Using machine learning for language and structure annotation in an 18th century dictionary

Petra Bago, Nikola Ljubešić

Department of Information and Communication Sciences, Faculty of Humanities and Social Sciences,
University of Zagreb, Ivana Lučića 3, HR-10000
{pbago, nljubesi}@ffzg.hr

Abstract

The accessibility of digitized historical texts is increasing, which, consequently, has resulted in a growing interest in applying machine learning methods to enrich this type of content. The need for applying machine learning is even greater than in modern texts given the high level of inconsistency in historical texts even within the same document. In this paper we investigate the application of a supervised structural machine learning method on language and structure annotation of 18th century dictionary entries. Our research is conducted on the first volume of a trilingual dictionary 'Dizionario italiano–latino–illirico' (Italian–Latin–Croatian Dictionary) compiled by Ardellio della Bella and printed in Dubrovnik in 1785. We assume that by using this method, we can significantly reduce time for manual annotation and simplify the process for the annotators. We reach accuracy of approximately 98% for language annotation and around 96% for structure annotation. A final experiment on the time gain obtained by pre-annotating the data shows that only correcting the generated labels is roughly five times faster than full manual annotation.

Keywords: historical dictionaries; language annotation; structure annotation; supervised machine learning

1. Introduction

The accessibility of digitized historical texts is increasing, which, consequently, has resulted in a growing interest in applying natural language processing and machine learning methods for processing and enriching this type of content. Using these methods, some of the problems approached are mapping historical spelling variants to modern equivalents (Archer et al., 2015), identifying and extracting mentions of times present in historical resources (Foley and Allan, 2015), improving verb phrase extraction (Pettersson and Nivre, 2015) or developing a web-based application for editing manuscripts (Raaf, 2015). The need for applying machine learning is even greater than in modern texts given the high level of inconsistency in historical texts even within the same document (Piotrowski, 2012). In this paper we investigate the application of a supervised structural machine learning method on language and structure annotation of 18th century dictionary entries.

Our research is conducted on the first volume of a second edition of a trilingual dictionary 'Dizionario italiano–latino–illirico' (Italian–Latin–Croatian Dictionary) compiled by Ardellio della Bella and printed in Dubrovnik in 1785 (della Bella, 1785). The dictionary was intended for Italian Jesuit

missionaries to help them spread the faith in a national language i.e. Croatian language, but also other Slavic languages. For this reason a Croatian grammar can be found inside the dictionary preamble. The dictionary consists of 899 pages and two parts. The first part is a preamble written in Italian on 54 pages. The second part is the dictionary, containing around 19,000 headwords. The dictionary is printed in two volumes: the first volume contains the preamble and the dictionary part from letters A to H, while the second volume contains the dictionary part from letters I to Z. For the first time in Croatian lexicography, della Bella's dictionary contains examples of uses of headwords in various literary works and oral literature.

In the paper we approach two separate annotation, i.e. enrichment problems, using the state-of-the-art supervised machine learning algorithm for labeling sequences – conditional random fields (CRFs). We first approach the problem of annotating each token with its corresponding language label which is a ternary classification task given the three languages that are represented in the dictionary. Having the language label at our disposal, we then approach the problem of annotating each token with the corresponding structure label. The structure level has 19 different labels based on the Text Encoding Initiative (TEI) encoding scheme for dictionaries (TEI Consortium, 2014).

We approach both annotation problems by determining first whether the original or lowercased tokens produce better results, defining that feature as our basic feature. Next, we measure the performance of adding several other features to the basic one like whether the token is originally lowercased, the frequency of a specific token trigraph, the previous and the next token, whether the previous and the next token is lowercased, etc. Finally, we combine all features that show increase over the results obtained with the basic feature.

2. Related work

Historical texts are written in historical languages that are natural languages, just like the modern languages found in modern texts. Consequently, both historical and modern languages share the same challenges when it comes to natural language processing (NLP) of these types of texts, such as homonymy and polysemy. However, historical texts have further characteristics that pose additional challenges to NLP tools trained on modern texts: the lack of a standard variant, the lack of a standard orthography, the lack of electronically available texts, and the lack of existing NLP resources and tools for this type of text (Piotrowski, 2012).

