ISO-TimeML Event Extraction in Persian Text

Yadollah Yaghoobzadeh[1] *Gholamreza Ghassem-sani*[1] *Seyed Abolghasem Mirroshandel*[2] *Mahbaneh Eshaghzadeh*[1]

(1) Department of Computer Engineering, Sharif University of Technology, Tehran, Iran
(2) Faculty of Engineering, University of Guilan, Rasht, Iran

`yaghoobzadeh@ce.sharif.edu, sani@sharif.edu,`
`mirroshandel@guilan.ac.ir,` *`eshaghzadeh`*`@ce.sharif.edu`

ABSTRACT

Recognizing TimeML events and identifying their attributes, are important tasks in natural language processing (NLP). Several NLP applications like question answering, information retrieval, summarization, and temporal information extraction need to have some knowledge about events of the input documents. Existing methods developed for this task are restricted to limited number of languages, and for many other languages including Persian, there has not been any effort yet. In this paper, we introduce two different approaches for automatic event recognition and classification in Persian. For this purpose, a corpus of events has been built based on a specific version of ISO-TimeML for Persian. We present the specification of this corpus together with the results of applying mentioned approaches to the corpus. Considering these methods are the first effort towards Persian event extraction, the results are comparable to that of successful methods in English.

TITLE AND ABSTRACT IN PERSIAN

استخراج رویدادها از متون فارسی بنا بر تعریف ISO-TimeML

یافتن رویدادها و ویژگی‌های آنها بر اساس TimeML یکی از مسائل مهم در حوزه‌ی پردازش زبان های طبیعی است. بسیاری از کاربردهای پردازش زبان‌های طبیعی مانند سامانه‌های پرسش و پاسخ، استخراج اطلاعات، خلاصه سازی و استخراج اطلاعات زمانی نیاز دارند تا دانشی درباره رویدادهای موجود در متون ورودی داشته باشند. روش‌هایی که تاکنون در مورد این مسئله ایجاد شده، محدود به چند زبان خاص است و در بسیاری از زبان‌ها از جمله زبان فارسی، تاکنون کاری در این رابطه انجام نشده است. در این مقاله، ما دو روش مختلف برای استخراج رویدادها در زبان فارسی ارائه می‌دهیم. برای این کار، پیکره‌ای مطابق با ISO-TimeML، البته نسخه‌ی خاص فارسی آن، ساخته شد. ما مشخصات این پیکره و نتایج حاصل بر روی آن، را نشان می‌دهیم. نتایج روش‌های ارائه‌شده در این مقاله، به عنوان اولین روش‌های پیاده‌سازی شده بر روی زبان فارسی، با روش های موفق در زبان انگلیسی قابل مقایسه است.

KEYWORDS : Event Mention, Temporal Information Extraction, Classification, Annotation Scheme, TimeBank, ISO-TimeML, Persian Language.

KEYWORDS IN PERSIAN : ذکر رویداد، استخراج اطلاعات زمانی، رده بندی، شمای برچسب‌گذاری، تایم بانک، استاندارد تایم ام ال، زبان فارسی

Proceedings of COLING 2012: Technical Papers, pages 2931–2944,
COLING 2012, Mumbai, December 2012.

1 Introduction

Event extraction is a demanding task in natural language processing (NLP). Several applications like question answering (QA), information retrieval (IR), summarization, and temporal information extraction need to have some knowledge about events for better operation. This task remains a challenging task. Recognizing the various forms in which an event may be expressed (verbs, nouns, adjectives and prepositions), distinguishing events of different classes, and finding the features of an event are all difficult task (Verhagen et al., 2010).

In this paper, the events are defined based on TimeML view. In TimeML, events are "*situations* that *occur* or *happen*, or predicates that describe *states* or *circumstances* in which something obtains or holds the truth" (Pustejovsky et al., 2003). The task of event extraction includes two major stages that are introduced by TimeML: 1) detection and annotation of a text span (i.e., verbs, nouns, predicative constructions, prepositional phrases, and adjectival phrases) that is an event, and 2) determining the semantic class of events (i.e., *Reporting*, *Perception*, *I_Action*, *I_State*, *State*, *Occurrence*, and *Aspectual*).

