Turkish Journal of Electrical Engineering & Computer Sciences

http://journals.tubitak.gov.tr/elektrik/

Research Article

Turk J Elec Eng & Comp Sci
(2018) 26: 570 – 581

doi:10.3906/elk-1703-79

Construction of a Turkish proposition bank

Koray AK*, Cansu TOPRAK, Volkan ESGEL, Olcay Taner YILDIZ
Department of Computer Engineering, Faculty of Engineering, Işık University, İstanbul, Turkey

Received: 07.03.2017 • **Accepted/Published Online:** 14.11.2017 • **Final Version:** 26.01.2018

Abstract: This paper describes our approach to developing the Turkish PropBank by adopting the semantic role-labeling guidelines of the original PropBank and using the translation of the English Penn-TreeBank as a resource. We discuss the semantic annotation process of the PropBank and language-specific cases for Turkish, the tools we have developed for annotation, and quality control for multiuser annotation. In the current phase of the project, more than 9500 sentences are semantically analyzed and predicate–argument information is extracted for 1330 verbs and 1914 verb senses. Our plan is to annotate 17,000 sentences by the end of 2017.

Key words: PropBank, semantic role labeling

1. Introduction

Over the past decade, adding a semantic information layer to syntactic annotation has gathered more attention, since improvements in machine learning algorithms and the development of computer systems enable and encourage processing large information resources. Semantic analysis of the corpora, along with the syntactic architecture, enhances natural language processing (NLP) applications such as information retrieval, machine translation, information extraction, and question answering. Via the added semantic layer, syntactic parser refinements can be achieved, which increases efficiency and improves application performance.

PropBank [1–3] is a study on this concept widely accepted by the computational linguistics community; it has been applied to different languages such as Arabic [4,5], Brazilian [6], Chinese [7], French [8], Hindi [9], Japanese [10], Spanish [11,12], Urdu [13], and Finnish [14].

In all those studies, PropBank is an essential part of understanding the predicate argument structure of the language. Only preliminary studies had been carried out in terms of the Turkish PropBank when we started working on this topic. However, Şahin recently presented her study for creating Turkish PropBank frames [15,16].

This paper is organized as follows: we first give brief information about PropBank annotation in Section 2. In Section 3, the semantically annotated dataset is described in detail. We describe the annotation process of the Turkish PropBank and the differences between our study and the study presented recently for the Turkish language in Section 4. We present the tool we have developed for the creation of frame files and annotation in Section 5, and conclude with the future goals of the project in Section 6.

*Correspondence: koray.ak@isik.edu.tr

2. PropBank annotation

PropBank [1–3] is a bank of propositions where predicate–argument information of the corpora is annotated and the semantic roles or arguments that each verb can take is posited. PropBank uses conceptual labels for arguments from Arg0 to Arg5. Only Arg0 and Arg1 indicate the same roles across different verbs, where Arg0 means agent or causer and Arg1 is the patient or theme. The rest of the argument roles can vary across different verbs. They can be instrument, start point, end point, beneficiary, or attribute. Moreover, PropBank uses ArgMs as modifier labels where the role is not specific to the verb group and generalizes over the corpora.

A list of ArgMs is presented in Figure 1.

DIR: Directionals
LOC: Locatives
MNR: Manner
EXT: Extent
REC: Reciprocals
PRD: Secondary Predication
PNC: Purpose
CAU: cause
DIS: discourse
ADV: adverbials
MOD: modals
NEG: negation

Figure 1. The list of ArgMs for tagging access modifiers of the predicate.

- Directional modifiers give information regarding the path of motion in the sentence. Directional modifiers may be mistakenly tagged as locatives.
- Locatives are used for the place where the action takes place.
- Manners define how the action is performed.
- Extent markers represent the amount of change that occurs in the action.
- Temporal modifiers keep the time of the action.
- Reciprocals are reflexives that refer to other arguments, like "himself," "itself," "together," "each other," and "both."
- Secondary predication markers are used for adjuncts of the predicate, which holds predicate structure.
- Purpose clauses show the motivation for the action. Cause clauses simply show the reason for an action.
- Discourse markers connect the sentence to the previous sentence, such as "also," "however," "as well," and "but."
- Adverbials are used for syntactic elements that modify the sentence and are not labeled with one of the modifier tags stated above.
- "Will," "may," "can," "must," "shall," "might," "should," "could," "would," and also "going (to)," "have (to)," and "used (to)" are modality adjuncts of the predicate and tagged as modal in PropBank.
- And finally, negation is used to tag negative markers of the sentences.

Prior to annotation, verbs of the corpora were analyzed and frame files were created as stated in the framing guidelines of the PropBank (http://verbs.colorado.edu/∼mpalmer/projects/ace/FramingGuidelines.pdf). Each verb has a frame file, which contains arguments applicable to that verb. Frame files may include more than one roleset with respect to the senses of the given verb. In the roleset of a verb sense, argument labels Arg0 to Arg5 are described with the meaning of the verb. For instance, Figure 2 presents the roles of predicate "attack" from PropBank: Arg0 is "attacker," Arg1 is "entity attacked," and Arg2 is "attribute."