Nevertheless, machine learning methods have been applied to historical texts approaching various problems. (Buchler et al., 2014) address the issue of complication to historical text-reuse detection, because of its longer time span, thereby having a larger set of morphological, linguistic, syntactic, semantic and copying variations. (Mitankin et al., 2014) present an approach to historical text normalisation, achieving 81.79% normalisation accuracy of 17^{th} century English texts in a fully unsupervised setup. Furthermore, (Kettunen et al., 2014) experimented with methods based on corpus statistics, language technology and machine learning in order to find ways to automate

the process of analyzing and improving the quality of a historical news collection. (Horton et al., 2009) trained a supervised machine learning algorithm to determine classes of knowledge of the articles in the the Encyclopédie of Denis Diderot and Jean le Rond d'Alembert. (Hendrickx et al., 2011) presented an approach to automatic text segmentation of historical letters in Portuguesein formal/informal parts using a statistical n-gram based technique, achieving the result of 86% micro-averaged F-score. Additionally, they presented an approach to semantic labeling of the formal parts of the letters using supervised machine learning, achieving the result of 66.3% micro-averaged F-score.

In the paper we approach two separate annotation, i.e. enrichment problems, using the state-of-the-art supervised machine learning algorithm for labeling sequences – conditional random fields (CRFs). Conditional random fields (CRFs) are a statistical method for structure prediction, that has the ability to predict labels based on several dependent variables. The models are applied to image labeling, e.g. (He et al., 2004), (Kumar and Hebert, 2003), various bioinformatics problems, e.g. (Sato and Sakakibara, 2005), (Liu et al., 2005), speech processing, e.g. (Yu et al., 2010), (Boonsuk et al., 2014), and, the most relevant to the paper, textual data, e.g. (Sha and Pereira, 2003), (McCallum and Li, 2003), (Taskar et al., 2002), (Pinto et al., 2003), (Shen et al., 2007), (Choi et al., 2005).

In digital humanities, annotating the structure of a digitized text is a manual task, that is time consuming and tedious, thereby paving the way for an annotator to introduce inconsistencies. By automating the process of annotation, we consider it to reduce cognitive load in annotators and time spent on the task. As far as we know, the present work is the first to apply conditional random fields on a historical text. Additionally, we have not come across an application of CRFs on language labeling on textual data, nor on structure labeling based on a *de facto* standard for encoding textual resources in digital form.

3. Dataset

Our research is conducted on the first volume of the second edition of a trilingual dictionary 'Dizionario italiano–latino–illirico' (Italian–Latin–Croatian Dictionary) compiled by Ardellio della Bella and printed in Dubrovnik in 1785 (della Bella, 1785). The digitization process of the printed 18^{th} dictionary was conducted as part of the project 'Croatian dictionary heritage and Croatian European identity' and was not the scope of this research. However, we will briefly describe the digitization process in order to better describe the data used in this research. The dictionary was photographed and the images were processed with an optical character recognition software. Since the software produced many errors detecting characters, the text was manually compared and checked to corresponding pictures by undergraduate students. Furthermore, during the manual inspection, markup was added for distinct section breaks such as line breaks, new paragraphs, column breaks, and page breaks. Additional markup was manually inserted to encode the beginning and the end of the Latin parts of the entry, and the beginning and the end of the citations from works used as a corpus for dictionary compilation by della Bella. The manual part of the digitization process is the

most tedious and time-consuming. Aforementioned text is stored in a proprietary word processor that we converted into a plain text file for further processing.

The first volume of the trilingual dictionary consists of 7,972 dictionary entries starting with the letter A and ending with the letter H (`Huquang`), comprising 403,128 tokens that were automatically segmented. The average length of the dictionary entry is 50.57 tokens.

Following the tokenization phase, for our training sample we randomly selected 101 dictionary entries for manual annotation. The training sample comprises of 8,340 tokens (2,07%), while the unlabelled set contains 394,788 tokens (97,93%).

Every token out of the selected entries is annotated on two levels: the language level and the structure level. The language level has three distinct labels, while the structure level has 19. Label distributions of both levels are depicted in Tables 1 and 2. Altogether 8,340 labels are manually annotated on each level, that is 16,680 labels in total. The average length of the selected entries is 82.57 token, i.e. 32 tokens more than the average entry of the first volume of the dictionary.

There are three labels of the language level based on three languages that can be found in della Bella's dictionary. The labels used for the language annotation, its explanation and frequency distribution are given in Table 1.

label explanation	frequency
`hr` a token in Croatian	4,395
`it` a token in Italian	2,164
`la` a token in Latin	1,781

Table 1: The labels used for the language annotation, its explanation and frequency distribution

In Table 1 it is interesting to note that more than half (53%) of the tokens are in Croatian language, while Italian is more frequent than Latin (26% vs. 21%). This can be interpreted as the lexicographer's attempt to include all possible words with similar senses in the Croatian language, while for the Latin language there can usually be found only one word sense, probably because of the similarity between Italian and Latin.

There are 19 labels of the structure level that are based on the Text Encoding Initiative module for dictionaries (TEI Consortium, 2014). The labels used for the structure annotation, its explanation and frequency distribution are given in Table 2.

