Synthesis Lectures on
Human Language Technologies

SYNTHESIS
COLLECTION OF TECHNOLOGY

Tommi Jauhiainen · Marcos Zampieri ·
Timothy Baldwin · Krister Lindén

Automatic Language Identification in Texts

Springer

Synthesis Lectures on Human Language Technologies

Series Editor

Graeme Hirst, Department of Computer Science, University of Toronto, Toronto, ON, Canada

The series publishes topics relating to natural language processing, computational linguistics, information retrieval, and spoken language understanding. Emphasis is on important new techniques, on new applications, and on topics that combine two or more HLT subfields.

Tommi Jauhiainen · Marcos Zampieri ·
Timothy Baldwin · Krister Lindén

Automatic Language Identification in Texts

Springer

Tommi Jauhiainen
University of Helsinki
Helsinki, Finland

Marcos Zampieri
George Mason University
Fairfax, VA, USA

Timothy Baldwin
MBZUAI
Abu Dhabi, United Arab Emirates

Krister Lindén
University of Helsinki
Helsinki, Finland

ISSN 1947-4040 ISSN 1947-4059 (electronic)
Synthesis Lectures on Human Language Technologies
ISBN 978-3-031-45821-7 ISBN 978-3-031-45822-4 (eBook)
https://doi.org/10.1007/978-3-031-45822-4

This Springer imprint is published by the registered company Springer Nature Switzerland AG
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

Paper in this product is recyclable.

Foreword

The fields of Computational Linguistics and Natural Language Processing (NLP) have advanced extraordinarily in recent years. This has been enabled by: (a) improved hardware, in terms of computational power and storage capacity; (b) data availability, in terms of variety of corpora; and (c) new algorithms, including recent breakthroughs in deep learning and the emergence of large-scale pretrained language models. Unfortunately, human languages have not benefited from these advances equally, as mainstream research has focused primarily on English, and to a lesser extent on just a handful of other "high-resource" languages.

Fortunately, over time, as the Internet has become increasingly multilingual, resources and tools have been gradually developed for many other languages, both monolingual and multilingual. With the emergence of NLP applications that can handle multiple languages, there has also been an increasing need for these applications to know the language of their input. For example, a system for extracting the text from a scanned document using optical character recognition (OCR) needs to know the language of the input document, or it would risk generating a non-sense sequence of letters, e.g., just because French and Hungarian share the same alphabet, does not mean that one can use a model developed for the former to do OCR for the latter. Similarly, a machine translation system needs to know the language of its input, or it could choose a wrong translation, e.g., due to false friends: for example, 'gift' can be an English word, but also a German one meaning 'poison', or a Norwegian one meaning 'married'. Practically, any NLP application dealing with text needs to know the language of its input so that it can choose appropriate text processing components in its NLP pipeline, as even very basic processing such as tokenization is done differently for different languages, e.g., Arabic, Chinese, and Vietnamese, typically require word segmentation, while for languages like English, Spanish, and Russian, this is typically not needed.

Of course, the system could ask the user to specify the language, but the user might select the wrong language or the wrong language variety (e.g., in the case of a long list of language choices), or the user might simply not know it for sure (e.g., a random tweet that is written using the Arabic script could be in Arabic, but also in Persian; similarly, a document using the Cyrillic alphabet could be in Russian, but also in Ukrainian, Bulgarian, and even in Mongolian). Moreover, there are cases where asking the user is not an option, e.g., when crawling documents from the Web. Therefore, many real-world systems have a built-in language identification component, e.g., Google Translate.

Yet, it is natural to ask how hard the problem actually is. After all, if the input is long enough (e.g., a full article) and if the distinction is between somewhat unrelated languages (e.g., between Portuguese and German), language identification can be trivial for speakers of the respective language. In such a scenario, the task can also easily be solved by automatic systems with almost 100% accuracy, e.g., using a character-level language model, or using word and character *n*-grams as features in a classifier. However, this is much harder if the system has only seen a small piece of text as input, but it needs to make a decision. For example, even a speaker of the respective languages cannot be sure whether the fragment “Alle mennesker er født frie og…” is in Danish or in Norwegian. This is because Danish and Norwegian are closely related languages with substantial overlap in terms of grammar and vocabulary, and this fragment is perfectly fluent in both: indeed, it is the beginning of the Universal Declaration of Human Rights in both languages. It is even harder to distinguish between language varieties, e.g., between Portuguese from Brazil and Portuguese from Portugal, or between German from Germany and German from Austria. Dialects are even trickier as they might not have a universally accepted grammar and spelling convention, nor do they have clear boundaries. Finally, there is the case of code-switching, where languages, language varieties, and dialects get mixed together in the same text, and even in the same sentence. For example, the sentence “*Pero* why do I have to go *a la casa*?” mixes Spanish (in italic) and English. It is important for a system to recognize such switches, so that it can process each language fragment accordingly, e.g., a speech recognition system or a machine translation trying to generate an Arabic translation from such an input would suffer badly when presented with such code-switched text; yet, asking for user guidance on the language of each fragment is not practical in such a scenario, and it is best if the system can perform the automatic detection on the fly.