Roleset id: attack.01 , *to make an attack, criticize strongly*,

attack.01: Member of Vncls judgement-33.

Roles:

Arg0-PAG: *attacker* (vnrole: 33-agent)
Arg1-PPT: *entity attacked* (vnrole: 33-theme)
Arg2-PRD: *attribute*

Figure 2. Roleset attack.01 from English PropBank for the verb "attack," which includes Arg0, Arg1, and Arg2 roles.

Since roles are described with the meaning of the verb, annotators can understand and annotate the verb instances. After the frame files are generated, annotation is straightforward. Whenever a sentence is annotated, the annotator selects a suitable frameset with respect to the predicate and then tags the sentence with the arguments provided in the frameset file. In Figure 3, a sample annotation is given where numbered arguments and ArgM's are represented.

He met Mr. Trump, in the White House, on Wednesday.

Rel : meet
Arg0: He
Arg1: Mr. Trump
ArgM-LOC : in the White House
ArgM-TMP : on Wednesday

Figure 3. Sample annotation. Arguments are annotated with Arg0, Arg1, and ArgMs.

In the annotation process, the PropBank Annotation Guidelines (https://verbs.colorado.edu/∼mpalmer/projects/ace/PBguidelines.pdf) are used as an instructive source, since they include several examples for different usages of the verbs. It guides annotators for linguistically complex argument structures of unaccusative and unergative verb instances that confuse annotators for selecting the right argument roles for causal agent, agent, and theme.

PropBank also offers solutions to annotation disagreements by adopting blind and double annotation. Whenever a disagreement occurs, an adjudicator decides the correct annotation and new roles may be added to the roleset.

3. Semantically annotated dataset

Before introducing the PropBank, one needs semantically annotated sentences. Each semantically annotated sentence will have a main verb for which we later create frame files. For that reason, we selected 9560 sentences containing a maximum of 15 tokens from the translated Penn Treebank II [17,18]. These include 8660 sentences from the training set of the Penn Treebank [19], 360 sentences from its development set, and 540 sentences from its test set.

Using the Penn Treebank structure has the advantage of building a fully tagged data set that has coherency between syntactic labels, morphological labels, and parallel sentences in English and Turkish. Moreover, the data set is stored in a simple format covering all linguistic label stages. This simple format enables automatic semantic labeling using other corresponding linguistic labels or high-level machine learning methods.

3.1. Semantic annotation tool

Manual annotation is a fallible task, especially in a big data set. A minor mistake from an annotator can lead to a major impact on data. The existence of multiple annotators and possible disagreement among them further complicates this issue. While it is not possible to completely eliminate this source of error, we tried to minimize the impact. Our tool visualizes sentences with their possible meanings below each word. While our tool shows possible meanings for all words in the sentence, we only consider the predicate of the sentence from the PropBank point of view. Since semantic annotation of the verbs is crucial before creating the frame files, we detect predicates in the sentences before analyzing sentences morphologically and semantically.

The bottom of the tool displays the sentence pairs that the word in question belongs to. The color of the Turkish sentence indicates whether all of the words have been tagged or not. Here we only need the semantic annotation of the verbs, but we also annotated the rest of the sentence too.

4. Turkish PropBank

The semantically annotated dataset described in Section 3 is processed with respect to predicate–argument structure. Prior to semantic role labeling, preliminary steps are described in the next subsection, and annotation is clarified in the latter.

4.1. Preliminary steps

After morphologically and semantically analyzing predicates, rolesets for each verb sense in the corpora should be specified in advance for the annotation process. In order to do this, a verb-sense list for semantically analyzed corpora is consolidated as in Table 1.

Table 1. Sample verb sense list. Number of occurrences is presented for each sense of the predicate.

Verb	Sense ID	# Occurrences
işlemek	1	13
işlemek	5	2
işlemek	8	9
işlemek	12	1
işlemek	13	2
itmek	1	1
itmek	3	1
itmek	4	2
izlemek	1	5
izlemek	2	1
izlemek	3	3

