Enhancements to the BOUN Treebank Reflecting the Agglutinative Nature of Turkish

Büşra Marşan[1], Salih Furkan Akkurt[2], Muhammet Şen[2], Merve Gürbüz[2], Onur Güngör[2], Şaziye Betül Özateş[2], Suzan Üsküdarlı[2], Arzucan Özgür[2], Tunga Güngör[2] and Balkız Öztürk[1]

[1]*Boğaziçi University, Linguistics*
[2]*Boğaziçi University, Computer Engineering*

Abstract

In this study, we aim to offer linguistically motivated solutions to resolve the issues of the lack of representation of null morphemes, highly productive derivational processes, and syncretic morphemes of Turkish in the BOUN Treebank without diverging from the Universal Dependencies framework. In order to tackle these issues, new annotation conventions were introduced by splitting certain lemmas and employing the MISC (miscellaneous) tab in the UD framework to denote derivation. Representational capabilities of the re-annotated treebank were tested on a LSTM-based dependency parser and an updated version of the BoAT Tool is introduced.

Keywords

Universal Dependencies, Turkish, morphological analysis, dependency annotation, dependency parsing

1. Introduction

Following the dependency grammar framework first proposed by Tesniére [1], dependency trees illustrate how sentence elements relate to one another through head and dependent relations. Universal Dependencies[1] (UD) is an international cooperative treebank project based on the dependency grammar framework and it aims to offer a standardized and comprehensive dependency treebank collection covering 121 languages.

With the addition of new UD treebanks, Turkish does not qualify as a low resource language anymore. With a total of 733,000 tokens, it is the 12th largest UD treebank in the UD repository. Although the coverage of the treebanks plays an essential role in improving the performance of natural language processing (NLP) systems [2], their ability to correctly and consistently illustrate the morphosyntactic features of the target language should not be overlooked. As Vincze *et al.* [3]'s study shows, the better a treebank's ability to represent the morphology

The International Conference and Workshop on Agglutinative Language Technologies as a challenge of Natural Language Processing (ALTNLP), June 7-8, Koper, Slovenia
busra.marsan@boun.edu.tr (B. Marşan); furkan.akkurt@boun.edu.tr (S. F. Akkurt); muhammet.sen@boun.edu.tr (M. Şen); merve.gurbuz@boun.edu.tr (M. Gürbüz); onurgu@boun.edu.tr (O. Güngör); saziye.bilgin@boun.edu.tr (B. Özateş); suzan.uskudarli@boun.edu.tr (S. Üsküdarlı); arzucan.ozgur@boun.edu.tr (A. Özgür); gungort@boun.edu.tr (T. Güngör); balkiz.ozturk@boun.edu.tr (B. Öztürk)

[1]https://universaldependencies.org

and syntax of the target language, the better the performance of the NLP systems using that treebank as a resource.

In this paper, we aim to abide by the linguistic framework set by Bedir *et al.* [4] and offer an updated and comprehensive UD treebank for Turkish, the BOUN Treebank, along with an improved UD annotation interface, the BoAT Tool, first introduced in Türk *et al.* [5].

The decisions made in the re-annotation process of the BOUN Treebank aim to offer solutions to the issues posed by the morphologically rich and complex nature of Turkish: null morphemes are frequently employed, agglutinative processes are heavily used to create new forms, and numerous morphemes like copula and -ki are very syncretic. The main goal of this study is to illustrate these phenomena without compromising the compliance with the UD framework.

This paper is organized as follows. Previous attempts at creating dependency treebanks in Turkish are laid out in Section 2. Annotation changes made in the current version of the BOUN Treebank and their linguistic justification are discussed in Section 3. Statistics regarding the changes made to the previous version are stated in Section 4. Improvements made to the BoAT Annotation Tool are explained in Section 5. Finally, the performance of the parser trained using the current version of the treebank is reviewed in Section 6.

