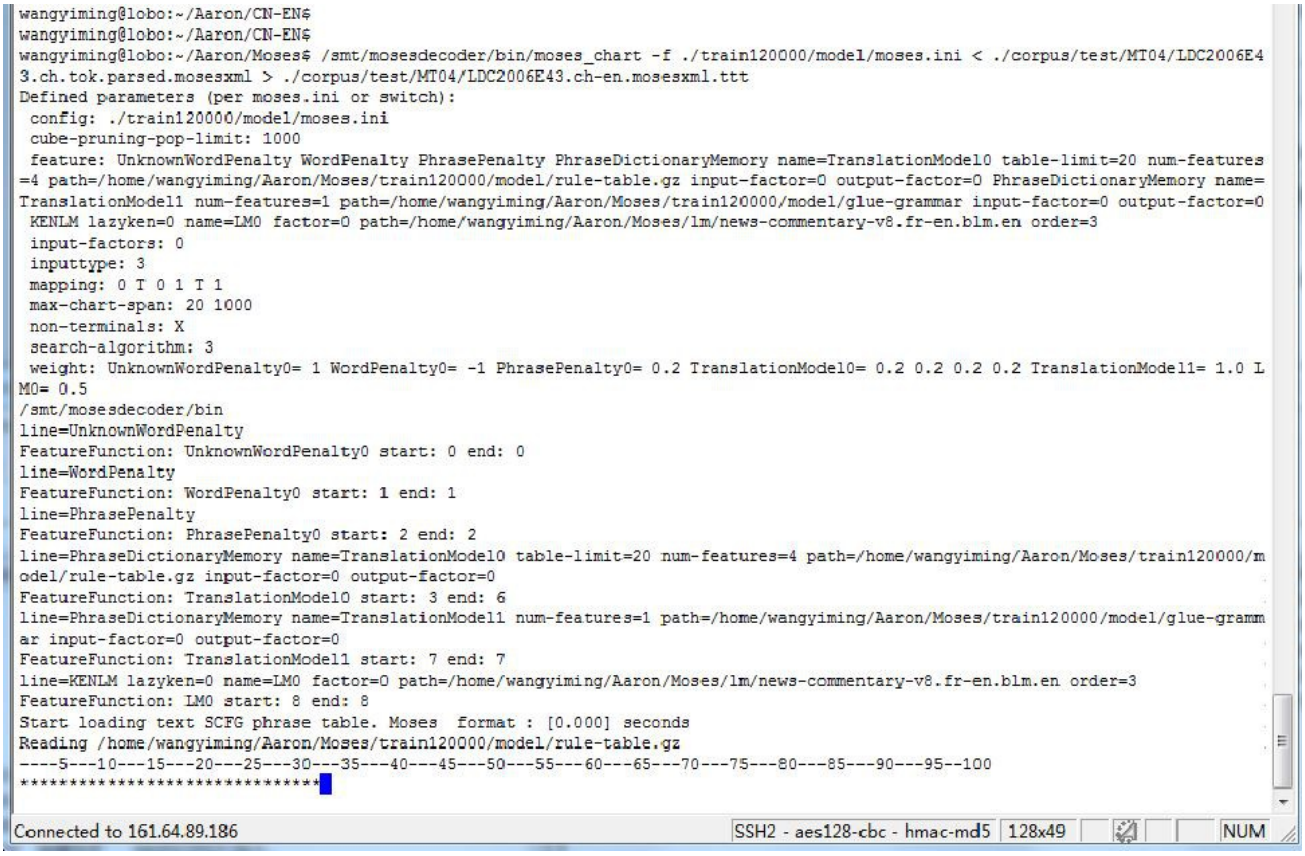
<u>Once more, parse the sentence, and then testing:</u>



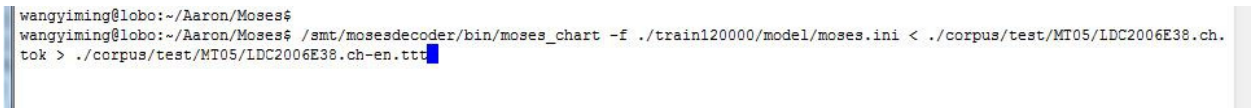
The warning is due to that the 1267th sentence is too long to parse successfully. The parsed document shows that the 1267th sentence is lost, however the previous and following sentences are normal:



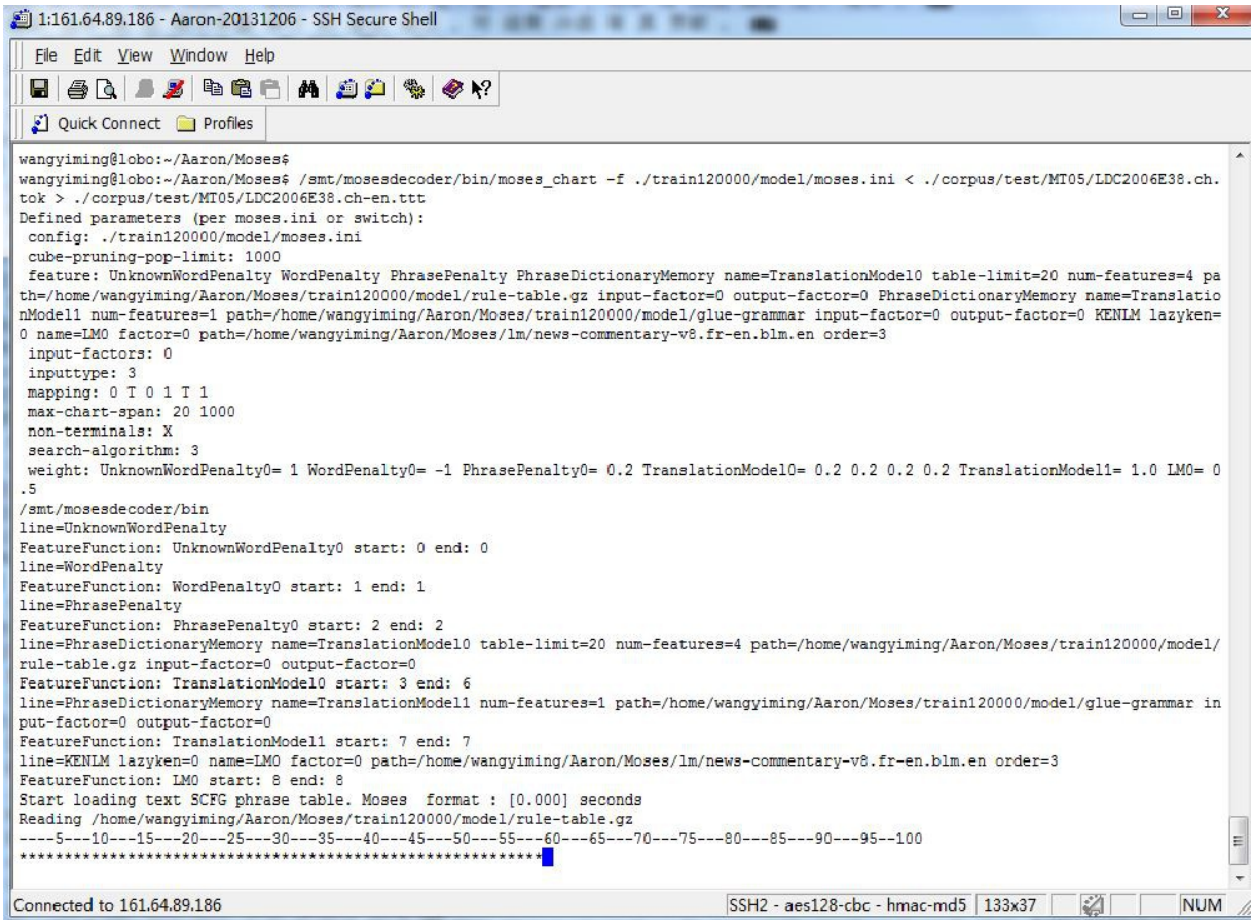
The testing command and run:



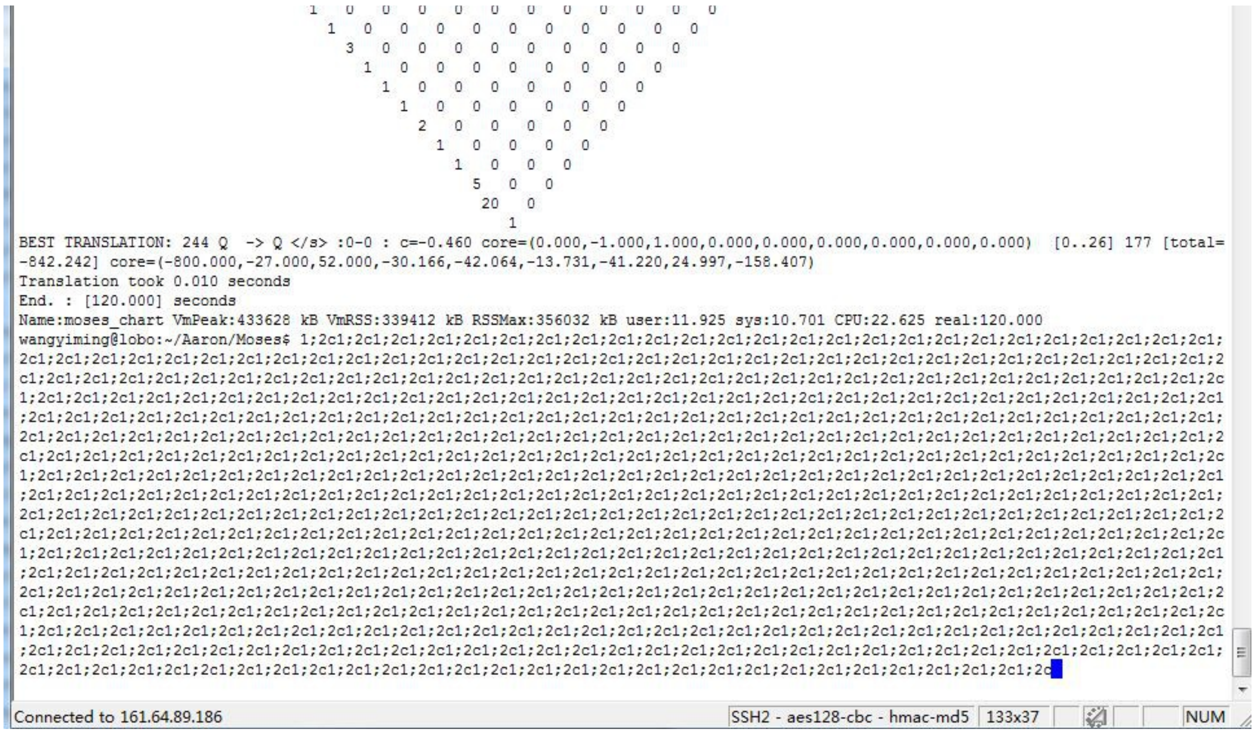
Thirdly, test the NIST-MT05 testing data:



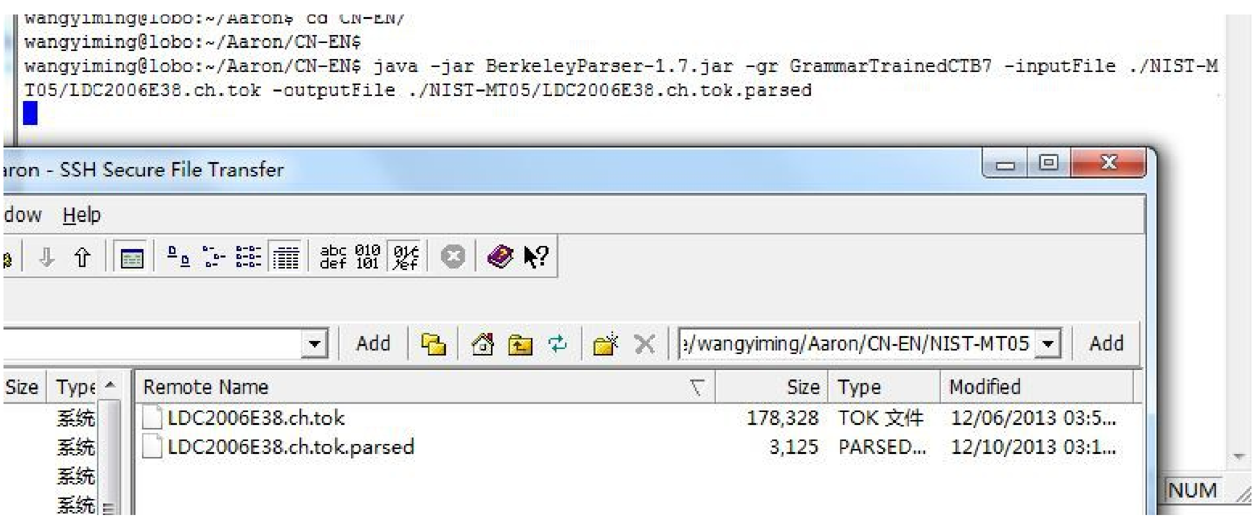
It runs as below, the reading of tule-table.gz, etc.:



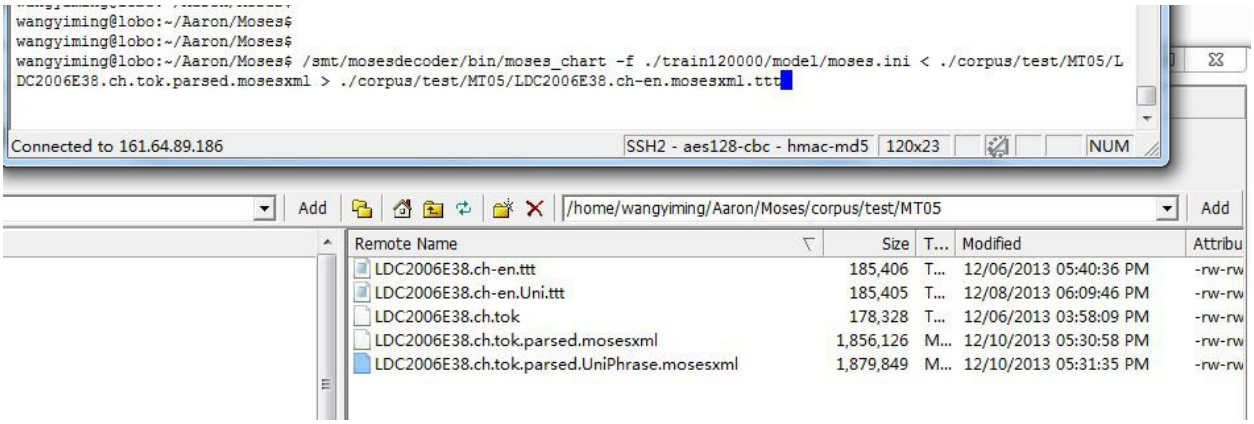
The testing finished as:



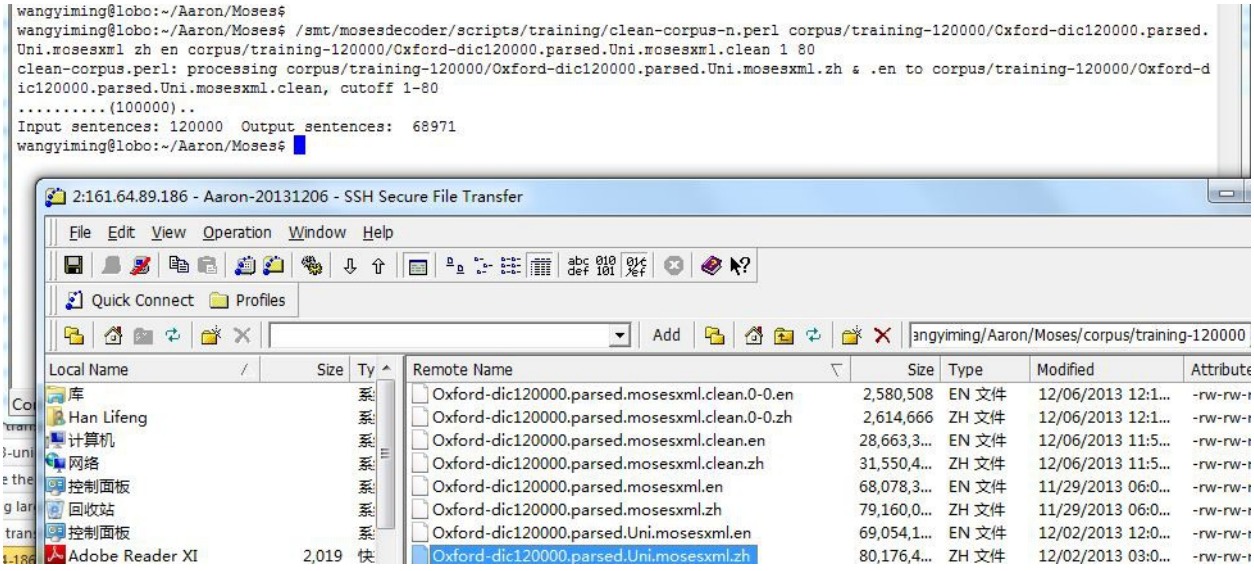
Once more, we should first parse the input sentence, then testing:



