IJICTR
International Journal of Information & Communication Technology Research

Volume 13- Number 2 – 2021 (39 -48)

Improving Persian Named Entity Recognition Through Multi Task Learning

Mohammad Hadi Bokaei[*]
Information Technology Institute
Telecommunication research
Center, Tehran, Iran
mh.bokaei@itrc.ac.ir

Mohammad Nouri
Information Technology Institute
Telecommunication Research
Center, Tehran, Iran
nourimohammad.92@gmail.com

Abdolah Sepahvand
Information Technology Institut
Telecommunication Research
Center, Tehran, Iran
a.sepahvand@aut.ac.ir

Received: 17 April 2021 - Accepted: 14 May 2021

Abstract—**Named Entity Recognition is a challenging task, specially for low resource languages, such as Persian, due to the lack of massive gold data. As developing manually-annotated datasets is time consuming and expensive, we use a multitask learning (MTL) framework to exploit different datasets to enrich the extracted features and improve the accuracy of recognizing named entities in Persian news articles. Highly motivated auxiliary tasks are chosen to be included in a deep learning based structure. Additionally, we investigate the effect of chosen datasets on performance of the model. Our best model significantly outperformed the state of the art model by 1.95%, according to F1 score in the phrase level.**

Keywords—Named-Entity Recognition; Deep Learning; Multi-Task Learning; Persian Language; Low-recourse Languages

I. Introduction

Named Entity Recognition (NER) is an important upstream task in Natural Language Processing aimed to recognize named entities and classify them into pre-defined categories, such as persons, organizations, locations, etc. NER has a wide range of applications in NLP, namely in Machine Translation, Information Extraction, and Question Answering.

In NER, several important and difficult issues need to be addressed, such as ambiguous words, abbreviations, spelling variations, foreign words, the structures of coordination, shortened names, etc. These challenges are common in all languages. Nevertheless, owing to Perso-Arabic writing system, Persian language has additional problems, including the lack of some important clues, like capitalization patterns and diacritic, which result in ambiguous words. Furthermore, the lack of hand-annotated data is viewed as another issue in tagging named entities in low-resource languages like Persian.

In order to correctly recognize the named entities within a given document, one must identify the correct boundaries of entities first. In the Persian language, identifying the correct boundaries of an entity is highly related to other tasks, like part of speech tagging as well as detecting Ezafe (a morpheme, which is pronounced but usually is not written and links head words to its modifiers in a noun phrase).

[*] Corresponding Author

[DOI: 10.52547/itrc.13.2.39]

In this article, we tackle the problem of NER for the Persian language. There are specific small-size datasets for this task in Persian, albeit with different tagging schema. There are also other datasets for related auxiliary tasks, such as POS and Ezafe. Here is the main question: "Is there any way to get the most out of these datasets to solve our main problem which is NER?" An obvious answer to this question can be "multitask learning" which is shown to be a very good approach to solve many problems, including NER, in other languages [1, 2, 3].

Recently, a shared task has been defined for Persian NER [4], and several models have been evaluated according to two shared test sets. We regard the best performed models in that competition as the baseline of our work and propose a model to improve them.

The main contribution of this work is as follows:

1. We propose a model based on MTL to tackle the problem of NER in the Persian language. As far as we know, there is no other work that evaluates the efficiency of MTL in the Persian language NER. Our best model outperforms the state of the art models significantly by 1.95%, according to F1 score in the phrase level.
2. The proposed model is trained via two publicly available NER datasets with different tagging schema. The resulted model outperforms the state of the art models. Additionally, we study the effectiveness of adding more auxiliary tasks to the training phase. We show that adding more auxiliary tasks can improve the overall performance, but not significantly.
3. We finally study the source of weaknesses and strengths of our proposed model compared to other baselines and pave the way for researchers to further improve the results.

The structure of this article is as follows: In Section 2, previous work on multitask learning and its effectiveness in NER is reviewed. A brief overview of NER for the Persian language is also provided in this section. Section 3 introduces the architecture of the proposed model. In Section 4, the experimental reports are presented, including the datasets, evaluation metrics and results as well as detailed error analysis of the proposed model in compared to the previous state of the art ones. Finally, Section 5 concludes the paper and suggests future works.

II. Background

In this section, we first review the multitask learning technique and its effectiveness in different tasks specifically in NER. Then, a brief overview of previous works on Persian NER is presented.

A. *Multitask Learning for Named Entity Recognition*

In Machine Learning (ML), the main goal is to obtain a model according to the provided training data, which has a good generalization capability and achieves an acceptable performance on different test sets (but extracted from the same distribution). Multitask learning sets out to improve the performance of a target (main) task, using the information extracted from related (auxiliary) tasks. This technique can be considered as a good solution for specific tasks with limited training data [5, 6].

There are plenty of fields in machine learning that use MTL to improve the performances of algorithms, consisting of computer vision [7, 8], bioinformatics and health informatics [9, 10], web search ranking [11, 12], etc. In natural language processing, this framework has been used in different tasks such as text classification [13, 14, 15], semantic representation and semantic parsing [16], machine translation [17, 18], speech recognition [19], and sequence tagging [20, 21], to name but a few.

Named entity recognition is a challenging task that has been extensively studied in the literature. There are plenty of algorithms proposed to do the task including earlier methods, such as Hidden Markov Models (HMM) [22], Decision Trees [23], Maximum Entropy [24], and Conditional Random Fields (CRF) [25, 26]. However, the development of deep learning has yielded a state-of-the-art performance in NLP tasks including, NER systems in English [27, 1, 28, 29, 30] and other languages, such as Portuguese [31], German [32], Indonesian [33], Indian [34], etc.

Some studies have focused on the effectiveness of MTL techniques in NER tasks. As an early study, [35] defines a general single neural network architecture suitable for different tasks, including POS tagging, chunking, NER, and semantic role labeling. All tasks are jointly learned using a weight-sharing strategy.