2. Dependency Treebanks in Turkish

Shortly after the first dependency treebank for Turkish was presented by Atalay *et al.* [6], Eryiğit and Pamay [7] offered a smaller dependency treebank consisting of 300 sentences as part of the CoNLL 2007 Shared Task: MST Treebank. 13 years after its publication, Sulubacak et al. [8] re-annotated this dataset, converted it to the UD framework, and published the updated dataset as IMST-UD. Çöltekin's The Grammar Book Treebank (GB) [9] which consists of 2,803 sentences extracted by a reference book on Turkish grammar by Göksel and Kerslake [10] marks the very first effort in creating the first UD-style Turkish treebank. Another pioneer in Turkish dependency treebanks is IWT-UD as it is the first Turkish dependency treebank that covers informal texts. IWT-UD was introduced by Sulubacak and Eryiğit [11] three years after a constituency-style treebank, IWT, was presented by Pamay et al. [12].

Tourism is one of the two domain-specific dependency treebanks in Turkish and consists of hotel and restaurant reviews. The other one is ATIS treebank that covers the Turkish translation of English ATIS (Airline Travel Information System) corpus [13].

Other Turkish UD-style treebanks include Kenet UD Treebank, Penn Treebank, and FrameNet treebank. Penn Treebank consists of Turkish translations of English Penn Treebank [14] while FrameNet consists of 2,700 sentences from the Turkish FrameNet database [15].

With 9,761 sentences and 121,214 tokens randomly selected from Turkish National Corpus (TNC) [16], the BOUN Treebank is one of the largest UD-style treebanks in Turkish. Covering five different registers (broadsheet national newspapers, biographical texts, essays, popular culture articles, and instructional texts), it offers word order and sentence length variance in addition to linguistically motivated dependency annotations (see Section 3).

3. Improving BOUN Treebank

The first step of the previous annotation process of the BOUN Treebank (see [5] for a detailed discussion) was parsing the raw text to create CoNLL-U files using Kanerva et al.'s [17] pipeline tool. During this parsing process, UPOS tags and certain morphological information were automatically annotated. Then the dependency relations were manually annotated by two native speakers of Turkish who are linguists. To ensure inter-annotator agreement, randomly selected 1000 sentences were double annotated. Using Cohen's Kappa measure, inter-annotator agreement was calculated. Dependency label match score was 0.82, unlabelled attachment score was 0.83, and labelled attachment score was 0.75.

3.1. The Re-annotation Process: Overcoming the Challenges

For the re-annotation process, a team of linguists detected shortcomings and problematic annotations of the previous version of the BOUN Treebank [5]. Two major representation challenges were detected: derivation and null copula. The following subsections discuss the strategies employed to overcome these challenges.

Annotations were done by two linguists who are native speakers of Turkish. In order to ensure inter-annotator agreement, 100 sentences were double annotated. Unlabelled attachment and labelled attachment scores were calculated using Cohen's Kappa measure. Their respective values are 98.61 and 97.81.

3.1.1. Derivation

Having its focus on syntax, the UD framework falls short of representing derivational processes. The official guide of UD argues that the final derivational suffix in a word is opaque in the sense that it does not permit access to the morphemes that come before it. Hence in a construction as the one shown in Figure 1, numerous derivational and inflectional morphemes before the last derivational morpheme (-ki) are lost.

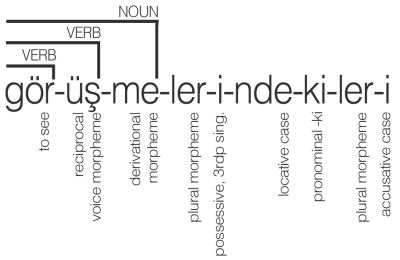

Figure 1: Derivational and inflectional analysis of "görüşmelerindeki" (*"those who were/are at his/her meetings"*)

As one of the primary concerns of this study was finding ways to illustrate derivation processes like those in Figure 1 without diverging from the UD framework, two strategies were employed for different cases:

- For morphemes like -lI (*"with"*), `df=` function is introduced in the MISC tab.
- Lemmas containing -ki morpheme are splitted.

These derivational morphemes imply that the host lemma and its modifier form a syntactic unit together. As a result, the correct bracketing of the -lI adjective, its modifier, and the constituent they modify together should look like (2) instead of (1). Representing the derivational morphemes like -lI and -sIz allows keeping this crucial syntactic information.[2]

(1) *[[[kahverengi] tüy-lü] kedi]*
brown fur-ATTR cat
"a cat with brown fur"

(2) *[[kahverengi tüy]-lü kedi]*
brown fur-ATTR cat
"a cat with brown fur"

With the new `df=` function proposed in this study, "tüylü" is not decomposed as two lemmas: "tüy" (*"fur"*) and "lü" (*"with"*). Instead, it is left intact and `df=tüy` ("derived from=fur") function is annotated in the MISC tab.