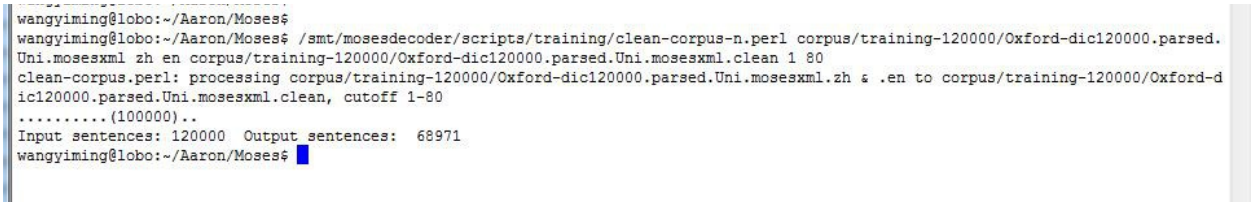
Testing command:



Preparing larger corpus-4-Uni-186sever



Clean the corpus:

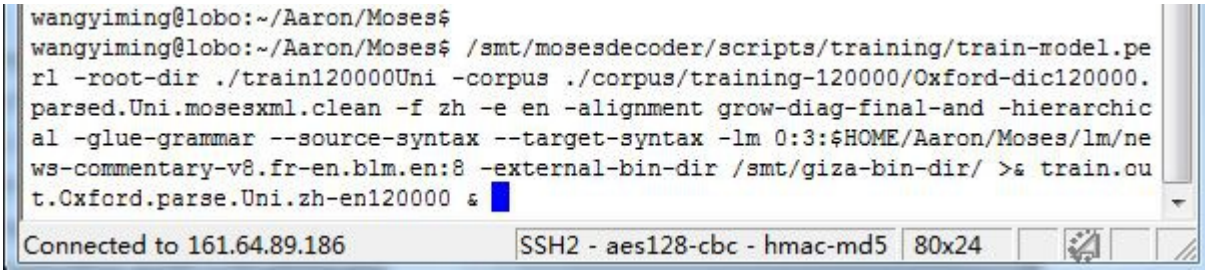


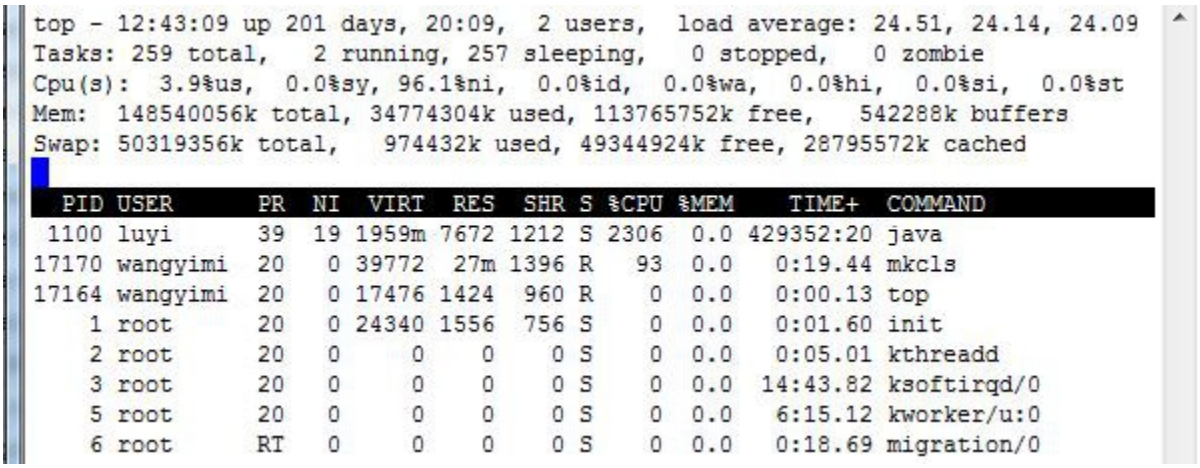
Training translation model-4-Uni-186sever

Use the following training command:

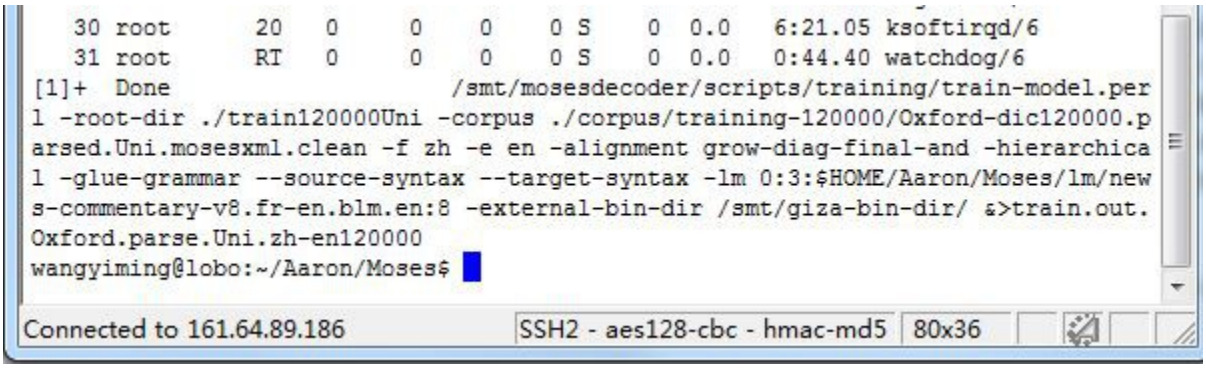
wangyiming@lobo:~/Aaron/Moses$

/smt/mosesdecoder/scripts/training/train-model.perl -root-dir ./train120000Uni -corpus ./corpus/training-120000/Oxford-dic120000.parsed.Uni.mosesxml.clean -f zh -e en -alignment grow-diag-final-and -hierarchical -glue-grammar --source-syntax --target-syntax -lm 0:3:$HOME/Aaron/Moses/lm/news-commentary-v8.fr-en.blm.en:8 -external-bin-dir /smt/giza-bin-dir/ &>train.out.Oxford.parse.Uni.zh-en120000

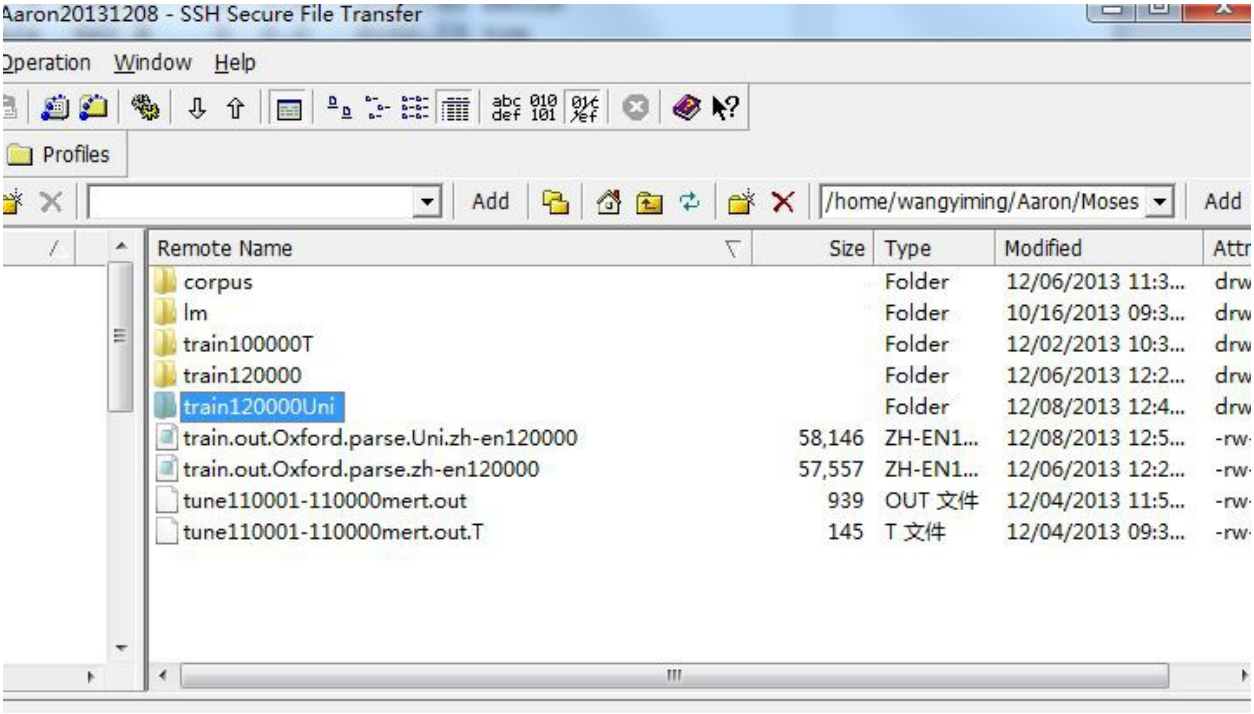


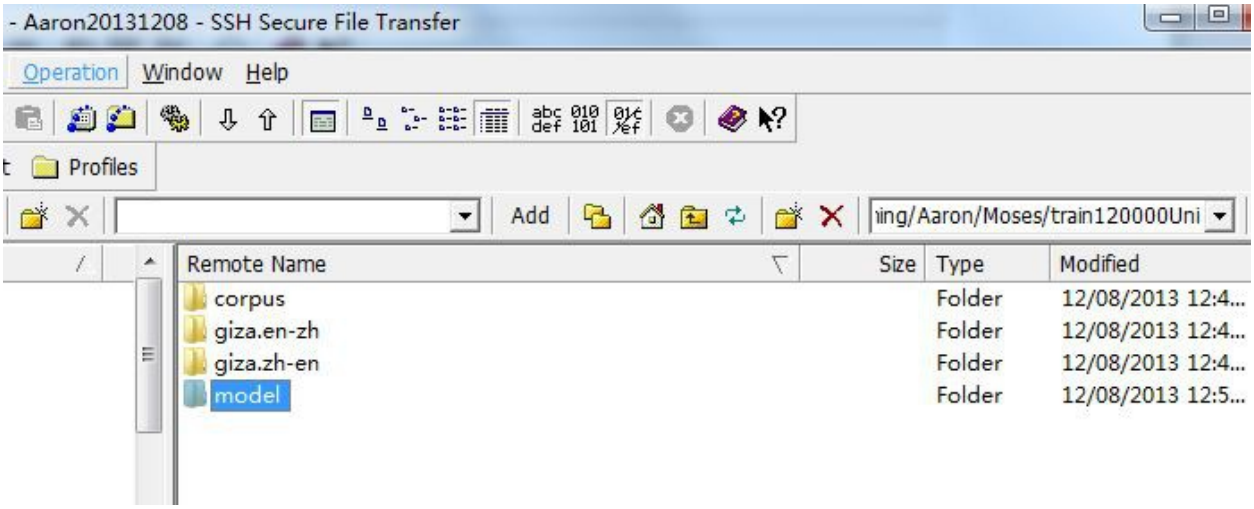


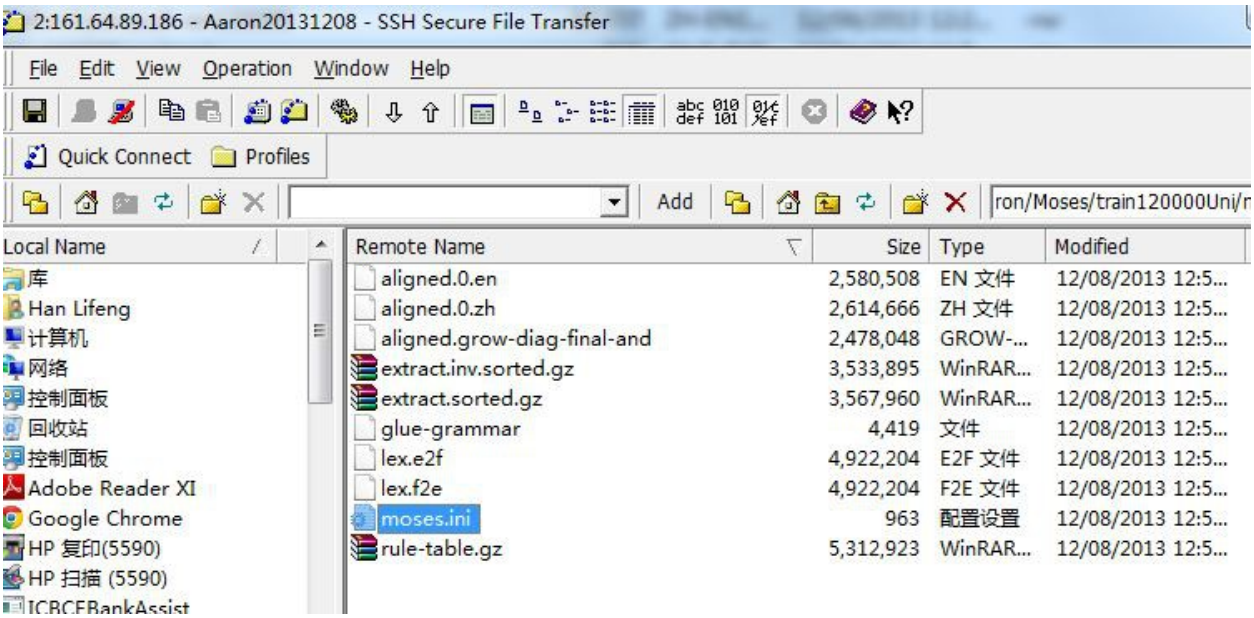
Training finished as:



The file-folder will generate:







Uni120000-Training time from “preparing corpus @ Sun Dec 8 12:42:38 CST 2013” to “create moses.ini @ Sun Dec 8 12:51:57 CST 2013”—total: 9mins+19s

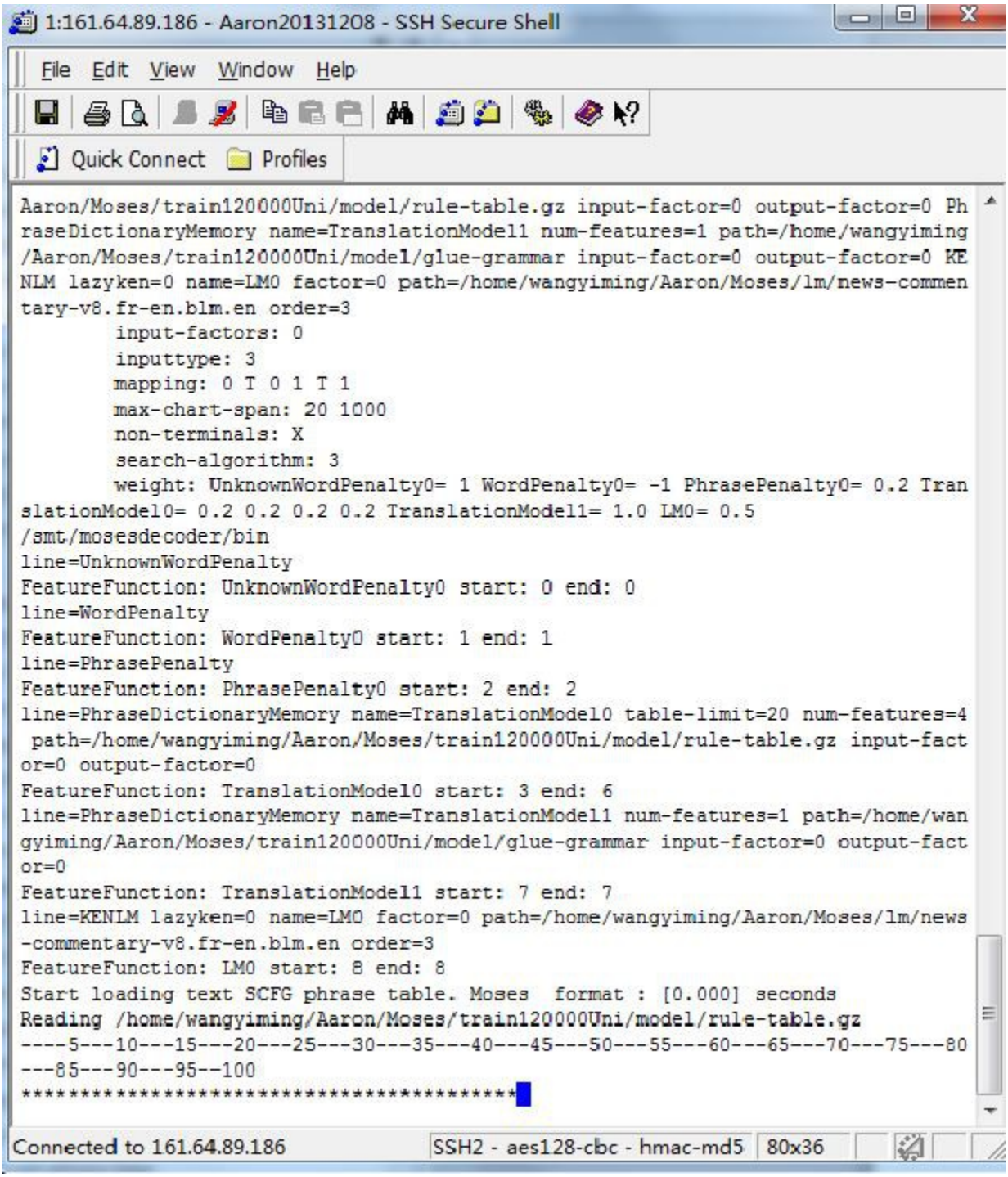
Compare-120000-training time from “(1) preparing corpus @ Fri Dec 6 12:19:27 CST 2013” to “(9) create moses.ini @ Fri Dec 6 12:28:49 CST 2013”—total: 9mins+22s

