HAL open science

Grapheme to phoneme conversion using an SMT system

Antoine Laurent, Paul Deléglise, Sylvain Meignier

To cite this version:

Antoine Laurent, Paul Deléglise, Sylvain Meignier. Grapheme to phoneme conversion using an SMT system. 10th Annual Conference of the International Speech Communication Association 2009 (INTERSPEECH 2009) , Sep 2009, Brighton, United Kingdom. pp.716-719. hal-01451534

HAL Id: hal-01451534

https://hal.science/hal-01451534v1

Submitted on 22 Mar 2017

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Grapheme to phoneme conversion using an SMT system

Antoine Laurent[1 2], *Paul Deléglise*[1], *Sylvain Meignier*[1]

[1] LIUM (Computer Science Research Center – Université du Maine) – Le Mans, France
[2] Spécinov – Trélazé, France
first.last@lium.univ-lemans.fr

Abstract

This paper presents an automatic grapheme to phoneme conversion system that uses statistical machine translation techniques provided by the Moses Toolkit. The generated word pronunciations are employed in the dictionary of an automatic speech recognition system and evaluated using the ESTER 2 French broadcast news corpus. Grapheme to phoneme conversion based on Moses is compared to two other methods: *G2P*, and a dictionary look-up method supplemented by a rule-based tool for phonetic transcriptions of words unavailable in the dictionary. Moses gives better results than *G2P*, and have performance comparable to the dictionary look-up strategy.

Index Terms: SMT, Moses, *G2P*, phonetic transcription

1. Introduction

The open source toolkit Moses [1] makes it easy to develop Statistical Machine Translation (SMT) applications and is widely used by researchers and companies. However, Moses can solve generic transduction problems and was applied successfully in morphological applications such as [2, 3].

In this article, we propose an automatic grapheme to phoneme conversion method based on SMT techniques. Instead of translating word sequences of a source language into word sequences of a target language, a sequence of words is re-written as a sequence of phonetic transcription represented by phonemes.

Common approaches to the problem of automatic grapheme to phoneme conversion were proposed in the literature, the most popular are: the dictionary look-up strategy, the rule-based approach [4], and the knowledge-based approach [5].

Word pronunciations are generated in order to be used in the dictionary of a speech recognition system. Evaluation is not focused on the accuracy of the phonetic transcription, but on the accuracy of the Automatic Speech Recognition (ASR) system, measured in terms of Word Error Rate (WER). The experiments are carried out using French broadcast news from the ESTER 2 evaluation corpus (2008).

One of the advantages of the proposed method is to convert a word depending of the word context. This permits to get rid of some ambiguous cases, such as heteronyms and liaisons (useful in French). A second advantage is that the use of an SMT system allows to strongly decrease the need of phonetics knowledge. While training is performed only over pronunciations of the 18k different words present in 23 hours of broadcast news, performance is close to the baseline ASR system.

2. Experimental context

2.1. Corpus

The methods are developed and tested with data from the ESTER 2 (2008) evaluation campaign [6]. The corpus was recorded from seven radio stations in French: France Inter, France Info, RFI, RTM, France Culture, Africa One and Radio Classique. They are divided into 3 corpus: the training corpus of 280 hours recorded from 1998 to 2003, the development corpus and the test corpus of 6 hours each, recorded both in 2007-2008.

2.2. Baseline transcription system

The LIUM ASR system is based on the CMU Sphinx system. This system was the best open source ASR system of the ESTER 2 evaluation campaign with 24.2% of WER on the development corpus and 19.3% on the test corpus.

The transcription decoding process is based on multi-pass decoding employing 39 dimensional PLP features. After segmentation and classification of the signal by speaker, a first decoding pass permits to compute a CMLLR transformation for each speaker. The second decoding pass using SAT and MPE acoustic models generates lattices used to drive a graph-decoding with full 3-phone contexts. All decoding employs tied-state word-position 3-phone acoustic models dependent on the gender of the speaker and on the bandwidh and 3-gram language model.