Performing these two tasks (event mention detection and classification) in any language requires a corpus of annotated events, at least for measuring the accuracy of the algorithm. Currently, there are no such corpora for many languages including Persian, which is the native natural language of Iran, Afghanistan and Tajikistan. We have developed a corpus of annotated events in order to extract events from Persian texts. This corpus contains 4237 events. The annotation process has been based on an adapted version of ISO-TimeML guidelines. We have applied some changes to event attributes, the value of these attributes, annotation rules, and event extents.

In this paper, we also propose a system for automatic event recognition. In the system, various morphological, syntactic, and semantic features have been used. The syntactic features are in the form of dependency parse trees. Semantic features are taken from a Persian version of WordNet. For identification of event mentions and classifying them, the system uses these features in two different methods: a rule-based and a learning-based method. In the rule-based method, we proposed several rules for different types of events. In the learning-based method, a classification technique has been used for identification of events. We have used different models for different forms of events (i.e., verb, noun, and adjective). Our experiments show that the proposed methods, which are the first attempt in Persian event extraction, are quite effective.

The remainder of this paper is organized as follows: section 2 is about ISO-TimeML adaptation for Persian. Section 3 explains previous work in event extraction. Our event extraction system is proposed in section 4. The experimental results of the system are presented in section 5. Finally, the last section of the paper includes conclusion and some possible future work.

2 Adapting ISO-TimeML Event Guidelines for Persian

To apply ISO-TimeML as an annotation scheme to a new language, the language specific issues should be considered carefully. Accordingly, some aspects of scheme must be modified and some others must be restated for target language. The adapted schema may go through various changes regarding event attributes, event attribute values, event annotation rules and event extent rules according to the target language structure.

A number of languages including Korean (Im et al., 2009), Italian (Caselli et al., 2011), French (Bittar et al., 2011) have already adapted TimeML and ISO-TimeML guidelines to their needs. In the following subsections, we present the adapted version of ISO-TimeML for Persian (let us call it PersTimeML) in three main categories: event annotation, event extent and event attributes.

2.1 Event Annotation

Event annotation in PersTimeML is mostly based on ISO-TimeML. Generally, for simplicity it is assumed in current version of PersTimeML that generic events must be annotated. Also there have been special cases that have been tailored particularly for the specific properties of Persian. Here, we only discuss these cases disregarding common situations with ISO-TimeML.

2.1.1 Nouns

In Persian, gerund phrases, known as "esm-e masdar", must always be annotated as events, even when they represent generic events. These are built by affixing a particular Persian letter, i.e. "nôn", to the verb stem. There are also some categories of nouns that function like gerund phrases but do not have any lexical mark. These nouns were named *predicative nouns* and defined as nouns that inherit some verb and some noun characteristics (Karimi-Doostan, 2011). In Persian sentences, these nouns are usually the starting point of an NP or a PP. We always annotate these nouns as events, too. Following examples are instances of these cases:

a. **Barresê-e** (Review) maqâle-hâ (papers) chand (a few) rôz (day) tôl mikeshad (takes).
   Translation: **Reviewing** of the papers takes a few days.
b. Ostâd (Instructor) bâ (with) **taavêq-e** (postpone) emtehân (exam) mokhâlefat kard (disagreed).
   Translation: The professor disagreed with **postponing** the exam.
c. Alê (Ali) be (to) **jostojô** (search) dar miân-e (through) sâyt (site) edâme dâd (continued).
   Translation: Ali continued to **search** in the site.

In examples (a) and (b), "barresêy-e" (review) and "taavêq-e" (postpone) are *predicative nouns*, when have an "e" mark in their end and are linked to their subsequent nouns. In example (c), "jostojô" (search) is followed by "be" (in) as a preposition.

2.1.2 Adjectives

In addition to TimeML guidelines for annotating adjectives, we must also consider *objective deverbal adjectives* in PersTimeML. These are adjectives that derived from passive modes of verbs (Lesani, 2003). Two examples are "neveshte shode" (written) and "gerefte shode" (taken). *Objective deverbal adjectives* translate to the past participle form of the verbs in English. These adjectives always must be annotated as events because they are implying verbal events that have occurred in the past. For example, in the following sentence the "neveshte shode" (written) is an *objective deverbal adjective* that must be tagged as an event.

a. Ô (He) dastân-e (story) **neveshte shode** (written) dar (in) ân (that) rôzname (newspaper) râ bâvar nakard (didn't believe).
   Translation: He didn't believe the **written** story in that newspaper.