Another challenge regarding the derivational processes is posed by -ki. There are two different -ki morphemes in Turkish [18]. One is used as noun and the other derives adjectives from nouns.

Statistically, the vast majority of inflectional suffixes in Turkish occur after the derivational ones. However, both types of the -ki morphemes diverge from this distribution as it can be observed in Figure 1, where the inflectional suffix -ler precedes the derivational -ki. Hence allowing -ki to "block access" [4][3] to the morphemes that were attached before it reduces the capabilities of the annotation to represent a great deal of morphosyntactic information. With the aim of offering a solution to this issue, a decision to split lemmas that contain either type of -ki was made.[4] Although splitting lemmas is not a common practice within the UD framework, we believe that the theoretical motivation behind this decision justifies the divergence from the framework. (For a detailed linguistic discussion of this issue, please refer to Bedir *et al.* [4])

3.1.2. Null Morpheme

Languages such as Turkish, Russian, Arabic, and Coptic have null morphemes, however, the UD framework does not officially support such phenomena. As a result, independent strategies have emerged to represent null morphemes in these languages. For example, Coptic avoids null subject nodes by using fused forms[19], Marathi annotates the feature values of the null morphemes, however, does not introduce any information (i.e. annotation) to indicate that they are null morphemes[20]. Widely employing null morpheme for pluralization, Arabic distinguishes between two types of annotation: Form-based and function-based. In their UD-style dependency treebank annotation, Marton *et al.*[21] follow a function-based annotation framework and annotate the feature values of null morphemes.

In Turkish, the copula can surface in three different forms [10]: i-, -y-, and ∅.After considering UD guidelines and particularities of Turkish copula, it was decided to employ the MISC tab

[2] Abbreviations: ATTR = attributive.

[3] It is stated in the UD guidelines that "the lemma does not remove derivational morphology, so the lemma of [en] "organizations" is "organization" not "organize" (nor "organ")." See https://universaldependencies.org/u/overview/morphology.html for the full picture.

[4] Refer to the Appendix to compare previous and updated annotation schemes for both -ki morphemes.

again by introducing two new functions for the null copula: `nullcop=3s` (singular) and `nullcop=3p` (plural). By following a function-based annotation schema, we were able to offer more linguistically accurate annotations without diverging from the UD framework.

3.1.3. Copula

In Turkish, ol- copula has six distinct functions [4]: An intransitive verb meaning “to be suitable/fit", a transitive verb meaning “to become", an auxiliary verb in embedded sentences, an auxiliary verb following the participle, a light verb forming complex verbal constructions (such as “sorun olmak" (*“to be/become an issue"*)), and finally the existential predicate that surfaces as “var" (*“to exist"*) and “yok" (*“not to exist"*). Yet the previous annotation scheme of the BOUN Treebank made no distinction between these different usages. To offer more accurate representations, certain annotation changes were made regarding these functions.

3.2. Newly Introduced XPOS Tags and Dependency Relations

In an attempt to overcome the challenges thoroughly discussed in the previous subsection, a set of new XPOS tags and dependency relations were introduced in the updated version of the BOUN Treebank. A comprehensive list can be found in Table 1.

Lemma	**var**	**yok**	**ol- (after participle)**	**ol- (in embedded sentences)**	**ol- (in light verb constructions)**	**ol- (as transitive or intransitive verb)**	**-ki (adjectivizer)**	**-ki (pronominal)**
UPOS	NOUN	NOUN	AUX	AUX	VERB	VERB	PART	PRON
XPOS	Exist	Exist		Ptcp			Attr	Partic
Deprel	root	root	aux	cop	compound:lvc	root	dep:der	

Table 1
New dependency relations and XPOS tags proposed for the updated BOUN Treebank.

4. Statistics

As part of the re-annotation process, 117,732 changes were made in the following tabs: UPOS, XPOS, Deprel, MISC, and Features. The majority of annotation changes targeted UPOS and XPOS tags (see Table 2). Since morphological information was automatically annotated in the previous version of the treebank by the parsing tool [17], refinements by the annotators were required in order to ensure accuracy.

