AUT Journal of Mathematics and Computing

AUT J. Math. Comput., 7(4) (2026) 407-422
https://doi.org/10.22060/AJMC.2025.23058.1225

Original Article

A persian benchmark for joint intent detection and slot filling

Masoud Akbari[a], Amir Hossein Karimi[a], Tayyebeh Saeedi[a], Zeinab Saeidi[a], Kiana Ghezelbash[a], Fatemeh Shamsezat[a,b], Mohammad Akbari[a], Ali Mohades Khorasani[*a]

[a] *Department of Computer Science, Amirkabir University of Technology (Tehran Polytechnic), Tehran, Iran*
[b] *Department of Computer Science, Faculty of Mathematics and Computer, Fasa University, Fasa, Iran*

ABSTRACT: Natural Language Understanding (**NLU**) is important in today's technology as it enables machines to comprehend and process human languages, leading to improved human-computer interactions and advancements in fields such as virtual assistants, chatbots, and language-based AI systems. This paper highlights the significance of advancing the field of **NLU** for low-resource languages. With intent detection and slot filling being crucial tasks in **NLU**, the widely used datasets **ATIS** and **SNIPS** have been utilized in the past. However, these datasets only cater to the English language and do not support other languages. In this work, we aim to address this gap by creating a Persian benchmark for joint intent detection and slot filling based on the **ATIS** dataset. To evaluate the effectiveness of our benchmark, we employ state-of-the-art methods for intent detection and slot filling.

Review History:

Received:19 March 2024
Revised:02 September 204
Accepted:29 January 2025
Available Online:01 October 2026

Keywords:

ATIS dataset
Persian benchmark
Intent detection
Slot filling

MSC (2020):

68T50; 68P20; 91C20; 62D05

1. Introduction

Each goal-oriented dialogue system must include Natural Language Understanding (**NLU**) in order to process an utterance and attempt to determine the user's need. NLU aims to automatically identify what the user wants to do (or *intent* of the user), extract associated arguments (or *slots*), and respond correctly to satisfy the user's request [35]. In such a context, intent detection and slot filling are essential components of NLU. In the last decade, several works have investigated these tasks; however, most of them work on English Language Understanding [4, 29, 37].

Some works on NLU have been done in other languages, such as Romanian [32], Chinese [23], Arabic [2], and Vietnamese [6]. Despite the fact that the Persian language is the 24th most spoken language in the world (more than 78 million native speakers)[1], to based best of our knowledge there is no specialized research on Persian Language Understanding due to the lack of a Persian public dataset. Without such a dataset, it would be difficult for researchers to work in NLU or, in general, in Natural Language Processing (**NLP**) tasks in Persian. To address this problem, we develop a dataset in the Persian language and apply some of the state-of-the-art approaches to introduce a public benchmark in the Persian language.

[*] *Corresponding author.*

E-mail addresses: ma.akbari421@aut.ac.ir (M. Akbari), ahkarimi@aut.ac.ir (A. H. Karimi), t.saeedi@aut.ac.ir (T. Saeedi), zsaeidi2007@aut.ac.ir (Z. Saeidi), kghezelbash@aut.ac.ir (K. Ghezelbash), shamsezat@aut.ac.ir (F. Shamsezat), akbari.ma@aut.ac.ir (M. Akbari), mohades@aut.ac.ir (A. Mohades Khorasani)

[1] https://www.ethnologue.com/language/pes

The Persian language is one of the most popular languages in the world which is spoken by people in the Middle East and Central Asia, and it is known as Farsi in Iran, Dari in Afghanistan, and Tajik in Tajikistan[2]. However, this language just has 1.3% of content on the internet until 16 August 2024[3]. So, it shows that even though this language has more than 78 million native speakers and over 120 million speakers[4], it does not have wide content in the written form in the world. Moreover, based on the W3techs website, the Chinese language has 1.2% content and it is just one rank after the Persian, but to the best of our knowledge, it has more benchmark datasets in NLP than Persian[5]. For example, you can see [14, 19, 21, 27, 33, 38, 40, 41] for Chinese datasets and [1, 3, 9] for Persian. Also, despite the English language which its sentence structure is a Subject-Verb-Object (**SVO**) language, Persian is a Subject-Object-Verb (**SOV**) language in formal form, and in informal forms, sometimes it turns to SVO. For instance, the sentence "Ashkan arrived at the factory" is written in the formal form of Persian as "اشکان به کارخانه رسید" ([æʃkaːn]$_{S}$ [bɜ kaːrkhaːnɜ]$_{PP}$ [rɜsiːd]$_{V}$) and written in the informal form as "اشکان رسیدش کارخونه" ([æʃkaːn]$_{S}$ [rɜsiːd-ɜʃ]$_{V}$ [kaːrkhuːnɜ]$_{NP}$). Further, you can see in the informal form of this example, that the preposition is attached to the verb (-ɜʃ), and the written form of the translation "factory" has a different form in the formal and the informal form. Another attribute of the Persian language is that it is written and read from right to left which is opposite of English language. So, it is important to understand the Persian language by machine. We provide a benchmark dataset of the Persian in this paper in order to help researchers for further works on the Persian language processing.

One of the most widely used datasets in intent detection and slot filling is the **ATIS** (Air Transform Information System) dataset. This dataset is originally in English and includes intent and slot annotations. In 2018, Upadhyay et al. extended the ATIS to Hindi and Turkish [36]. Also, after that, in 2020, the authors in [39] introduced the **MultiATIS++** and extended ATIS to six more languages in four families. Hence, MultiATIS++ covers nine languages: English, Spanish, German, French, Portuguese, Chinese, Japanese, Hindi, and Turkish. As mentioned before, the MultiATIS++ covers four families: Indo-European, Sino-Tibetan, Japonic, and Altaic [39]. However, the MultiATIS++ fails to cover the Persian language, which belongs to the Indo-European family. Moreover, FitzGerald et al. recently presented the MASSIVE dataset, a multilingual dataset covering 51 languages, including Persian, for intent detection and slot filling [12]. The authors in [12] used existing languages available in major virtual assistants and applied the Amazon Mechanical Turk[6] manner for checking the translations and slots of each sentence. Since this dataset was collected by utterances in which users talk to their virtual assistants, sentences are too short (most of them are commands), and the average length of the Persian sentences is 7.3. Although this dataset covers the Persian language, the authors did not mention that they had checked their sentences with a Persian native speaker. In their work, some samples in this dataset do not make sense in Persian, such as the 1170th sample.

In this paper, we want to introduce a Persian public benchmark for all researchers in the NLP field. We use the original ATIS dataset and apply a semi-automatic approach to provide a Persian dataset. Since the sentences in the ATIS dataset are conversational, they are not very short. Our presented dataset has an average length of 12.3 sentences in Persian. Also, we have checked all the sentences by native speakers. To provide a baseline for researchers, we perform a comparison performance analysis to have a comprehensive and public Persian benchmark. This benchmark will help researchers interested in Persian Language Understanding tasks improve their works.

Furthermore, it's crucial to note that Large Language Models **(LLMs)** have demonstrated remarkable capabilities across various tasks, including intent detection and slot filling [24]. While support for the Persian language in LLMs has improved, many models still struggle with nuanced understanding and generation in Persian, particularly for specialized tasks. As presented in this paper, the introduction of a dedicated Persian benchmark dataset plays a pivotal role in facilitating the evaluation and fine-tuning of LLMs on tasks such as intent detection and slot filling in the context of the Persian language. This benchmark not only enhances the accessibility of Persian language resources but also provides a standardized platform for the assessment and improvement of LLMs in the specific domain of Persian NLP, helping to address the ongoing challenges in Persian language processing.

Our contributions are summarized as follows:

- We construct an intent detection and slot filling dataset in Persian by extending the ATIS dataset to Persian,
- We perform a comparative analysis of state-of-the-art models on the Persian dataset which we constructed,
- We make our benchmark publicly available for research and educational purposes. Our benchmark, we believe, will serve as a starting point for future Persian NLU research and applications[7].

