Lexical and Semantic Features for Cross-lingual Text Reuse Classification: an Experiment in English and Latin Paraphrases

Maria Moritz[1], David Steding[2]
[1]Institute of Computer Science, University of Göttingen, Germany
[2]Computer Science Faculty, University of Bremen, Germany
maria.moritz@stud.uni-goettingen.de, dsteding@uni-bremen.de

Abstract

Analyzing historical languages, such as Ancient Greek and Latin, is challenging. Such languages are often under-resourced and lack primary material for certain time periods. This prevents applying advanced natural-language processing (NLP) techniques and requires resorting to basic NLP not relying on machine learning. An important analysis is the discovery and classification of paraphrastic text reuse in historical languages. This reuse is often paraphrastic and challenges basic NLP techniques. Our goal is to improve the applicability of advanced NLP techniques on historical text reuse. We present an experiment of cross-applying classifiers—that we trained for paraphrase recognition on modern English text corpora—on historical texts. We analyze the impact of four different lexical and semantic features, on the resulting reuse-detection accuracy. We find out that—against initial conjecture—word embeddings can help to drastically improve accuracy if lexical features (such as the overlap of similar words) fail.

Keywords: cross-lingual classification, collocations, semantic features, word vectors

1. Introduction

Linguistic analyses, such as paraphrastic text-reuse discovery and classification, typically require advanced natural-language processing (NLP) techniques relying on machine learning. Yet, under-resourced historical languages, such as Ancient Greek and Latin, often lack enough primary material for certain time periods to properly train machine-learning models (a.k.a., classifiers). Consequently, only basic NLP techniques (e.g., checking similarity thresholds over string- and n-gram-shingles), which are independent from an advanced global knowledge or training experience, are applicable for analyzing historical corpora (Büchler et al., 2010). To improve the applicability of advanced NLP techniques, we need to experiment and systematically study the performance of such techniques on historical texts. Specifically, we must understand if—and in what way—ancient languages behave differently than contemporary languages when they are transferred and reused, especially when reused paraphrastically (where the reuse is not a literal copy of the source).

In this work, we focus on an important linguistic analysis—the detection and classification of paraphrastic text reuse in historical texts. We study a range of different machine-learning classifiers trained on modern and applied to ancient text-reuse excerpts. We analyze how the trained classifier models behave different or similar to each other for detecting paraphrastic text reuse.

We use two modern English text corpora—one containing about 2700 original-and-reuse pairs, the other containing 2600 compressed news articles (the so called "banner") together with their headlines—and a data set of Latin Bible reuse with around 1100 pairs of original and reused Bible verse.

We test whether three classifiers (K-Nearest Neighbor, Decision Tree, Support Vector Machine) relying on a handful of lexical features (e.g., no. of repeating words) and semantic features (e.g., word vector-based features) can correctly classify the Latin text excerpts as reuse when trained on the modern English data. By identifying features that positively affect the classification, we show their usefulness for cross-lingual reuse detection. The result of this study will help us to understand if cross-lingual application from modern to ancient is worth pursuing.

2. Related Work

Most research focuses on modern corpora. For example, Islam and Inkpen (2008) measure the semantic similarity of texts using corpus-based semantic word similarity and a modified version of the longest common substring algorithm. In contrast, the automated detection of historical text reuse is not thoroughly investigated yet. Büchler et al. (2013; 2010) combine basic NLP techniques for detecting reuse with overlapping features for historical texts using a fingerprinting approach (selecting n-grams from a pre-segmentized corpus).

Historical Text Reuse is studied by Lee (2007) who investigates reuse among the Gospels of the Bible's New Testament, aimed at aligning similar sentences. Similar to techniques used in the field of query expansion and retrieval, they develop so-called alternation patterns using the cosine similarity measure (a source-verse proximity measure) and the source-verse order. The field of paraphrastic reuse detection in historical texts is even more sparse. Bamman (2011) uses word-sense disambiguation. Utilizing a bilingual sense inventory, up to 72% of the word senses are classified correctly.