Traditionally, research on language identification has been done separately in the speech and in the NLP communities due to the difference in the input (sound versus text). The applications of language identification in speech versus text, as well as the approaches, are also quite different, and they have been studied by different communities. Thus, it makes sense to focus on one or the other, which is also the approach taken in this book.

“Automatic Language Identification in Texts” offers a thorough survey of recent work on language identification, covering a number of important aspects such as closely related languages, low-resource and open-class scenarios, short input, code-switching, etc.

It further discusses a variety of approaches, datasets, shared tasks, and applications to various NLP tasks. Overall, the authors have done an excellent job, and the book is a must-read for anybody interested in the challenging but important problem of language identification.

Abu Dhabi, United Arab Emirates
September 2022

Preslav Nakov

Acknowledgments

In his Ph.D. dissertation, Tommi Jauhiainen, the first author of this book, investigated the task of identifying the language of digitally encoded text (Jauhiainen 2019). The dissertation included seven scientific articles, the first of which was the largest survey on language identification available at the time (Jauhiainen et al. 2019e) which was co-authored by the other authors of this book. In authoring this book, we drew on both the Ph.D. dissertation of the first author as well as the survey. The survey itself was partly based on the dissertation of Marco Lui (Lui 2014), who we thank for his support with this book as well. We would also like to acknowledge and thank Graeme Hirst and Mike Morgan for their extraordinary patience.

April 2023

Tommi Jauhiainen
Marcos Zampieri
Timothy Baldwin
Krister Lindén

References

T. Jauhiainen, Language identification in texts, PhD thesis, University of Helsinki, Finland, 2019

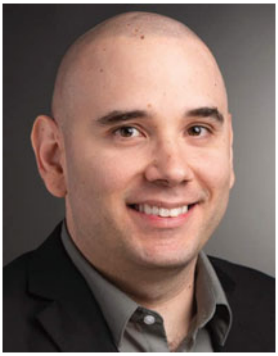
T. Jauhiainen, M. Lui, M. Zampieri, T. Baldwin, K. Lindén, Automatic language identification in texts: A survey. J. Artifi. Intelligence Res. **65**, 675–782, (2019e). ISSN 1076-9757. URL https://doi.org/10.1613/jair.1.11675

M. Lui, Generalized language identification. PhD thesis, The University of Melbourne, 2014

Contents

About the Authors

Tommi Jauhiainen wrote his master's thesis on automatic language identification in 2010. He continued his research on the same subject in the Finno-Ugric Languages and the Internet project as a doctoral student under the guidance of Dr. Krister Lindén and Prof. em. Kimmo Koskenniemi. During his studies, he was also enlisted as an information systems manager at the National Library of Finland, where he gained experience in software engineering projects, both small and large. He defended his doctoral thesis in 2019 on the topic of "Language Identification in Texts". He organized the first shared task in Cuneiform Language Identification (CLI) in 2019 and the Uralic Language Identification (ULI) shared tasks in 2020 and 2021. Currently, he works as a post-doctoral researcher at the University of Helsinki. He is the first author of c. 20 peer-reviewed publications on language identification.



Marcos Zampieri is an Assistant Professor at George Mason University in Virginia, USA. He received his Ph.D. from Saarland University with a thesis on computational modelling of language variation. He has published over 100 peer-reviewed papers on various topics in computational linguistics and NLP such as language and dialect identification, native language identification, machine translation, lexical complexity prediction, and social media mining. He is one of the editors of Similar Languages, Varieties, and Dialects: A Computational Perspective that appeared in the Cambridge University Press series Studies in Natural Language Processing.

Timothy Baldwin is Acting Provost and Chair of the Department of Natural Language Processing, Mohamed bin Zayed University of Artificial Intelligence (MBZUAI) in addition to being a Melbourne Laureate Professor in the School of Computing and Information Systems, The University of Melbourne. Prior to joining The University of Melbourne in 2004, he was a Senior Research Engineer at the Center for the Study of Language and Information, Stanford University (2001–2004). He is the author of over 450 peer-reviewed publications across diverse topics in natural language processing and AI, in addition to being an ARC Future Fellow, and the recipient of a number of prestigious awards at top conferences. He is currently Past President of the Association for Computational Linguistics (ACL), in addition to being a Permanent Member of the International Committee on Computational Linguistics (ICCL).