[2] https://www.languageconnectsfoundation.org/connect-with-language/choose-your-language/connect-with-persian
[3] https://w3techs.com/technologies/overview/content_language
[4] https://en.irna.ir/news/85516375/Persian-among-five-sweetest-languages-in-world-Iranian-official
[5] https://metatext.io/datasets-list
[6] https://www.mturk.com/worker
[7] https://github.com/Makbari1997/Persian-Atis

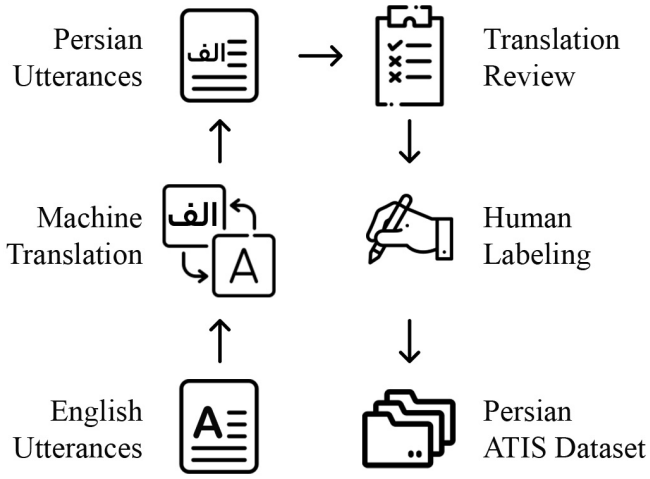


Figure 1: Process flow of constructing Persian ATIS via a semi-automatic approach using translation

2. Problem statement

In this paper, we focus on intent detection and slot filling, which are two main tasks in NLU. We apply the algorithms that can perform intent detection and slot filling simultaneously on our provided Persian dataset. At first, we explain these problems and their applications in NLU.

2.1. Intent detection

Intent detection identifies user intentions to provide appropriate responses. It's typically modeled as sentence classification, with a focus on spoken language utterances. This task requires extracting semantic features before applying classification algorithms. Research on intent detection spans search engines, dialogue systems, question-answering systems, and text categorization [37].

2.2. Slot filling

Slot filling is a sequence labeling task that assigns semantic labels to word sequences. It processes sentences as input and outputs tags for each word [22, 30, 37, 42]. Commonly used in dialogue systems, slot filling interprets user sentences to determine information retrieval or task completion. It's also applied in task-oriented dialog systems for online shopping, improving product search queries and customer assistance [37].

3. Description of Persian dataset

There are Persian datasets for NLP tasks like question-answering [7, 11, 16], language modeling[8], or sentiment analysis [31]. However, there is no specific Persian benchmark dataset for the NLU task. As NLU significantly affects a chatbot's performance, a Persian dataset is necessary for further research in this branch. ATIS is an English benchmark dataset for NLU. As mentioned before, in this work, we chose the ATIS dataset and translated it into Persian automatically. Then, we double-checked each sentence by native speakers. We used an automatic translator to speed up and avoid typing each utterance by native speakers in order to reduce typo errors. We chose this dataset because equivalences of ATIS utterances in Persian are more meaningful than datasets like SNIPS. For example, the sentence "*Find The Passion of Michel Foucault novel.*" contains the name of an object that is not common in Persian. On the other side, the translation of a sentence in the ATIS dataset, like "I want to fly from Boston at 8 am and arrive in Denver at 11 in the morning," contains more common words in Persian. Furthermore, the ATIS was constructed on an actual popular travel domain, making it meaningful in various languages. It's worth noting that the ATIS dataset, while not entirely free from bias, offers several advantages from this perspective. Its focus on common travel scenarios and use of neutral language helps minimize potential cultural or gender-specific biases. Additionally, the straightforward nature of flight-related queries tends to be less prone to subjective interpretations across different languages and cultures.

[8]Persian Wikipedia Dataset

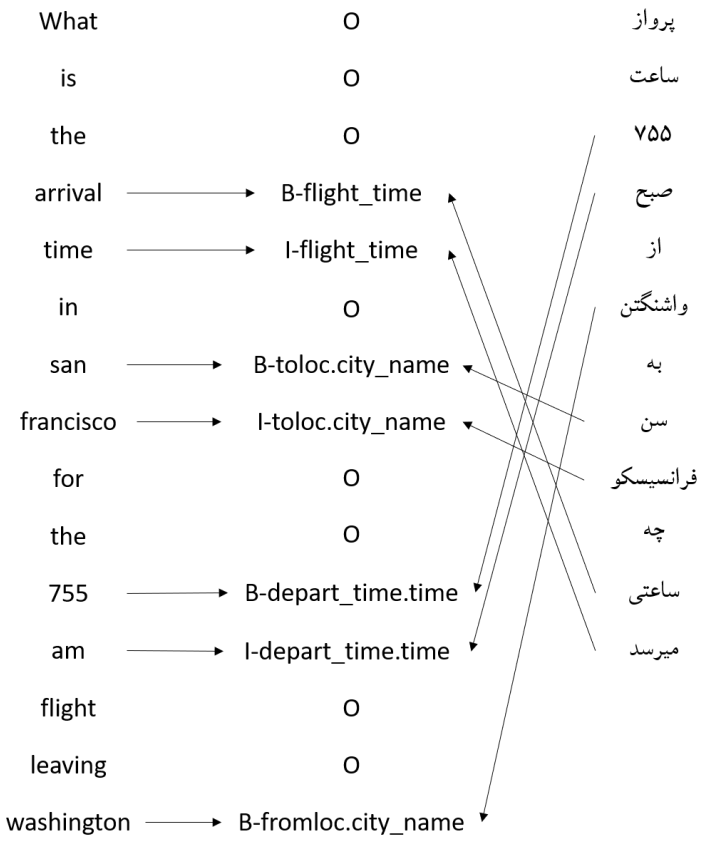


Figure 2: An example of supplants of words after translation and the need to relabel slot tags.

The ATIS dataset comprises 5871 utterances in English (4978 utterances for training and 893 for evaluation). Fig. 1 shows the process flow of creating the Persian ATIS dataset. The first step was to translate ATIS into Persian using machine translation. We used the **EasyNMT**[9] package as the machine translator. EasyNMT contains three different models for neural machine translation and covers more than 100 languages. mBart50 [34] was one of the models in EasyNMT that covers the Persian language, too. We also used mBart50 to translate English ATIS to Persian.

Since the machine translations may make minor mistakes, such as misplacing translated words in the destination language, all the translations were double-checked and corrected by Persian natives. For instance, the sentence "I would like to fly from Atlanta to Denver on September fifteenth," which was originally translated to "من می‌خواهم یک پرواز از آتلانتا به دنور در پانزده سپتامبر" by machine translation, was transformed to its correct format which is "من یک پرواز از آتلانتا به دنور در پانزده سپتامبر می‌خواهم". As mentioned before, Persian is written and read from right to left, and it is an SOV language in the formal form, in this instance, you can see the verb "would like," which translates to "می‌خواهم", has come in inappropriate position in the translated sentence by machine translation and the correct position is fixed by native speakers. The third step was labeling. The intent of an utterance remains the same during translation, so no effort is needed to set intent labels of utterances. However, as some words are supplanted after translation, it is necessary to do the slot labeling process. An example of this problem is shown in Fig. 2. It should be noted that NLP experts did the labeling process.

