Named Entity Recognition from Scratch on Social Media

Kezban Dilek Onal and Pinar Karagoz

Middle East Technical University, Ankara, Turkey
{dilek, karagoz}@ceng.metu.edu.tr

Abstract. With the extensive amount of textual data flowing through social media platforms, the interest in Information Extraction (IE) on such textual data has increased. Named Entity Recognition (NER) is one of the basic problems of IE. State-of-the-art solutions for NER face an adaptation problem to informal texts from social media platforms. In this study, we addressed this generalization problem with the *NLP from scratch* idea that has been shown to be successful for several NLP tasks on formal text. Experimental results have shown that word embeddings can be successfully used for NER on informal text.

Keywords: NER, word embeding, NLP From Scratch, Social Media

1 Introduction

Recently, with the extensive amount of data flowing through social media platforms, the interest in information extraction from informal text has increased. Named entity recognition (NER), being one of the basic subtasks of Information Extraction, aims to extract and classify entity names from text. Extracted entities are utilized in applications involving the semantics of the content such as the topic of the text or location of the mentioned event.

The NER problem has been studied widely in the last decade and state-of-the-art algorithms achieve performance close to human on formal texts for English [13] and several other languages including Turkish [7, 8]. However, currently, existing NER algorithms face a generalization problem for textual data from social media. The recognition performance of state-of-the art algorithms degrade dramatically on English tweets as reported in [9] and similarly in Turkish tweets, forum text and speech data [4].

The main reason for the decrease in recognition performance of NER algorithms is the informal and noisy nature of social media text [9]. Social media text comprises spelling errors, incorrect use of punctuation, grammar and capitalization. NER algorithms fail to adapt to this new genre of text because algorithms are designed for formal text and are based on features present in well-formed text yet absent in social media. Commonly used features by NER algorithms are existence in a gazetteer, letter cases of words, Part of Speech (POS) tags and morphological substructures of words. Quality of these features degrade on social

media. For instance, a single misspelled character in an entity name causes the 'existence in gazetteer' feature to become invalid. Moreover, misspellings cannot be tolerated by POS tagging and morphological analysis tools.

Besides the peculiarities in text structure, types of entities and context in social media text differ from the newswire text commonly used for training NER systems [9]. For instance, the most common person names mentioned in newswire text are politicians, businessman and celebrities whereas social media text contains names of friends, artists, fictional characters from movies [9]. Similarly, names of local spots and cafes are common in social media text whereas larger geographical units like cities, districts, countries are frequent in newswire text [9].

Currently, normalization and domain adaptation are the two basic methods for adapting existing NLP systems to social media text [11]. Normalization is integrated to NLP systems as a pre-processing step. Normalization converts social media text by correct misspellings and removing noise so that existing NLP algorithms can perform better on the converted text. It improves recognition scores [9, 10], yet it is not able to address the issues of lack of context and different entity types mentioned previously. The other method, domain adaptation, update of existing systems for social media. In rule based systems resource extension or definition of new rules is required. Adaptation of machine learning models for NER requires both re-training of classifiers and re-design of features. Experiments from [9] have shown that re-training the Stanford CRF model [13] on social media data without feature re-design yields lower recognition performance. In addition, [4] reports that the performance of a CRF based model for Turkish on social media is increased when it is re-trained with capitalization excluded from the feature set.

Although the recognition accuracy is improved by both normalization and domain adaptation, they require considerable additional effort and yet it is not sufficient to approximate formal text performance. A system that can adapt to new genres should be designed to make the most of the syntactic and semantic context so that it can generalize to various uses of language.

In this study, we investigate the recognition performance of the *NLP from Scratch* method by Collobert et al. [6] on social media text. *NLP from Scratch* is a semi-supervised machine learning approach for NLP tasks. It includes an unsupervised learning step for learning word embeddings from a large unannotated text corpus. Word embeddings are representations for words in a vector space that can encode semantic similariites. Word embeddings can be used as features for representing words in classification problems. Previous work has shown that word embeddings are very powerful word features for several NLP tasks since they can encode the semantic similarity between words [6].

The semi-supervised *NLP From Scratch* approach for training NER classifiers has gained attention in recent years [34, 16] since it enables exploitation of huge amount of unannotated text that is produced on social media platforms like blogs, forums and microblogs like Twitter. To the best of our knowledge, this is

the first work to study the adaptation ability of the NLP from Scratch approach to social media through word embeddings.

In order to measure the recognition performance of the approach, we performed experiments on English and Turkish texts. We included several data sets from different platforms. Experiment results suggest that state-of-the art systems can be outperformed for morphologically rich languages such as Turkish by NLP from Scratch approach without normalization and any human effort for extending gazetteers. For English, an average ranking system can be obtained without any normalization and gazetteer extension effort.

