Temporal Relation Classification in Persian and English Contexts

Mahbaneh Eshaghzadeh Torbati
Department of Computer Engineering,
Sharif University of Technology,
Tehran, Iran
Mahbaneh.eshaghzadeh@gmail.com

Gholamreza Ghassem-sani
Department of Computer Engineering,
Sharif University of Technology,
Tehran, Iran
sani@sharif.edu

Seyed Abolghasem Mirroshandel
Faculty of Engineering,
University of Guilan,
Rasht, Iran
mirroshandel@guilan.ac.ir

Yadollah Yaghoobzadeh
Department of Computer Engineering,
Sharif University of Technology, Tehran, Iran
yaghoobzadeh@ce.sharif.edu

Negin Karimi Hosseini
School of Computer Science,
Faculty of Information Science & Technology, Universiti Kebangsaan Malaysia
Negin62_k@yahoo.com

Abstract

This paper introduces the first pattern-based Persian Temporal Relation Classifier (PTRC) that finds the type of temporal relations between pairs of events in the Persian texts. The proposed system uses support vector machines (SVMs) equipped by combinations of simple, convolution tree, and string subsequence kernels (SSK). In order to evaluate the algorithm, we have developed a Persian TimeBank (PTB) corpus. PTRC not only increases the performance of the classification by applying new features and SSK, but also alleviates the probable adverse effects of the Free Word Orderness (FWO) of Persian on temporal relation classification. We have also applied our proposed algorithm to two standard corpora on English (i.e., TimeBank and TempEval-2) to measure the efficiency of the new features and SSK. The experiments show the accuracies of 65.6%, 59.53%, 50.2%, and 62.17% on an augmented version of PTB, TimeBank, tasks E and F of TempEval-2, respectively. Consequently, we have achieved the third best result on TimeBank, and the second best result on the task F of TempEval-2.

1 Introduction

The goal in temporal relation classification is to find the temporal ordering between temporal entities of the input text. As a result, these relations can be used in applications such as question answering and summarization systems.

In general, temporal relation classification is the task of determining when an event/time expression has taken place with respect to some other event/time expressions. In this study, we only try to find these relations between events, not between events and time expressions.

In temporal corpora that have been created so far, different temporal relation classes have been considered. In TimeBank (Pustejovsky et al., 2003), the first corpus that has changed the research trend towards machine learning methods, there are six different temporal relations, namely SIMULTANEOUS, INCLUDES, BEFORE, IBEFORE, BEGINS, and ENDS. On the other hand, in TempEval-1 (Verhagen et al., 2007) and TempEval-2 (Verhagen et al., 2010), the temporal relations are BEFORE, OVERLAP, AFTER, BEFORE-OR-OVERLAP, OVERLAP-OR-AFTER, and VAGUE.

Despite the multitude of speakers of Persian (Bateni, 1995), there has not existed any corpus tagged with temporal relations in Persian yet. Thus, as the first step, events and its attributes were tagged in the PTB corpus (Yaghoobzadeh et al., 2012). We have continued their work by annotating temporal relations between tagged events and Signals, manually and based on an adapted version of the ISO-TimeML guideline (Pustejovsky et al., 2010).

In the second step, our goal has been designing a system that classifies temporal relations in Persian texts. Considering that Free Word Orderness (FWO) could have a negative impact on classification, we have aimed to design our Persian Temporal Relation Classifier (PTRC) in a way that prevents side-effects as much as possible. Thus, a simple kernel was applied to a group of lexical and semantic features that were inherently resistant against FWO. Then, according to the efficiency of dependency relations in temporal classification, as well as their robustness and stability in dealing with FWO, for each sentence two dependency-based

Proceedings of Recent Advances in Natural Language Processing, pages 261–269,
Hissar, Bulgaria, 7-13 September 2013.

tree structures were built. In addition, two different convolution tree kernels with various weighting methods were applied to them subsequently. Finally, a novel FWO-resistant kernel named string subsequent kernel (SSK) was applied to aforementioned structures.

In the third step, in order to further evaluate the efficiency of the new features and SSK in temporal classification, PTRC was applied to TimeBank and tasks E and F of TempEval-2.

The remainder of this paper is as follows: Section 2 is about temporal classification methods. Section 3 explains some challenges in Persian, and accordingly Section 4 represents the solution for tackling such difficulties. Section 5 includes the explanation of proposed system. Finally, in Sections 6 and 7, the results of the experiments and our conclusion are reported.