Testing-4-uni-186sever:

Use the following command for testing the NIST-MT03 testing CN-.EN data

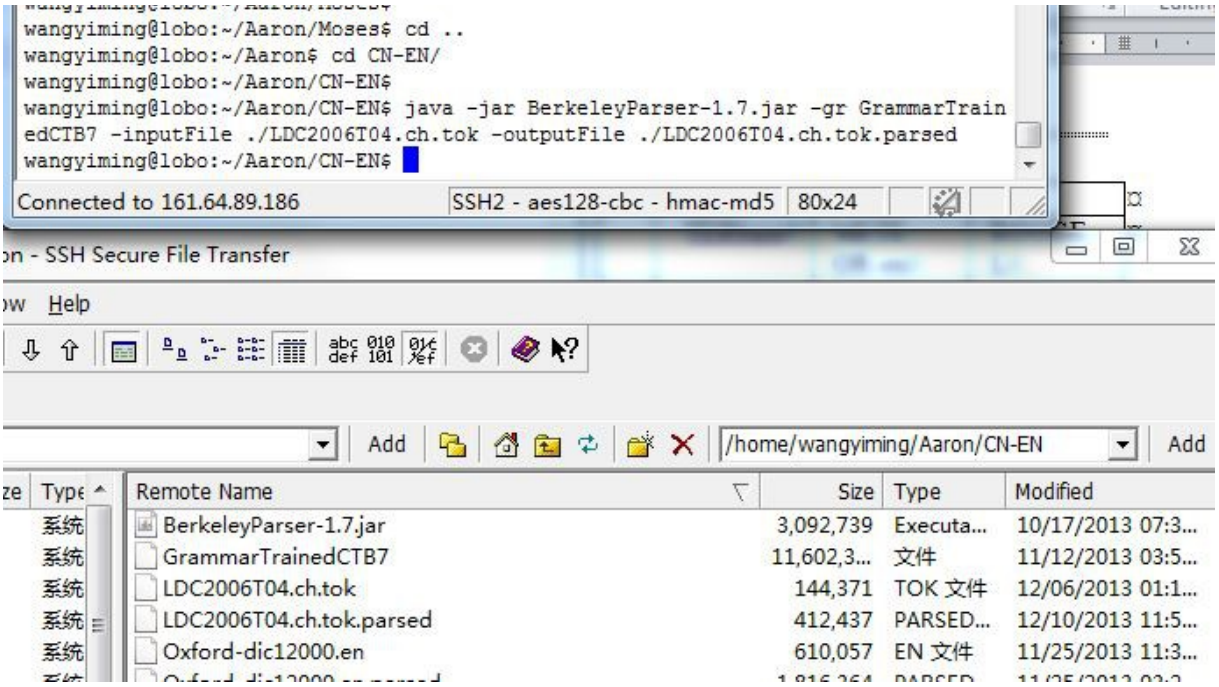
```
wangyiming@lobo:~/Aaron/Moses$
wangyiming@lobo:~/Aaron/Moses$ /smt/mosesdecoder/bin/moses_chart -f ./train12000
0Uni/model/moses.ini < ./corpus/test/LDC2006T04.ch.tok > ./corpus/test/LDC2006T0
4.ch-en.Uni.ttt
```

Testing runs as:

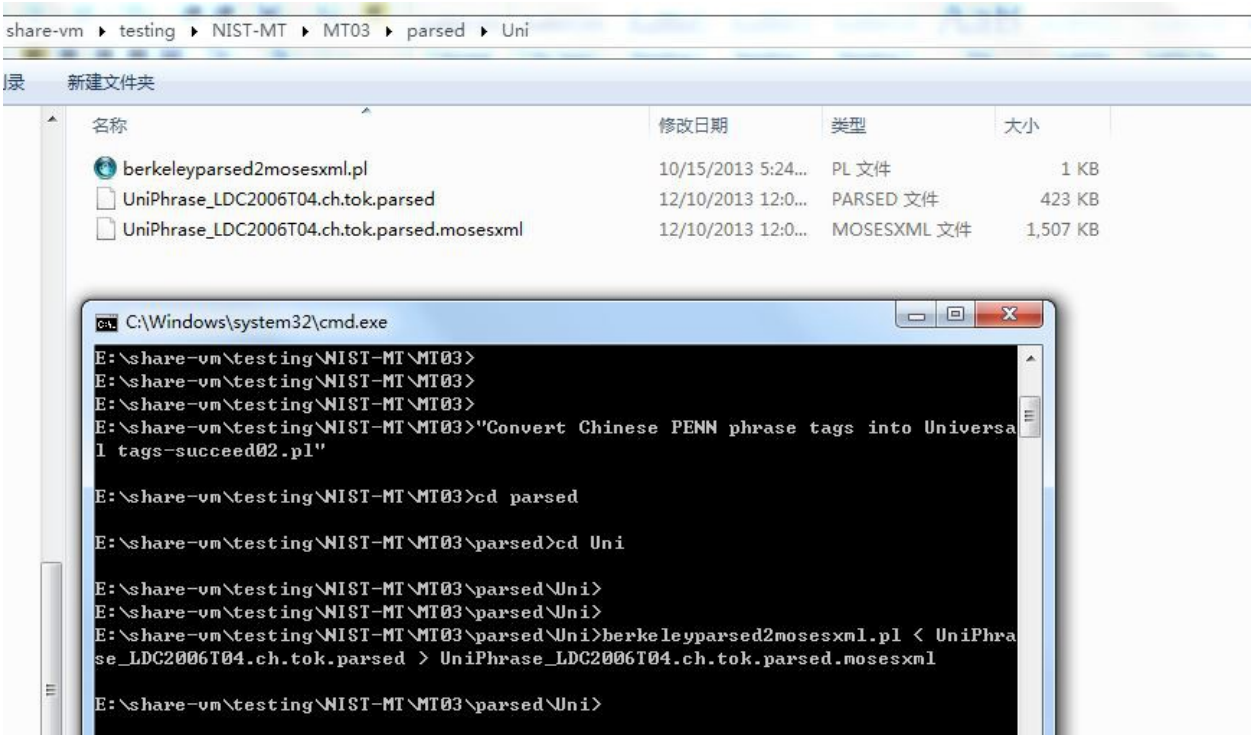


The testing translation result is totally the same with the result using the original phrase tags trained model. Why? Is that due to the testing input is plain sentences?

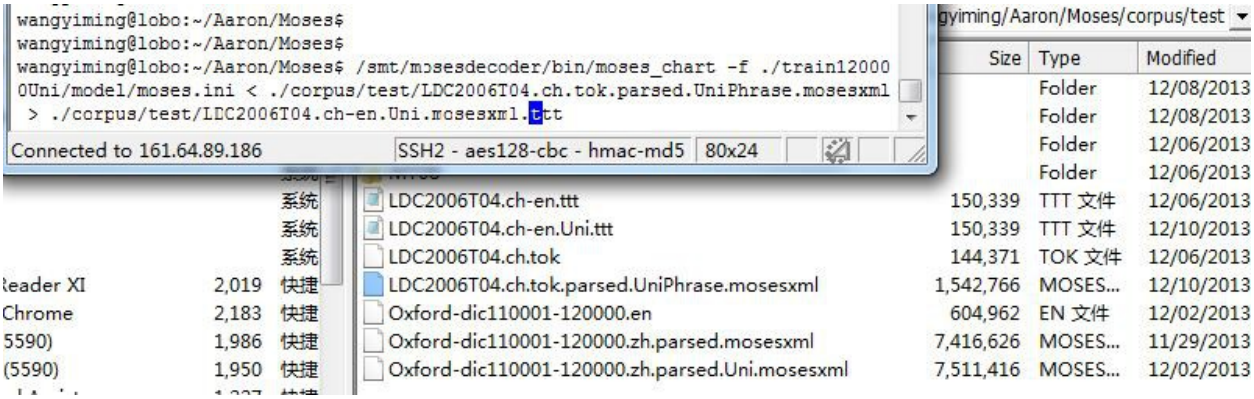
Parse the input sentences, and have a check of the re-testing result. Use the following command to parse the NIST-MT03 testing sentences:



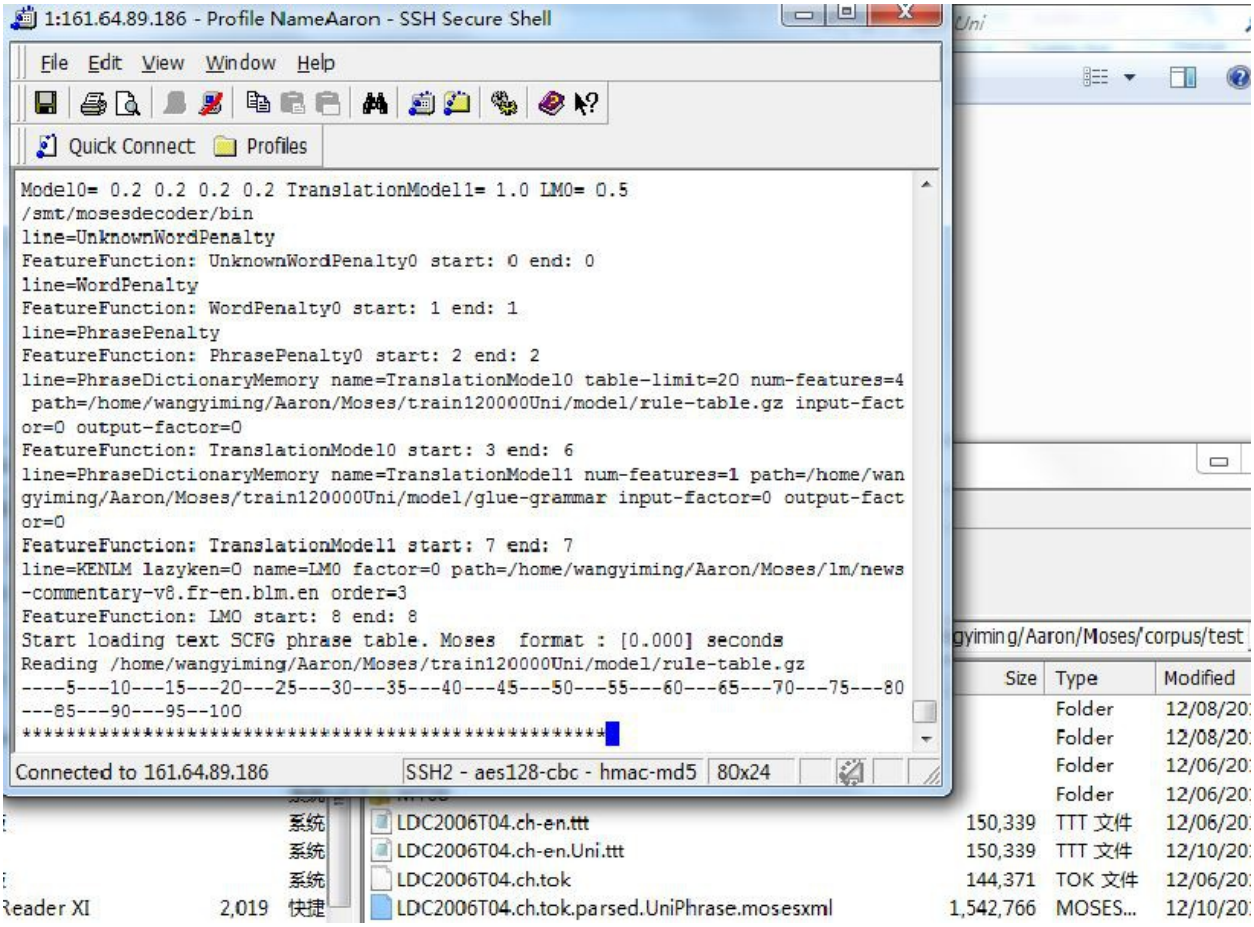
Replace the parsed phrase tags into uni-phrase tags and Convert the prased format into moses.xml format:



Run the translation testing again:



It runs as:



The translation result is different with the translation that uses the plain input text.

Secondly, we used the NIST-MT04 CN-EN test data:

```
wangyiming@lobo:~/Aaron/Moses$
wangyiming@lobo:~/Aaron/Moses$ /smt/mosesdecoder/bin/moses_chart -f ./train12000
0Uni/model/moses.ini < ./corpus/test/MT04/LDC2006E43.ch.tok > ./corpus/test/MT04
/LDC2006E43.ch-en.Uni.ttt
```

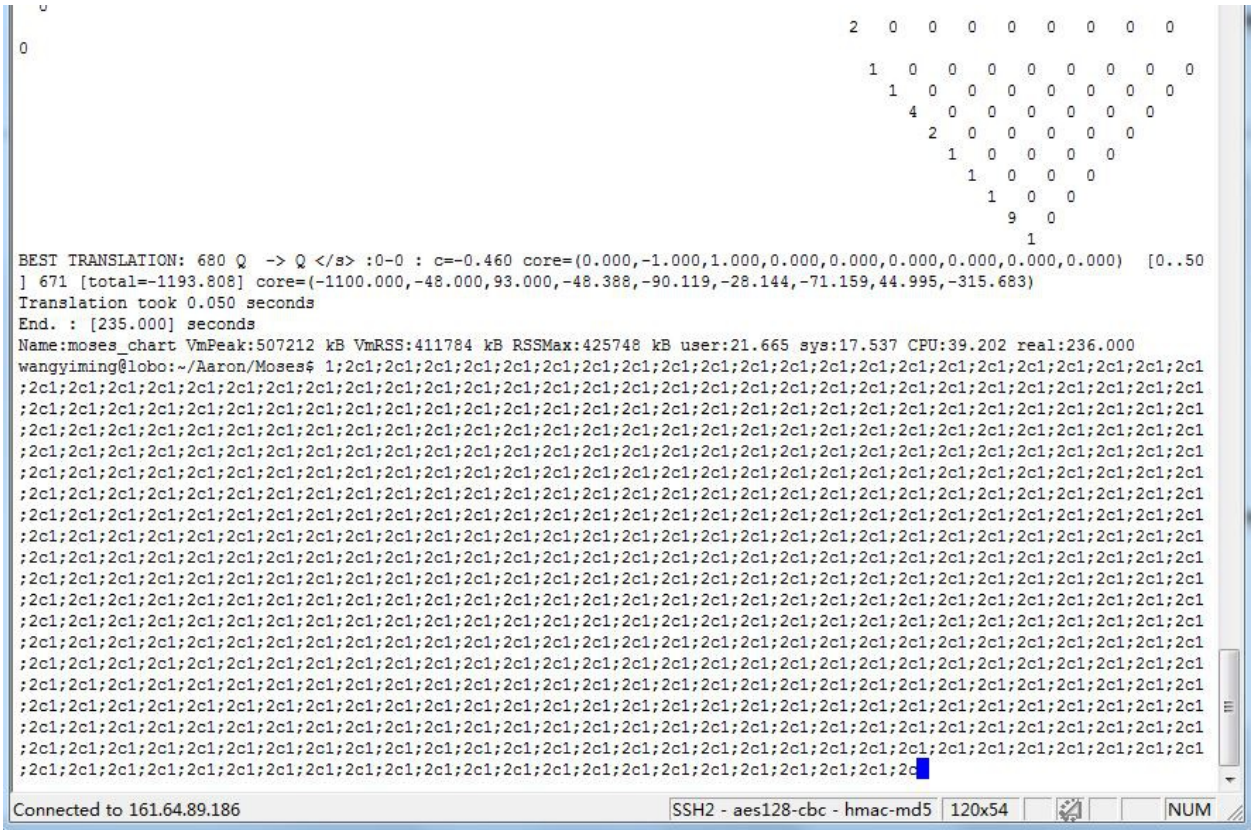
Connected to 161.64.89.186 | SSH2 - aes128-cbc - hmac-md5 | 80x40