The rest of the paper is composed of six sections. In Section 2, we discuss related work on NER on social media and present background information on the NLP from Scratch approach. In Section 3, we present how we applied the NLP from Scratch approach for social media. We report our experiment results in Section 4. We give a discussion on the results in Section 5 and conclude the paper with an overview in Section 6.

2 Related Work

2.1 NER on Social Media for English

State of the art NLP tools for English can achieve very high accuracy on formal texts yet their performance declines significantly on social media text. For instance, Ritter et al. reported [29] that F-measure score achieved by the ConNLL-trained Stanford recognizer drops from 86% to 46% when tested on tweets. Consequently, there are several recent studies [23, 20, 30, 22, 14] that focus on NER on tweets.

Normalization as a pre-processing step is widely studied [21, 15, 36, 19] to improve performance of not only NER yet several NLP tasks on social media. Regarding domain adaptation studies, TwitIE [3] is a version of the ANNIE component of the GATE platform tailored for tweets. NERD-ML [35] is an adaptation of NERD, a system which integrates the power of semantic web entity extractors into NER process.

Besides, there are solutions in the literature that adopt neither normalization nor domain adaptation. Ritter et al. [30] proposed a two step algorithm specific to tweets which first exploits a CRF model to segment named entities and then utilizes LabeledLDA to classify entities. Li et al. proposed TwiNER [20] which extracts candidate entities using external resources Wikipedia and Web-N Gram corpus and then performs random walk to rank candidates.

2.2 NER on Social Media for Turkish

There exists a considerable number of studies on NER on Turkish texts. Earlier studies are focused on formal texts whereas the recent studies focus on informal text specifically on the messages from the microblogging platform Twitter. For formal text, an HMM based statistical model by Tur et al. [33] and a rule-based

system by Kucuk et al. [18] are the earliest studies. A CRF based model by Seker et al. [7] and a semi-supervised word embedding based classifier by Demir et al. [8] are the state-of-the-art NER algorithms for Turkish. Both algorithms report performance close to human, 92 % on formal text.

Despite the high accuracy on formal text, a recent study reports that the CRF model has a recognition accuracy of 5% CoNLL F1 measure on tweets without any pre-processing or normalization [4]. There is also a dramatic decrease reported on forum and speech data sets. Previous studies that address NER on informal Turkish texts [4, 17] present normalization methods to fix misspellings. Dictionary extension is another proposed solution in [10]. Normalization of Turkish social media texts has been addressed by [10, 32, 1] in order to correct the spelling errors. The results are improved to at most 48% [10] on tweets with both dictionary extension or normalization which is still low compared to formal text and requires human knowledge for resource and rule base extension.

The NLP From Scratch approach has shown to be successful on Turkish formal texts [8]. The performance of the CRF based classifier can be approximated by the *NLP From Scratch* approach without any human effort for compiling gazetteers and defining rules [8].

2.3 Word Embeddings and NLP from Scratch

The NLP from Scratch approach was introduced by Collobert et al. [5]. In [6], word embeddings have been shown to be successful features for several NLP tasks such as POS Tagging, chunking, NER and semantic role labeling. Word embeddings are distributed representations for words [2]. A distributed representation of a symbol is a continuous valued vector which captures various characteristic features of the symbol. Word embedding representation comes as an alternative to one-hot representation for words which is a discrete high dimensional sparse vector. The idea behind word embeddings is to map all the words in a language into a relatively low dimensional space such that words that occur in similar contexts have similar vectors.

Although learning distributed word representations has been studied for a long time, the concept of word embeddings became notable together with the Neural Language Model (NNLM) by Bengio et al. [2]. Bengio et al.'s approach enables learning a language model and the word embeddings simultaneously. The NNLM is trained to predict the next word given a sequence of words. The model is a neural network placed on top of a linear lookup layer which maps each word to its embedding vector. This embedding layer is treated as an ordinary layer of a network, its weights are initialized randomly and updated with backpropagation during training of the neural network. The final weights correspond to the word embeddings. The neural network layers above the linear layer constitute the language model for predicting the next word.

Following Bengio's study, several other neural network models for learning word embeddings have been proposed. Neural language models such as the NNLM [2], the hierarchical probabilistic neural network language model [26],

the recurrent neural network model (RNNLM) [25] are complex models that require extensive computational resources to be trained. The high computational complexity of the models led to scalability concerns and the very recent word embedding learning methods like Skip-Gram [24] and Glove [28] lean on simpler and scalable models. For example, Skip-Gram shared within the Word2Vec framework enables training on a corpus of a billion words on a personal computer in an hour without compromising the quality of word embeddings measured on question sets [24].