As with other natural language processing tasks, preprocessing the input text is necessary in the last step. There is a challenge in Persian NLP datasets related to the similarity between the Arabic and Persian alphabets. This means that two words with the same shape may be considered different tokens because their characters have different ASCI codes. Normalizing the data and unifying the characters can solve this problem. We used the **Hazm**[10] Persian NLP library with some other changes for normalization. After that, the punctuation was also removed as a step in preprocessing the NLU task. The number of unique characters changed from 113 to 69. Some exceptional names were in the translated data, such as the city or airport names in the ATIS dataset, even after the translation process. We substituted them with Persian scripts; for example, as shown in Fig. 2, the "Washington"

[9] https://github.com/UKPLab/EasyNMT
[10] https://github.com/roshan-research/hazm

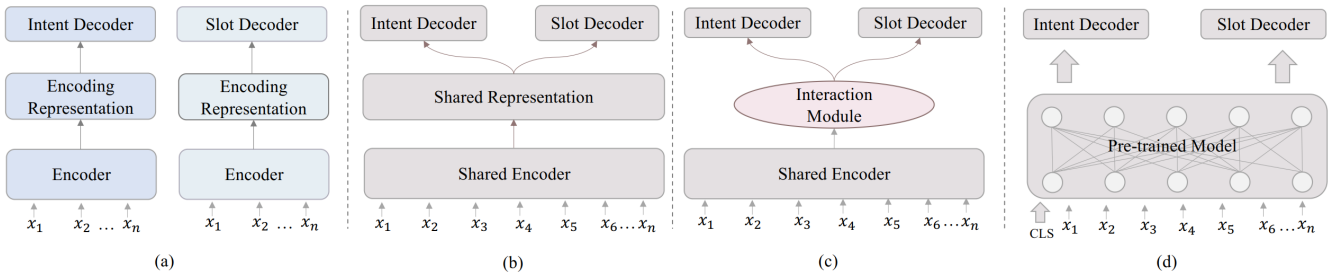


Figure 3: (a) Single models. (b) Implicit Joint Modeling. (c) Explicit Joint Modeling. (d) Pre-trained Model Paradigm. [26]

city is just written by the Persian script as "واشنگتن" (ˈwɒʃ.ɪŋ.tən).

It is essential to highlight the significance of having a dedicated Persian dataset for joint intent detection and slot filling, in addition to the technical aspects of dataset creation. Persian has unique linguistic characteristics that directly impact the performance of natural language processing (NLP) tasks. Here are some of the features:

- **Linguistic Uniqueness of Persian**
  - *Script and Orthography:* Persian uses a modified form of the Arabic script, which introduces complexities such as the omission of short vowels and the use of contextually varied letter forms. These features pose specific challenges in text processing, necessitating tailored NLP techniques. For example, the word "ترک" pronounced as [tæræk] is a noun meaning "gap," while "ترک" with the pronunciation [tærk] is a verb that means "to leave."
  - *Morphological Richness:* The inflectional nature and rich morphology of Persian impact linguistic tasks like stemming and lemmatization significantly. For instance, the phrase "I want" in English is equivalent to "می‌خواهم" in Persian, whereas the negative form "I do not want" corresponds to "نمی‌خواهم", in Persian, both of which require distinct processing during linguistic analysis.
  - *Lexical Gaps:* When translating concepts from other languages into Persian, there may be lexical gaps that require a native dataset to capture language nuances accurately. For instance, the word "through" in the English sentence "I'd like to fly from Philadelphia to Dallas through Atlanta" lacks an exact equivalent in Persian and requires careful translation.
- **Cultural and Regional Relevance**
  - *Contextual Understanding:* Cultural nuances and idiomatic expressions deeply embedded in the Persian language require a dataset that can capture these subtleties effectively. This is critical for accurate intent detection and slot filling.
- **Technical and Resource Challenges**
  - *Model Training and Adaptation:* Specialized datasets facilitate more effective training of machine learning models and allow for fine-tuning to the specific characteristics of the language. A Persian dataset supports training on local nuances, ensuring better performance in NLP tasks. For instance, correctly identifying intent and slots in a Persian travel booking conversation requires familiarity with local city names and travel-related terms in Persian [18].

In summary, the creation of a dedicated Persian dataset not only addresses linguistic challenges but also enhances the contextual and cultural understanding necessary for robust joint intent detection and slot filling in Persian NLP applications.

4. A taxonomy of intent detection and slot filling

In this section, we announced some state-of-the-art methods were chosen to evaluate the Persian ATIS dataset. These methods had to cover various approaches to produce a reliable benchmark. We chose these methods based on the taxonomy defined in [26]. This taxonomy classifies methods in NLU into single models, joint models, and pre-trained paradigms. The paradigms are visualized in Fig. 3, which was presented initially in [26].

The single models paradigm has no interaction between intent detection and slot-filling tasks. Hence, intent detection and slot-filling will not be able to take advantage of their correlation and share helpful information to improve performance. On the other side, joint models leverage the shared knowledge between intent detection and slot-filling.

The joint paradigm is also separated into two classes: explicit and implicit. The implicit joint modeling only uses a shared encoder (representation) and does not explicitly model the interaction, which may result in low interpretability and performance. The explicit joint modeling paradigm is classified into a single and bidirectional flow interaction. The single flow only considers a single information flow from intent to the slot. In contrast, the bidirectional flow means that the model considered the cross-impact between intent detection and slot filling.

Lastly, the pre-trained paradigm is defined as a paradigm where pre-trained language models (like BERT) are used as encoders to extract representation. Pre-trained models can provide rich semantic features, which can help to improve the performance of SLU tasks.

In the following, we will discuss the selected methods based on the taxonomy to evaluate our provided Persian dataset. The goal of using these models is just to get their performance on our dataset; it does not compare models with each other.

4.1. Models

- **SF-ID [10]**: This model lies in the explicit joint modeling paradigm. In this approach, A bidirectional model considers the cross-impact of slot filling and intent detection. The first part is calculating the context vectors by an attention mechanism. Next is the network consisting of slot-filling and intent detection subnets, where their order can be customized (slot-filling first or intent detection first). Finally, a CRF layer is added above the slot-filling subnet outputs to consider the correlations between the labels in neighborhoods. It jointly decodes the best chain of labels of the utterance.

- **Attention RNN [20]**: This approach uses a neural network based on the attention mechanism to solve the problem of joint intent detection and slot filling. This approach also lies in the implicit joint modeling category. The proposed model in [20] consists of an encoder and a decoder, where the encoder uses a BiLSTM network and the decoder uses an LSTM. The concatenation of the forward and backward walks related to the last hidden state will be used to initiate the decoder's hidden state. The context vector is also used to compute the attention. The output of the decoder will be used for slot filling, and the output of the encoder for intent detection.

- **CNN-LSTM-CRF [15]**: This model can be trained as a single or joint model. The CNN-LSTM-CRF model first uses a CNN network to encode utterances at the word level and capture morphological information, such as prefixes and suffixes. The CNN outputs will go through a BiLSTM network, and the BiLSTM outputs are passed to a CRF layer for slot filling and a classifier to detect the intent.

- **Slot-Gated [13]**: The Attention-based RNN model achieves the best performance on joint ID and SF (where the attention weights of ID and SF are independent). This explicitly modeled method proposes a slot-gated structure, which focuses on learning the relationship between intent and slot attention vector and obtains a better semantic frame through global optimization. They explicitly model the relationships between slots and intents using a single flow interaction from intents to slots. The proposed slot gating model introduces an additional gate that utilizes intent context vectors to model slot-intent relations to improve slot-filling performance. First, the slot context vector and intent context vector combine. In other words, in the encoder, for each time stamp, a slot context vector and a global intent context vector are computed using an attention mechanism. They introduced slot-gate g, computed as a function of context vectors. Then, g is used as a weight for the input's slot label of the i'th word. Then, they formulated the objective as the conditional probability of understanding the result (slot filling and intent prediction) given the input word sequence to obtain both slots and intent labels.

- **Co-Interactive transformer [25]**: As a multi-task learning model, the CIT model can fill slots and detect intent simultaneously. This network consists of two sub-networks: the slot-filling network and the intent detection network. A co-interactive mechanism connects these two sub-networks, allowing them to interact and share information. Slot-filling networks encode input text using transformers and predict slot labels based on linear layers. Another transformer-based encoder encodes the input text, and a linear layer predicts the intent label for the entire text. A multi-head attention mechanism is used to implement the co-interactive mechanism, which allows the slot-filling and intent detection networks to interact and share information. Specifically, the multi-head attention mechanism calculates the attention weights between the slot-filling and intent detection representations and then combines the two representations according to these attention weights. This method is an explicit joint modeling method. However, since BERT representations can be applied along with this model, the proposed method can also be classified as a pre-trained modeling paradigm. In this paper, we only used it as an explicit joint modeling method as the official implementation using BERT was unavailable.