Testing once more using the parsed corpus:

```
wangyiming@lobo:~/Aaron/Moses$
wangyiming@lobo:~/Aaron/Moses$ /smt/mosesdecoder/bin/moses_chart -f ./train120000Uni/model/moses.ini < ./corpus/test/MT0
4/LDC2006E43.ch.tok.parsed.UniPhrase.mosesxml > ./corpus/test/MT04/LDC2006E43.ch-en.Uniphrase.mosesxml.ttt
Defined parameters (per moses.ini or switch):
 config: ./train120000Uni/model/moses.ini
 cube-pruning-pop-limit: 1000
 feature: UnknownWordPenalty WordPenalty PhrasePenalty PhraseDictionaryMemory name=TranslationModel0 table-limit=20 num-
features=4 path=/home/wangyiming/Aaron/Moses/train120000Uni/model/rule-table.gz input-factor=0 output-factor=0 PhraseDic
tionaryMemory name=TranslationModel1 num-features=1 path=/home/wangyiming/Aaron/Moses/train120000Uni/model/glue-grammar
input-factor=0 output-factor=0 KENLM lazyken=0 name=LM0 factor=0 path=/home/wangyiming/Aaron/Moses/lm/news-commentary-v8
.fr-en.blm.en order=3
 input-factors: 0
 inputtype: 3
 mapping: 0 T 0 1 T 1
 max-chart-span: 20 1000
 non-terminals: X
 search-algorithm: 3
 weight: UnknownWordPenalty0= 1 WordPenalty0= -1 PhrasePenalty0= 0.2 TranslationModel0= 0.2 0.2 0.2 0.2 TranslationModel
1= 1.0 LM0= 0.5
/smt/mosesdecoder/bin
line=UnknownWordPenalty
FeatureFunction: UnknownWordPenalty0 start: 0 end: 0
line=WordPenalty
FeatureFunction: WordPenalty0 start: 1 end: 1
line=PhrasePenalty
FeatureFunction: PhrasePenalty0 start: 2 end: 2
line=PhraseDictionaryMemory name=TranslationModel0 table-limit=20 num-features=4 path=/home/wangyiming/Aaron/Moses/train
120000Uni/model/rule-table.gz input-factor=0 output-factor=0
FeatureFunction: TranslationModel0 start: 3 end: 6
line=PhraseDictionaryMemory name=TranslationModel1 num-features=1 path=/home/wangyiming/Aaron/Moses/train120000Uni/model
/glue-grammar input-factor=0 output-factor=0
FeatureFunction: TranslationModel1 start: 7 end: 7
line=KENLM lazyken=0 name=LM0 factor=0 path=/home/wangyiming/Aaron/Moses/lm/news-commentary-v8.fr-en.blm.en order=3
FeatureFunction: LM0 start: 8 end: 8
Start loading text SCFG phrase table. Moses  format : [0.000] seconds
Reading /home/wangyiming/Aaron/Moses/train120000Uni/model/rule-table.gz
----5---10---15---20---25---30---35---40---45---50---55---60---65---70---75---80---85---90---95--100
**************************
```

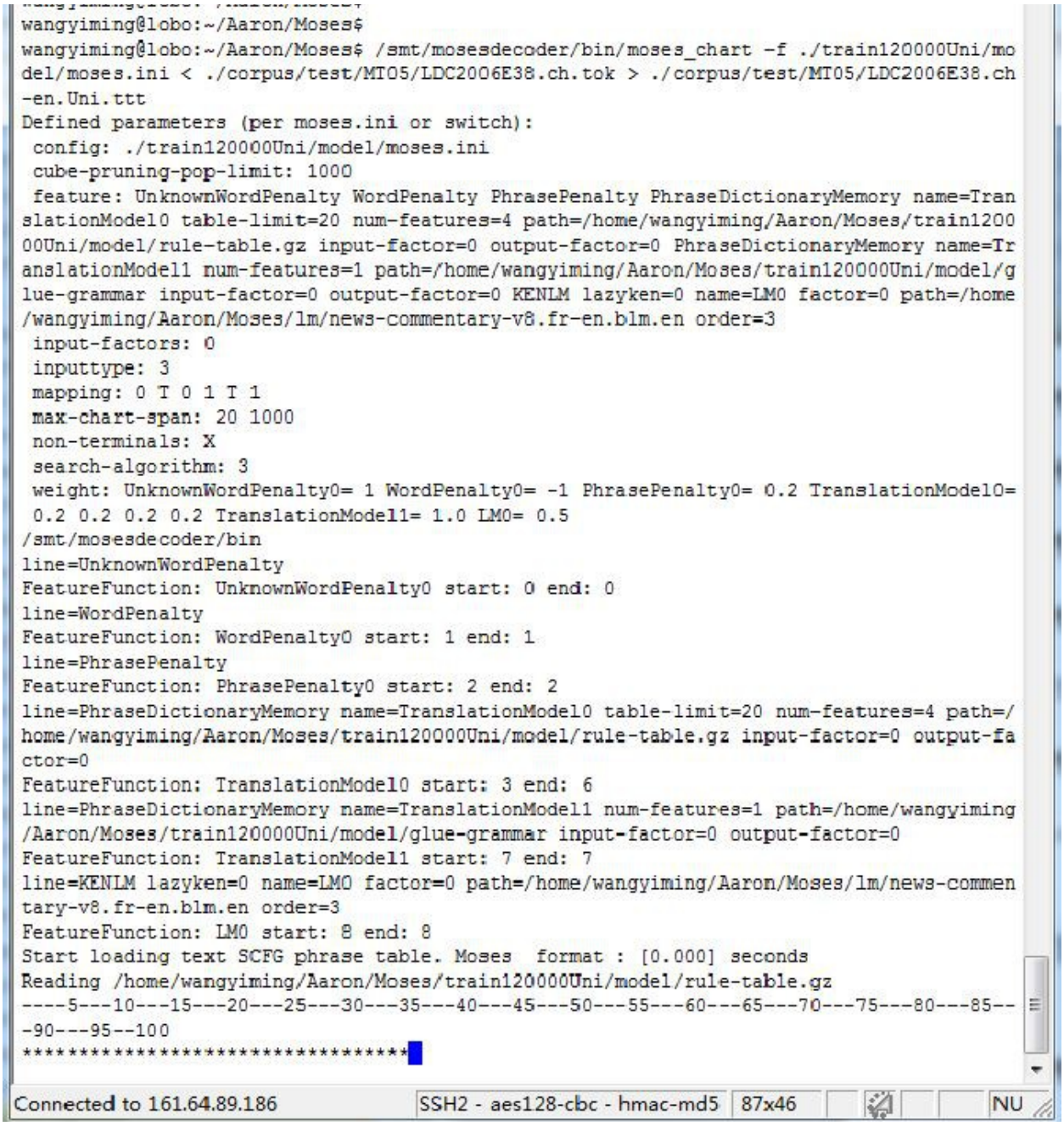
Connected to 161.64.89.186 | SSH2 - aes128-cbc - hmac-md5 | 120x54 | NUM

It finished as:

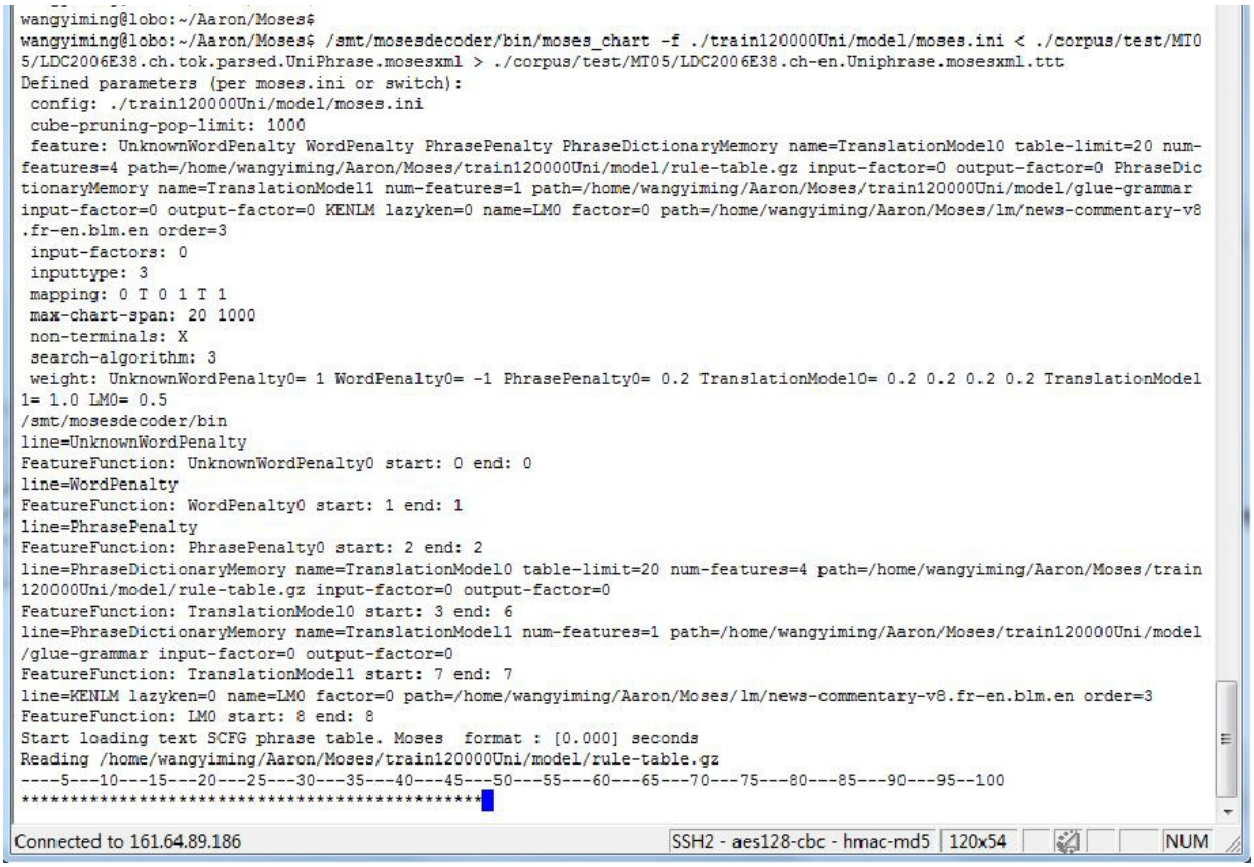


Thirdly, test the NIST-MT05 testing data:

The testing runs as:



Testing once more using the parsed corpus:



==========

Build CN-EN tree-to-string translation using moses:

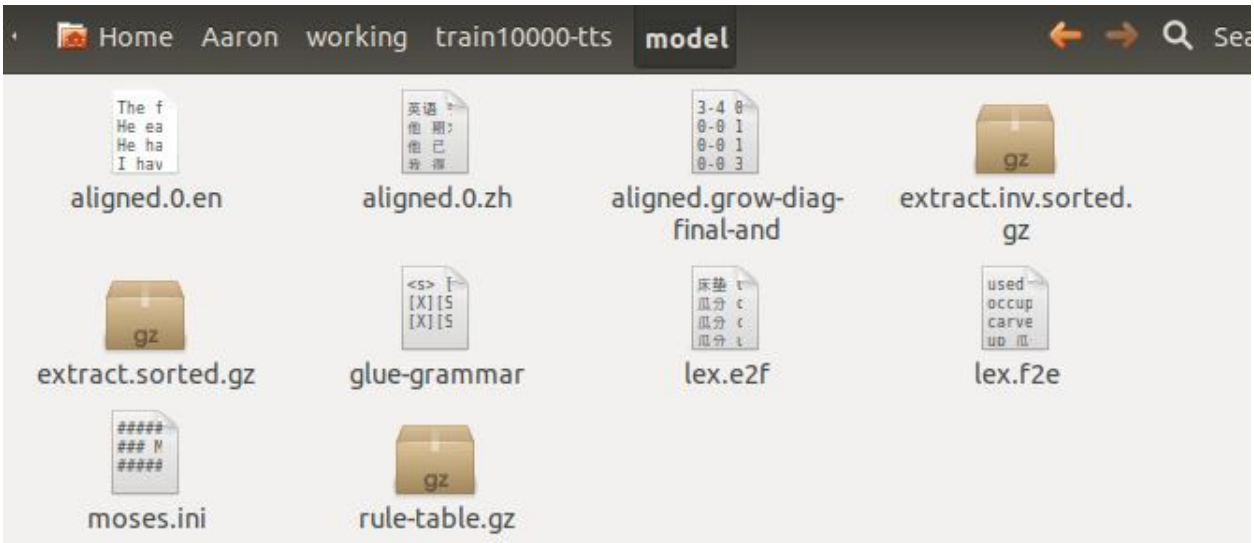
Training translation Corpus: source-parsed, target-parsed

Training:

```
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$ ~/Aaron/Moses/mosesdecoder/scripts/training/train-model.perl -root-dir train10000-tts -corpus ~/Aaron/corpus/training/train10000Oxf/Oxford-dic10000.parsed.mosesxml -f zh -e en -alignment grow-diag-final-and -hierarchical -glue-grammar --source-syntax -lm 0:3:$HOME/Aaron/lm/news-commentary-v8.fr-en.blm.en:8 -external-bin-dir ~/Aaron/Moses/mosesdecoder/tools >& training.out.Oxford.parse.zh-en10000-tts &
```

```
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$ ~/Aaron/Moses/mosesdecoder/scripts/tra
ining/train-model.perl -root-dir train10000-tts -corpus ~/Aaron/corpus/training/
train10000Oxf/Oxford-dic10000.parsed.mosesxml -f zh -e en -alignment grow-diag-f
inal-and -hierarchical -glue-grammar --source-syntax -lm 0:3:$HOME/Aaron/lm/news
-commentary-v8.fr-en.blm.en:8 -external-bin-dir ~/Aaron/Moses/mosesdecoder/tools
 >& training.out.Oxford.parse.zh-en10000-tts &
[1] 2354
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$ jobs
[1]+  Running                 ~/Aaron/Moses/mosesdecoder/scripts/training/train-
model.perl -root-dir train10000-tts -corpus ~/Aaron/corpus/training/train10000Ox
f/Oxford-dic10000.parsed.mosesxml -f zh -e en -alignment grow-diag-final-and -hi
erarchical -glue-grammar --source-syntax -lm 0:3:$HOME/Aaron/lm/news-commentary-
v8.fr-en.blm.en:8 -external-bin-dir ~/Aaron/Moses/mosesdecoder/tools &>training.
out.Oxford.parse.zh-en10000-tts &
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$
```

```
   11 root      RT    0      0     0     0 S     0  0.0   0:00.02 watchdog/1
   12 root       0 -20      0     0     0 S     0  0.0   0:00.00 cpuset
[1]+  Done                    ~/Aaron/Moses/mosesdecoder/scripts/training/train-
model.perl -root-dir train10000-tts -corpus ~/Aaron/corpus/training/train10000Ox
f/Oxford-dic10000.parsed.mosesxml -f zh -e en -alignment grow-diag-final-and -hi
erarchical -glue-grammar --source-syntax -lm 0:3:$HOME/Aaron/lm/news-commentary-
v8.fr-en.blm.en:8 -external-bin-dir ~/Aaron/Moses/mosesdecoder/tools &>training.
out.Oxford.parse.zh-en10000-tts
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$
```



Testing:

```
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$ ~/Aaron/Moses/mosesdecoder/bin/moses_c
hart -f train10000-tts/model/moses.ini < ~/Aaron/corpus/test/tree-to-string/Oxfo
rd-dic11001-12000.zh.parsed.mosesxml > Oxford-dic11001-12000.zh.parsed.tts-out
```

```
                                1   0   0   0
                                  1   0   0
                                    4   0
                                      1
BEST TRANSLATION: 200 S  -> S </s> :0-0 : c=-0.460 core=(0.000,-1.000,1.000,0.00
0,0.000,0.000,0.000,0.000,0.000)  [0..18] 196 [total=-316.465] core=(-300.000,-1
9.000,36.000,-70.037,0.000,-32.019,0.000,16.998,-78.504)
Translation took 0.020 seconds
End. : [22.000] seconds
Name:moses_chart        VmPeak:319520 kB        VmRSS:161260 kB RSSMax:162652 kB
user:2.412      sys:9.941       CPU:12.353      real:22.394
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$
```

commands.txt ✖ | Oxford-dic11001-12000.zh.parsed.tts-out ✖

```
</tree> label="TOP"> label="S"> label="JJ"> Hit label="SINV"> </tree> label=","> </tree>
<tree label="TOP"> 皈依 基督教 </tree>
十字架 label="TOP"> 基督教 <tree 象征 </tree>
<tree label="VBP"> label="NN"> intruder 异教徒 宣传 基督教 </tree>
<tree label="S"> 改 label="VP"> 基督教 <tree </tree>
基督教 义 label="S"> <tree 三位一体 指 <tree label="TOP"> 圣父 tasteless 圣子 colorless 圣灵 </tree>
十字架 label="TOP"> 基督教 <tree label="PRP"> 徵 </tree>
label="IN"> label="TOP"> label="TOP"> 克里斯蒂娜 <tree <tree <tree 继任 label="TOP"> </tree>
恭祝 圣诞 <tree Ramsay 贺新禧 label="FRAG">
label="FRAG"> <tree label="NNP"> 互 <tree 贺卡 colorless label="PP"> </tree>
label="WRB"> <tree label="NNP"> label="S"> <tree label="TOP"> label="TOP"> <tree label="NP"> </tree>
they </tree> <tree label="NNP"> label="TOP"> <tree label="S"> <tree </tree>
<tree label="TOP"> <tree label="TOP"> <tree label="NNP"> </tree> label="WHNP">
label="NNP"> label="WHNP"> 停 label="S"> <tree </tree>
label="NNP"> sergeant label="S"> 火鸡 label="TOP"> label="S"> <tree <tree label="NP"> </tree>
```

Build CN-EN tree-to-string translation using moses-2:

Training translation corpus: source-parsed, target-plain

Training:

```
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$ ~/Aaron/Moses/mosesdecoder/scripts/tra
ining/train-model.perl -root-dir train10000-tts-2 -corpus ~/Aaron/corpus/trainin
g/train10000Oxf/tts/cn-en/Oxford-dic10000-tts-cn-en -f zh -e en -alignment grow-
diag-final-and -hierarchical -glue-grammar --source-syntax -lm 0:3:$HOME/Aaron/l
m/news-commentary-v8.fr-en.blm.en:8 -external-bin-dir ~/Aaron/Moses/mosesdecoder
/tools >& translation-Oxford-tts-zh-en10000 &
```

