Proceedings of the 12th Conference on Language Resources and Evaluation (LREC 2020), pages 5156–5163
Marseille, 11–16 May 2020

A Gold Standard Dependency Treebank for Turkish

Tolga Kayadelen†, Adnan Öztürel†, Bernd Bohnet‡
Google
†London, UK
‡Amsterdam, Netherlands
{tkayadelen, ozturel, bohnetbd}@google.com

Abstract

We introduce TWT; a new treebank for Turkish which consists of web and Wikipedia sentences that are annotated for segmentation, morphology, part-of-speech and dependency relations. To date, it is the largest publicly available human-annotated morpho-syntactic Turkish treebank in terms of the annotated word count. It is also the first large Turkish dependency treebank that has a dedicated Wikipedia section. We present the tagsets and the methodology that are used in annotating the treebank and also the results of the baseline experiments on Turkish dependency parsing with this treebank.

Keywords: treebank, dependency parsing, Turkish

1. Introduction

Dependency parsing is an important building block in improving the performance of downstream NLP tasks such as semantic role labeling (Marcheggiani et al., 2017), relation extraction (Zhang et al., 2018) or machine translation (Chen et al., 2017). Treebanks are invaluable resources for developing and training accurate dependency parsers in supervised settings and more and more research has been invested in developing high quality treebanks for various languages over the years.

Compared to other well studied languages in NLP, the publicly available treebanks for Turkish has remained meagre in size and domain variation until very recently. The METU Sabancı Treebank (MST) (Atalay et al., 2003; Oflazer et al., 2003) had been the only treebank for Turkish for over a decade. Even though the cumulative size of morpho-syntactically annotated datasets has increased with the release of new language resources such as ITU Web Treebank (IWT), (Pamay et al., 2015) and Parallel Universal Dependencies Treebank (PUD) (Zeman et al., 2018), Turkish continues to stay relatively under resourced as the number of annotated sentences in all of these above-mentioned language resources amount to approximately 12K unique sentences in total.

Together with this paper we open-source the Turkish Web Treebank (TWT)[1], which is freely available, large in size, and also involves annotated sentences from previously under represented domains such as Wikipedia. We present our part-of-speech (PoS) and morphological feature tagsets and dependency label set, our annotation methodology, inter-annotator agreement scores, and discuss some of the highlights of our annotation decisions – especially the cases where they diverge from the previously implemented annotation practices of the Turkish treebanking literature. We also show that the tagset used in the annotation of TWT is the most elaborate one that has been developed so far for Turkish. It is adequate in representing the complex derivational morphology of Turkish throughout part-of-speech and dependency labeling. We illustrate that this level of representational granularity does not come at the expense of parsing performance.

[1] https://github.com/google-research-datasets/turkish-treebanks/

2. A Brief History of Turkish Treebanking

Turkish treebanking dates back to MST (Atalay et al., 2003; Oflazer et al., 2003), which was created by sampling 5,635 sentences from METU Turkish Corpus (Say et al., 2002). It contains examples that belong to 10 genres (such as News, Fiction, Research Papers, Memoirs, Newspaper Columns and Essays, etc.). This treebank and the precursor research on modeling Turkish morphology (Oflazer, 1994; Oflazer et al., 1994) set the standards for many annotation principles for Turkish. Especially innovative in this work was the introduction of Inflectional Group (IG) formalism to deal with the productive derivational morphology of the language (see Section 3.2.1. for a further discussion on IGs).

Recently, Sulubacak et al. (2016a) have critically reviewed the MST and re-annotated its dependency layer, proposing new labeling practices and fixing some inconsistent annotations that they report from the original MST. Notable highlights from this research are the unification of the treatment of adjunct dependencies under a uniform tag named MODIFIER, introduction of a new dependency label called ARGUMENT to distinguish objects of postposition from the object of the main predicate, and an enriched scheme for handling multiword expressions (MWEs). The latter change introduced a significant amount of 17 dependency labels for annotating MWEs at dependency level. They report an improvement in dependency parsing metrics where LAS increased from 65.9% to 75.3% and UAS from 76.0% to 83.7%, when tested with the MALT parser (Nivre et al., 2007) using the original vs. re-annotated version of the treebank.

The updated dependency treebank, named IMST, has been the standard resource for training Turkish dependency parsers after its creation, and lately there have been efforts to make it compliant with the Universal Dependencies (UD) v2.0 standards. IMST-UD (Sulubacak et al., 2016b; Sulubacak and Eryiğit, 2018) is the outcome of this effort,

where the IMST treebank is automatically mapped to UD annotation scheme. Some significant changes were also applied to the data during this mapping. For example, at the segmentation and morphology level, some derivational morphemes that were deemed not sufficiently productive were no longer segmented but merged with their stem. Similarly, most of the deadjectival, denominal, and deadverbial verb suffixes which were previously tokenized and PoS tagged were merged with the stem and the derivations that they represent were encoded in a new morphological feature *VerbForm*. At the dependency relations level, a finer grained handling of nominal and verbal modification was introduced to comply with UD standards. Different syntactic types of verbal and nominal modification, which were collapsed in the IMST into the generic MODIFIER label, were now represented by a set of more linguistically expressive and transparent dependency labels such as *Amod*, *Advmod*, *Advcl*, *Nmod*, depending on the syntactic category of the modifier.

