Phramer - An Open Source Statistical Phrase-Based Translator

Marian Olteanu, Chris Davis, Ionut Volosen and Dan Moldovan
Human Language Technology Research Institute
The University of Texas at Dallas
Richardson, TX 75080
{marian,phoo,volosen,moldovan}@hlt.utdallas.edu

Abstract

This paper describes the open-source Phrase-Based Statistical Machine Translation Decoder - Phramer. The paper also presents the UTD (HLTRI) system build for the WMT06 shared task. Our goal was to improve the translation quality by enhancing the translation table and by preprocessing the source language text

1 Introduction

Despite the fact that the research in Statistical Machine Translation (SMT) is very active, there isn't an abundance of open-source tools available to the community. In this paper, we present Phramer, an open-source system that embeds a phrase-based decoder, a minimum error rate training (Och, 2003) module and various tools related to Machine Translation (MT). The software is released under BSD license and it is available at http://www.phramer.org/.

We also describe our Phramer-based system that we build for the WMT06 shared task.

2 Phramer

Phramer is a phrase-based SMT system written in Java. It includes:

- A decoder that is compatible with Pharaoh (Koehn, 2004),
- A minimum error rate training (MERT) module, compatible with Phramer's decoder, with Pharaoh and easily adaptable to other SMT or non-SMT tasks and
- various tools.

The decoder is fully compatible with Pharaoh 1.2 in the algorithms that are implemented, input files (configuration file, translation table, language models) and command line. Some of the advantages of Phramer over Pharaoh are: (1) source code availability and its permissive license; (2) it is very fast (1.5–3 times faster for most of the configurations); (3) it can work with various storage layers for the translation table (TT) and the language models (LMs): memory, remote (access through TCP/IP), disk (using SQLite databases[1]). Extensions for other storage layers can be very easily implemented; (4) it is more configurable; (5) it accepts compressed data files (TTs and LMs); (6) it is very easy to extend; an example is provided in the package – part-of-speech decoding on either source language, target language or both; support for POS-based language models; (7) it can internally generate n-best lists. Thus no external tools are required.

The MERT module is a highly modular, efficient and customizable implementation of the algorithm described in (Och, 2003). The release has implementations for BLEU (Papineni et al., 2002), WER and PER error criteria and it has decoding interfaces for Phramer and Pharaoh. It can be used to search parameters over more than one million variables. It offers features as resume search, reuse hypotheses from previous runs and various strategies to search for optimal λ weight vectors.

[1] http://www.sqlite.org/

Proceedings of the Workshop on Statistical Machine Translation, pages 146–149,
New York City, June 2006. ©2006 Association for Computational Linguistics

The package contains a set of tools that include:

- Distributed decoding (compatible with both `Phramer` and `Pharaoh`) – it automatically splits decoding jobs and distributes them to workers and assembles the results. It is compatible with lattice generation, therefore it can also be used during weights search (using MERT).
- Tools to process translation tables – filter the TT based on the input file, flip TT to reuse it for English-to-Foreign translation, filter the TT by phrase length, convert the TT to a database.

3 WMT06 Shared Task

We have assembled a system for participation in the WMT 2006 shared task based on `Phramer` and other tools. We participated in 5 subtasks: DE→EN, FR→EN, ES→EN, EN→FR and EN→ES.

3.1 Baseline system

3.1.1 Translation table generation

To generate a translation table for each pair of languages starting from a sentence-aligned parallel corpus, we used a modified version of the `Pharaoh` training software [2]. The software also required GIZA++ word alignment tool(Och and Ney, 2003).

We generated for each phrase pair in the translation table 5 features: phrase translation probability (both directions), lexical weighting (Koehn et al., 2003) (both directions) and phrase penalty (constant value).

3.1.2 Decoder

The `Phramer` decoder was used to translate the *devtest2006* and *test2006* files. We accelerated the decoding process by using the *distributed decoding* tool.

3.1.3 Minimum Error Rate Training

We determined the weights to combine the models using the MERT component in `Phramer`. Because of the time constrains for the shared task submission[3], we used `Pharaoh` + `Carmel`[4] as the decoder for the MERT algorithm.

3.1.4 Preprocessing

We removed from the source text the words that don't appear either in the source side of the training corpus (thus we know that the translation table will not be able to translate them) or in the language model for the target language (and we estimate that there is a low chance that the untranslated word might actually be part of the reference translation). The purpose of this procedure is to minimize the risk of inserting words into the automatic translation that are not in the reference translation.

We applied this preprocessing step only when the target language was English.

