A New Approach to Automatically Find and Fix Erroneous Labels in Dependency Parsing Treebanks

Metin Bilgin
Department of Computer Engineering, Bursa Uludağ University, Turkey

Abstract: *Dependency Parsing (DP) is the existence of sub-term/upper-term relations between the words that make up that sentence for each sentence in the text. DP serves to produce meaningful information for high-level applications. Correct labeling of the text corpus used in DP studies is very important. There will be mistakes in the results of the studies that will be performed with the wrongly-labeled text corpus. If text corpus is labeled manually or automatically by human beings, then faulty cases will occur. As a result of the cases that may arise from human factors or annotations used for labeling, faulty labels will be on treebanks. In order to prevent these errors, detection, and correction of possible faulty labeling is very important in terms of increasing the accuracy of the studies to be carried out. Manual correction of possible faulty labels requires great effort and time. The purpose of this study is to create a model that automatically finds possible faulty labels and offers new label suggestions for faulty labels. With the help of the proposed model, it is aimed to detect and correct possible faulty labels that are included in a text corpus, and to increase consistency among the text corpus of the same language. With the help of the developed model, suggesting new labels for faulty labels by a language expert will be a great convenient for the specialist. Another advantage of the model is that the developed model provides a language-independent structure. It has succeeded in obtaining successful results in finding and correcting potentially faulty labels in experimental studies for Turkish. An increase in accuracy has been detected in studies carried out for languages other than Turkish. In investigating the accuracy of the results obtained by the system, the results were analyzed with the help of 10 different language experts.*

Keywords: *Natural language processing, dependency parsing, universal dependency, error detection, treebank consistency*.

Received July 27, 2020; accepted January 19, 2021
https://doi.org/10.34028/iajit/18/3/12

1. Introduction

Dependency Analysis (DA) approach put forward by Tesnière has been a method used extensively in the field of syntactic analysis since the nineties [35]. The most important reasons for this are the ability of dependency trees to produce outputs that are directly processable and closer to semantic deductions for high-level Natural Language Processing (NLP) studies. In this approach, the syntactic analysis of sentences is done by determining binary dependency relations and types between the units that make up the sentence.

According to Tesnière, “The sentence is a regular set whose elements consist of vocables” [35]. The mind finds relations between the vocables that make up the sentence and its neighbors, and all of these relations form the skeleton of the sentence. Each relation links a sub-term to an upper term. Dependency Parsing (DP), which is used in the field of Natural Language Understanding (NLU) today, is defined as the dependent-governor relationship [22].

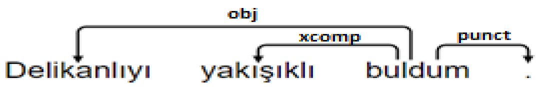


Figure 1. Dependency tree for a turkish sentence [Turkish-GB].

For example, in Figure 1, the word “Delikanlıyı” is related to the action of “buldum” with the “obj” relation [32]. Generally, every text corpus study will inevitably contain some errors. Some of these errors may be due to the annotation guidelines leaving some cases open, some related to the knowledge-attention of the marker, and some may be due to automated processes [15]. In addition, there may be problems arising from the fact that expressed language is really a difficult issue to decide on or that linguists could not agree on. Even if you are very meticulous in text corpus studies, it is common to have mistakes or mark the same types of structures in different ways. It is not an easy task to make these by hand, considering the size and volume of the text corpus. [14]

As an example of simple human errors, marking the word “onlar” as Pronoun (PRON) when saying “onlar basamağındaki sayı can be said as a simple error of carelessness. According to Universal Dependency (UD), is the word “onlar NOUN? or is it NUM(number)? might be the subject of discussion [32].

As an example of a difficult issue to decide; some of the derivational affixes in Turkish can be given. According to the current Turkish UD rules, in some words that have the suffix -li, the suffix is separated from the stem. However, if the word has been lexicalized, there is no need to separate it. If we look at the examples from Turkish UD, when we say “uzun parmaklı elleri, the suffix -lı is separated, when you say “çeşitli davalar”, the vocable is separated, and but when you say “sağlıklı”, there are both those who want to separate it and those who do not want. Which derivational affixes are separated and which are not has been discussed for a long time and are not agreed on 100% yet [32].

It can be said that the biggest problems of the current text corpora are inconsistencies both within themselves and among each other. The problems in the quality of text corpora are reflected negatively on the parsing results. Studies carried out so far are mainly on decomposition experiments on treebank. It can be said that the innovative aspect of this study is the first study on the automatic detection and correction of the errors that the text corpora contain, especially for Turkish. That treebank includes less error that will have a positive impact on subsequent studies that will use the output from DP as input in addition to increasing the accuracy of results of dependency analysis to be carried out.