Both IMST/IMST-UD and their ancestor lacked data from user generated texts such as the web. Pamay et al. (2015) developed a new treebank that specifically involved sentences from a variety of web domains. During the annotation of this data some additional practices were implemented to deal with non-canonical language usages that are abundant in the web. To handle frequently occurring cases like spelling mistakes, abbreviated writing, emphatic character repetition (e.g. '*pleaseeeee*') or wrong capitalization, a manual normalization process was introduced to the annotation pipeline and those uses were replaced with their normalized forms by the annotators. Furthermore, some additional part-of-speech tags were introduced to handle emoticons, formally accepted abbreviations, URLs, e-mail addresses, and hashtags. Ongoing efforts to create a UD complaint version for this treebank called IWT-UD has also been reported recently (Sulubacak and Eryiğit, 2018).

Finally, another treebank that is available for Turkish is the Turkish Parallel Universal Dependencies Treebank (TR-PUD) (Zeman et al., 2018). The creation of this treebank was a collaborative effort between DFKI translators, UD community and Google linguistic teams. Data was translated from English to Turkish and passed to Google linguists for annotation. Any conflicts between the annotation standards of UD and Google annotations were resolved by converting the annotations to the effective UD standards of the time automatically. As this data was meant to be used for evaluation in CoNLL 2017 shared task, it is smaller in size compared to its sisters, comprising a total of 1,000 sentences from Wikipedia and news domains. Therefore, it is not very suitable for use as a standalone language resource to train a model.[2]

3. Turkish Web Treebank

3.1. Treebank Statistics

The Turkish Web Treebank (TWT) consists of 4,851 sentences sampled from the Turkish web and Wikipedia pages where each sentence is manually annotated for segmentation, morphology, part-of-speech and dependency relations. We provide some basic comparative statistics about TWT in Table 1.

	IMST	**IMST-UD**	**IWT**	**IWT-UD**	**TWT**
Sentences	5,635	5,635	5,009	5,009	4,851
Words	56,422	58,085	43,191	44,463	66,466
Tokens	63,066	58,146	47,226	44,545	81,370
Unique PoS (Coarse)	11	14	11	15	13
Unique Morph. Features	47	74	46	64	51
Unique Dep. Relations	33	29	32	28	44

Table 1: Comparison of TWT to other Turkish treebanks.

Even though TWT has fewer number of sentences than its predecessors, it is the largest Turkish treebank in terms of the number of orthographic words and tokens. The higher number of words is due to the fact that it contains a good amount of Wikipedia sentences, which are customarily longer than the sentences in other web domains. It has an even higher number of tokens because of the very fine grained segmentation model that it employs (see Section 3.2.1.).

In terms of *Unique Dependency Relations*, TWT distinguishes between a wider variety of syntactic relations than its predecessors. The *Unique Morphological Features* reported in the Table 1 only refer to the inflectional features, and in that respect TWT is similar to IMST and IWT rather than their UD counterparts. However, we will see in the next section that in terms of representing morphological derivation, TWT is the most detailed Turkish treebank to date, as it segments and marks a generous number of 62 derivational morphemes with specific tags. Finally, TWT also employs an elaborate fine part-of-speech tagset which will be illustrated in Section 3.2.2..

The treebank is divided into two sections as TWT-Wiki and TWT-Web, where the TWT-Web section involves sentences from non-Wikipedia domains such as Forums, Blogs, How-to contents, Guides and Reviews. Some basic statistics of the two sections are provided in Table 2. The average number of words per sentence is marginally higher in TWT-Wiki section in comparison to TWT-Web section. Given that Wikipedia has been an under represented domain in Turkish treebanks so far, we believe that this property of TWT will be particularly important for researchers who want to get an idea of how their models perform on relatively longer Turkish sentences.

No specific sampling technique was used for Wikipedia sentences except that we sampled at most one sentence from each Wikipedia page. For sampling sentences from other web pages, we tried to establish a balanced corpus that is representative of the five domains mentioned above.

[2]But see Türk et al. (2019) for some experiments and their results.

	Sentences	Words	Tokens	Word Avg.	Token Avg.
TWT Wiki	2,310	39,947	48,948	17.2	21.1
TWT Web	2,541	26,519	32,422	10.4	12.7

Table 2: TWT-Wiki and TWT-Web.