2.2 Event Extents

According to the new paradigm in ISO-TimeML about the stand-off annotation instead of in-line annotation (Pustejovsky et al., 2010), we can annotate multiple tokens in an event tag even though the tokens are not located consecutively in the sentence. This new approach simplifies the handling of compound words (e.g. compound verbs in Persian) by tagging all the associated tokens as just one event. The majority of the Persian verbs are compound. Persian compound verbs consist of a light verb and a number of non-verbal elements. For example, "barkhord kardan" (to hit) is a compound verb including the light verb of "kardan" (do) and the non-verbal element of "barkhord" (hit) (Rasooli et al., 2011).

Since there is not a fixed list of compound verbs, recognizing them in the sentences is difficult. Besides, detecting all parts of compound verbs can be challenging for an annotator because they may be located separately with long distances. The following examples show how we annotate compound verbs with event tags.

a. Bârân (Rain) be (to) mantaq-e (area) **sadame-e** (damage) zyâdê (large) **khâhad zad** (will do).
   Translation: The rain will largely damage the area.
   - Part of PersTimeML output will be:
     <Event xml:id="e1" target="#token3#token5" text= "sadame-e khâhad zad"… />

b. Bâ (with) oo (he) **sohbat** (talk) **kardam** (did).
   Translation: I talked with him.
   - Part of PersTimeML output will be:
     <Event xml:id="e1" target="#token2#token3" text=" sohbat kardam"… />

Segmentation of Persian verbs is another difficulty regarding annotation of verbs (Shamsfard, 2011). For tackling this problem, we need to pre-process the sentences to merge all Tense/Aspect/Mood (TMA) mark tokens with their verbal head. Therefore, in example (a) above, in the pre-processing, the whole "khâhad zad" (will do) will be merged as one token by inserting short space between the verb parts.

2.3 Event Attributes

Persian, as a natural language has its own specific aspects in event attributes. As we currently know, in Persian TMAs are not separated. However, we can map the TMA values to *tense*, *aspect* and *mood* appropriately to follow the ISO-TimeML guideline. According to this mapping, the possible values for each of these attributes will be: *tense***:** "past", "present", "future", "none", *aspect***:** "perfective", "progressive", "imperfective_perfective", "imperfective", "none" and *mood***:** "subjunctive", "imperative", "indicative", "none".

3 Automatic Event Extraction Methods

There are many ongoing researches on event extraction according to TimeML specification language. Almost all detection systems act in three following stages: 1) pre-processing; 2) event mention detection; and 3) event attributes detection. Existing methods can be divided into rule-based, statistical, and combined approaches, which are explained in more details in the next following subsections.

3.1 Rule-based Methods

One of the first event recognition systems in French used a rule-based method (Bittar, 2009). The system utilizes specific hand-made dictionaries for event detection and classification. Besides, a number of rules were used to reduce the errors. In other words, event recognition in this system includes two types of processing: lexical, using specific dictionaries and contextual, using a number of rules.

The first system on Italian also used a rule-based approach that utilized dependency parse trees of the input sentences (Robaldo et al., 2011). The rules of this system have designed based on both syntactic and lexical information using a number of keywords. Some extra resources like WordNet (Fellbaum, 1998) plus a list of specific words were used to build a comprehensive keyword list. To identify the class of events, in addition to dependency structure of sentences, a list of Italian verbs with three different semantic categories (state, process, and movement) has been also utilized.

Edinburg is another rule-based system that used a named entity recognition system for nominal event detection. Special lists, which were extracted from an annotated corpus, in conjunction with WordNet were also used for detection of nominal events (Grover et al., 2010).

3.2 Statistical Methods

Bethard and Martin proposed a statistical system for event extraction using a multi-class classification method (Bethard and Martin, 2006). The system automatically annotated each token by "Inside", "Outside", or "Begin" tags. It also determined the semantic class of each event. This system used various morphological, syntactic, and semantic features. Their reported results have shown an acceptable rate in event detection.

Another statistical method for event extraction applied a classification algorithm based on Support Vector Machines (SVM) in both sentence and word levels (March and Baldwin, 2008). First, it filters out the sentences without events. Then, in the remaining sentences, it searches for events. This method does not determine the features of events (i.e., the third stage of the event extraction task).

TipSem is one of the most successful TimeML event extraction methods, which has used Conditional Random Fields (CRF) classification technique (Llorens et al., 2010). It utilizes morphological, syntactic, and semantic features in addition to semantic roles for event extraction.

