Improving Named Entity Recognition for Morphologically Rich Languages using Word Embeddings

Hakan Demir*† and Arzucan Özgür†
*TÜBİTAK BİLGEM, Gebze, Kocaeli, Turkey
†Department of Computer Engineering, Boğaziçi University, Bebek, İstanbul, Turkey
hakan.demir@{tubitak.gov.tr, boun.edu.tr}, arzucan.ozgur@boun.edu.tr

***Abstract*—In this paper, we addressed the Named Entity Recognition (NER) problem for morphologically rich languages by employing a semi-supervised learning approach based on neural networks. We adopted a fast unsupervised method for learning continuous vector representations of words, and used these representations along with language independent features to develop a NER system. We evaluated our system for the highly inflectional Turkish and Czech languages. We improved the state-of-the-art F-score obtained for Turkish without using gazetteers by 2.26% and for Czech by 1.53%. Unlike the previous state-of-the-art systems developed for these languages, our system does not make use of any language dependent features. Therefore, we believe it can easily be applied to other morphologically rich languages.**

***Keywords*—Named Entity Recognition, Word Embeddings, Skip-gram, Turkish NER, Czech NER**

I. Introduction

Named Entity Recognition (NER), which is an important task of natural language processing (NLP), is a constituent of many NLP tasks including information extraction, machine translation, and question answering. The goal of NER is to locate and classify words in text into predefined categories such as names of persons, organizations, locations, expressions of times, quantities, monetary values, and percentages.

Conditional Random Fields (CRF) [1] are one of the most successful and widely used techniques for sequence labeling in several NLP tasks including NER [2], [3]. However, recently, the neural network based semi-supervised learning approach has gained attention in the English NER studies [4]–[7]. In the unsupervised stage of this approach, continuous vector representation of words are attained using a large amount of unlabeled data by employing a neural network. In the supervised stage, these feature vectors along with additional features are fed to another neural network to train a NER system. Since word representations constitute an important part of this approach, it is crucial to find good representations. The initial methods proposed to learn word representations have the drawback of large training durations with typical values of a few weeks [6], [7]. Recently, Mikolov et al. [8] showed that vector representations of words can be attained considerably faster, in a matter of hours, by employing a simpler neural network model. Although both representations are comparable, to our best knowledge, up until now, these representations have not been used for NER before.

Morphologically rich languages such as Turkish, Czech, and Hungarian differ substantially from English, since they have agglutinative or inflectional morphologies. In such languages production of hundreds of words from a given root is possible, which results in the data sparsity problem. To illustrate, consider the following Turkish word *bul-un-ama-yabil-en*. It corresponds to *The one that may possibly not be found* in English. To be more concrete, using English and Turkish corpora of around 10 million words, Hakkani-Tür [9] showed that the vocabulary size for English is $97,734$ and for Turkish is $474,957$. However, when only root forms are considered, the vocabulary size for Turkish drops to $94,235$. This analysis shows that on average 5 different Turkish word forms are generated from the same root. Due to this data sparsity problem, state-of-the-art systems for NER in morphologically rich languages usually make use of the analysis of the morphological structures of the languages and require language specific feature engineering [10], [11]. However, this makes them difficult to adapt to other languages.

In this paper, we investigated using the neural network based semi-supervised learning approach for NER in morphologically rich languages. Unlike the previous semi-supervised NER studies, in the unsupervised stage for obtaining the word representations, we used the approach of Mikolov et al. [8]. In the supervised stage, in addition to these feature vectors, we benefited from additional language independent features such as word capitalization patterns, previous tag prediction, etc. Our system is completely language independent and can easily be applied to other languages.

We evaluated the performance of our system on two highly inflectional morphologically rich languages, namely Turkish and Czech. We reported our results using the commonly accepted CoNLL metric [12] and compared them with the state-of-the-art works for Turkish and Czech. The current state-of-the-art systems for Turkish and Czech NER are based on CRF and make use of language dependent features. The state-of-the-art result evaluated with respect to CoNLL metric for NER task in Turkish is reported by Şeker and Eryiğit [10] and has an F-score performance of 89.59% without gazetteers and 91.94% with gazetteers. For the Czech language, it is reported by Konkol and Konopík [11] and has a 74.08% performance with gazetteers. Our system achieved an F-score performance of 91.85% for Turkish and 75.61% for Czech without using gazetteers and any language-specific features.