3.2. Tagsets and the Annotation Scheme

3.2.1. Segmentation and Morphology

Turkish poses a challenge for morphological processing due to the high number of productive derivational morphemes. The challenge comes from the fact that a word root can successively derive into different syntactic categories by affixation of derivational morphemes, which results in cases where sub-spans of the word engage in different syntactic relations with different words. An example of this is presented in Fig. 1 with corresponding dependency tree. Particularly interesting is the verb *bit-* ("*to end*") and the derivation it undergoes, which affects the dependency relations it partakes in. Lexically intransitive, the verb is first transitivized with the causative morpheme *-Hr*, thanks to which it can take the argument *öğrenimi* as an object. It is then nominalized by a further derivation step and the resulting noun itself serves as a modifier of another category.

(a)

Üst	öğrenimi		
üst	öğrenim+i		
high	education+Acc		
bitirmenin	sağladığı	haklar	
bit-ir-me+nin	sağla-dığ+ı	hak+lar	
end-Cau-Nonf+Gen	provide-PastPart+P3sg	right+A3pl	

(b)

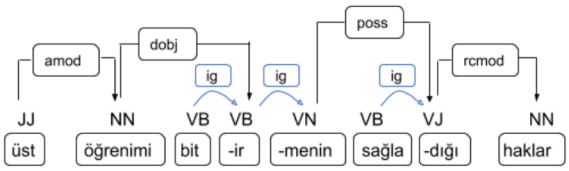

Figure 1: Dependency tree of inflectional group tokenized sentence "*üst öğrenimi bitirmenin sağladığı haklar...*".

The tension between word internal derivational processes and sentence level dependency relations has traditionally motivated researchers to represent Turkish words in terms of what are called *Inflectional Groups* (IGs). IGs tokenize words into their morphological segments based on its derivational boundaries. In Fig. 1, for example, the word *bitirmenin* is segmented into three tokens: *bit-*, *-ir*, and *-menin* to account for the successive derivation it undergoes as VB → VB → VN. Each IG is treated as a token with its own part-of-speech tag and morphological features (which were not shown in Fig. 1 for clarity). Morphemic tokens of the same orthographic word are linked to each other at the dependency layer with a special label named "*ig*".

The benefits of using IGs to segment words into their subtokens are two-fold. First, it provides a much more expressive and linguistically sound analysis of syntactic relations in Turkish that also benefits learning. For example, in a framework where words were not segmented according to IGs, a sentence like in Fig. 1 would have to link the whole word "*bitirmenin*_NOUN" to the word "*öğrenimi*_NOUN" with a "*dobj*" dependency relation, creating confusion for a parser in disambiguating dependency labels. Secondly, it reduces data sparsity in cascaded NLP architectures where dependency parsing uses morphological processing units as features. As shown in Eryiğit and Oflazer (2006) which evaluates parsing performance using word-based and IG-based versions of the MST treebank, having IG-based segmentation significantly improves parsing performance in a pipelined learning architecture.

The appropriate level of granularity within which to segment derivational morphemes in a treebank is still an open research topic in Turkish parsing. The baseline assumption is that productive morphemes should be segmented; however, different studies might have differing interpretations of what makes a morpheme productive enough to represent it as a separate token. For example, Sulubacak et al. (2016b) note that while creating the IMST-UD treebank, they have considered derivations like -(H)CH (+Agt, which derives agentive nouns from verb stems) and -lAn (+Acq, which derives verbs from nouns or adjectives with the semantics "*acquire X*") to be not sufficiently productive and therefore they kept them merged with their stems. The distribution of these derivations in our treebank, however, indicates that they can be treated as productive: -(H)CH is the 14th most frequently occurring suffix among the 62 suffixes we tokenize, while -lAn ranks as 16th. Similarly, while Sulubacak et al. (2016b) does not segment the denominal/deadjectival verb suffix -lA (+Make), research shows that it is the most productive verb deriving suffix in modern Turkish (Nakipoğlu and Üntak, 2008).

The segmentation model employed in TWT is the Turkish morphology model from Öztürel et al. (2019). We tag all the derivational morphemes that are segmented by the finite-state transducer analyzer presented in that study. Comparatively, this makes TWT the most expressive treebank in terms of the number of derivations it represents.[3]

Table 3 presents 15 most frequent derivational morphemes in our treebank, together with their counts and examples. The total count of derivational morphemes that are segmented in TWT is 14,907. Among those, some morphemes occur very infrequently in the data. For example, the affix +Doct (which we use to tag noun-to-noun derivation such as *sosyal-izm* "*socialism*"), occurs only 13 times. However, while making the decision of whether we should segment and tag a morpheme, we did not restrict ourselves only with the properties of the data we were working on but kept in mind that based on the domain from which the data is sourced, the productivity of a morpheme can show considerable variation. For example, one can easily imagine the +Doct morpheme occuring much more frequently in a philosophical text and in such a case having a morphological model that can annotate this morpheme can be useful. Therefore, one of our motivations was to provide the most detailed morphological representation possible to