Field	**UPOS**	**XPOS**	**Features**	**Deprel**	**MISC**
Changes	11,396	63,829	27,098	23,32	4,973

Table 2
Changes made in the re-annotation process.

The changes in UPOS values reflect the linguistics-based decisions made in the re-annotation process. Previous version of BOUN Treebank made no distinction between two -ki morphemes in Turkish. As a result, almost all -ki instances were labeled as `CConj` (clausal conjunction).

After deciding to make a distinction between adjectivizer -ki and pronominal -ki, the UPOS tag of the former was changed to `Part`.

Field	UPOS			XPOS		
Change	**Adj ->Noun**	**CConj ->Part**	**Noun ->Propn**	**Verb ->Ptcp**	**Verb ->Vnoun**	**ANum ->Indef**
Count	1,595	1,025	968	2,459	1,664	1,622

Table 3
Most frequent changes targeting the UPOS and XPOS tags

Due to the shortcomings of automatic morphological tagging, some proper nouns were labeled as nouns. Re-annotation process targeted them as well: UPOS tags of 986 proper nouns were changed. In addition, XPOS tags of verbal nouns and participle forms were updated (see Table 3).

5. The BoAT Tool

The BoAT tool offered in Türk et al. [5] is a desktop application for manually annotating sentences parsed by a dependency parser. In the scope of the current work, in addition to the reannotation of the BOUN Treebank, we enhanced the tool with additional functionalities. We will publish the tool as open source on GitLab accompanied with a user manual.

5.1. Changes

BoAT is a desktop application, written in Python and based on Qt. The Qt version has been incremented from 5 to 6. This resulted in a more modern-looking user interface (UI). With ample feedback from the annotators who had used the tool for the BOUN Treebank, some requested features and improvements have been surfaced.

Clutter: Annotation table's columns were being shown or hidden by checkboxes above the table. These were removed and a textbox beside the other buttons has been added to replace them. This textbox serves exactly the same purpose while taking less space.

Autocompletion: Annotators use the tool for long hours at a time. Thus after a while, mistakes tend to occur. Another requested feature, autocompletion of the table fields, aims to prevent such mistakes. Many fields of the table have predetermined sets of values they can take. By not allowing values outside these sets and having a shorthand writing system whereby an annotator enters only the start of a value and it gets filled automatically, we implemented this much requested feature.

Saving as CoNLL-U: Another change was regarding saving of the *CoNLL-U* documents. The initial tool saved every edit automatically, yet treebanks with thousands of sentences tend to take time to save. This mishap seemed to slow our annotators. We added a save button instead of the autosave feature. Currently, a user uses the save button to edit the actual *CoNLL-U* file.

Shortcuts: The initial tool already had shortcut support. Going with the focus-oriented approach, all the new tasks have keyboard shortcuts associated with them as well. This alleviates the need to use a mouse while annotating via keyboard.

Dependency graphs: The vertical dependency graph in the initial version was replaced by the horizontal dependency graph of spaCy [22], displaCy, due to the spatial concerns.

Validation: The validation script for annotations has been upgraded to the latest version written by the UD framework, which has much more detailed explanations for why a specific annotation is invalid.

6. Parser Performance

Within the scope of this study, an NLP task was conducted. BiLSTM-based biaffine dependency parser proposed by Dozat and Manning [23] was trained using the updated BOUN Treebank. The train set contained 7.803 sentences, development set contained 982 sentences and test set contained 979 sentences. Considering the average arc length and average token count (see Table 4 for details) of each set, BOUN Treebank offers a well-balanced data set.

	Train	**Development**	**Test**	**Entire Data**
Average Arc Length	2.91	2.88	2.82	2.90
Average Token Count	12.83	12.42	12.36	12.74
Number of Sentences	7,803	982	979	9,761

Table 4
Sizes and specifications of train, development and test data sets

The previous version of the BOUN Treebank [5] yielded 77.36 unlabeled attachment score (UAS) and 70.37 labeled attachment score (LAS). After being trained on the new dataset, UAS is increased by 0.59 points to reach 77.96 while LAS showed a 0.10 decrease by dropping to 70.26 points. After the re-annotation process, several faulty or disputed dependency relations were fixed in the data, hence a rise in UAS was observed.