2 Related work

One of the most widely used temporal logics, which is the foundation of the most existing achievements related to temporal relation classification, was proposed by Allen (1984). Various rule-based studies were conducted based on 13 temporal relations defined between intervals in this logic. By creation of different temporal corpora, the research trend turned into machine learning methods, which so far achieved the best results in this regard.

Among the outstanding methods performed on TimeBank, we can report four researches by (Lapata and Lascarides, 2006), (Chambers et al., 2007), and (Mirroshandel et al., 2011a, b). The first method extracts novel syntactic features in an ensemble classification method (Lapata and Lascarides, 2006). They have simplified the problem by restricting the diversity of temporal classes. In the second method, a two-stage SVM-based classification technique was proposed, in which event and attribute extraction in addition to temporal relation classification were executed (Chambers et al., 2007). Mirroshandel et al. (2011a, b) showed that the parse tree structures can be used as informative features in the temporal classification process. By applying convolution tree kernels to constituent and dependency parse trees, they developed two separated systems. Moreover, Mirroshandel and Ghassem-Sani (2010) have applied a bootstrapping method to their system and outperformed all related works.

In TempEval workshops, systems with more innovative classifiers were presented. For instance, a classifier named Conditional Random Field (CRF) algorithm was applied in both (Kolya et al., 2010) and (Llorens et al., 2010). The system presented in (Yoshikawa et al., 2009) can be considered as the first advent of Markov Logic Network (MLN) in temporal classification participated in the TempEval-1 competition. Ha et al. (2010) also achieved the best accuracy for Task F in TempEval-2 by use of MLN.

3 Persian Language Challenges

3.1 Compound Verbs and Free Word Orderness

Persian compound verbs are a kind of multiword light verb construction that still has remained as one of Persian challenges in NLP tasks (Rasooli et al., 2011). The complexity is due to the variety in count and type of nonverbal elements, in addition to syntactic flexibility such as unlimited word distance between the light verb and its components. *Delxor kardan* (to annoy), *talâq dâdan* (to divorce), and *pas dâdan* (to return) are some examples of compound verbs in Persian.

Although formal sentences in Persian have the SOV structure, it is also a free word order language, in which the sentential constituents can be arbitrarily moved around in the sentence.

3.2 Tackling Persian Challenges

The task of temporal relation classification in Persian is more complicated than in other languages such as English. High Frequency of compound verbs and their by-product noun and adjective phrases in Persian, makes the feature extraction more complex. Fortunately, by the multiword annotation method that has been performed on PTB, feature extraction and dependency tree pruning (to be discussed in Section 5) have become straightforward. Furthermore, the syntactic feature efficiency can be devalued, due to the existence of FWO in sentence structures. Hence, in order to alleviate the adverse impact of FWO, a combination of three FWO-resistant kernels has been employed in the SVM classifier.

The first kernel, named K_{simple}, is a linear kernel that neutralizes the FWO side-effects by exploiting a collection of lexical and semantic features. These features are inherently stable against FWO. The second group of kernels consists of two weighted convolution tree kernels applied to two tree structures constructed and valued based on dependency relations and POS tags of sentence elements. These kernels take advantage of both dependency structures and a bi-gram estimation of tree-constructing features. By utilization of dependency relations and a tree sorting method, the FWO side-effects can be eliminated from these kernels. The third kernel is known as a string subsequence kernel (SSK) that evaluates the identical sub-strings of the tree paths joining the events involved in temporal relations. This kernel is being used in temporal classification for the first time and since it is operated on a dependency-based path, it is independent of sentence structure and FWO problem. In the following section, each kernel group will be discussed in more detail.

4 Proposed Features and Kernels

4.1 K_{Simple} kernel and relevant features

In this section the FWO-resistant feature set for both Persian and English systems as well as the K_{Simple} kernel are discussed.

Features: We divided features into three categories of Event-based, Temporal-Relation (TR)-based, and governing-based features. All new features are marked by * in this section.

Event-based features: These features are determined for each event involved in a temporal relation. *Tense*, *Mood*, *Aspect*, *Modality*, *Polarity*, and *Class* are human annotated features extracted from related Persian and English corpora. The others, consist of Lemma, Voice* and Synset, are extracted automatically.

Voice*: It is a binary feature, based on verb transitivity status, assigned to verbal events.