[3]For an exhaustive list refer to: `https://github.com/google-research/turkish-morphology/blob/master/src/analyzer/morphotactics/README.md#derivational-morphemes`

ID	FORM	LEMMA	UPOS	XPOS	FEATS	HEAD	DEPREL	MISC
1	Üst	üst	ADJ	JJ	Proper=False	2	amod	_
2	öğrenimi	öğrenim	NOUN	NN	PersonNumber=A3sg|Possessive=Pnon|Case=Acc|Proper=False	4	dobj	_
3	bit	bit	VERB	VB	Proper=False	4	ig	SpaceAfter=No
4	ir	_	VERB	VB	Derivation=Cau|Polarity=Pos|Proper=False	5	ig	SpaceAfter=No
5	menin	_	NOUN	VN	Derivation=Nonf|PersonNumber=A3sg|Possessive=Pnon|Case=Gen|Proper=False	7	poss	_
6	sağla	sağla	VERB	VB	Polarity=Pos|Proper=False	7	ig	SpaceAfter=No
7	dığı	_	ADJ	VJ	Derivation=PastPart|Possessive=P3sg|Proper=False	8	rcmod	_
8	haklar	hak	NOUN	NN	PersonNumber=A3pl|Possessive=Pnon|Case=Bare|Proper=False	0	root	SpaceAfter=No
9	...	...	PUNCT	.	Proper=False	8	p	_

Figure 2: Annotation of the sentence "*üst öğrenimi bitirmenin sağladığı haklar...*" from Fig. 1 in CoNLL-U format. Note that we use the original UPOS and XPOS fields to respectively specify the coarse and fine part-of-speech tags.

Tag	Meta-Morpheme	Count	Example
Pass	-Hl, -Hn	2,043	yap-ıl+dı
Nonf	-mA, -YHş	1,891	konuş-ma, bak-ış
PresPart	-YAn	1,279	kazan-an
With	-lH, -HlH	1,065	uyku-lu
Cau	-DHr, -Hr, -Ht, -t	883	yap-tır
Ness	-lHk	880	insan-lık
Make	-lA	768	işaret-le
Able	-YAbil, -YA	620	gel-ebil-ir
PastNom	-DHk	617	yap-tık+larım
Ger	-YArAk, -DAn	583	koş-arak, koş+ma-dan
Rel	-ki	503	okulda-ki
Inf	-mAk	500	koş-mak
PastPart	-DHk	356	yap-tığ+ım
Agt	-CH	276	koş-ucu
After	-YHp	206	gel-ip

Table 3: Frequency statistics of common derivational morphological features in TWT.

the consumers of our treebank and morphological models. This enables the opportunity to represent morpho-syntax on data from different domains and train models using them. Eventually, we let the users of TWT decide which affixes to keep segmented while building cascaded language understanding models, since any affix segmented in the treebank can be easily merged back to its stem with a simple preprocessing step before training morphological analyzers or dependency parsers using TWT.

Fig. 2 illustrates the annotation of example from Fig. 1 in our treebank. Morphological derivations are explicitly marked by the feature *Derivation*, which is different from the previous treebanks where they were annotated as fine part-of-speech tags. This way users can easily trace the derivations that exist in TWT or in a specific sentence of it.

3.2.2. Part-of-Speech Tags

Table 4 shows the full tagset that is used in annotating the part-of-speech in TWT as well as their definition and distribution in the data.

It might be useful to briefly compare the part-of-speech tagset here with the IMST-UD treebank to get an understanding of how different it is from the current UD annotation scheme. Our coarse tags are mostly aligned with the UD v2 standard. The differences are the slightly different naming of coordinating conjunctions and particles (CONJ instead of CCONJ[4], PRT instead of PART), tagging proper nouns as subtypes of the NOUN rather than treating it as a stand-alone category, and the extra tag called ONOM. On the other hand, the list of fine tags are more comprehensive than the ones currently used in UD treebanks for Turkish. There are 45 fine tags in TWT compared to the 34 tags in IMST-UD. Some of the main differences are summarized below.