In the next step, all sentences that contain the verb-sense instance are examined in terms of predicate argument structure; semantic roles for that sense are identified. Unlike the original PropBank frame files, where each verb has a file with different rolesets for each different sense, we decide to use one xml file, which contains

all verbs and their senses, respectively, for the sake of simplicity in the current architecture. For each verb sense, a "FRAMESET" tag with the unique sense id as attribute is inserted inside the "FRAME" tags in the file. We use "ARG" tags for each role that a predicate has. "ARG" tags take a name attribute as the name of the argument and semantic meaning inside. Figure 4 shows a part of the frame file. For each verb sense we have assigned a unique synset id. These synset ids are used to individuate frames. Each sense has different number of arguments with different names with respect to the semantic information. The first frame in Figure 4 has synset id "TUR10-0006410," which corresponds to the third sense of the verb "açmak" in our dictionary file. Inside the frameset tags, arguments are listed for this verb sense.

```
<FRAMES>
    <FRAMESET id="TUR10-0006410">
        <ARG name="ARG0">Açan</ARG>
        <ARG name="ARG1">Açılan şey</ARG>
        <ARG name="ARGMTMP">Açılma zamanı</ARG>
        <ARG name="ARGMADV">Zarf tümleci</ARG>
    </FRAMESET>
    <FRAMESET id="TUR10-0006500">
        <ARG name="ARG0">Açan</ARG>
        <ARG name="ARG1">Açılan şey</ARG>
        <ARG name="ARGMTMP">Açılma zamanı</ARG>
        <ARG name="ARGMEXT">Miktar</ARG>
        <ARG name="ARGMDIS">Bağlaç""</ARG>
        <ARG name="ARGMLOC">Açma yeri</ARG>
    </FRAMESET>
    <FRAMESET id="TUR10-0042580">
        <ARG name="ARG0">Arayan</ARG>
        <ARG name="ARG1">Aranan</ARG>
        <ARG name="ARGMTMP">Arama zamanı</ARG>
        <ARG name="ARGMLOC">Arama yeri</ARG>
    </FRAMESET>
    <FRAMESET id="TUR10-0432220">
        <ARG name="ARG0">Arayan</ARG>
        <ARG name="ARG1">Aranan şey</ARG>
        <ARG name="ARGMPNC">Arama amacı</ARG>
        <ARG name="ARGMLOC">Arama yeri</ARG>
    </FRAMESET>
```

Figure 4. A part of the frame file. Different framesets with unique verb sense ids are listed under FRAMES tag. In each frameset, a list of roles for that verb sense is stored.

We used seven framers, all part of an NLP research group, to construct rolesets. Video guidelines for creating frame files and annotation are prepared based on the original PropBank framing and annotation guidelines. These framers are educated before starting to build frame files. The corpus is divided into seven parts in terms of verb occurrences, and each verb interval is assigned to a team member.

In the framing phase, the team discover several errors such as mistranslations, false morphological analyses, and erroneous sense selections. We keep track of these errors and raise the problem with the team immediately. Translation errors are sent to the translators to be retranslated. Analysis errors are solved inside

the team. If the sense of the predicate is false, a correct sense is selected and the sentence is reexamined for extracting the new sense's semantic roles.

Since we combine related verb senses into the same frameset, we create 1740 framesets for 1914 verb senses. In Table 2, the numbers of occurrences of the arguments and modifiers are listed. As we expected, more than 78% of the verb senses have Agent and 74% have Theme/Patient. Note that we did not cover the prodropped subjects while deciding the roles of the predicate. In Turkish, we can detect a hidden subject from the inflected forms of the verbs, which may increase the number of Agent roles among the verb senses. Moreover, modifiers are extensively detected in the sentences, with a total of 1951 ArgM's inserted into the framesets. The most frequent modifiers are ArgM-DIS, ArgM-LOC, ArgM-MNR, and ArgM-TMP, with more than 200 occurrences.

Table 2. Number of arguments and modifiers for 1914 verb senses in Turkish PropBank.

Argument	# Occurrences
Arg0	1360
Arg1	1289
Arg2	142
Arg3	10
Arg4	3
Arg5	1
ArgM-CAU	63
ArgM-DIS	336
ArgM-DIR	27
ArgM-EXT	154
ArgM-LOC	267
ArgM-MNR	319
ArgM-ADV	172
ArgM-PNC	127
ArgM-TMP	486

4.2. Annotation process

Once framesets are constructed, annotation of the translated corpora is straightforward. The same team members perform annotation. This time, the divided corpora are distributed in different order to avoid having the same person complete both framing and annotation. Thus, any defect in the framing phase is revealed during the annotation phase. In such a case, annotator requests adding the necessary role to the roleset of the verb sense and the annotation phase becomes a control for the framing phase.

Although PropBank annotation guidelines provide a crucial basis for our guidelines, some of the arguments are omitted in our context. For some argument types, such as ArgM-MOD and ArgM-NEG, there is no equivalent word in Turkish. Instead, this semantic information is concealed in the suffixes added to the verb, as in Figure 5. Turkish, as an agglutinative language, permits any given verb to use modal or negative suffixes. Furthermore, English passive or causative word groups can be equivalent to a single Turkish verb that contains derivational affixes. These derived verbs are included as different verbs in our dictionary, regardless of having the same semantic basis, and so they are interpreted as different verbs in the current architecture.

