TAGGING NAMED ENTITIES IN CROATIAN TWEETS

Krešimir Baksa, Dino Dolović, Goran Glavaš, Jan Šnajder
Text Analysis and Knowledge Engineering Lab
Faculty of Electrical Engineering and Computing, University of Zagreb

Baksa, K., Dolović, D., Glavaš, G., Šnajder, J. (2016): Tagging Named Entities in Croatian Tweets. Slovenščina 2.0, 4 (1): 20–41.
DOI: http://dx.doi.org/10.4312/slo2.0.2016.1.20-41.

Named entity extraction tools designed for recognizing named entities in texts written in standard language (e.g., news stories or legal texts) have been shown to be inadequate for user-generated textual content (e.g., tweets, forum posts). In this work, we propose a supervised approach to named entity recognition and classification for Croatian tweets. We compare two sequence labelling models: a hidden Markov model (HMM) and conditional random fields (CRF). Our experiments reveal that CRF is the best model for the task, achieving a very good performance of over 87% micro-averaged F_1 score. We analyse the contributions of different feature groups and influence of the training set size on the performance of the CRF model.

Keywords: information extraction, named entity recognition, machine learning, social media, tweets, Croatian language.

1 INTRODUCTION

Named Entity Recognition and Classification (NERC) – a well-known task in information extraction (IE) and natural language processing (NLP) – aims at extracting and classifying names (personal names, organizations, locations), temporal expressions, and numerical expressions occurring in natural language texts. In many domains (e.g., journalism, intelligence, historical research) named entities constitute the key information for understanding and interpreting the

text. Robust named entity recognition and classification is also crucial for higher-level IE and NLP tasks such as relation extraction and sentiment analysis. For example, to determine the targets of sentiment, one first needs to recognize the people and organizations being mentioned in text.

Traditional NERC systems typically extract named entities from documents written in standard language (e.g., news stories, legal documents, police reports). In such professionally edited text, the correctness of language – in particular spelling, grammar, and vocabulary – is typically checked prior to publishing. In contrast, a large portion of textual content on the web (e.g., forum posts, blogs, and tweets) is user-generated and written in a non-standard language. Non-standard language is informal and colloquial, quite often orthographically and grammatically incorrect, and abundant with social-media jargon. This makes user-generated texts more challenging for automated processing than standard-language texts. It has been shown that the performance of the standard NERC systems drops significantly when applied to informal text (Liu, Zhang, Wei & Zhou, 2011).

In this article we address the task of named entity extraction and classification from tweets in Croatian. Tweets are messages from the micro-blogging service Twitter in which users post information ranging from news and trending events to personal information. The approach taken in this work is a well-trodden one: we first manually annotate tweets with named entities and then train supervised machine learning models to automatically recognize and classify named entities in tweets. We experiment with two supervised models – a Hidden Markov Model (HMM) and Conditional Random Fields (CRF) – and compare their performance in strinct and lenient evaluation setups. We then analyse how different feature groups and training set sizes affect the accuracy of the best-performing CRF model. We also show that the tweet-specific NERC model

significantly outperforms the state-of-the-art NERC model for Croatian trained on standard-language texts and applied on tweets. To the best of our knowledge, this is one of the very first works on named entity extraction from tweets for a Slavic language.

The rest of the article is structured as follows. In the next section, we provide an overview of work on NERC from tweets as well as on standard-language NERC for Croatian. In Section 3, we describe the dataset and the annotation process. In Section 4, we describe the different models and features used for the task, while in Section 5 we present the experimental results, including a feature ablation study and learning curve analysis. Finally, in Section 6, we conclude and outline ideas for future work.

2 RELATED WORK

The body of work on named entity recognition and classification from standard-language texts is overwhelming (Finkel, Grenager & Manning, 2005; Faruqui & Padó, 2010; Cucchiarelli & Velardi, 2001; Poibeau, 2003). In contrast, the work on NERC from tweets is scarce and so far limited mostly to English (Finin et al., 2010; Liu et al., 2011; Ritter, Clark, Etzioni et al., 2011; Li et al., 2012).