Synset: WordNet and FarsNet (Shamsfard et al., 2010) synsets are categorized based on their part of speech tags. Hence, the synset feature is partly evaluated incorrectly due to the probable dissimilarity of POS tags of events, although they are semantically related. Temporal pair of (*Announced*, *Denote*), which involves adjectival and verbal event respectively, is a constructive example in this respect. As a solution, we have developed this feature and estimate it based on all event derivations that exist in WordNet. Comparing with Wordnet, there still exist some deficiencies in Farsnet. Therefore in Persian synset extraction process, words have been initially mapped to their English peers in Wordnet, and then the required information has been extracted from Wordnet database.

TR-based: These features are defined for each temporal relation listed as follows:

Text order: This feature refers to the event appearance orders in the context.

Inter/Intra relation: This feature defines whether the events are within the same sentence or not.

Be numerical*: It defines whether the nominal events have numerical essence or not.

Be aspectual*: It defines whether the events have a triggering or terminating essence.

Context topic*: This feature categorizes each context in one of the narrative, financial, biography, or accidental fields.

Classified distance*: It classifies the eventual distance in the adjacent, near, or far classes.

Signal lemma*: It contains the lemma of involved signal in a temporal relation.

Signal class*: It classifies the signals into temporal classes based on (Mortazavinia, 2010).

Governing-based features: Clearly, features such as *Tense, Aspect, Voice*, and *Mood* are verb-specific and also crucial to temporal classification. Therefore, based on "NONE" values allocated to their mentioned features, non-verbal events may be devalued in the classification process. In order to alleviate this probable impact, these feature values owned by governing verb of non-verbal events have been selected as substitute for the former ones. The governing verbs have been distinguished based on dependency relations.

K_{Simple} kernel: By utilization of this kernel, we try to calculate the temporal relation similarity in features in section 5.1. By defining K_S, *TR, E, f, Tf, n*, and *C* as K_{Simple} kernel, temporal relation, event, event-based feature, TR-based feature, the feature count, and function of counting the number of common features of events involved in a temporal relation respectively, the K_{Simple} kernel can be introduced as follows:

$$K_S(TR_1, TR_2) = \sum_{i=1,2} K_E(TR_1.E_i, TR_2.E_i) + C(TR_1.Tf, TR_2.Tf) \quad (1)$$

$$K_E(E_1, E_2) = \sum_{i=1}^{n} C(E_1.f_i, E_2.f_i) \quad (2)$$

It should be noted that equation (1) is the manipulated version of the kernel introduced in (Mirroshandel et al., 2011b) utilized for the involving TR-based features in kernel evaluation.

4.2 Tree kernels and syntactic feature

Dependency relation transformation: The dependency relation contributes to utilize a mostly FWO-resistant version of sentence structure, in temporal classification. In order to construct dependency trees, two structures named $Trans_1$ and $Trans_2$ proposed in (Mirroshandel and Ghassem-Sani, 2011a) have been implemented. Afterwards, a minor manipulation for applying tree kernels to inter-sentence relations has been exerted on the parse trees. This process includes combining tree structures of each sentence by selecting them as children to arbitrary augmented node. The tree constructions are shown in figure 1 and 2.

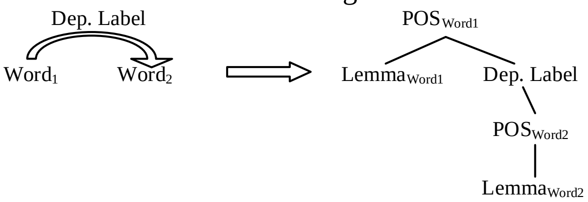

Figure 1: *Trans₁* transformation.
(Mirroshandel and Ghassem-Sani, 2011a)

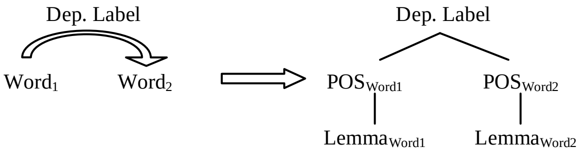

Figure 2: *Trans₂* transformation.
(Mirroshandel and Ghassem-Sani, 2011a)

Trans₂ transformation is partially similar to constituent parse tree. As a result, it can be substituted for the original one in the proposed system. However, this structure would partly be FWO-affected. In other words, the priority of node appearance in a tree is dependent on their orders in the sentence. In *Trans₁*, just children priority is manipulated by FWO, therefore a sorting method, based on ordered list of whole tree node values, has solved the problem and finally made *Trans₁* completely FWO-resistant. In *Trans₂*, both dependency relation and sentence element order assign children of nodes, therefore this manipulation has been too complicated to be solved by a simple sorting method. Based on these explanations, *Trans₂* still remains FWO-affected and would be just efficient for English temporal classifier. As we will see in Section 6, this structure will be automatically omitted among best Persian classifiers.