As another step to ensure the quality and reliability of the annotations, an expert annotator outside the team annotated randomly selected 250 sentences with previously constructed framesets. The annotator added

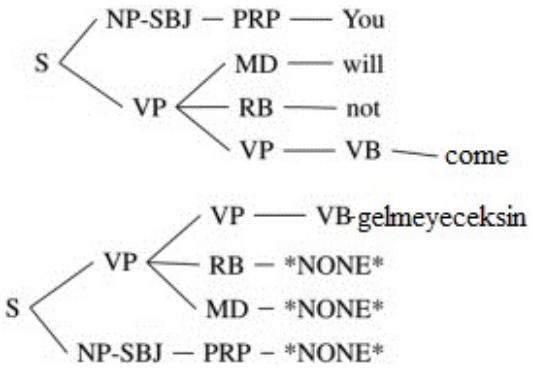

Figure 5. English parse tree and translated Turkish counterpart. Modal and negation words in the English sentence are represented as suffixes of the predicate in Turkish sentence.

new rolesets to the frame file whenever a new role was needed. Then the sentences annotated by the team were compared with the sentences annotated by the expert annotator. Out of the 2213 tagged roles, 1879 matched, which corresponds to an 85% agreement.

4.3. Other PropBank studies for Turkish language

Although PropBank studies are widespread among different languages, there exist only a couple of studies for Turkish. Recently, Şahin presented their study for creating Turkish PropBank frames [15,16] by incorporating crowdsourcing techniques. They construct PropBank by using the ITU-METU-Sabancı Treebank. They used crowd intelligence to deal with verb sense annotation prior to the frame creation task. Then they created frame files for 1262 verb senses by using Turkish Dependency Treebank as a resource. On the other hand, we started to work on the subject before these publications were issued. Our intention was to create a Turkish PropBank by using the resources available for English PropBank, and we used the translated corpus of the English PropBank Kingsbury and Palmer [1–3] to create framesets. The parallelism between our work and English PropBank may give us an opportunity to get enough corpora for manual annotation. Moreover, since our data are aligned with the English PropBank, we can integrate automatic PropBank generation algorithms such as annotation projection into our study, which can speed up the creation and expansion of the Turkish PropBank. Also via this parallelism, we can adopt any future changes to the Turkish PropBank easily. Another difference between our study and Şahin is that we use an in-house NLP toolkit, whereas she used an open source tool and Cornerstone for frame creation. The NLP toolkit allows us to process corpora before starting frame creation and annotation. It consists of different modules that support translation, morphologic, and semantic analysis. Finally, to the best of our knowledge, in these studies, only frame files of the Turkish PropBank were constructed. We have already annotated 5000 sentences from the translated corpus of English PropBank.

5. Annotation tool

As PropBank construction is spread out across many different languages, many different tools are proposed for frame composition and annotation. Available tools for PropBank annotation and frames file edition are Jubilee and Cornerstone [20,21], SALTO [22], etc. Yet already, we are using an in-house NLP toolkit that supports all the operations mentioned in the Turkish PropBank section above. Therefore we have integrated a PropBank editor into our toolkit to use the same infrastructure.

A screenshot from the PropBank editor is presented in Figure 6. The list of verbs in the corpora is placed on the left side of the screen. Verbs are presented in different colors according to the stages they are currently in. If a verb does not have a frame for all senses, then it is shown in red. If all the senses of the verb have frames, then it turns green. Finally, if all the instances of the verb are annotated, it turns black. Once a verb is clicked in the list, sense numbers are listed below. Sense selection triggers the application to load the first sentence tree to the middle, and both English and Turkish sentences to the bar below. The data file name and sense explanations are also written above the sentence tree.