2.3. Vocabulary

The dictionary of this baseline system contains word of the BDLEX dictionary [7] (look-up strategy). Words that

are not present in this database have their phonetic transcriptions generated using LIA_PHON [4], a rule based grapheme to phoneme tool. The baseline decoding dictionary contains 122k words for 320k variants.

An analysis of the vocabulary present in the development corpus shows that it contains 8541 words and 64731 occurrences of those words. 220 words (507 occurrences) are not present in the decoding vocabulary. 93% of the words of the development corpus are also present in the training corpus.

3. Grapheme to phoneme conversion using Moses

A Statistical Machine Translation system (SMT) is used to transform text from a source language into a target language. The training step needs a data corpus which is composed of bitext data: source language sentences associated in parallel with target language sentences. The SMT system is based on the Moses toolkit. This toolkit is commonly used to translate corpus in which the elementary unit is the word in both the source and target parts.

3.1. Bitext corpus format

To convert graphemes to phonemes, a bitext would associate sequences of letters with sequences of phonemes. Table 1 shows three representation examples of the bitext corpus denoted A, B and C. In representation A, the sequence of letters corresponds to a word. In the two others representations, B and C, the sequence of letters corresponds to a group of words.

Groups of words are the longest sequence of words between two fillers. Indeed, we are doing the hypothesis that the influence of a word on the pronunciation of its neighbors is negligible when they are separated by a filler. In addition, representation C introduces a symbol to mark the limit of each word.

Table 1: *Representations A, B and C of the bitext corpus examples (phonemes given in Sampa format)*

Rep.	Graphemes	Phonemes
A	d e s j e u n e s f i l l e s	d E Z 9 n f i j
B	d e s j e u n e s f i l l e s	d E Z 9 n f i j
C	d e s # j e u n e s # f i l l e s #	d E # Z 9 n # f i j #

Representations B and C allow to take into account phonological rules; representation C allows to differentiate inter- and intra-word influences.

In representations B and C, the sequence of phonemes is built by a forced alignment using the baseline acoustic models and the baseline dictionary.

3.2. Learning a statistical translation system

The training of a grapheme to phoneme translation model is similar to the one of a translation model as described in the Moses documentation. We optimize the five weights present in the model. However two training strategies are proposed: the first one corresponds to the standard Moses training framework based on the maximization of the BLEU score [8]; the second one, based on the Levenshtein metric, minimizes the insertion, deletion and substitution errors of phonemes.

3.2.1. BLEU score

The training reserves 3% of the corpus for optimization of the parameters according to the BLEU score. Experiments show that the best score is obtained by using a distortion model (allowing to permute phonemes) for representation A, while models B and C give their best score without it.

3.2.2. Levenshtein score

We propose a different criteria than the BLEU score based on the edition distance of Levenshtein.

At the end of a training iteration, 3-best phonetic transcriptions for each training example (sequence of letters) are generated using the current translation model. The sum of the normalized Levenshtein measures, S, is computed between the phonetic transcriptions and the references (equation 1).

$$S = \sum_{t \in T} \log(1 - \frac{d_t}{l_t}) \quad (1)$$

where d_t is the edition distance of Levenshtein of phonetic transcription t, l_t is the length of the reference of phonetic transcription corresponding to t, T is the set of generated phonetic transcriptions.

Until getting the lowest S over all the training examples, a simplex framework[1] permits to tune the model parameters.

In this method, the language model weight is fixed to 0.1 and we do not use a distortion model.

3.2.3. Performance (evaluation and results)

Using the Levenshtein criteria, optimization is done in about 128 passes, and takes about 10 hours, whereas optimization time using BLEU score is only 4 hours (plus 2 hours to compute the phrase table).

The phonetic transcriptions of the 122k words of the baseline dictionary are generated using representation A, B or C of the training data and one of the 2 optimization criteria. WER over the development corpus using the 6 generated dictionaries are presented in table 2.