Tree pruning and weighting methods: It has been shown that tree kernels operate more efficiently by being applied to pruned trees (Zhang et. al., 2006). Based on this observation, the path enclosed tree (PET) method has been exerted on the desired dependency trees. In this method, all the nodes of the path (the path from event nodes to their common parent) and the ones among this path would be designated as the desired portion of tree.

In the next stage, three various weighting methods, inspired by (Mirroshandel et al., 2011b), are applied to the pruned trees. The first method, named Argument Ancestor Path (AAP), just considers the nodes on the path enclosed by the event nodes, as well as their immediate descendants. The second one, named Argument Ancestor Path Distance (AAPD), allocates weights to all pruned tree nodes based on their distance from the nearest ancestor of one of the events in the path. The third method, known as Argument Distance Kernel (AD) is very similar to AAPD except that weights are evaluated based on the distance from the nearest event.

Convolution tree kernels: Sentence structure can be referenced as one of the invaluable knowledge sources in the NLP applications. Convolution tree kernels compute the similarity between two trees by counting the number of common sub-trees. In our method, among various tree kernels, both subset tree (SST) (Collins and Duffy, 2001) and partial tree (PT) kernels (Moschitti, 2006b) have been applied to pruned and weighted tree structures. SST and PT have been reported to result more efficiently on constituent and dependency parse trees respectively (Moschitti, 2006b). SST sub-trees are restricted by the rule that states all nodes of sub-tree must appear with either all or none of its children. In contrast, PT sub-trees have no limitation on their structures and can have any arbitrary construction.

4.3 Dependency path in SSK kernel

Dependency path: The dependency path is a sequence of nodes enclosed between *Trans₁* event nodes. Based on the *Trans₁* design, this path contains the dependency relations among the components of the dependents of the root of each sentence that contains temporal related events. Considering that FWO just changes the children orders of *Trans₁*, the path will be FWO-resistant. Consequently, no extra method is required for tackling the probable side-effects.

SSK Kernel: SSK was initially proposed for estimating a similarity measure between sequences (Lodhi et. al., 2002). This similarity measure is based on the number of weighted sub-string matches that occur among sequences. The length of a sub-string, *K*, can be initialized manually based on the problem definition. In this method, both kinds of continuous and discrete matches are acceptable. For instance, both pairs of (*car*, *card*) and (*car*, *custard*) have the matches with the sub-string length of three as continuous and discrete matches, respectively.

SSK adaptation process: Benefiting from discrete match recognition, SSK contributes to compare extracted paths according to various sub-strings of POS and/or dependency tags, which is not possible by the aid of tree kernels. In order to take advantage of this capability, at first, a simple adaptation process needs to be executed on SSK. In original SSK, an alphabet letter is assumed as a comparing unit that can be expanded to sub-string by increasing the K value. On the other hand, in this study the comparing unit has been changed to POS and/or dependency labels. Therefore, a simple mapping method that relates a node label to an individual ASCII character can be used for the SSK adaptation.

4.4 Kernel normalization and composition

Normalization: The process of normalization is achieved by performing the equation $\frac{K(TR_1,TR_2)}{\sqrt{K(TR_1,TR_1).K(TR_2,TR_2)}}$ on kernel value.

Composition: The proposed kernels have been combined in two types of linear (K_L) and polynomial (K_P) forms. Considering α as an adapted parameter, the definitions of these compositions are as follows:

$$K_L(TR_1,TR_2) = \alpha K_1(TR_1,TR_2) + (1-\alpha)K_2(TR_1,TR_2) \quad (3)$$

$$K_P(TR_1,TR_2) = \alpha K_1(TR_1,TR_2) + (1-\alpha)K_2^P(TR_1,TR_2) \quad (4)$$

$$K_2^P = (1+K_2)^2 \quad (5)$$

5 Evaluation

5.1 Characteristic of the Persian corpus

Since there has not been created any temporal corpus in Persian yet, signals (as temporal entities) and event-event temporal relations were tagged in PTB (augmented PTB). For the evaluation purpose, PTRC in addition to English-adapted version of this system were implemented and evaluated over various corpora such as augmented PTB, TimeBank and TempEval-2. The annotation process was performed according to the ISO-TimeML guideline. 401 signals and 1,613 temporal relations were extracted within 72 texts selected from PTB. The statistics of temporal relation classes are reported in Table 1.