2. Related Works

In recent years diverse approaches have been proposed to detect errors and inconsistency of the treebanks in different languages [16]. These approaches are categorized into three groups such as metaheuristic [4, 17, 18] statistical and rule-based [2]. Also, the error types could be classified as POS, morphology, chunk, dependency relation, etc., This study aims to find possible dependency relation errors.

Dale and Kilgarriff [9] measured performance with 6 different teams for an assisting system that helps the author when writing. They did the process on Extensible Markup Language (XML) data that includes 19 different sources and means 940 words.

Dale *et al*. [10] measured performance with 14 different approaches that are created by different teams to detect prepositions and determiners in English. They used the First Certificate in English (FCE) dataset that has 1000 files for train and test. The training set has 900 files, 374680 words and each file has an average 375 words. The test set has 100 files, 18013 words and each file has an average of 180 words. Grundkiewicz and Junczys-Dowmunt [23] used Wiked Error dataset that has 12 million sentences.

Ng *et al*. [25] measured performance with 17 different approaches that are created by different teams to approve grammatical errors from 1447 articles on the Nucle dataset. They have examined 5 error types such as Article or determiner, Preposition, Noun number, Verb form, Subject-Verb agreement. The train set has 1397 articles, 57151 sentences, and 1161567 tokens and the test set has 50 articles, 1381 sentences, and 29207 tokens. Ng *et al*. [26] did likely study in 2014. In this study, 13 different teams examined 28 different error types such as verb tense, verb form, etc..

Bryant *et al*. [5] designed Grammatical Error Annotation Toolkit. This system has been rule-based and did sentence classification. They used 25 different error types (adjective big=wide, contractiont=not, etc.,). They measured performance 12 different approaches based on the CoNLL-2014 dataset.

Ambati *et al*. [2] studied to detect dependency errors on Hindi Treebank. They used the Maximum Entropy Markov Model. This study used a statistical model based on the frequency and a statistical model based on probability. It has achieved to detect errors with 76.33% in the recall.

Ambati *et al*. [1] realized the system that is to verify the treebank as semi-automatic. In this study on Hindi Treebank, errors such as POS tagging, Chunk, and Dependency was found. They proposed a hybrid system that is based on rules and statistics. When it has achieved to guess 340 of 843 errors, it has reached an accuracy of 40.33% in the recall.

Dickinson and Smith [19] proposed a system that is based on n-gram for parse error detection. This study has used the Wall Street Journal Corpus which has converted to Stanford Dependency.

Tezcan *et al*. [36] proposed two new methods to detect grammatical errors for Dutch. This study has used the Scate Corpus which has 160201 sentences. They did detect errors both sentence-level and word-level. Finally, they used a hybrid system with two methods. This study has achieved the best accuracy value on sentence-level.

Hovy *et al*. [24] tried to found the errors on the manual and automatically labeled the dataset by Bayes methods based on active learning. This study has tried to detect errors and increase accuracy. The proposed system is unsupervised and generative based on Multi-Annotator Competence Estimation (MACE). Rehbein and Ruppenhofer combined their model and MACE. They used five different parsers for the pre-processing part and five different datasets. This system has needed to enter true labels and ID by an expert person [31].

In addition to these studies, there are also studies for Korean [6, 28] and Russian [20].

3. Material and Methods

In this part, after briefly discussing the Universal Dependencies project (UD), the information about used treebanks and proposed methods are going to be presented.

3.1. Universal Dependency Project (UD)

UD is a framework that includes reciprocal consistent explanations among different languages. The objectives of the UD might be assumed to analyze the researches from the perspective of a language as well as to develop multilingual decompositions and facilitate the learning process among languages. It is benefited from Stanford dependencies [11, 12, 13], Google Universal part-of-speech tags [30], and Interset Interlingua for morphosyntactic tagsets [38] to form analysis schemas.

UD uses the revised version of the CoNLL-X (Computational Natural Language Learning- The tenth CoNLL) format which is called as CoNLL-U (Computational Natural Language Learning-Universal Dependencies). Every word is defined with 10 different fields and separated with tab characters. Comment lines start with \# character. Sentences can be composed of one or more word lines and word lines represent the fields to be seen in the following findings. The examples of the Turkish language in the CoNLL-U format can be seen in Table 1.