[1] Thanks to the Condor toolkit [9]

Table 2: *WER on development corpus using various optimization methods*

Optimization method	Representation	WER
BLEU score	A	26.9 %
BLEU score	B	27.2 %
BLEU score	C	26.9 %
Levenshtein	A	29.0 %
Levenshtein	B	27.5 %
Levenshtein	C	26.0 %

The translation model, computed according to the Levenshtein criteria and representation C, gives the lowest WER. Other experiments reported in this paper are performed in these conditions. The introduction of a symbol to mark the limit of each word in representation C decreases the WER.

4. Training an ASR system with few lexical resources

We make the hypothesis that an expert proposed the phonetic transcriptions of a small amount of words, and the goal is to build an ASR system that only uses that linguistic knowledge.

4.1. Linguistics data

The number of words with their phonetic transcriptions is limited to the 18k words of the first 23 hours of the training corpus. These words cover about 90% of the occurrences of words in the development corpus (292k occurrences) and represent 64.73% of the words of the development corpus.

Training data is selected in order to keep about 10% of word occurrences out of the vocabulary (about 3k words).

4.2. Dictionary generation

The translation system generates the phonetic transcription of 122k words of the baseline dictionary. This dictionary is denoted as *Auto* and supply an overall of 301k phonetic transcription variants.

The phonetic transcriptions drawn from *Auto* are filtered to discard unfit phonetic transcriptions. The filtering method consists in keeping only the phonetic transcriptions that can be aligned along the training data [10] (using the acoustic models of the baseline system). The new dictionary is denoted *Filtered*. *Filtered* consists of 247k pronunciations, *ie* filtering step discards about 54k pronunciations.

The 18k words given by the expert have 55k phonetic transcriptions drawn from *BDLEX*. We denote this dictionary as $Expert$. The *Union* contains the $Expert$ dictionary and the 104k remaining words (122k - 18k) drawn from the *Filtered* dictionary.

4.3. Moses vs G2P

G2P is a grapheme to phoneme conversion system that uses joint-sequence models [5]. It is a *data-driven* conversion system that is based on the idea that, given enough examples, it should be possible to predict the pronunciation of unseen words, purely by analogy.

In order to evaluate SMT, we compare the performance with *G2P* under the same conditions. Because computing time on representations B and C is very expensive using *G2P*, it is trained only over representation A. Learning time on this small corpus is about 2.5 times as much for *G2P* (28 hours) as it is for Moses (12 hours).

G2P generates 358k variants for the 122k words. After filtering, the number of variants decreased to 279k.

Table 3: *WER using G2P and SMT methods with various dictionaries (development corpus)*

Method	Dictionary	WER
SMT	*Auto*	27.1 %
SMT	*Filtered*	26.5 %
SMT	*Union*	24.8 %
G2P	*Auto*	27.7 %
G2P	*Filtered*	27.5 %
G2P	*Union*	25.0 %

Table 3 shows WER over the development corpus for the SMT and *G2P* methods and their 3 dictionaries. Filtering decreases the WER in both systems. SMT gives lower WER than *G2P* with every dictionary. The best result is obtained using SMT with the *Union* dictionary: 18k words from the $Expert$ dictionary and 104k words automatically generated by SMT and filtered along the training corpus.

Another comparison is proposed using the Phoneme Error Rate [5] (PER). Table 4 shows that the WER is not linked to the PER. The PER is not reliable enough to evaluate a phonetic transcription generation system if we want to use it with an ASR system.

Table 4: *PER and WER using Auto dictionaries (development corpus)*

Method	PER	WER
SMT	13.1 %	27.1 %
G2P	11.3 %	27.7 %

4.4. New acoustic model

In every experiment made previously, the acoustic model was built using the baseline dictionary and this model was not called into question. We propose to build a new

acoustic model from scratch. Training is done by adding new informations at each step. The first acoustic model is trained using the 18k words from the $Expert$ dictionary and the 23 hours training corpus. The last step uses the 122k words from the SMT *Union* dictionary over the full 280 hours training corpus.