3.3 Combined Methods

Evita is the first system that has been designed for extraction of TimeML events (Sauri et al., 2005). The system has benefited from both statistical and rule-based techniques. In pre-processing stage, it extracts part of speech tags, phrase chunks, and lemmas of the sentences' tokens using some existing tools. For event detection of nouns, Evita uses WordNet, and in ambiguous cases, i.e. nouns that may or may not be an event in sentences, a Bayesian classifier is used for disambiguation. This classifier has been trained on the SemCore data. For adjectives, the cases that have been annotated in TimeBank are considered as events. The heads of the predicative complements are also regarded as events. Evita uses TimeBank information for finding the class of events. It simply chooses the majority class for the specific event in the corpus.

TRIOS is another combined method, which first utilizes the TRIPS semantic parser (UzZaman and Allen, 2010). Based on the output of the parser and applying some rules, it detects the events. Then, for improving the accuracy, a Markov Logic Network (MLN) classifier is used. This classifier is also used for extraction of event features.

4 Automatic Event Extraction from Persian Text

As mentioned before, most of the event recognition and classification methods include three phases: pre-processing, event mention detection and event attributes extraction. These phases also exist in our system for Persian texts. We called this system Persian Event Tagger (PET). In pre-processing stage of PET, we convert input documents into the CoNLL-2009 Shared Task (CoNLL09) format by using Dadegan tools[1]. It means that the Dadegan tools extract part of speech (POS) tags, dependency labels, and other necessary information from input documents. The dependency labels are according to Persian dependency Treebank, which is the first released dependency Treebank for Persian. This corpus currently contains 30,000 sentences that were manually annotated[2].

We apply our feature extractor subsystem to the converted document to obtain useful and meaningful features for recognizing and classifying of events. The outputs are data set files, which contain all features for each token in a separate line. We employ these data set files to perform event mention detection and event attribute detection stages. A rule-based and a statistical learning-based approach are implemented for both of these stages. In event attribute detection, we just find *class* because other attributes like *tense*, *mood,* and *aspect* were previously found with Dadegan tools. For other event attributes, we just choose the default values; for instance, we set *polarity* to “Positive”. In section 4.1, the features are discussed. The rule-based and learning-based subsystems are explained in sections 4.2 and 4.3, respectively.

4.1 Features

The features that have been used in PET can be clustered into three types: lexical, syntactic, and semantic. In the following, each type of these features is explained in more details:

4.1.1 Lexical Features

Lexical features that we have used in the system are token’s text, coarse-grained POS, fine-grained POS, word’s stem, word’s postfix (i.e., last three letters of word), and a Boolean feature *isModAux, which* is “true” for modal or auxiliary verbs. For recognizing these features, we rely on the raw text, coarse-grained, and fine-grained POS from the pre-processed input file.

4.1.2 Syntactic Features

In some event tagging situations, PET requires deeper knowledge about sentences. Dependency parsing is a new and effective approach for obtaining this knowledge. We can extract various syntactic features from dependency labels of the input sentences.

The extracted syntactic features, for each candidate, include: dependency label, lemma of the head token, text of the head token, POS of the head token, dependency label of the head token, governing verb text, governing verb lemma, governing noun text, and governing noun lemma.

[1] Freely abailable for download at http://dadegan.ir/en/tools

[2] Freely available for download at http://dadegan.ir/en/persiandependencytreebank

Other extracted syntactic features are *isPartOfCompoundVerb* and *isPartOfCompoundNoun*, which indicate that candidate word is part of a compound verb or noun, respectively. These features can be assigned by searching the dependency parse tree of the input sentence.

4.1.3 Semantic Features

A number of semantic features are required for recognizing nominal events. They are also needed for classification of both nominal and verbal events. These features can be obtained from resources like WordNet by searching through word senses and checking their hypernyms. In Persian, we have two resources that are similar to WordNet. One of them is FarsNet, which has been developed semi-automatically with 9,266 synsets and 13,155 words (Shamsfard et al., 2010). Another resource, which we call it PersianWN, has been developed automatically. This resource, has covered 29,716 Persian phrases with reported precision 82.6% (Montazery and Faili, 2010).

To recognize nominal events, a number of Boolean features are extracted from both FarsNet and PersianWN. We consider several synsets including: "event", "human_action", "human_activity", "act", "phenomenon", and "action" to be eventive. Thus, when hypernyms of a sense fall into these synsets, we will consider this sense to be eventive. In this way, we can extract following features for each noun according to its senses: *isAllSensesEvent*, *isAllSensesState*, *isMainSenseEvent* (i.e., "true" when more than 1/3 of senses are eventive), *isOneSenseEvent*, *isOneSenseState*.