Cross-lingual Training was performed before. Rigutini et al. (2005) propose an algorithm based on expectation maximization. They train a classifier using a predefined category set and a labeled training data set for one language, followed by training a classifier for a different language on unlabeled documents, but using a translation of the label set. Wang et al. (2008) use labeled data from a related domain as auxiliary information to classify Wikipedia data in

a target domain. Showing the latent semantic relationship between two domains by co-clustering, they can propagate inter-domain labels, which capture common words and semantic concepts. Li et al. (2012) propose Topic Correlation Analysis, which enables the grouping of shared and domain-specific latent features. By inferring correlations between both groups, a mapping to domain-specific topics from different domains is established. The newly derived topics then open a shared feature space. Pan et al. (2008) use dimension reduction to apply transfer learning in a target domain with different feature distributions. They select a low-dimensional feature space, which minimizes the distribution distance in different domains. Via projection in related domains, standard learning is applied.

Cross-lingual training with Word Embeddings: Upadhyay et al. (2016) perform a comparative study by investigating four techniques for inducing cross-lingual embeddings on four different languages. The tasks described range from mono-lingual to cross-lingual similarity evaluation, and from rather "word-centered" semantic to more syntactic cases, where each requires a different degree of supervision. Upadhyay et al. show that models working on expensive cross-lingual knowledge often perform best.

Our work is motivated by the lack of studies on machine-learning approaches for historical text-reuse and the existence of promising results of cross-lingual training in the literature.

3. Study Design

We are interested to learn if we can cross-apply models for paraphrastic reuse detection from English paraphrases to Latin text reuse. Therefore, we identify text features that are language independent. We prepare a corpus with specific characteristics. Some of these characteristics (i.e., length reuse compared to original text, paraphrasic reuse) are similar to a Latin reuse corpus—our test set—and, hence, enables us with a comparable data base. Below, we define our research questions to formulate our work:

Overarchingly we ask: Is it possible to cross-apply classification models for non-literal reuse between modern and ancient-language text? We split this question into two more specific question that we address in this work:

RQ1: *What features support such a cross-lingual reuse classification?*

RQ2: *What characteristics should a source training text have to enable classification of the target language?*

3.1. Methodology

We first define three features (explained shortly) that i) are language independent, and ii) that we can directly infer from the text. We then train and test three classifiers on modern English data to create a baseline for the accuracy that could potentially be achieved when cross-applying each classifier. We finally cross-apply the generated models of English reuse to our Latin data set. In a second experiment (see 4.2.) we introduce a new feature based on word vectors and compare it against the results from the former experiment. This helps us to obtain a comprehensive view on current techniques and how they support our research.

3.2. Data sets

i) The Microsoft Research Paraphrase Corpus (henceforth MSRP) (Dolan and Brockett, 2005) consists of 5801 paraphrastic text-reuse pairs of modern English having a similar length. Out of these, about two thirds are considered semantically equivalent based on manual judgment. Positive and negative training examples are provided in the form of a training set and a test set.

ii) About 2600 instances of Antonio Gulli's English news articles corpus (henceforth Gulli's) collected in 2004 and 2005.[1] It contains XML-formatted data where the headline of a news article is associated with its banner—a very short summary (one sentence) of the article.

iii) Extracts from a total of twelve works and two work collections from the medieval Latin writer Bernard of Clairvaux (henceforth Bernard) who lived in the 12th century and reused text from the Bible. The latter was manually extracted by the Biblindex team (Mellerin, 2016) into a data set of over 1,100 reuse instances in alphabetical order. Every instance relates to a Bible verse. Typically, the reuse is about half as long as the verse. The Bible edition used to obtain the verses is the Biblia Sacra Juxta Vulgatam Versionem (Weber R., 1969 1994 2007). The works from Bernard where the texts are extracted from were published between 1957 and 2010 in the Sources Chrétiennes edition.

