Mosesを使ったフレーズ機械翻訳の演習
- 2日目 -

林 克彦

日本電信電話株式会社 NTT コミュニケーション科学基礎研究所

平成 27 年 2 月 27 日

1　環境設定

1.1　初日の演習資料の項目 0.1 と 0.2 に対応

初日の演習と同じツール、データを利用して演習を進めます。任意のディレクトリ (ここではホームディレクトリ) に

- `~/alagin2015_binary_x86_64-Cygwin`
- `~/kftt-data-1.0`

があるか確かめて下さい。“Cygwin” は “Linux” や “Darwin13” などで読み替えてください。
次に以下の環境変数を設定して下さい。これも初日の演習と同じです。

- `export ALAGIN_HOME=~/alagin2015_binary_x86_64-Cygwin`
- `export SCRIPTS_ROOTDIR=${ALAGIN_HOME}/scripts`
- `export KFTT_DATA=~/kftt-data-1.0`
- `export LD_LIBRARY_PATH=${ALAGIN_HOME}/gcc/lib64:`
  `${ALAGIN_HOME}/gcc/lib:${ALAGIN_HOME}/lib`

1.2　初日の演習資料の項目 1、2、3、4、5 に対応

初日の演習で構築したデータやモデルはありますか？

- `${EXP}/exp`

の下に `data`、`model`、`tuning`、`test` が構築されていますか?`${EXP}`は初日の資料では`${PWD}`と記載して、演習を行っていた場所です。特に、

- `${EXP}/exp/tuning/en-ja/moses.ini`
- `${EXP}/exp/model/en-ja/model/phrase-table.gz`

などのファイルはありますか？もし無い場合は、初日の演習資料 1、2、3、4、5 を振り返り、必要な場所からもう一度やり直して下さい。

2　パラメータと翻訳精度・速度の関係を調べる

2.1　念のため実験環境をコピーしておく

これから${EXP}/expの中にあるファイルを色々と書き換えたりするため、念のためコピーをとりましょう。

- cp -r ${EXP}/exp ${EXP}/exp-copy

cp -r a bというコマンドはディレクトリaをディレクトリbにコピーします。もし以後の演習でexpの中身がおかしくなったりしたら、

- cp -r ${EXP}/exp-copy ${EXP}/exp

で復元しましょう。

2.2　テストと評価の仕方をおさらい (初日の演習資料 [38]、[40])

まず、初日の演習資料項目 5 の [38] と同じことをしましょう。

- ${ALAGIN_HOME}/bin/moses -f exp/tuning/en-ja/moses.ini <

  exp/test/kyoto-test.en > exp/test/en-ja/kyoto-test.ja 2>

  exp/test/en-ja/kyoto-test.err

翻訳を評価しましょう。初日の演習資料項目 5 の [40] と同じです。

- ${ALAGIN_HOME}/bin/mt-evaluator -eval "bleu ribes" -ref

  exp/test/kyoto-test.ja exp/test/en-ja/kyoto-test.ja

2.3　スタックサイズと翻訳精度・速度の関係を調べる

スタックサイズはデコーダの探索エラーを減らすためのオプションです。スタックサイズは-sオプションを下のような形で付けることにより変更できます。

- time ${ALAGIN_HOME}/bin/moses -f exp/tuning/en-ja/moses.ini -s 10 <

  exp/test/kyoto-test.en > exp/test/en-ja/kyoto-test-s10.ja 2>

  exp/test/en-ja/kyoto-test.err

ここではスタックサイズを 10 に設定しています。翻訳結果はexp/test/en-ja/kyoto-test-s10.jaに出力されます。timeはプロセスの実行時間を計測するコマンドです。user タイムを参考にして下さい。動かない場合は、time を消し、ご自身で時間を計測して下さい。

サイズを 1、10、100 と変化させたときの翻訳精度と速度をはかり、下の表のようになるか確認して下さい。環境が異なるので、厳密に同じ結果となる必要はありません。出力ファイルはexp/test/en-ja/kyoto-test-s10.jaのように-s10などにより名前を変えた方がどの翻訳結果か分かりやすくなります。

表 1: スタックサイズと翻訳精度・速度の関係

stack size	BLEU	RIBES	翻訳速度
1	0.09	0.56	36 秒
10	0.10	0.57	37 秒
100	0.10	0.58	1 文 7 秒

