Available online at www.sciencedirect.com

SciVerse ScienceDirect

Computer Speech and Language 28 (2014) 141–162

COMPUTER SPEECH AND LANGUAGE

www.elsevier.com/locate/csl

An improved two-stage mixed language model approach for handling out-of-vocabulary words in large vocabulary continuous speech recognition☆

Bert Réveil*, Kris Demuynck, Jean-Pierre Martens

Ghent University – iMinds, ELIS Multimedia Lab, Sint-Pietersnieuwstraat 41, 9000 Gent, Belgium

Received 17 May 2012; received in revised form 23 March 2013; accepted 3 April 2013
Available online 17 April 2013

Abstract

This paper presents a two-stage mixed language model technique for detecting and recognizing words that are not included in the vocabulary of a large vocabulary continuous speech recognition system. The main idea is to spot the out-of-vocabulary words and to produce a transcription for these words in terms of subword units with the help of a mixed word/subword language model in the first stage, and to convert the subword transcriptions to word hypotheses by means of a look-up table in the second stage. The performance of the proposed approach is compared to that of the state-of-the-art hybrid method reported in the literature, both on in-domain and on out-of-domain Dutch spoken material, where the term 'domain' refers to the ensemble of topics that were covered in the material from which the lexicon and language model were retrieved. It turns out that the proposed approach is at least equally effective as a hybrid approach when it comes to recognizing in-domain material, and significantly more effective when applied to out-of-domain data. This proves that the proposed approach is easily adaptable to new domains and to new words (e.g. proper names) in the same domain. On the out-of-domain recognition task, the word error rate could be reduced by 12% relative over a baseline system incorporating a 100k word vocabulary and a basic garbage OOV word model.

Keywords: Out-of-vocabulary words; OOV detection; OOV modeling; Phoneme-to-grapheme conversion

1. Introduction

The performance of any large vocabulary continuous speech recognizer (LVCSR) is bounded by the finity of its vocabulary. Words outside that vocabulary are called out-of-vocabulary (OOV) words and cannot be recognized without additional processing steps. If such steps are not pursued, it is generally acknowledged that each OOV word occurring in the speech signal translates to multiple errors in the recognition output: an early measurement for English (Hetheringthon, 1995) counted 1.22 errors per OOV word, but currently, one typically anticipates an average of 1.5 to more than 2 errors per OOV word (Adda-Decker and Lamel, 2000; Bisani and Ney, 2005).

Given that computer resources (memory and computation speed) are nowadays less of a concern, there is a tendency to reduce the word error rate (WER) by simply including more words in the vocabulary. It has been argued (Rastrow

☆ This paper has been recommended for acceptance by A. Potamianos.

* Corresponding author. Tel.: +32 09 264 33 97; fax: +32 09 264 35 94.
E-mail addresses: breveil@elis.ugent.be (B. Réveil), kris.demuynck@elis.ugent.be (K. Demuynck), martens@elis.ugent.be (J.-P. Martens).

0885-2308/$ – see front matter © 2013 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.csl.2013.04.003

et al., 2009; Parada et al., 2010a) that this approach has a down-side, namely that it can raise the confusability among the in-vocabulary (IV) words, and as such end up in raising the WER. Moreover, it is impossible to create a vocabulary that contains all words that will ever be spoken to the recognizer during its deployment. This is especially unfortunate in case of unforeseen content words. In Demuynck et al. (2009), it has been observed that for a vocabulary size of more than 100k words, 90% of the OOV words in a large Dutch newspaper text corpus are content words such as compound nouns and proper names. It is clear that an incorrect recognition of these content words will hamper the comprehension or the further processing of the recognition output in applications such as automatic translation or spoken term detection. That is why pursuing the lowest possible OOV word rates is always important.

A first step in that pursuit is choosing an optimal set of lexical units. This choice is language-specific. English for instance is known to exhibit only small amounts of inflection and compounding, which implies that regular words make good lexical units. In highly agglutinative languages such as Finnish or Turkish on the other hand, morphemes are typically preferred over words. The usage of subwords as lexical units of course implies that the language model needs to be adjusted to be able to join subwords back to words. A complete description of such a system, including the automatic derivation of morphemes, can be found in Hirsimäki et al. (2006). Other languages such as Mandarin Chinese may even benefit from combining syllabic character-based lexical units with the more traditional word-based units (Hieronymus et al., 2009).

For a mildly generative language such as Dutch, it has been shown (Demuynck et al., 2009) that for vocabulary sizes of 100k entries or more, regular words make good lexical units: word inflections are effectively covered by vocabularies of such sizes, while OOV compound words can to a large extent be recovered by means of a post-processing step that merges individual successive words to form compounds.

However, as mentioned above, a large portion of the OOV words (in Dutch as well as in other languages) comprise proper names and other word categories such as foreign words that do not follow the language-specific morphological rules. Hence, OOV words remain a problem, irrespective of the lexical units that are chosen. In this paper, we therefore try to go beyond the specific optimization of the recognition vocabulary and conceive a methodology that aims to recover all types of OOV words.

Conceptually, most of such complete OOV word handling methods consist of the following three steps: (1) the detection (spotting) of regions in the speech signal that contain an OOV word, (2) the generation of a subword representation of the detected region, and (3) the generation of an orthographic representation for the detected region. From a practical viewpoint, the different steps may be inter-twinned. In their simplest form for instance, the second and third step together may just generate a filler symbol like "<OOV>" or an in-vocabulary word between brackets to alert the user that the word at that position is probably an unknown word. Using this approach, one could already obtain WER improvements: e.g. if the recognizer is forced to ignore presumed OOV word segments in its decoding of the surrounding speech, a recognition error caused by an OOV word might not propagate.

If the user is a human being, like a hearing-impaired person reading subtitles, it may also be acceptable to generate a pseudo-orthographic transcription of the OOV word, provided that this word obeys the pronunciation rules of the language and reads as the OOV word. Such a system could for instance output '[Perris]' for the capital of France, with the brackets indicating that it is a pseudo-transcription. In that respect, we refer to the work of Decadt et al. (2001) where alternative orthographic transcriptions were created for words in the recognition output that had a low word confidence score (most of these words were OOV words). Their system uses a phoneme recognizer incorporating a 5-gram phoneme LM to generate a phonemic transcription for these words, after which a phoneme-to-grapheme (P2G) converter is employed to generate a graphemic transcription. Although the obtained graphemic strings were usually not fully correct, the resulting speech transcriptions were easier to read than the original ones.

The ultimate goal of an OOV word handling method is however to produce a full and correct orthographic representation of every OOV word, and to recover most, if not all, of the errors in non-OOV regions emanating from the presence of OOV words.

In the rest of this paper, we first review some previously proposed OOV word handling methods that aim at detecting OOV word regions in the speech and at producing an orthographic transcription for these regions (Section 2). Our own OOV word detection and recovery method is introduced in Section 3. The approach combines the knowledge of an open-vocabulary word-level language model (LM) with that of a subword-level generative model of OOV words and a background pronunciation dictionary that was created in a fully automatic way. Section 4 discusses our experimental set-up. In Section 5, we conduct an experimental study (for the Dutch language) to assess the performance of the most popular existing method and our proposed method as a function of the control variables they embed for optimizing their