The main contributions of our paper can be summarized as follows: Firstly, we show that the neural network based semi-supervised learning approach that makes use of word embeddings can be successfully applied to morphologically rich languages without performing any language specific morphological analysis. Secondly, we show that word representations obtained very fast by employing the approach of Mikolov et al. [8] are useful for NER. Finally, we outperform the state-of-the-art results obtained for Turkish without using gazetteers and for Czech obtained with using gazetteers.

The remainder of this paper is organized as follows. The related work is discussed in the next section. The architecture of our system is described in Section 3. The data sets are given in Section 4. The experiments and results are presented in Section 5, and the paper is concluded in Section 6.

II. Related Work

Finding distributed representations has a long history [13], [14]. A neural network based architecture for estimating vector representations of words was proposed by Bengio et al. [15]. Collobert and Weston [4] showed that these distributed representation of words are useful for a supervised neural network that aims to accomplish various NLP tasks including NER. In their later work, by implementing the same technique Collobert et al. [7] achieved state-of-the-art performance results in several NLP tasks including NER. Their work improved the state-of-the-art accuracy for English NER from 89.31% to 89.59%. One of the challenges for using this approach is the long neural network training times, which can take up to a few weeks, in the unsupervised stage for obtaining the distributed representation of words. Mikolov et al. [8] showed that vector representations of words can be attained very fast (in a few hours) by employing a simpler model.

Morphologically rich languages pose challenges for several NLP tasks including NER. State-of-the-art systems developed for such languages usually depend on manually designed language specific features that utilize the rich morphological structures of the words. In this study we propose using the semi-supervised neural network based approach for NER in morphologically rich languages without making use of any language dependent features. We applied the proposed system to two morphologically rich languages: Turkish and Czech. One of the first studies on Turkish NER was conducted by Tür et al. [16], who employed a Hidden Markov Model (HMM) based approach and evaluated it with respect to the MUC metrics [17], [18]. They reported an F-score performance of 91.56% on the general news domain with ENAMEX (person, location and organization) type. Tatar and Çiçekli [19] developed an automated rule learning system and reported 91.08% F-score on terrorism news using the MUC metric with ENAMEX and TIMEX (date and time) entity types. Yeniterzi [20] obtained an F-score performance of 88.94% by using CRF and exploiting the effect of morphology. Finally, the state-of-the-art work for Turkish NER was introduced by Şeker and Eryiğit [10]. They also employed CRF with some additional features based on the morphological analysis of the text. They evaluated their system using both the MUC and CoNLL metrics and reported an F-score performance of 89.59% without gazetteers and 91.94% with gazetteers in CoNLL metric. They achieved an F-score of 92.83% without gazetteers and 94.59% with gazetteers in MUC metric.

Most work on Czech NER differs from traditional NER tasks because the most widely used corpus, which is the publicly available Czech Named Entity Corpus (CNEC) [21], is tagged in a hierarchical manner. In this type of annotation, a named entity belongs to a supertype and a type that results in different evaluation structures. For Czech NER, there have been a number of studies based on decision trees [21], support vector machines [22], maximum entropy classifier [23], [24] and CRF [11], [25]. Among these studies, Straková et al. [24] and Konkol and Konopík [11] held state-of-the-art results. However, only Konkol and Konopík [11] evaluated their system according to the CoNLL metric and reported an F-score performance of 74.08%.

III. Semi-supervised Learning based Model for NER

Our neural network based system consists of two stages. The first stage makes use of a huge amount of unlabeled data, whereas the second stage uses a rather restricted amount of labeled data. This kind of learning is called semi-supervised learning due to the fact that it makes use of both labeled and unlabeled data. The following subsections describe the details of our system.

A. Unsupervised Stage: Learning Word Representations

The main feature used by our NER model is the continuous word representations learned in the unsupervised stage. Therefore, the ability of our method to learn good vector representations of words, which map semantically similar words close to each other in the continuous vector space, is vital.