The new acoustic model calls into question the filtering stage of dictionary generation. A new *Union* dictionary is built according to the new *Filtered* dictionary.

Table 5 presents results using the acoustic baseline model and our new acoustic model using various dictionaries. The best result is obtained with the baseline dictionary using the *BDLEX* database. However, a system built with only 18k words made by an human expert gives relatively close results.

Table 5: *WER using alignment with the new acoustic model (development corpus)*

Acoustic models	Dictionary	WER
Reference	Reference	24.2 %
Reference	Old *Union*	24.8 %
New	New *Union*	24.7 %

Finally, we evaluated the decoding system on the ESTER 2 test corpus (Table 6). Results show a relatively weak gap between the two systems. Thus, the contribution of phonetic knowledge can be strongly reduced in an ASR system using SMT.

Table 6: *WER on ESTER 2 test corpus using our new system compared to the baseline system*

Acoustic models	Dictionary	WER
Reference	*Reference*	19.3 %
New	*Union*	19.5 %

5. Conclusion

This paper presents an automatic grapheme to phoneme conversion system based on statistical machine translation techniques employing the Moses Toolkit. We show that Moses is a good alternative to *G2P*: the learning time of Moses is shorter, and the phonetic transcriptions generated by Moses and employed in an ASR dictionary give a lower WER. Moreover, only 18k words with theirs phonetic transcriptions are enough to learn an SMT-based phonetic transcription system. The performance of this system is close to the baseline system, based on a look-up strategy supplemented by a rule-based tool. Furthermore, the proposed method could be very useful to develop an ASR system for a new language, especially if we have few available linguistic knowledge for this language.

6. Acknowledgements

We thank Holger Schwenk for helpful discussions and comments about SMT strategy and tools.

7. References

[1] P. Koehn, H. Hoang, A. Birch, C. Calisson-Burch, M. Federico, N. Bertholdi, B. Cowan, W. Shen, C. Moran, R. Zens, C. Dyer, O. Bojar, A. Constantin, and E. Herbst, "Moses : Open-source toolkit for statistical machine translation," in *Proc. of Association for Computational Linguistics*, 2007.

[2] M. Dreyer, J. R. Smith, and J. Eisner, "Latent-variable modeling of string transductions with finite-state methods," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Honolulu, October 2008, pp. 1080–1089.

[3] C. Kobus, F. Yvon, and G. Damnati, "Normalizing SMS: are two metaphors better than one?" in *COLING '08*, 2008.

[4] F. Béchet, "LIA_PHON : un système complet de phonétisation de textes," in *TAL, Traitement Automatique des Langues*, 2001, pp. 47–67.

[5] M. Bisani and H. Ney, "Joint-sequence models for grapheme-to-phoneme conversion," *Speech Comm.*, vol. 50, no. 5, pp. 434–451, 2008.

[6] AFCP, "Évaluation des systèmes de transcription enrichie d'émissions radiophoniques, plan d'évaluation d'ESTER phases 1 et 2," October 2008.

[7] I. Ferrané, M. De Calmes, J. Pecatte, and G. Perennou, "Besoins lexicaux à la lumière de l'analyse statistique du corpus de textes du projet BREF : le lexique BDLEX du français écrit et oral," in *Proc. of Association for Computational Linguistics*, 1992.

[8] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "BLEU: a method for automatic evaluation of machine translation," in *Proc. of Association for Computational Linguistics*, 2002.

[9] F. V. Berghen and H. Bersini, "CONDOR, a new parallel, constrained extension of Powell's UOBYQA algorithm: Experimental results and comparison with the DFO algorithm," *Journal of Computational and Applied Mathematics*, vol. 181, pp. 157–175, 2005.

[10] A. Laurent, T. Merlin, S. Meignier, Y. Estève, and P. Deléglise, "Iterative filtering of phonetic transcriptions of proper nouns," in *Proc. of ICASSP, International Conference on Acoustics, Speech, and Signal Processing*, April 2009.