ParsTwiNER: A Corpus for Named Entity Recognition at Informal Persian

MohammadMahdi Aghajani
Sharif University of Technology
mmaghajani@ce.sharif.edu

AliAkbar Badri
Amirkabir University of Technology
aabadri@aut.ac.ir

Hamid Beigy
Sharif University of Technology
beigy@sharif.edu

Abstract

As a result of unstructured sentences and some misspellings and errors, finding named entities in a noisy environment such as social media takes much more effort. ParsTwiNER contains about 250k tokens, based on standard instructions like MUC-6 or CoNLL 2003, gathered from Persian Twitter. Using Cohen's Kappa coefficient, the consistency of annotators is 0.95, a high score. In this study, we demonstrate that some state-of-the-art models degrade on these corpora, and trained a new model using parallel transfer learning based on the BERT architecture. Experimental results show that the model works well in informal Persian as well as in formal Persian.

1 Introduction

Identifying named entities involves finding a sequence of words that refer to a specific concept, such as a person, a place, or an organization. This task is one of the essential components for other NLP tasks such as information extraction (Tjong Kim Sang and De Meulder, 2003). When masking a named entity in a sentence, the context helps to identify the entity type. For example, in the sentence "Doctor X says...", we find that the masked symbol "X" represents a person. On the other hand, sometimes a specific phrase refers to an entity, like "The United Nations." Hence, it is necessary to analyze word sequences to capture the context and the structure of a sentence. Furthermore, recognizing named entities requires generating a sequence of tags. Therefore, the problem type is a sequence-to-sequence problem.

Often, informal languages contain a mess of words, misspellings, and grammar errors, which degrades the performance of models (Baldwin et al., 2015; Strauss et al., 2016). Accordingly, some pre-trained models based on formal corpora fail to perform well in this context. It also would be expensive to build a language model from scratch to resolve this issue. As a result, we can use some pre-trained language models on formal language and train an end-to-end model using transfer learning. These models become more robust against noisy environments, such as Twitter or other user-generated content.

In this paper, we begin by preparing a corpus derived from Persian Twitter, called ParsTwiNER. It has excellent diversity and coverage due to the variety of users and topics. Then, we use transfer learning to use knowledge from the Persian formal model, ParsBert Farahani et al. (2020), and train a new model using ParsTwiNER to overcome messy structure and misspellings. At the initial stages of training, we use a novel variation of data annealing for better learning.

The rest of the paper is arranged as follows. We describe related work in Section 2. Next, we present the ParsTwiNER corpus in Section 3 and explain corpus statistics and annotation quality in 3.2 and 3.3, respectively. Then, we describe the proposed approach in Section 4 and the paper is concluded in Section 5.

2 Related work

This section describes NER corpora for informal languages like Persian and English. We also discuss the NER methods for these languages.

2.1 NER corpora

In informal Persian, there are not many standard works around NER who have followed the instructions of CoNLL and MUC (Tjong Kim Sang and De Meulder, 2003; Grishman and Sundheim, 1996). Asgari-Bidhendi et al. (2021) proposed a corpus, ParsNER-Social, that crawled from only 10 Telegram channels and supported only three types of

Proceedings of the 2021 EMNLP Workshop W-NUT: The Seventh Workshop on Noisy User-generated Text, pages 131–136
November 11, 2021. ©2021 Association for Computational Linguistics

NER: person, location, organization, and miscellaneous. Because no suitable and standard dataset covers a wide range of informal Persian content, in this paper, we create a highly diverse corpus of informal Persian content.

In formal Persian, Shahshahani et al. (2018) introduced PEYMA dataset for named entity recognition. This dataset contains 302530 tokens and 41148 named entities. Poostchi et al. (2018) proposed ARMAN dataset, which consists of 2,5016 tokens and 75,091 entities, for the NER task. In many Persian named entity recognition papers, these two datasets have been employed.

In English, Ritter et al. (2011) proposed a Ritter-11 corpus. The corpus contains 2400 tweets and 34K tokens. For the shared task for WNUT2017, Derczynski et al. (2017) proposed a corpus. The dataset focuses on rare and unusual entities. There are six types of entities in the corpus: person, location, corporation, product, creative work, and group.