In order to obtain the continuous space vector representations of words, we used the publicly available implementation of Mikolov et al. [8], word2vec[1], since it is much faster than the methods proposed by Bengio et al. [15] and Collobert et al. [7]. Most of the complexity in the work of Bengio et al. [15] and Collobert et al. [7] is introduced by the non-linear hidden layer in their models. Although this is what makes their models strong, it has the drawback of the long training times to obtain the vector representations of words, which restricts the amount of unlabeled data that can be used. Due to the fact that the non-linear hidden layer is removed and the projection layer is shared for all words in the architecture of Mikolov et al. [8], we were able to train our model with a huge amount of unlabeled data. Benefiting from large amounts of data is important, since as the amount of data increases, the obtained feature vectors of words become more representative.

Among the techniques described in [8], we used the continuous Skip-gram model since it has been shown to be more successful at obtaining semantic representations of words [8]. The Skip-gram architecture tries to maximize the classification of a word based on the other words in the same sentence. Initially, words are mapped to random vectors of a specified dimension. Then, each word is used as an input to a log-linear classifier with a projection layer, and representations of

[1] https://code.google.com/p/word2vec/

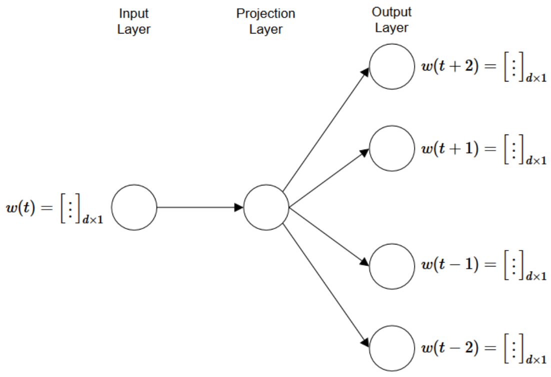
Input Layer
Projection Layer
Output Layer
$w(t) = \left[\vdots\right]_{d\times1}$
$w(t+2) = \left[\vdots\right]_{d\times1}$
$w(t+1) = \left[\vdots\right]_{d\times1}$
$w(t-1) = \left[\vdots\right]_{d\times1}$
$w(t-2) = \left[\vdots\right]_{d\times1}$

Fig. 1. Architecture of the Skip-gram model that is employed to learn continuous vector representation of words [8].

words are predicted within a certain range before and after the current word. Given a sequence of words, $w_1, w_2, \ldots, w_T$, the objective of the Skip-gram model is to maximize the average log probability

$$\frac{1}{T}\sum_{t=1}^{T}\left[\sum_{\substack{j=-k\\ j\neq 0}}^{k}\log p(w_{t+j}|w_t)\right]$$

where k is the size of the predefined range. Given a word, the inner summation computes the sum of the log probabilities of the previous and next k words to it. The outer summation repeats this for all words.

As the range or dimension of vectors increases, the quality of the resulting vectors increases as well as the complexity. For our task, we chose the dimension as 200 and the range as 5, that is the representation of the previous and the next two words are predicted from the current token. The architecture we used is shown in Figure 1.

Using the Skip-gram architecture with a huge amount of unlabeled data in Turkish and Czech, we got the continuous space vector representations of words for both languages. By using the word2vec tool, we examined sample words and their closest neighbours in the vector space. We observed that the nearest neighbours are highly related semantically to the queried words. Sample words in Turkish and their seven nearest neighbours in the vector space are shown in Table I. The first words in columns 1 and 2 are person names in Turkish, and our model lists seven other person names as their nearest neighbours in the vector space. The first words in columns 3 and 4 are location names, “elazığ” is a city in Turkey and “ingiltere” (England) is a country name. The seven closest neighbours to “elazığ” are also cities in Turkey and the closest neighbors to “ingiltere” are also country names in Turkish. Finally, the first words of columns 5 and 6 are organization names, and organization entities are brought by our model as the nearest neighbours.

The results in Table I show that semantically similar words in natural language are placed close to each other in the vector space. This is a very useful feature, especially for NER, where the semantic roles of words have important effects on distinguishing the named entity classes.