Krister Lindén is Research Director of Language Technology at the University of Helsinki, Finland, and National Coordinator of FIN-CLARIN—the Finnish Node of CLARIN ERIC—a European research infrastructure for Social Sciences and the Humanities. He is Chair of the CLARIN National Coordinators Forum and a member of CLIC (Committee for Legal and Ethical Issues in CLARIN). He holds a doctoral degree in Language Technology from the University of Helsinki. He is the co-author of more than 160 publications related to language technology and its utilization in digital humanities and language resource processing. He is currently also a deputy team leader in the Centre of Excellence of Ancient Near Eastern Empires.

1 Introduction to Language Identification

Language identification (LI) is the task of predicting the language(s) in a text or speech input. The main difference between LI of text and speech is that the characters that make up the text are discrete, whereas with speech, the input is usually a continuous signal. This means that different styles of mathematical methods are needed to process text and speech, traditionally with little methodological overlap between them. In this book, we focus on the language identification of digital text, although we do touch on applications to speech in the case that the speech signal has been translated into a sequence of (discrete) phones.

Recognizing the language(s) that a text is written in comes naturally to a human reader familiar with the language(s). Table 1.1 presents excerpts from Wikipedia articles in four different European languages on the topic of Natural Language Processing (NLP), labeled according to the language they are written in. Without referring to the labels, readers of this book will certainly recognize at least one language, and many are likely to identify all of them, even if they can’t read the content in all cases.

Research into LI aims to mimic this human ability to recognize specific languages. Over the years, several computational approaches have been developed that, through the use of specially-designed algorithms and indexing structures, are able to infer the language being used without the need for human intervention. In terms of coverage, the capability of such systems could be described as super-human: an average person may be able to identify a handful of languages, and a trained linguist or translator may be familiar with many dozens, but most of us will have, at some point, encountered written texts in languages we cannot place. However, LI research aims to develop systems that are able to identify *any* human language, a set which numbers in the thousands (Simons and Fennig 2018).

Research to date on LI has traditionally focused on *monolingual* documents (Hughes et al. 2006) (we discuss LI for multilingual documents in Chap. 5). In monolingual LI, the task is to assign a unique label to each document. Some work has reported near-perfect accuracy for LI of large documents in a small number of languages, prompting some researchers to label

T. Jauhiainen et al., *Automatic Language Identification in Texts*, Synthesis Lectures on Human Language Technologies, https://doi.org/10.1007/978-3-031-45822-4_1

Table 1.1 Excerpts from Wikipedia articles on NLP in different languages

Text line	Language
Natural language processing is a field of computer science, artificial intelligence, and linguistics concerned with the interactions between computers and human (natural) languages.	English
L'Elaborazione del linguaggio naturale è il processo di trattamento automatico mediante un calcolatore elettronico delle informazioni scritte o parlate nel linguaggio umano o naturale.	Italian
Luonnollisen kielen prosessointi liittyy ihmisten puhumien kielten ja tietokoneiden väliseen vuorovaikutukseen.	Finnish
In der Computerlinguistik (CL) oder linguistischen Datenverarbeitung (LDV) wird untersucht, wie natürliche Sprache in Form von Text- oder Sprachdaten mit Hilfe des Computers algorithmisch verarbeitet werden kann.	German

it a "solved task" (McNamee 2005). However, in order to attain such accuracy, simplifying assumptions have to be made, such as the aforementioned monolinguality of each document, as well as assumptions about the type and quantity of data, and the number of languages considered.

The ability to accurately detect the language that a document is written in is an enabling technology that increases accessibility of data and has a wide variety of applications. For example, presenting information in a user's native language has been found to be a critical factor in attracting website visitors (Kralisch and Mandl 2006). Text processing techniques developed in natural language processing and information retrieval generally presuppose that the language of the input text is known, and many techniques assume that all documents are in the same language. In order to apply text processing techniques to real-world data, automatic LI is used to ensure that only documents in relevant languages are subjected to further processing. In information storage and retrieval, it is common to index documents in a multilingual collection by the language they are written in, and LI is necessary for document collections where the languages of documents are not known a-priori, such as for data crawled from the World Wide Web. Another application of LI that predates computational methods is the detection of the language of a document for routing to a suitable translator. This application has become even more prominent due to the advent of Machine Translation (MT) methods: to apply MT to translate a document to a target language, it is generally necessary to determine the source language of the document, and this is the task of LI. LI also plays a part in providing support for the documentation and use of low-resource languages. One area where LI is frequently used in this regard is in linguistic corpus creation, where LI is used to process targeted web crawls to collect text resources for low-resource languages.

It should be noted that in this book, we do not make a distinction between languages, language varieties, and dialects. Whatever demarcation is made between languages, varieties and dialects, a LI system is trained to identify the associated classes. Of course, the more