5.2 Feature selection

In feature selection, we performed a two-stage analysis on the feature set by measuring the accuracies of both *single-feature-included* and *single-feature-excluded* models for each feature. In other words, two K_{Simple} kernels were trained on two feature sets. In the *single-feature-included* kernel, feature set just includes a target feature. On the other hand, in the *single-feature-excluded* kernel, the feature set comprises all the features except the target feature. The final judgment about feature efficiency was made based on two measures named IncEva and ExcEva. The IncEva measure is based on *single-feature-included* model and presents the accuracy in sole presence of the feature. The ExcEva is based on *single-feature-excluded* model and presents the accuracy decrement encountering the feature omission.

Relation Type	**Frequency**	**Frequency(%)**
BEFORE	807	50
IBEFORE	83	5.15
Begins	72	4.46
Ends	47	2.91
SIMULTENOUS	461	28.58
INCLUDES	143	8.87
TOTAL	1613	100

Table 1: Temporal relation statistics in PTB.

Features	**ExcEva** (%)	**G-ExcEva** (%)	**G-IncEva** (%)
Lemma	0.31	0.49	55.26
Class	0.49	0.80	50.22
POS	0.19	0.31	51.45
Tense	-0.18	0.43	50.28
Mood*	-0.12	0.43	49.91
Aspect	-0.18	0.12	49.29
Voice*	0	0.31	49.91
Synset	0.43	0.62	45.42
Signal class*	0.92	1.23	2.89
Signal lemma*	0.12	0.49	2.89
Be numerical*	0	0.06	49.60
Be Aspectual*	0.43	0.8	51.63
Text order	0.19	0.25	49.91
Inter/Intra Relation	0.12	0.12	49.91
Context Subject*	0	0.06	49.91
Classified Distance*	0	0.12	49.91
Tree	1.11	1.48	52.86
K_{Simple} Accuracy	-	-	61.6

Table 2: Feature selection evaluations on PTB.

Persian feature selection: Table 2 shows the feature selection results on the feature set

explained in Section 5, as well as $Trans_1$ that is a tree feature extracted from augmented PTB.
The table has been designed in a way that features were separated into two event-based and TR-based parts. Governing-based evaluations have been specified by the "G" prefix and governing features have been highlighted. In addition, the new features have been marked by *. The highlighted features contribute more efficiently than the simple event-based ones. Furthermore, as the number of signal-involved temporal relations is insignificant (about 199 relations), the unsatisfactory G-IncEva value is justifiable. In fact, the signal-based features have been designed in a way to improve the classification accuracy in cooperation with other features. High G-ExcEva of the signal class is an evidence of this improvement. All features in Table 2, except $Trans_1$, are exploited by the K_{Simple} kernel. The last row shows the accuracy obtained by K_{Simple} kernel on the standard test set.

Features	TE2-E	TE2-F	TimeBank
Lemma	✓	✓	✓
Class	✓	✓	✓
POS	✓	✓	✓
Tense	G[1]	✓	✓
Aspect	G	✓	✓
Polarity	✓	✓	-
Modality	✓	✓	-
Synset	✓	-	✓
Signal class	-	-	✓
Signal lemma	-	-	✓
Be numerical	-	-	✓
Be Aspectual	✓	✓	-
Text order	✓	✓	-
Inter/Intra Relation	✓	✓	✓
Context Subject	-	-	✓
Classified Distance	✓	-	✓
K_{simple} Accuracy	49%	58.2%	57.98%

Table 3. Selected features for TimeBank and TempEval-2 task E and F.

English feature selection: Table 3 contains the designated features through the feature selection process on TimeBank (TB), the task E of TempEval-2 (TE2-E) and the task F of TempEval-2 (TE2-F). Signals are not annotated in the TempEval-2 database. As a result, the Signal-based features are ignored in the TempEval tasks. Similar to Table 2, the last row includes the K_{Simple}-trained SVM results based on the marked features in the table.
Table 4 contains the ExcEva evaluations of the novel features extracted from the English corpora. Despite the negative ExcEva value of the *Classified Distance* feature, its acceptable IncEva value, 50.2%, can justify the selection of this feature. It can be inferred from this table that the new features are also beneficial in English temporal classification.