We perform two separate annotation problems: the problem of annotating each token with the corresponding language label and the problem of annotating each token with the corresponding structure label, having at that point the language label at our disposal.

label	explanation	frequency
`abbr`	an abbreviation	55
`adj`	a suffix for an adjective	2
`adjf`	a suffix for a feminine singular adjective	109
`adjn`	a suffix for a neuter singular adjective	118
`bibl`	a source of citation	90
`cb`	a column break when it is not separating a token[1]	7
`citex`	a citation	729
`cittrans`	a translation of the headword or another word within the dictionary entry	3,167
`formlem`	a headword[2]	125
`genpl`	a suffix for a genitive plural noun	1
`gensg`	a suffix for a genitive singular noun	185
`hint`	a token that guides the sense of the headword or another word within the dictionary entry	415
`lb`	a line break when it is within one entry and does not separate a token[3]	506
`pb`	a page break when it is not within one token[4]	6
`pc`	a punctuation character that is not part of an abbreviation	2,329
`pos`	a part of speech (masculine, feminine and neuter gender of a noun, plural if a noun is in that form, adjective, adverb)	198
`ref`	a reference to another entry	35
`v`	a suffix for a verb form, usually first person singular present and first person singular perfect	230
`xr`	a token for a cross-reference phrase	33

Table 2: The labels used for the structure annotation, its explanation and frequency distribution

4. Experimental setup

In our experiment we use state-of-the-art supervised machine learning algorithm for labeling sequences named conditional random fields (CRFs) (Lafferty et al., 2001). CRFs are a statistical method for structure prediction, that has the ability to predict labels based on several dependent variables. These models are successfully applied in different fields, such as text processing, bioinformatics and computer vision (Sutton and McCallum, 2012).

We train and evaluate CRFs with the `CRFsuite tool` (Okazaki, 2007). The tool implements several different state-of-the-art methods of machine learning and we use the passive aggressive training algorithm since it obtained the best results. The software has features like fast training and tagging data, simple data format and the ability to design an arbitrary number of features for each item. Additionally the tool has the ability to compute performance evaluation of the model evaluated on test set (precision, recall and F_1 scores).

We perform two separate annotation problems: the problem of annotating each each token with the corresponding language label and the problem of annotating each token with the corresponding structure label, having at that point the language label at our disposal. Our approach to both problems is similar. Firstly we define potentially interesting sets of features that could obtain better results than the data alone. Next we measure performance of the selected features. Finally, we combine all features that show an increase over the result obtained with the basic feature thereby achieving the best possible result with the defined features. We compute the usual metrics used for model evaluation in the field of natural language processing: precision, recall, F measure and accuracy.

Our experiment is conducted in three phases. The first phase consists of testing the most obvious feature, i.e. does the spelling of the token have an effect on the result: original spelling of the token and lowercased spelling of the token. We expect that one of the forms of spelling will yield a better result. Consequently we will be using the feature that achieved better results as the basic feature in further testing.

In the second phase of the experiment we test the effect of additional features on the results of machine learning. As the basic feature we use the one from the first phase of the experiment. On the language level as additional feature we measure a Boolean variable of whether the original token is lowercased or not. Next we measure the frequency of a specific trigraph. Furthermore we test the effect of `N` tokens before and after the specific token, for `N` ranging from 1 to 3. The final tested measure is a Boolean variable of whether tokens before and after are lowercased or not.

On the structure level as an additional feature we measure a Boolean variable of whether the original token is lowecased or not. Next we test the effect of `N` tokens before and after the specific token, for `N` ranging from one to four. Furthermore, we measure a Boolean variable of whether tokens before and after are lowercased or not. Since the dataset for this phase contains data about the language of the token, we test the effect of that feature on the results.

In the final phase of the experiment, we combine in one experiment all features that show an increase over the result obtained with the basic feature. Thereby we achieve the best possible result with the defined features.

To estimate how accurately our predictive model will perform on an independent dataset, we evaluate each parameter by calculating accuracy via a 10-fold cross-validation.

5. Results

5.1 The language annotation

The language annotation has a set of three labels. The experiment is conducted with the following features:

- `token`: a token in its original form,
- `ltoken`: lowercased token,
- `lcasebool`: a Boolean variable whether a token is lowercased or not,
- `trigraphfreq`: a frequency of a specific trigraph,
- `prevNtoken` and `nextNtoken`: `N` tokens before and after a specific token, for `N` = 1..3,
- `prevNlcasebool` and `nextNlcasebool`: a Boolean variable whether `N` tokens before and after are lowercased.