TABLE I. EXAMPLE WORDS IN TURKISH AND THEIR SEVEN CLOSEST NEIGHBORS

ahmet (name)	**ayşe (name)**	**elazığ (city)**	**ingiltere (england)**	**huawei (org.)**	**dell (org.)**
osman (name)	zeynep (name)	çorum (city)	italya (italy)	zte (org.)	toshiba (org.)
mehmet (name)	necla (name)	erzurum (city)	almanya (germany)	ericsson (org.)	lenovo (org.)
ismail (name)	zeliha (name)	mardin (city)	fransa (france)	ibm (org.)	nokia (org.)
ali (name)	hatice (name)	bitlis (city)	hollanda (holland)	cisco (org.)	samsung (org.)
mustafa (name)	fatma (name)	yozgat (city)	belçika (belgium)	fujitsu (org.)	microsoft (org.)
cafer (name)	elif (name)	sivas (city)	ispanya (spain)	lenovo (org.)	apple (org.)
salih (name)	filiz (name)	gümüşhane (city)	isveç (sweden)	nokia (org.)	ibm (org.)

In addition to learning word representations, we also investigated the impact of incorporating word clustering to our NER system. We utilized the word2vec tool to cluster the resulting vector representations of the words using the k-means algorithm, and included the computed cluster ids of the words as additional features to our final NER system. Interestingly, these word vector clusters, which have not been used for NER before, led to slight improvement in performance.

B. *Supervised Stage: Training NER Models*

The supervised learning stage is where the NER models are formed. Since the features incorporated to form a model determine the quality of the resulting system, researchers tend to use language dependent features to increase the performances of their systems. The current state-of-the-art NER systems developed for morphologically rich languages are also usually based on manually designed features that utilize the language specific morphological analysis of the text. Although this approach usually improves the performance of a system due to the usage of linguistic knowledge, it is not portable and cannot be easily applied to different languages. In this study, we refrained from employing such engineered features. Instead, we restricted ourselves to only language independent features.

In order to train our models, we used the publicly available neural network implementation of Ratinov and Roth [26][2]. In their work, they implemented a regularized averaged perceptron, which is a classic and effective learning algorithm for neural networks. In our framework, we did not change the model architecture, but added some extra features. We exploited both local and non-local features. The local features are related to the neighbors of the current token, x_i, whereas the non-local ones ignore sentence boundaries and consider global dependencies. The features that we used are summarized below.

- *Context:* The tokens in the window of size two $c_i = (x_{i-2}, x_{i-1}, x_i, x_{i+1}, x_{i+2})$
- *Previous tags:* Named entity tag predictions of the previous three tokens
- *Type information:* Type information of the window c_i, i.e. is-capitalized, all-capitalized, all-digits, is-alphanumeric, and contains-apostrophe.

[2] http://cogcomp.cs.illinois.edu/Data/ACL2010_NER_Exp eriments.php

- *Prefixes:* First three and four characters of x_i if it contains that many characters
- *Suffixes:* Last one, two, three, and four characters of x_i if it contains that many characters
- *Word representations:* Vector representation of each element in the window c_i
- *word2vec clusters:* We investigated the contribution of word2vec clusters and tried different numbers of clusters to obtain the best performance improvement. Finally, we found that 2000 and 256 clusters suit best for Turkish and Czech NER tasks, respectively. We used the cluster id of each element in the window c_i.
- *Context aggregation:* The tokens that are the same as the current token within a window of size 200 are investigated. The context features of each of these tokens are aggregated [26].
- *Extended prediction history:* Tag predictions of the tokens that are the same as the current token within the previous 1000 words are investigated. Then, the tag distribution of the token is used as a feature.

Normalization is applied to numerical expressions as in the work of Turian et al. [6]. To illustrate, 2014 is represented as *DDDD* and (0212) 153 69 74 is represented as (*DDDD*) *DDD* *DD* *DD*. Normalization is employed since it enables achieving a degree of abstraction to numerical expressions [6].