The language models proposed for learning word embeddings adopt different neural network structures and training criteria. The model proposed by Collobert et al. [6], is trained to predict the center word given the surrounding symmetric context. The Skip-Gram model [24] utilized in this work aims to predict the surrounding context at the maximum distance of C from the center word. In addition, the model does not contain any non-linear layers and is considered as a log-linear model. In order to create training samples for the Skip-Gram model, R words from the future and history context are selected as the context where R is a randomly selected number in the range [1,C].

The most important property of the word embeddings is that the similarity between the words can be measured by the vector similarity between their embeddings. Word embedding based NLP classifiers can generalize much better owing to this property. For example, if a NER classifier has seen the sentence *I visited jennifer* with *jennifer* labeled as person name during training, it can infer that *kate* may be a person name in the sentence *I visited kate* since the word embeddings of *jennifer* and *kate* are similar.

3 NER From Scratch on Social Media

The NER from Scratch approach is composed of two steps:

1. A language model is trained on a large unannotated text corpus. Word embeddings and a language model are learned jointly during training.
2. A NER classifier is trained on supervised NER data where word embeddings obtained from Step 1 are leveraged as features for representing words.

This two step approach has shown to yield successful classifiers for several NLP tasks including NER on English, in [6]. Different algorithms and models can be selected for the two steps. In this study, for learning word embeddings we utilized the Skip-gram model and the Negative Sampling algorithm [24] within the Word2Vec framework. As the NER classifier model, we experimented with the Window Approach Network (WAN) of the SENNA, the NLP from Scratch framework proposed in [6].

Realization of these steps require a large corpus and supervised NER data. We applied three pre-processing steps to both unannotated corpus text and annotated NER data sets. First of all, the URLs in text are replaced with a unique token. Secondly, numbers are normalized by replacing each digit by "D" character, as typical in the literature [34]. Finally, all the words are lowercased meaning that capitalization feature is completely removed from the data.

3.1 Learning Word Embeddings

The Skip-Gram [24] model is a log-linear language model designed to predict the context of a word. The model takes a word as input and predicts words within a certain range C before and after the input word. The training process is performed on unsupervised raw textual data by creating training samples from the sentences in the corpus. For each word w in the sentence, the context to predict is the words that occur in a context window of size R centered around w where R is a randomly selected number in the range [1,C].

In this study, we obtained word embeddings of size 50 by training the Skip Gram model with negative sampling and the context range of 7. We obtained 50-dimensional embedding vectors using the open source implementation of Word2Vec.[1]

3.2 Training NER model

The NER classifier model is learned in the supervised learning step. As the NER classifier model, we experimented with the Window Approach Network (WAN) of the SENNA, the NLP from Scratch framework proposed in [6]. The WAN network is trained to predict the NER label of the center word given a context window. It is a typical neural network classifier with one hidden layer and a softmax layer placed on top of the output layer. The input layer is a context window of radius r. Word embeddings are used to represent words therefore a context window is represented by the vector obtained by concatenating the embedding vectors of words within the window.

The model is trained to map the context window vector to the NER label of the center word of the window. We trained the WAN network to minimize the Word-Level Log Likelihood (WLL) [6] cost function that considers each word independently. In other words, relationships between tags of the models is not considered for determination of the NER label.

In the experiments of this study, we trained a WAN network 350 input units that corresponds to a context window of size 7 and with 175 hidden layer units.

4 Experiments

We experimented on several English and Turkish data sets in order to measure the generalization of the NLP from Scratch approach on social media. English is an analytical language whereas Turkish is an agglutinative language. To be precise, grammatical structure of Turkish relies on morphemes for inflection. On the contrary, grammatical structure of English is based on word order and auxiliary words. We believe that experiments on these two languages from different paradigms is important to observe the performance of the proposed approach.

We measured the performance of our proposed approach with the CoNLL metric. It is considered as a strict metric since it accepts a labeling to be true

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

when both the boundary and the type of the named entity is detected correctly. For instance, the entity *patti smith* is considered to be recognized correctly only if the whole phrase is classified as a person entity. Details for the ConLL metric can be found in [27]. We report Precision, Recall and F1 values based on the CoNLL metric. We used the evaluation script from CoNLL 2003 Shared Task Language-Independent Named Entity Recognition [2] for computing the evaluation metrics.