Below we depict 7 tokens labelled on both the language and the structure level:

```
radici   it hint
.        it pc
V.       it xr
Barbare  it ref
.        it pc
Radicare it ref
.        it pc
```

The feature values for the token `Barbare` of the abovementioned sequence are as follows:

```
token=Barbare
ltoken=barbare
lcasebool=False
trigraphfreq=_ba:1
trigraphfreq=bar:2
trigraphfreq=arb:1
trigraphfreq=rba:1
trigraphfreq=are:1
trigraphfreq=re_:1
prev1token=V.
prev2token=.
prev3token=radici
next1token=.
next2token=radicare
next3token=.
prev1lcasebool=False
prev2lcasebool=True
prev3lcasebool=True
next1lcasebool=True
next2lcasebool=True
next3lcasebool=True
```

The results of the accuracy of the language annotation with specific features are given in Table 3. Since lowercased tokens perform better than the original ones, the remainder of the experiments use the lowercased tokens as the basic feature.

Additionally, the most informative features are token trigraphs and tokens before and after the specific token. Using two tokens before and after a specific token gives slightly better results than using just one or three tokens before and after. This is why in the last parameter we combine the best performing features: lowecased tokens, token trigraphs, a Boolean variable whether a token is lowercased or not, a Boolean variable whether tokens before and after are lowercased, and two tokens before and after a specific token. This selected feature set obtains the best results, i.e. the accuracy of the language annotation of 98.413%.

features	accuracy
token	0.93224
ltoken	**0.94143**
ltoken lcasebool	0.95405
ltoken trigraph	0.97107
ltoken prevNtoken nextNtoken N=1	0.97188
ltoken prevNtoken nextNtoken N=1..2	0.97997
ltoken prevNtoken nextNtoken N=1..3	0.97697
ltoken prevNlcasebool nextNlcasebool N=1	0.94475
ltoken prevNlcasebool nextNlcasebool N=1..2	0.95086
ltoken prevNlcasebool nextNlcasebool N=1..3	0.94142
ltoken lcasebool trigraph prevNtoken nextNtoken prevNlcasebool nextNlcasebool N=1..2	**0.98413**

Table 3: The accuracy of language annotation with various features

Table 4 gives the results of precision, recall and F_1 measure of the final language classifier by category. The classifier obtains the best results for the Latin language for all three measures: a precision (P) score of 0.99815, a recall (R) score of 0.99938, and an F_1 score of 0.99878. Since the Latin part of the dictionary entries is always wrapped in special markup, the results are expected. The classifier accomplishes better precision scores for the Italian language (0.9829) than for Croatian (0.97953). The reason for this could be due to the fact that the beginning of a dictionary entry is always in Italian. Better results of the recall scores are obtained for the Croatian language (0.99067) than for Italian (0.95831), which can be interpreted by the fact that over half (53%) of the tokens are labelled as Croatian, but just over one quarter (26%) as Italian.

lang	Precision	Recall	F_1
hr	0.97953	0.99067	0.98507
it	0.9829	0.95831	0.97045
la	**0.99815**	**0.99938**	**0.99876**

Table 4: The performance of the final language classifier by category

5.2 The structure annotation

The structure annotation has a set of 19 labels. The experiment on the structure level follows the same methodology as for the language level. The experiment is conducted with following features:

- `token`: a token in its original form,
- `ltoken`: lowercased token,
- `lcasebool`: a Boolean variable whether a token is lowercased or not,
- `prevNtoken` and `nextNtoken`: `N` tokens before and after a specific token, for `N` = 1..4,
- `prevNlcasebool` and `nextNlcasebool`: a Boolean variable whether `N` tokens before and after are lowercased.
- `lang`: a language label of the token,
- `suffixN`: a suffix of a specific token of length N=4.

The results of the accuracy of the structure annotation with specific features are given in Table 5. Since tokens in its original form perform better than lowercased tokens, the remainder of the experiment uses the original form of tokens as the basic feature.

features	accuracy
`token`	**0.85993**
`ltoken`	0.85538
`token lcasebool`	0.8934
`token prevNtoken nextNtoken N=1`	0.90388
`token prevNtoken nextNtoken N=1..2`	0.93794
`token prevNtoken nextNtoken N=1..3`	0.94994
`token prevNtoken nextNtoken N=1..4`	0.94219
`token prevNlcasebool nextNlcasebool N=1`	0.87586
`token prevNlcasebool nextNlcasebool N=1..2`	0.87706
`token prevNlcasebool nextNlcasebool N=1..3`	0.88755
`token prevNlcasebool nextNlcasebool N=1..4`	0.89588
`token lang`	0.86555
`token suffixN N=1..4`	0.87192
`token lcasebool lang prevNtoken nextNtoken prevNlcasebool nextNlcasebool suffixN N=1..4`	0.96111
`token lcasebool prevNtoken nextNtoken N=1..3 prevNlcasebool nextNlcasebool suffixN N=1..4`	**0.96372**