- **JointBERT [4]**: This method uses representations from the pre-trained BERT model to solve the joint intent detection and slot-filling task. A softmax activation function is used over the representation computed for the CLS tag of each input to detect intents and a softmax over the final hidden states of other tokens to classify slots. This method lies in the pre-trained paradigm. We used the BERT's multi-lingual version to apply the model to Persian and English ATIS [8].
- **CTran [28]**: The CTran model is a joint Intent Detection and Slot Filling architecture that utilizes a pre-trained language model (BERT and ELMo), a convolutional layer, a window feature sequence structure, and stacked Transformer encoders. The model includes a Shared Encoder, an SF Decoder, and an ID Decoder that enable joint learning. The Intent-Detection Decoder and Slot-Filling Decoder utilize self-attention layers and linear feed-forward networks for classification. The model's effectiveness is demonstrated through the described components, which contribute to joint intent detection and slot-filling objectives. We have utilized multilingual BERT to run this model.
- **Code-Switching [17]**: The proposed methodology involves enhancing the monolingual data by introducing multilingual code-switching through translating data chunks into various languages. This approach is designed to improve the language-neutral capabilities of models like mBERT while fine-tuning them for downstream tasks such as intent prediction and slot filling. The method aims to enhance model performance in zero-shot scenarios where the target language is unknown during training by translating and aligning slot labels to create artificial code-switched data. Our experiment involved running the model in the code-switching mode and evaluating it on our proposed dataset to measure its effectiveness for the Persian language. Although the code-switching method uses pre-trained multi-lingual BERT, we will consider it a separate category from pre-trained approaches as the model can be directly tested on other languages.

In the next section, we will provide the performances of the mentioned models on the Persian ATIS dataset.

5. Experiments

In this section, we evaluate our proposed corpus by state-of-the-art Intent detection and Slot filling models and compare that with the English ATIS. We use the models mentioned in the previous section.

5.1. Evaluation Metrics

The two metrics used for comparison are accuracy for intent detection and F1-score for slot-filling. As the ATIS dataset is imbalanced for both intent and slot classes, we have also computed accuracy and F1-score for each class separately to measure how well the selected models perform on each dataset.

5.2. Implementation Setup

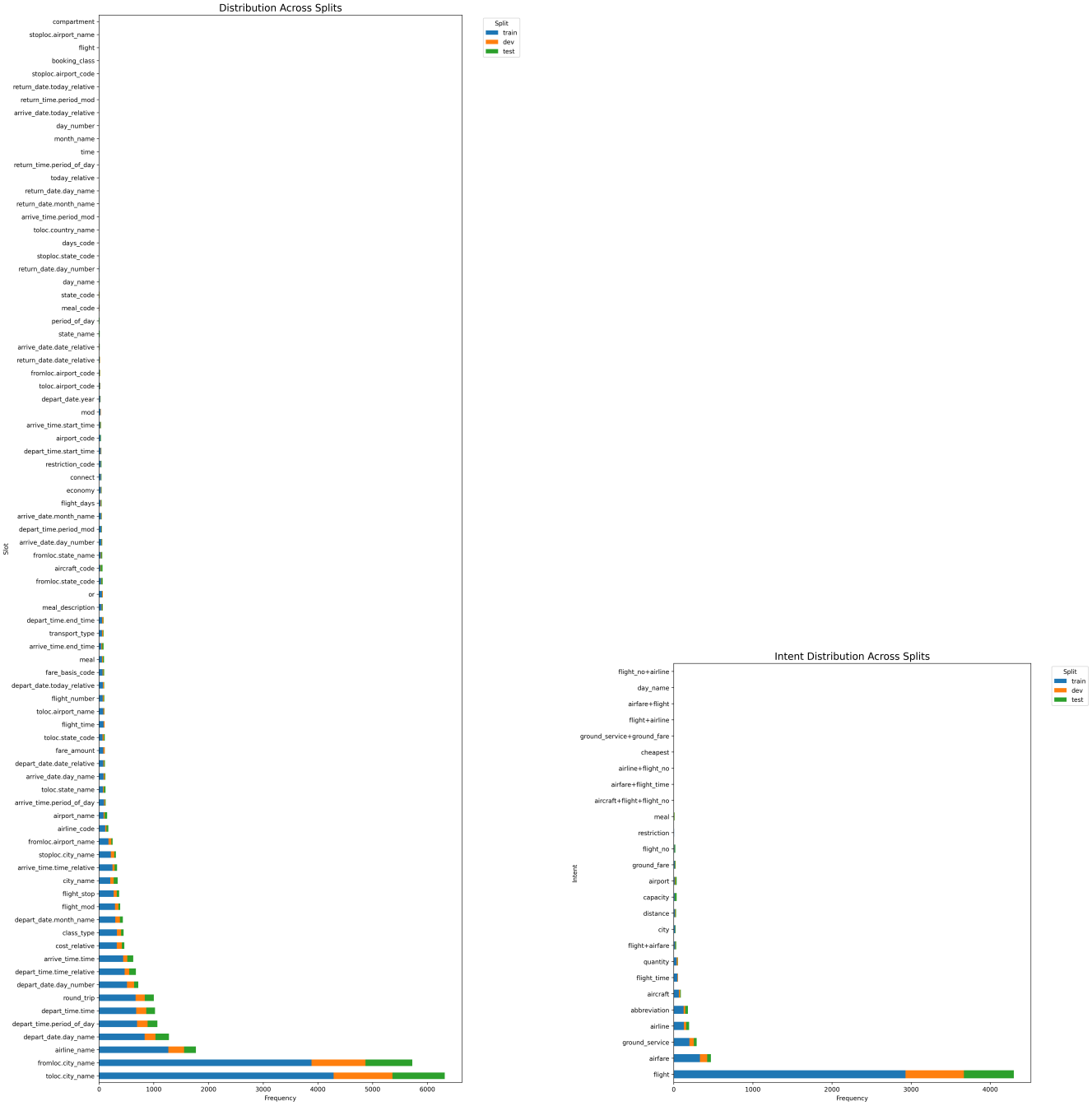
At first, we maintained the original test set of the Persian and English ATIS dataset and focused on creating a validation set from the training data. We split the original training set into new training and validation subsets, ensuring that the correspondence between English and Persian data was preserved throughout this process. Notably, the primary divergence from the original ATIS dataset lies in the inclusion of a validation set, which was not specified in the original dataset. The validation split comprises a randomly selected 20% of the original training set, while the remaining 80% constitutes the new training set. The test set remains unchanged from the original ATIS dataset. Table 1 and Figure 4 illustrate the distribution of data in each split. Given that the original ATIS is imbalanced based on intent classes and slot labels, the test set contains 20 intent categories out of 26 intent classes and 69 slot categories out of 83 slot labels. The categories of intent and slot in the test set are shown in Tables 3 and 4, respectively. Furthermore, the experiments' configuration for each model remains consistent with the original setup, and we utilized the available codes provided by the authors.

Table 1: Comparison of English and Persian ATIS datasets

Metric	English ATIS	Persian ATIS
Train samples	3982	3982
Test samples	893	893
Validation samples	996	996
Vocabulary size	889	1433
Slot tags (I- prefix)	3985	5891

5.3. Discussion

In the following sections, we perform a thorough analysis of the results comparing the performance of models trained on Persian and English datasets. In addition, we provide guidelines for further development ideas, which should help researchers increase the accuracy and stability of the model. The following research recommendations have been developed based on the objectives and conclusions of this study to inform the direction of future research in this field.