[25] proposes a joint model of parsing and NER. The model is composed of three models, namely the NER model in which a semi-CRF is used to segment and label name entities simultaneously, the parsing model using a CRF-based context-free grammar parser (CRF-CFG), and the joint model that requires jointly-annotated data. The proposed model uses single-task annotated data as additional information to improve the performance of a model for jointly learning two tasks over five datasets.

The authors in [36] proposed a model which obtains the first position in the 3rd Workshop on Noisy User-generated Text (WNUT-2017) [37]. Their model uses multitask learning framework in which the main NER task and an auxiliary but related secondary task called NE segmentation (i.e. finding the boundary of entities) are used simultaneously to train the model.

In order to address the limited availability of labeled training data in a special purpose NER tasks, [38, 39] investigated the benefits of MTL to biomedical NER. [38] investigates the performance of two MTL architectures using the information in two related tasks: POS and NER. In the first architecture, shared features are extracted and fed into the output layers which are separated for each task the model learns. In the second architecture the model is first trained on the auxiliary task (POS tagging) and then the trained model is used in the training of the main task (NER) by concatenating the output of the fully connected layers of both tasks.

[DOI: 10.52547/itrc.13.2.39]

[40] puts forward two novel techniques, namely Multitask Data Selection and Constrained Decoding using Knowledge Base, to improve the BiLSTM-CRF architecture for entity recognition system, proposed by [28]. Multitask Data Selection ensures the homogeneity between auxiliary and main tasks by filtering out instances with different distribution. On the other hand, the goal of the second technique is to use the document level information in the decoding time.

[41] proposes a multi-lingual multitask architecture of POS tagging and NER tasks to low-resource languages. They jointly train models using a parameter sharing method and then share character embeddings between (Spanish and English) languages and mix both different languages corpora to train word-embeddings. In the LSTM layer, each word and its context is encoded to a vector to be passed to the final (CRF) layer which is shared across languages.

Nearly all of the previous works consider a shared hidden layer and a separated output layer (either CRF or softmax) for the main and auxiliary tasks. Another example of this approach is [42]. In our work, we consider another approach in which shared features are extracted explicitly and used for the tagging purpose, like the one proposed in [13] for the classification task. We talk more about the model in Section 3.

B. *NER in Persian*

A few studies of Persian NER have been conducted including rule-based methods, statistical methods, hybrid methods and deep learning methods. As one of the earliest studies of Persian NER, [43] uses rule-based methods and gazetteers, in which morphological analyses and some heuristics are used to recognize NEs. [44] also presents a dictionary-based recognizer to detect named entities. To create a dictionary of named entity, they use Bijankhan corpus [45] as well as Wikipedia.

As a hybrid research, [46] combines the rule-based method (using a gazetteer) and a HMM model to recognize NEs including the names of people, locations and organizations. [47] develops a NER system to extract the names of people, locations and dates. They utilize linguistic grammar rules, a gazetteer containing 2500 entries and as well as a trained SVM.

In another work, [48] introduces a named-entity annotated dataset called ArmanPersoNER corpus including six categories of NER tags, namely person, organization, location, facility (like universities, research center etc.), product (including TV shows, movies, newspaper etc.) and event (such as wars, earthquakes, national holidays etc.). This corpus alongside Hellinger PCA word vectors are used to train three models: CRF, SVM-HMM and RNN-based models. Owing to the low-size of the annotated data, the experiments show that the F1 score of SVM-HMM is higher than that of the deep learning model. Another work on statistical methods is [49] in which a NE corpus called A'laam corpus is introduced, which contains 250,000 tokens annotated with 13 NE tags. a simple CRF model is trained and evaluated on this corpus.

In the field of deep learning, two simultaneous work are presented [50, 51] with somehow similar network structures. Both of the works uses Bi-LSTM and CNN structures and extract feature sets in the word and character levels for each word in a given sentence. Extracted features are flattened and then fed into a fully connected network with one hidden layer. Finally, a CRF output layer is used to calculate the probability distribution over NER tags.

In the most recent work, a shared task is defined in the NSURL2019 workshop [4] [1] and several algorithms are evaluated and ranked according to two different test sets. Both of the best two models, namely MorphoBERT and Taheri&Hosseini, use a similar network structure. They use the BERT model [52] for training a highly accurate representation of Persian tokens. These word embeddings are used by a BiLSTM network. Finally, a CRF layer is used to tag the words in the input sentence. We consider these state-of-the-art works as our baselines and try to improve them using multitask learning techniques.

III. Model

The overall structure of the proposed model is depicted in Fig. 1 [2]. The architecture of this model is inspired by [53] which adopts the feature learning approach to improve the performance of a classification task. The details of the model is discussed in the next paragraphs.

The input of the model is a sentence containing n words. At the first layer, a representation of each word must be extracted to be fed to the next layers. We have used FastText word embedding with 300 dimensions and a window of size 10 [54]. We also use a CNN to model character sequence inside a word to better handle out-of-vocabulary words. The architecture of the implemented CNN is depicted in Fig. 2. In this figure w_i^j is the j-th character in the i-th word. Characters are first passed to a dynamic character embedding look-up table which is initialized randomly and then tuned in the training phase. The embedding of the characters constructs a $M * chel$ image in which M is the number of characters in the given word and $chel$ is the character embedding size. The image is then passed to a convolution layer which consists of the NF number of filters with size FS. The output of the convolution layer is finally passed to a max pooling layer and the final character-based representation vector for the given word is extracted. The embedding vector and the output of the CNN model are concatenated and form the final feature vector of each word in the input sentence.

The next layer in the proposed model has three separate parts. Two parts are devoted to the main and the auxiliary tasks separately and the third part is shared between them. In the training phase the word representation is passed to these parts based on the task

[1] http://nsurl.org/tasks/task-7-named-entity-recognition-ner-for-farsi/

[2] This figure shows the model with a main task and just one auxiliary task. The extension of the model to more than one auxiliary tasks is discussed later.

