Multilingualism in Ancient Texts: Language Detection by Example of Old High German and Old Saxon

Zahurul Islam[1], Roland Mittmann[2], Alexander Mehler[1]
[1]AG Texttechnology, Institut für Informatik, Goethe-Universität Frankfurt
[2]Institut für Empirische Sprachwissenschaft, Goethe-Universität Frankfurt
E-mail: zahurul, mittmann, mehler@em.uni-frankfurt.de

Abstract

In this paper, we present an approach to language detection in streams of multilingual ancient texts. We introduce a supervised classifier that detects, amongst others, Old High German (OHG) and Old Saxon (OS). We evaluate our model by means of three experiments that show that language detection is possible even for dead languages. Finally, we present an experiment in unsupervised language detection as a tertium comparationis for our supervised classifier.

Keywords: Language identification, Ancient text, n-gram, classification, clustering

1. Introduction

With the rise of the web, we face more and more on-line resources that mix different languages. This multilingualism of textual resources poses a challenge for many tasks in Natural Language Processing (NLP). As a consequence, Language Identification (LI) is now an indispensable step of preprocessing for many NLP applications. This includes machine translation, automatic speech recognition, text-to-speech systems as well as text classification in multilingual scenarios.

Obviously, LI is a well-established field of application of NLP. However, if one looks at documents that were written in low-density languages or documents that mix several dead languages, adequate models of language detection are rarely found. In any event, ancient languages are becoming more and more central in approach to computational Humanities, historical semantics and studies on language evolution. Thus, we are in need of models of language detection of dead languages.

In this paper, we present such a model. We introduce a supervised classifier that detects amongst others, OHG and OS. To do so, we extend the model of (Waltinger and Mehler, 2009) so that it also accounts for dead languages. For any segment of the logical document structure of a text, our task is to detect the corresponding language in which it was written. This detection at the segment level rather than at the level of whole texts allows us to make explicit the multilingualism of ancient documents starting from the level of words via the level of sentences up to the level of texts. As a result, language-specific preprocessing tools can be used in such a way that they focus on those segments that provide relevant input for them. In this way, our approach is a first step towards building a preprocessor of multilingual ancient texts.

The paper is organized as follows: Section 3 describes the corpus of texts that we have used for our experiments. Section 4 briefly introduces our approach to supervised language detection, which is evaluated in Section 5. Section 6 describes unsupervised language classifier. Finally, a conclusion is given in Section 7.

2. Related Work

As we present a model of n-gram-based language detection, we briefly discuss work in this area.

(Cavnar and Trenkle, 1994) describe a system of *n-gram* based text and language categorization. Basically, they calculate *n-gram* profiles for each target category. Categorization occurs by means of measuring the distances of the profiles of input documents with those of the target categories. Regarding language classification, the accuracy of this system is 99.8%.

The same technique has been applied by (Mansur et al., 2006) for text categorization. In this approach, a corpus of newspaper articles has been used as input to categorization. (Mansur et al., 2006) show that *n-grams* of length 2 and 3 are most efficiently used as features for text

categorization.

(Kanaris and Stamatatos, 2007) used character level *n-grams* to categorize web genres. Their approach is based on *n-grams* of characters of variable length that were combined with information about most frequently used HTML-tags.

Note that the language detection toolkit of Google translator may also be considered as a related work. However, at present, this system does not recognize sentences in OHG. We have tested 10 example sentences. The toolkit categorized only one of these input sentences as modern German; other sentences were categorized as different languages (e.g., Italian, French, English and Danish).

These approaches basically explore *n-grams* as features of language classification. However, they do that for modern languages. In this paper we present an approach that fills the gap of ancient language detection.

3. The Corpus

The corpus used consists of 160 complete texts in six diachronically and diatopically diverging stages of the German language plus the OS glosses, all collected from the TITUS[1] online database. High German is the language variety spoken historically south of a bundle of isogloss lines stretching from Aachen through Düsseldorf, Siegen, Kassel and Halle to Frankfurt (Oder) and has developed into what today constitutes standard German. Low German was spoken historically north of this line but has undergone a decline in native speakers to the point that it is now considered a regional vernacular of and alongside standard German, despite the fact that Low German and High German were once distinct languages. Table 1 shows the historical and geographical varieties of older German.