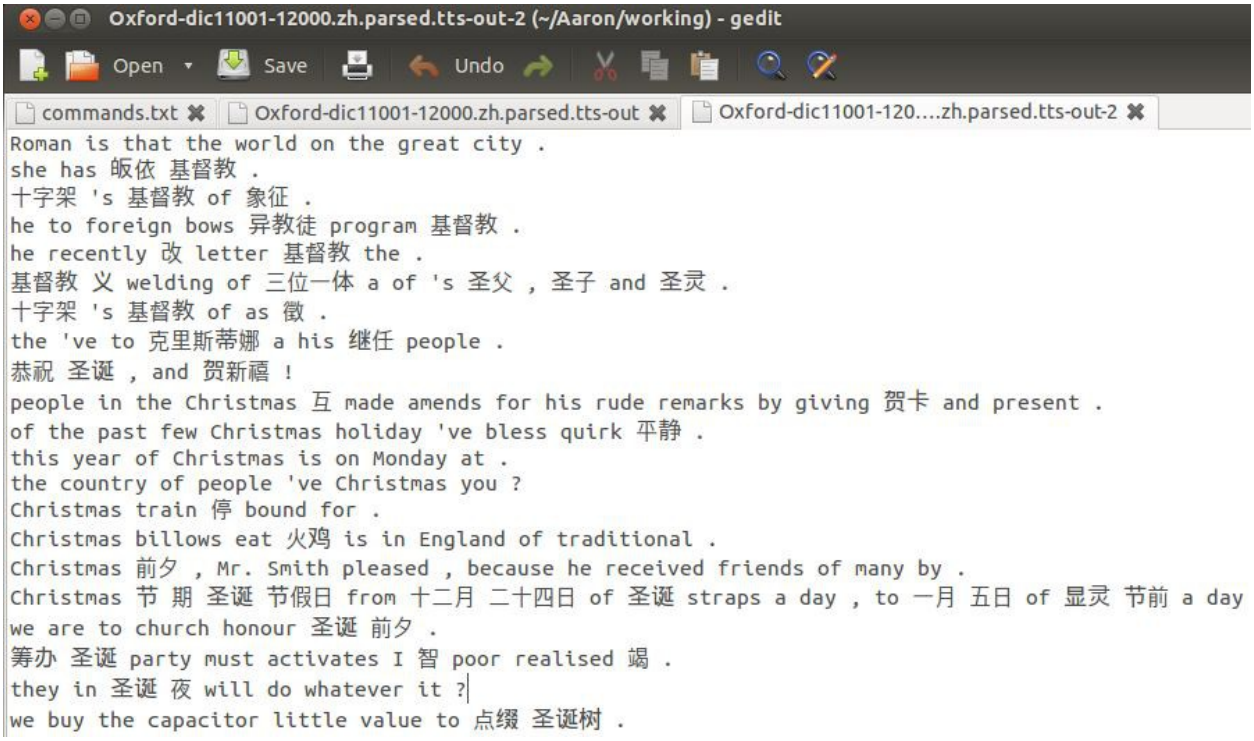
```
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$ ~/Aaron/Moses/mosesdecoder/scripts/tra
ining/train-model.perl -root-dir train10000-tts-2 -corpus ~/Aaron/corpus/trainin
g/train10000Oxf/tts/cn-en/Oxford-dic10000-tts-cn-en -f zh -e en -alignment grow-
diag-final-and -hierarchical -glue-grammar --source-syntax -lm 0:3:$HOME/Aaron/l
m/news-commentary-v8.fr-en.blm.en:8 -external-bin-dir ~/Aaron/Moses/mosesdecoder
/tools >& translation-Oxford-tts-zh-en10000 &
[1] 3684
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$ jobs
[1]+  Running                 ~/Aaron/Moses/mosesdecoder/scripts/training/train-
model.perl -root-dir train10000-tts-2 -corpus ~/Aaron/corpus/training/train10000
Oxf/tts/cn-en/Oxford-dic10000-tts-cn-en -f zh -e en -alignment grow-diag-final-a
nd -hierarchical -glue-grammar --source-syntax -lm 0:3:$HOME/Aaron/lm/news-comme
ntary-v8.fr-en.blm.en:8 -external-bin-dir ~/Aaron/Moses/mosesdecoder/tools &>tra
nslation-Oxford-tts-zh-en10000 &
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$
```

```
   11 root      RT   0     0    0     0 S    0  0.0   0:01.37 watchdog/1
   12 root       0 -20     0    0     0 S    0  0.0   0:00.00 cpuset
[1]+  Done                    ~/Aaron/Moses/mosesdecoder/scripts/training/train-
model.perl -root-dir train10000-tts-2 -corpus ~/Aaron/corpus/training/train10000
Oxf/tts/cn-en/Oxford-dic10000-tts-cn-en -f zh -e en -alignment grow-diag-final-a
nd -hierarchical -glue-grammar --source-syntax -lm 0:3:$HOME/Aaron/lm/news-comme
ntary-v8.fr-en.blm.en:8 -external-bin-dir ~/Aaron/Moses/mosesdecoder/tools &>tra
nslation-Oxford-tts-zh-en10000
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$
```

Testing:

```
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$ ~/Aaron/Moses/mosesdecoder/bin/moses_c
hart -f train10000-tts-2/model/moses.ini < ~/Aaron/corpus/test/tree-to-string/Ox
ford-dic11001-12000.zh.parsed.mosesxml > Oxford-dic11001-12000.zh.parsed.tts-out
-2
```

```
                                1   0   0   0   0   0
                              5   0   0   0   0
                                1   0   0   0
                                  1   0   0
                                    15   0
                                      1
BEST TRANSLATION: 954 S  -> S </s> :0-0 : c=-0.460 core=(0.000,-1.000,1.000,0.00
0,0.000,0.000,0.000,0.000,0.000)  [0..18] 937 [total=-323.282] core=(-300.000,-1
8.000,34.000,-19.144,-28.310,-12.946,-17.314,15.998,-97.074)
Translation took 0.020 seconds
End. : [25.000] seconds
Name:moses_chart        VmPeak:265848 kB        VmRSS:121316 kB RSSMax:123312 kB
user:4.048      sys:10.593      CPU:14.641      real:24.648
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$
```



Evaluation:

Using the following command, we calculate the BLEU score of the 1000 translated sentences:

Export result and evaluate using Asiya-tool?

Build CN-EN tree-to-string translation using moses-Uni:

Training translation corpus: source-parsed, target-plain

Training:

```
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$ ~/Aaron/Moses/mosesdecoder/scripts/tra
ining/train-model.perl -root-dir train10000-tts-Uni -corpus ~/Aaron/corpus/train
ing/train10000Oxf/tts/cn-en/Oxford-dic10000-tts-cn-en-Uni -f zh -e en -alignment
 grow-diag-final-and -hierarchical -glue-grammar --source-syntax -lm 0:3:$HOME/A
aron/lm/news-commentary-v8.fr-en.blm.en:8 -external-bin-dir ~/Aaron/Moses/mosesd
ecoder/tools >& translation-Oxford-tts-Uni-zh-en10000 &
```

```
  PID USER      PR  NI  VIRT  RES  SHR S %CPU %MEM    TIME+  COMMAND
 3924 nlp2ct    20   0 24460  12m 1460 R  100  0.2   0:14.18 mkcls
  940 root      20   0  388m 151m  12m S    1  2.3   8:43.27 Xorg
 1660 nlp2ct    20   0 1322m  84m  34m S    1  1.3  11:31.69 compiz
 2222 nlp2ct    20   0  662m  43m  20m S    1  0.7   0:15.62 gedit
    1 root      20   0 24436 2360 1352 S    0  0.0   0:01.15 init
    2 root      20   0     0    0    0 S    0  0.0   0:00.06 kthreadd
    3 root      20   0     0    0    0 S    0  0.0   0:06.80 ksoftirqd/0
    5 root      20   0     0    0    0 S    0  0.0   0:00.89 kworker/u:0
```

```
   17 root      20   0     0    0    0 S    0  0.0   0:01.57 sync_supers
[1]+  Done                    ~/Aaron/Moses/mosesdecoder/scripts/training/train-
model.perl -root-dir train10000-tts-Uni -corpus ~/Aaron/corpus/training/train100
00Oxf/tts/cn-en/Oxford-dic10000-tts-cn-en-Uni -f zh -e en -alignment grow-diag-f
inal-and -hierarchical -glue-grammar --source-syntax -lm 0:3:$HOME/Aaron/lm/news
-commentary-v8.fr-en.blm.en:8 -external-bin-dir ~/Aaron/Moses/mosesdecoder/tools
 &>translation-Oxford-tts-Uni-zh-en10000
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$
```

Testing:

```
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$ ~/Aaron/Moses/mosesdecoder/bin/moses_c
hart -f train10000-tts-Uni/model/moses.ini < ~/Aaron/corpus/test/tree-to-string/
Oxford-dic11001-12000.zh.parsed.Uni.mosesxml > Oxford-dic11001-12000.zh.parsed.t
ts-out-Uni
```

```
                              5   0   0   0   0
                                1   0   0   0
                                  1   0   0
                                    15   0
                                       1
BEST TRANSLATION: 1143 S  -> S </s> :0-0 : c=-0.460 core=(0.000,-1.000,1.000,0.0
00,0.000,0.000,0.000,0.000,0.000)  [0..18] 1122 [total=-323.282] core=(-300.000,
-18.000,34.000,-19.145,-28.310,-12.946,-17.314,15.998,-97.074)
Translation took 0.030 seconds
End. : [26.000] seconds
Name:moses_chart        VmPeak:265568 kB        VmRSS:115248 kB RSSMax:122844 kB
user:4.564      sys:11.157      CPU:15.721      real:26.023
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$
```

Oxford-dic11001-120...h.parsed.tts-out-Uni

```
Roman is that the world on the great city .
she has 皈依 基督教 .
十字架 's 基督教 of 象征 .
he to foreign bows 异教徒 program 基督教 .
he recently 改 letter 基督教 the .
基督教 义 welding of 三位一体 a of 's 圣父 , 圣子 and 圣灵 .
十字架 's 基督教 of as 徵 .
the 've to 克里斯蒂娜 a his 继任 people .
恭祝 圣诞 , and 贺新禧 !
people in the Christmas 互 made amends for his rude remarks by giving 贺卡 and present .
of the past few Christmas holiday 've bless quirk 平静 .
this year of Christmas is on Monday at .
the country of people 've Christmas you ?
Christmas train 停 bound for .
Christmas billows eat 火鸡 is in England of traditional .
Christmas 前夕 , Mr. Smith pleased , because he received friends of many by .
Christmas 节 期 圣诞 节假日 from 十二月 二十四日 of 圣诞 straps a day , to 一月 五日 of 显灵 节前 a day
we are to church honour 圣诞 前夕 .
筹办 圣诞 party must activates I 智 poor realised 竭 .
they in 圣诞 夜 will do whatever it ?
```

Tree-to-string: Using the ori-phrase and uni-phrase, the evaluation scores using Asiya-tool are different.

Build CN-EN string-to-tree translation using moses-ori:

Training translation Corpus: source-plain, target-parsed.

Training:

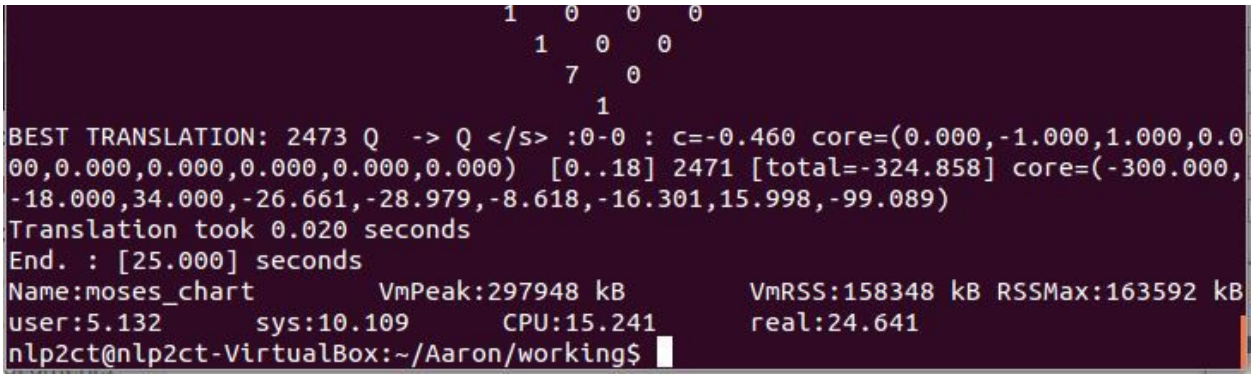
```
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$ ~/Aaron/Moses/mosesdecoder/scripts/tra
ining/train-model.perl -root-dir train10000-stt-ori -corpus ~/Aaron/corpus/train
ing/train10000Oxf/stt/cn-en/Oxford-dic10000.stt-cn-en-ori -f zh -e en -alignment
 grow-diag-final-and -hierarchical -glue-grammar --target-syntax -lm 0:3:$HOME/A
aron/lm/news-commentary-v8.fr-en.blm.en:8 -external-bin-dir ~/Aaron/Moses/mosesd
ecoder/tools >& translation-Oxford-stt-ori-zh-en10000 &
```

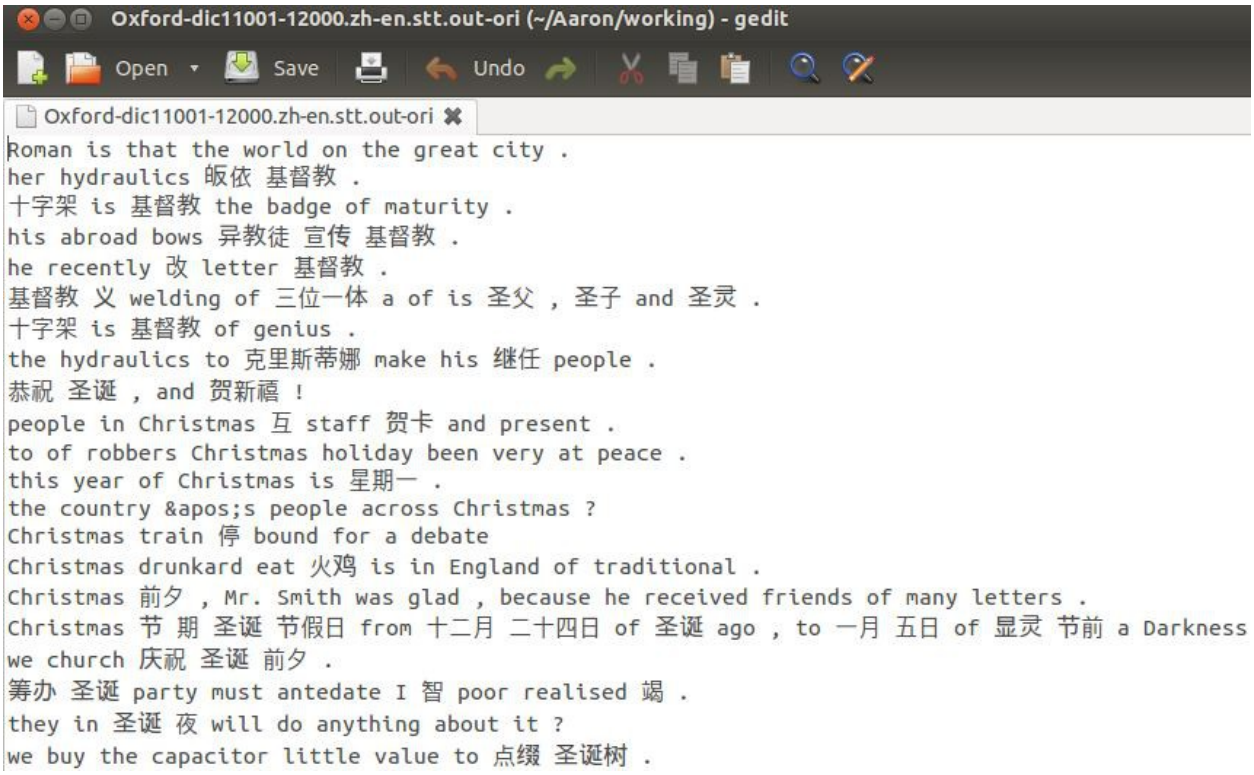
```
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$ ~/Aaron/Moses/mosesdecoder/scripts/tra
ining/train-model.perl -root-dir train10000-stt-ori -corpus ~/Aaron/corpus/train
ing/train10000Oxf/stt/cn-en/Oxford-dic10000.stt-cn-en-ori -f zh -e en -alignment
 grow-diag-final-and -hierarchical -glue-grammar --target-syntax -lm 0:3:$HOME/A
aron/lm/news-commentary-v8.fr-en.blm.en:8 -external-bin-dir ~/Aaron/Moses/mosesd
ecoder/tools >& translation-Oxford-stt-ori-zh-en10000 &
[1] 4124
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$ jobs
[1]+  Running                 ~/Aaron/Moses/mosesdecoder/scripts/training/train-
model.perl -root-dir train10000-stt-ori -corpus ~/Aaron/corpus/training/train100
00Oxf/stt/cn-en/Oxford-dic10000.stt-cn-en-ori -f zh -e en -alignment grow-diag-f
inal-and -hierarchical -glue-grammar --target-syntax -lm 0:3:$HOME/Aaron/lm/news
-commentary-v8.fr-en.blm.en:8 -external-bin-dir ~/Aaron/Moses/mosesdecoder/tools
 &>translation-Oxford-stt-ori-zh-en10000 &
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$
```

```
   10 root      20   0     0     0     0 S     0  0.0   0:03.53 ksoftirqd/1
   11 root      RT   0     0     0     0 S     0  0.0   0:01.54 watchdog/1
[1]+  Done                    ~/Aaron/Moses/mosesdecoder/scripts/training/train-
model.perl -root-dir train10000-stt-ori -corpus ~/Aaron/corpus/training/train100
00Oxf/stt/cn-en/Oxford-dic10000.stt-cn-en-ori -f zh -e en -alignment grow-diag-f
inal-and -hierarchical -glue-grammar --target-syntax -lm 0:3:$HOME/Aaron/lm/news
-commentary-v8.fr-en.blm.en:8 -external-bin-dir ~/Aaron/Moses/mosesdecoder/tools
 &>translation-Oxford-stt-ori-zh-en10000
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$
```

Testing:

```
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$ ~/Aaron/Moses/mosesdecoder/bin/moses_c
hart -f train10000-stt-ori/model/moses.ini < ~/Aaron/corpus/test/string-to-tree/
Oxford-dic11001-12000.zh > Oxford-dic11001-12000.zh-en.stt.out-ori
```





Build CN-EN string-to-tree translation using moses-uni:

Training translation Corpus: source-plain, target-parsed.