[DOI: 10.52547/itrc.13.2.39]

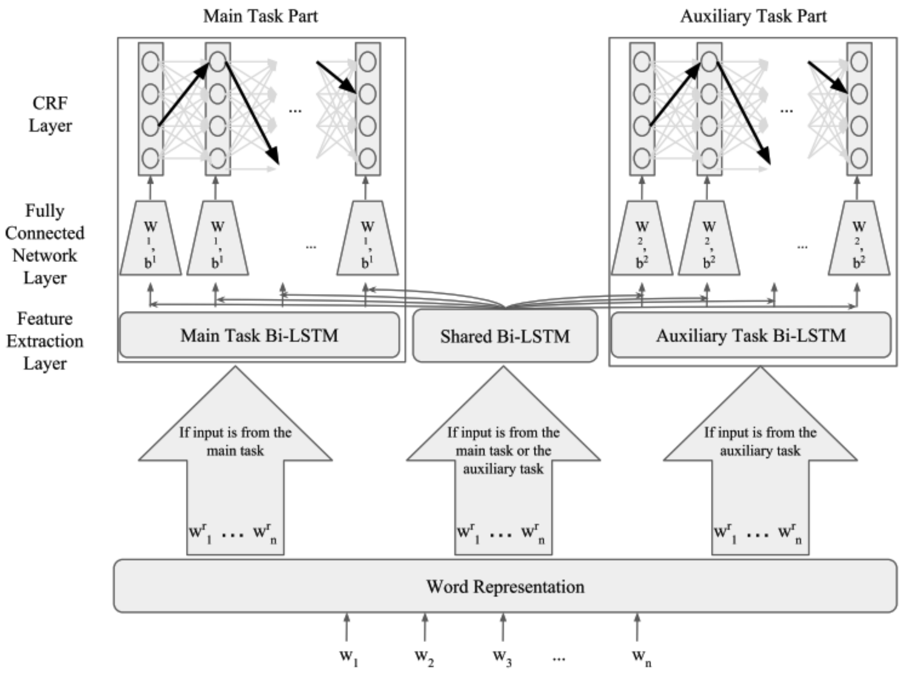

Figure 1. The overall structure of the proposed model..

of the input sentence: If the input sentence is transmitted from the main (or auxiliary) task, the extracted word representation is passed through the main (or auxiliary) part as well as the shared part. Accordingly, the sentences from each task train the

In the following, we assume that we are in the train phase and the input to the model is from the main task: Both feature vectors extracted from the main part BiLSTM and the shared BiLSTM are concatenated for each word and fed to a fully connected network layer. The logits are calculated in this layer according to the Equation 1.

$$logits = W_1.F + b_1 \qquad (1)$$

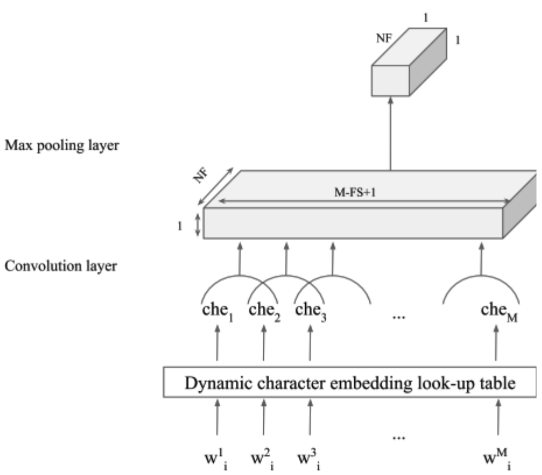

Figure 2. The architecture of the Convolutional Neural Network to extract the word representation.

parameters in their respected parts as well as the ones in the shared part. All parts are implemented using a BiLSTM netowrk as shown in Fig. 3.

Where W_1 and b_1 are the parameters for the fully connected layer in the main part and are shared across all words. F is the concatenated feature vector. The logits are then passed to a CRF layer in order to find the global best tag sequence for the input sentence according to the Equation 2.

$$s(y_1, \dots, y_n) = \sum_i logits(y_i) + trans(y_i, y_{i-1}) \qquad (2)$$

Where $s(y_1, \dots, y_n)$ is the score of the tag sequence $y_1, \dots, y_n$ which is assigned to the input sentence, $logits$ are calculated according to the Equation 1 and $trans(y_i, y_{i-1})$ is the transition score of going from y_{i-1} to the y_i, which are trained alongside other parameters in the training phase. Finally, the found tag sequence is compared with the provided gold tag and the error is back-propagated in the network to tune the parameters.

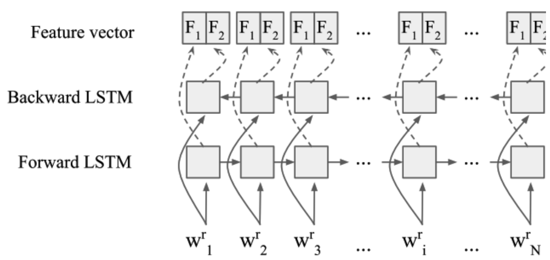

Figure 3. The architecture of the bilstm network to extract features from each word representation.

[DOI: 10.52547/itrc.13.2.39]

What is said above can be extended to the case where the input sentence is from the auxiliary task trivially. According to this procedure, the main and auxiliary parts are trained based on the respective sentences, but the shared part is trained with sentences in both tasks and can represent concrete features related to both tasks. In the test phase, the auxiliary part is removed since we are only interested in the results of the main task.

Extending the model to more than one auxiliary task is straightforward. The shared BiLSTM is shared between the main task as well as all auxiliary tasks. The auxiliary part is also duplicated for each added task. In the training phase all sentences from all tasks contribute to enrich the features extracted from the shared BiLSTM. In the testing phase all auxiliary parts are removed and the main part and the shared part are kept to extract the best tag sequence for the input sentence.

In order to summarize the symbols used in this work, Table 1 shows all the hyper-parameters and their meanings. For each hyper-parameter, the value used in this work is also prepared in this table.

TABLE I. THE SUMMARY OF THE HYPER PARAMETERS USED IN THIS WORK AND THEIR VALUES.