Besides the features used, the representation of the named entities also affects the performance of a NER system. Among the alternative encoding schemes, such as BIO and BILOU, we used BILOU as the representation scheme, since it has been shown to perform better than the BIO representation [26]. In the BILOU representation scheme a named entity that has multiple tokens is encoded as **B**eginning, **I**nside and **L**ast, and as **U**nit if it has one token. If the token is not a named entity it is encoded as **O**utside of any type of named entity. During testing, after tagging with respect to the BILOU scheme, the tags are converted to BIO tags. That is the tokens tagged as U and L are changed to B and I, respectively. This is required for using the standard performance evaluation method.

IV. Data Sets

In this section, we provide the details of the unlabeled and labeled data sets that we used for training and testing our system.

A. Turkish Data Sets

In the unsupervised stage, we used data collected from several Turkish news sites. We tokenized the data by using the publicly available zemberek[3] tool and then lowercased it. By lowercasing, we aimed to limit the number of words. The data that we used in this stage contain 63.72M sentences that correspond to a total of 1.02B words and 1.36M unique words.

TABLE II. Number of entities in the data sets

(a) Turkish data set

	Train	Test
PER	14481	1594
ORG	9034	864
LOC	9409	1090

(b) Czech data set

	Train	Dev.	Test
Addr.	109	23	14
Geo.	2890	399	340
Inst.	2595	322	309
Media	244	34	32
Person	3704	497	472
Time	2384	275	361
Other	2432	321	378

In the supervised stage, we used the data set prepared by Tür et al. [16] which is the most commonly used one for evaluating Turkish NER systems including the state-of-the-art system. It is partitioned into training and test sets, that contain 450K words and 50K words, respectively. This data set contains person (PER), location (LOC) and organization (ORG) entities. The number of entities are shown in Table II(a). It is worth noting that no matter whether an entity consists of only one token or more, it is counted as one, since the CoNLL evaluation task considers phrases and not tokens.

B. Czech Data Sets

For the unlabeled data, we used the publicly available data crawled from Czech news sites provided by the ACL machine translation workshop[4]. We tokenized the data using the Moses tokenizer[5] and then applied lowercasing. This data set contains 36.42M sentences corresponding to 635.99M words and 906K unique words.

While training and testing our Czech NER system in the supervised stage, we used the CNEC 1.1 data set prepared by Ševcíková et al. [21]. It is divided into training, development, and test sets, which contain 124K, 15K, and 15K tokens, respectively. The number of entities in these sets are shown in Table II(b). Unlike the traditional tagging schemes, CNEC is annotated by using two level hierarchical named entities. The first level is named as *supertype* and the second level is named as *type*. An example sentence from the data set is shown below (It means: *It's a bitter disappointment and warning for our hockey, but the misery continued even in the duel with Devils Milan.*).

```
  Pro náš hokej trpké poznání a výstraha ,
ale trápení pokračovalo i v souboji s
<ic Devils <gu Milán>> .
```

The first character of a tag determines the supertype of a named entity and the second character determines its type. In the example sentence, "i" tells us that "Devils Milán" is an institution name, "c" tells us that "Devils Milán" is a cultural/educational/scientific institution, "g" tells us that "Milán" is a geographical name, and lastly "u" tells us that "Milán" is a castle/chateau. Although this type of tagging is much more informative than the traditional ones, it leads to different types of evaluation approaches. Therefore, we used the transformed version of this data set prepared by Konkol and Konopík [11]. The original corpus used 10 supertypes and 62 types, whilst the transformed corpus uses only 7 supertypes. This transformation, which in fact makes the task more challenging [11], aims to make the data set compatible

[3] https://github.com/ahmetaa/zemberek-nlp

[4] http://www.statmt.org/wmt14/translation-task.html

[5] https://github.com/moses-smt/mosesdecoder/tree/master/scripts/tokenizer

TABLE III. F-MEASURE FOR EACH FEATURE IN CZECH NER. CONTEXT FEATURES ARE CHOSEN TO BE BASELINE AND EACH FEATURE IS ADDED TO IT INDEPENDENTLY.