4.1 Experiments on Turkish Texts

4.2 Data Sets

For the unsupervised word embedding learning phase, we compiled a large corpus by merging the Boun Web Corpus [31] and Vikipedi (Turkish Wikipedia)[3]. Boun Web Corpus was compiled from three newspaper web pages and a general sampling of Turkish web pages [31]. The merged corpus contains about 40 million sentences with 500 million tokens and a vocabulary of size 954K.

For evaluation of the proposed approach for NER, we experimented with six annotated NER datasets from the literature. Two of the data sets, namely **Formal Set 1** [33] and **Formal Set 2** [18], contain well-formed text compiled from newspaper resources. We included two Twitter NER data sets, namely *Twitter Set 1* [17] and *Twitter Set 2* [4] in experiments. Moreover, we experimented on the Forum Data Set and Speech Data Set from [4]. The number of entities per type in the data sets are given in Table 1.

Data Set	PER	LOC	ORG
Formal Set 1	16356	10889	10198
Formal Set 2	398	571	456
Twitter Set 1	458	282	246
Twitter Set 2	4261	240	445
Forum Data Set	21	34	858
Speech Data Set	85	112	72

Table 1. Number Of Named Entities Per Each Type In NER Data Sets

Twitter Set 1 includes 2320 tweets that are collected on July 26, 2013 between 12:00 and 13:00 GMT. The **Twitter Set 2** includes 5040 tweets. *Twitter Data Set 2* contains many person annotations embedded in hashtags and mentions whereas there are no such annotations in Twitter Data Set 1.

Forum Data Set [4] is collected from a popular online hardware forum.[4] It includes very specific technology brand and device names tagged as organization entities. Data contains many spelling errors and incorrect use of capitalization.

[2] `http://www.cnts.ua.ac.be/conll2002/ner/bin/`
[3] `https://tr.wikipedia.org`
[4] http://www.donanimhaber.com

Speech Data Set [4] is reported to be obtained via a mobile assistant application that converts the spoken utterance into written text by using Google Speech Recognition Service. As noted in [4], a group of people are asked to give relevant orders to the mobile application. Orders recorded by the application are included in the data set.

4.3 Experiment Results

In order to measure the adaptation performance of word embedding features, we trained the WAN network on the well-formed Formal Set 1 and reported results on the other data sets without re-training. We shuffled Formal Data Set 1 and divided it into three partitions for training, cross validation and test with percentages of 80%, 10% and 10% of the data set, respectively.

	Algorithm	PLO	PER	LOC	ORG
Formal Set 1 [33]	Seker et al.[7]	**91.94**	92.94	**92.93**	88.77
	Demir et al. [8]	91.85	**94.69**	85.78	**92.40**
	NER From Scratch	83.84	87.82	86.85	73.5
Formal Set 2 [18]	Kucuk et al. [18]	69.12	72.61	68.97	65.54
	NER From Scratch	**83.79**	**81.94**	**92.27**	**74.31**

Table 2. CoNLL F1 Scores on Turkish Formal Data Sets

In Table 2 and Table 3, we present F1 scores of the *NER From Scratch* approach in comparison with the highest scores reported in the literature, on the related data sets. F1 scores on the three type of entities PER, LOC and ORG, are given separately. In addition, the PLO column in the tables indicate the average F1 score over three types. The x signs in the tables indicate that F1 score is not reported for the entity type.

The *NER From Scratch* approach outperforms the rule-based system by Kucuk et al. on Formal Set 2 with a large margin. However, the proposed approach fails to outperform the previous studies on Formal Set 1. Both of the previous systems are machine learning classifiers. The CRF model of Seker et al. exploits gazetteers and capitalization as a feature which are powerful features for formal text. As mentioned previously, the *NER From Scratch* approach relies only on word embeddings. The other NER algorithm by Demir et al. uses word embeddings as features yet includes additional features such as affixes and word type as features.

The focus of our study is based on generalization to new platforms therefore a decrease in formal text performance is acceptable since *NER From Scratch* approach ignores many of the clues in formal text. It is also crucial to note that the CRF based model is reported to achieve at most 19% F1 score under CoNLL schema on tweets [4] with a normalization step. As a final note on the formal

	Algorithm	PLO	PER	LOC	ORG
Twitter Data Set 1 [17]	Kucuk et al.[17]	42.68	38.79	47.96	44.18
	Kucuk et al. [10]	48.13	39.60	67.17	44.16
	NER Pipeline+Normalization[10]	48.13	39.60	67.17	44.16
	NER From Scratch	**57.26**	**53.29**	**72.55**	**48.57**
Twitter Data Set 2 [4]	Celikkaya et al.[4]	15.27	x	x	x
	Kucuk et al. [10]	**36.63**	36.94	52.23	23.61
	NER From Scratch	15.43	10.23	**52.89**	**34.03**
Speech Data Set [4]	Celikkaya et al. [4]	50.84	x	x	x
	NER From Scratch	**71.54**	76.92	77.06	55.64
Forum Data Set [4]	Celikkaya et al. [4]	5.92	x	x	x
	NER From Scratch	**7.06**	4.9	17.11	6.6