Table 1. A Turkish sentence in CoNLL-U format.

ID	Form	Lemma	UPOS	XPOS	Feats	Head	Deprel	Deps	Misc

#sent_id=mst-0036
text=Nefes nefese kalmıştım.									

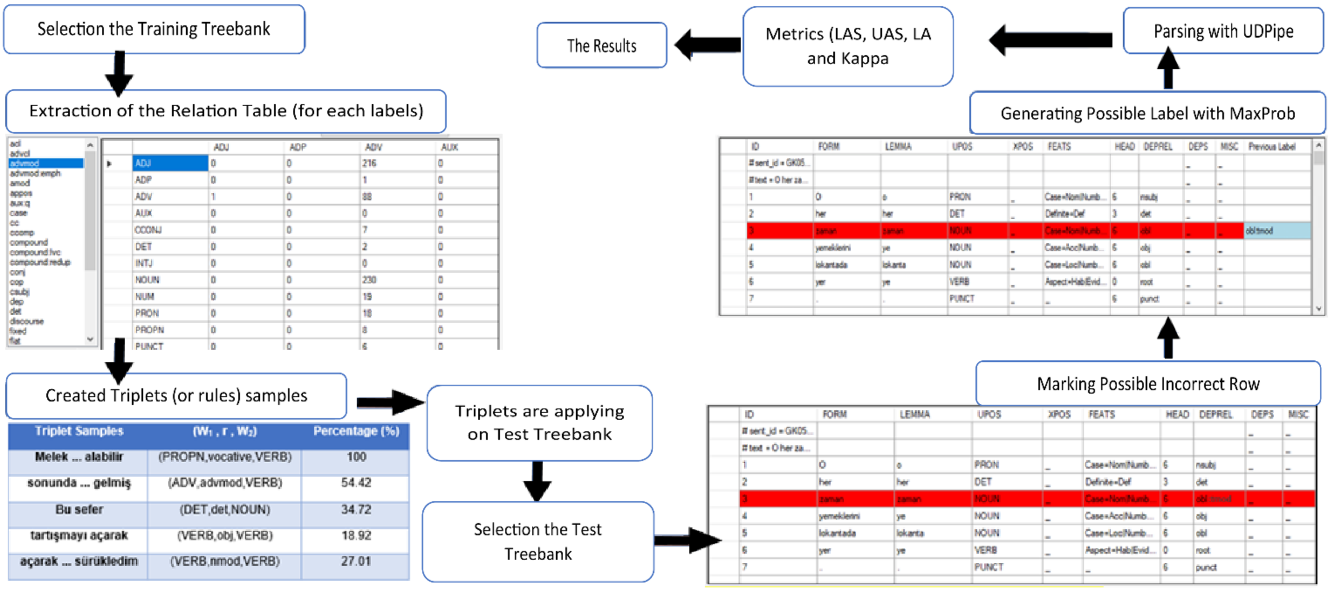
1	Nefes	nefes	NOUN	noun	Case=Nom|Number=Sing|Person=3	0	root	_	_
2	nefese	nefes	NOUN	noun	Case=Dat|Number=Sing|Person=3	1	compound	_	_
3	kalmıştım	kal	VERB	Verb	Aspect=Perf|Mood=Ind|Number=Sing|Person=1| Polarity=Pos|Tense=Pqp	1	compound	_	_
4	.	.	PUNCT	punct	_	1	punct	_	_

3.2. Treebanks

In this study, 12 different treebanks [37] were used, 3 of which are from Turkish [8, 29, 33, 34] ITU-METU-Sabancı Treebank (IMST) for trainingand GB (GrammarBook)-PUD (Parallel Universal Dependencies) fortesting, 2 from German GSD (German Stanford Style dependencies) for training and Parallel Universal Dependencies (PUD) fortesting, 3 fromSwedishTalbanken for training and PUD-Lines (Linköping English-Swedish Parallel Treebank) fortesting, 2 from Norwegian Nynorsk for training and NynorskLIA fortesting, and 2 fromPortuguese (GSD-Google Universal Dependency Treebank) fortraining and PUD fortesting. Treebanks were involved in the 2.5 version and created within the scope of UD.

3.3. Proposed Methods

The approach that we developed is presented in Figure 2. Firstly, we chose to use the treebank with the highest number of sentences in the studied language in order to identify possible faulty labels in a treebank. During the training stage, the system is started by selecting the treebank whose Triplet patterns will be identified. The system identifies the labels in the selected treebank and creates a Relation Table (RT) for each selected label. Triplet patterns and percentages can be created using the created RTs. With the flexibility provided by the developed system, these stages can be done step by step and the results can be seen, and all steps can be operated automatically and results can be obtained.