2.2 NER methods

As mentioned above, there are not many works in informal Persian. Asgari-Bidhendi et al. (2021) trained a model using the Mono-Lingual BERT conduct on the ParsNER-Social corpus and obtained an F1-score of 89.65%.

In formal Persian, Shahshahani et al. (2018) combined rule-based methods with statistical methods and obtained an F1-score 84% on PEYMA corpus. They used regular expressions in the rule-based part and conditional random fields in the statistical component. Hosseinnejad et al. (2017) only used the conditional random fields on a collected corpus consists of 13 types of named entities. Ahmadi and Moradi (2015) presented a method by combining rule-based techniques and hidden Markov models and evaluated using a corpus of 32K tokens. They obtained an F1-score of 85.93% on this corpus.

Zafarian et al. (2015) used semi-supervised learning using a dataset with small number of labeled samples. The tokens that are recognized with high probability at each step will be used as training tokens for the next step. Iteratively, these steps are repeated. PersoNER is a good model for formal Persian, which uses the BiLSTM-CRF (Poostchi et al., 2016). Lastly, the ParsBert model obtained an F1-score of 90.59% on PEYMA corpus using the BERT encoder for Persian language (Farahani et al., 2020).

von Däniken and Cieliebak (2017) proposed a method for improving the model's performance on the WNUT2017 shared task based on tweets by combining transfer learning and sentence-level features. They achieved an F1 score of 40.78%. Aguilar et al. (2018) proposed an approach that uses phonetics and phonology, word embeddings, and Part-of-Speech tags to overcome noisy data. On the WNUT2017 shared task, they improved the F1-score by 3.69%. In informal English, Gu and Yu (2020) improved F1-score from 66.53% to 69.69% on on Ritter-11 dataset using data annealing and BERT.

3 ParsTwiNER Corpus

We describe the proposed method to create the corpus in this section. In addition to annotation instructions, we present statistics about the corpus and annotation quality based on the agreement of annotators.

3.1 Annotation instructions

Twitter is widely used by Persian language users, so we choose Twitter to collect data in informal Persian. The Persian on Twitter is unstructured and has no formal grammar. This corpus is capable of capturing some irregular structures and misspelled words. We crawled 10053 ordinary Persian Twitter accounts to collect 10014 tweets that formed an informal raw text corpus. Ordinary accounts tweet in informal language more than news or official accounts. After that, we removed all emojis, links, user IDs, and hashtag sign. Next, we removed all tweets containing racist, offensive, and violent content. The final number of tweets was 7632. In the next step, we used Parsivar[1] to tokenize and normalize the data. We considered persons, organizations, locations, events, groups, and nations as named entities. Two authors of the paper annotated the data. Both of the annotators are experts in natural language processing, so the data got labeled precisely. To tag the corpus, we used the IOB format. The corpus was created through an agreement between annotators to annotate according to CoNLL and MUC instructions (Tjong Kim Sang and De Meulder, 2003; Grishman and Sundheim, 1996). If a non-named entity appears inside a named entity phrase, then all of those tokens are tagged as named entities. Furthermore, we do not consider titles like 'Doctor' and 'Professor' at

[1] https://github.com/ICTRC/Parsivar

Category	# of Tokens	# of Unique Tokens
Persons	9586	3932
Organizations	5122	1875
Locations	6676	1914
Events	1146	428
Nations	956	341
Political Groups	575	255
Total	24061	8727

Table 1: The number of tokens and unique tokens in the proposed corpus.

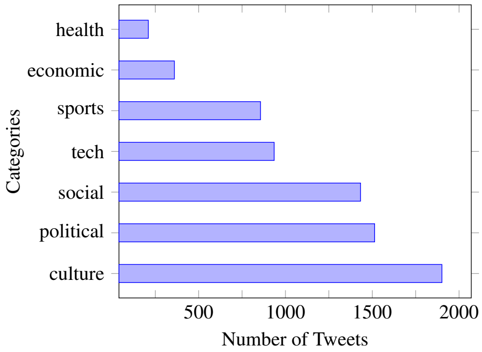

Figure 1: The number of tokens and unique tokens in the proposed corpus.

the beginning of tokens to be named entities. As part of these annotation instructions, a variety of items will be made public along with the corpus. The corpus and annotation instructions are publicly available on Github[2].