Table 3. Results on Turkish Informal Data Sets

text performance, within the scope of this study, we measured the performance with a simple neural network classifier that considers only word context. The results by word embeddings can be further improved by a classifier that can incorporate the sentence level context and dependencies between NER tags like CRF models or the Sentence Approach Network (SAN) in [6].

In Table 3 we report the results on Turkish informal data sets. In Twitter Set 1, the *NER From Scratch* approach outperforms previous studies [17, 10] by Kucuk et al. and the NER Pipeline with Normalization [12]. In [17] performance of a multilingual rule-based NER system adapted to Turkish is reported. The NER system in [17] is improved by normalization and dictionary extension in [10]. The CoNLL F1 score of the ITU NLP Pipeline with Normalization that includes the CRF-based model [2] is obtained from [10].

On Twitter Set 1, the *NER From Scratch* approach outperforms the rule-based and CRF based solutions without any need for normalization, dictionary construction or extension. Many of the misspellings in tweets are tolerated and the misspelled words are recognized correctly due to their closeness to the original word in the embedding space. Most of the misspellings in Turkish tweets originate from replacement of Turkish characters with diacritics (ç,g,ı,ö,ş,ü) with non-accentuated characters (c,g,i,o,s,u). For instance, the words *besiktas* (original: beşiktaş, football team), *sahın* (original: şahin, male name) are recognized correctly. In addition, we have observed that the large amount of unannotated text can cover other types of misspellings such as single character replacement. The misspelled name *kılıçtaroğlu* (correct version: *kılıçdaroğlu* surname of a Turkish politician) could also be recognized correctly by the proposed approach.

The proposed solution achieves lower F1-score under CoNLL schema than that of the rule based system of Kucuk et al. [10] on Twitter Data Set 2. The *NER From Scratch* outperforms the rule based system on organization names and location names yet the low F1 score on person entity type leads to a lower PLO

F1 score. Majority of the entities in Twitter Set 2 are person names embedded in mentions. Approximately 80% of the person names occur in mentions. The proposed approach fails to recognize these person names since the training set does not contain any annotations embedded in mentions. For instance, the name *patti smith* exists as *@pattismith* in a mention. This situation causes the extent of the possible names space to be very large.

The *NER From Scratch* approach outperforms the CRF based model on the Speech Data Set and the Forum Data Set. However, the improvement by the proposed approach on the forum data is less than the improvement on the speech data. Forum data set is collected from a hardware forum [4] and includes very specific technology brand and device names tagged as organization entities. The training data set includes very formal organization names such as *Milli Eğitim Bakanlığı* (meaning Ministry of National Education) different from technology brand names. Despite the domain specific tags, *NER From Scratch* performed the best without any human effort for dictionary construction. In addition, proposed approach is able to recognize the brand names *steelsies* and *tp-link* that were seen neither in the corpus nor in the training set.

4.4 Experiments on English Texts

The recent survey on Twitter NER by Derczynski et al. [9] reports results on the performance of existing systems on three different Twitter data sets. We experimented on two of these data sets that are publicly available. The first data set is the data set from the study of Ritter et al. [29]. [5] The second data set is the data set from the Concept Extraction Challenge [6] in Making Sense of Microblog Posts (MSM) Worskhop in WWW 2013. We refer to this data set *MSM2013 Data Set* in the rest of the document. The Ritter data set contains tokenized content. On the other hand, we tokenized the MSM2013 data set using the Stanford Core NLP tokenizer. [7]

In this set of experiments, we utilized Wikipedia [8] as the corpus for learning word embeddings. We performed two different sets of experiments reported as *NER From Scratch (CoNLL2003)* and *NER From Scratch (CoNLL2003 + MSM 2013)* in Table 5. In the *NER From Scratch (CoNLL2003)* experiments, we used the original CoNLL 2003 data set for training. CoNLL 2003 NER Shared Task data set is used for training the From Scratch classifier. This data set is the commonly used for training supervised classifiers for English NER. In the *NER From Scratch (CoNLL2003 + MSM 2013)* experiments, we extended training data set by merging the CoNLL 2003 and MSM 2013 training sets and measured the performance of the classifier trained with this data set.