Selection the Training Treebank
Extraction of the Relation Table (for each labels)
acl
advcl
advmod
advmod:emph
amod
appos
aux:q
case
cc
ccomp
compound
compound:lvc
compound:redup
conj
cop
csubj
dep
det
discourse
fixed
flat
Created Triplets (or rules) samples
Triplets are applying on Test Treebank
Selection the Test Treebank
Marking Possible Incorrect Row
Generating Possible Label with MaxProb
Parsing with UDPipe
Metrics (LAS, UAS, LA and Kappa
The Results

	ADJ	ADP	ADV	AUX
ADJ	0	0	216	0
ADP	0	0	1	0
ADV	1	0	88	0
AUX	0	0	0	0
CCONJ	0	0	7	0
DET	0	0	2	0
INTJ	0	0	0	0
NOUN	0	0	230	0
NUM	0	0	19	0
PRON	0	0	18	0
PROPN	0	0	8	0
PUNCT	0	0	6	0

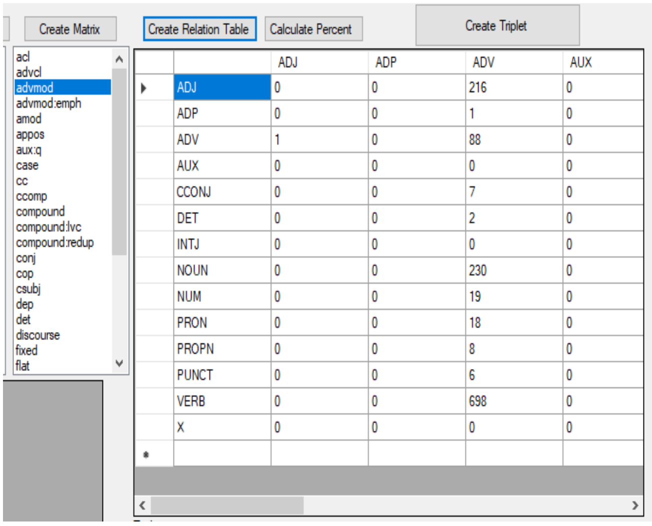
Triplet Samples	(W_1, r, W_2)	Percentage (%)
Melek ... alabilir	(PROPN,vocative,VERB)	100
sonunda ... gelmiş	(ADV,advmod,VERB)	54.42
Bu sefer	(DET,det,NOUN)	34.72
tartışmayı açarak	(VERB,obj,VERB)	18.92
açarak ... sürükledim	(VERB,nmod,VERB)	27.01

ID	FORM	LEMMA	UPOS	XPOS	FEATS	HEAD	DEPREL	DEPS	MISC	Previous Label
# sent_id = GK05...								_	_	
# text = O her za...								_	_	
1	O	o	PRON	_	Case=Nom|Numb...	6	nsubj	_	_	
2	her	her	DET	_	Definite=Def	3	det	_	_	
3	zaman	zaman	NOUN	_	Case=Nom|Numb...	6	obl	_	_	obl:tmod
4	yemeklerini	ye	NOUN	_	Case=Acc|Numb...	6	obj	_	_	
5	lokantada	lokanta	NOUN	_	Case=Loc|Numb...	6	obl	_	_	
6	yer	ye	VERB	_	Aspect=Hab|Evid...	0	root	_	_	
7	.	.	PUNCT	_	_	6	punct	_	_	

ID	FORM	LEMMA	UPOS	XPOS	FEATS	HEAD	DEPREL	DEPS	MISC
# sent_id = GK05...								_	_
# text = O her za...								_	_
1	O	o	PRON	_	Case=Nom|Numb...	6	nsubj	_	_
2	her	her	DET	_	Definite=Def	3	det	_	_
3	zaman	zaman	NOUN	_	Case=Nom|Numb...	6	obl:tmod	_	_
4	yemeklerini	ye	NOUN	_	Case=Acc|Numb...	6	obj	_	_
5	lokantada	lokanta	NOUN	_	Case=Loc|Numb...	6	obl	_	_
6	yer	ye	VERB	_	Aspect=Hab|Evid...	0	root	_	_
7	.	.	PUNCT	_	_	6	punct	_	_

Figure 2. The flowchart of the developed approach (MaxProb).

The RT sample obtained for the selected label is given in Figure 3.