Finin et al. (2010) experimented with annotating named entities in tweets in English using crowdsourcing, showing that high-quality annotations can be obtained in a rather effective, fast, and cheap manner. Liu et al. (2011) used a semi-supervised approach to recognize and classify named entities in English tweets. They used a k-nearest neighbours classifier (k-NN) to pre-label the tweets, followed by sequence labelling with CRF for fine-grained named entity tagging. Ritter et al. (2011) developed a POS-tagger, a shallow parser, and a named entity recognizer for English tweets utilizing both in-domain and out-of-domain data. Their NERC system exploits the output of a tweet-adjusted POS-tagger, but

additionally relies on distant supervision by applying constrained topic modelling over a Freebase dictionary of entities. In contrast to the two aforementioned supervised approaches, Li et al. (2012) proposed an unsupervised, two-step NERC system for *targeted* Twitter streams (tweets filtered by user-specified criteria). The first step uses dynamic programming to segment the tweets into valid phrases constituting named entity candidates. The second step uses a random-walk model to rank the candidate phrases based on what the authors call *gregarious property*: the interaction of named entities and their co-ocurrence in targeted Twitter streams.

As regards NERC systems for the Croatian language, a number of systems for standard-language texts have been developed, both rule-based (Bekavac & Tadić, 2007) and statistical ones (Ljubešić, Stupar & Jurić, 2012; Karan et al., 2013). Ljubešić et al. (2012) trained the Stanford NER model (Finkel et al., 2005) on Croatian data manually annotated with basic named entity classes (Person, Organization, Location, Misc). Karan et al. (2013) developed CroNER, a supervised NERC system for Croatian that recognizes nine named entity classes (Person, Organization, Location, Ethnic, Date, Time, Currency, and Percentage). CroNER uses sequence labeling with conditional random fields (CRF), a rich set of lexical and gazetteer-based features, and enforces document-level consistency over individual classification decisions. CroNER is considered a state-of-the-art NER system for Croatian (Agić & Bekavac, 2013). Finally, the recently developed HeidelTime.Hr (Skukan, Glavaš & Šnajder, 2014) is a rule-based temporal expression tagger for Croatian that recognizes, classifies, and normalizes a variety of named entities belonging to the class of temporal expressions.

Following the work of Liu et al. (2011) and Karan et al. (2013), in this work we also rely on sequence labelling algorithms for named entity recognition and classification. However, our models are trained on manually annotated tweets

instead of standard-language texts. Similarly to Ljubešić et al. (2012), we focus on three main classes of named entities: Person, Organization, and Location.

3 DATASET AND ANNOTATIONS

3.1 Twitter corpus

To compile a dataset of tweets annotated with named entities, we adopt the Croatian Twitter Corpus[1] built by Ljubešić, Fišer and Erjavec (2014) with the open-source tool TweetCaT. TweetCaT[2] was created specifically to compile Twitter corpora for smaller languages, by collecting the URLs of web pages starting from a set of seed terms.

One challenge involved with compiling a Croatian corpus of tweets has to do with the fact that the Croatian language is quite similar to Bosnian, Montenegrin, and Serbian. A naïve approach to filtering out the non-Croatian tweets would be to resort to standard, n-gram based language identification. However, the problem with tweets is that the text is too short to allow for reliable language identification, and the problem is further exacerbated by the fact that standard language identification techniques often fail to discriminate between closely related languages (Tiedemann & Ljubešić, 2012). Thus, instead of relying on tweet-level language identification, Ljubešić et al. (2014) filtered the tweets at the user level, by analysing, for each user, the language in which he or she tweets most often. The so-obtained Croatian Twitter Corpus (hrTwitterCorpus) contains about 2 million tweets. From these, we sampled 5000 tweets for manual annotation. Subsequently, we removed some tweets that we deemed informationally irrelevant (e.g., *Ivana Ivana Ivana Ivana*), leaving us with the final dataset of 4,667 tweets.