Symbol	Meaning	Value
whs	word LSTM hidden size	300
wel	word embedding vector length	300
chel	character embedding vector length	100
NF	number of filters for each filter size	128
FSS	filter sizes set	[2,3,4,5,6]
epoch	number of epochs	100
dr	dropout rate	0.5
lr	learning rate	0.001

IV. EXPRIMENTS

In this section, we first talk about the experimental setup and the copora used to train and test the models. The results are then reported and compared with the single-task baselines. Finally, the error analyses of the proposed model are discussed in order to open the way for the future studies of the topic.

A. *Experimental Setup*

In order to evaluate the performance of the MTL model, two different NER corpora are used to train the model introduced in Section 3. The first one is the ITRC corpus which is considered as the main task. The corpus consists of 900K tokens with the tag set Person, Location, Organization, Date, Time, Money, and Percent. Table 1 summarizes the number of tokens and types of this corpus for each Named entity class. This corpus is available online [3].

Another public Persian NER corpus is ArmanPerso which is also available online[4]. This corpus contains 250,015 tokens in 7,682 sentences with NE tags in IOB format. Table 2 shows the number of tokens and the percentage of them for each entity class in the ArmanPerso corpus. The tagging schema in this corpus is different from the one used in ITRC corpus and covers Location, Organization, Person, Facility, Event and Product.

In order to evaluate the models we use the second test data in the NSURL-2019 report [4]. This corpus includes 416,642 words. The detailed information is summarized in Table 3.

TABLE II. ITRC CORPUS INFORMATION IN DETAILS

No.	Named Entity Tag	Tokens Absolute Frequency	Tokens Relative Frequency	Types Absolute Frequency	Types Relative Frequency
1	Location	20, 999	0.22	3, 245	0.2
2	Organization	34, 340	0.36	4, 211	0.26
3	Person	20, 845	0.21	5, 887	0.36
4	Date	10, 228	0.10	1, 231	0.07
5	Time	1, 732	0.01	354	0.02
6	Money	4, 721	0.04	747	0.04
7	Percent	2, 385	0.02	386	0.02
-	Total	95, 250	1	16, 061	1

TABLE III. ARMANPERSONER CORPUS INFORMATION IN DETAILS

No.	Named Entity Tag	Tokens Absolute Frequency	Tokens Relative Frequency	Types Absolute Frequency	Types Relative Frequency
1	Location	4, 308	0.17	832	0.14
2	Organization	10, 036	0.40	1, 290	0.22
3	Person	5, 215	0.21	1, 829	0.32
4	Facility	1, 485	0.06	548	0.10
5	Event	2, 518	0.10	556	0.10
6	Product	1, 463	0.6	634	0.11
-	Total	25, 025	1	5, 689	1

[3] http://en.itrc.ac.ir/sites/default/files/pictures/NER.rar

[4] https://github.com/AminMozhgani/Persian_NER/tree/master/data

[DOI: 10.52547/itrc.13.2.39]

TABLE IV. BIJANKHAN NAMED ENTITY CORPUS INFORMATION IN DETAILS

No.	Named Entity Tag	Tokens Absolute Frequency	Tokens Relative Frequency	Types Absolute Frequency	Types Relative Frequency
1	Location	21, 760	0.27	3, 960	0.30
2	Organization	32, 719	0.41	3, 647	0.27
3	Person	10, 484	0.13	4, 236	0.32
4	Date	8, 240	0.10	654	0.04
5	Time	2, 457	0.03	204	0.01
6	Money	2, 404	0.03	339	0.02
7	Percent	1, 189	0.01	149	0.01
-	Total	79, 253	1	13, 189	1

B. Results

We consider the best performed models of the NSURL-2019, namely MorphoBERT and Taheri&hosseini as the baselines of the proposed MTL model [4]. These systems are evaluated and compared according to the prepared conll script evaluation metric at both word level and phrase level[5]. Specifically, we calculate word-level and phrase-level precision, recall and F1 scores. At the phrase-level mode, the tags of all words in a named entity should be correct to be considered as one correct instance. Precision and recall are calculated for each tag and then micro-averaged to conclude the overall performance. Finally, a statistically significant test is done according to p-value with significant level 0.05.

Table 4 demonstrates the F1 scores of each framework respectively. From this table, it can be seen that the best result is obtained by MorphoBert model in word level. However, at phrase level, which is more important, the suggested MTL model can achieve 1.73% improvement on F1 score over the previously best reported result by Taher&Hosseini model.

C. Persian Ezafe and Part-of-speech tag feature

In order to examine the fact that whether adding more auxiliary tasks can improve the performance of the proposed model, two other auxiliary tasks are selected based on two strong hypotheses:

1. Detecting Ezafe[6] phenomenon, as one of the most challenging issues in Persian language processing, can lead to a better understanding of phrase boundaries and result in an improvement in NE boundary detection.
2. Detecting POS tags can improve the capability of the system to recognize the named entities better, since the POS tags of words in a named entity obeys specific limited structures.

In order to address these hypotheses, we have included the Peykare corpus [55] that consists of about ten million words with 16 grained POS tags and 608 fined POS tags containing the Ezafe tag. The texts of this corpus are gathered from various data sources like newspapers, magazines, journals, books, letters, hand-written texts, movie scripts, news etc.

In summary, we have included 4 corpora in the multitask framework. The main task is ITRC (NER-

TABLE V. THE NER RESULTS OF BASELINES AND THE PROPOSED MULTITASK FRAMEWORK ACCORDING TO PRECISION, RECALL AND F1 SCORE AT WORD AND PHRASE LEVELS

		Word Level			Phrase Level		
		Precision	Recall	F1 score	Precision	Recall	F1 score
Baseline (single task)	MorphoBert	89.64%	83.80%	86.62%	81.47%	82.44%	81.95%
	Taher&Hosseini	87.99%	85.00%	86.47%	80.94%	83.23%	82.07%
the proposed model	MTL	88.40%	84.46%	86.59%	84.39%	83.22%	83.80%

TABLE VI. THE NER RESULTS OF THE PROPOSED MULTITASK FRAMEWORK USING PERSIAN EZAFE AND PART-OF-SPEECH FEATURES ACCORDING TO PRECISION, RECALL AND F1 SCORE AT WORD AND PHRASE LEVELS.