TWT has a more detailed use of the X category, divided into fine tags based on different types of non canonical usages most frequently found in web data. The FW fine tag is used when there are foreign words in an otherwise Turkish sentence. Note that this tag does not apply to named entities that have a non-Turkish name, which are annotated as proper nouns as usual. The GW is used for mistokenized or wrongly segmented instances found commonly on the web. Emoticons and other symbols are handled by the SYM tag. URLs and e-mail addresses are tagged as types of the NOUN category instead of X, as these are named entities. Similarly, TWT divides adverbs into subgroups depending on whether they are lexical adverbs (RB), converbs that are derived from verbs (CRB), or wh-words that function as an adverb in the context of the sentence. Having a CRB tag helps us deal with syntactic intricacies of Turkish where the base verb establishes a predicate-argument relationship with an object before deriving into an adverb and modifying another verb. There is an extended classification of the PRT tag, which involves annotating the question particle (-mH), the negation particle (*değil*), the clitic (dA), and the connective (*eğer*) with separate fine tags. The EP fine tag is preserved for cases where users add particles to the sentence to achieve an emphatic effect (e.g. "*Yaa*" in the sentence "*Yaa, çok şaşırdım (Ohh, I'm very surprised)*" would be annotated as EP).

Finally, some other minor differences exist in terms of the

[4]In fact, CONJ was also the standard tag coordinating conjunctions in UD v1.

Coarse Tag	Fine Tag	Count	%	Definition
ADJ	JJ	4,651	5.72	adjective
	VJ	1,977	2.43	deverbal adjective
ADP	IN	2,762	3.39	adposition
ADV	CRB	996	1.12	converb
	RB	1,639	2.01	adverb
	WRB	92	0.11	interrogative adverb (wh-adverb)
AFFIX	PFX	3	0.00	prefix
CONJ	CC	2,674	3.29	coordinating conjunction
DET	DT	2,264	2.78	determiner
	PDT	106	0.13	predeterminer
	WDT	51	0.06	negation particle
NOUN	ADD	39	0.05	electronic addresses and URLs
	NN	24,920	30.63	noun
	NNP	5,642	6.93	proper noun
	VN	3,216	3.95	deverbal noun
NUM	CD	1,534	1.89	cardinal
ONOM	DUP	13	0.02	onomatopoeic
PRON	PRD	113	0.16	demonstrative pronoun
	PRF	20	0.02	derived pronominal
	PRI	564	0.69	indefinite pronoun
	PRP	281	0.35	personal pronoun
	PRP$	105	0.13	possessive pronoun
	PRR	109	0.13	reflexive pronoun
	WP	93	0.11	interrogative pronoun (wh-pronoun)
PRT	EP	20	0.02	emphatic particle
	OP	42	0.05	connective particle
	RPC	791	0.97	clitic
	RPNEG	45	0.06	negation particle
	RPQ	107	0.13	question particle
PUNCT	”	313	0.38	closing quotation mark or similar
	(	348	0.43	left bracket punctuation
	)	348	0.43	right bracket punctuation
	,	2,340	2.88	comma or similar
	-	394	0.48	hyphens, dashes or similar
	.	4,731	5.81	sentence final punctuation
	:	267	0.33	colon and semi-colon
	“	310	0.38	opening quotation mark or similar
VERB	VB	15,424	18.96	verb
	NOMP	1,571	1.93	nominal predicate
X	FW	7	0.01	foreign word
	GW	181	0.22	goes with
	LS	24	0.03	list
	NFP	51	0.06	non-final punctuation
	SYM	104	0.13	symbols, emoticons or similar
	UH	68	0.08	interjection
	XX	0	0.00	total garbage

Table 4: Part-of-speech tagset of TWT.

treatment of verbs and punctuation. TWT differentiates between lexical verbs and predicates that are derived from adjectives or nouns with the copula morpheme, marking the latter as NOMP and the former as VB. As for punctuation, rather than using the PUNCT coarse tag in an all-encompassing manner, we annotate different sub-types of punctuation with dedicated fine tags.

Given that efforts for determining the most useful fine tagset for Turkish is still ongoing, we believe the tagset developed in TWT can be an important contribution to Turkish treebanking studies.

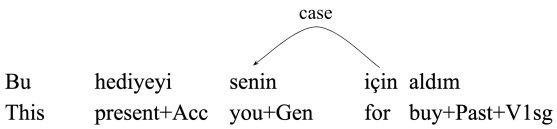

Figure 3: Marking postpositions with “*case*” dependency relation using UD Turkish annotation scheme: an example on the sentence “*I bought this present for you.*”.

3.2.3. Dependency Labels

In terms of dependency labels, some notable differences from the Turkish UD annotation scheme are the treatment of postpositional modifiers (“*prep*”, “*pobj*”, and “*pcomp*”), and marking of indirect objects (“*iobj*”).

As illustrated in Fig. 3 current UD practice for annotating postpositions in Turkish is to mark them with the dependency label “*case*”. As well as marking postpositional objects, the case label also relates some derivational suffixes to their stems within the UD scheme. We find this treatment problematic in terms of the linguistic properties of the language. First, in Turkish case is an inflectional feature that marks grammatical roles a verb assigns to its arguments rather than a derivational one. Second, the relation between a postposition and its object is a predicate-argument relation that we believe should be better handled by a syntactic label. We therefore prefer to represent the relation between a postposition and its object as “*pobj*”. The postpositions themselves are linked to their head with the “*prep*” label and we also distinguish clausal vs. nominal objects of postpositions via the two labels “*pcomp*” vs. “*pobj*”.