In order to better account for the particularities of Turkish morphosyntax, four new dependency labels were introduced: `dep:der` for adjectivizer -ki, `obl:tmod` for obliques that offer temporal information regarding the predicate, `advmod:emph` for dA clitics, and `compound:lvc` for light verb constructions with ol- copula. Moreover, new use cases for `cop` dependency label were offered to represent different functions of the ol- copula. These changes in the annotation framework added four new dependency types, thus the total class number was increased.

A sum of 1,032 `dep:der`, 894 `obl:tmod`, 1,860 `advmod:emph`, and 1,545 `compound:lvc` tags were added. Newly introduced tags and classes introduced added complexity, hence they might be the reason behind the slight decrease in the LAS results.

The main annotation changes made as part of this study were focused on morphology yet BiLSTM-based biaffine dependency parser proposed by Dozat and Manning [23] doesn't refer to the morphological information. In fact, it almost completely ignores the morphology features while parsing. Hence the significance of the improvements offered by this study can be better gauged by a morphology-aware parser or a morphological analysis based downstream task.

7. Conclusion and Further Research

The UD framework highly emphasizes syntactic relations and aims to offer a universal foundation to represent typologically different languages in a uniform way. While doing so, certain particularities of these languages tend to get lost in the annotation process in an attempt to abide by the UD convention. The aim of this study is to offer linguistically sound solutions to illustrate syntactically relevant morphological features of Turkish such as null morpheme realizations and derivational processes without diverging significantly from the UD framework.

In order to test the morphological capabilities of the re-annotated BOUN Treebank, a parser that refers to morphological information can be implemented in further research.

Acknowledgments

This work was supported by Boğaziçi University Research Fund Grant Number 16909. TUBAGE-BIP Award of the Turkish Science Academy (to A.O.) is gratefully acknowledged.

References

[1] L. Tesnière, Eléments de syntaxe structurale, paris: Klinck sieck, German translation:(1980) Grundzüge der strukturalen Syntax Stuttgart: Klett-Cotta (1959).

[2] K. Foth, A. Köhn, N. Beuck, W. Menzel, Because size does matter: The hamburg dependency treebank (2014).

[3] V. Vincze, K. I. Simkó, Z. Szántó, R. Farkas, Universal dependencies and morphology for hungarian-and on the price of universality, Association for Computational Linguistics, 2017.

[4] T. Bedir, K. Şahin, O. Güngör, S. Uskudarli, A. Özgür, T. Güngör, B. Ö. Başaran, Overcoming the challenges in morphological annotation of turkish in universal dependencies framework, in: Proceedings of The Joint 15th Linguistic Annotation Workshop (LAW) and 3rd Designing Meaning Representations (DMR) Workshop, 2021, pp. 112–122.

[5] U. Türk, F. Atmaca, Ş. B. Özateş, G. Berk, S. T. Bedir, A. Köksal, B. Ö. Başaran, T. Güngör, A. Özgür, Resources for turkish dependency parsing: Introducing the boun treebank and the boat annotation tool, Language Resources and Evaluation (2021) 1–49.

[6] N. B. Atalay, K. Oflazer, B. Say, The annotation process in the turkish treebank, in: Proceedings of 4th International Workshop on Linguistically Interpreted Corpora (LINC-03) at EACL 2003, 2003.

[7] G. Eryiğit, T. PAMAY, Itu validation set for metu-sabancı turkish treebank, Türkiye Bilişim Vakfı Bilgisayar Bilimleri ve Mühendisliği Dergisi 7 (2007) 31–37.

[8] U. Sulubacak, M. Gökırmak, F. Tyers, Ç. Çöltekin, J. Nivre, G. Eryiğit, et al., Universal dependencies for turkish, in: Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers, The Association for Computational Linguistics, 2016.

[9] C. Cöltekin, A grammar-book treebank of turkish, in: Proceedings of the 14th workshop on Treebanks and Linguistic Theories (TLT 14), 2015, pp. 35–49.