Features	TE2-E (%)	TE2-F (%)	TB (%)
Signal class	-	-	0.29
Signal lemma	-	-	0.15
Be numerical	-	-	0.50
Be Aspectual	0.39	0.33	-
Context Topic	-	-	0.32
Classified Distance	0.39	-	-0.32

Table 4. Feature selection measures on TimeBank and TempEval-2 task E and F.

5.3 Experimental Results

We made use of LIBSVM Matlab source (Chang and Lin, 2001) for SVM classification, the MateParser (Bohnet, 2010) for dependency parsing, and JAWS (Spell, 2008) for retrieving information from WordNet. The implemented systems were applied to augmented PTB, TimeBank, tasks E and F of TempEval-2. We applied the five-fold cross validations method to PTB and TimeBank as well as simple classification to TempEval tasks. The evaluated accuracies are reported in tables 5, 6, 7, and 8. For more clarity, kernel compositions are formulated. In formulation method, names related to kernel compositions and either of tree and sequential kernels are subscripted by weighting and kernel methods, respectively. Moreover, "1" and "2" postfixes are added to the tree and sequential kernel names to indicate $Trans_1$ and $Trans_2$ structures.
Experiments on PTB: In order to measure the effectiveness of PTRC kernel, a variety of linear and polynomial kernel compositions and different weighting methods have been implemented and evaluated. Among these compositions, the most efficient ones, based on three weighting methods, are reported in a two-stage process in Table 5. In the first stage (SSK-excluded), various tree kernels and K_{Simple} compositions are examined. In the second stage

[1] Governing version of selected feature.

(SSK-included), the former compositions include the SSK to utilize its efficiency. Finally, Sorted-PK_{AAPD}, a sorted version of PK_{AAPD}, is selected as the PTRC kernel. As it is shown in Table 5, the last kernel outperforms the other compositions. The definitions of these compositions are as follows (PK_{AAPD} and Sorted-PK_{AAPD} exclude the *Trans₂* structure):

$$PK_{AAP} = \alpha(K_{simple}) + (1-\alpha)(1 + K1_{SST} + K2_{SST} + K1_{SSK})^2 \quad (6)$$

$$PK_{AD} = \alpha(K_{simple}) + (1-\alpha)(1 + K1_{SST} + K1_{SSK})^2 \quad (7)$$

$$PK_{AAPD} = \alpha(K_{simple}) + (1-\alpha)(1 + K1_{SST} + K1_{PT} + K1_{SSK})^2 \quad (8)$$

Methods	SSK-excluded (%)	SSK-included (%)
Baseline[2]	50	50
PK_{AAP}	64.43	65.17
PK_{AD}	63.63	65.17
PK_{AAPD}	**64.68**	65.30
Sorted-PK_{AAPD}	64.55	**65.60**

Table 5. The accuracy of PTRC on PTB.

Experiments on TimeBank: Various compositions have been tested on AAPD weighted trees. Comparing to both supervised and semi-supervised methods, our system has gained the third best accuracy that have been achieved so far. Although, by excluding the state-of-the-art method, Mir-semi-supervised (Mirroshandel and Ghassem-Sani, 2010), which profits from external sources, the proposed system has gained second best place inferior to Chambers (Chambers et al., 2007). However, our method has outperformed the equivalent method, Mir-supervised (Mirroshandel et al., 2011b), which benefits from both constituent and dependency parse trees. The TB-K_{AAPD} definition is as follow and the mentioned accuracies are reported in Table 6.

$$TB - K_{AAPD} = \alpha(Ksimple) + (1-\alpha)(K1_{PT} + K2_{SST} + K_{SSK} + 1)^2 \quad (9)$$

Methods	Accuracy (%)
Mir-semi-supervised	**66.18**
Chambers	60.45
TB-K_{AAPD}	59.53
Mir-supervised	58.76

Table 6. Accuracy of methods on TimeBank.

[2] The Baseline is the majority class for relations.