(a) Distribution of slots over splits

(b) Distribution of intents over splits

Figure 4: Statistics of the ATIS dataset

5.3.1. Intent Detection

The evaluation of the ATIS dataset's English and Persian versions produced intriguing findings about the intent detection capabilities of the models listed in Table 2. The CTran model achieved an outstanding accuracy of 97.98% on the English dataset, leveraging pre-trained BERT embeddings. JointBERT proved to be the top-performing

Table 2: Intent detection accuracy percent and slot filling f1-score percent of state-of-the-art models on English and Persian ATIS dataset

Taxonomy	Model	Intent Acc (%)		Slot F1-score (%)	
		English	Persian	English	Persian
Single	CNN-LSTM-CRF [15]	93.62	89.70	94.46	88.41
Joint	CNN-LSTM-CRF [15]	93.73	91.83	85.31	81.68
	Attention RNN [20]	93.84	90.93	95.59	87.96
	Slot-Gated [13]	94.62	92.94	94.91	87.70
	SF-ID,SF-first [10]	96.65	92.38	94.65	85.38
	SF-ID+CRF,SF-first [10]	97.31	92.83	94.72	85.56
	SF-ID,ID-first [10]	97.09	92.83	95.06	85.32
	SF-ID+CRF,ID-first [10]	95.41	92.05	94.55	85.57
	Co-Interactive transformer (Glove) [25]	97.54	93.73	95.69	86.14
Pre-trained	JointBERT [4]	97.42	**97.65**	95.20	**96.75**
	CTran [28]	**97.98**	96.86	81.89	75.33
Pre-trained + Cross-Lingual	Code-Switching [17]	97.42	79.39	**96.86**	38.74

model for the Persian dataset, with an impressive accuracy of 97.65%. Examining the confusion matrices (Figure 5 and Figure 6) reveals some interesting patterns:

- **Common misclassifications**: Across both languages, the most frequent misclassification occurs between "flight" and "airfare" intents. This suggests that these two categories share similar contextual features, making them challenging to distinguish.
- **Language-specific errors**: In Persian, we observe more confusion between "airport" and "flight" intents compared to English. This could be due to linguistic nuances in Persian that make these concepts more closely related in expression.
- **Low-frequency intent performance**: As shown in Table 3, there's a noticeable decrease in accuracy as the frequency of intent classes decreases. This trend is consistent across both languages, indicating a bias towards more common intents in the learning process.
- **Asymmetric errors**: In both languages, certain intent pairs show asymmetric confusion. For example, "ground_service" is more often misclassified as "ground_fare" than vice versa. This suggests that some intents may have more distinctive features than others.

The Code-Switching method performed well on the English dataset (97.42% accuracy) but struggled with Persian (79.39% accuracy). This significant drop in performance indicates that the method's effectiveness is limited when applied to Persian without specific adaptation.

Co-Interactive Transformer secured the second-highest accuracy in English (97.54%) but faced challenges in Persian, achieving 93.73% accuracy. This performance gap might be attributed to the difference in embedding dimensions: 50-dimensional Persian Glove embeddings versus 300-dimensional English counterparts, potentially resulting in less separable representations for Persian. The overall lower performance on the Persian dataset could be due to several factors:

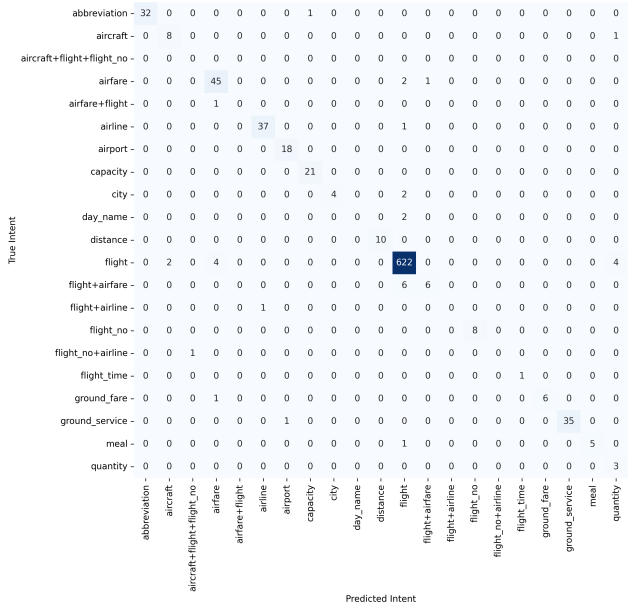
- **Linguistic complexity**: Persian's syntactic and morphological differences from English may pose challenges for models originally optimized for English.
- **Embedding quality**: The reliance on embeddings initially developed for English may not capture the nuances of Persian as effectively.
- **Data characteristics**: The translation process might have introduced subtle changes in the way intents are expressed, potentially making some distinctions less clear in Persian.

To improve Persian intent detection, future work should focus on:

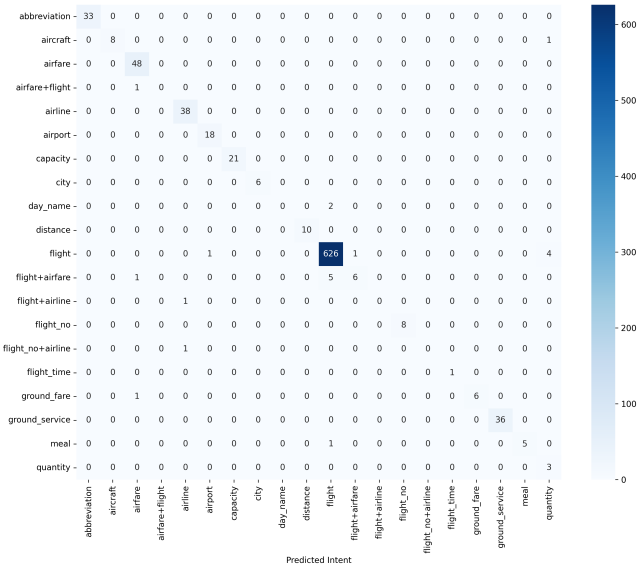
- Developing more robust Persian-specific embeddings

Table 3: Comparison of the best methods in intent detection on both Persian and English datasets; numbers show the accuracy of each method

Intent Categories	CTran		JointBERT		Intent Categories	CTran		JointBERT	
	English	Persian	English	Persian		English	Persian	English	Persian
abbreviation	1.0	0.97	1.0	1.0	ground_service	1.0	1.0	1.0	1.0
flight_no	1.0	1.0	1.0	1.0	flight_time	1.0	1.0	1.0	1.0
quantity	1.0	1.0	1.0	1.0	capacity	1.0	1.0	1.0	1.0
airline	1.0	0.97	1.0	0.95	airfare	1.0	1.0	1.0	0.94
distance	1.0	0.80	1.0	0.60	flight	0.99	0.98	0.99	0.97
airport	0.89	1.0	0.94	0.61	aircraft	1.0	0.89	0.89	0.78
meal	1.0	0.83	0.83	0.67	ground_fare	1.0	0.86	0.71	0.71
city	0.67	0.89	0.67	0.50	flight+airfare	0.67	0.67	0.67	0.25
flight_no+airline	1.0	0.97	1.0	1.0	flight+airline	0.0	0.25	0.0	0.0
day_name	1.0	1.0	0.0	0.0	airfare+flight	0.0	0.0	0.0	0.0



(a) Confusion matrix for intent detection (Persian)



(b) Confusion matrix for intent detection (English)

Figure 5: Confusion matrices for intent detection for the CTran. (a) Persian (b) English