[5] `https://github.com/aritter/twitter_nlp/blob/master/data/annotated/ner.txt`

[6] `http://oak.dcs.shef.ac.uk/msm2013/ie_challenge/MSM2013-CEChallengeFinal.zip`

[7] `http://nlp.stanford.edu/software/corenlp.shtml`

[8] `http://www.wikipedia.org/`

Algorithm	ALL	PER	LOC	ORG	MISC
NER From Scratch (CoNLL2003)	51.46	66.28	42.52	23.95	1.9
NER From Scratch (CoNLL2003 + MSM 2013)	62.53	71.64	51.43	41.71	4.88
Bottom Score from [9] (Zemanta)	28.42	45.71	46.59	6.62	x
Top Score from [9] (NERD-ML)	**77.18**	**86.74**	**64.08**	**50.36**	x

Table 4. Results on the MSM 2013 Data Set, ConLL F1 scores

Algorithm	ALL	PER	LOC	ORG	MISC
NER From Scratch (CoNLL2003)	22.15	29.35	43.57	6.86	2.5
NER From Scratch (CoNLL2003 + MSM 2013)	34.75	49.15	49.78	21.16	0.76
Bottom Score from [9] (ANNIE)	22.46	24.81	40.23	16.00	0.00
Top Score from [9] (NERD-ML)	**51.49**	**71.28**	**61.94**	**32.73**	**23.73**

Table 5. Results on the Ritter Data Set, ConLL F1 scores

In Table 5, we present the CoNLL F1 scores of trained models in comparison with the top and bottom scores in the rank on related data sets from Derczynski et al.'s study [9]. It is crucial to note that we performed the exact category mapping in Derczynski et al.'s work for the Ritter data set so that the scores are comparable. In addition, reported MSM2013 results are the F1 scores computed on the test partition of the data set. Test partitions are not included in training.

Results on the MSM2013 show that the *NER From Scratch* classifier trained with only CoNLL 2003 data can achieve an average performance. Inclusion of tweets in the training data set improves the results with 20% on MSM 2013. This is expected since the training data set and test data set include similar tweets. A notable increase is observed in the Ritter set with inclusion of tweets.

The difference between the results obtained by the classifier trained with the formal training set and the expanded training set can be attributed to two issues. First of all, the corpus for embedding learning contains only Wikipedia text yet it cannot cover any misspellings or informal abbreviations that are found in social media. Secondly, there is a 10 year gap between the training data and the test data sets in the first setup. For instance, companies like Facebook that did not exist in 2003 are tagged as organizations names in Twitter data sets. Although, entities unseen in the training set have word embeddings close to similar known entities, the training set should include samples with the same context.

5 Discussion

Gazetteers are important resources for NER algorithms. Existence of a word in a gazetteer is exploited for both rule-based and machine learning systems. We believe that word embeddings can encode gazetteers when supported with

training data. Entity names are clustered in the embedding space. Whenever any of the names is seen in training data, the model can make connections on new samples via the vector similarities in embedding space. In Table 6, the closest 5 words in embedding space are given for the person name *kazım* and the city name *niğde*. In addition, we included common misspellings of these names and their neighbors in the table. The misspelled versions have similar neighbors. This implies that a gazetteer which can also cover misspelled words can be obtained without any human effort by Word2Vec word embeddings. This is one of the major reasons that can explain the success of the *NER From Scratch* approach on social media NER. Misspelled words and abbreviations occur in similar contexts with the original word.

kazım (PER)	**kazim (PER)** (misspelled)	**niğde (LOC)**	**nigde (LOC)** (misspelled)
alpaslan *(male name)*	cuneyt *(male name)*	balıkesir *(city)*	diyarbakir *(city)*
esat *(male name)*	kursat *(male name)*	adıyaman *(city)*	izmir *(city)*
tahir *(male name)*	namik *(male name)*	kırşehir *(city)*	amasya *(city)*
suphi *(male name)*	yucel *(male name)*	çorum *(city)*	asfa
nurettin *(male name)*	ertugrul*(male name)*	nevşehir*(city)*	balikesir *(city)*

Table 6. Top-5 Neighbours wrt Turkish Word Embeddings

Besides the gazetteer effect, word embeddings can capture the similarity of the surrounding context of a word. Similar to entity names, words that co-occur with a celebrity name are also close in the embedding space. For instance, the words *album* and *single* are mostly referred together with singer names and these two words are located very closely in the embedding space.

Performance of the *NLP From Scratch* approach is affected by three major factors:

- Coverage of the corpus utilized for obtaining word embeddings
- Coverage of the training data sets
- The ability of the classifier model to capture syntactic and semantic context.