We tokenize all three data sets with the inbuilt tokenizer of NLTK (Bird and Loper, 2004) and lemmatize with TreeTagger's (Schmid, 1999) corresponding models for Latin and English. For Gulli's, we extract the text inside "<title></title>" and "<description></description >." We discard tokens from the title if they contain an opening or closing bracket, which often introduces the publisher name. In the description, we replace backslashes by white-spaces and discard dashes, because they indicate a topic-unrelated prefix (e.g., publisher place or newspaper name). Only if neither title nor banner exceed nine tokens and do not contain any character beyond the English alphabet, the exclamation mark, comma, full stop, and question mark, we use them for our experiment. Henceforth, we refer to a text snippet as "text1" and to its reuse as "text2."

3.3. Classification Procedure

Classifiers. We train a variety of well-known classifiers: a K-Nearest Neighbors (KNN), a Support Vector Machine (SVM), and a Decision Tree (DT). These are established approaches and proved successful for many text classification problems. The DT classifier is our own implementation using a maximum information gain metric to decide early on which feature the data has to be split. Our implementation uses discrete feature values only, hence our non-discrete features have to be discretized before they can be handed to the DT classifer. The SVM and KNN classifiers stem from the *sklearn* (Pedregosa et al., 2011) package.

Features. The classifiers are fed with feature values for both, negative and positive training examples. These feature values are calculated on positive reuse couples and negative reuse couples (i.e., two text excepts that are no reuse

[1] https://www.di.unipi.it/~gulli/AG_corpus_of_news_articles.html

of each other). We use three features that can be quickly calculated and directly derived from the text—without requiring any extra resources or annotations. These are: the no. of words that text1 and text2 have in common, similar words that both texts have in common, as well as collocations that both texts have in common. The latter is defined as follows: How many words of text2 are collocations of words from text1, where collocations are considered within a three-word windows and collocations are only considered when they appear at least twice in the union of text1 and text2. Following, these features are listed and formally defined:

1. # words in common relative
2. # words in common .8 similarity relative
3. # 3-window collocations relative

Feature one $f1$ is defined as the relative amount of words that both text excerpts have in common:

$$f1(text1, text2) = \frac{|text1 \cap text2|}{min(|text1|, |text2|)} \quad (1)$$

Feature two $f2$ is defined as the relative amount of similar words that both text excerpts have in common:

$$f2(text1, text2) = \frac{|text2_{sim2\_text1}|}{|text2|} \quad (2)$$

Where $|text2_{sim2\_text1}|$ is the number of words form $text2$ that match with at least one word of $text1$ so that:

$$0.2 \geq \frac{edit(a, b)}{\frac{|a|+|b|}{2}} \quad (3)$$

Where $edit$ is the common edit distance by Levenshtein (Levenshtein, 1965).
Feature three $f3$ is defined as the relative amount of words from $text2$ that are collocations of the words from $text1$:

$$f3(text1, text2) = \frac{|text2_{sim3\_text1}|}{|text2|} \quad (4)$$

Where $|text2_{sim3\_text1}|$ determines the number of words from $text2$ that are collocations of any word from $text1$. Where collocations are calculated within $text1$ or $text2$ for each word of the corresponding text. Collocations are defined in a window of 3 with a maximum distance of 2 from a given word.
The classifiers calculate their models based on the feature values that are calculated form the examples. Examples with shorter reuse than source text may results in higher feature values, because the probability that all words form a short reuse are found in a longer source increases.

Sampling and Training. We use 10-fold-cross validation to train the classifiers. The ten complementing data parts for the training evaluation are generated randomly for each corpus, but every classifier gets the same input data. We train on 70% of the overall data sets and test on the remaining 30%. Note that to obtain data sets of negative examples (with similar sizes as those of the positive examples) for Gulli's and Bernard, we randomly generate links between any two text pairs.

4. Results

We first show the baseline performance of the classifiers on our corpora. Thereafter, we introduce a new feature, which we add to our feature set. We repeat the experiment afterwards.