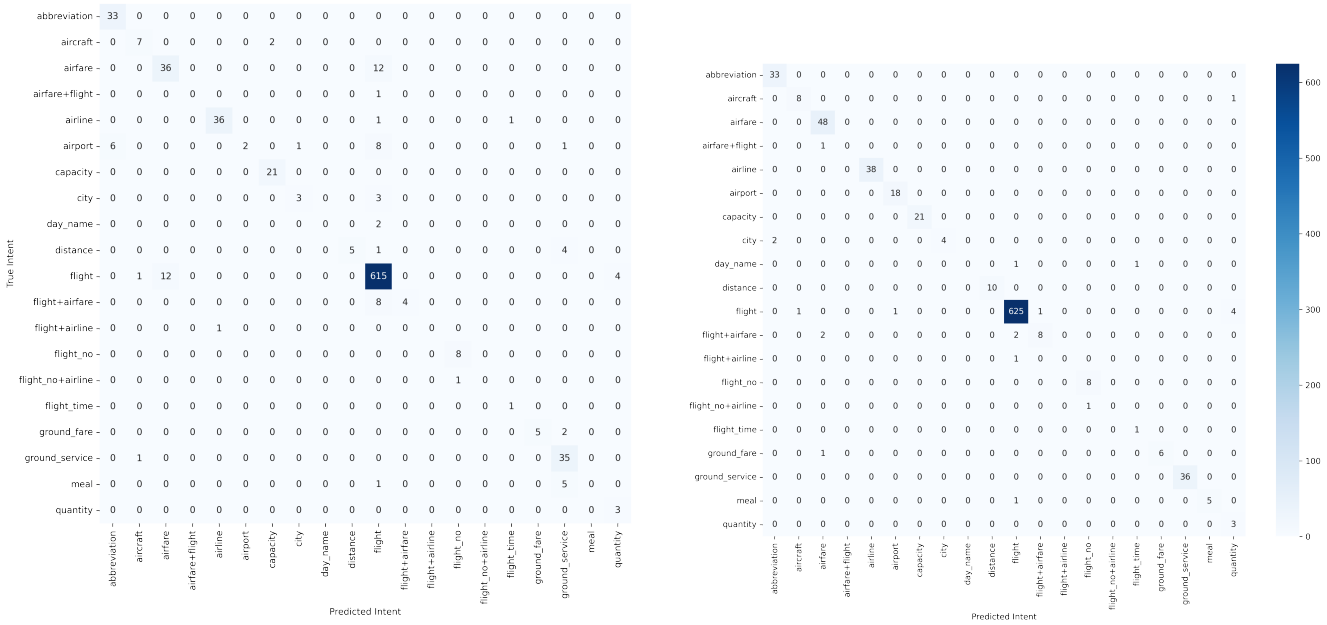
- Adapting model architectures to better handle Persian linguistic features
- Augmenting the dataset with more examples of low-frequency intents to reduce bias

5.3.2. Slot Filling

The comparative analysis of slot filling performance between English and Persian ATIS datasets reveals significant insights into the challenges and opportunities in cross-lingual natural language understanding tasks.

The Persian ATIS dataset, while maintaining the same number of samples as its English counterpart, exhibits a larger vocabulary size and a higher number of slot tags with I-prefix (Table 1). This increased lexical diversity and more complex word-to-slot mappings in Persian underscore the linguistic intricacies that models must navigate. The structural differences between Persian, an SOV (Subject-Object-Verb) language, and English (SVO) further compound these challenges, potentially affecting slot boundary identification and overall model performance.

The results presented in Table 2 demonstrate a consistent pattern of superior performance on the English dataset compared to Persian across most models. JointBERT emerged as the top performer for Persian slot filling with an F1-score of 96.75%, closely followed by its 95.20% score on English. Notably, CNN-LSTM-CRF exhibited unexpected strength in Persian (F1-score: 88.41%), outperforming some more sophisticated architectures. This suggests that certain architectural elements, such as the use of CNNs for capturing morphological features, may be



(a) Confusion matrix for intent detection (Persian)

(b) Confusion matrix for intent detection (English)

Figure 6: Confusion matrices for intent detection for the JointBERT. (a) Persian (b) English

particularly beneficial for Persian language processing [5].

The stark contrast in the Code-Switching model's performance between English (96.86% F1-score) and Persian (38.74% F1-score) highlights the limitations of cross-lingual approaches relying on machine translation for Persian. This disparity underscores the need for language-specific adaptations in transfer learning techniques, especially for languages with distinct linguistic properties.

A granular examination of slot-wise performance (Table 4) reveals nuanced patterns in model behavior across languages:

- **Language-Agnostic Success**: Simple, high-frequency slots such as "fare_amount", "airline_name", and "toloc.city_name" consistently achieve high F1-scores across both languages and models. This suggests that certain semantic concepts transfer well between languages, regardless of syntactic differences.
- **Persian-Specific Challenges**: Time-related slots (e.g., "depart_time.time", "arrive_time.period_of_day") and location-related slots (e.g., "fromloc.airport_code", "toloc.airport_name") show markedly lower performance in Persian compared to English. This discrepancy may be attributed to:
  - The complex morphological structure of Persian, which can obscure slot boundaries.
  - Differences in expressing time and location concepts between the two languages.
  - The absence of capitalization in Persian, which may impact the identification of named entities.
- **Model-Specific Observations**: JointBERT consistently outperforms the Code-Switching model on Persian slots, particularly for location and time-related information. This suggests that JointBERT's architecture is better suited to capturing Persian-specific linguistic features, possibly due to its more effective use of contextual information.

The performance discrepancies observed between English and Persian can be attributed to several linguistic factors:

- **Morphological Complexity**: Persian's rich morphological structure poses challenges in identifying slot boundaries, particularly for time and location entities. This is evidenced by the lower performance on slots like "depart_time.time" and "toloc.airport_name" in Persian.
- **Syntactic Differences**: The SOV structure of Persian, contrasting with English's SVO order, may affect models' ability to correctly identify certain slots, particularly those related to action or movement (e.g., departure and arrival information).

Table 4: Comparison of the best methods in slot filling on both Persian and English datasets; numbers show the F1-score of each method

Slot Categories	Code-Switching		JointBERT		Slot Categories	Code-Switching		JointBERT	
	English	Persian	English	Persian		English	Persian	English	Persian
fare_amount	1.0	1.0	1.0	1.0	state_code	1.0	0.0	1.0	1.0
days_code	0.0	0.0	1.0	1.0	fromloc.city_name	0.99	0.40	0.99	1.0
class_type	1.0	0.02	0.98	1.0	connect	1.0	0.0	1.0	1.0
depart_time.time_relative	0.96	0.05	0.97	1.0	economy	1.0	0.43	1.0	1.0
toloc.state_name	0.91	0.13	0.93	1.0	arrive_time.time_relative	0.82	0.11	0.85	1.0
arrive_time.end_time	0.94	0.0	0.89	1.0	arrive_time.time	96	0.05	0.96	1.0
fromloc.airport_code	0.66	0.15	0.71	1.0	toloc.country_name	1.0	0.0	1.0	1.0
arrive_date.month_name	0.77	0.0	0.77	1.0	period_of_day	0.57	0.0	0.67	1.0
airline_name	0.97	0.1	0.97	1.0	city_name	0.67	0.21	0.67	0.99
toloc.city_name	0.98	0.39	0.98	0.99	airport_name	0.56	0.04	0.51	0.99
depart_date.day_name	0.99	0.24	0.99	0.98	depart_date.day_number	0.97	0.22	0.96	0.97
depart_time.period_mod	0.99	0.14	0.91	0.97	cost_relative	0.98	0.19	0.99	0.97
depart_time.time	0.92	0.19	0.92	0.97	depart_date.year	1.0	0.91	1.0	0.97
arrive_time.period_of_day	0.75	0.17	0.86	0.96	arrive_time.start_time	0.94	0.0	0.82	0.96
depart_date.today_relative	0.88	0.0	0.89	0.96	depart_time.start_time	0.80	0.27	0.80	0.92
toloc.airport_name	1.0	0.0	1.0	0.92	arrive_date.date_relative	1.0	0.0	0.80	0.91
flight_days	1.0	0.0	1.0	0.91	depart_date.date_relative	0.97	0.0	1.0	0.90
toloc.airport_code	0.86	0.07	0.67	0.90	fromloc.state_code	1.0	0.0	1.0	0.80
arrive_date.day_name	0.85	0.0	0.85	0.78	fromloc.state_name	0.97	0.04	0.97	0.77
meal_code	0.0	0.0	0.0	0.75	flight_mod	0.79	0.07	0.70	0.71
stoploc.city_name	1.0	0.18	1.0	0.68	return_date.date_relative	0.0	0.0	0.0	0.67
round_trip	0.99	0.0	0.0	0.67	flight_number	0.92	0.62	0.77	0.63
meal_description	1.0	0.06	1.0	0.63	depart_date.month_name	0.97	0.54	0.97	0.51
flight_stop	1.0	0.0	1.0	0.0	depart_time.period_of_day	0.94	0.24	0.96	0.0
transport_type	0.93	0.11	0.90	0.0	fromloc.airport_name	0.63	0.0	0.61	0.0
flight_time	1.0	0.0	1.0	0.0	toloc.state_code	1.0	0.0	1.0	0.0
meal	0.94	0.0	0.97	0.0	airline_code	0.93	0.20	0.93	0.0
aircraft_code	0.96	0.76	0.92	0.0	fare_basis_code	0.92	0.0	0.91	0.0
restriction_code	0.91	0.0	0.89	0.0	depart_time.end_time	0.4	0.11	0.80	0.0
arrive_date.day_number	0.77	0.0	0.77	0.0	or	0.67	0.5	0.75	0.0
airport_code	0.57	0.20	0.50	0.0	day_name	0.3	0.0	0.0	0.0
state_name	0.98	0.0	0.0	0.0	booking_class	0.0	0.0	0.0	0.0
compartment	0.0	0.0	0.0	0.0	flight	0.0	0.0	0.0	0.0
return_date.day_name	0.0	0.0	0.0	0.0	stoploc.airport_code	0.0	0.0	0.0	0.0