Experiments on TempEval tasks: Both tasks E and F are discussed in this section. As it is reported in Table 7, we have surpassed Mir-semi-supervised system (Mirroshandel, Ghassem-sani, 2012) with notable improvement, although the acquired accuracy is still far from the state-of-the-art system named TRIPS (UzZaman and Allen, 2010). However, the result in task E is more promising, as we have achieved the second best result after NCSU (Ha et al., 2010). Obviously our method has outperformed Mir-semi-supervised (Mirroshandel and Ghassem-sani, 2012) in this task, too. The definitions of tasks E and F are as follows:

Task E:

$$TE - K_{AAPD} = \alpha(K_{simple}) + (1-\alpha)(1 + K2_{SST} + K1_{SSK})^2 \quad (10)$$

Task F:

$$TE - K_{AAPD} = \alpha(K_{simple}) + (1-\alpha)(1 + K1_{SST} + K2_{PT} + K1_{SSK})^2 \quad (11)$$

Methods	Task E (%)	Task F (%)
TRIPS|NCSU-indi	**58**	**66**
TE-K_{AAPD}	50.20	62.17
Mir-semi-Supervised	45.62	50.41

Table 7. Accuracy of system on TempEval.

Tree and SSK efficiency: The accuracy increases caused by applying tree and string subsequence kernels to both English and Persian corpora are more observable in Table 7, 8.

Methods	SSK-excluded (%)	SSK-included (%)
Sorted-PK_{AAPD}	64.55	65.60
TB-K_{AAPD}	58.76	59.53
TE-K_{AAPD}	49.80	50.20
TE-K_{AAPD}	60.85	62.17

Table 8. Results of all implemented systems on Persian and English corpora.

6 Conclusion

In this paper, we have addressed the problem of temporal relation classification in Persian and English and SSK kernel applicable to both languages. As the first Persian temporal corpus, signals and event-event temporal relations have been annotated in PTB. Variety of compositions including tree structures, various kernels and several weighting methods were examined and consequently the best compositions were selected as kernels in SVM. The experiments show notable improvement in both languages.

References

James F. Allen. 1984. *Towards a general theory of action and time*. Articial intelligence, 23(2):123-154.

Mohammadreza Bateni. 1995. Tosif-e Sakhtari Zaban-e Farsi (Describing the Persian Structure). Amir-Kabir Press, Tehran, Iran (in Persian).

Bernd Bohent. 2010. *Top accuracy and Fast Dependency Parsing is not a Contradiction*. In Proceedings of Coling 2010, pp. 89-7.

Nathanael Chambers, Shan Wang and Dan Jurafsky. 2007. *Classifying temporal relations between events.* In Proceedings of the 45th Annual Meeting of the ACL on Interactive Poster and Demonstration Sessions, pp. 173-176. Association for Computational Linguistics.

Chih C. Chang and Chih J. Lin. 2001. *Libsvm: a library for support vector machines.* ACM Transactions on Intelligent Systems and Technology (TIST), 2(3).

Michael Collins and Nigel Duffy. 2001. *Convolution kernels for natural language.* In Proceedings of NIPS, Vol. 14, pp. 625-632.

Christiane Fellbaum. 1998. *WordNet: An Electronic Lexical Database*. MIT Press.

Eun Y. Ha, Alok Baikadi, Carlyle Licata and James C. Lester. 2010. *Ncsu: Modeling temporal relations with markov logic and lexical ontology.* In Proceedings of the 5th International Workshop on Semantic Evaluation, pp. 341-344. Association for Computational Linguistics.

Anup K. Kolya, Asif Ekbal and Sivaji Bandyopadhyay. 2010. *Ju-cse-temp: A first step towards evaluating events, time expressions and temporal relations.* In Proceedings of the 5th International Workshop on Semantic Evaluation, pp. 345-350. Association for Computational Linguistics.

Mirella Lapata and Allex Lascarides. 2006. *Learning sentence-internal temporal relations.* Journal of Articial Intelligence Research, 27(1): 85-117.

Hector Llorens, Estela Saquete and Borja Navarro. 2010. *Tipsem (english and spanish): Evaluating crfs and semantic roles in tempeval-2.* In Proceedings of the 5th International Workshop on Semantic Evaluation, pp. 284-291. Association for Computational Linguistics.

Huma Lodhi, Craig Saunders, John Shawe-Taylor, Nello Cristianini and Chris Watkins. 2010. *Text Classification using String Kernels*. In Proceedings of Neural Information Processing Systems, NIPS'00. , MIT Press, pp. 563-569.

Ryan McDonald, Koby Crammer and Fernando Pereira. 2005. *Online large-margin training of dependency parsers*. In Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics, pp. 91-98.