4.1. Initial Experiment

We are interested in the classifiers' performance when trained on the modern and applied to the ancient Bernard corpus. Table 1 shows the precision where all implemented features are considered. It shows first how a trained model performs when applied to the test set of the data set it is trained on, and below, how it performs when applied to the Latin data set (Bernard). We see clearly that classifiers trained on Gulli's perform stable when their model is applied to the Latin data set. This can be explained by the fact that—even though a news headline is much shorter than its banner, both text excerpts do strongly overlap in their content words.
The models trained on MSRP show a comparable poor performance on the negative data when applied to the 30% test set of MSRP, which is partly caused by the comparably high ratio of positive examples compared to negative examples. Another reason is the characteristic of MSRP, which—serving as a benchmark for semantic equivalence—contains examples of close similarity, and its negative samples are surfacially very similar to the positive examples. (Finch et al., 2005) discuss some ambiguous characteristics of the MSRP corpus and give related examples. Thus, when two sentences are annotated for semantic equivalence that does not necessarily require them to be parahrasaes of each other. We take (Finch et al., 2005)'s examples to demonstrate this.

Example 1 (semantically equivalent following MSRP's annotators):

1. Amrozi accused his brother, whom he called "the witness", of deliberately distorting his evidence.

2. Referring to him as only "the witness", Amrozi accused his brother of deliberately distorting his evidence.

Example 2 (not semantically equivalent following MSRP's annotators):

1. Yucaipa owned Dominick's before selling the chain to Safeway in 1998 for $2.5 billion.

2. Yucaipa bought Dominick's in 1995 for $693 million and sold it to Safeway for $1.8 billion in 1998.

Example 3 (semantically equivalent following MSRP's annotators):

1. The stock rose $2.11, or about 11 percent, to close Friday at $21.51 on the New York Stock Exchange.

2. PG&E Corp. shares jumped $1.63 or 8 percent to $21.03 on the New York Stock Exchange on Friday.

train	test	precision	recall	fscore	precision	recall	fscore	accuracy		accuracy_new
KNN		**positive**			**negative**					
MSRP	MSRP	.74	.68	.71	.42	.50	.46	**.62**		**.65**
MSRP	Bernard	.62	.45	.53	.58	.73	.65	**.60**		**.68**
Gulli's	Gulli's	.83	.81	.82	.83	.85	.84	**.83**		**.85**
Gulli's	Bernard	.82	.82	.82	.83	.83	.83	**.82**		**.84**
DT		**positive**			**negative**					
MSRP	MSRP	.72	.86	.78	.50	.29	.37	**.68**		**.68**
MSRP	Bernard	.49	1.0	.66	-	0.0	-	**.49**		**.65**
Gulli's	Gulli's	.88	.82	.85	.84	.90	.87	**.86**		**.85**
Gulli's	Bernard	.86	.34	.48	.59	.94	.73	**.64**		**.78**
SVM		**positive**			**negative**					
MSRP	MSRP	.72	.94	.81	.62	.21	.31	**.71**		**.71**
MSRP	Bernard	.96	.51	.67	.67	.98	.80	**.75**		**.76**
Gulli's	Gulli's	.87	.84	.86	.86	.88	.87	**.86**		**.88**
Gulli's	Bernard	.87	.83	.86	.84	.90	.87	**.86**		**.87**

Table 1: Performance of the classifiers K-Nearest Neighbors, Decision Tree and Support Vector Machine showing precision, recall, fscore, accuracy, and accuracy of the new feature accuracy_new

Especially using the SVM classifier, precision is high for the positive test sample when the MSRP model is applied to Bernard, but recall is low. This again can be explained by the fact that some similar texts which are marked as negative text reuse in MSRP would be marked as positive text reuse examples in the Bernard data set.