3.2 Enhancements to the baseline systems

Our goal was to improve the translation quality by enhancing the the translation table.

The following enhancements were implemented:

- reduce the vocabulary size perceived by the GIZA++ and preset alignment for certain words
- "normalize" distortion between pairs of languages by reordering noun-adjective constructions

The first enhancement identifies pairs of tokens in the parallel sentences that, with a very high probability, align together and they don't align with other tokens in the sentence. These tokens are replaced with a special identifier, chosen so that GIZA++ will learn the alignment between them easier than before replacement. The targeted token types are proper nouns (detected when the same upper-cased token were present in both the foreign sentence and the English sentence) and numbers, also taking into account the differences between number representation in different languages (i.e.: 399.99 vs. 399,99). Each distinct proper noun to be replaced in the sentence was replaced with a specific identifier, distinct from other replacement identifiers already used in the sentence. The same procedure was applied also for numbers. The specific identifiers were reused in other sentences. This has the effect of reducing the vocabulary, thus it provides a large number of instances for the special token forms. The change in

[2] http://www.iccs.inf.ed.ac.uk/~pkoehn/training.tgz

[3] After the shared task submission, we optimized a lot our decoder. Before the optimizations (LM optimizations, fixing bugs that affected performance), `Phramer` was 5 to 15 times slower than `Pharaoh`.

[4] http://www.isi.edu/licensed-sw/carmel/

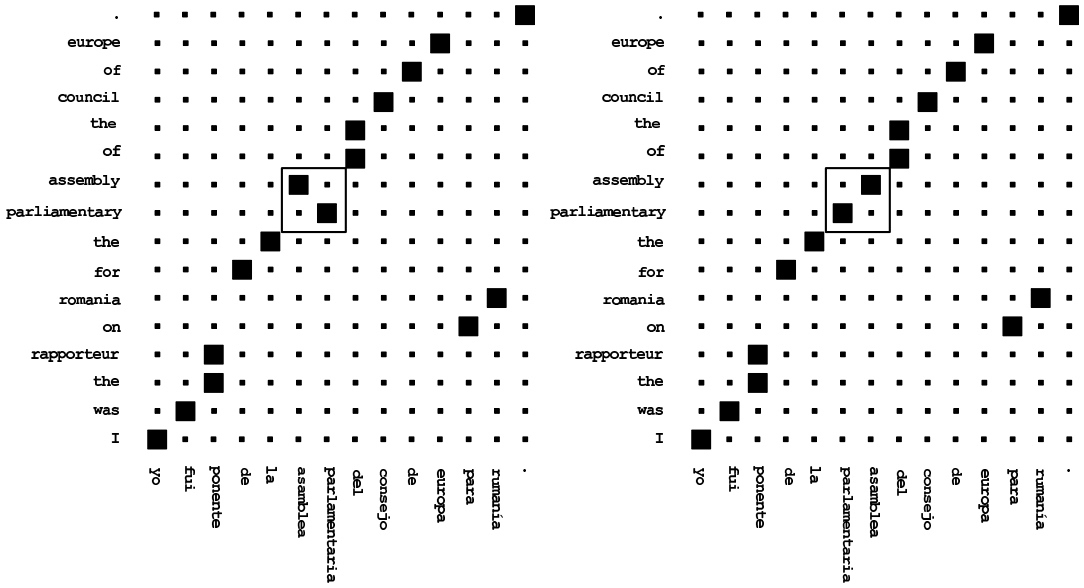

Figure 1: NN-ADJ reordering

Corpus	Before	After
DE	195,290	184,754
FR	80,348	70,623
ES	102,885	92,827

Table 1: Vocabulary size change due to forced alignment

the vocabulary size is shown in Table 1. To simplify the process, we limited the replacement of tokens to one-to-one (one real token to one special token), so that the word alignment file can be directly used together with the original parallel corpus to extract phrases required for the generation of the translation table. Table 2 shows an example of the output.

The second enhancement tries to improve the quality of the translation by rearranging the words in the source sentence to better match the correct word order in the target language (Collins et al., 2005). We focused on a very specific pattern – based on the part-of-speech tags, changing the order of NN-ADJ phrases in the non-English sentences. This process was also applied to the input dev/test files, when the target language was English. Figure 1 shows the reordering process and its effect on the alignment.

The expected benefits are:

- Better word alignment due to an alignment closer to the expected alignment (monotone).
- More phrases extracted from the word aligned corpus. Monotone alignment tends to generate more phrases than a random alignment.
- Higher mixture weight for the monotone distortion model because of fewer reordering constraints during MERT, thus the value of the monotone distortion model increases, "tightening" the translation.