For the experiments on Turkish texts, the corpus utilized for learning embeddings contained both formal text from both newspaper web sites and ordinary web pages. Owing to the diversity of text sources, the NLP from Scratch classifier trained on formal text is able to outperform previous studies on Turkish data sets. However, in English experiments, the corpus contained only formal text from Wikipedia text. In this study, we improved performance of the *from scratch* approach on English by increasing the coverage of the training data set since obtaining a large enough Twitter corpus requires a long time. It is possible to investigate the effect of corpus expansion and compare with the effect of training set expansion, as a future work.

6 Conclusion

In this study, we investigated application of the *NLP from scratch* idea for solving the NER problem on social media. We obtained word embeddings by unsupervised training on a large corpus and used them as features to train a neural network classifier. We measured the adaptation ability of the model based on word embeddings to new text genres. We experimented on both English and Turkish data sets. The *NER from Scratch* approach is able to achieve an average performance on English Twitter data sets without any human effort for defining gazetteers and dictionary extension. We observed that the recognition performance of the approach can be improved by extending the training set with tweets. For Turkish, without any efforts for gazetteer construction or rule extension, state-of-the-art algorithms can be outperformed on the social media data sets.

Word embeddings are powerful word features yet there are some issues in social media that require special techniques. Entities embedded in mentions and hashtags cannot be discovered without a segmentation step prior to classification. Sentence and phrase hashtags are very common in tweets and they incorporate valuable information.

The classifier we utilized in this study is a simple model that exploits only word level context. As a future work, we plan to experiment with more complex models that can integrate sentence level context and dependencies betwwen NER tags. Besides, we would like to investigate from scratch normalization methods for social media.

References

1. K. Adalı and G. Eryiğit. Vowel and diacritic restoration for social media texts. In *5th Workshop on Language Analysis for Social Media (LASM) at EACL*, Gothenburg, Sweden, April 2014. Association for Computational Linguistics.
2. Y. Bengio, R. Ducharme, P. Vincent, and C. Janvin. A neural probabilistic language model. *J. Mach. Learn. Res.*, 3:1137–1155, Mar. 2003.
3. K. Bontcheva, L. Derczynski, A. Funk, M. A. Greenwood, D. Maynard, and N. Aswani. TwitIE: An open-source information extraction pipeline for microblog text. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing*. Association for Computational Linguistics, 2013.
4. G. Çelikkaya, D. Torunoğlu, and G. Eryiğit. Named entity recognition on real data: A preliminary investigation for turkish. In *Proceedings of the 7th International Conference on Application of Information and Communication Technologies, AICT2013*, Baku, Azarbeijan, October 2013. IEEE.
5. 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*, ICML '08, pages 160–167, New York, NY, USA, 2008. ACM.
6. R. Collobert, J. Weston, L. Bottou, M. Karlen, K. Kavukcuoglu, and P. Kuksa. Natural language processing (almost) from scratch. *Journal of Machine Learning Scratch*, 12:2493–2537, Nov. 2011.