Table 5: The accuracy of the structure annotation with various features

Additionally, the most informative feature is four tokens before and after a specific token. However, when we combine the best performing features, the accuracy score increases almost 2% and totals 0.96372. Those features are: tokens in their original form, a Boolean variable whether a token is lowercased or not, four tokens before and after a specific token, a Boolean variable whether four tokens before and after a specific token are lowercased or not, a language label, and a suffix of a specific token of length N=1..4.

Table 6 gives the results of precision, recall and F_1 measure of the final structural classifier by category. The classifier obtains 100% precision for column breaks and line breaks, which is expected since these properties are explicitly tagged in the dictionary corpus. The next best accuracy score is 0.9981 for punctuation characters. Since in the dictionary corpus there is always a space before a punctuation character that is not part of an abbreviation, this result is likewise expected. The worst results obtained by the classifier are for labels that are rare in the manually annotated corpus. There is only one occurrence of the label `genpl`, and the precision score is 0.0. The same result is obtained for the label `adj`, that has only two occurrences. The third worse result (0.6) is obtained for the label `pb`, that has only six occurrences in the manually annotated corpus.

The classifier obtains 100% recall for the label `lb`, while the second best result (0.99762) is for the label `pc`. Both results can be interpreted as with the precision. The label `lb` is explicitly tagged in the dictionary corpus, while there is always a space before punctuation character that is not part of an abbreviation. The classifier obtains the third best result (0.99087) for the label `v` and the reason for this could be the fact that this label refers to the suffixes for verbs that regularly have the same form. The worst results obtained by the classifier are for the labels `genpl` and `adj`, like with the

precision scores, because the labels rarely occur in the manually annotated corpus. The recall score for the label `cb` is surprising and only totals to 0.57143. The column break is explicitly tagged in the dictionary corpus in two ways: it can be a standalone tag, but it can also be found within a token, where it is left as part of that token, and not separately tokenized. The assumption is that the tag within a token generates obstacles for the classifier to obtain higher recall score.

The classifier obtains the top three results for the F_1 measure for the labels `lb` (1.0), `pc` (0.99786) and `v` (0.9819). If observing all three measures combined, the classifier obtains the best result for the label `lb`, while the label `pc` is in top three results for all three measures. On the contrary, the worst results are obtained for the labels `adj`, `genpl` and **pb**, on account of the labels rarely occurring in the manually annotated corpus.

lang	Precision	Recall	F_1
`abbr`	0.85714	0.78261	0.81818
`adj`	0.0	0.0	0.0
`adjf`	0.97196	0.99048	0.98113
`adjn`	0.94595	0.92105	0.93333
`bibl`	0.95122	0.98734	0.96894
`cb`	**1.0**	0.57143	0.72727
`citex`	0.95477	0.95323	0.954
`cittrans`	0.97875	0.95736	0.96794
`formlem`	0.97087	0.9009	0.93458
`genpl`	0.0	0.0	0.0
`gensg`	0.97093	0.98235	0.97661
`hint`	0.76027	0.91484	0.83042
`lb`	**1.0**	**1.0**	**1.0**
`pb`	0.6	0.6	0.6
`pc`	**0.9981**	**0.99762**	**0.99786**
`pos`	0.97297	0.98361	0.97826
`ref`	0.96875	0.91176	0.93939
`v`	0.97309	**0.99087**	**0.9819**
`xr`	0.96875	0.96875	0.96875

Table 6: The performance of the final structural classifier by category

5.3 Testing the time reduction for the manual annotation

Our next experiment answers the question whether correcting automatically assigning language and structure labels reduces the time for the manual annotation, and if confirmed, by how much. The experiment has two 60-minute parts: a manual token annotation and a correction of automatically labelled tokens. Both parts are carried out by an annotator knowledgeable of della Bella's dictionary. The results of the experiment are given in Table 7.

In the first part of this experiment, an annotator manually annotates tokens on the language and structure level for 60 minutes. The starting token is randomly chosen, after which the tokens are annotated in the order of their appearance in the corpus. During this period 741 tokens (i.e. 482 labels) are annotated. In one minute, 12.35 tokens can be manually annotated.

	number of tokens	tokens per minute
manual annotation	741	12.35
correction	3,439	57.32

Table 7: The number of tokens manually annotated and corrected

In the second part of this experiment, an annotator reviews and corrects the automatic labels on the language and structure level for 60 minutes. The starting token is randomly chosen, after which the tokens are reviewed and corrected in the order of their appearance in the corpus. During this period 3,439 tokens (i.e. 6,878 labels) are reviewed and corrected. In one minute, 57.32 tokens can be reviewed and corrected: specifically this method is 4.64 times faster than manual annotation, which we consider clearly more productive than the manual annotation.