Another difference from the current UD dependency labeling scheme is the tag “*iobj*”. We use this tag to mark the re-

Label	Count	%	Definition
ROOT	4,851	5.96	root of the sentence
acomp	540	0.66	adjectival complement
advcl	1,587	1.95	adverbial clause
advmod	2,036	2.50	adverbial modifier
amod	4,059	4.99	adjectival modifier of NP
appos	340	0.42	appositional modifier of NP
attr	778	0.96	attribute dependent of a copular verb
aux	111	0.14	auxiliary verb
cc	2,212	2.72	coordinating conjunction
ccomp	120	0.15	clausal complement of a verb or adjective
clas	79	0.10	classifier
conj	3,317	4.08	conjunct
csubj	598	0.73	clausal subject
det	2,310	2.84	determiner
discourse	124	0.15	interjections and other discourse elements
dislocated	3	0.00	dislocated elements
dobj	3,570	4.39	direct object
goeswith	183	0.22	parts of a word that were mistokenized
ig	14,904	18.32	inflectional group
iobj	420	0.52	indirect object
list	8	0.01	list for chains of comparable items
mark	31	0.04	complementizer (words introducing a finite subordinate clause)
mwe	756	0.93	multiword expression
narg	311	0.38	argument of a nominal
neg	38	0.05	negation
nn	7,642	9.39	nominal modifier
npadvmod	3,721	4.57	noun phrase used as an adverbial modifier of a verb
nsubj	3,835	4.71	nominal subject
num	659	0.81	numeric modifier of a noun
number	36	0.04	element of compound number
p	9,185	11.29	punctuation
parataxis	531	0.65	parataxis
pcomp	681	0.84	clausal complement of a postposition
pobj	2,065	2.54	object of postposition
poss	2,390	2.94	possessive modifier
preconj	46	0.06	preconjunct
predet	110	0.14	predeterminer
prep	2,640	3.24	postposition
prt	1,007	1.24	particle
rcmod	1,886	2.32	relative clause modifier
remnant	181	0.22	ellipsis
tmod	706	0.87	temporal modifier
vocative	71	0.09	vocative
xcomp	692	0.85	open clausal complement

Figure 4: Dependency label set of TWT.

lation between verbs and indirect objects for verbs that subcategorize for them as part of their lexical argument structure. We believe this practice can be helpful to induce fine grained argument structure templates from the data.[5]

3.2.4. The Annotation Procedure

The treebank was annotated with a team of 7 linguists, where 6 of them were annotators and the 7th was the reviewer/arbitrator. 6 annotators were divided into 3 groups and each group was responsible for different portions of the data. 2 linguists in every group worked on the annotation of the same set of sentences, so the whole treebank is 2-way annotated for all layers. The annotations were then diffed and any disagreements were sent to the reviewer for arbitration.

Batches	Annotators
Batch A	G1, G2
Batch B	G1, G3
Batch C	G2, G3

Table 5: Distribution of annotators for IAA evaluation.

To ensure that the annotators were aligned in their decisions, after the initial training period an inter-annotator agreement (IAA) evaluation was done. To evaluate IAA, we randomly sampled 900 sentences from Wikipedia and created 3 batches each of which consisted of 300 sentences. The annotators were divided again into three groups (G1, G2, and G3), and each batch of the IAA data was annotated by 2 groups, a total of 4 annotators, as in Table 5. We then computed the agreement between the 4 annotators in each batch for morphology, part-of-speech tagging and dependency relation labels. Table 6 presents the IAA scores for all layers.

Layer	Agreement Metric	Batch A	Batch B	Batch C
Morphology	full token agreement	94.16	95.35	95.39
Part-of-Speech	PoS tag agreement	97.39	97.67	97.24
Dependency	unlabeled attachment	95.25	95.24	95.49
Dependency	labeled attachment	92.18	92.06	91.78

Table 6: Inter-annotator agreement scores.

The full token agreement at the morphology layer is an evaluation of the percentage of tokens that were annotated with identical segmentation and morphological features by all annotators. The IAA scores are above 95% for all metrics except for the labeled attachment metric, for which it is still above the 90% that we took as a threshold before starting annotation of the treebank. Overall, the high IAA score is an indication of the consistency of annotations in TWT, which is an important factor for producing high qual-

[5]For an explanation of the rest of the additions and differences between the Turkish UD labels and TWT refer to: `https://github.com/google-research-datasets/turkish-treebanks/README.md`

ity NLP models using it as training data.