3.2 Corpus statistics

The corpus includes 7667 tweets, 232917 tokens, and 24061 named entities. In Table 1, a summary of the number of tokens and unique tokens in each category are given. Additionally, the average tweet length is 32 tokens. There are seven categories for tweets: political, economic, social, sports, health, culture, and technology. Figure 1 shows the number of tweets in each category. We split the corpus into training, test, and validation sets based on the distribution of categories and named entity types.

3.3 Annotation quality

As part of the investigation of the annotation quality, the annotators relabelled a fraction of the data. To determine the quality of the labeling, we check the differences between the annotators. As a result, we pick 9673 tokens from the corpus for relabeling. Finally, we found 76 differences. Mistakes occur when annotators misunderstand the word's meaning or context of the sentence. We calculate the degree of agreement at 99% when considering the survey data and the 76 differences between annotators. Due to a large number of **O** tokens, only 983 tokens are labeled entities, and so despite 76 differences, the level of agreement between the annotators is about 93%.

In addition, we use the Cohen's Kappa criterion to measure the level of agreement in a statistical sense on this test set (Cohen, 1960). This criterion determines the degree of agreement based on the distance of the annotators from the random tagging mode. Equation (1) shows Cohen's Kappa criterion

$$\kappa = \frac{p_0 - p_e}{1 - p_e} \tag{1}$$

where p_0 represents an annotator's agreement on every test instance, and p_e indicates the amount of agreement in which the tagging was done randomly. The criterion lies between $[-1, +1]$. A higher value indicates a better agreement among the annotators. The Kappa criterion for the corpus is 0.95, which implies high agreement among annotators.

4 Proposed model

We use transfer learning to design a named entity recognizer in informal Persian. Current state-of-the-art models, such as ParsBert, were trained on the formal corpus, and their performance degrades on the collected corpus. This degradation is due to unstructured and noisy grammar in informal Persian. As a first step, we fine-tune the ParsBert language model on the corpus using sequential transfer learning. Then, we use parallel multi-task learning to train a model in informal and formal Persian simultaneously. Experiments show the effectiveness of multi-task training.

4.1 Sequential transfer learning

It is expensive to train a language model from scratch. Thus, we fine-tune the ParsBert language model to adapt to informal Persian. We use BERT basic architecture with 768 hidden states, 12 transformer encoder layers, and 12 multi-head attention layers. Since the named entity recognition problem is a multi-class classification, we use a fully connected feed-forward layer above BERT to classify entities based on 13 classes based on the corpus. This dataset has six types of named entities. Using the IOB format, each type of named entity has two labels and one **O** label for non named entity tokens. Hence, we have 13 classes.

[2] https://github.com/overfit-ir/parstwiner

We train this model by placing the [CLS] token at the beginning and the [SEP] token at the end of each tweet. We also set all segment embedding to 1. With this configuration, BERT treats tweets like sentences. We then set the learning rate to $5 * 10^{-5}$ at the initial step and scheduled it to reduce linearly over time. For overfitting prevention, we use dropout with the rate of 0.1. Table 2 shows the results of the evaluation compared with ParsBert. Training the model in this manner does not leverage embedded knowledge in the ParsBert NER model. Using the above approach, we only rely on the ParsBert language model. In the ParsBert NER model, we exploit knowledge through parallel transfer learning.

4.2 Parallel transfer learning

To train an informal Persian NER model in conjunction with a formal Persian NER model, we share a BERT encoder between two models and use two different task heads as a fully connected feedforward classifier for each model. Figure 2 shows the overall architecture of the proposed parallel transfer learning method. We first use PEYMA dataset to train the formal Persian model, and then ParsTwiNER corpus to train the informal Persian model.