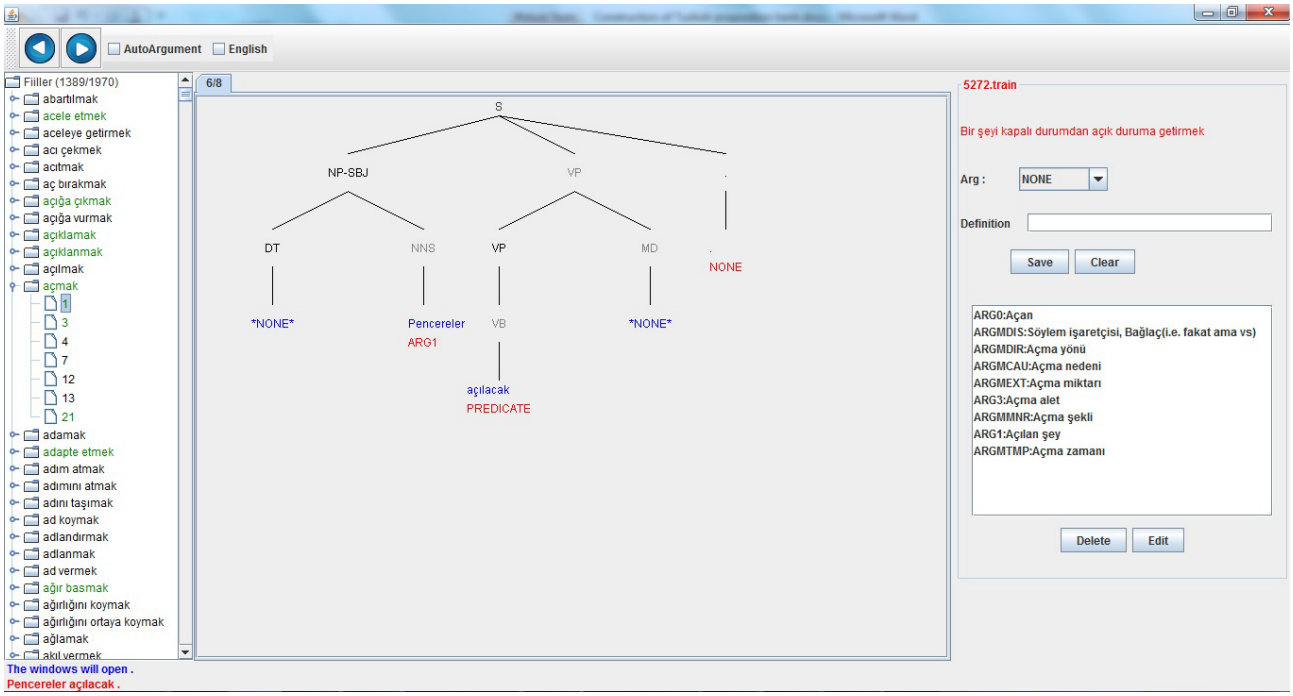

Figure 6. A screenshot of the PropBank semantic labeling tool. Verbs are listed in the left pane and sentences with the selected verb are positioned in the middle pane where the annotator can tag each leaf in terms of PropBank roles. Role list for that verb is managed in the right pane.

In the framing phase, framers use this screen to examine all sentences of the verb sense by visiting all the trees via arrow buttons. Whenever a new role is found, it can be added to a frame file by filling the form in the right panel. This panel helps framers select the desired argument type in the dropdown menu above the semantic definition of the argument. After the save button is clicked, an argument record is added to the frameset of the verb sense. Framers can also delete or edit the argument records with the corresponding buttons (Figure 7).

When a framer finishes all the sentences for a verb sense, rolesets are ready for the annotation. Annotation again takes place in the same screen. Tree nodes in the middle area are clickable. Once the user clicks a node, a role list that can be assigned to the node pops up (Figure 8). After the selection is made, the selected role is printed below the node. In the same manner, the annotator can visit all the sentences of the verb sense with the arrow buttons above.

As previously stated, annotation of sentences with unergative and unaccusative verbs as predicates is highly complex. These verb structures require additional attention, since agent and theme annotation is

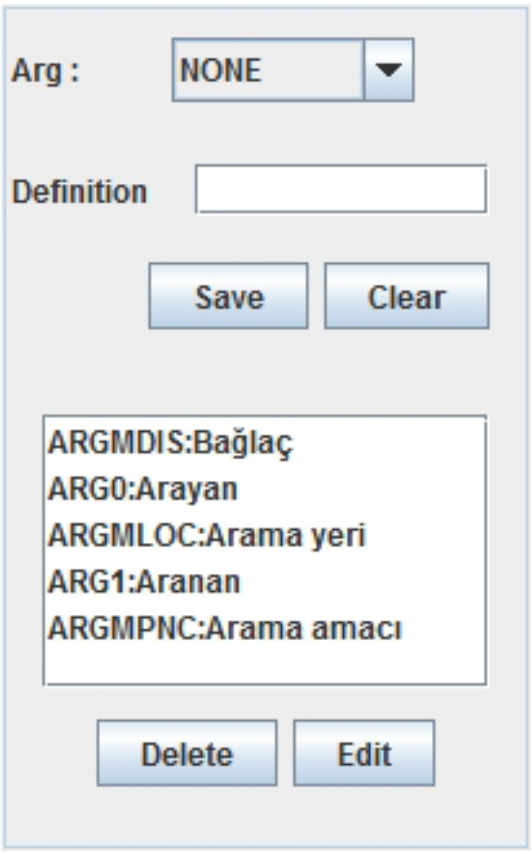

Figure 7. Argument insertion for the selected verb. Users first select the argument type and then edit the definition field with the meaning of the argument for the selected verb.