New discoveries of texts in the various historical forms and varieties of German are being made continually. Due to the steadily increasing number of transmitted texts from throughout the history of the German language, the focus of the TITUS corpus is on the older stages: it comprises the whole OHG corpus (apart from the glosses) as well as the entire OS corpus, including one mixed OHG and OS text. Of the younger language stages only unrepresentative amounts of texts are contained: several dozen Middle High German (MHG) texts, some Middle Low German (MLG) texts, a sample of Early New High German (ENHG) texts and one mixed ENHG and Early New Low German (ENLG) text all of them varying considerably in length, from a few words to several tens of thousands per text.

Language Stage	Period of Time
OHG	ca. 750 – 1050 CE
MHG	ca. 1050 – 1350 CE
ENHG	ca. 1350 – 1650 CE
OS	ca. 800 – 1200 CE
MLG	ca. 1200 – 1600 CE
ENLG	ca. 1600 – 1750 CE

Table 1: Historical and geographical varieties

Among the oldest transmissions are interlinear translations of Latin texts, but also free translations and adaptations as well as mixed German-Latin texts. Translations consist mainly of religious literature, prayers, hymns, but also of ancient authors and scientific writings. These are later on complemented by epic and lyrical poetry (minnesongs), prose literature, sermons and other religious works, specialist books, chronicles, legislative texts and philosophical treatises. The latest texts of the corpus cover a biographical and a historical work, a collection of legal texts for a prince, an experimental re-narration of a parodistic novel as well as the German parts of two bilingual texts, a High German-Old Prussian enchiridion and a mixed High and Low German textbook for learning Russian.

Language Stage	#Texts	#Tokens
OHG	101	437,390
MHG	31	1,776,900
ENHG	6	237,432
OS	17	62,706
MLG	4	133,584
ENLG	1	26,679
Total	160	2,674,691

Table 2: Composition of the corpus

The corpus was generated by entering plain text, either completely by hand or by scanning, performing OCR recognition and correcting it manually. The texts were then indexed and provided with information on languages and subdivisions using the Word-Cruncher

[1] Thesaurus of Indo-European Text and Language Materials – see http://titus.uni-frankfurt.de

[2]software developed by Brigham Young University in Provo, Utah. They were then converted into HTML format and were simultaneously conveyed into several SQL database files, classified by the words' language family, to enable the set-up of an on-line search.

4. Approach

In this section, we describe our language detection approach. We start with describing how we prepared the corpus from TITUS database to get input for our classifier (Section 4.1), introduce our model (Section 4.2) and describe its system design (Section 4.3).

4.1. Corpus Preparation

The training and test corpora that we used in our experiments were extracted from the database dump of TITUS (see Section 3). Each word in this extraction has been annotated with its corresponding language name (example: German), sub-language name (example: Old High German), document number, division number and its position within the underlying HTML corpus files. TITUS only annotates the boundaries of divisions so that any division may contain one or more sentences. For any sub-language (i.e., OHG, OS, MHG, MLG, ENLG and ENHG), we extracted text as reported in Table 2.

4.2. Language Detection Toolkit

Our approach for language detection is based on (Cavnar and Trenkle, 1994) and (Waltinger and Mehler, 2009). As in these studies, for every target category we learn an ordered list of most frequent *n-grams* that occur in descending order. The same is done for any input text so that categorization is done by measuring the distance between *n-gram* profiles of the target categories and the *n-gram* profiles of the test data.

The idea behind this approach is that the more similar two texts are, the more they share features that are equally ordered.

In general, classification is done by using a range of corpus features as are listed in (Waltinger and Mehler, 2009). Predefined information is extracted from the corpus to build sub-models based on those features. Each sub-model consists of a ranked frequency distribution of subset of corpus features. Corresponding *n-gram* information are extracted for $n = 1$ to 5. Each *n-gram* gets its own frequency counter. The normalized frequency distribution of relevant features is calculated according to

$$\widehat{f_{ij}} = \frac{f_{ij}}{max_{ak} \in L(D_j) f_{kj}} \qquad \in (0,1]$$

$\widehat{f_{ij}}$ is the frequency of feature a_i in D_j, divided by the frequency of the most frequent feature a_k in the feature representation $L(D_j)$ of document D_j (see Waltinger and Mehler, 2009). To categorize any document D_m, it is compared to each category C_n using the distance d of the rank r_{mk} of feature a_k in the sub-model of D_m with the corresponding rank of that feature in the representation of C_n:

$$d(D_m, C_n, a_k) = \begin{cases} |r_{mk} - r_{nk}| & a_k \in L(D_m) \wedge a_k \in L(C_n) \\ \max & a_k \notin L(D_m) \vee a_k \notin L(C_n) \end{cases}$$

$d(D_m, C_n, a_k)$ equals *max* if feature a_k does not belong to the representation of D_m or to the one of category C_n. *max* is the maximum that the term $|r_{mk} - r_{nk}|$ can assume.