4. Evaluation

We evaluate TWT in terms of tagging and dependency parsing accuracy. We used the Stanford Dependency Parser version 3 (Dozat et al., 2017; Qi et al., 2018) for part-of-speech, morphological tagging and dependency parsing. The Stanford Parser won the 2017 CoNLL Shared Task for Multilingual Parsing from Raw Text to Universal Dependencies and was widely adopted in the 2018 Shared Task. For part-of-speech and morphological tagging in the 2018 Shared Task, the Meta-BiLSTM Tagger (Bohnet et al., 2018) performed best. Therefore, we provide additional accuracy scores for this system. We have split the treebank by uniformly sampling from the genres in it using 80% for training, 10% for development data and 10% for the test set. We use the development set for early stopping and use the standard hyperparameters of the Stanford Parser. We use the Turkish embeddings as provided by the CoNLL 17 Shared Task (Zeman et al., 2017). For the Meta-BiLSTM tagger, we use the predefined settings as well, using 2 LSTM layers with 300 cells each.

	Stanford	**Stanford**
UAS	86.06	89.96
LAS	77.63	84.17
	Stanford	**Meta-BiLSTM**
UPOS	94.36	96.20
XPOS	89.89	94.45
Morphological features	85.94	94.74

Table 7: The accuracy on the test set using the Stanford Parser for dependency parsing, and using the Stanford system vs. Meta-BiLSTM tagger for part-of-speech tagging.

	Predicted PoS/morphology	**Gold PoS/morphology**
UAS	86.06	92.95
LAS	77.63	90.19

Table 8: Stanford Parser's dependency parsing performance on TWT using predicted part-of-speech/morphology features vs. gold part-of-speech/morphology features.

Table 7 shows how the state-of-the-art parser performs when trained and evaluated on TWT. The dependency parsing accuracies have been in both cases obtained via the Stanford Parser but using different taggers (Stanford vs. Meta-BiLSTM).

It is noteworthy about the results that they are significantly higher than the ones reported in the Shared Task by the same parser for Turkish, which were 93.86% for UPOS, 93.11% for XPOS, 69.62% for UAS and 62.79% for LAS (morphological tagging was not evaluated). Compared to parser performance on the Shared Task data, the parser generates marginally higher scores for UAS and LAS, and slightly better scores for UPOS on TWT. Although it is a matter of future analysis and research to understand the data properties that lead to better parser performance, we reason that it might be attributed to two points. First, we believe the high inter-annotator agreement within which the treebank was annotated contributes towards the high performance of the parser, as it guarantees consistency in the annotations. Second, the elaborate segmentation scheme that Öztürel et al. (2019) employed and this treebank adopted might have had a significant impact on data sparsity, facilitating learning for the parser.

It is also quite clear from Table 7 that high accuracy in part-of-speech and morphological feature tagging has a direct impact on parsing accuracy. When the Stanford tagger is replaced with the META-BiLSTM tagger, we not only get significant improvements in UPOS and XPOS (respectively from 94.26% to 96.20%, and from 89.89% to 94.45%) but also the parsing results further improve from 86.06% to 89.96% for UAS, and from 77.63% to 84.17% for LAS.

In Table 7, UAS and LAS were obtained by the dependency parser using predicted part-of-speech and morphological tags as features. We have also experimented with a set up where these two layers were kept as gold annotations, to see how much gold vs. predicted features and part-of-speech tags impact the Stanford Parser's dependency parsing performance for Turkish. Table 8 illustrates the results.

5. Conclusion and Future Work

This paper introduced TWT, a new Turkish treebank comprising web and Wikipedia sentences which we are making publicly available. The treebank consists of 4,851 sentences annotated for morpho-syntax with a very high inter-annotator agreement. We presented our tagsets and dependency labels and reported baseline part-of-speech/morphology tagging and dependency parsing scores with state-of-the-art parsers using TWT.

Given that there are only a few treebanks available for Turkish NLP research, we believe TWT will be an important contribution to the field and help further development of high quality language understanding models for the language. As future work, we plan to implement conversion tools that can map TWT annotations to Universal Dependencies v2 standards and create a UD compliant version of the treebank.

6. Acknowledgements

We would like to thank Ryan McDonald and Jan Botha for their feedback on the design of the treebank; Ji Ma and Slav Petrov for their feedback on a draft version of this paper; Oddur Kjartansson and Clara Rivera for helping with the release of the dataset; Savaş Çetin, Nihal Meriç Atilla, Eda Aydın Oktay, Faruk Büyüktekin, Hakan Keser, Hilal Yıldırım, İlmiye Karakimseli, Ozan Mahir Yıldırım and Cengiz Dikme for their work on data annotation.

7. Bibliographical References

Atalay, N. B., Oflazer, K., and Say, B. (2003). The annotation process in the Turkish treebank. In *Proceedings of 4th International Workshop on Linguistically Interpreted Corpora (LINC-03) at EACL 2003*.