Additional value of this experiment is 7,987[5] tokens subsequently annotated or reviewed and corrected that can be incorporated into the training set, thereby possibly obtaining better accuracy scores with the classifier and yet further reducing the time for correction speed.

5.4 The final experiment on the test set

To closely analyse the performance of the classifier, we present the confusion matrices for the language level in Table 8 and the structure level in Table 9. The test set is the result of the experiment in the previous section.

The accuracy of the classifier for the language level is 0.97308. In the confusion matrix given in Table 8 it is evident that the classifier displays fewer problems with predicting the Latin text. Since the Latin part of dictionary entries is always wrapped in special markup, the results are expected. The classifier has the most problems with the Croatian–Italian language pair. The dictionary entries often do not follow the structure of a trilingual dictionary, thus the sequence of the languages appearing is not always Italian–Latin–Croatian. As mentioned before, the Latin part is always wrapped in special markup, which would be a great separator of the Italian from the Croatian. However, if there is a compound within an entry, then the Latin part is frequently absent, which creates a situation where the Croatian part follows the Italian part. Additionally, at the time the dictionary was created, there was no consensus over orthography for the Croatian language, so the lexicographer adopted the Italian practice to record Croatian phonemes. However, this practice introduces inconsistency in orthography within dictionary text. All of this could be the reason why the classifier has the most problems with the Croatian–Italian pair.

The accuracy of the classifier for the structure level is 0.954801. In the confusion matrix given in Table 9 it is evident that the classifier has the most problems with the label `cittrans`, and confuses it most frequently with the labels `hint`, `citex` and `v`. The reason behind this may be the fact that these parts of the entries contain free text. The classifier obtains the best results for the label `xr`,

[5] The second part of this experiment had to be repeated 3 times due to the fatigue of the annotator. This is the reason this number is larger than the sum of the tokens in the first and the second part of this experiment.

	hr	it	la
hr	5,128	19	1
it	194	1,351	1
la	0	0	1,293
accuracy	0.973081257043		

Table 8: The confusion matrix for the language level

which is correctly predicted in all the cases, and for the labels `lb` and `bibl` that are only once incorrectly classified. Since these parts are explicitly tagged in the dictionary corpus, the results are expected. Three labels are not found in the test set: `cb`, `pb` and `adj`.

	cit	trans	ref	lb	bibl	hint	cb	v	pos	pb	pc	abbr	citex	adjn	xr	gensg	form	lem	adj	adjf
cit trans	2,129	0	0	0	18	0	3	0	0	4	0	19	2	0	1	0	0	3		
ref	17	43	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
lb	0	0	506	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0		
bibl	2	0	0	80	0	0	0	0	0	0	0	1	0	0	0	0	0	0		
hint	70	0	0	0	173	0	0	0	0	0	0	9	2	0	2	6	0	0		
cb	5	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0		
v	32	0	0	0	5	0	568	0	0	0	0	0	19	0	1	0	0	2		
pos	2	0	0	1	0	0	0	253	0	1	1	0	0	0	0	0	0	0		
pb	2	0	1	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0		
pc	0	0	0	0	1	0	0	0	0	2,618	0	1	0	0	0	0	0	0		
abbr	2	0	0	0	4	0	0	2	0	2	19	0	0	0	0	0	0	0		
citex	48	0	0	0	2	0	0	0	0	0	0	556	0	0	1	0	0	0		
adjn	2	0	0	0	2	0	1	10	0	0	0	0	124	0	2	0	0	1		
xr	4	0	0	0	0	0	0	0	0	0	4	0	0	44	0	0	0	0		
gensg	5	0	0	0	0	0	1	0	0	0	0	1	1	0	245	0	0	0		
form lem	2	0	0	0	13	0	1	0	0	1	0	0	1	0	0	135	0	0		
adj	1	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0		
adjf	1	0	0	0	0	0	0	0	0	0	0	0	1	0	6	0	0	133		
accuracy	0.954801552523																			

Table 9: The confusion matrix for the structure level

5.5 The learning curve

The learning curve of the used algorithm is given in Figure 1. With regards to the language level having only three labels, while the structure level has 19, we expect the algorithm to generate better results for the former level than for the latter. Moreover, we expect that less data would be necessary for the algorithm to learn most rules for the language level, while the structure level will require more data.