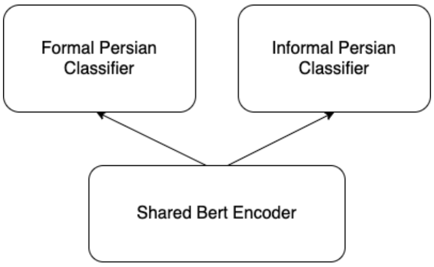

Figure 2: Parallel transfer learning architecture.

We feed eight tweets or sentences per training step to the BERT encoder. To train in this manner, we need to select either ParsTwiNER or PEYMA to supply data to the model at each training step. We select a dataset with a probability and then feed the selectd dataset to the BERT. We apply this policy using a novel variation of data annealing due to the difference in task classifiers. We can not mix PEYMA and ParsTwiNER data for data feeding because the backpropagated gradient for a selected corpus must affect the corresponding task head classifier. Initially, we choose PEYMA for data feeding with a higher probability than ParsTwiNER. Then, at each step, we reduce the probability of selecting the PEYMA corpus by the rate of λ. Equation (2) shows the schedule of the selection probability for each corpus.

Entity Type	ParsTwiNER			ParsBert
	Precision	Recall	F1	F1
PER	85%	91%	87%	80%
LOC	84.3%	85.5%	84.9%	68%
ORG	61.8%	62.7%	62.2%	55%
EVE	30%	28.5%	29.2%	12%
POG	36%	40.9%	38.3%	-
NAT	60.5%	76.6%	67.6%	-
Total	76%	80%	78%	69.5%

Table 2: The results of sequential transfer learning. ParsBert does not support POG and NAT. Results are in the exact-match mode, and the total is micro-averaged.

$$\begin{aligned} p_S^t &= \alpha\lambda^{t-1}, 0 < \alpha < 1, 0 < \lambda < 1 \\ p_T^t &= 1 - \alpha\lambda^{t-1} \end{aligned} \tag{2}$$

where p_S^t shows how likely it is that in step t of the training, the whole batch would be from formal Persian (PEYMA dataset), and p_T^t shows how likely it is for informal Persian (ParsTwiNER dataset). Table 3 shows the results of the model trained by using the parallel transfer learning compared with ParsBert.

As depicted in Table 3, we achieve about 11% improvement on the corpus by the proposed method. Also, we evaluated the model's performance on other formal Persian datasets and realized that the model's performance is competitive to ParseBert. Previous models suffer from the lack of generalization, but our model has a great performance on ParsTwiNER dataset and an acceptable result on the formal datasets. The source code is publicly available on Github[3] and the trained models are available for functionality test on HuggingFace[4].

5 Conclusions

We demonstrated that the performance of existing NER models like ParsBert degrades when applied to informal Persian. A crawled Persian Twitter corpus, named ParsTwiNER, was used for experiments. Based on some corpus statistics, we showed that the corpus was built using various standard instructions, such as MUC and CoNLL 2003. Using

[3] https://github.com/overfit-ir/parstwiner

[4] https://huggingface.co/overfit/twiner-bert-base-mtl

Entity Type	ParsTwiNER			ParsBert
	Precision	Recall	F1	F1
PER	90%	91%	91%	80%
LOC	81%	83%	82%	68%
ORG	70%	69%	69%	55%
EVE	35%	50%	41%	12%
POG	94%	77%	85%	-
NAT	68%	93.3%	82.3%	-
Total	80%	83%	81.5%	69.5%

Table 3: The results of parallel transfer learning. ParsBert does not support POG and NAT. Results are in the exact-match mode, and the total is micro-averaged.

parallel transfer learning, we proposed an approach to train a model that performed as well as possible in both formal and informal Persian. Experimental results indicate that the proposed model recognizes events with lower accuracy than other categories. For future works, we will investigate the causes of this quality degradation and finding a solution. Also, we will check the quality of the trained models on named entities that did not exist at all in the training phase.

Acknowledgements

The authors would like to thank Dr. Momtazi for her support. Furthermore, we would like to acknowledge the accompaniment provided by Mohammad Mahdi Samiei and Abbas Maazallahi.

References

Gustavo Aguilar, Adrian Pastor López-Monroy, Fabio González, and Thamar Solorio. 2018. Modeling noisiness to recognize named entities using multitask neural networks on social media. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1401–1412, New Orleans, Louisiana. Association for Computational Linguistics.