7. G. A. Şeker and G. Eryiğit. Initial explorations on using crfs for turkish named entity recognition. In *In Proceedings of the 24th International Conference on Computational Linguistics, COLING 2012.*, Mumbai, India, 8-15 December 2012.
8. H. Demir and A. Ozgur. Improving named entity recognition for morphologically rich languages using word embeddings. In *13th International Conference on Machine Learning and Applications, ICMLA 2014, Detroit, MI, USA, December 3-6, 2014*, pages 117–122, 2014.
9. L. Derczynski, D. Maynard, G. Rizzo, M. van Erp, G. Gorrell, R. Troncy, J. Petrak, and K. Bontcheva. Analysis of named entity recognition and linking for tweets. *Information Processing & Management*, 51(2):32–49, 2015.
10. K. Dilek and R. Steinberger. Experiments to Improve Named Entity Recognition on Turkish Tweets. In *Proceedings of the 5th Workshop on Language Analysis for Social Media (LASM)@ EACL*, pages 71–78, 2014.
11. J. Eisenstein. What to do about bad language on the internet. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 359–369, Atlanta, Georgia, June 2013. Association for Computational Linguistics.
12. G. Eryiğit. ITU Turkish NLP web service. In *Proceedings of the Demonstrations at the 14th Conference of the European Chapter of the Association for Computational Linguistics (EACL)*, Gothenburg, Sweden, April 2014. Association for Computational Linguistics.
13. J. R. Finkel, T. Grenager, and C. Manning. Incorporating non-local information into information extraction systems by gibbs sampling. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, pages 363–370. Association for Computational Linguistics, 2005.
14. J. Gelernter and N. Mushegian. Geo-parsing messages from microtext. *Transactions in GIS*, 15(6):753–773, 2011.
15. B. Han, P. Cook, and T. Baldwin. Automatically constructing a normalisation dictionary for microblogs. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, EMNLP-CoNLL '12, pages 421–432, Stroudsburg, PA, USA, 2012. Association for Computational Linguistics.
16. M. Konkol, T. Brychcín, and M. Konopík. Latent semantics in named entity recognition. *Expert Systems with Applications*, 42(7):3470 – 3479, 2015.
17. D. Küçük, G. Jacquet, and R. Steinberger. Named entity recognition on turkish tweets. In N. Calzolari, K. Choukri, T. Declerck, H. Loftsson, B. Maegaard, J. Mariani, A. Moreno, J. Odijk, and S. Piperidis, editors, *LREC*, pages 450–454. European Language Resources Association (ELRA), 2014.
18. D. Küçük and A. Yazici. Named entity recognition experiments on turkish texts. In *Flexible Query Answering Systems*, pages 524–535. Springer, 2009.
19. C. Li and Y. Liu. Improving text normalization via unsupervised model and discriminative reranking. *ACL 2014*, page 86, 2014.
20. C. Li, J. Weng, Q. He, Y. Yao, A. Datta, A. Sun, and B.-S. Lee. Twiner: Named entity recognition in targeted twitter stream. In *Proceedings of the 35th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '12, pages 721–730, New York, NY, USA, 2012. ACM.
21. F. Liu, F. Weng, and X. Jiang. A broad-coverage normalization system for social media language. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers - Volume 1*, ACL '12, pages 1035–1044, Stroudsburg, PA, USA, 2012. Association for Computational Linguistics.

22. X. Liu, F. Wei, S. Zhang, and M. Zhou. Named entity recognition for tweets. *ACM Trans. Intell. Syst. Technol.*, 4(1):3:1–3:15, Feb. 2013.
23. X. Liu, S. Zhang, F. Wei, and M. Zhou. Recognizing named entities in tweets. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1*, HLT '11, pages 359–367, Stroudsburg, PA, USA, 2011. Association for Computational Linguistics.
24. T. Mikolov, K. Chen, G. Corrado, and J. Dean. Efficient estimation of word representations in vector space. *CoRR*, abs/1301.3781, 2013.
25. T. Mikolov, M. Karafiát, L. Burget, J. Cernockỳ, and S. Khudanpur. Recurrent neural network based language model. In *INTERSPEECH*, pages 1045–1048, 2010.
26. F. Morin and Y. Bengio. Hierarchical probabilistic neural network language model. In *AISTATS'05*, pages 246–252, 2005.
27. D. Nadeau and S. Sekine. A survey of named entity recognition and classification. *Lingvisticae Investigationes*, 30(1):3–26, 2007-01-01T00:00:00.
28. J. Pennington, R. Socher, and C. D. Manning. Glove: Global vectors for word representation. In *Proceedings of EMNLP*, 2014.
29. A. Ritter, C. Cherry, and B. Dolan. Unsupervised modeling of twitter conversations. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, HLT '10, pages 172–180, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics.
30. A. Ritter, S. Clark, Mausam, and O. Etzioni. Named entity recognition in tweets: An experimental study. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '11, pages 1524–1534, Stroudsburg, PA, USA, 2011. Association for Computational Linguistics.
31. H. Sak, T. Güngör, and M. Saraçlar. Turkish language resources: Morphological parser, morphological disambiguator and web corpus. In *Proceedings of the 6th International Conference on Advances in Natural Language Processing*, GoTAL '08, pages 417–427, Berlin, Heidelberg, 2008. Springer-Verlag.
32. D. Torunoğlu and G. Eryiğit. A cascaded approach for social media text normalization of Turkish. In *5th Workshop on Language Analysis for Social Media (LASM) at EACL*, Gothenburg, Sweden, April 2014. Association for Computational Linguistics.
33. G. Tür, D. Hakkani-Tür, and K. Oflazer. A statistical information extraction system for turkish. *Natural Language Engineering*, 9(02):181–210, 2003.
34. 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 association for computational linguistics*, pages 384–394. Association for Computational Linguistics, 2010.
35. M. Van Erp, G. Rizzo, and R. Troncy. Learning with the web: Spotting named entities on the intersection of nerd and machine learning. In *In Proceedings of the 3rd Workshop on Making Sense of Microposts*, pages 27–30. Citeseer, 2013.
36. Y. Yang and J. Eisenstein. A log-linear model for unsupervised text normalization. In *EMNLP*, pages 61–72, 2013.