In Figure 1 it is evident that the algorithm discriminates the language better than the structure. Most of the language discrimination is learned after 20% of the data seen, when it reaches accuracy of almost 96%. The final accuracy score is 98.59%, which we regard as an excellent result considering the text is from the 18^{th} century when inconsistency in Croatian orthography was frequent and more than half (53%) of tokens in the manually annotated corpus are Croatian.

The most structure discrimination is learned at about 40% of the data seen, when it reaches accuracy of more than 94%. The final accuracy score is 95.92%, which is a result that exceeds our expectations considering the structure level has 19 labels.

Both curves are still significantly rising. By adding additional data to the training set from the experiment with speed comparison, we could improve accuracy scores for both the language and the structure level, but also decrease the time needed for manual processing of the data.

Finally, we consider the existing algorithm to be beneficial in the language and structure annotation of 18^{th} century dictionary entries, with the accuracy scores being sufficiently high and considerably speeding up the process of the manual processing.

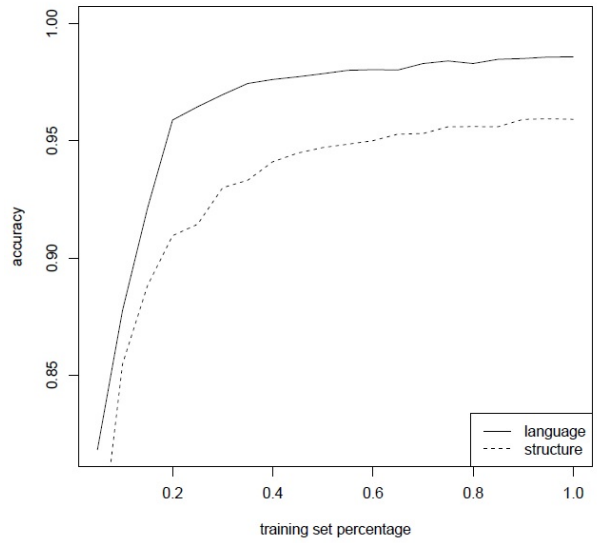

Fig. 1: The learning curve for the language and structure labels

6. Conclusion

In this paper we investigate the application of a supervised structural machine learning method on the language and structure annotation of 18^{th} century dictionary entries. We use state-of-the art supervised machine learning algorithm for labeling sequences – conditional random fields (CRFs). Our research is conducted on the first volume of a trilingual dictionary 'Dizionario italiano–latino–illirico' (Italian–Latin–Croatian Dictionary) compiled by Ardellio della Bella and printed in Dubrovnik in 1785. The training sample comprises of 8,340 tokens out of 403,128 found in the whole of the dictionary corpus. We measure the performance of several features, finally combining all features that show increase over the results obtained with the basic feature for the best result.

We reach the accuracy of approximately 98% for the language annotation with three labels and around 96% for the structure annotation with 19 labels. We compute the usual metrics used for

model evaluation in the field of natural language processing (precision, recall, F measure and accuracy) for both levels of annotation.

In this paper we answered the question whether correcting automatically assigned language and structure labels reduces the time for the manual annotation, and if confirmed, by how much. This experiment confirmed that pre-annotating the data is roughly five times faster than the full manual annotation.

The learning curves for both the language and the structure level are still significantly rising. By adding additional data to the training set from the experiment with speed comparison, we could improve accuracy scores for both language and structure level, but also decrease the time needed for manual processing of data.

7. Acknowledgements

This work was partially supported by the Swiss National Science Foundation grant IZ74Z0_160501.

8. References

Archer, D., Kytö, M., Baron, A. & Rayson, P. (2015). Guidelines for normalising early modern english corpora: Decisions and justifications. *ICAME Journal*, 39(1), pp. 5–24.

Boonsuk, S., Suchato, A., Punyabukkana, P., Wutiwiwatchai, C. & Thatphithakkul, N. (2014). Language recognition using latent dynamic conditional random field model with phonological features. *Mathematical Problems in Engineering*, 2014.

Buchler, M., Franzini, G., Franzini, E. & Moritz, M. (2014). Scaling historical text re-use. In *Big Data (Big Data), 2014 IEEE International Conference on*, pp. 23–31.

Choi, Y., Cardie, C., Riloff, E. & Patwardhan, S. (2005). Identifying sources of opinions with conditional random fields and extraction patterns. In *Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing*, HLT '05, Stroudsburg, PA, USA. Association for Computational Linguistics, pp. 355–362.

della Bella, A. (1785). *Dizionatio italiano-latino-illirico*. Nella Stamperia Privilegiata, prima edizione ragusea edition.