	Word Level			Phrase Level		
	Precision	Recall	F1 score	Precision	Recall	F1 score
MTL2 MTL+EZF	87.40%	85.95%	86.67%	84.09%	83.70%	83.89%
MTL3 MTL+EZF+POS	88.02%	85.13%	86.55%	84.65%	83.39%	84.02%

[5] https://github.com/sighsmile/conlleval

[6] Ezafe is a special characteristic of the Persian language. This is a morpheme which is pronounced but usually is not written. So it results in some ambiguities in the analysis and understanding of Persian documents especially in NLP applications. This phenomenon (used as -e after consonants and –ye after vowels) links head words to its modifiers in noun phrases (*Raisjomhur-**e** Irān*: 'The president of Iran'), adjective phrases (*Ābi-**ye** kamrang*: 'light blue'), prepositional phrases (*Post-**e** Miz*: 'Behind the table') and adverb phrases (*Nazir-**e** in ketāb*: 'Such as this book'). Recognizing the positions of this morpheme in a given sentence helps determine the phrase boundaries that is necessary for determining named entities.

[DOI: 10.52547/itrc.13.2.39]

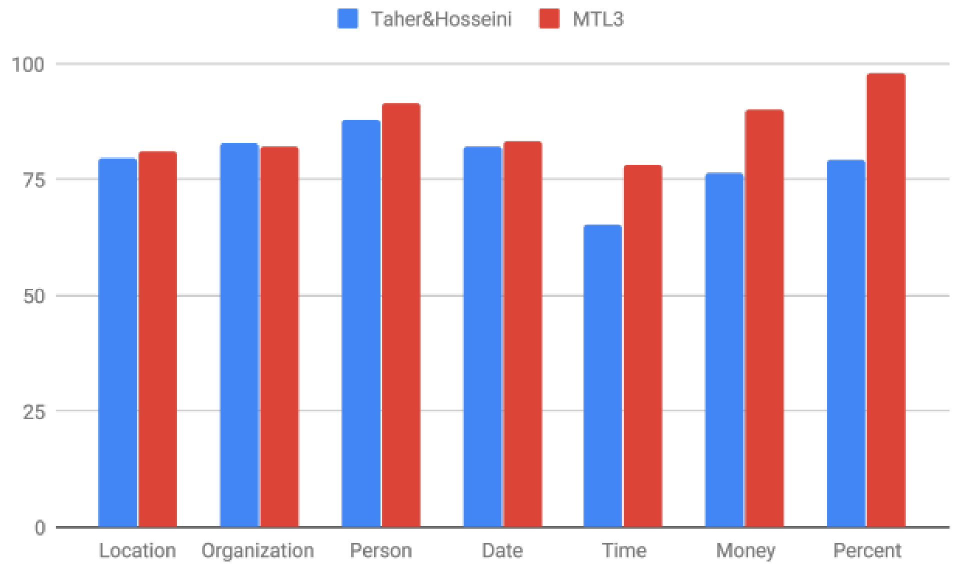

Figure 4. The detailed results of the baselines in comparison with the state-of-the-art results of multitask framework (MTL3). All the results are reported according to the F1 score at the phrase level.

Main) and the auxiliary tasks are ArmanPerso (NER-Aux), Peykare POS (POS) and Peykare Ezafe (EZF). We first include EZF task (called MTL2) and then include all the auxiliary tasks (valled MTL3). Results are shown in Table 5. MTL2 obtains 0.05% improvement on F1 score over the previous best result in word level and an interesting result can be seen at phrase level when all corpora are simultaneously used to train the model (MTL3), yielding a state-of-the-art performance at phrase level by 84.02%.

Fig. 4 compares the detailed information of the previous best results and the MTL3 ones based on F1 scores at phrase level. According to the figure, the interesting fact is that the most notable improvements are achieved for detecting “Time", “Money" and “Percent" tags.

Finally, the confusion matrices of the best baseline (Tahe&Hosseini) and the state-of-the-art MTL model (MTL3) is provided in Table 6 and Table 7. Taking into account these figures, it can be seen that the most common errors are in distinguishing between location and organization named entities in the models. Besides, the baseline has a poor performance in the recognition of date and percent tags. Based on these results, in the next section, we will present the error analyses of the proposed model in details.

D. Discussion

In the single tasks, there are different kinds of errors. The poor performance of the single tasks is due to the low number of training data. It can be seen that (almost) the more data we used to train the model, the better results are obtained in both word level and phrase level. In order to clarify the discussion, some examples are also brought in *a*: ’b’ format, where *a* is the transliteration of the Persian word in the test set and ’b’ is its English translation. The improvements of the multitask framework can be categorized into the following categories based on their origin:

TABLE VII. THE CONFUSION MATRIX OF THE STATE-OF-THE-ART MTL MODEL (MTL3).