4.3. System Design

The language detection toolkit (Waltinger and Mehler, 2009) is used to build training models. It creates several *n-gram* models for each language which are used by the same tool for detection. Figure 1 shows the basic system diagram.

To detect the language of a document, the toolkit traverses the document sentence by sentence and detects the language of each sentence. If the document is homogeneous, (i.e., all sentences belong to the same language), then sentence level detection suffices to trigger other tools for further processing (e.g., Parsing, Tagging and Morpho-syntactic analysis) of that document, where language detection is necessary for preprocessing.

In the case that the sentences belong to more than one language (i.e., in the case of a heterogeneous document), the toolkit process the document word by word and detect the language of each token separately. This step is necessary in the case of multilingual documents that contain words from different languages are in single sentences. For example: in a scenario of lemmatization or morphological analysis of a multilingual document, it is necessary to trigger language specific tools to avoid errors. Just one tool needs to be triggered for further processing of a homogeneous document, whereas for a heterogeneous

[2] http://wordcruncher.byu.edu

document the same kind of tool has to be triggered based on the word level.

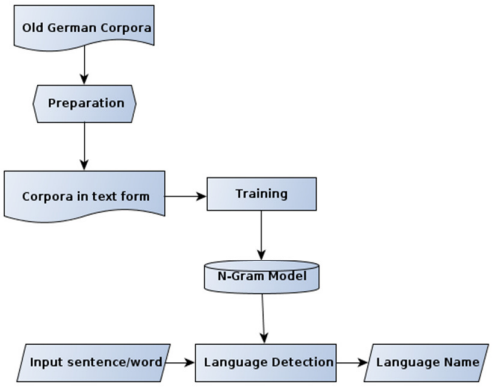

Figure 1: Basic system diagram

Language	Accuracy	F-score
OHG	100%	1
OS	100%	1

Table 3: Sentence level evaluation

5. Evaluation

In order to evaluate the language detection system, we extracted 200 sentences from the OHG corpus and 200 sentences from the OS corpus. These evaluation sets had not been used for training. There are many evaluation metrics used to evaluate NLP tools, we decided to use Accuracy and F-score (Hotho et al., 2005). Table 3 shows the evaluation result of the sentence level language detection, where we obtained 100% accuracy for both test sets. Table 4 shows the evaluation result of the word level language detection. 153 out of 1,259 words in the OHG test set were detected as OS and 33 out of 799 words in the OS test set were classified as OHG. The accuracy of the test set was 79.95% and 91.36% respectively. The evaluation result shows that the OHG test set might contain words from other languages, which is basically true. (Petrova et al., 2009) show that the OHG diachronic corpus contains many Latin words. The evaluation becomes more effective when the result is compared with a gold-standard reference set. We came up with a list of 1,548 words (818 types) where each token is manually annotated with the name of the language to which the word belongs. Of 1,548 words, 564 overlapped with training data. Each word in the gold-standard test set is detected by the toolkit and the result was compared with the reference set. We obtained 91.66% accuracy and an F-score of 95%.

Language	Accuracy	F-score
OHG	79.95%	0.88
OS	91.36%	0.96

Table 4: Word level evaluation

6. Unsupervised Language Classification

In addition to the classifier presented above, we experimented with an unsupervised classifier. The reason was twofold: one the one hand, we wanted to detect the added-value of an unsupervised classifier in comparison to its supervised counterpart. On the other hand, we aimed at extending the number of target languages to be detected. We collected several documents per target language, where each document was represented by a separate feature vector that counts the frequencies of a selected set of lexical features. As target classes we referred to six languages (whose cardinalities are displayed in Table 6): Early New High German (ENHG), Early New Low German (ENLG), Middle High German (MHG), Middle Low German (MLG), Old High German (OHG), and Old Saxon (OS). In order to implement an unsupervised language classifier, we followed the approach described in (Mehler, 2008). That is, we performed a hierarchical agglomerative clustering together with a subsequent partitioning that is informed about the number of target classes. However, other than in (Mehler, 2008), we did not perform a genetic search of the best performing subset of features as in the present case their number is to large. Table 5 shows the classification results. Performing a hierarchical-agglomerative clustering based on the cosine measure as the operative measure of object distance, we get an F-score of around 78%. This is a promising result as it is accompanied by a remarkable high accuracy. However, as seen in Table 4, the target classes perform quite differently: while we fail to separate ENHG and ENLG (certainly due to the small number of respective target documents), we separate MHG, MLG, OHG and OS to a reasonable degree. In this sense, the unsupervised classifier makes expectable even higher F-score supposed that we look for better performing features in conjunction with well-trained supervised

classifiers. At least, the present study provides a baseline that can be referred to in future experiments in this area.