Features	Addr.	Geo.	Inst.	Media	Other	Person	Time	Overall
Context features	22.22	56.18	29.88	34.04	50.14	40.59	86.42	53.70
Previous tags	33.33	56.51	36.65	37.50	58.13	45.31	88.24	58.36
Word type	25.00	61.04	37.44	40.82	48.78	61.69	87.34	59.84
Affixes	23.53	61.49	33.52	36.74	49.59	54.39	85.60	57.45
Word representations	35.29	69.25	46.26	42.31	51.58	66.25	88.98	64.72
word2vec clusters	22.22	64.07	37.62	42.31	51.25	58.23	88.46	60.53

TABLE IV. F-MEASURE FOR EACH FEATURE IN TURKISH NER. CONTEXT FEATURES ARE CHOSEN TO BE BASELINE AND EACH FEATURE IS ADDED TO IT INDEPENDENTLY.

Features	PER	ORG	LOC	Overall
Context features	81.65	74.96	88.43	82.21
Previous tags	84.84	78.86	89.23	84.84
Word type	88.49	78.85	89.32	86.43
Affixes	83.04	76.82	87.96	83.10
Word representations	91.24	79.95	90.50	88.28
word2vec clusters	88.73	77.03	89.43	86.15

TABLE V. F-MEASURES WHEN FEATURES ARE ADDED CUMULATIVELY IN TURKISH NER.

Features	PER	ORG	LOC	Overall
Context features	81.65	74.96	88.43	82.21
+ Previous tags	84.84	78.86	89.23	84.84
+ Word type	91.45	82.45	90.17	88.94
+ Affixes	92.26	83.53	90.73	89.73
+ Word representations	94.36	85.51	92.61	91.71
+ word2vec clusters	94.69	85.78	92.40	**91.85**

with the CoNLL evaluation metric, so that the results become comparable with other systems.

V. EXPERIMENTS AND RESULTS

In the evaluation phase, we first trained our word representations. Then, we trained our NER models using these word representations. The first stage took around 1 hour for Turkish words and 30 minutes for Czech words due to the different sizes of the corpora for these languages. In the second stage, training a NER model for a language took around 7 hours. All experiments are performed with a computer having 16 GB RAM and Intel Core i7 processor.

In order to explore the contribution of each feature we used, we trained six different models for each language. We chose *context features* to be the base and added each feature to it. The results are evaluated with respect to the CoNLL metric and shown in Tables III and IV. The first rows in the tables correspond to the results obtained when *context features* are used solely. Each successive row shows the performance of the corresponding feature combined with the base feature. These experiments show that the order of importance of each feature to NER performance seems to be parallel. We think that this is because both languages are similar in terms of their morphology. The results indicate that the *word representation* feature contributes most and the *word2vec clusters* feature, which has not been tried for NER before, contributes remarkably.

We also examined the cumulative contribution of the features. The results are shown in Tables V and VI. The first rows in the tables correspond to the results obtained when only *context features* are used. Each successive row shows the performance obtained after including the corresponding feature to the model in the previous row. Since *previous tag*, *word type* and *affix* features are disjoint, they contribute as much as they did to the base feature. However, this is not the same for the *word representation* feature. This is because it learns a part of these features as well.

The comparison between our system and the state-of-the-art system for Turkish [10] is given in Table VIII. In Şeker and Eryiğit [10], CRF is employed as the learning algorithm. In addition to some language independent features, a number of language dependent features are also used. To be more precise, the stems of words, their part of speech tags, all inflectional features of the tokens, and information whether a token is a proper noun or not, are used. Including these language dependent features resulted in an F-measure of 89.59% in CoNLL metric without using gazetteers, and an F-measure performance of 91.94% with gazetteers. Our system outperforms these results without using any language dependent features, when gazetteers are not included[6].

We also compared our system with the state-of-the-art Czech NER system [11]. The comparison is shown in Table VII. In fact, both Straková et al. [24] and Konkol and Konopík [11] hold state-of-the-art results for Czech NER. However, only Konkol and Konopík [11] evaluated their system according to the CoNLL metric. Therefore, we were able to compare our system with theirs. They report an F-score performance of 74.08% with gazetteers. As the learning algorithm, they used CRF. Their approach includes some language dependent features such as word lemmas obtained by language specific morphological analysis and gazetteer lists. It is worth noting that our system does not use any gazetteers and still outperforms their approach with an F-score of 75.61%.