To classify events, some other features are extracted. For each event *class*, a list of phrases is created. These lists are initially filled by sample phrases that have been mentioned in the ISO-TimeML guideline. Then, we augment these lists by adding their synonyms taken from FarsNet. After creating and enriching these lists, we set a number of features (i.e., *isReporting*, *isAspectual*, *isPerception*, *isI_Action*, *isI_State*, *isState*, and *IsOccurance*) for each input phrase. We search a phrase in each list and if it is successfully found, the corresponding feature, which is related to that list, will be assigned to "true". For instance, when verb lemma exists in the aspectual phrase list, the *isAspectual* feature will be set to "true".

It must be noted that there is a difficulty in searching compound nouns/verbs in a list, lexicon, or dictionary. In these cases, the system has to combine all parts of the noun/verb before starting to look up. These parts can be obtained and then combined using related labels. For example, in sentence (a) we must look up the whole text "bazgoo kard" (restate) in dictionaries or lexicons.

a. Ô (He) moshkelât (problems) râ (-) **bâzgô** (restate) **kard** (did).
   Translation: He **restated** the problems.

4.2 Rule-based Method

For event mention detection, we apply PersTimeML guidelines to the input text by utilizing the previously mentioned features. Besides, a number of lists including special phrases like aspectual and causative signal words are used for event tagging. Therefore, for each candidate word, we apply a number of if-then-else rules, which are based on one or more features. These rules are explained in the following subsections.

4.2.1 Recognizing Verbal Events

According to PersTimeML, we annotate all verbs as events except for modals, auxiliary verbs, and verb "to be". We can easily find these verbs by checking the Boolean feature *isModAux*.

Furthermore, for each verb, we search the sentence for its probable non-verbal elements. If any such elements are found, they will be combined with the verb. These searches can be performed using related syntactic features, i.e., *isPartOfCompoundVerb* and position of the head token.

4.2.2 Recognizing Nominal Events

Recognizing nominal events is more challenging than other forms of events and requires a deeper analysis of input sentences. Some PersTimeML rules can be applied to the nouns by utilizing the mentioned syntactic features. In current version of PET, we apply causative and aspectual rules. This means that a noun when appears in a specific position in an aspectual or a causative structure is tagged as an event. These aspectual and causative structures are found by searching the context of the noun for occurrence of an aspectual or a causative signal word.

For instance, in sentence (a) below, there is a causative structure and therefore according to the PersTimeML, we must annotate all of "bârân-e" (rain), "seyl" (flooding), and "môjeb-e" (cause) as separate events. With dependency labels, we can determine the subject of a causative sentence. Then, if the subject is a phenomenal noun, we will annotate it as an event. Furthermore, "seyl" (flooding) can be recognized by annotating the head of the noun phrase that immediately appear after the signal word "môjeb-e" (cause).

Sentence (b), has an aspectual structure, which is triggered when encountering the verb "âghâz shod" (has started). We must annotate the subject of this structure, i.e., "marg" (death), as an event. In other structures, when an aspectual or a causative signal word is triggered, we can apply similar rules for finding the events.

a. [Bârân-e] (Rain) zyâd (heavy) [**môjeb-e]** (cause) [seyl] (flooding) shod (become).
   Translation: The heavy rain **caused** flooding.
b. [Marg-e] (Death) khôkhây-e (pigs) **âghâz shod** (has started).
   Translation: The death of pigs **has started**.

For other nouns, we first disregard each noun that have *isPartOfCompoundVerb* feature with value of "true", because in fact, it is part of a compound verb in the sentence. Besides, if a noun is part of another noun, i.e., has value of "true" for *isPartOfCompoundNoun* feature, it will be annotated in conjunction with its governing noun as an event tag.

Finally, for remaining nouns, our rule-based module can only use the semantic features, because currently there is not any acceptable Word Sense Disambiguation (WSD) system or even a semantic tagged corpus for Persian. In the module, we only use the feature *isAllSensesEvent*. When value of this feature is "true" for a noun, the noun will be annotated as an event. Another solution for improving the result of event recognition for nouns is creating a list of all *predicative nouns*. This is discussed in the evaluation section in more detail.