S. Abolghasem Mirroshandel and Gholamreza Ghassem-Sani. 2010. *Temporal Relations Learning with a Bootstrapped Crossdocument Classifier.* In Proceeding of the 2010 conference on ECAI 2010: 19th European Conference on Artificial Intelligence, pp. 829–834. IOS Press.

S. Abolghasem Mirroshandel, and Gholamreza Ghassem-Sani. 2011a. *Temporal Relation Classification Using Dependency Convolution Tree Kernels*. In Proceedings of the 5th Language & Technology Conference: Human Language Technologies as a Challenge for Computer Science and Linguistics, pp. 146-150.

S. Abolghasem Mirroshandel and Gholamreza Ghassem-Sani. 2012. *Towards Unsupervised Learning of Temporal Relations between Events*. Journal of Artificial Intelligence Research 45:125-163. DOI:10.1613/jair.3693.

S. Abolghasem Mirroshandel, Gholamreza Ghassem-Sani and Mahdy Khayyamian. 2011b. *Using syntactic-based kernels for classifying temporal relations.* Journal of Computer Science and Technology, 26(1):68-80.

Marzieh Mortazavinia. 2010. *A rule-based event detection System.* (Unpublished Persian Master degree Thesis). University of Tehran, Tehran, Iran.

Alessandro Moschitti. 2006a. *Efficient convolution tree kernels for dependency and constituent syntactic trees*, In Proceedings of the 17th European Conference on Machine Learning, pp. 318-329.

Alessandro Moschitti. 2006b. *Making tree kernels practical for natural language learning,* In Proceedings of EACL, pp. 113-120.

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

James Pustejovsky, Patrick Hanks, Roser Sauri, Andrew See, Robert Gaizauskas, Andrea Setzer, Dragomir Radev, Beth Sundheim, David Day, Lisa Ferro and Marcia Lazo. 2003. *The timebank corpus.* In Corpus Linguistics, Vol. 2003, pp. 40.

Mohammad S. Rasooli, Heshaam Faili, Behruz Minaei-Bidgoli, 2011. *Unsupervised identication of Persian compound verbs.* In Proceedings of the Mexican international conference on arti-cial intelligence (MICAI), pp. 394-406

Mehrnoush Shamsfard, Akbar Hesabi, Hakimeh Fadaei, Niloofar Mansoory, Ali Famian, Somayeh Bagherbeigi, Elham Fekri, Maliheh Monshizadeh, and S. Mostafa Assi. 2010. *Semi-automatic development of FarsNet; the Persian wordnet*. In Proceedings of 5th Global WordNet Conference (GWA2010).

Brett Spell. 2010. *Java API for WordNet Searching (JAWS)*, lyle.smu.edu/~tspell/jaws/index.html.

Naushad UzZaman and James F. Allen. 2010. *Trips and trios system for tempeval-2: Extracting temporal information from text.* In Proceedings of the 5th International Workshop on Semantic Evaluation, pp. 276-283. Association for Computational Linguistics.

Mark Verhagen, Robert Gaizauskas, Frank Schilder, Mark Hepple, Graham Katz and James Pustejovsky. 2007. *Semeval-2007 task 15: Tempeval temporal relation identification.* In Proceedings of the 4th International Workshop on Semantic Evaluations, pp. 75-80. Association for Computational Linguistics.

Mark Verhagen, Roser Sauri, Tommaso Caselli and James Pustejovsky. 2010. *Semeval-2010 task 13: Tempeval-2.* In Proceedings of the 5th International Workshop on Semantic Evaluation, pp. 57-62. Association for Computational Linguistics.

Yadollah Yaghoobzadeh, Gholamreza Ghassem-Sani, S. Abolghasem Mirroshandel, and Mahbaneh Eshaghzadeh. 2012. *ISO-TimeML Event Extraction in Persian Text*. In Proceedings of the 24th International Conference on Computational Linguistics (COLING), Mumbai, India, December 2012.

Katsumasa Yoshikawa, Sebastian Riedel, Masayuki Asahara and Yuji Matsumoto. 2009. *Jointly identifying temporal relations with markov logic.* In Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Vol. 1, pp. 405-413. Association for Computational Linguistics.

Min Zhang, Jie Zhang, Jian Su and Guodong Zhou. 2006. *A composite kernel to extract relations between entities with both flat and structured features.* In Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics, pp. 825-832. Association for Com-putational Linguistics.