The number of entities in the training sets are not distributed uniformly, see Tables II(a) and II(b). Analyzing the training set sizes for each tag suggests that the performance of our system is relatively lower when there is less training data, as expected. For instance, there are only 109 *Address* tags in the Czech training set but over 2000 *Time* tags. This is one possible reason why *Address* tag perform worse compared to *Time* tag. In addition to this, the test sizes for the *Address* and *Media* tags are 14 and 32 respectively, which make their performance evaluation fragile. Therefore, results obtained for these classes of entities may not represent the quality of the model.

VI. CONCLUSION

In this paper, we investigated NER in morphologically rich languages. First, we learned continuous space vector representations of words from unlabeled data collected from a number of news sites. Then, by using these word representations and

[6] We were not able to make a comparison of our system with the gazetteer feature added, since the gazetteers used in Şeker and Eryiğit [10] are not publicly available.

TABLE VI. F-MEASURES WHEN FEATURES ARE ADDED CUMULATIVELY IN CZECH NER.

Features	**Addr.**	**Geo.**	**Inst.**	**Media**	**Other**	**Person**	**Time**	**Overall**
Context features	22.22	56.18	29.88	34.04	50.14	40.59	86.42	53.70
+ Previous tags	33.33	56.51	36.65	37.50	58.13	45.31	88.24	58.36
+ Word type	44.44	61.13	44.52	47.83	60.09	65.59	88.86	64.68
+ Affixes	44.44	67.14	48.00	44.00	62.99	70.31	90.27	68.38
+ Word representations	33.33	76.77	64.18	53.85	64.82	80.36	89.90	75.52
+ word2vec clusters	33.33	76.03	62.86	54.90	65.81	81.39	89.96	**75.61**

TABLE VII. COMPARISON OF OUR SYSTEM WITH THE STATE-OF-THE-ART CZECH NER

System	**Addr.**	**Geo.**	**Inst.**	**Media**	**Other**	**Person**	**Time**	**Overall**
Konkol and Konopík, (2013)	58.33	77.37	67.02	39.13	55.96	82.29	86.68	74.08
Our final system	33.33	76.03	62.86	54.90	65.81	81.39	89.96	**75.61**

TABLE VIII. COMPARISON OF OUR SYSTEM WITH THE STATE-OF-THE-ART TURKISH NER

System	**PER**	**ORG**	**LOC**	**Overall**
Şeker and Eryiğit, (2012) without gazetteer	90.65	86.12	90.74	89.59
Our final system	94.69	85.78	92.40	**91.85**

additional features extracted from the data, we trained an averaged perceptron using labeled data sets to learn NER models for Turkish and Czech, which are highly inflectional morphologically rich languages. Finally, we evaluated our method using the CoNNL metrics and compared our results with the state-of-the-art systems proposed for Turkish and Czech, which make use of language-specific morphological analysis. We showed that utilizing the continuous vector space representations of words in a semi-supervised setting is a powerful approach for NER, and can result in state-of-the-art performance without using any language dependent features for morphologically rich languages. Therefore, we believe, this approach can be easily applied to other languages.

As a future work, we plan to investigate whether the performance of the proposed system can be further improved by incorporating language-specific features and gazetteers. We also plan to evaluate the system for other domains including social media.

REFERENCES

[1] J. D. Lafferty, A. McCallum, and F. C. N. Pereira, "Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data," in *Proceedings of the Eighteenth ICML*, 2001, pp. 282–289.

[2] A. McCallum and W. Li, "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*, 2003, pp. 188–191.

[3] A. Ekbal and S. Bandyopadhyay, "A Conditional Random Field Approach for Named Entity Recognition in Bengali and Hindi," *Linguistic Issues in Language Technology*, vol. 2, no. 1, 2009.

[4] R. Collobert and J. Weston, "A Unified Architecture for Natural Language Processing: Deep Neural Networks with Multitask Learning," in *Proceedings of the 25th International Conference on Machine Learning*, 2008, pp. 160–167.

[5] J. Turian, Y. Bengio, L. Ratinov, and D. Roth, "A Preliminary Evaluation of Word Representations for Named-entity Recognition," in *NIPS Workshop on Grammar Induction, Representation of Language and Language Learning*, 2009.