		B-LOC	I-LOC	B-ORG	I-ORG	B-PER	I-PER	B-DAT	I-DAT	B-TIM	I-TIM	B-MON	I-MON	B-PCT	I-PCT	O	Total
	B-LOC	11380	676	205	266	19	2	0	1	0	1	0	0	0	0	664	13214
	I-LOC	444	7931	37	696	21	53	2	3	0	0	0	0	0	0	718	9905
	B-ORG	224	41	8832	301	20	1	0	2	0	0	0	0	0	0	831	10252
	I-ORG	209	634	161	19224	28	34	1	6	0	0	0	0	0	0	1212	21509
	B-PER	23	26	34	17	5284	37	0	2	0	0	0	0	0	0	266	5689
	I-PER	1	43	1	40	71	4528	1	2	0	0	0	0	0	0	186	4873
	B-DAT	1	1	2	3	0	1	2904	122	13	5	0	0	0	0	306	3358
	I-DAT	1	2	0	4	0	0	217	4245	5	58	0	0	0	0	234	4766
	B-TIM	0	0	0	0	0	0	19	8	593	54	0	0	0	0	47	721
	I-TIM	4	1	0	0	0	0	17	140	34	1580	0	0	0	0	131	1907
Reference	B-MON	1	1	0	0	0	0	0	0	0	0	577	33	0	0	18	630
	I-MON	0	3	0	0	0	0	0	0	0	0	10	1699	0	0	37	1749
	B-PCT	0	0	0	0	0	0	0	0	0	0	1	0	502	2	7	512
	I-PCT	0	0	0	0	0	0	0	0	0	0	0	6	2	676	9	693
	O	586	1089	1061	1835	278	107	327	221	65	46	40	38	4	3	332723	338426

[DOI: 10.52547/itrc.13.2.39]

TABLE VIII. THE CONFUSION MATRIX OF THE BEST BASELINE (TAHER&HOSSEINI).

		System															
		B-LOC	I-LOC	B-ORG	I-ORG	B-PER	I-PER	B-DAT	I-DAT	B-TIM	I-TIM	B-MON	I-MON	B-PCT	I-PCT	O	Total
Reference	B-LOC	9566	673	142	209	20	0	4	3	0	1	0	0	0	0	759	11377
	I-LOC	373	7381	8	373	10	12	2	4	0	0	0	0	0	0	458	8572
	B-ORG	323	51	9152	329	32	1	3	1	0	0	0	0	0	0	855	10747
	I-ORG	225	892	114	19810	20	41	2	11	0	0	0	0	0	0	956	22071
	B-PER	57	40	63	15	5362	40	0	4	0	2	0	0	0	0	300	5883
	I-PER	2	51	1	27	62	4232	0	2	0	0	0	0	0	0	101	4378
	B-DAT	4	4	0	2	1	0	2961	21	4	0	1	0	0	289	337	3443
	I-DAT	2	7	1	2	0	1	197	4145	3	81	0	1	0	0	167	4607
	B-TIM	1	1	0	0	0	0	31	4	614	99	0	0	0	0	42	792
	I-TIM	2	1	0	0	0	0	13	111	27	1246	0	0	0	0	37	1437
	B-MON	1	1	0	0	0	0	0	0	0	0	523	110	0	0	12	647
	I-MON	0	3	0	0	0	0	0	0	0	0	8	1543	0	0	20	1574
	B-PCT	0	0	0	0	0	0	0	0	0	0	1	0	461	68	13	543
	I-PCT	0	0	0	0	0	0	0	0	0	0	0	0	4	470	16	490
	O	805	1343	852	1619	214	536	275	311	48	311	96	121	43	143	333364	340081

1. *Improvements due to the better understanding of the POS structure*

Compared to the single task framework, the structures of coordinations are better determined in the multitask learning framework[7]. In fact, the POS task boosts the model to better identify coordinationg conjuctions, which leads to better understanding of coordinate structures. For instance, in *Tehrān va Isfahān*: 'Tehran and Isfahan' (two cities in Iran), the conjunction *va*: 'and' is correctly recognized as a linker of the two location entities.

Moreover, there are some named entities within which conjunctions are included as a part, like names of cities, organizations, etc. These entities are also better recognized in the MTL framework in comparison with the single task counterpart. For example, *Cāhārmahāl va Baxtiāri*: 'Chaharmahal and Bakhtiari' (the name of a city in Iran) is correctly identified as a single entity in the MTL framework and the conjunction va: 'and' is correctly tagged as I-LOC.

Furthermore, propositional phrases are better identified by adding the POS as the auxiliary task. This may lead to an improvement in detecting location entities. For example, in *az Tehrān tā Isfahān*: 'From Tehran to Isfahan', both prepositions *az*: 'from' and *tā*: 'to' help correctly determine *Tehran* and *Isfahan* as the location entities.

2. *Improvements due to the better understanding of the Ezafe phenomenon*

Thanks to adding data to the better recognition of Ezafe phenomenon, the entities boundaries are better identified in the MTL framework. For instance, in *Estādiom-e varzeši-e Tehrān*: 'The sports stadium in Tehran', all words are correctly recognized as a location entity in the MTL model; But in the single tasks, the words *Estādiom-e varzeši*: 'the sports stadium' are tagged as Other mistakenly.

3. ***Improvements due to better modeling of words through adding more training data***

The large amount of data has a considerable influence on recognizing the correct tags of polisemic words in the MTL tasks. In fact, this massive data helps consider the context and predict the correct NE tag. For example, the word *maqām* has two different meanings in Persian; The first meaning refers to a position of a job in organizations or politics and the other one means the place of someone in a race or competition in relation to the other competitors. This word is correctly tagged in *maqām-e mo'azzam-e rahbari*: 'supreme leadership authority' and *maqām-e noxost*: 'first position'. The first one is tagged as Person and the second one is tagged as Other in multitask framework. As another example we can refer to the following example:

Bozorg-tarin meydān-e nafti ke kašf šod, meydān-e nafti-e Āzādegān bud. :'The biggest oil field founded was Azadegan oil field.'

In this example, the first *meydān-e nafti*: 'oil field' is tagged as Other and the second one is recognized as a named entity due to correct recognizing the proper noun 'Azadegan'.

In spite of the aforementioned improvements, there are some issues that are still unsolved using the proposed model:

- Abbreviations are not tagged as a named entity in both single task and Multitask frameworks.
- Some proper nouns are followed by attributive adjectives. All models fail to separate these adjectives and mistakenly include them in the named entity. *Bu š -e jomhurixah*: 'Republican Bush' is an example that the adjective is tagged as I-pers mistakenly.
- A large source of error is due to the error in the training data. In addition to words with wrong tags, there are

[7] In linguistics, coordination is a frequently occurring complex syntactic structure that links together two or more elements, known as conjuncts or conjoins. The presence of coordination is often signaled by the appearance of a coordinator (coordinating conjunction), e.g. and, or, but (in English). https://en.wikipedia.org/wiki/Coordination_(linguistics)

[DOI: 10.52547/itrc.13.2.39]

proper nouns such as the name of person, city, organization etc. that have different POS or NER tags in corpora.