[1] http://nlp.ffzg.hr/resources/corpora/twitter/

[2] https://github.com/nljubesi/tweetcat

As noted by Ljubešić et al. (2014), after user-level filtering, the hrTwitterCorpus still contains a considerable amount of tweets in languages other than Croatian as well as mixed-language tweets. Our manual analysis of a sample from 4,667 tweets revealed that roughly 30% of tweets tagged as Croatian are actually written in Serbian. Obtaining a perfecty filtered dataset would require considerable manual effort. We thus decided not to perform additional filtering, but instead to use the corpus with mixed Croatian and Serbian tweets. Arguably, from a machine learning perspective, using a mixed Croatian-Serbian corpus as the train set introduces some noise in all the cases in which the differences between the two languages are reflected in the feature values. On the other hand, our preliminary experiments, carried out on separate Croatian and Serbian test sets, have shown that the model performs equally well on both test sets. Thus, it seems that the upside of using a noisy dataset in this case is that one gets a model that works reasonably well for both languages.

3.2 Annotation

For the annotation of named entities, we compiled the annotation guidelines, partially adopted from Finin et al. (2010). The guidelines essentially amount to the following eight rules:

1. Annotate each token separately, following the B-I-O annotation scheme (e.g., *Hrvatska [B-ORG] narodna [I-ORG] banka [I-ORG]*);
2. Annotate names, surnames, and nicknames but not their titles (e.g., *doc. dr. sc.* as instances of the Person class (e.g., *Marko [B-PER]*; *dr. Ivo [B-PER] Josipović [I-PER]*);
3. Annotate names of concrete organizations, institutions, state authorities, sport clubs, national teams, but not generic terms like *government* or *party*, as instances of the Organization class (e.g., *NK [B-ORG] Rijeka [I-ORG]*);

4. Annotate mentions of places, regions, states, rivers, mountains, squares, streets, etc. as instances of the Location class (e.g., *Velika [B-LOC] Gorica [I-LOC]*);
5. Do not annotate tokens starting with "@" (usually indicating user names);
6. Do annotate named entities preceded by "#" (used for hashtags);
7. Annotate words considering the full context (e.g., the token *"Rijeka"* may denote a location but it may also be part of an organization mention, e.g., *"NK Rijeka"*);
8. When in doubt whether to annotate the word as an instance of Location or Organization class – a situation typical for metonimically used location names – give preference to Organization.

To reduce the annotation effort, we performed semi-automated annotation. It consists of two steps: (1) automated annotation of all mentions found in any of the precompiled gazetteers and (2) manual correction of errors (both false positives and false negatives) made in the first step.

Automated annotation. For the automated annotation, we first needed to compile a set of gazetteers. Gazetteers with personal names (2,413 entries) and locations (71 entries) were obtained from the Croatian Genealogy and Family History page[3] and a list of Croatian cities from Wikipedia,[4] respectively. For organization names, we did not use a proper gazetteer, but merely a list of 109 cue words, such as *tvrtka* (company), *firma* (company), *NK* (abbreviation for *football club*), etc. Following the automated gazetteer-based annotation, we manually corrected all errors and also labeled named entity mentions omitted by the automated annotation. Most omissions were in the organization names, due to the limited size of the cue words list and the fact that most organization names

3 http://www.croatian-genealogy.com

4 https://hr.wikipedia.org/wiki/Dodatak:Popis_gradova_u_Hrvatskoj

Class	MUC F_1 (%)	Exact F_1 (%)
Person	94.7	92.8
Organization	85.7	81.2
Location	86.6	85.2
Micro-average	91.3	88.8

Table 1: Inter-annotator agreement.

are multiword units, whereas our cue words list contained only single words. Also omitted were many location names, as our locations gazetteer contained only the names of Croatian cities and counties. Person names were mostly not omitted.

Manual annotation. The manual annotation was carried out by two annotators. Initially, both annotators independently annotated the same set of 500 tweets on which we measured the inter-annotator agreement (IAA) and assessed how well the annotation guidelines were followed. The IAA was measured by computing MUC and Exact F_1-scores between the annotations of the two annotators. In the MUC scheme, two annotations are considered the same if they have the same class and their extents overlap in at least one token. In the Exact evaluation scheme, the match is only counted when the two annotations are exactly the same (same class and exactly the same extent). IAA scores for the three considered named entity classes are given in Table 1.