Bohnet, B., McDonald, R., Simões, G., Andor, D., Pitler, E., and Maynez, J. (2018). Morphosyntactic tagging

with a meta-bilstm model over context sensitive token encodings. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2642–2652.

Chen, K., Wang, R., Utiyama, M., Liu, L., Tamura, A., Sumita, E., and Zhao, T. (2017). Neural machine translation with source dependency representation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2846–2852.

Dozat, T., Qi, P., and Manning, C. D. (2017). 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*, pages 20–30.

Eryiğit, G. and Oflazer, K. (2006). Statistical dependency parsing for Turkish. In *11th Conference of the European Chapter of the Association for Computational Linguistics*.

Marcheggiani, D., Frolov, A., and Titov, I. (2017). A simple and accurate syntax-agnostic neural model for dependency-based semantic role labeling. In *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017)*, pages 411–420, Vancouver, Canada, August. Association for Computational Linguistics.

Nakipoğlu, M. and Üntak, A. (2008). A complete verb lexicon of Turkish based on morphemic analysis. *Turkic Languages*, 12:221–80.

Nivre, J., Hall, J., Nilsson, J., Chanev, A., Eryiğit, G., Kübler, S., Marinov, S., and Marsi, E. (2007). Maltparser: A language-independent system for data-driven dependency parsing. *Natural Language Engineering*, 13(2):95–135.

Oflazer, K., Göçmen, E., and Bozşahin, C. (1994). An outline of Yurkish morphology. *Report to NATO Science Division SfS III (TU-LANGUAGE), Brussels*.

Oflazer, K., Say, B., Hakkani-Tür, D. Z., and Tür, G. (2003). Building a Turkish treebank. In *Treebanks*, pages 261–277. Springer.

Oflazer, K. (1994). Two-level description of Turkish morphology. *Literary and linguistic computing*, 9(2):137–148.

Öztürel, A., Kayadelen, T., and Demirşahin, I. (2019). A syntactically expressive morphological analyzer for Turkish. In *Proceedings of the 14th International Conference on Finite-State Methods and Natural Language Processing*, pages 65–75.

Pamay, T., Sulubacak, U., Torunoğlu-Selamet, D., and Eryiğit, G. (2015). The annotation process of the ITU web treebank. In *Proceedings of the 9th Linguistic Annotation Workshop*, pages 95–101.

Qi, P., Dozat, T., Zhang, Y., and Manning, C. D. (2018). Universal dependency parsing from scratch. In *Proceedings of the CoNLL 2018 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 160–170, Brussels, Belgium, October. Association for Computational Linguistics.

Say, B., Zeyrek, D., Oflazer, K., and Özge, U. (2002). Development of a corpus and a treebank for present-day written Turkish. In *Proceedings of the eleventh international conference of Turkish linguistics*, pages 183–192. Eastern Mediterranean University.

Sulubacak, U. and Eryiğit, G. (2018). Implementing universal dependency, morphology, and multiword expression annotation standards for Turkish language processing. *Turkish Journal of Electrical Engineering & Computer Sciences*, 26(3):1662–1672.

Sulubacak, U., Eryiğit, G., Pamay, T., et al. (2016a). IMST: A revisited Turkish dependency treebank. In *Proceedings of TurCLing 2016, the 1st International Conference on Turkic Computational Linguistics*. Ege University Press.

Sulubacak, U., Gökirmak, M., Tyers, F., Çöltekin, Ç., Nivre, J., and Eryiğit, G. (2016b). Universal dependencies for Turkish. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 3444–3454.

Türk, U., Atmaca, F., Özateş, Ş. B., Köksal, A., Basaran, B. O., Gungor, T., and Özgür, A. (2019). Turkish treebanking: Unifying and constructing efforts. In *Proceedings of the 13th Linguistic Annotation Workshop*, pages 166–177.

Zeman, D., Popel, M., Straka, M., Hajič, J., Nivre, J., Ginter, F., Luotolahti, J., Pyysalo, S., Petrov, S., Potthast, M., et al. (2017). CoNLL 2017 shared task: Multilingual parsing from raw text to universal dependencies. In *CoNLL 2017: The SIGNLL Conference on Computational Natural Language Learning*, pages 1–19. Association for Computational Linguistics.

Zeman, D., Hajič, J., Popel, M., Potthast, M., Straka, M., Ginter, F., Nivre, J., and Petrov, S. (2018). CoNLL 2018 shared task: multilingual parsing from raw text to universal dependencies. In *Proceedings of the CoNLL 2018 Shared Task: Multilingual Parsing from Raw Text to Universal Dependencies*, pages 1–21.

Zhang, Y., Qi, P., and Manning, C. D. (2018). Graph convolution over pruned dependency trees improves relation extraction. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2205–2215, Brussels, Belgium, October-November. Association for Computational Linguistics.