4.2.3 Recognizing Adjective Events

To recognize adjective events, the system first checks to see if the adjective is a predicative complement in the sentence. It can be performed by using the both dependency label and governing verb of the adjective. The adjective will be regarded as an event, if its dependency label is "MOS" (i.e., the adjective is predicate in the sentence) and its governing verb exists in a list of special predicate verbs such as "shodan" (become) and "kardan" (do).

A second rule is employed for recognizing objective deverbal adjectives by watching the postfix feature. When the value of this feature equals to “shode”, we will annotate the adjective as an event.

4.2.4 Identifying the *Class* of Events

To determine *class* of events, we rely only on event *class* features (e.g., *isAspectual, isPerception, isI_State* and so on). For instance, if the value of *isReporting* feature is equal to “true”, we will assign the “Reporting” value to *class* attribute. For those events, not having any of these features with value of “true”, we consider the default values (i.e., ”occurrence” for nouns and verbs and “state” for adjectives)

4.3 Learning-based Method

By applying the TempEval 2010 format to the developed corpus and using the previously mentioned features, a learning model can be trained. The whole relevant processes including data loading, data pre-processing, creating and applying a model, and the evaluation can be performed using an open source software called RapidMiner[3].

By RapidMiner GUI, we can easily design our learning process and obtain various desired evaluations. Therefore, our learning-based module identifies event mentions and determines the *class* of them using RapidMiner processes. We utilize a feature selection process based on a naïve Bayesian classifier in this module. The process detects optimized subsets of features for each classification task individually. The classification cases are expressed in the evaluation section.

5 Evaluation

For evaluating both PersTimeML and PET, we built a suitable corpus with annotated events. The corpus is in fact the first and a preliminary Persian version of well-known English corpus, TimeBank. We called this corpus PTB. An iterative incremental process has been used to create PersTimeML, PTB, and PET.

The input documents have been taken from Peykareh, also known as Bijankhan corpus. Peykareh is currently the most popular Persian corpus, which contains more than seven million tokens (Bijankhan et al., 2010). We selected a number of documents from diverse topics including political, economic, sport news, stories, etc. Then, we pre-processed these selected documents with Dadegan tools for tokenizing and converting to the conll09 format. We had extracted sentence texts from Conll09 files because of the necessary tokenization pre-processing that input sentences had required. Then, we applied the rule-based PET to primarily annotate events, followed by a manual correction of the system output. The manual correction was performed using the MAE (Multi-purpose Annotation Environment) annotation tool[4]. The MAE output files then converted into the TempEval 2010 data format for simplifying the evaluations.

We annotated 43 documents from Peykareh. This contained 26,949 tokens and 4,237 events. Statistics about frequency of events for each POS tag and also, frequency of events in each event *class* are shown in TABLE 1.

[3] Freely available at: http://rapid-i.com/

[4] Freely available at: http://pages.cs.brandeis.edu/~astubbs/mae.html

POS of Event	Frequency		Event Class	Frequency
Noun	1,960		*Occurrence*	2,488
Verb	1,875		*State*	623
Adjective	312		*Reporting*	433
Preposition	51		*I_Action*	276
None	39		*I_State*	218
			Aspectual	133
			Perception	66

TABLE 1 – Some statistics about PTB

To evaluate the PET, the corpus was split into a development set, a training set, and an evaluation set. The evaluation was performed token by token even for multi token events. It means that the scorer programs took each token individually and then calculated the value of each performance property like recall, precision and f-measure. The training and evaluation sets were both used in evaluating the rule-based method. We used a five-fold cross-validation with a stratified sampling over the training and evaluation sets for evaluating the learning-based method.

The results of the PET in event mention detection for both rule-based and learning-based methods are shown in TABLE 2. In order to calculate the results for each event category, we only considered tokens in that category in evaluations. In the learning-based method, this led to an individual model for each individual event category. It should also be noted that the learning-based method assigns each individual token to either an *event* class or a *non-event* one.

	Rule-based			Learning-based		
Category	Precision	Recall	F	Precision	Recall	F
All	78.9	72.5	75.6	79.2	87.5	83.1
Verb	96.5	99.3	97.9	97.1	99.5	98.3
Noun	66.3	64.4	65.3	82.1	81.8	82.0
Adjective	88.5	55.8	68.4	78.3	76.4	77.3

TABLE 2 – Evaluation results for PET for event recognition

In the evaluation for all categories, the both modules showed high precisions, while for recall the learning-based method is 15% better than the rule-based one. This weaker recall for the rule-based method is due to the lower recall for nouns and adjectives. Therefore, it showed that although the existing rules have gained a satisfactory precision, we should add extra rules or modify the lexicons for finding more event tokens.