Following the initial annotation of 500 tweets, each of the annotators annotated a separate set of approximately 2,230 tweets. These tweets were used for training and testing the supervised models. We make the annotated dataset freely available.[5]

[5] Available under the Creative Commons BY-NC-SA license from http://takelab.fer.hr/cronertweet

4 NERC MODELS

In this section we describe the three supervised machine learning models with which we experimented as well as the set of features employed by these models.

4.1 Machine learning models

We experiment with two supervised machine learning models to extract and classify named entities in tweets: (1) a Hidden Markov Model (HMM) and (2) Conditional Random Fields (CRF). For both models, we used the implementation in NLTK,[6] a widely used Python library for natural language processing.

Hidden Markov Model. This model is an extension of the Markov process where each state has all observations joined by the probability of the current state generating observation (Blunsom, 2004). Formally, HMM is defined as a tuple $\text{HMM} = (S, O, A, B, \pi)$, where S denotes hidden states (in our case the token-level labels) and O denotes outputs in each state (in our case all words observed in tweets). The remaining three components are the three parameters estimated from the training data: (1) the starting probabilities π (i.e., the probabilities of a Markov process starting from a certain state), (2) transition probabilities A of moving from one state to another, and output probabilities B of states emitting outputs, i.e., the probability of seeing a word when in a particular token-level NERC state.

Conditional Random Fields. CRF is a discriminative probabilistic graphical model that can model overlapping, non-independent features in a sequence of data. A special case, linear-chain CRF, can be thought of as the undirected graphical model variant of the HMM. Unlike HMM, which can essentially encode only words as features, CRF allows to extract arbitrary features for the current

[6] http://www.nltk.org/

token as well as for preceding and following tokens. We used a window of size five for extracting the features, i.e., all of the features were computed for the current token, its two preceding tokens, and its two following tokens.

4.2 Features

For the CRF model, we use the following set of 14 features:

- f_1 – The lemma of the token;
- f_2 – The length of the token;
- f_3 – A feature indicating whether the token contains an alphanumeric character;
- f_4 – A feature indicating whether the token contains a non-alphanumeric character (e.g., *Lovrić-Merzel*);
- f_5 – A feature indicating whether the token contains only non-alphanumeric characters (e.g., *?!*);
- f_6 – The shape of the token, encoding the lower/upper casing of the word (e.g., the shape of the word *Ana* is ULL);
- f_7 – A feature indicating whether the token contains a lower-cased letter;
- f_8 – A feature indicating whether the token contains only lower-cased letters;
- f_9 – A feature indicating whether the token contains an upper-cased letter;
- f_{10} – A feature indicating whether the token contains only upper-cased letters;
- f_{11} – A feature indicating whether the token contains digits (e.g., *sk8*);
- f_{12} – A feature indicating whether the token consists of four digits (useful for recognizing years);
- f_{13} – Features indicating whether the token matches a gazetteer entry (one feature per gazetteer, as a token can match multiple gazetteer entries);
- f_{14} – Features indicating whether the token is the first or the last token in the tweet;

For the HMM model, we used only one feature – the lemma of the word (f_1) – as other features cannot be incorporated into the standard HMM model.

5 EVALUATION

In this section we describe the experimental setup and discuss the performance of different models. To gain some more insights into the workings of the models, we carry out a feature analysis and an error analysis.

5.1 Experimental setup

We split our tweets dataset consisting of 4,667 tweets into three subsets: a train set (3,399 tweets), a validation set (423 tweets), and a test set (845 tweets). For the CRF model, we use the validation set for feature selection. For HMM, which uses only a single feature, we make no use of the validation set.