3.3 Experimental Setup

We implemented the first enhancement on ES→EN subtask by part-of-speech tagging the Spanish text using `TreeTagger`[5] followed by a NN-ADJ inversion heuristic.

The language models provided for the task was used.

We used the 1,000 out of the 2,000 sentences in each of the *dev2006* datasets to determine weights for the 8 models used during decoding (one monotone distortion mode, one language model, five translation models, one sentence length model) through MERT. The weights were determined individually for each pair of source-target languages.

[5] http://www.ims.uni-stuttgart.de/projekte/corplex/TreeTagger/DecisionTreeTagger.html

There are **145** settlements in the West Bank , **16** in **Gaza** , **9** in East Jerusalem ; **400,000** people live in them .
Existen **145** asentamientos en Cisjordania , **16** en **Gaza** y **9** en Jerusaln Este ; en ellos viven **400.000** personas .
There are **[x1]** settlements in the West Bank , **[x2]** in **[y1]** , **[x3]** in East Jerusalem ; **[x4]** people live in them .
Existen **[x1]** asentamientos en Cisjordania , **[x2]** en **[y1]** y **[x3]** en Jerusaln Este ; en ellos viven **[x4]** personas .

Table 2: Forced alignment example

Subtask	OOV filtering	forced alignment	NN-ADJ inversion	BLEU score
DE→EN	√	—	—	25.45
	√	√	—	**25.53**
FR→EN	√	—	—	30.70
	√	√	—	**30.70**
ES→EN	√	—	—	30.77
	√	√	—	30.84
	√	√	√	**30.92**
EN→FR	—	—	—	31.67
	—	√	—	**31.79**
EN→ES	—	—	—	30.17
	—	√	—	**30.11**

Table 3: Results on the *devtest2006* files

Subtask	BLEU	1/2/3/4-gram precision (bp)
DE→EN	22.96	58.8/28.8/16.5/9.9 (1.000)
FR→EN	27.78	61.8/33.6/21.0/13.7 (1.000)
ES→EN	29.93	63.5/36.0/23.0/15.2 (1.000)
EN→FR	28.87	60.0/34.7/22.7/15.2 (0.991)
EN→ES	29.00	62.9/35.8/23.0/15.1 (0.975)

Table 4: Results on the *test2006* files

Using these weights, we measured the BLEU score on the *devtest2006* datasets. Based on the model chosen, we decoded the *test2006* datasets using the same weights as for *devtest2006*.

3.4 Results

Table 3 presents the results on the *devtest2006* files using different settings. Bold values represent the result for the settings that were also chosen for the final test. Table 4 shows the results on the submitted files (*test2006*).

3.5 Conclusions

The enhancements that we proposed provide small improvements on the *devtest2006* files. As expected, when we used the NN-ADJ inversion the ratio $\frac{\lambda_D}{\lambda_{LM}}$ increased from 0.545 to 0.675. The LM is the only model that opposes the tendency of the distortion model towards monotone phrase order.

`Phramer` delivers a very good baseline system. Using only the baseline system, we obtain +0.68 on DE→EN, +0.43 on FR→EN and -0.18 on ES→EN difference in BLEU score compared to WPT05's best system (Koehn and Monz, 2005). This fact is caused by the MERT module. This module is capable of estimating parameters over a large development corpus in a reasonable time, thus it is able to generate highly relevant parameters.

References

Michael Collins, Philipp Koehn, and Ivona Kucerova. 2005. Clause restructuring for statistical machine translation. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL'05)*, pages 531–540, Ann Arbor, Michigan, June. Association for Computational Linguistics.

Philipp Koehn and Christof Monz. 2005. Shared task: Statistical machine translation between European languages. In *Proceedings of the ACL Workshop on Building and Using Parallel Texts*, pages 119–124, Ann Arbor, Michigan, June. Association for Computational Linguistics.

Philipp Koehn, Franz Josef Och, and Daniel Marcu. 2003. Statistical phrase-based translation. In *Proceedings of HLT/NAACL 2003*, Edmonton, Canada.

Philipp Koehn. 2004. Pharaoh: A beam search decoder for phrase-based statistical machine translation models. In *Proceedings of AMTA*.

Franz Josef Och and Hermann Ney. 2003. A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29(1):19–51.

Franz Josef Och. 2003. Minimum error rate training in statistical machine translation. In Erhard Hinrichs and Dan Roth, editors, *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics*, pages 160–167.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 311–318.