Available online at www.sciencedirect.com

ScienceDirect

Computer Speech and Language 39 (2016) 47–66

COMPUTER SPEECH AND LANGUAGE

www.elsevier.com/locate/csl

Unsupervised language identification based on Latent Dirichlet Allocation☆

Wei Zhang [a,b,*], Robert A.J. Clark [b,**], Yongyuan Wang [a], Wen Li [a]

[a] *Dept. of Comp. Sci. and Tech., Ocean University of China, Qingdao 266100, China*
[b] *The CSTR, University of Edinburgh, Edinburgh EH89AB, United Kingdom*

Received 21 May 2015; received in revised form 7 December 2015; accepted 24 February 2016
Available online 14 March 2016

Abstract

To automatically build, from scratch, the language processing component for a speech synthesis system in a new language, a purified text corpora is needed where any words and phrases from other languages are clearly identified or excluded. When using found data and where there is no inherent linguistic knowledge of the language/languages contained in the data, identifying the pure data is a difficult problem. We propose an unsupervised language identification approach based on Latent Dirichlet Allocation where we take the raw *n*-gram count as features without any smoothing, pruning or interpolation. The Latent Dirichlet Allocation topic model is reformulated for the language identification task and Collapsed Gibbs Sampling is used to train an unsupervised language identification model. In order to find the number of languages present, we compared four kinds of measure and also the Hierarchical Dirichlet process on several configurations of the ECI/UCI benchmark. Experiments on the ECI/MCI data and a Wikipedia based Swahili corpus shows this LDA method, without any annotation, has comparable precisions, recalls and *F*-scores to state of the art supervised language identification techniques.

Keywords: Language filtering; Language purifying; Language identification

1. Introduction

The motivation for our approach to language identification comes from the specific application of building speech synthesis systems in under-resourced languages. In the field of speech synthesis we have developed techniques to automatically prepare recording scripts and derive front-ends—the front end of a speech synthesis system is the language processing part of the system and generally consists of a series of modules that convert the input text to a phonemic representation including linguistic context that can then be synthesised by the acoustic back-end—from *found data* with little or no expert linguistic knowledge of the language. The general scenario is that we would take a text corpora in the new language in question and from it derive a front-end to phonetically process input text in this

☆ This paper has been recommended for acceptance by Geoffrey Zweig.

* Corresponding author at: Dep. of Comp. Sci. and Tech., Ocean University of China, Qingdao 266100, China. Tel.: +86 53266781727.
** Corresponding author.
E-mail addresses: weizhang@ouc.edu.cn (W. Zhang), robert@cstr.ed.ac.uk (R.A.J. Clark).

http://dx.doi.org/10.1016/j.csl.2016.02.001
0885-2308/

language, additionally we would create a recording script of up to a few thousand sentences to maximise phonetic coverage of the language which would be then recorded from a native speaker. Watts et al. (2013) show that we can efficiently use vector-space models to automatically build the language processing front-end of a speech synthesis system, and the recording script can trivially be created from a larger corpus using a greedy algorithm to optimise coverage. However, each of these steps is adversely affected by the purity of the larger corpus—the recording script is difficult to read if it contains other languages and the front-end models become polluted with characteristics of the other languages.

As our found data generally comes from sources such as Wikipedia or web-orientated news material and is in minority languages, it is often quite impure, either as the result of code switching, the inclusion of foreign names, or of being a partial translation from another language. We require a language purification tool to identify which sentences in a found corpus are suitable for inclusion in the sub-corpus used for speech synthesis. Existing language identification tools are generally unsuited to this task, primarily because the languages in question would not necessarily be supported by them. This paper discusses the development of a text purification tool along with its evaluation as both a tool for text purification and for the more general case of language identification.

2. Background and problem analysis

Language identification is generally viewed as a form of text categorization, many classification approaches have been used to identify the language of a document: Markov models combined with Bayesian classification (Dunning, 1994), discrete hidden Markov models (Xafopoulos et al., 2004), Kullback–Leibler divergence–namely relative entropy (Sibun and Reynar, 1996), minimum cross-entropy (Teahan, 2000), decision trees (Hakkinen and Tian, 2001), neural networks (Tian and Suontausta, 2003), support vector machines (Zhai et al., 2006), multiple linear regression (Murthy and Kumar, 2006), centroid-based classifications (Kruengkrai et al., 2005) and improvements to the previous method (Takçıand Güngör, 2012), conditional random fields (King and Abney, 2013), minimum description length with dynamic programming (Yamaguchi and Tanaka-Ishii, 2012) and bootstrapped methods such as Mayer (2012), Goldszmidt et al. (2013). These methods are all supervised and require clean editorially managed corpora for training. They are appropriate only for a limited number of languages, and require relatively large-sized documents. Lui and Baldwin (2012) do however provide a pre-trained off the shelf model for language identification. These tools and approaches perform well when there is sufficient data available, but there are problems when the text to be identified is out of domain, out of style, or includes language not found in the training corpora.

In the task we are addressing, other than assuming that the majority of the text is from the language we are interested in, we can make no assumptions about the number or quantity of other languages present. We require individual sentences to be classified as being purely from the language in question or containing other languages, this is slightly different task than determining the primary language of a larger document. For our requirement to purify a text where we have little linguistic knowledge of the language or languages present, this presents a problem and raises the key question: can the language of a sentence be automatically identified using unsupervised methods or can we at least identify sentences as being of different languages. Amine et al. (2010) demonstrate an approach using similarity measures, but performance is greatly reduced when compared to supervised methods. Biemann and Teresniak (2005) present a promising co-occurrence words graph approach, namely Chinese Whispers (Biemann, 2006), claiming an $F1$ score of 99%, but their work focuses on long documents (each language present must have a minimum of 100 sentences). Shiells and Pham (2010) incorporate what they call the "purity" and "authority" into Chinese Whispers to identify the language of one million short Tweets. They find that the algorithm does not seem to converge when using Twitter data as opposed to when using much longer documents (Biemann and Teresniak, 2005; Biemann, 2006) due to many short Tweets mixing words from more than one language. That is there are many more edges between language clusters. Another issue is that using words directly as features means that this kind of algorithm is affected by the tokenization performance for languages that do not use white space to mark word boundaries.

Collectively this means that although the Chinese Whispers approach and its improvements are effective for some scenarios, it is unsuitable for our particular task. Furthermore, experimentation with existing techniques highlights three main requirements we need to fulfill. We require a technique that: (1) equally applicable to long and short documents; (2) that when used in filtering, can filter those documents mixed with words from other languages; and (3) that can deal with all languages including those that cannot be tokenised using white-space.