Feature selection. We designed the above described set of features following the typical "kitchen sink approach": we included in the model all the features that seem reasonable for the problem at hand and that can be easily computed. However, some of the features might be uninformative or even redundant, and may reduce the classifier performance. To select an optimal subset of features, we performed wrapper feature selection – a greedy search over the space of all possible features, using classifier's Exact F_1 score on the validation set as the objective function.

The resulting optimal subset of features contains the following 11 features: f_1–f_5, $f_7, f_8, f_{10}, f_{11}, f_{13}, f_{14}$. In other words, the three features that were droped are: f_6 (token shape), f_9 (whether the word contains any upper-cased letters), and f_{12} (whether the token consists of four digits).

	Baseline			HMM			CRF		
NE class	P	R	F_1	P	R	F_1	P	R	F_1
Person	96.5	84.4	90.0	93.8	81.5	87.2	94.8	92.7	93.8
Location	50.0	27.3	35.3	90.0	16.0	27.2	78.4	68.8	70.3
Organization	74.3	45.6	56.5	87.6	45.9	60.2	77.0	75.8	76.4
Macro	73.6	52.4	60.6	**90.5**	47.8	58.2	83.4	**79.1**	**81.1**
Micro	88.4	68.4	77.1	**92.6**	65.2	76.6	89.0	**86.1**	**87.5**

Table 2: MUC evaluation results.

Baseline. As the baseline, we use the automated approach that we used as the first step of the semi-automated annotation process – a token is tagged as a named entity of some type if it can be found in the gazetteer of the corresponding named entity type. The baseline model then joins adjacent tokens found in the same gazetteer into a single named entity mention. For instance, the sequence *KK Zadar* tagged in the first step by the baseline as *KK[ORG] Zadar[ORG]*, would be joined in the second step in to the sequence *KK[B-ORG] Zadar[I-ORG]*, tagged according to the B-I-O scheme.

5.2 Results

The results for both models and the baseline, for both MUC and Exact evaluation setups, are shown in Tables 2 and 3, respectively. The performance is reported for each of the named entity classes, along with both micro-averaged and macro-averaged performance.

The CRF outperforms HMM by a wide margin in both evaluation settings, which is in line with previous results where CRF has exhibited superior performance on various sequence labelling tasks in NLP. The CRF model reaches 87.5% of micro-averaged F_1 score in MUC evaluation setting and 81.0% of micro-averaged

	Baseline			HMM			CRF		
NE class	P	R	F_1	P	R	F_1	P	R	F_1
Person	64.4	55.9	59.9	84.6	73.9	78.9	89.2	88.0	88.6
Location	46.2	25.2	32.6	86.7	15.4	26.2	71.7	62.9	67.0
Organization	38.1	23.9	29.4	69.5	35.6	47.1	66.1	65.4	65.8
Macro	49.6	35.0	40.6	**80.3**	41.7	50.7	75.6	**72.1**	**73.8**
Micro	57.9	44.7	50.4	**82.1**	57.9	67.9	82.1	**80.0**	**81.0**

Table 3: Exact evaluation results.

F_1 score in exact evaluation setting. The CRF performance varies considerably across the named entity classes: in MUC evaluation setting, the best performance is achieved for Person class (98.8% F_1 score), whereas the worst performance is for the Location class (70.3% F_1 score). In exact setting, the best performance is again for the Person class (88.6% F_1 score), while the performance for both Location and Organization classes is considerably lower, 67.0% and 65.8% of F_1 score, respectively. Note that organization names are much more often multiword units than location names, hence Exact scores for Organization class are generally lower than MUC scores for the same class. The precision and recall are balanced for Person and Organization classes; for Location class the recall is about 10 percent points lower than precision.

Interestingly, HMM exhibits best precision but very low recall in both evaluation settings. In the MUC setting, HMM model does not even outperform the baseline in terms of F_1 score.