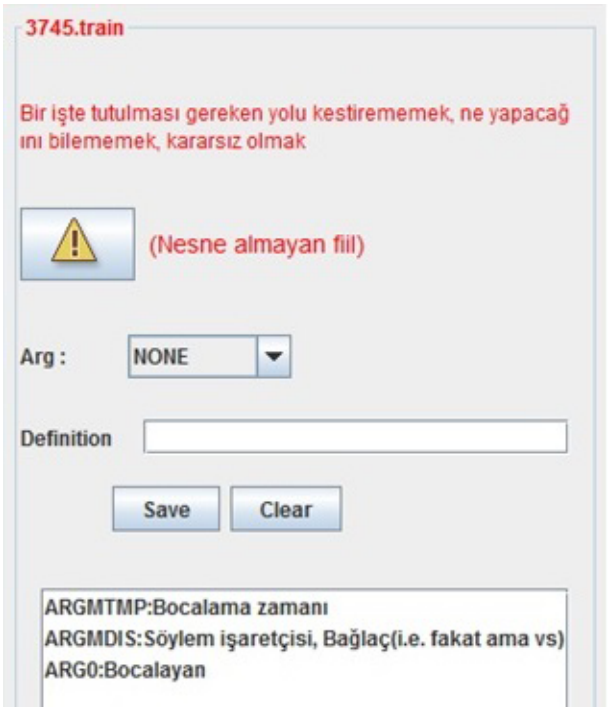

Figure 8. Annotation for the selected sentence is done by simply selecting the arguments of the predicate from the dropdown for each word.

confusable. Subjects of unaccusative verbs should be tagged with Arg1 just like passive sentences. As an example, in the sentence "Gemi battı," "Gemi" is not Arg0. In addition, in some cases, the subject of the sentences becomes neither Arg0 nor Arg1 but an indirect object. To get rid of erroneous annotations, we also add warnings to the annotation interface for complex verb structures. Whenever an annotator attempts to annotate a complex verb instance, a warning sign with a description about the verb structure appears below the verb meaning, as in Figure 9.

We also provide more functionality to make the annotation process even easier. Two check boxes are added to the top of the screen. The "Auto Argument" check box is used for predicting arguments in the sentence by evaluating the simple rules we have designed. These rules use a greedy approach for determining the argument tag. We simply detect nodes in the path from root to the leaf node where the word resides, and look to the syntactic tags for determining the word tag.

The second check box, "English," shows the argument labels for the words in the original English sentence below the corresponding Turkish words. Since we use translated sentences from English PropBank, most of our sentences have equivalent annotated English sentences. These annotated sentences guide and help annotators to check their annotation. Figure 10 shows an example view where arguments of the original sentence are listed under Turkish translation.