2.4 moses.ini の中身を見てみる

exp/tuning/en-ja/moses.ini の中身を見てみましょう。

- less exp/tuning/en-ja/moses.ini

```
# MERT optimized configuration
# decoder /Users/katsuhiko-h/alagin2015_binary.x86_64-Darwin12/bin/moses
# BLEU 0.140057 on dev /Users/katsuhiko-h/alagin2015_binary.x86_64-Darwin12/exp/tuning/kyoto-tune.en
# We were before running iteration 9
# finished 2015 年 2 月 25 日 水曜日 14 時 45 分 45 秒 JST
### MOSES CONFIG FILE ###
#########################

# input factors
[input-factors]
0

# mapping steps
[mapping]
0 T 0

[distortion-limit]
6

# feature functions
[feature]
UnknownWordPenalty
...
```

2.5 Distortion-limit と翻訳精度・速度の関係を調べる

exp/tuning/en-ja/moses.ini の中にある [distortion-limit] という場所の下の数値を変えてみましょう。コマンドで上書きするには

- sed -i -e "/\[distortion-limit\]/{n;s/[0-9]*/12/;}"

  exp/tuning/en-ja/moses.ini

で行えます。上の例では distortion-limit を 12 に設定しています。12 を 0 にすれば、0 に設定することができます。vi、emacs、ワード、メモ帳などのエディタで書き換えても問題ありません。書き換え方がわからない場合、講師に質問して下さい。ちなみに moses にはスタックサイズのオプションと同様に、distortion-limit を設定する -dl オプションがあるので、それを使うこともできます。

例えば、0、6、12、24 と変化させたときに、翻訳精度と速度がどうなるか確認してください。

- time ${ALAGIN_HOME}/bin/moses -f exp/tuning/en-ja/moses.ini <

```
exp/test/kyoto-test.en > exp/test/en-ja/kyoto-test-d12.ja 2>

exp/test/en-ja/kyoto-test.err
```

time コマンドは環境に合わせて、適宜外して下さい。翻訳結果は exp/test/en-ja/kyoto-test-d12.ja に出力されます。d12 など適宜名前を変えて、どの翻訳結果かをわかりやすくした方が良いでしょう。

下の表のように、distortion-limit は語順の並べ替えに効くパラメータであり、英語と日本語の語順の違いを考えると、ある程度大きく設定する必要があると思います。作業が終わったら、moses.ini の distortion limit を 6 に戻しておいて下さい。

表 2: distortion-limit と翻訳精度・速度の関係

-dl	BLEU	RIBES	翻訳速度
0	0.09	0.56	37 秒
6	0.10	0.58	1 分 38 秒
12	0.11	0.57	2 分 23 秒
24	0.10	0.57	3 分 25 秒

2.6 モデルの重みと翻訳精度の関係を調べる

対数線形モデルの各モデルに対する重みパラメータを変えることで、翻訳精度への影響を調べます。演習では言語モデルに対する重みパラメータを変更することで、翻訳精度への影響を調べてみます。

exp/tunining/en-ja/moses.ini の中に、

```
# dense weights for feature functions
[weight]

LexicalReordering0= 0.0681218 -0.0452658 0.0912827 0.0829503 0.122046 0.00805037
Distortion0= 0.0244465
LM0= 0.0687026
WordPenalty0= -0.296319
PhrasePenalty0= 0.040592
TranslationModel0= 0.00688065 0.027698 0.072288 0.0453568
```

となっている箇所があると思います (数値は異なるかも知れません)。この中で、LM0 に対する数値を 0.0 に書き換えて翻訳してみましょう。

- sed -i -e "s/LM0= [\.0-9]*/LM0= 0.0"/

  exp/tuning/en-ja/moses.ini

エディタで書き換えることもできます。その場合、# LM0 =0.0687026 のようにしてコメントアウトし、LM0 =0.0 と追加してもらっても構いません。

翻訳は次のようにし、

- `${ALAGIN_HOME}/bin/moses -f exp/tuning/en-ja/moses.ini < exp/test/kyoto-test.en > exp/test/en-ja/kyoto-test-lm0.ja 2> exp/test/en-ja/kyoto-test.err`

exp/test/en-ja/kyoto-test-lm0.ja に翻訳結果が出力されます。これを

- `head -n 3 exp/test/en-ja/kyoto-test-lm0.ja`
- `head -n 3 exp/test/en-ja/kyoto-test.ja`