As an additional reference point, we evaluated CroNER (Karan et al., 2013) – a NERC system for stardard-language-texts – on our Twitter test set. CroNER exhibited micro-averaged performance of 35.8% F_1 score in the MUC setting, and merely 27.4% F_1 score of in the Exact evaluation setting. These results are

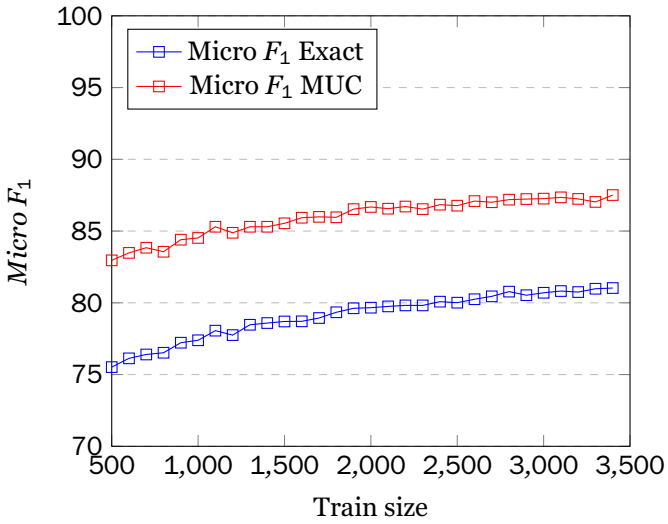

Figure 1: Learning curve of the CRF model.

in line with the observations for English (Liu et al., 2011) – the performance of the tagger built for texts written in standard language drops significantly when applied to tweets.

5.3 Learning curve

Machine learning models typically improve their performance when provided more training data. To determine whether this also holds in our case, we analyzed the learning curve of the CRF model. We trained the CRF classifier on datasets of different sizes, ranging from 500 to 3,400 tweets in increments of 100 tweets, and tested each on our test set. The so-obtained learning curve is shown in Figure 1. We notice that there is no substantial improvement in performance after training set size reaches approximately 2,000 tweets, suggesting that our initial training set (3.4K tweets) was sufficiently large for the chosen model.

5.4 Feature analysis

In Section 5.1 we explained how we used feature selection to obtain an optimal set of features for the CRF model. Though the feature set as a whole is optimal, the contribution of the individual features to the classifier decision might vary. To analyse the importance of the individual features, we carry out a feature ablation study: we group the 11 features chosen by feature selection into groups of related features and analyse how the performance of the CRF model changes when the model is trained without each of these feature groups. The groups of features are the following:

- $g_1 = \{f_2, f_5\}$
- $g_2 = \{f_3, f_4, f_7\}$
- $g_3 = \{f_{14}\}$ (token position)
- $g_4 = \{f_6, f_7, f_9\}$ (upper/lower case information)
- $g_5 = \{f_{13}\}$ (gazeteer feature)
- $g_6 = \mathrm{F} \setminus \{f_1\}$ (all futures but the lemma)
- $g_7 = \{f_1\}$ (the lemma)

For each of the groups, we removed all features from that group and trained the CRF model on the train set using only the remaining features. Each such model was then evaluated on test set. Table 4 shows the micro-averaged $F1$ score for different ablation settings (both MUC and Exact evaluation).

Removing feature group g_7 results in the largest performance drop, implying that lemma is the most important feature. Removing the gazetteer feature (g_5) also causes a significant drop in performance, confirming the intuition that gazetteer-based features are very important for named entity recognition. Dropping all features except for the lemma (feature group g_6) also results in significant performance drop, even such feature-deprived CRF model still outperforms HMM

	MUC		Exact	
Group	F_1	Δ	F_1	Δ
g_1	86.98	-0.53	80.53	-0.50
g_2	87.09	-0.42	80.76	-0.27
g_3	87.15	-0.36	80.80	-0.23
g_4	87.46	-0.05	80.91	-0.12
g_5	85.06	-2.45	78.93	-2.10
g_6	85.33	-2.18	79.21	-1.82
g_7	80.95	-6.56	73.07	-7.96
All	87.51		81.03	

Table 4: Feature ablation micro-averaged F_1 scores for the CRF model. Column Δ indicates the difference in model's performance between the full and ablated feature sets.

by a large margin (85.33% vs. 76.6% MUC and 79.21% vs. 67.9% Exact), which can be traced back to the discriminative vs. generative distinction.