Training:

```
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$ ~/Aaron/Moses/mosesdecoder/scripts/tra
ining/train-model.perl -root-dir train10000-stt-uni -corpus ~/Aaron/corpus/train
ing/train10000Oxf/stt/cn-en/Oxford-dic10000.stt-cn-en-uni -f zh -e en -alignment
 grow-diag-final-and -hierarchical -glue-grammar --target-syntax -lm 0:3:$HOME/A
aron/lm/news-commentary-v8.fr-en.blm.en:8 -external-bin-dir ~/Aaron/Moses/mosesd
ecoder/tools >& translation-Oxford-stt-uni-zh-en10000 &
```

```
  PID USER      PR  NI    VIRT    RES    SHR S %CPU %MEM     TIME+ COMMAND
 4401 nlp2ct    20   0   37616   24m   1792 R  100  0.4   0:05.43 GIZA++
  940 root      20   0     393m 156m    12m S    1  2.4   9:34.05 Xorg
 1660 nlp2ct    20   0    1322m  84m    34m S    1  1.3  12:12.47 compiz
 1622 nlp2ct    20   0     109m 1576 1024 S    0  0.0   4:20.59 VBoxClient
 1682 nlp2ct    20   0    1461m  59m    21m S    0  0.9   1:19.87 nautilus
 1968 nlp2ct    20   0     530m  25m    16m S    0  0.4   1:14.63 gnome-terminal
```

```
   15 root       0 -20     0    0    0 S    0  0.0   0:00.00 netns
[1]+  Done                    ~/Aaron/Moses/mosesdecoder/scripts/training/train-
model.perl -root-dir train10000-stt-uni -corpus ~/Aaron/corpus/training/train100
00Oxf/stt/cn-en/Oxford-dic10000.stt-cn-en-uni -f zh -e en -alignment grow-diag-f
inal-and -hierarchical -glue-grammar --target-syntax -lm 0:3:$HOME/Aaron/lm/news
-commentary-v8.fr-en.blm.en:8 -external-bin-dir ~/Aaron/Moses/mosesdecoder/tools
 &>translation-Oxford-stt-uni-zh-en10000
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$
```

Testing:

```
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$ ~/Aaron/Moses/mosesdecoder/bin/moses_c
hart -f train10000-stt-uni/model/moses.ini < ~/Aaron/corpus/test/string-to-tree/
Oxford-dic11001-12000.zh > Oxford-dic11001-12000.zh-en.stt.out-uni
```

```
                              1   0   0   0
                                1   0   0
                                  7   0
                                    1
BEST TRANSLATION: 2469 Q  -> Q </s> :0-0 : c=-0.460 core=(0.000,-1.000,1.000,0.0
00,0.000,0.000,0.000,0.000,0.000)  [0..18] 2467 [total=-324.858] core=(-300.000,
-18.000,34.000,-26.661,-28.979,-8.618,-16.301,15.998,-99.089)
Translation took 0.030 seconds
End. : [25.000] seconds
Name:moses_chart        VmPeak:297124 kB        VmRSS:157604 kB RSSMax:163204 kB
user:5.412      sys:10.017      CPU:15.429      real:24.695
nlp2ct@nlp2ct-VirtualBox:~/Aaron/working$
```

Oxford-dic11001-12000.zh-en.stt.out-uni (~/Aaron/working) - gedit

Open Save Undo

Oxford-dic11001-12000.zh-en.stt.out-uni

```
Roman is that the world on the great city .
her hydraulics 皈依 基督教 .
十字架 is 基督教 the badge of maturity .
his abroad bows 异教徒 宣传 基督教 .
he recently 改 letter 基督教 .
基督教 义 welding of 三位一体 a of is 圣父 , 圣子 and 圣灵 .
十字架 is 基督教 of genius .
the hydraulics to 克里斯蒂娜 make his 继任 people .
恭祝 圣诞 , and 贺新禧 !
people in Christmas 互 staff 贺卡 and present .
to of robbers Christmas holiday been very at peace .
this year of Christmas is 星期一 .
the country &apos;s people across Christmas ?
Christmas train 停 bound for a debate
Christmas drunkard eat 火鸡 is in England of traditional .
Christmas 前夕 , Mr. Smith was glad , because he received friends of many letters .
Christmas 节 期 圣诞 节假日 from 十二月 二十四日 of 圣诞 ago , to 一月 五日 of 显灵 节前 a Darkness
we church 庆祝 圣诞 前夕 .
筹办 圣诞 party must antedate I 智 poor realised 竭 .
they in 圣诞 夜 will do anything about it ?
we buy the capacitor little value to 点缀 圣诞树 .
Somebody the the switch , 圣诞树 on all 灯 expenses 亮 the knotted .
```

String-to-tree: Using the ori-phrase and uni-phrase, the evaluation scores using Asiya-tool are the same, why?

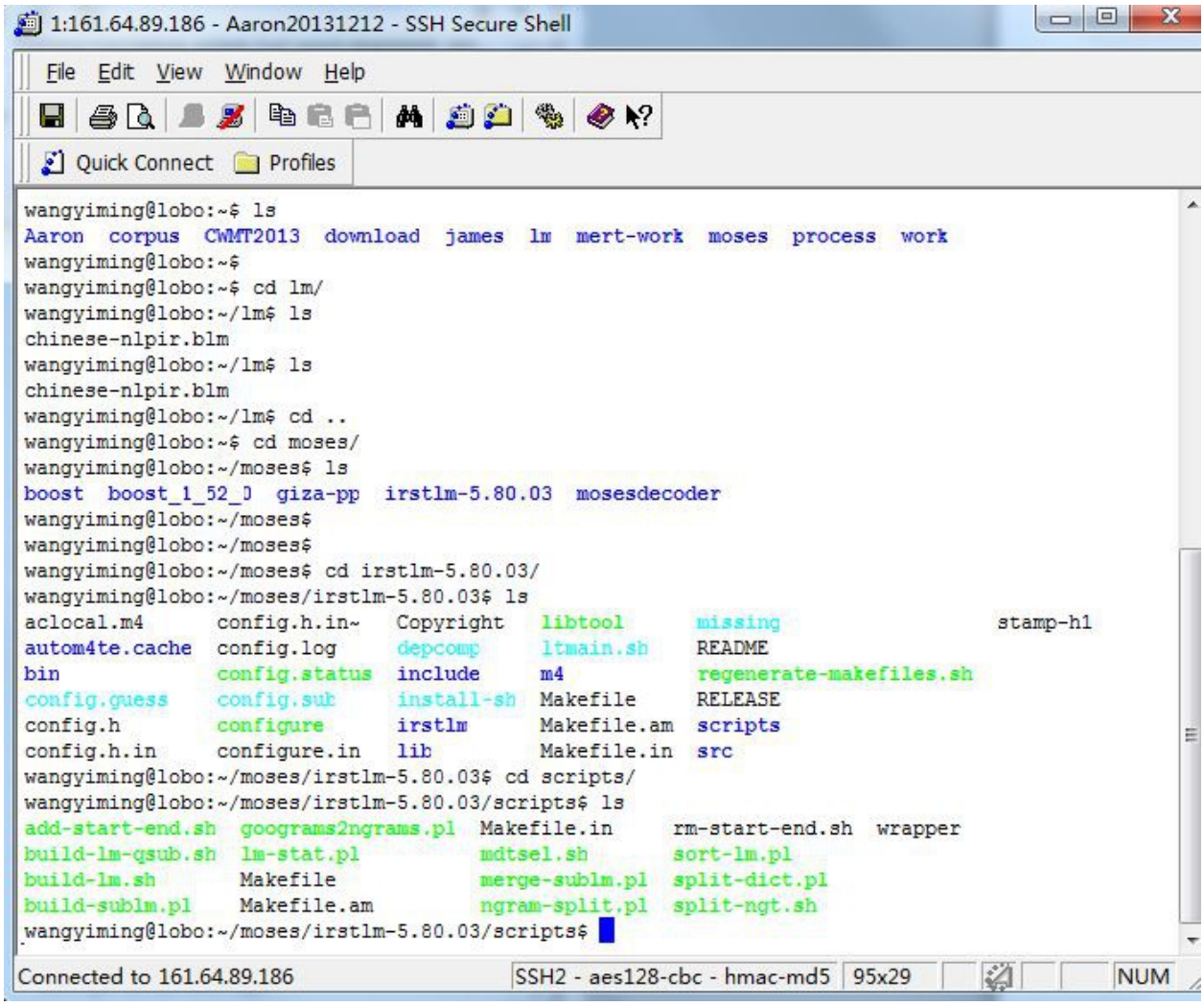
Build PT->CN tree-to-tree translation model with ori-phrase tags on 186-server:

Prepare corpus:

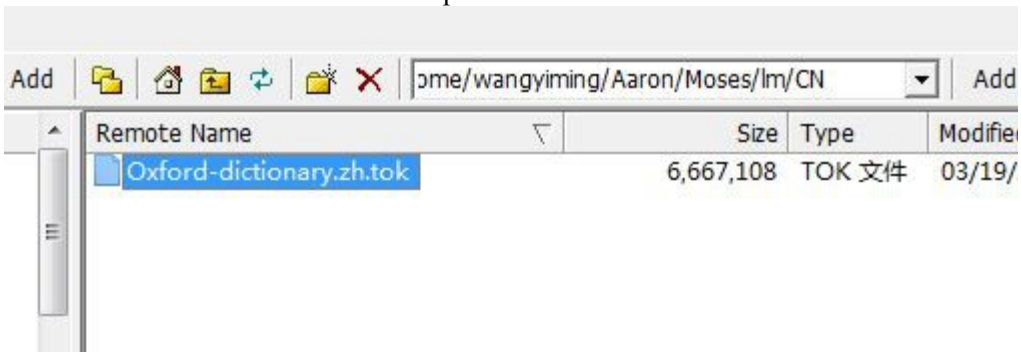
Train language model:

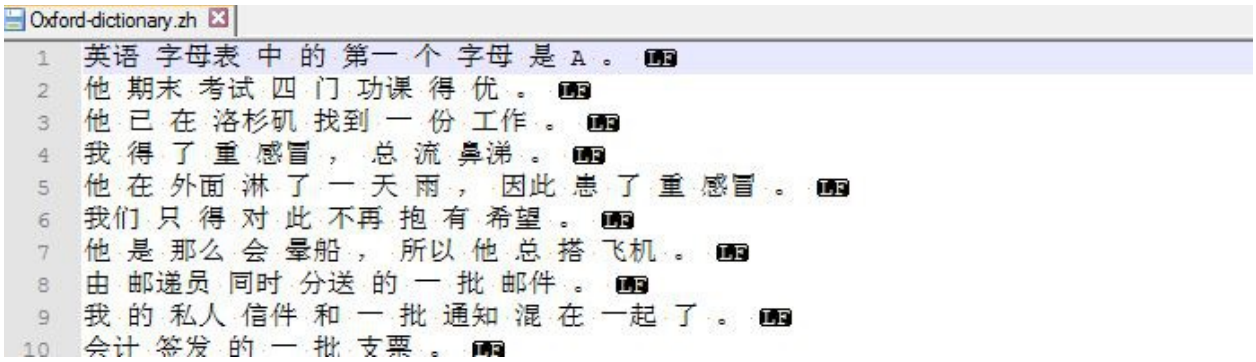
Train language model of CN uses the 120,000 Oxford simplified CN sentences.

Use the following command to check the moses installation on the server:

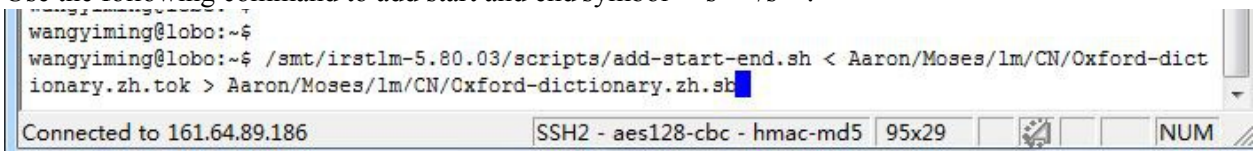


Put the 120000 sentences chinese corpus:

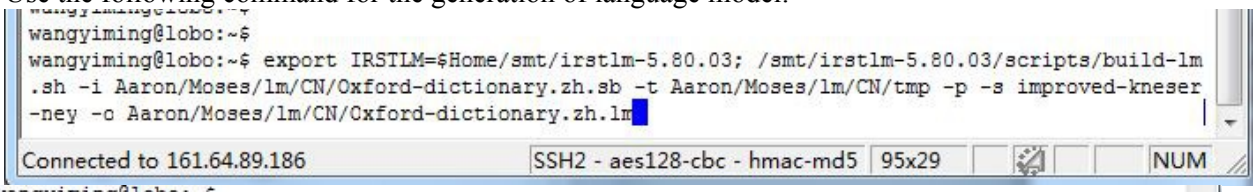




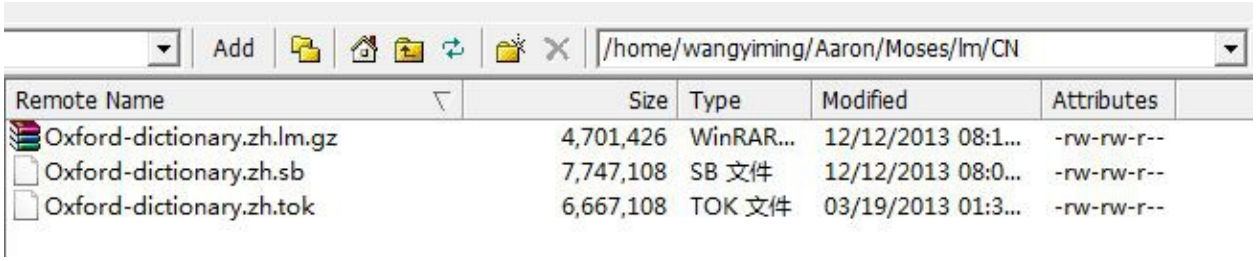
Use the following command to add start and end symbol "<s> </s>":



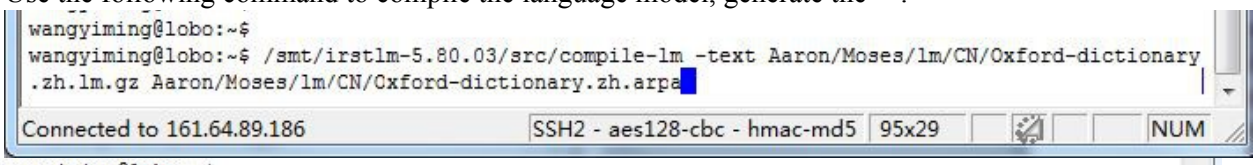
Use the following command for the generation of language model:

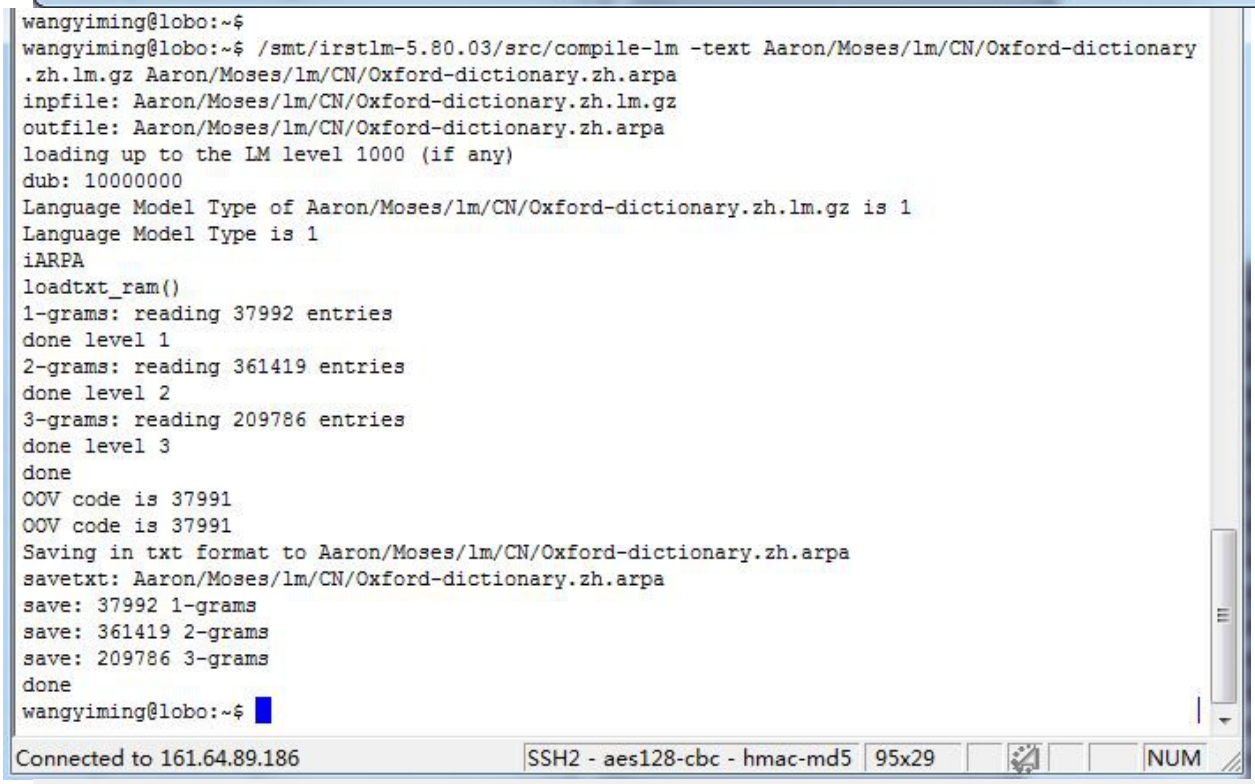


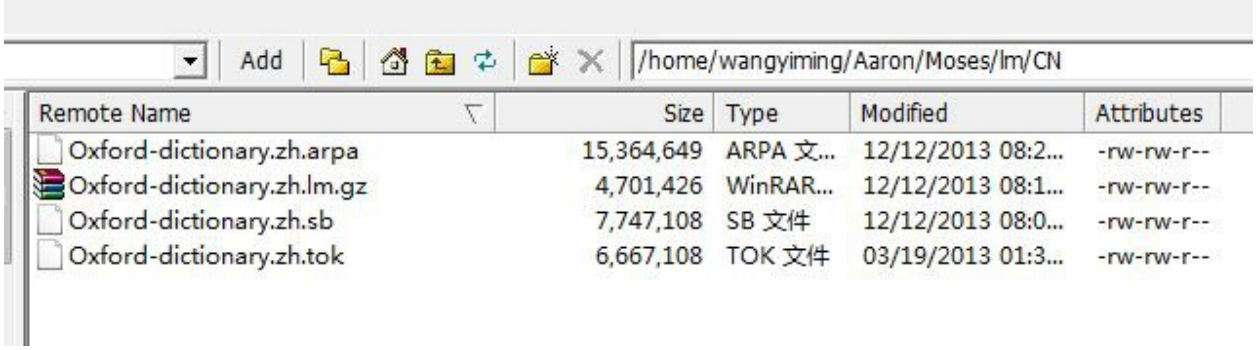
The generated language model:



Use the following command to compile the language model, generate the “”:







The content of the zh.arpa file:

Oxford-dictionary.zh | Oxford-dictionary.zh.arpa

```
1
2  \data\
3  ngram 1=  37992
4  ngram 2=  361419
5  ngram 3=  209786
6
7
8  \1-grams:
9  -5.84994 <s> -1.00137
10 -3.55218 英语 -0.628051
11 -5.03702 字母表 -0.694482
12 -2.51118 中 -0.692795
13 -1.28505 的 -0.839764
14 -3.5863 第一 -0.70014
15 -2.46891 个 -0.550793
16 -4.17784 字母 -0.579907
17 -1.93236 是 -0.741366
18 -5.00484 A -0.214711
19 -1.1442 。 -2.67583
20 -1.07178 </s>
89023 -4.32159 我们 深入
89024 -4.31424 我们 雇
89025 -3.35469 我们 介绍 -0.134097
89026 -3.40422 我们 因为 -0.567763
89027 -4.12123 我们 具有
89028 -4.2685 我们 计算
89029 -4.28012 我们 有所
89030 -2.39817 我们 听到 -0.132928
89031 -4.25878 我们 首先
89032 -4.05647 我们 无论如何 -0.175264
89033 -4.04555 我们 谨慎 -0.175264
89034 -4.01343 我们 正确 -0.175264
89035 -3.65107 我们 老板
89036 -3.90187 我们 当
89037 -4.2301 我们 顿
89038 -3.17385 我们 听说 -0.31132
89039 -2.71077 我们 很快 -0.754989
89040 -4.19345 我们 不久
89041 -4.05547 我们 恳求
89042 -4.20533 我们 以后
```

```
431166  -1.02001 ⟶是 身体 的LF
431167  -1.06692 ⟶是 身体 上LF
431168  -0.567157 ⟶是 身体 健康LF
431169  -1.27857 ⟶是 身体 关节LF
431170  -0.77255 ⟶是 经过 适当LF
431171  -1.11653 ⟶是 关于 一个LF
431172  -1.12168 ⟶是 关于 教会LF
431173  -0.388953 ⟶是 是否 有LF
431174  -0.204048 ⟶是 一切 。LF
431175  -0.578119 ⟶是 500 美元LF
431176  -0.0641712 ⟶是 荒谬 的LF
431177  -1.09409 ⟶是 谁 的LF
431178  -0.863096 ⟶是 谁 。LF
431179  -0.51727 ⟶是 谁 ？LF
431180  -1.5328 ⟶是 谁 把LF
```

To make the faster loading, use the command to binary the zh.arpa file:

```
wangyiming@lobo:~$
wangyiming@lobo:~$
wangyiming@lobo:~$ /smt/mosesdecoder/bin/build_binary Aaron/Moses/lm/CN/Oxford-dictionary.zh.ar
pa Aaron/Moses/lm/CN/Oxford-dictionary.zh.blm
Connected to 161.64.89.186   SSH2 - aes128-cbc - hmac-md5   95x29   NUM
```

```
wangyiming@lobo:~$
wangyiming@lobo:~$ /smt/mosesdecoder/bin/build_binary Aaron/Moses/lm/CN/Oxford-dictionary.zh.ar
pa Aaron/Moses/lm/CN/Oxford-dictionary.zh.blm
Reading Aaron/Moses/lm/CN/Oxford-dictionary.zh.arpa
----5---10---15---20---25---30---35---40---45---50---55---60---65---70---75---80---85---90---95
--100
**************************************************************************************************
*****
SUCCESS
wangyiming@lobo:~$
Connected to 161.64.89.186   SSH2 - aes128-cbc - hmac-md5   95x29   NUM
```

Add | /home/wangyiming/Aaron/Moses/lm/CN

Remote Name	Size	Type	Modified	Attributes
Oxford-dictionary.zh.arpa	15,364,649	ARPA 文...	12/12/2013 08:2...	-rw-rw-r--
Oxford-dictionary.zh.blm	13,745,321	BLM 文件	12/12/2013 08:3...	-rw-r--r--
Oxford-dictionary.zh.lm.gz	4,701,426	WinRAR...	12/12/2013 08:1...	-rw-rw-r--
Oxford-dictionary.zh.sb	7,747,108	SB 文件	12/12/2013 08:0...	-rw-rw-r--
Oxford-dictionary.zh.tok	6,667,108	TOK 文件	03/19/2013 01:3...	-rw-rw-r--

If use the following command, it will show the following result:

```
wangyiming@lobo:~$
wangyiming@lobo:~$
wangyiming@lobo:~$ $ echo "我来学校了i come to university" | /smt/mosesdecoder/bin/query Aaron/
ses/lm/CN/Oxford-dictionary.zh.blm
Loading statistics:
Name:query      VmPeak:30424 kB VmRSS:14460 kB  RSSMax:14460 kB user:0.004      sys:0   CPU:0.0
04      real:0
$: command not found
After queries:
Name:query      VmPeak:30428 kB VmRSS:14460 kB  RSSMax:14460 kB user:0.004      sys:0   CPU:0.0
04      real:0
Total time including destruction:
Name:query      VmPeak:30428 kB VmRSS:1532 kB   RSSMax:14460 kB user:0.004      sys:0   CPU:0.0
04      real:0
wangyiming@lobo:~$
```

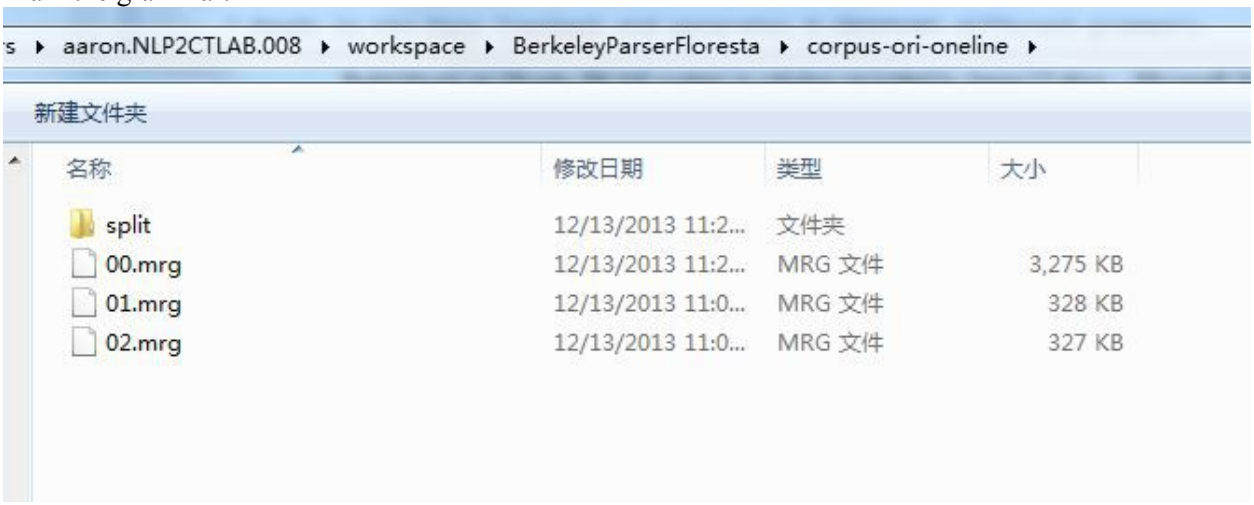
Corpus for Training translation model:

Use the corpus 200,000 PT-CN bilingual sentences extracted from online websites of Macau SAR.

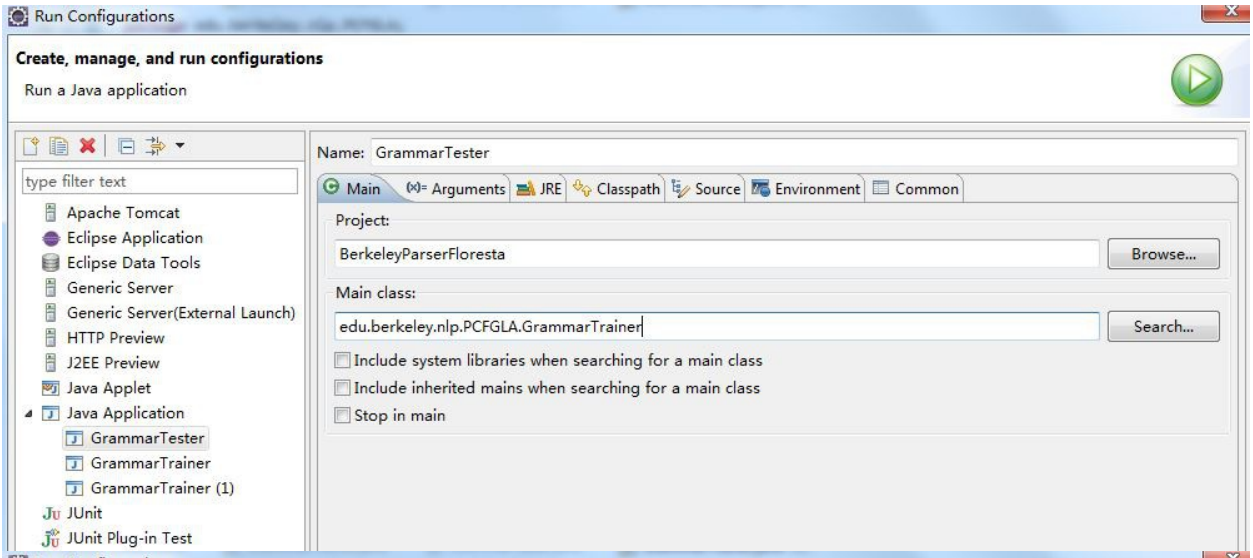
Prepare PT corpus:

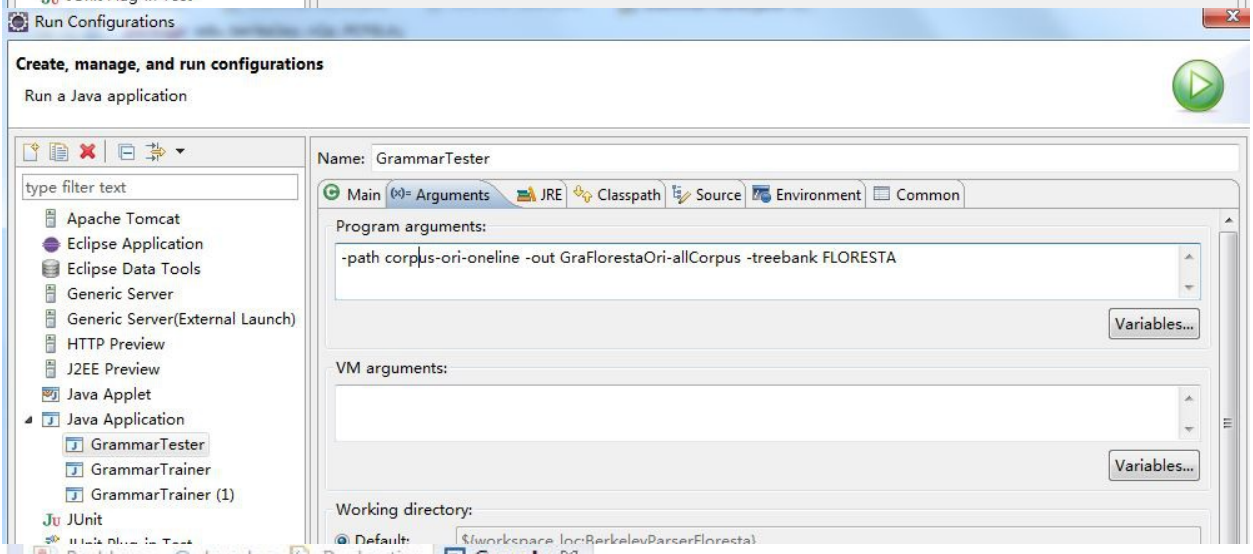
Parse the PT.200000.tok sentence using the trained grammar on all the Floresta-Bosque corpus (train-deve-test):

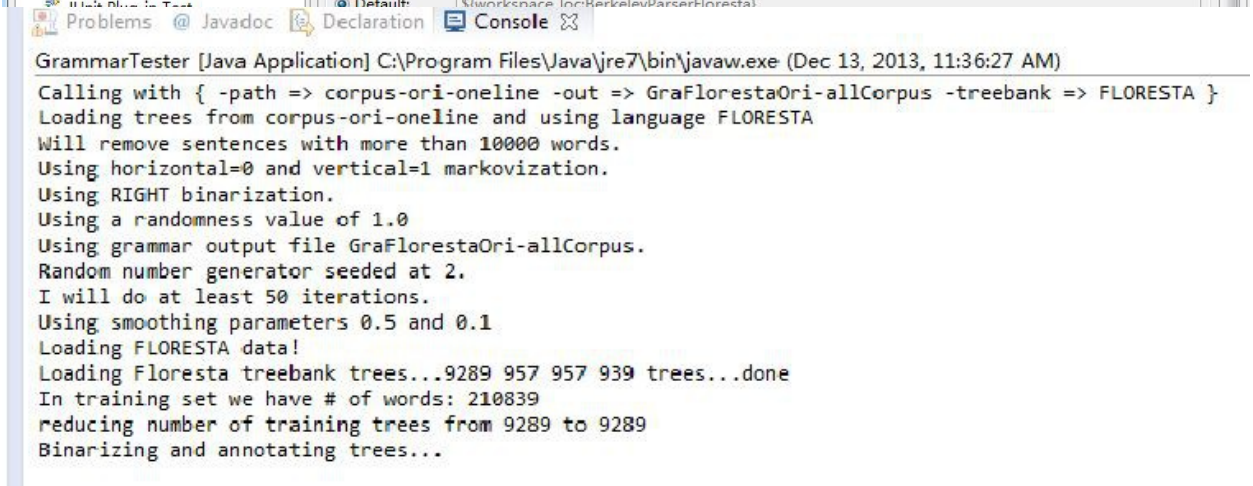
Train the grammar:



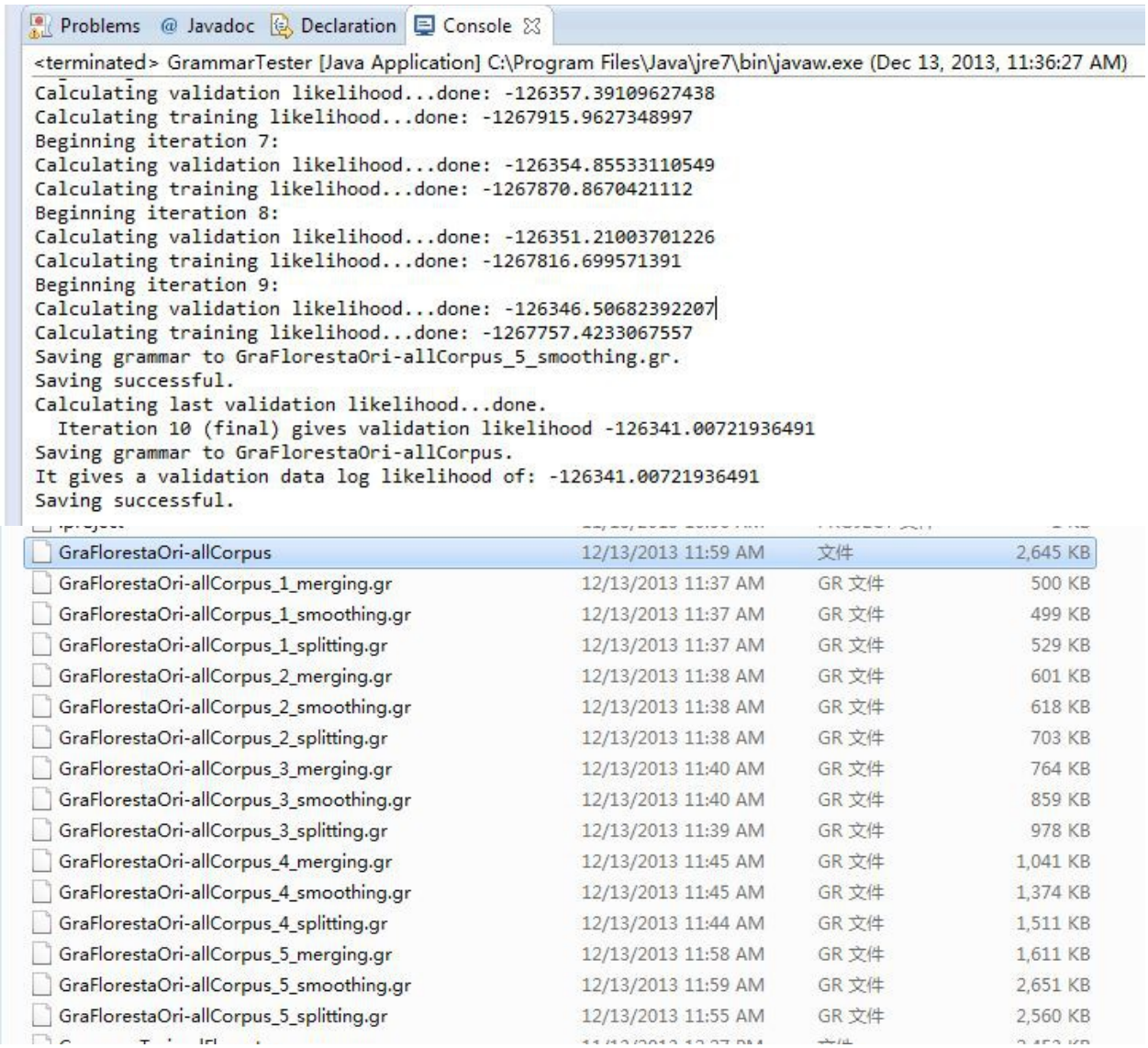
This time, the 00.mrg contains “train+deve+test” sentences, 01.mrg contains deve sentences, 02.mrg contains testing sentences.


Run Configurations
Create, manage, and run configurations
Run a Java application
type filter text
Apache Tomcat
Eclipse Application
Eclipse Data Tools
Generic Server
Generic Server(External Launch)
HTTP Preview
J2EE Preview
Java Applet
Java Application
GrammarTester
GrammarTrainer
GrammarTrainer (1)
JUnit
JUnit Plug-in Test
Name: GrammarTester
Main
Arguments
JRE
Classpath
Source
Environment
Common
Project:
BerkeleyParserFloresta
Browse...
Main class:
edu.berkeley.nlp.PCFGLA.GrammarTrainer
Search...
Include system libraries when searching for a main class
Include inherited mains when searching for a main class
Stop in main

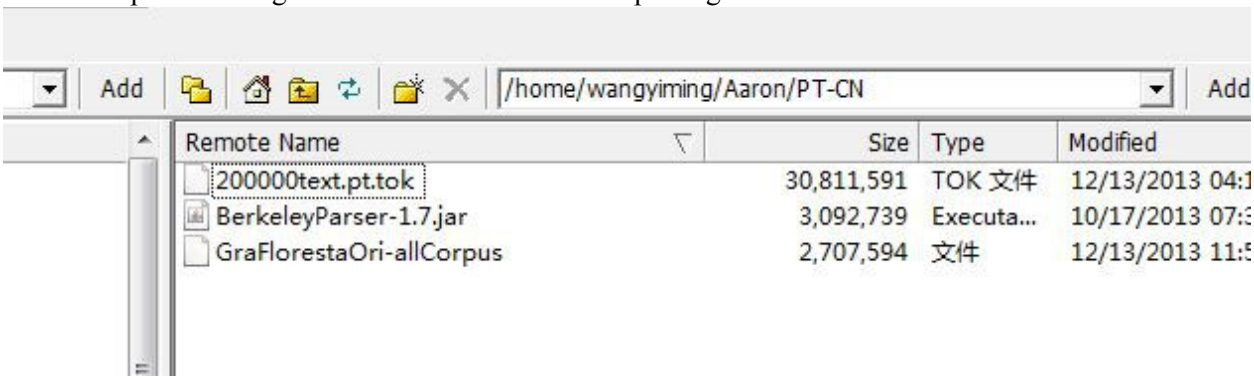

Run Configurations
Create, manage, and run configurations
Run a Java application
type filter text
Apache Tomcat
Eclipse Application
Eclipse Data Tools
Generic Server
Generic Server(External Launch)
HTTP Preview
J2EE Preview
Java Applet
Java Application
GrammarTester
GrammarTrainer
GrammarTrainer (1)
JUnit
Name: GrammarTester
Main
Arguments
JRE
Classpath
Source
Environment
Common
Program arguments:
-path corpus-ori-oneline -out GraFlorestaOri-allCorpus -treebank FLORESTA
Variables...
VM arguments:
Variables...
Working directory:


Problems
Javadoc
Declaration
Console
GrammarTester [Java Application] C:\Program Files\Java\jre7\bin\javaw.exe (Dec 13, 2013, 11:36:27 AM)
Calling with { -path => corpus-ori-oneline -out => GraFlorestaOri-allCorpus -treebank => FLORESTA }
Loading trees from corpus-ori-oneline and using language FLORESTA
Will remove sentences with more than 10000 words.
Using horizontal=0 and vertical=1 markovization.
Using RIGHT binarization.
Using a randomness value of 1.0
Using grammar output file GraFlorestaOri-allCorpus.
Random number generator seeded at 2.
I will do at least 50 iterations.
Using smoothing parameters 0.5 and 0.1
Loading FLORESTA data!
Loading Floresta treebank trees...9289 957 957 939 trees...done
In training set we have # of words: 210839
reducing number of training trees from 9289 to 9289
Binarizing and annotating trees...

Finished as:



Put the corpus and the grammar in the 186-server for parsing:

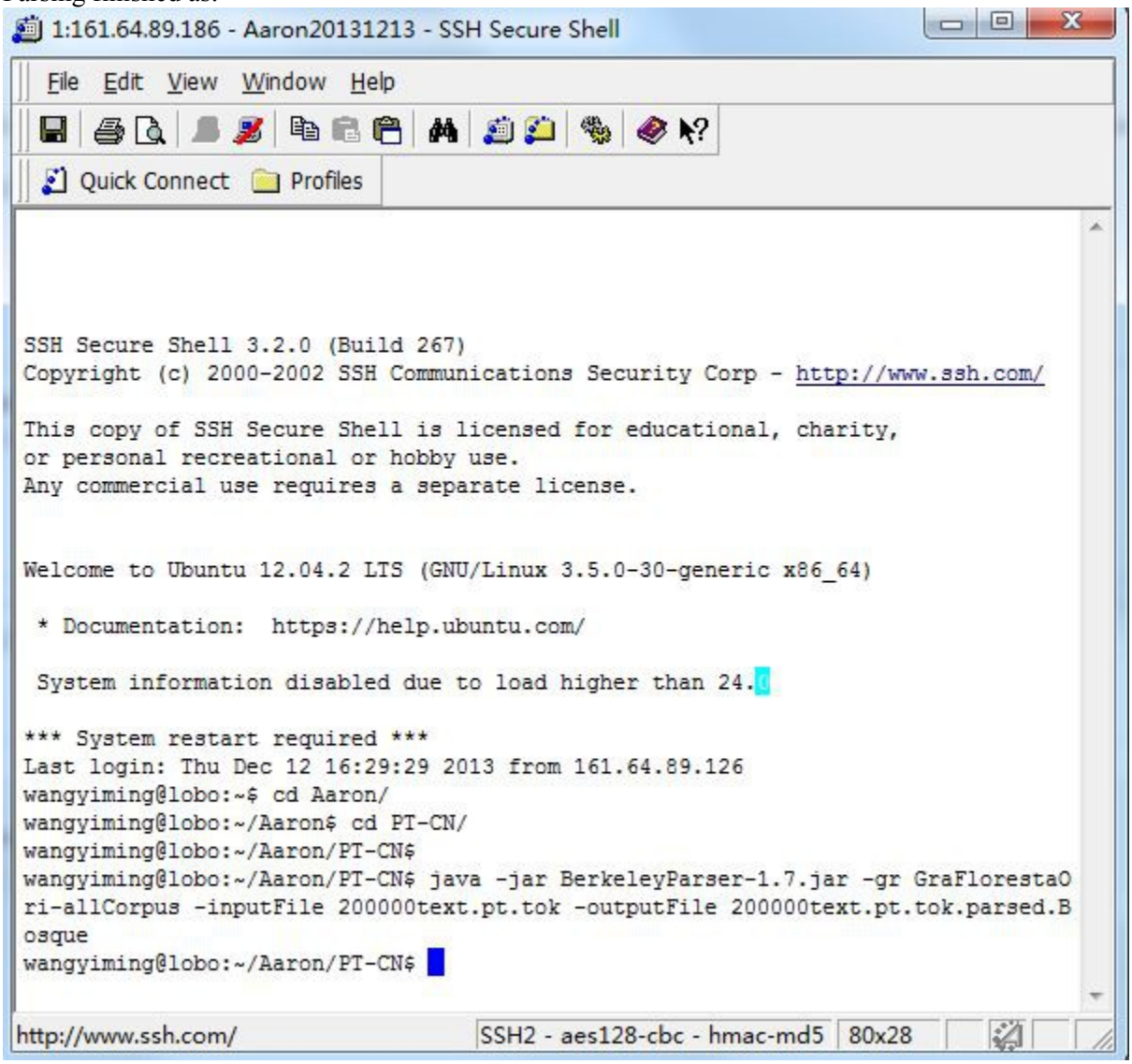


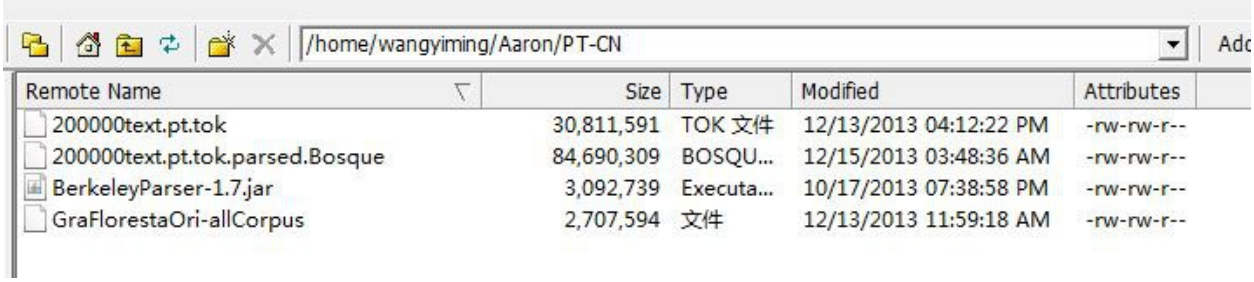
The parsing command, the parsing on server-186 began at 20131213-4:17 PM, finished at 20131215-03:48:36AM

```
wangyiming@lobo:~/Aaron/PT-CN$
wangyiming@lobo:~/Aaron/PT-CN$ java -jar BerkeleyParser-1.7.jar -gr GraFlorestaO
ri-allCorpus -inputFile 200000text.pt.tok -outputFile 200000text.pt.tok.parsed.B
osque
```

Connected to 161.64.89.186 | SSH2 - aes128-cbc - hmac-md5 | 80x24

Parsing finished as:

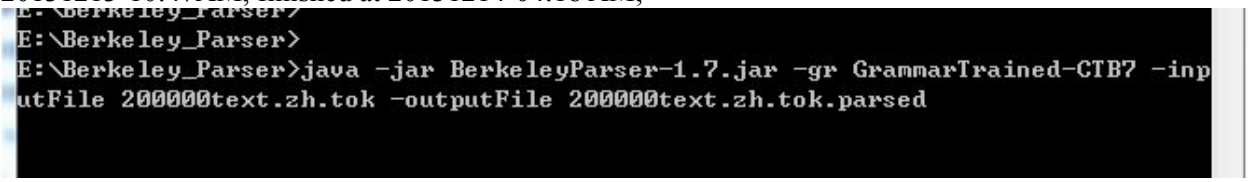




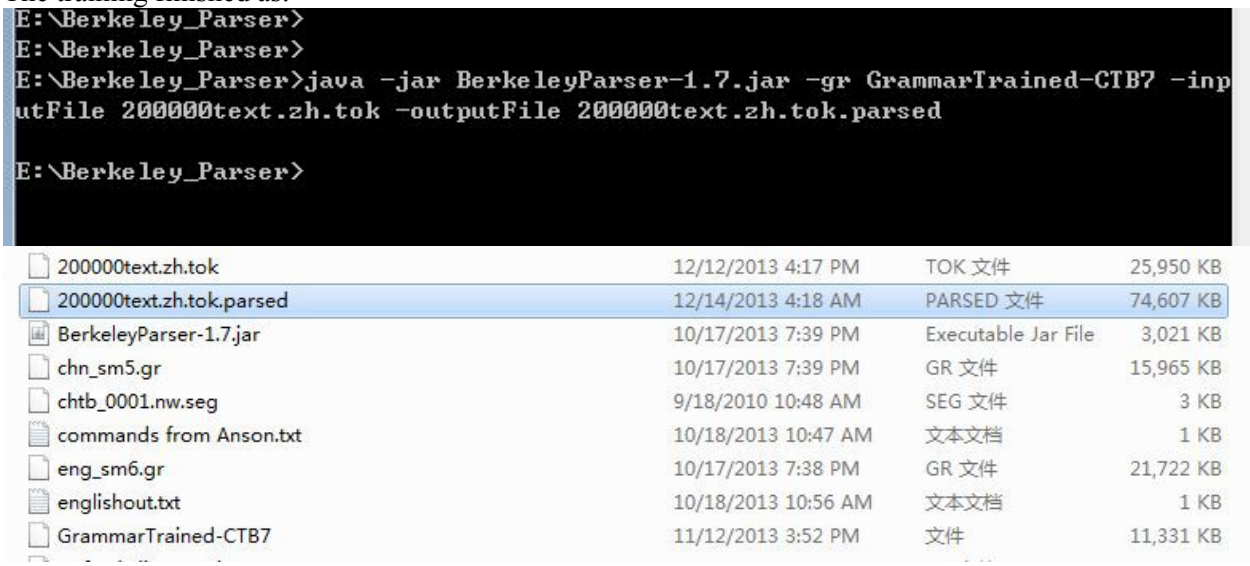
For the uni-nofuntag-Floresta-Bosque seting: We should replace the ori-phrase tags parsed from last step with the uni-phrase. (If we use the uni-phrase-grammar-trained to parse the plain.tok PT text to gain the uni-phrase taggest corpus, it is not suitable because the grammar-trained-pt-floresta and the grammar-trained-pt-floresta-uniphrase have different accuracies.)

Prepare the CN corpus:

We parse the 200,000 chinese.tok sentences using the GrammarTrainedCTB-7; parsing began at 20131213-10:47AM, finished at 20131214-04:18 AM;



The training finished as:



After the parsing finish of the CN corpus, we replace the ori-phrase-tags using the uni-nofuntag-phrase-tags

Training translation model use ori-tags:

Convert the 200,000 PT and CN parsed sentences into mosesxml format.

Test translation model sue ori-tags:

Training translation model use uni-tags:

Convert the 200,000 PT and CN parsed sentences into mosesxml format.

Test translation model sue uni-tags:

Tune the MT system using LEPOR metric (external metric):

1. Change the xxxx file as below to call the LEPOR metric.

2. Change the LEPOR source document as below:

3. Put the source document of LEPOR metric at yyy

4. Run the system like below to test

5.

==========

Reference:

Moses manual: [http://www.statmt.org/moses/manual/manual.pdf], accessed 2013.10.31

TianLiang' blog: [http://www.tianliang123.com/moses_installation], accessed 2013.10.31.