[6] J. Turian, L. Ratinov, and Y. Bengio, "Word Representations: A Simple and General Method for Semi-supervised Learning," in *Proceedings of the 48th Annual Meeting of the ACL*, 2010, pp. 384–394.

[7] R. Collobert, J. Weston, L. Bottou, M. Karlen, K. Kavukcuoglu, and P. Kuksa, "Natural Language Processing (Almost) from Scratch," *Journal of Machine Learning Research*, vol. 12, pp. 2493–2537, 2011.

[8] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient Estimation of Word Representations in Vector Space," *CoRR*, vol. abs/1301.3781, 2013.

[9] D. Z. Hakkani-Tür, "Statistical Language Modelling for Turkish." Ph.D. thesis, Bilkent University, 2000.

[10] G. A. Seker and G. Eryigit, "Initial Explorations on using CRFs for Turkish Named Entity Recognition," in *Proceedings of the 24th International Conference on Computational Linguistics*, 2012, pp. 2459–2474.

[11] M. Konkol and M. Konopík, "CRF-Based Czech Named Entity Recognizer and Consolidation of Czech NER Research," in *Text, Speech and Dialogue (TSD)*, 2013, pp. 153–160.

[12] E. F. Tjong Kim Sang and F. De Meulder, "Introduction to the CoNLL-2003 Shared Task: Language-independent Named Entity Recognition," in *Proc. of the 7th CoNLL at HLT-NAACL*, vol. 4, 2003, pp. 142–147.

[13] G. E. Hinton, J. L. McClelland, and D. E. Rumelhart, "Distributed Representations," in *Parallel distributed processing: Explorations in the microstructure of cognition. Volume 1: Foundations*, 1986, pp. 77–109.

[14] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning Representations by Back-propagating Errors," *Nature*, vol. 323, no. 6088, pp. 533–536, 1986.

[15] Y. Bengio, R. Ducharme, P. Vincent, and C. Janvin, "A Neural Probabilistic Language Model," *Journal of Machine Learning Research*, vol. 3, pp. 1137–1155, 2003.

[16] G. Tür, D. Hakkani-tür, and K. Oflazer, "A Statistical Information Extraction System for Turkish," *Natural Language Engineering*, vol. 9, no. 2, pp. 181–210, 2003.

[17] R. Grishman and B. Sundheim, "Design of the MUC-6 Evaluation," in *Proceedings of the 6th Conference on Message Understanding*, 1995, pp. 1–11.

[18] N. A. Chinchor, "Named Entity Task Definition," in *Proceedings of the Seventh Message Understanding Conference (MUC-7)*, 1998, p. Appendix E.

[19] S. Tatar and I. Cicekli, "Automatic Rule Learning Exploiting Morphological Features for Named Entity Recognition in Turkish," *Journal of Information Science*, vol. 37, no. 2, pp. 137–151, 2011.

[20] R. Yeniterzi, "Exploiting Morphology in Turkish Named Entity Recognition System," in *Proc. of the ACL Student Session*, 2011, pp. 105–110.

[21] M. Ševčíková, Z. Žabokrtsky, and O. Krůza, "Named Entities in Czech: Annotating Data and Developing NE Tagger," in *Proceedings of the 10th International Conference on TSD*, 2007, pp. 188–195.

[22] J. Kravalová and Z. Žabokrtský, "Czech Named Entity Corpus and SVM-based Recognizer," in *Proceedings of the 2009 Named Entities Workshop: Shared Task on Transliteration*, 2009, pp. 194–201.

[23] M. Konkol and M. Konopík, "Maximum Entropy Named Entity Recognition for Czech Language," in *Proceedings of the 14th International Conference on TSD*, 2011, pp. 203–210.

[24] J. Straková, M. Straka, and J. Hajic, "A New State-of-the-Art Czech Named Entity Recognizer." in *Text, Speech and Dialogue (TSD)*, 2013, pp. 68–75.

[25] P. Král, "Features for named entity recognition in czech language," in *Knowledge Engineering and Ontology Development (KEOD)*, 2011, pp. 437–441.

[26] L. Ratinov and D. Roth, "Design Challenges and Misconceptions in Named Entity Recognition," in *Proceedings of the Thirteenth Conference on Computational Natural Language Learning*, 2009, pp. 147–155.