For nouns, the most effective features that have been used in the PET are semantic features. These features had been extracted from special dictionaries and lexicons. An experiment on six random documents of the PTB showed that 60.7% of nominal events were *predicative nouns*. As we said before, *predicative nouns* in Persian when function as gerunds, are events. Therefore, it must be noted that the quality of these resources for finding eventive nouns or having a list of all *predicative nouns* immensely affects the performance of the event tagger systems.

Some coverage tests of the lexical databases were performed for the nouns of the PTB corpus. From 11,942 nouns of PTB, 7,625 nouns were found in FarsNet (63.8%) and 7,934 nouns were found in PersianWN (66.4%). Although the PersianWN has had greater coverage, it suffers from lower accuracy because as we mentioned in 4.1, it was developed in a fully automatic process with 82.6% accuracy. On the other hand, the FarsNet was manually validated and therefore, has had high accuracy. When we experimented with PersianWN for recognizing noun events, we gained a low recall (35.1%) but a high precision (80.0%) that is a sign for its incorrect senses or hypernyms relationship between synsets.

	Rule-based			Learning-based
Category	Correct	Incorrect	Accuracy (%)	Accuracy (%)
All	1,946	819	70.3	74.9
Verb	475	353	57.4	73.5
Noun	1,301	398	76.5	82.3
Adjective	147	79	65.0	78.8

TABLE 3 – Evaluation results for event class detection

The evaluation results represented in TABLE 3 show the accuracy for detection the *class* of events. In the learning-based method, one model is created for each POS category and one classification per event is performed to assign the proper *class* to each event. The high F-measure of the rule-based approach (70.3%) is just achieved by using event *class* lists. One reason is that the majority of events have "occurrence" value in their *class* attribute. According to TABLE 2, 2,488 out of 4,237 events (i.e., 58.7% of events) have *class* attribute with value of "occurrence". Thus, a baseline system that assign "occurrence" to all events, will achieve 58.7% accuracy. By this baseline, the accuracy of 74.9% by learning-based module is both acceptable and remarkable.

	Learning-based event class detection		
Classes	Precision	Recall	F
Occurrence	73.8	91.5	81.7
State	72.2	30.7	43.1
I_State	58.2	61.5	59.8
I_Action	67.8	35.2	46.3
Reporting	92.9	86.7	89.7
Perception	75.0	46.1	57.1
Aspectual	86.2	60.1	70.9

TABLE 4 – Evaluation results for each event class

Evaluation results for the various event classes in the learning-based method are shown in TABLE 4. The worst results are in "state", "I_Action" and "perception" classes. "Perception" is scarcely occurred in the corpus and therefore, had low result. "State" and "I_Action" classes occurred in more distinct phrases in comparison with "reporting" and "aspectual" classes and thus, had lower accuracies.

Conclusion and Perspectives

In this paper, we have addressed the problem of event recognition and classification, which has been a challenging task since early days of statistical natural language processing. More specifically, we focused on ISO-TimeML event annotation for Persian. Since there have not been any suitably tagged corpus in Persian, we have developed an annotated corpus. In the annotation process, we adapted the ISO-TimeML standard for Persian. We have also proposed two different methods for automatic identifying event mentions and their corresponding attributes as part of a

Persian event tagger system. The first was utilized a rule-based approach with different rules. The second was a statistical learning-based approach, which used a classification technique for tackling the problem. Our experimental results show an acceptable accuracy, considering our system as being the first effort for event recognition in Persian.

Currently, we are working on finding ways for further improvement of our system, PET, and at the same time annotating more documents for increasing the size of the corpus. It seems that using cross-lingual techniques can further improve the accuracy of existing methods for Persian (and languages that currently lack rich resources for NLP applications). Other possible future work includes employing richer learning models such as Conditional Random Fields (CRFs) for event recognition and classification. It is also the case that PTB should be retagged by other annotators to meet the inter-annotator agreement criterion. PTB can also be further improved by annotating other ISO-TimeML tags such as time expressions and temporal relations.

Acknowledgments

This paper has been partially funded by the Iran Telecommunication Research Center (ITRC).