などで他の翻訳結果と見比べてみて下さい。

3　エラー分析と翻訳辞書へのフレーズの追加

ここからはエラー分析をし、フレーズテーブルに変更を加えることで、精度向上をはかります。ここまでの演習で moses.ini を書き換えてきたので、

- `cp -r exp-copy exp`

を行い、元の状態に戻して下さい。これは必須ではありませんが、言語モデルの重みを適切なものに戻した方が、以後の作業による結果を分析しやすいです。

まず、今まで通り、テストして翻訳文を作り、評価を行って下さい。

- `${ALAGIN_HOME}/bin/moses -f exp/tuning/en-ja/moses.ini < exp/test/kyoto-test.en > exp/test/en-ja/kyoto-test.ja 2> exp/test/en-ja/kyoto-test.err`
- `${ALAGIN_HOME}/bin/mt-evaluator -eval "bleu ribes" -ref exp/test/kyoto-test.ja exp/test/en-ja/kyoto-test.ja`

この環境では、

BLEU = 0.105333
RIBES = 0.585175

となっています。

次に、

- `less exp/test/en-ja/kyoto-test.ja`

を見て下さい。この環境では次のような翻訳結果ができています。

infobox 仏教
道元 は 、 禅宗 の 僧侶 は 鎌倉 時代 初期 の 僧 で あ る 。
曹洞 禅 の 祖 と さ れ る 。
その 後 も に 渡 っ た 生涯 に よ っ て kigen は さ れ て い る 。
に は 宗 と 呼 ば れ 名誉 高僧 の 称号 で あ る 。
と 仏 所 を 追贈 さ れ 、 伝灯 国師 joyo-daishi と し て い る 。
一般 に は 道元 禅師 と 呼 ば れ て い る 。
三 と し て い る の は 、 tooth の brushing い て は 、 日本 に お い て 食事 作法 の cleaning と し て い る 。
...

ここではkigenという単語は未知語として出力されています。これは日本語の単語では希玄という単語になります。

このような未知語を翻訳するには、フレーズテーブルにその情報を登録する必要があります。ここではそれを手で追加することにより、翻訳できるようにしてみましょう。

- `gunzip exp/model/en-ja/model/phrase-table.gz`
- `echo "kigen ||| 希玄 ||| 1 1 1 1 ||| 0-0 ||| 1 1 1 |||" >> exp/model/en-ja/model/phrase-table`
- `gzip exp/model/en-ja/model/phrase-table`

再度、翻訳と評価をしてみましょう。

- `${ALAGIN_HOME}/bin/moses -f exp/tuning/en-ja/moses.ini < exp/test/kyoto-test.en > exp/test/en-ja/kyoto-test-kigen.ja 2> exp/test/en-ja/kyoto-test.err`
- `${ALAGIN_HOME}/bin/mt-evaluator -eval "bleu ribes" -ref exp/test/kyoto-test.ja exp/test/en-ja/kyoto-test-kigen.ja`

この環境では、

BLEU = 0.105375
RIBES = 0.585266

となり、わずかに改善が見られました。実際に、翻訳結果を見てみると、

- `less exp/test/en-ja/kyoto-test-kigen.ja`

infobox 仏教
道元は、禅宗の僧侶は鎌倉時代初期の僧である。
曹洞禅の祖とされる。
その後もに渡った生涯によって希玄はされている。
には宗と呼ばれ名誉高僧の称号である。
と仏所を追贈され、伝灯国師joyo-daishiとしている。
一般には道元禅師と呼ばれている。
三としているのは、toothのbrushingいては、日本において食事作法のcleaningとしている。
...

となっており、kigenが希玄と訳されているのがわかります。

翻訳辞書のエントリは

入力フレーズ `f` ||| 出力フレーズ `e` ||| フレーズ翻訳確率 `P(e|f)` 単語翻訳確率 P_{lex}`(e|f)` フレーズ翻訳確率 `P(f|e)` 単語翻訳確率 P_{lex}`(f|e)` ||| 単語対応 ||| 出力フレーズの出現頻度 `c(e)` 出力フレーズの出現頻度 `c(f)` 両フレーズの同時出現頻度 `c(f,e)` |||

となっています。ご自身でエラー分析し、効果的と思われるフレーズを追加し、翻訳精度を向上させてみましょう。