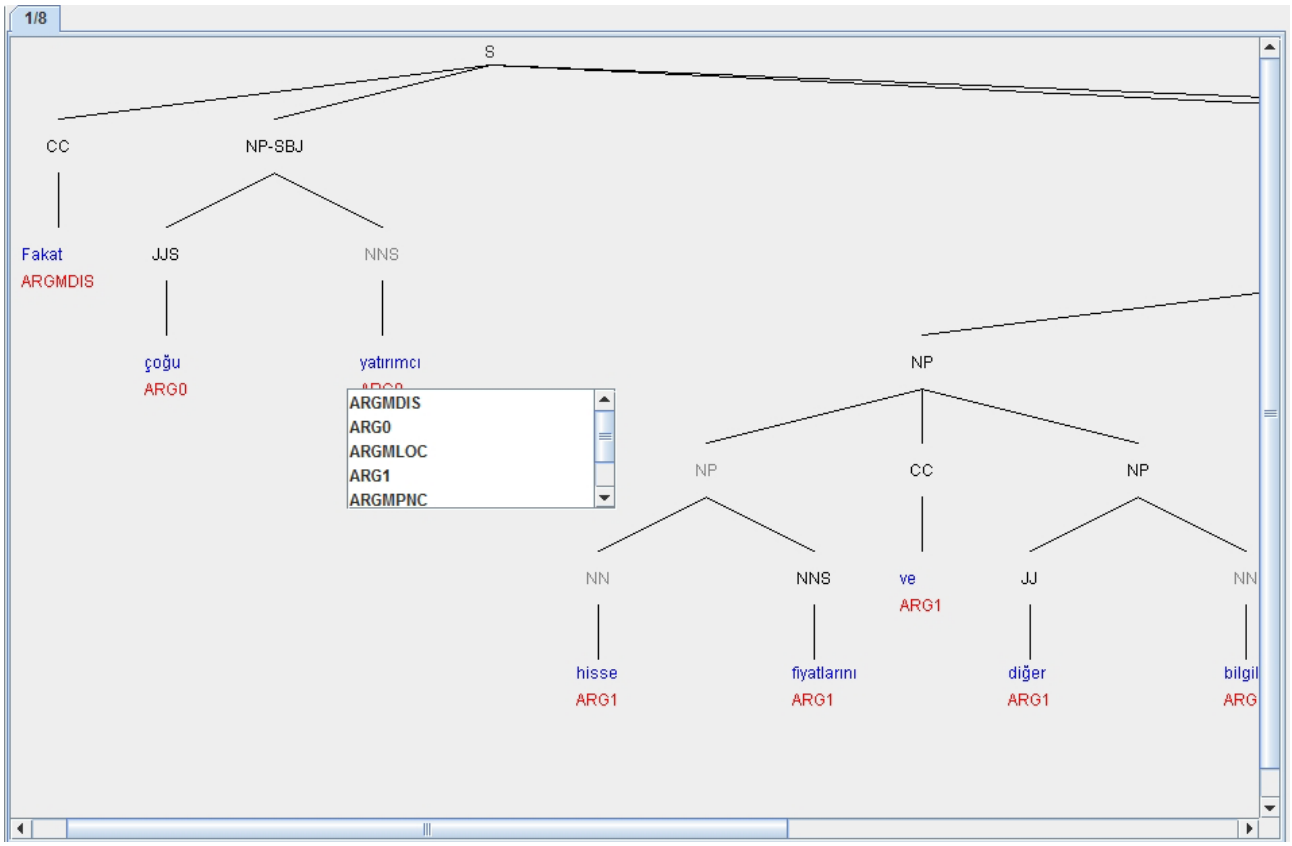

Figure 9. Information about verb structure is shown on the interface as a warning for complex verbs.

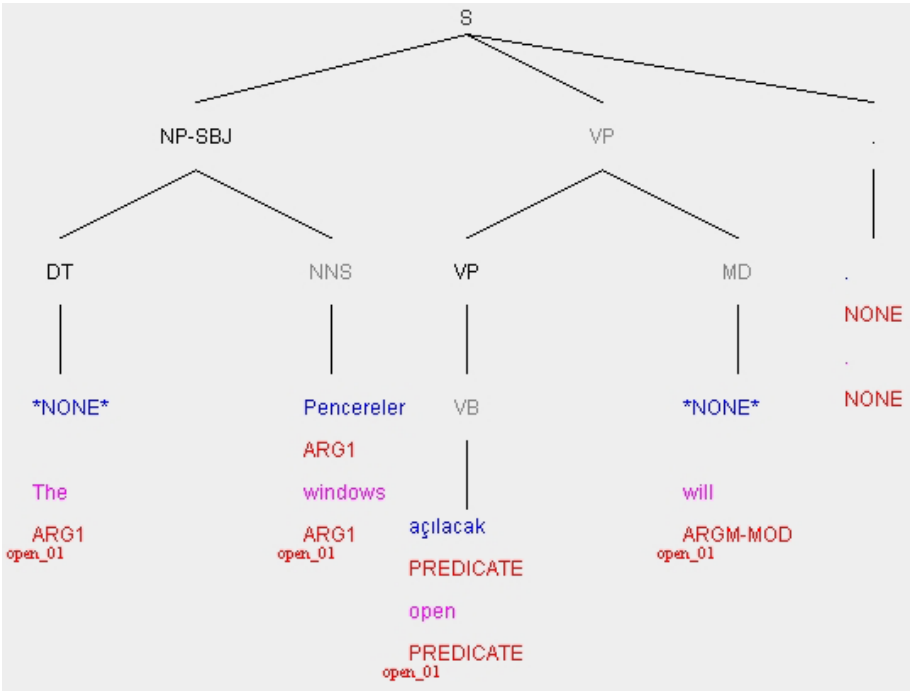

Figure 10. Arguments annotated for English sentence under the Turkish equivalent.

6. Conclusion

PropBank annotation is well defined and applied to various different languages. As a morphologically rich language, Turkish is overdue for investigation with PropBank. We started the study on the construction of Turkish PropBank with the path sketched by the previous works on the field. Using the translation of the English Penn-TreeBank as a resource speeds up our study.

As a future work, we should clarify annotation rules for the agglutinative structure of the language. Modal and negation like semantic information may be generalized for all verb senses. Derivational affixes, which make new verbs from already annotated verbs, can be stated and automatically added to the same verb sense, such as in the causative and passive sentences case mentioned before.

Currently, more than 9500 sentences are semantically analyzed and 1914 verb senses are annotated. We will simultaneously add new verbs and continue the annotation of our corpora. The developed Turkish PropBank will be made public and users can access the frames and annotations through a website (http://haydut.isikun.edu.tr/propbank-web). We are planning to annotate 17,000 sentences by the end of 2017. Afterwards, we are planning to use Turkish PropBank in other NLP topics. [7]

Acknowledgment

This work was supported by Işık University BAP projects 14B206 and 15B201.

References

[1] Kingsbury P, Palmer M. From treebank to propbank. In: International Conference on Language Resources and Evaluation; 29–31 May 2002; Canary Islands, Spain. pp. 1989-1993.

[2] Kingsbury P, Palmer M. Propbank: the next level of treebank. In: Workshop of Treebanks and Lexical Theories; 2003; Växsjö, Sweden.