The DT classifier performs particularly bad especially for the cross-application task. This is intuitive considering the comparably primitive model behind these types of classifiers. During discretizing our DT implementation maps the feature values to 50 different intervals. When trained on MSRP and Gulli's the DT classifier prefers a feature which's values enable the soonest and highest information gain. From 1 we can see that a feature that is significant for negative reuse, however, is not a good choice for the negative reuse in Bernard's data (84% vs. 59% precision on the negative data set). The SVM classifier treats feature values better in that respect that it creates a hyper plain equally based on all features.

In the following, we add a new feature to our experiment: We calculate a normalized context vector for each side of each reuse pair and add the angle between the two text excerpts as an extra feature.

4.2. Adding the Angle between the Context Vector as Feature

We now show how a new feature affects the accuracy in our experiment. We use normalized word vectors that represent context information for each word of a text excerpt. The angle between two vectors representing one text excerpt each serves as a new feature for reuse classification. To conceive word vectors for the English data, we use the pre-trained word vectors from GloVe (c.f., Pennington et al. (2014)), which are calculated on a dump of the English Wikipedia in 2014 and the Gigaword5 (Robert Parker, 2011) corpus. To conceive word vectors for our Latin data set, we pretrain vectors on the corpus from the Latin Library (Johnson, 2014) of the CLTK (Johnson et al., 2014 2016). We determine the new feature from the positive and negative training set for Gulli's, MSRP and Bernard. This new feature $f4$ is defined as the cosine of the angle between the averaged word vectors of $text1$ and $text2$. Those averaged vectors are defined as vec_{text1} and vec_{text2}:

$$vec_{text1} = \frac{\sum_{i=0}^{|text1|} v_{w_i}}{|text1|} \tag{5}$$

$$vec_{text2} = \frac{\sum_{j=0}^{|text2|} v_{w_j}}{|text2|} \tag{6}$$

Where v_w is the word embedding vector of a running word in $text1$ and $text2$ respectively.

The last column of Table 1 shows the results for this experiment. Nearly every classifier model on every data set shows an increase in accuracy. Especially for classifiers with a less complex model, the new feature causes a huge accuracy gain applied to both, the data it is trained on and the new target data set of the cross-lingual application task.

4.3. Discussion

RQ1: We learn from the experiments that lexical features can serve well for classification in a cross-lingual task, and that semantic characteristics, such as those that can be derived from word embeddings, support the identification of paraphrastic reuse. One should, however, be aware that—for our features to be suitable for the task—our training and testing data share common characteristics, i.e., texts behave similar in their surface and semantic features.

RQ2: When the training text is composed similarly to the testing text, a well-working cross-lingual classification can be achieved. If semantic equivalence is defined on a more granular level—as it its the case in the MSRP training data—recall and precision scores tend to excel each other widely. Further, MSRP corpus data differ from the other two data sets in the length of the text excerpts of a reuse pair. In the MSRP reuse data, the lengths are largely equal, as opposed to the other data sets. The reuse (text2) in Gulli's and Bernard corpus is about half as long as text1 and often contains words that are repeated from the former text

or slightly modified. This is another characteristic that explains why especially the more advanced classifier methods work better when trained on Gulli's and applied to Bernard. Summarizing, it can be useful to consider the advantage of available, modern text corpora for a learning task on historical text if the properties for which a classifier shall be trained remain comparable.

5. Conclusion

We presented a feasibility study of classification for cross-lingual training. Our study shows that the approach under a simple feature selection (based on shared similar words and collocations) can perform well with an accuracy of up to 86%, and even higher for models trained additionally on a new feature that is determined by the angle between the normalized word vectors of a reuse pair. We found that especially for less advanced classifiers, this new feature drastically improves the accuracy. We showed that it is valuable to consider using modern resources in a classification task for historical languages when the investigated data sets share similar features, such as structural characteristics.

Acknowledgments

Our work is funded by the German Federal Ministry of Education and Research (grant 01UG1509).

Bibliographical References

Bamman, D. and Crane, G. (2011). Measuring historical word sense variation. In *Proceedings of the 11th ACM/IEEE-CS Joint Conference on Digital libraries (JCDL 2011)*, pages 1–10. ACM Digital Library.