[10] A. Göksel, C. Kerslake, Turkish: A comprehensive grammar, Routledge, 2004.
[11] U. Sulubacak, G. Eryiğit, Implementing universal dependency, morphology, and multiword expression annotation standards for turkish language processing, Turkish Journal of Electrical Engineering & Computer Sciences 26 (2018) 1662–1672.
[12] T. Pamay, U. Sulubacak, D. Torunoğlu-Selamet, G. Eryiğit, The annotation process of the itu web treebank, in: Proceedings of the 9th Linguistic Annotation Workshop, 2015, pp. 95–101.
[13] C. T. Hemphill, J. J. Godfrey, G. R. Doddington, The atis spoken language systems pilot corpus, in: Speech and Natural Language: Proceedings of a Workshop Held at Hidden Valley, Pennsylvania, June 24-27, 1990, 1990.
[14] A. Taylor, M. Marcus, B. Santorini, The penn treebank: an overview, Treebanks (2003) 5–22.
[15] B. Marşan, N. Kara, M. Özçelik, B. N. Arıcan, N. Cesur, A. Kuzgun, E. Sanıyar, O. Kuyrukçu, O. T. Yıldız, Building the turkish framenet, in: Proceedings of the 11th Global Wordnet Conference, 2021, pp. 118–125.
[16] Y. Aksan, M. Aksan, A. Koltuksuz, T. Sezer, Ü. Mersinli, U. U. Demirhan, H. Yılmazer, G. Atasoy, S. Öz, İ. Yıldız, et al., Construction of the turkish national corpus (tnc), in: Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12), 2012, pp. 3223–3227.
[17] J. Kanerva, F. Ginter, N. Miekka, A. Leino, T. Salakoski, Turku neural parser pipeline: An end-to-end system for the conll 2018 shared task, in: Proceedings of the CoNLL 2018 Shared Task: Multilingual parsing from raw text to universal dependencies, 2018, pp. 133–142.
[18] J. Hankamer, Why there are two–ki's in turkish, Current research in Turkish linguistics, Eastern Mediterranean University (2004).
[19] A. Zeldes, M. Abrams, The coptic universal dependency treebank, in: Proceedings of the Second Workshop on Universal Dependencies (UDW 2018), 2018, pp. 192–201.
[20] V. Ravishankar, A universal dependencies treebank for marathi, in: Proceedings of the 16th International Workshop on Treebanks and Linguistic Theories, 2017, pp. 190–200.
[21] Y. Marton, N. Habash, O. Rambow, Improving arabic dependency parsing with form-based and functional morphological features, in: Proceedings of the 49th annual meeting of the association for computational linguistics: Human language technologies, 2011, pp. 1586–1596.
[22] Explosion, spaCy, 2022. URL: https://spacy.io/, [Online; last accessed 27 March 2022].
[23] T. Dozat, P. Qi, C. D. Manning, Stanford's graph-based neural dependency parser at the conll 2017 shared task, in: Proceedings of the CoNLL 2017 shared task: Multilingual parsing from raw text to universal dependencies, 2017, pp. 20–30.

A. A comparison of previous and updated annotation schemes

Token	Form	Lemma	UPOS	XPOS	Features	Head	Deprel	MISC
1	başındaki	başındaki	ADJ	Adj		2	amod	
3	şapkası	şapka	NOUN		Case=Nom|Number=Sing|Number[psor]=Sing|Person=3|Person[psor]=3	0	root	

Table 5
Previous annotation scheme for “başındaki şapkası" (*“the hat on his/her head”*)

Token	Form	Lemma	UPOS	XPOS	Features	Head	Deprel	MISC
1-2	başındaki	başındaki						
1	başında	baş	NOUN		Case=Loc|Number=Sing|Number[psor]=Sing|Person=3|Person[psor]=3	3	nmod	
2	ki	ki	PART	Attr		1	dep:der	
3	şapkası	şapka	NOUN		Case=Nom|Number=Sing|Number[psor]=Sing|Person=3|Person[psor]=3	0	root	

Table 6
Updated annotation scheme for “başındaki şapkası" (*“the hat on his/her head”*)

Token	Form	Lemma	UPOS	XPOS	Features	Head	Deprel	MISC
1	seninki	seninki	NOUN		Case=Nom|Number=Sing	0	root	

Table 7
Previous annotation scheme for “seninki" (*“that of yours”*)

Token	Form	Lemma	UPOS	XPOS	Features	Head	Deprel	MISC
1-2	seninki	seninki						
1	senin	sen	PRON	PERS	Case=Gen|Number=Sing|Person=2|PronType=Prs	2	nmod:poss	
2	ki	ki	PRON	Partic	Case=Nom|Number=Sing	0	root	

Table 8
Updated annotation scheme for “seninki" (*“that of yours”*)