- **Orthographic Features**: The lack of capitalization in Persian for proper nouns potentially impacts the identification of named entities, as seen in the lower performance on location-related slots.
- **Cross-lingual Transfer Limitations**: The poor performance of the Code-Switching model on Persian (38.74% F1-score) indicates that simple machine translation-based approaches are inadequate for capturing the nuances of Persian. This highlights the need for more sophisticated cross-lingual transfer methods that can preserve crucial linguistic information.

Common error patterns observed include confusion between similar slots (e.g., "toloc.airport_code" vs. "fromloc.airport_code"), boundary detection issues in multi-word expressions, and poor performance on rare slots like "aircraft_code" or "restriction_code", particularly in Persian.

To address these challenges and improve slot filling performance in Persian, following directions can be considered:

- Development of Persian-specific pre-training techniques to better capture the language's unique morphological and syntactic features.
- Data augmentation focusing on challenging slots, particularly time and location-related expressions in Persian.
- Exploration of character-level or subword-level tokenization strategies to better handle Persian's complex morphology.

- Integration of Persian-specific linguistic features or rules to enhance model performance on challenging slots.
- Investigation into more advanced cross-lingual transfer learning methods that can effectively bridge the linguistic gap between source and target languages.

In conclusion, while state-of-the-art models demonstrate impressive performance on both English and Persian ATIS datasets, significant challenges remain in achieving parity across languages. The insights gained from this comparative analysis underscore the importance of language-specific considerations in the development of multilingual NLU systems and provide a roadmap for future improvements in Persian language processing tasks.

6. Conclusion

Research into natural Persian language processing has gotten more challenging due to the size of the number of individuals who speak Persian, as well as the lack of datasets devoted to intent recognition and slot filling in Persian. In order to address this issue, we presented a publicly available benchmark in the Persian language for intent detection and slot filling. We then investigated the state-of-the-art models for intent detection and slot filling to apply them to our recently founded benchmark and explain how they perform on this particular dataset.

We hope that the newly developed Persian benchmark for joint intent detection and slot filling will provide valuable training data as well as a novel and demanding testing ground for NLU models. Building upon the existing foundation, we propose the following avenues for future research and development: It is possible to consider more accurate tokenization, for example, character-level or subword-level to deal with Persian morphologically. Furthermore, modification of the model structures to incorporate Persian-related characteristics would possibly provide improved results. Last but not least, increasing the number of examples of low frequent intents in the input dataset would also contribute to the reduction of bias and increase the generalization of the research.

Moreover, this dataset provides scope for a number of future research possibilities. These include studying many methods of transfer learning across languages with Persian as either the source or the target language for improving the performance of low-resource languages. Also, the investigation of the events of different pre-training strategies on downstream efficiency in Persian NLU, or compare simple and multilingual models as well. Furthermore, as future work, it is possible to extend this research to include few-shot and zero-shot learning approaches for intent detection and slot filling in Persian with the potential of other Persian dialects and other languages in general.

Acknowledgement

We want to express our great appreciation to Shahrzad Zolghadr, Fatemeh Ghoochani, Fatemeh Inanloo, and Sepehr Moradi, who helped and gave us their time to facilitate reaching the goals of this research.

References

[1] H. Amirkhani, M. AzariJafari, S. Faridan-Jahromi, Z. Kouhkan, Z. Pourjafari, and A. Amirak, *FarsTail: a persian natural language inference dataset*, Soft Computing, 30 (2026), pp. 891–903.

[2] W. Antoun, F. Baly, and H. Hajj, *AraBERT: Transformer-based model for Arabic language understanding*, 2020.

[3] M. Asgari-Bidhendi, M. Nasser, B. Janfada, and B. Minaei-Bidgoli, *PERLEX: A bilingual persian-english gold dataset for relation extraction*, Scientific Programming, 2021 (2021), p. 8893270.

[4] Q. Chen, Z. Zhuo, and W. Wang, *BERT for joint intent classification and slot filling*, arXiv preprint, (2019).

[5] J. P. Chiu and E. Nichols, *Named entity recognition with bidirectional lstm-cnns*, Transactions of the Association for Computational Linguistics, 4 (2016), pp. 357–370.

[6] M. H. Dao, T. H. Truong, and D. Q. Nguyen, *Intent detection and slot filling for Vietnamese*, in Interspeech 2021, 2021, pp. 4698–4702.

[7] K. Darvishi, N. Shahbodaghkhan, Z. Abbasiantaeb, and S. Momtazi, *PQuAD: A persian question answering dataset*, Computer Speech & Language, 80 (2023), p. 101486.

[8] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, *BERT: Pre-training of deep bidirectional transformers for language understanding*, in Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), J. Burstein, C. Doran, and T. Solorio, eds., Minneapolis, Minnesota, 2019, Association for Computational Linguistics, pp. 4171–4186.

[9] E. Doostmohammadi, M. H. Bokaei, and H. Sameti, *PerKey: A persian news corpus for keyphrase extraction and generation*, in 2018 9th International Symposium on Telecommunications (IST), 2018, pp. 460–465.

[10] H. E, P. Niu, Z. Chen, and M. Song, *A novel bi-directional interrelated model for joint intent detection and slot filling*, in Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, A. Korhonen, D. Traum, and L. Màrquez, eds., Florence, Italy, 2019, Association for Computational Linguistics, pp. 5467–5471.

[11] R. Etezadi and M. Shamsfard, *PeCoQ: A dataset for persian complex question answering over knowledge graph*, in 2020 11th International Conference on Information and Knowledge Technology (IKT), 2020, pp. 102–106.

[12] J. FitzGerald, C. Hench, C. Peris, S. Mackie, K. Rottmann, A. Sanchez, A. Nash, L. Urbach, V. Kakarala, R. Singh, S. Ranganath, L. Crist, M. Britan, W. Leeuwis, G. Tur, and P. Natarajan, *MASSIVE: A 1M-example multilingual natural language understanding dataset with 51 typologically-diverse languages*, in Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), A. Rogers, J. Boyd-Graber, and N. Okazaki, eds., Toronto, Canada, 2023, Association for Computational Linguistics, pp. 4277–4302.

[13] C.-W. Goo, G. Gao, Y.-K. Hsu, C.-L. Huo, T.-C. Chen, K.-W. Hsu, and Y.-N. Chen, *Slot-gated modeling for joint slot filling and intent prediction*, in Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers), M. Walker, H. Ji, and A. Stent, eds., New Orleans, Louisiana, 2018, Association for Computational Linguistics, pp. 753–757.

[14] B. Hu, Q. Chen, and F. Zhu, *LCSTS: A large scale Chinese short text summarization dataset*, in Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, L. Màrquez, C. Callison-Burch, and J. Su, eds., Lisbon, Portugal, 2015, Association for Computational Linguistics, pp. 1967–1972.

[15] B. Kane, F. Rossi, O. Guinaudeau, V. Chiesa, I. Quénel, and S. Chau, *Joint intent detection and slot filling via CNN-LSTM-CRF*, in 2020 6th IEEE Congress on Information Science and Technology (CiSt), 2020, pp. 342–347.

[16] A. Kazemi, J. Mozafari, and M. A. Nematbakhsh, *PersianQuAD: The native question answering dataset for the persian language*, IEEE Access, 10 (2022), pp. 26045–26057.