Bird, S. and Loper, E. (2004). Nltk: The natural language toolkit. In *Proceedings of the ACL 2004 on Interactive Poster and Demonstration Sessions*, ACLdemo '04. Association for Computational Linguistics.

Büchler, M., Geßner, A., Eckart, T., and Heyer, G. (2010). Unsupervised detection and visualisation of textual reuse on ancient greek texts. *Journal of the Chicago Colloquium on Digital Humanities and Computer Science*, 1(2).

Büchler, M. (2013). *Informationstechnische Aspekte des Historical Text Re-use (English: Computational Aspects of Historical Text Re-use*. Ph.D. thesis, Leipzig University, Germany.

Dolan, B. and Brockett, C. (2005). Automatically constructing a corpus of sentential paraphrases. In *Third International Workshop on Paraphrasing (IWP2005)*. Asia Federation of Natural Language Processing.

Finch, A., Hwang, Y.-S., and Sumita, E. (2005). Using machine translation evaluation techniques to determine sentence-level semantic equivalence. In *Proceedings of the Third International Workshop on Paraphrasing*, pages 17–24. Association for Computational Linguistics.

Islam, A. and Inkpen, D. (2008). Semantic text similarity using corpus-based word similarity and string similarity. *ACM Trans. Knowl. Discov. Data*, 2(2):10:1–10:25.

Johnson, K. P., Burns, P. J., Hollis, L., Pozzi, M., Shilo, A., Margheim, S., Badger, G., and Bell, E. (2014–2016). Cltk: The classical language toolkit. `https://github.com/cltk/cltk`. DOI 10.5281/zenodo.44555 v0.1.32.

Johnson, K. P. (2014). Cltk latin library. `https://github.com/cltk/latin_text_latin_library`. Acsessed Aug. 2017.

Lee, J. (2007). A computational model of text reuse in ancient literary texts. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics, Prague, Czech Republic*, pages 472–479. Association for Computational Linguistics.

Levenshtein, V. I. (1965). Binary codes capable of correcting deletions, insertions, and reversals. *Doklady Akademii Nauk SSSR*, 163(4):845–848. (1966) Russisch, Englische Übersetzung. In: Soviet Physics Doklady Vol. 10, No. 8: 707–710.

Li, L., Jin, X., and Long, M. (2012). Topic correlation analysis for cross-domain text classification.

Mellerin, L. (2016). Biblindex. `http://www.biblindex.mom.fr/`.

Pan, S. J., Kwok, J. T., and Yang, Q. (2008). Transfer learning via dimensionality reduction. In *Proceedings of the 23rd National Conference on Artificial Intelligence - Volume 2*, AAAI'08, pages 677–682. AAAI Press.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

Pennington, J., Socher, R., and Manning, C. D. (2014). Glove: Global vectors for word representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543.

Rigutini, L., Maggini, M., and Liu, B. (2005). An em based training algorithm for cross-language text categorization. In *Proceedings of the 2005 IEEE/WIC/ACM International Conference on Web Intelligence*, WI '05, pages 529–535, Washington, DC, USA. IEEE Computer Society.

Robert Parker, David Graff, J. K. K. C. K. M. (2011). English gigaword fifth edition. ldc2011t07. dvd.

Schmid, H. (1999). Improvements in part-of-speech tagging with an application to german. In *Natural language processing using very large corpora*, pages 13–25. Springer.

Upadhyay, S., Faruqui, M., Dyer, C., and Roth, D. (2016). Cross-lingual models of word embeddings: An empirical comparison. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1661–1670, Berlin, Germany, August. Association for Computational Linguistics.

Wang, P., Domeniconi, C., and Hu, J. (2008). Using wikipedia for co-clustering based cross-domain text classification. In *2008 Eighth IEEE International Conference on Data Mining*, pages 1085–1090, Dec.

Gribomont J. Weber R., Fischer B., editor. (1969, 1994, 2007). *Biblia sacra juxta vulgatam versionem*. Deutsche Bibelgesellschaft.