5.5 Language-based data filtering

As mentioned in Section 3.1, Serbian tweets account for approximately 30% of our "Croatian" tweet dataset. Although closely related, the two languages have non-negligible differences, especially with respect to the writing of named entities (e.g., foreign names are phonetically transcribed in Serbian, whereas in Croatian they are written in their original form and transliterated in the Latin script if the original is non-Latin). Due to these differences, Serbian tweets may be considered as noise when training machine learning models for NER for Croatian.

To verify whether tweets in Serbian introduce noise and have any impact on the overall model performance, we carry out an experiment in which we au-

	MUC		Exact	
Dataset	F_1 micro	F_1 macro	F_1 micro	F_1 macro
Original	87.5	81.1	81.0	73.8
Filitered	86.9	80.6	82.3	76.0

Table 5: Comparison of CRF performance on the original and filtered dataset.

tomatically removed Serbian tweets from our dataset. To this end, we used the language identification tool for discriminating between very closely related languages developed by Tiedemann and Ljubešić (2012). The tool uses a Naïve Bayes classifier to predict the posterior probability of language given a tweet. To fine-tune the tool to our data, we use the manually annotated validation set of 423 tweets to optimize the decision threshold for which a tweet is considered to be in Serbian, using classifier's F_1 score as the objective function. The optimal threshold was 0.64, yielding F_1 score of 81.95%.

In Table 5 we compare the F_1 scores of the model trained on the original and filtered datasets. We observe that filtering the dataset by removing Serbian tweets did not yield any substantial improvement in performance, contradicting our intuition that Serbian tweets introduce noise in the learning process. Considering that filtering requires additional processing and that it reduces the size of the training set, we conclude that, for the task of NER from tweets, training on mixed Croatian and Serbian tweets may actually be beneficial.

6 CONCLUSION

Traditional IE and NLP tools have been shown ineffective when applied to user-generated content. This is especially true for tweets, micro-blogging messages filled with jargon vocabulary and abbreviations. In this article, we presented the work on named entity recognition from Croatian tweets. We semi-automatically

annotated the collection of almost 5.000 tweets in Croatian. We experimented with two sequence labeling models, demonstrating that CRF, being able to incorporate contextual features and labels, outperforms HMM as well as the competitive gazetteer-based baseline. The overall performance of the CRF model (87% micro-averaged MUC F_1-score) is comparable to the performance of the state-of-the-art NER system for Croatian standard language (90% micro-averaged MUC F_1-score) reported by Karan et al., 2013, which we consider very encouraging considering the lack of part-of-speech and syntactic information in current models.

There are several possible extensions of the work presented in this article. First, we intend to extend the models with part-of-speech and syntactic information. This means that a designated POS-tagger and (shallow) parser for tweets need to be created for Croatian as, similar to NER, respective tools built for standard-language texts have been shown inefficient. Secondly, considering that the removal of Serbian tweets from the training set did not improve the performance for Croatian tweets, we intend to evaluate the best-performing CRF model on tweets written in closely related languages like Serbian and Bosnian. Finally, we believe that we could further improve the extraction and classification performance by enforcing consistency of individual named entity decisions across tweets of the same thread (re-tweets) or across tweets of the same user, as was done for standard Croatian NER by Karan et al., 2013.

ACKNOWLEDGMENTS

This work has been supported by the Croatian Science Foundation under the project UIP-2014-09-7312.

REFERENCES

Agić, Ž. & Bekavac, B. (2013). Domain-aware Evaluation of Named Entity Recognition Systems for Croatian. *CIT. Journal of Computing and Information Technology*, *21*(3), 185–199.

Bekavac, B. & Tadić, M. (2007). Implementation of Croatian NERC system. In *Proceedings of the Workshop on Balto-Slavonic Natural Language Processing: Information Extraction and Enabling Technologies* (pp. 11–18). Association for Computational Linguistics.

Blunsom, P. (2004). Hidden Markov Models. *Lecture Notes*, *15*, 18–19.

