University of Macau

CISC7021 – **Applied Natural Language Processing**

Assignment 2, 2022/2023

(Due date: *6 Nov, 2022*)

Introduction

In this assignment, we will set up an environment for installing and running MOSES [Koehn et al., 2007], a widely used statistical machine translation system. You will learn how to build up a *Chinese-English* (or other language pairs[1]) phrase-based translation system based on MOSES, create and train a translation model, and use it together with a language model in handling the translation task. A detailed description of the MOSES can be found at the official website: http://www.statmt.org/moses/.

Environment Setup

You may consider reusing the previous virtual machine environment for this assignment. You need several packages before compiling MOSES: *Boost*, *g++*, *git*, *subversion*, *automake*, *libtool*, *zlib1g-dev*, *libboost-all-dev*, *libbz2-dev*, *liblzma-dev*, *python-dev*. Moreover, in order to run MOSES, you are required to have a language model toolkit (for example, SRILM[2] [Stolcke, 2002]) and a word-alignment package for aligning the parallel corpus (for example, GIZA++ [Och and Ney, 2003]). By default, MOSES uses KenLM [Heafield et al., 2013] language model toolkit. From the Internet, it is easy to find out detailed information about how to install and run the packages and toolkits. Some good tutorials about the installations can be found at:

- http://www.statmt.org/moses/?n=Development.GetStarted
- http://www.statmt.org/moses/?n=Moses.Baseline

Data

The training, development, and test data for this assignment come from the International Workshop on Spoken Language Translation (IWSLT) 2014 evaluation campaign.[3] Those Chinese-English data have been made available at the UMMoodle.[4] (*Note*: *Chinese text must be tokenized prior to the construction processes.*) If you would like to create a translation system for other language pairs, you may download the related data from the official website of the IWSLT conference.

---

[1] https://wit3.fbk.eu/mt.php?release=2014-01

[2] http://www.speech.sri.com/projects/srilm/download.html

[3] https://wit3.fbk.eu/

[4] https://ummoodle.um.edu.mo/mod/assign/view.php?id=2661729

Tasks

1. Pre-process the parallel corpus and test data for MOSES.
   (i) Tokenize the parallel corpora to insert spaces between the words and punctuations. For the Chinese data, since it does not show the explicit word boundary indicators, you need to tokenize them by using the ANJS toolkit[5] [ANJS, 2016]. For other language pairs data, you may use the *tokenize.perl* script of MOSES.
   (ii) Reduce the size of the parallel corpus by removing the parallel sentences which have more than 50 words.
2. Build a 3-*gram* language model by using the SRILM/KenLM toolkit for the parallel corpora. You may consider using the monolingual training data provided or other monolingual data. In your report, you need to state the sources you used for creating the language model.
3. Build a Chinese-English translation model for a Phrase-based translation system using the MOSES. Notes:
   - Use the created 3-*gram* language model in training the translation model.
   - You may use the default options of MOSES to train the models: alignment heuristic parameter for word alignment is "*grow-diag-final*" [Och and Ney, 2003]; the reordering model is "*msd-bidirectional-fe*" [Koehn et al., 2005].
4. *This is an optional process*. The parameters of the built translation model can be further tuned to give a better translation quality. You can use the provided development data for tuning the model parameters. The development data is disjoint from both the training and test data.
5. Translate the test data into English using your trained models with MOSES.
6. Evaluate the quality of the translated sentences using an online automatic evaluation system[6] or the `multi-bleu.perl` script provided by MOSES.

Report

You need to submit a report of the work you have done (2~4 pages) which includes the following contents: (1) difficulties encountered during installation, execution, or anything related to this assignment; (2) list of core stages performed during the model training; (3) tables (or graphs) of the training data before and after pre-processing, size of language and translation model; (4) report the translation quality of the model in terms of BLUE, METERO, TER, WER, etc.; and (5) analysis of selected translation results in terms of their quality (grammatical errors, translation errors, fluency, etc.)

---

[5] https://ummoodle.um.edu.mo/pluginfile.php/3993491/mod_assign/introattachment/0/ansjTokenizer.zip?forcedownload=1

[6] http://asiya.cs.upc.edu/demo/asiya_online.php#

The report should follow the format of ACL proceeding.[7]

References

1. ANJS. 2016. ANJS Toolkit: http://github.com/NLPchina/ansj_seg.
2. Kenneth Heafield, Ivan Pouzyrevsky, Jonathan H Clark, and Philipp Koehn. 2013. Scalable Modified Kneser-Ney Language Model Estimation. In *ACL (2)*, pages 690–696.
3. Philipp Koehn, Amittai Axelrod, Alexandra Birch, Chris Callison-Burch, Miles Osborne, David Talbot, and Michael White. 2005. Edinburgh system description for the 2005 IWSLT speech translation evaluation. In *IWSLT*, pages 68–75.
4. Philipp Koehn, Hieu Hoang, Alexandra Birch, Chris Callison-Burch, Marcello Federico, Nicola Bertoldi, Brooke Cowan, Wade Shen, Christine Moran, Richard Zens, and others. 2007. Moses: Open source toolkit for statistical machine translation. In *Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions*, pages 177–180.
5. Franz Josef Och and Hermann Ney. 2003. A Systematic Comparison of Various Statistical Alignment Models. *Computational Linguistics*, 29(1):19–51.
6. Andreas Stolcke. 2002. SRILM - an extensible language modeling toolkit. In *7th International Conference on Spoken Language Processing, ICSLP2002 - INTERSPEECH 2002, Denver, Colorado, USA, September 16-20, 2002*.

[7] http://acl2015.org/files/acl2015.tex or http://acl2015.org/files/acl2015.dot