V. Conclusion

In this paper, we investigated whether the multitask learning could improve the performance of the NER task in low-resource Persian language. We used three auxiliary tasks (NER task, Ezafe task and POS task) to share their features to improve the performance of the main (NER) task. The results show that a good number of training data and considering Ezafe constructions play a significant role to gain the better accuracy in the phrase level. The next step of this research can be using dependency parsing. Based on error analyses, the main and most of errors are related to the recognition of phrase boundaries in Persian. Dependency relations between heads and their modifiers can significantly help obtain a more accuracy in both word and phrase levels. Another track of work is regarding the Generative Adversarial Network to purify the features extracted in the shared and private feature spaces, like the one used in [53].

References

[1] Chiu, Jason PC, and Eric Nichols. "Named entity recognition with bidirectional LSTM-CNNs." Transactions of the Association for Computational Linguistics 4 (2016): 357-370.

[2] Peng, Nanyun, and Mark Dredze. "Improving named entity recognition for chinese social media with word segmentation representation learning." arXiv preprint arXiv:1603.00786 (2016).

[3] Riedl, Martin, and Sebastian Padó. "A named entity recognition shootout for german." Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers). 2018.

[4] Taghizadeh, Nasrin, et al. "NSURL-2019 Task 7: Named entity recognition for Farsi." Proceedings of The First International Workshop on NLP Solutions for Under Resourced Languages (NSURL 2019) co-located with ICNLSP 2019-Short Papers. 2019.

[5] Caruana, Rich. "Algorithms and applications for multitask learning." ICML. 1996.

[6] R. Caruana, Multitask learning, Machine learning 28 (1) (1997) 41–75.

[7] Abdulnabi, Abrar H., et al. "Multi-task CNN model for attribute prediction." IEEE Transactions on Multimedia 17.11 (2015): 1949-1959.

[8] Chu, Xiao, et al. "Multi-task recurrent neural network for immediacy prediction." Proceedings of the IEEE international conference on computer vision. 2015.

[9] Xu, Jianpeng, Jiayu Zhou, and Pang-Ning Tan. "Formula: F act OR ized MU lti-task L e A rning for task discovery in personalized medical models." Proceedings of the 2015 SIAM International Conference on Data Mining. Society for Industrial and Applied Mathematics, 2015.

[10] He, Dan, David Kuhn, and Laxmi Parida. "Novel applications of multitask learning and multiple output regression to multiple genetic trait prediction." Bioinformatics 32.12 (2016): i37-i43.

[11] Bai, Jing, et al. "Multi-task learning for learning to rank in web search." Proceedings of the 18th ACM conference on Information and knowledge management. 2009.

[12] Chapelle, Olivier, et al. "Multi-task learning for boosting with application to web search ranking." Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining. 2010.

[13] Liu, Pengfei, Xipeng Qiu, and Xuanjing Huang. "Recurrent neural network for text classification with multi-task learning." arXiv preprint arXiv:1605.05101 (2016).

[14] Liu, Pengfei, Xipeng Qiu, and Xuanjing Huang. "Deep multi-task learning with shared memory." arXiv preprint arXiv:1609.07222 (2016).

[15] Xiao, Liqiang, Honglun Zhang, and Wenqing Chen. "Gated multi-task network for text classification." Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers). 2018.

[16] Hershcovich, Daniel, Omri Abend, and Ari Rappoport. "Multitask parsing across semantic representations." arXiv preprint arXiv:1805.00287 (2018).

[17] Luong, Minh-Thang, et al. "Multi-task sequence to sequence learning." arXiv preprint arXiv:1511.06114 (2015).

[18] Zaremoodi, Poorya, Wray Buntine, and Gholamreza Haffari. "Adaptive knowledge sharing in multi-task learning: Improving low-resource neural machine translation." Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers). 2018.

[19] Rao, Jinfeng, Ferhan Ture, and Jimmy Lin. "Multi-task learning with neural networks for voice query understanding on an entertainment platform." Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. 2018.

[20] Bingel, Joachim, and Anders Søgaard. "Identifying beneficial task relations for multi-task learning in deep neural networks." arXiv preprint arXiv:1702.08303 (2017).

[21] Alonso, Héctor Martínez, and Barbara Plank. "When is multitask learning effective? Semantic sequence prediction under varying data conditions." arXiv preprint arXiv:1612.02251 (2016).

[22] Bikel, Daniel M., et al. "Nymble: a high-performance learning name-finder." arXiv preprint cmp-lg/9803003 (1998).

[23] Carreras, Xavier, Lluis Marquez, and Lluís Padró. "Named entity extraction using adaboost." COLING-02: The 6th Conference on Natural Language Learning 2002 (CoNLL-2002). 2002.

[24] Chieu, Hai Leong, and Hwee Tou Ng. "Named entity recognition: a maximum entropy approach using global information." COLING 2002: The 19th International Conference on Computational Linguistics. 2002.

[25] Finkel, Jenny Rose, and Christopher D. Manning. "Hierarchical joint learning: Improving joint parsing and named entity recognition with non-jointly labeled data." Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics. 2010.

[26] Sikdar, Utpal Kumar, and Björn Gambäck. "Feature-rich twitter named entity recognition and classification." Proceedings of the 2nd Workshop on Noisy User-generated Text (WNUT). 2016.

[27] Huang, Zhiheng, Wei Xu, and Kai Yu. "Bidirectional LSTM-CRF models for sequence tagging." arXiv preprint arXiv:1508.01991 (2015).

[28] Lample, Guillaume, et al. "Neural architectures for named entity recognition." arXiv preprint arXiv:1603.01360 (2016).