[3] Palmer M, Gildea D, Kingsbury P. The proposition bank: an annotated corpus of semantic roles. Comput Linguist 2005; 31: 71-106.

[4] Palmer M, Babko-Malaya O, Bies A, Diab M, Maamouri M, Mansouri A, Zaghouani W. A pilot Arabic propbank. In: International Conference on Language Resources and Evaluation; 26 May–1 June 2008; Marrakech, Morocco. pp. 3467-3472.

[5] Zaghouani W, Diab M, Mansouri A, Pradhan S, Palmer M. The revised Arabic propbank. In: Linguistic Annotation Workshop; 15–16 July 2010; Uppsala, Sweden. Stroudsburg, PA, USA: The Association for Computational Linguistics. pp. 222-226.

[6] Duran MS, Alusio SM. Propbank-br: a Brazilian treebank annotated with semantic role labels. In: International Conference on Language Resources and Evaluation; 23–25 May 2012; İstanbul, Turkey. pp. 1862-1867.

[7] Xue N, Palmer M. Adding semantic roles to the Chinese treebank. Nat Lang Eng 2009; 15: 143-172.

[8] Van der Plas L, Samardžić T, Merlo P. Cross-lingual validity of Propbank in the manual annotation of French. In: Linguistic Annotation Workshop; 15–16 July 2010; Uppsala, Sweden. Stroudsburg, PA, USA: The Association for Computational Linguistics. pp. 113-117.

[9] Vaidya A, Choi JD, Palmer M, Narasimhan B. Analysis of the Hindi proposition bank using dependency structure. In: Linguistic Annotation Workshop; 23–24 June 2011; Portland, OR, USA. Stroudsburg, PA, USA: The Association for Computational Linguistics. pp. 21-29.

[10] Kawahara D, Kurohashi S, Hasida K. Construction of a Japanese relevance-tagged corpus. In: International Conference on Language Resources and Evaluation; 29–31 May 2002; Canary Islands, Spain. pp. 2008-2013.

[11] Màrquez L, Villarejo L, Martí MA, Taulé M. Semeval-2007 task 09: Multilevel semantic annotation of Catalan and Spanish. In: International Workshop on Semantic Evaluations; 23–24 June 2007; Prague, Czech Republic. Stroudsburg, PA, USA: The Association for Computational Linguistics. pp. 42-47.

[12] Taulé M, Martí MA, Recasens M. Ancora: multilevel annotated corpora for Catalan and Spanish. In: International Conference on Language Resources and Evaluation; 26 May–1 June 2008; Marrakech, Morocco. pp. 96-101.

[13] Mukund S, Ghosh D, Srihari RK. Using cross-lingual projections to generate semantic role labeled corpus for Urdu: a resource poor language. In: International Conference on Computational Linguistics; 23–27 August 2010; Beijing, China. pp. 797-805.

[14] Haverinen K, Kanerva J, Kohonen, S Missilä A, Ojala S, Viljanen T, Laippala V, Ginter F. The Finnish Proposition Bank. Lang Resour Eval 2015; 4: 907-926.

[15] Şahin GG. Framing of verbs for Turkish propbank. In: Turkish Computational Linguistics; 3–9 April 2016; Konya, Turkey.

[16] Şahin GG. Verb sense annotation for Turkish propbank via crowdsourcing. In: International Conference on Intelligent Text Processing and Computational Linguistics; 3–9 April 2016; Konya, Turkey.

[17] Yıldız OT, Solak E, Görgün O, Ehsani R. Constructing a Turkish-English parallel treebank. In: International Conference on Computational Linguistics; 22–27 June 2014; Baltimore, MD, USA. pp. 112-117.

[18] Yıldız OT, Solak E, Çandır Ş, Ehsani R, Görgün O. Constructing a Turkish constituency parse treebank. In: International Symposium on Computer and Information Sciences; 21–24 December 2015; London, UK. pp. 339-347.

[19] Marcus MP, Marcinkiewicz MA, Santorini B. Building a large annotated corpus of English: the Penn Treebank. Comput Linguist 1993; 19: 313-330.

[20] Choi JD, Bonial C, Palmer M. Propbank instance annotation guidelines using a dedicated editor, Jubilee. In: International Conference on Language Resources and Evaluation; 17–23 May 2010; Valletta, Malta. pp. 1871-1875.

[21] Choi JD, Bonial C, Palmer M. Propbank frameset annotation guidelines using a dedicated editor, Cornerstone. In: International Conference on Language Resources and Evaluation; 17–23 May 2010; Valletta, Malta. pp. 3650-3653.

[22] Burchardt A, Erk K, Frank A, Kowalski A, Padó S, Pinkal M. Salto – a versatile multi-level annotation tool. In: International Conference on Language Resources and Evaluation; 24–26 May 2006; Genoa, Italy. pp. 517-520.