[17] J. Krishnan, A. Anastasopoulos, H. Purohit, and H. Rangwala, *Multilingual code-switching for zero-shot cross-lingual intent prediction and slot filling*, in Proceedings of the 1st Workshop on Multilingual Representation Learning, D. Ataman, A. Birch, A. Conneau, O. Firat, S. Ruder, and G. G. Sahin, eds., Punta Cana, Dominican Republic, Nov. 2021, Association for Computational Linguistics, pp. 211–223.

[18] I. Lauriola, A. Lavelli, and F. Aiolli, *An introduction to deep learning in natural language processing: Models, techniques, and tools*, Neurocomputing, 470 (2022), pp. 443–456.

[19] C.-H. Lee, S.-M. Wang, H.-C. Chang, and H.-Y. Lee, *ODSQA: Open-domain spoken question answering dataset*, in 2018 IEEE Spoken Language Technology Workshop (SLT), 2018, pp. 949–956.

[20] B. Liu and I. Lane, *Attention-based recurrent neural network models for joint intent detection and slot filling*, in Interspeech 2016, 2016, pp. 685–689.

[21] W. Liu, J. Tang, Y. Cheng, W. Li, Y. Zheng, and X. Liang, *MedDG: An entity-centric medical consultation dataset for entity-aware medical dialogue generation*, in Natural Language Processing and Chinese Computing, W. Lu, S. Huang, Y. Hong, and X. Zhou, eds., Cham, 2022, Springer International Publishing, pp. 447–459.

[22] G. Mesnil, Y. Dauphin, K. Yao, Y. Bengio, L. Deng, D. Hakkani-Tur, X. He, L. Heck, G. Tur, D. Yu, and G. Zweig, *Using recurrent neural networks for slot filling in spoken language understanding*, IEEE/ACM Transactions on Audio, Speech, and Language Processing, 23 (2015), pp. 530–539.

[23] P. Ni, Y. Li, G. Li, and V. Chang, *Natural language understanding approaches based on joint task of intent detection and slot filling for IoT voice interaction*, Neural Comput. & Appl., 32 (2020), pp. 16149–16166.

[24] W. Pan, Q. Chen, X. Xu, W. Che, and L. Qin, *A preliminary evaluation of ChatGPT for zero-shot dialogue understanding*, 2023. arXiv preprint, arXiv:2304.04256.

[25] L. Qin, T. Liu, W. Che, B. Kang, S. Zhao, and T. Liu, *A co-interactive transformer for joint slot filling and intent detection*, in ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2021, pp. 8193–8197.

[26] L. Qin, T. Xie, W. Che, and T. Liu, *A survey on spoken language understanding: Recent advances and new frontiers*, in Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI-21, Z.-H. Zhou, ed., International Joint Conferences on Artificial Intelligence Organization, 8 2021, pp. 4577–4584. Survey Track.

[27] J. Quan, S. Zhang, Q. Cao, Z. Li, and D. Xiong, *RiSAWOZ: A large-scale multi-domain Wizard-of-Oz dataset with rich semantic annotations for task-oriented dialogue modeling*, in Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), B. Webber, T. Cohn, Y. He, and Y. Liu, eds., Online, 2020, Association for Computational Linguistics, pp. 930–940.

[28] M. Rafiepour and J. S. Sartakhti, *CTRAN: CNN-transformer-based network for natural language understanding*, Engineering Applications of Artificial Intelligence, 126 (2023), p. 107013.

[29] M. Roman, A. Shahid, S. Khan, A. Koubaa, and L. Yu, *Citation intent classification using word embedding*, IEEE Access, 9 (2021), pp. 9982–9995.

[30] R. Sarikaya, G. E. Hinton, and A. Deoras, *Application of deep belief networks for natural language understanding*, IEEE/ACM Transactions on Audio, Speech, and Language Processing, 22 (2014), pp. 778–784.

[31] J. P. R. Sharami, P. A. Sarabestani, and S. A. Mirroshandel, *DeepSentiPers: Novel deep learning models trained over proposed augmented persian sentiment corpus*, 2020. arXiv preprint.

[32] A. Stoica, T. Kadar, C. Lemnaru, R. Potolea, and M. Dînşoreanu, *Intent detection and slot filling with capsule net architectures for a romanian home assistant*, Sensors, 21 (2021), p. 1230.

[33] K. Sun, D. Yu, D. Yu, and C. Cardie, *Investigating prior knowledge for challenging chinese machine reading comprehension*, Transactions of the Association for Computational Linguistics, 8 (2020), pp. 141–155.

[34] Y. Tang, C. Tran, X. Li, P.-J. Chen, N. Goyal, V. Chaudhary, J. Gu, and A. Fan, *Multilingual translation from denoising pre-training*, in Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021, C. Zong, F. Xia, W. Li, and R. Navigli, eds., Online, 2021, Association for Computational Linguistics, pp. 3450–3466.

[35] G. Tur, D. Hakkani-Tür, and L. Heck, *What is left to be understood in ATIS?*, in 2010 IEEE Spoken Language Technology Workshop, Berkeley, CA, USA, 2010, pp. 19–24.

[36] S. Upadhyay, M. Faruqui, G. Tür, H.-T. Dilek, and L. Heck, *(almost) zero-shot cross-lingual spoken language understanding*, in 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2018, pp. 6034–6038.

[37] H. Weld, X. Huang, S. Long, J. Poon, and S. C. Han, *A survey of joint intent detection and slot filling models in natural language understanding*, ACM Comput. Surv., 55 (2022), pp. 1–38.

[38] L. Xu, H. Hu, X. Zhang, L. Li, C. Cao, Y. Li, Y. Xu, K. Sun, D. Yu, C. Yu, Y. Tian, Q. Dong, W. Liu, B. Shi, Y. Cui, J. Li, J. Zeng, R. Wang, W. Xie, Y. Li, Y. Patterson, Z. Tian, Y. Zhang, H. Zhou, S. Liu, Z. Zhao, Q. Zhao, C. Yue, X. Zhang, Z. Yang, K. Richardson, and Z. Lan, *CLUE: A Chinese language understanding evaluation benchmark*, in Proceedings of the 28th International Conference on Computational Linguistics, D. Scott, N. Bel, and C. Zong, eds., Barcelona, Spain (Online), 2020, International Committee on Computational Linguistics, pp. 4762–4772.

[39] W. Xu, B. Haider, and S. Mansour, *End-to-end slot alignment and recognition for cross-lingual NLU*, in Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), B. Webber, T. Cohn, Y. He, and Y. Liu, eds., Online, 2020, Association for Computational Linguistics, pp. 5052–5063.

[40] G. Zeng, W. Yang, Z. Ju, Y. Yang, S. Wang, R. Zhang, M. Zhou, J. Zeng, X. Dong, R. Zhang, H. Fang, P. Zhu, S. Chen, and P. Xie, *MedDialog: Large-scale medical dialogue datasets*, in Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), B. Webber, T. Cohn, Y. He, and Y. Liu, eds., Association for Computational Linguistics, 2020, pp. 9241–9250.

[41] J. Zhu, Q. Wang, Y. Wang, Y. Zhou, J. Zhang, S. Wang, and C. Zong, *NCLS: Neural cross-lingual summarization*, in Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), K. Inui, J. Jiang, V. Ng, and X. Wan, eds., Hong Kong, China, 2019, Association for Computational Linguistics, pp. 3054–3064.

[42] S. Zhu, Z. Zhao, R. Ma, and K. Yu, *Prior knowledge driven label embedding for slot filling in natural language understanding*, IEEE/ACM Transactions on Audio, Speech, and Language Processing, 28 (2020), pp. 1440–1451.

Please cite this article using:

Masoud Akbari, Amir Hossein Karimi, Tayyebeh Saeedi, Zeinab Saeidi, Kiana Ghezelbash, Fatemeh Shamsezat, Mohammad Akbari, Ali Mohades Khorasani, A persian benchmark for joint intent detection and slot filling, AUT J. Math. Comput., 7(4) (2026) 407-422
https://doi.org/10.22060/AJMC.2025.23058.1225