Approach	Object Distance	F-Score	Accuracy
hierarchical/complete	cosine	0.78098	0.91134
hierarchical/weighted	cosine	0.69325	0.86934
hierarchical/average	cosine	0.61763	0.8307
hierarchical/single	cosine	0.56675	0.7926

Table 5: *F-scores* and accuracies of classifying historical language data in a semi semi-supervised environment

Language	#Texts	F-score	Recall	Precision
ENHG	6	0	0	0
ENLG	1	0	0	0
MHG	31	0.895	1	0.810
MLG	4	0.8	0.8	0.8
OHG	101	0.762	0.615	1
OS	17	0.889	0.889	0.889

Table 6: F-scores, recalls, and precisions differentiated by the target classes

7. Conclusion

Language detection plays an important role in processing multilingual documents. This is true especially for ancient documents that, due to their genealogy, mix different ancient languages. Here, documents need to be annotated in such a way that preprocessors can activate language specific routines on a segment by segment basis. In this paper, we presented an extended version of the language detection toolkit that allows us decide when to activate language specific analyses. Notwithstanding the low density of training material that is available for these languages, our classification results are very promising.

At this point one may object that corpora of ancient texts are essentially so small that language detection can be done by hand. Actually, this objection is wrong if one considers corpora like the Patrologia Latina (Jordan, 1995), which mixes classical Latin with medieval Latin as well as with French and other Romance languages that are used in commentaries. From the size of this corpus alone (more than 120 million tokens), it is evident that a reliable means of automatizing segment-based language detection needs to be a viable option. We also described an unsupervised language detector that is evaluated simultaneously by means of OHG, OS, MHG, MLG, ENLG and ENHG. Although this unsupervised classifier does not outperform its supervised counterpart, it shows that language detection in text streams of ancient languages comes into reach.

8. Acknowledgements

We would like to thank Ulli Waltinger, Armin Hoenen, Andy Lücking and Timothy Price for fruitful suggestions and comments. We also acknowledge funding by the LOEWE Digital-Humanities project in the Goethe-Universität Frankfurt.

9. References

William B. Cavnar and John M. Trenkle. 1994. Ngram-based text categorization. In In Proceedings of SDAIR-94, 3rd Annual Symposium on Document Analysis and Information Retrieval, pages 161–175.

Andreas Hotho, Andreas Nürnberger, and Gerhard Paaß. 2005. A Brief Survey of Text Mining. Journal for Language Technology and Computational Linguistics (JLCL), 20(1):19–62.

Mark D. Jordan, editor. 1995. Patrologia Latina database. Chadwyck-Healey, Cambridge.

I. Kanaris and E. Stamatatos. 2007. Webpage genre identification using variable-length character n-grams. In Proc. of the 19th IEEE Int. Conf. on Tools with Artificial Intelligence (ICTAI'07), Washington, DC, USA. IEEE Computer Society.

Monirul Mansur, Naushad UzZaman, and Mumit Khan. 2006. Analysis of n-gram based text categorization for Bangla in a newspaper corpus. In Proceedings of the 9th International Conference on Computer and Information Technology (ICCIT 2006).

Alexander Mehler. 2008. Structural similarities of complex networks: A computational model by example of wiki graphs. Applied Artificial Intelligence, 22(7&8):619–683.

Svetlana Petrova, Michael Solf, Julia Ritz, Christian Chiarcos, and Amir Zeldes. 2009. Building and using a richly annotated interlinear diachronic corpus: The

case of old high german tatian. Journal of Traitement automatique des langues (TAL), 50(2):47–71.

Ulli Waltinger and Alexander Mehler. 2009. The feature difference coefficient: Classification by means of feature distributions. In Proceedings of the Conference on Text Mining Services (TMS 2009), Leipziger Beiträge zur Informatik: Band XIV, pages 159–168. Leipzig University, Leipzig.