Farid Ahmadi and Hamed Moradi. 2015. A hybrid method for persian named entity recognition. In *2015 7th Conference on Information and Knowledge Technology (IKT)*, pages 1–7.

Majid Asgari-Bidhendi, Behrooz Janfada, Omid Reza Roshani Talab, and Behrouz Minaei-Bidgoli. 2021. Parsner-social: A corpus for named entity recognition in persian social media texts. *Journal of AI and Data Mining*, 9(2):181–192.

Timothy Baldwin, Marie Catherine de Marneffe, Bo Han, Young-Bum Kim, Alan Ritter, and Wei Xu. 2015. Shared tasks of the 2015 workshop on noisy user-generated text: Twitter lexical normalization and named entity recognition. In *Proceedings of the Workshop on Noisy User-generated Text*, pages 126–135, Beijing, China. Association for Computational Linguistics.

Jacob Cohen. 1960. A coefficient of agreement for nominal scales. *Educational and Psychological Measurement*, 20(1):37–46.

Leon Derczynski, Eric Nichols, Marieke van Erp, and Nut Limsopatham. 2017. Results of the WNUT2017 shared task on novel and emerging entity recognition. In *Proceedings of the 3rd Workshop on Noisy User-generated Text*, pages 140–147, Copenhagen, Denmark. Association for Computational Linguistics.

Mehrdad Farahani, Mohammad Gharachorloo, Marzieh Farahani, and Mohammad Manthouri. 2020. Parsbert: Transformer-based model for persian language understanding. *CoRR*, abs/2005.12515.

Ralph Grishman and Beth Sundheim. 1996. Message Understanding Conference- 6: A brief history. In *COLING 1996 Volume 1: The 16th International Conference on Computational Linguistics*.

Jing Gu and Zhou Yu. 2020. Data Annealing for Informal Language Understanding Tasks. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 3153–3159, Online. Association for Computational Linguistics.

Shadi Hosseinnejad, Yasser Shekofteh, and Tahereh and Emami Azadi. 2017. A'laam corpus: A standard corpus of named entity for persian language. *Signal and Data Processing*, 14(3).

Hanieh Poostchi, Ehsan Zare Borzeshi, Mohammad Abdous, and Massimo Piccardi. 2016. PersoNER: Persian named-entity recognition. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 3381–3389, Osaka, Japan. The COLING 2016 Organizing Committee.

Hanieh Poostchi, Ehsan Zare Borzeshi, and Massimo Piccardi. 2018. BiLSTM-CRF for Persian named-entity recognition ArmanPersoNERCorpus: the first entity-annotated Persian dataset. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Alan Ritter, Sam Clark, Mausam, and Oren Etzioni. 2011. Named entity recognition in tweets: An experimental study. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 1524–1534, Edinburgh, Scotland, UK. Association for Computational Linguistics.

Mahsa Sadat Shahshahani, Mahdi Mohseni, Azadeh Shakery, and Heshaam Faili. 2018. PEYMA: A tagged corpus for persian named entities. *CoRR*, abs/1801.09936.

Benjamin Strauss, Bethany Toma, Alan Ritter, Marie-Catherine de Marneffe, and Wei Xu. 2016. Results of the WNUT16 named entity recognition shared task. In *Proceedings of the 2nd Workshop on Noisy User-generated Text (WNUT)*, pages 138–144.

Erik F. Tjong Kim Sang and Fien De Meulder. 2003. Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition. In *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003*, pages 142–147.

Pius von Däniken and Mark Cieliebak. 2017. Transfer learning and sentence level features for named entity recognition on tweets. In *Proceedings of the 3rd Workshop on Noisy User-generated Text*, pages 166–171, Copenhagen, Denmark. Association for Computational Linguistics.

Atefeh Zafarian, Ali Rokni, Shahram Khadivi, and Sonia Ghiasifard. 2015. Semi-supervised learning for named entity recognition using weakly labeled training data. In *2015 The International Symposium on Artificial Intelligence and Signal Processing (AISP)*, pages 129–135.