[29] Cotterell, Ryan, and Kevin Duh. "Low-resource named entity recognition with cross-lingual, character-level neural conditional random fields." Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 2: Short Papers). 2017.

[30] Sato, Motoki, et al. "Segment-level neural conditional random fields for named entity recognition." Proceedings of the Eighth International Joint

[DOI: 10.52547/itrc.13.2.39]

Conference on Natural Language Processing (Volume 2: Short Papers). 2017.
[31] de Castro, Pedro Vitor Quinta, Nádia Félix Felipe da Silva, and Anderson da Silva Soares. "Portuguese named entity recognition using lstm-crf." International Conference on Computational Processing of the Portuguese Language. Springer, Cham, 2018.
[32] Ahmed, Sajawel, and Alexander Mehler. "Resource-size matters: Improving neural named entity recognition with optimized large corpora." 2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA). IEEE, 2018.
[33] Gunawan, William, et al. "Named-entity recognition for indonesian language using bidirectional lstm-cnns." Procedia Computer Science 135 (2018): 425-432.
[34] Ekbal, Asif, et al. "Language independent named entity recognition in indian languages." Proceedings of the IJCNLP-08 Workshop on Named Entity Recognition for South and South East Asian Languages. 2008.
[35] Collobert, Ronan, and Jason Weston. "A unified architecture for natural language processing: Deep neural networks with multitask learning." Proceedings of the 25th international conference on Machine learning. 2008.
[36] Aguilar, Gustavo, et al. "A multi-task approach for named entity recognition in social media data." arXiv preprint arXiv:1906.04135 (2019).
[37] Derczynski, Leon, et al. "Results of the WNUT2017 shared task on novel and emerging entity recognition." Proceedings of the 3rd Workshop on Noisy User-generated Text. 2017.
[38] Crichton, Gamal, et al. "A neural network multi-task learning approach to biomedical named entity recognition." BMC bioinformatics 18.1 (2017): 1-14.
[39] Wang, Xuan, et al. "Cross-type biomedical named entity recognition with deep multi-task learning." Bioinformatics 35.10 (2019): 1745-1752.
[40] Zhao, Huasha, et al. "Improve neural entity recognition via multi-task data selection and constrained decoding." Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers). 2018.
[41] Lin, Ying, et al. "A multi-lingual multi-task architecture for low-resource sequence labeling." Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). 2018.
[42] Nosirova, Nargiza, Mingbin Xu, and Hui Jiang. "A multi-task learning approach for named entity recognition using local detection." arXiv preprint arXiv:1904.03300 (2019).
[43] Mortazavi, P. S., and M. Shamsfard. "Named entity recognition in persian texts." 15th National CSI Computer Conference. 2009.
[44] Khormuji, Morteza Kolali, and Mehrnoosh Bazrafkan. "Persian named entity recognition based with local filters." International Journal of Computer Applications 100.4 (2014).
[45] Bijankhan, Mahmood. "The role of the corpus in writing a grammar: An introduction to a software." Iranian Journal of Linguistics 19.2 (2004): 48-67.
[46] Ahmadi, Farid, and Hamed Moradi. "A hybrid method for Persian named entity recognition." 2015 7th conference on information and knowledge technology (IKT). IEEE, 2015.
[47] Dashtipour, Kia, et al. "Persian named entity recognition." 2017 IEEE 16th International Conference on Cognitive Informatics & Cognitive Computing (ICCICC). IEEE, 2017.
[48] Poostchi, Hanieh, et al. "PersoNER: Persian named-entity recognition." COLING 2016-26th International Conference on Computational Linguistics, Proceedings of COLING 2016: Technical Papers. 2016.
[49] Hosseinnejad, Shadi, Yasser Shekofteh, and Tahereh Emami Azadi. "A'laam corpus: A standard corpus of named entity for Persian language." Signal and Data Processing 14.3 (2017): 127-142.
[50] Bokaei, Mohammad Hadi, and Maryam Mahmoudi. "Improved deep persian named entity recognition." 2018 9th International Symposium on Telecommunications (IST). IEEE, 2018.
[51] Poostchi, Hanieh, Ehsan Zare Borzeshi, and Massimo Piccardi. "Bilstm-crf for persian named-entity recognition armanpersonercorpus: The first entity-annotated persian dataset." Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018). 2018.
[52] Devlin, Jacob, et al. "Bert: Pre-training of deep bidirectional transformers for language understanding." arXiv preprint arXiv:1810.04805 (2018).
[53] Liu, Pengfei, Xipeng Qiu, and Xuanjing Huang. "Adversarial multi-task learning for text classification." arXiv preprint arXiv:1704.05742 (2017).
[54] Zahedi, Mohammad Sadegh, et al. "Persian word embedding evaluation benchmarks." Electrical Engineering (ICEE), Iranian Conference on. IEEE, 2018.
[55] Bijankhan, Mahmood, et al. "Lessons from building a Persian written corpus: Peykare." Language resources and evaluation 45.2 (2011): 143-164.

Mohammad Hadi Bokaei received the B.Sc. and M.Sc. degrees from Iranian University of Science and Technology (2008) and Sharif University of Technology (2011), respectively, and the Ph.D. degree in Artificial Intelligence from Sharif University of Technology in 2015. He is currently an Associate Professor at the Iran Telecommunication Research Center, Tehran, Iran. His research interests include the area of Deep Learning, Machine Learning, Spoken Language Processing and Natural Language Processing.

Mohammad Nouri is a researcher and developer in the field of Machine Learning. He received the M.Sc. degree in Artificial Intelligence from Kharazmi University in 2018. He was a Researcher at the Iran Telecommunication Research Center (ITRC), Tehran, Iran, from 2018 to 2021. His research interests include Machine Vision, Neural Networks, Natural Language Processing.

Abdolah Sepahvand is a Ph.D. student in the Department of Computer Engineering at Amirkabir University of Technology (AUT). He received the M.Sc. degree in Computer Science from Amirkabir University in 2015. His research interests include Computational Geometry and Approximation Algorithms.

[DOI: 10.52547/itrc.13.2.39]