Foley, J. and Allan, J. (2015). Retrieving time from scanned books. In *Advances in Information Retrieval*, Springer, pp. 221–232.

He, X., Zemel, R. S. & Carreira-Perpindn, M. (2004). Multiscale conditional random fields for image labeling. In *Computer vision and pattern recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE computer society conference on*, volume 2, IEEE, pp. 695–702.

Hendrickx, I., Généreux, M. & Marquilhas, R. (2011). Automatic pragmatic text segmentation of historical letters. In *Language Technology for Cultural Heritage*, Springer, pp. 135–153.

Horton, R., Morrissey, R., Olsen, M., Roe, G., Voyer, R., et al. (2009). Mining eighteenth century ontologies: Machine learning and knowledge classification in the encyclopédie.

Kettunen, K., Honkela, T., Lindén, K., Kauppinen, P., Pääkkönen, T., Kervinen, J., et al. (2014). Analyzing and improving the quality of a historical news collection using language technology and statistical machine learning methods. In *IFLA World Library and Information Congress Proceedings 80th IFLA General Conference and Assembly*.

Kumar, S. and Hebert, M. (2003). Discriminative fields for modeling spatial dependencies in natural images. In *In NIPS*. MIT Press.

Lafferty, J. D., McCallum, A. & Pereira, F. C. N. (2001). Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the Eighteenth International Conference on Machine Learning*, ICML '01, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc, pp. 282–289.

Liu, Y., Carbonell, J., Weigele, P. & Gopalakrishnan, V. (2005). Segmentation conditional random fields (scrfs): A new approach for protein fold recognition. In *Research in Computational Molecular Biology*, Springer, pp. 408–422.

McCallum, A. and Li, W. (2003). Early results for named entity recognition with conditional random fields, feature induction and web-enhanced lexicons. In *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003 - Volume 4*, CONLL '03, Stroudsburg, PA, USA. Association for Computational Linguistics, pp. 188–191.

Mitankin, P., Gerdjikov, S. & Mihov, S. (2014). An approach to unsupervised historical text normalisation. In *Proceedings of the First International Conference on Digital Access to Textual Cultural Heritage*, DATeCH '14, New York, NY, USA. ACM, pp. 29–34.

Okazaki, N. (2007). Crfsuite: a fast implementation of conditional random fields (crfs).

Pettersson, E. and Nivre, J. (2015). Improving verb phrase extraction from historical text by use of verb valency frames. In Megyesi, B., editor, *Proceedings of the 20th Nordic Conference of Computational Linguistics (NODALIDA 2015)*.

Pinto, D., McCallum, A., Wei, X. & Croft, W. B. (2003). Table extraction using conditional random fields. In *Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Informaion Retrieval*, SIGIR '03, New York, NY, USA. ACM, pp. 235–242.

Piotrowski, M. (2012). *Natural language processing for historical texts*. Synthesis Lectures on Human Language Technologies. Morgan & Claypool Publishers.

Raaf, M. (2015). *Historical Corpora: Challenges and Perspectives*, chapter A web-based application for editing manuscripts, Gunter Narr Verlag, pp. 365–372.

Sato, K. and Sakakibara, Y. (2005). Rna secondary structural alignment with conditional random fields. *Bioinformatics*, 21(suppl 2), pp. 237–242.

Sha, F. and Pereira, F. (2003). Shallow parsing with conditional random fields. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology - Volume 1*, NAACL '03, Stroudsburg, PA, USA. Association for Computational Linguistics, pp. 134–141.

Shen, D., Sun, J.-T., Li, H., Yang, Q. & Chen, Z. (2007). Document summarization using conditional random fields. In *Proceedings of the 20th International Joint Conference on Artifical Intelligence*, IJCAI'07, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc., pp. 2862–2867.

Sutton, C. and McCallum, A. (2012). An introduction to conditional random fields. *Found. Trends Mach. Learn.*, 4(4), pp. 267–373.

Taskar, B., Abbeel, P. & Koller, D. (2002). Discriminative probabilistic models for relational data. In *Proceedings of the Eighteenth Conference on Uncertainty in Artificial Intelligence*, UAI'02, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc., pp. 485–492.

TEI Consortium, T., editor (2014). *TEI P5: Guidelines for Electronic Text Encoding and Interchange*, chapter Dictionaries. TEI Consortium, 2.6.0 edition.

Yu, D., Wang, S., Karam, Z. & Deng, L. (2010). Language recognition using deep-structured conditional random fields. In *Acoustics Speech and Signal Processing (ICASSP), 2010 IEEE International Conference on*, IEEE, pp. 5030–5033.