References

Bethard, S. and Martin, J. (2006). Identification of event mentions and their semantic class. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing-EMNLP*, pages 146–154. ACL.

Bijankhan, M., Sheykhzadegan, J., Bahrani, M. and Ghayoomi, M. (2010). Lessons from building a Persian written corpus: Peykare. *Language Resources and Evaluation*, *45*(2): 143–164.

Bittar, A. (2009). Annotation of Events and Temporal Expressions in French Texts. In *The 19th Meeting of Computational Linguistics in the Netherlands,* pages 25–38.

Bittar, A., Amsili, P., Denis, P. and Danlos, L. (2011). French TimeBank: an ISO-TimeML annotated reference corpus. In *Proceedings of the 49th Annual Meeting of ACL,* pages 130–134.

Caselli, T., Lenzi, V. B., Sprugnoli, R., Pianta, E. and Prodanof, I. (2011). Annotating Events, Temporal Expressions and Relations in Italian : the It-TimeML Experience for the Ita-TimeBank. In *Proceedings of the Fifth Law Workshop (LAW V)*, pages 143–151. ACL.

Fellbaum, C. (1998). *WordNet: An Electronic Lexical Database*. MIT Press.

Grover, C., Tobin, R., Alex, B. and Byrne, K. (2010). Edinburgh-LTG: TempEval-2 system description. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, pages 333–336. ACL.

Im, S., You, H., Jang, H. and Nam, S. (2009). KTimeML: specification of temporal and event expressions in Korean text. In *7th Workshop on Asian Language*, pages 115–122.

Karimi-Doostan, G. (2011). Lexical categories in Persian. *Lingua*, *121*(2): 207–220.

Lesani, H. (2003). moghâyesey-e sefât-e feelêy-e fâelê va mafôlê dar zabân-e rôsê va fârsê. *pajoohesh zabanhaye khareji*, *15*: 75–84.

Llorens, H., Saquete, E. and Navarro-Colorado, B. (2010). TimeML Events Recognition and Classification: Learning CRF Models with Semantic Roles. In *Proceedings of the 23rd*

International Conference on Computational Linguistics-Coling, pages 729–733. Beijing, China.

March, O. and Baldwin, T. (2008). Automatic event reference identification. In *Australasian Language Technology Association Workshop 2008*, pages 79–87, Australia.

Montazery, M. and Faili, H. (2010). Automatic Persian WordNet Construction. In *Proceedings of the 23rd International Conference on Computational Linguistics: Posters*, pages 846–850. ACL.

Pustejovsky, J., Ingria, R., Setzer, A. and Katz, G. (2003). TimeML : Robust Specification of Event and Temporal Expressions in Text. *New Directions in Question Answering*.

Pustejovsky, J., Lee, K., Bunt, H. and Romary, L. (2010). ISO-TimeML : An International Standard for Semantic Annotation. In *Proceedings of the Seventh conference on International Language Resources and Evaluation (LREC'10)*.

Rasooli, M.S., Faili, H., Minaei-Bidgoli, B. (2011). Unsupervised Identification of Persian Compound Verbs. In *Advances in Artificial Intelligence*, pages 394–406.

Robaldo, L., Caselli, T. and Russo, I. (2011). From Italian Text to TimeML Document via Dependency Parsing. In *Proceeding of the 12th International Conference on Intelligent Text Processing and Computational Linguistics,* pages 177–187.

Sauri, R., Knippen, R., Verhagen, M. and Pustejovsky, J. (2005). Evita: A robust event recognizer for QA systems. In *LT/EMNLP*. ACL.

Shamsfard, M. (2011). Challenges and open problems in persian text processing. In *LRL*, pages 65–69.

Shamsfard, M., Hesabi, A., Fadaei, H., Mansoory, N., Famian, A., Bagherbeigi, S., Fekri, E., Monshizadeh, M. and Assi, SM. (2010). Semi Automatic Development of FarsNet; The Persian WordNet. In *Proceedings of 5th Global WordNet Conference, Mumbai, India*.

UzZaman, N. and Allen, J. F. (2010). Event and Temporal Expression extraction from raw text: first step towards a temporally aware system. *International Journal of Semantic Computing, 4(4)*.

Verhagen, M., Sauri, R., Caselli, T., and Pustejovsky, J. (2010). SemEval-2010 Task 13 : TempEval-2. In *Proceedings of the 5th International Workshop on Semantic Evaluation*, pages 130–134. ACL.