Cucchiarelli, A. & Velardi, P. (2001). Unsupervised named entity recognition using syntactic and semantic contextual evidence. *Computational Linguistics*, *27*(1), 123–131.

Faruqui, M. & Padó, S. (2010). Training and evaluating a German named entity recognizer with semantic generalization. In *Proc. of KONVENS* (pp. 129–133).

Finin, T., Murnane, W., Karandikar, A., Keller, N., Martineau, J. & Dredze, M. (2010). Annotating named entities in Twitter data with crowdsourcing. In *Proceedings of the NAACL HLT 2010 Workshop on Creating Speech and Language Data with Amazon's Mechanical Turk* (pp. 80–88). Association for Computational Linguistics.

Finkel, J. R., Grenager, T. & Manning, C. (2005). Incorporating non-local information into information extraction systems by Gibbs sampling. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics* (pp. 363–370). Association for Computational Linguistics.

Karan, M., Glavaš, G., Šarić, F., Šnajder, J., Mijić, J., Silić, A. & Bašić, B. D. (2013). CroNER: Recognizing Named Entities in Croatian Using Conditional Random Fields. *Informatica (Slovenia)*, *37*(2), 165–172.

Li, C., Weng, J., He, Q., Yao, Y., Datta, A., Sun, A. & Lee, B.-S. (2012). TwiNER: Named Entity Recognition in Targeted Twitter Stream. In *Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval* (pp. 721–730). ACM.

Liu, X., Zhang, S., Wei, F. & Zhou, M. (2011). Recognizing named entities in tweets. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1* (pp. 359–367). Association for Computational Linguistics.

Ljubešić, N., Fišer, D. & Erjavec, T. (2014). TweetCaT: A Tool for Building Twitter Corpora of Smaller Languages. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*. Reykjavik, Iceland: European Language Resources Association (ELRA).

Ljubešić, N., Stupar, M. & Jurić, T. (2012). Building Named Entity Recognition Models for Croatian and Slovene. In *Proceedings of the Eighth Information Society Language Technologies Conference* (pp. 117–122).

Poibeau, T. (2003). The multilingual named entity recognition framework. In *Proceedings of the tenth conference on European chapter of the Association for Computational Linguistics-Volume 2* (pp. 155–158). Association for Computational Linguistics.

Ritter, A., Clark, S., Etzioni, O. et al. (2011). Named entity recognition in tweets: An experimental study. In *Proceedings of the Conference on Empirical Methods in Natural*

Language Processing (pp. 1524–1534).
Association for Computational Linguistics.

Skukan, L., Glavaš, G. & Šnajder, J. (2014). HeidelTime.Hr: Extracting and Normalizing Temporal Expressions in Croatian. In *Proceedings of the 9th Slovenian Language Technologies Conferences (IS-LT 2014)* (pp. 99–103).

Tiedemann, J. & Ljubešić, N. (2012).
Efficient Discrimination Between Closely Related Languages.
In *Proceedings of COLING 2012* (pp. 2619–2634). Mumbai, India.

PREPOZNAVANJE IMENSKIH ENTITET V HRVAŠKIH TVITIH

Obstoječa orodja za prepoznavanje imenskih entitet, ki so tipično izdelana za formalna besedila, napisana v standardnem jeziku (npr. novice, eseji ali pravna besedila), ne delujejo dobro na vsebinah, ki jih ustvarjajo uporabniki (npr. tviti). V prispevku predstavimo voden način za prepoznavanje in klasifikacijo imenskih entitet v hrvaških tvitih. Primerjava treh različnih modelov za označevanje zaporedij (HMM, CRF in SVM) je pokazala, da je najboljši model za to nalogo CRF, ki doseže za mikropovprečeno mero F_1 rezultat prek 87 %. Pokažemo tudi, da najboljši model za prepoznavanje hrvaških imenskih entitet v standardnem jeziku deluje mnogo slabše kot naši modeli za prepoznavanje imenskih entitet v tvitih.

Ključne besede: information extraction, named entity recognition, machine learning, social media, tweets, Croatian language.