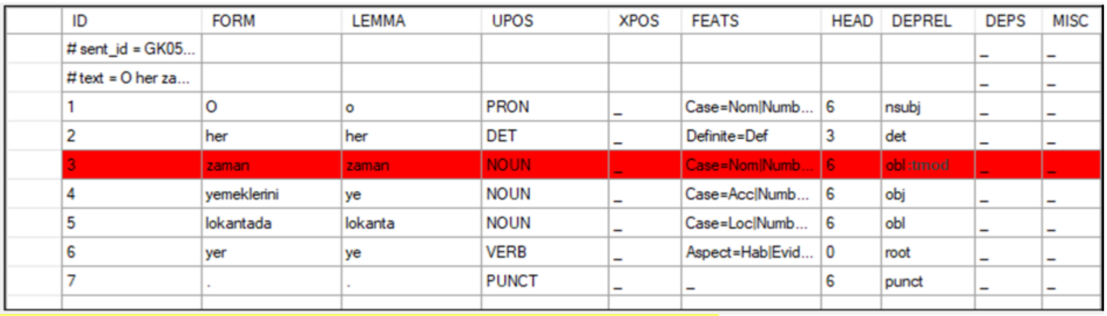


	ADJ	ADP	ADV	AUX
ADJ	0	0	216	0
ADP	0	0	1	0
ADV	1	0	88	0
AUX	0	0	0	0
CCONJ	0	0	7	0
DET	0	0	2	0
INTJ	0	0	0	0
NOUN	0	0	230	0
NUM	0	0	19	0
PRON	0	0	18	0
PROPN	0	0	8	0
PUNCT	0	0	6	0
VERB	0	0	698	0
X	0	0	0	0

Figure 3. The Relation Table sample for advmod label.

Every triplet pattern is expressed as (W_1, r, W_2). W_1 refers to a first syntactic word (UPOS), r to syntactic tag (DEPREL), and W_2 to the second syntactic word (UPOS). A piece of software that shows the relations between UPOS-DEPREL columns as triplet over 5 different treebanks used in the tests for feature extraction, is developed. As a result of this, a new feature independent from the language is tried to be defined by using word types and dependency tags instead of words.

Following the creation of triplets, treebank, which is thought to have possible faulty labels, is given to the system. The system scans every line in the test treebank using the created triplet patterns. The pattern of the relevant line is searched within the triplet set. If the related pattern cannot be found in the triplet set, that line is marked as incorrect. The example of this case is given in Figure 4.



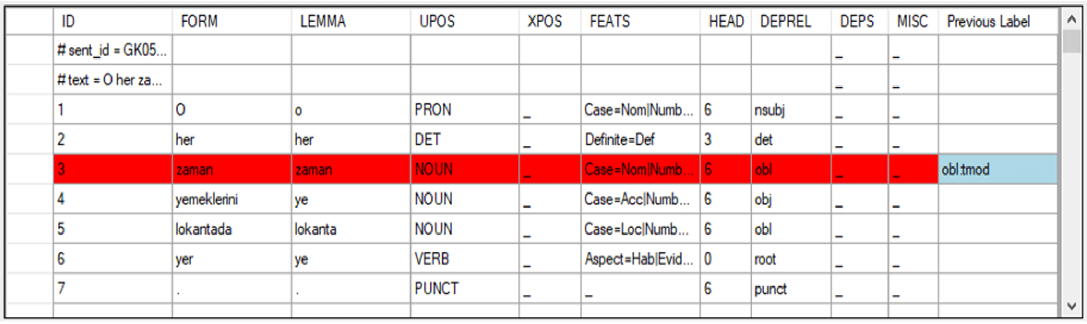
ID	FORM	LEMMA	UPOS	XPOS	FEATS	HEAD	DEPREL	DEPS	MISC
# sent_id = GK05...								_	_
# text = O her za...								_	_
1	O	o	PRON	_	Case=Nom|Numb...	6	nsubj	_	_
2	her	her	DET	_	Definite=Def	3	det	_	_
3	zaman	zaman	NOUN	_	Case=Nom|Numb...	6	obl:tmod	_	_
4	yemeklerini	ye	NOUN	_	Case=Acc|Numb...	6	obj	_	_
5	lokantada	lokanta	NOUN	_	Case=Loc|Numb...	6	obl	_	_
6	yer	ye	VERB	_	Aspect=Hab|Evid...	0	root	_	_
7	.	.	PUNCT	_	_	6	punct	_	_

Figure 4. Marking of the probable incorrect row.

Following the marking, new labeling can be made through two different approaches we suggest.

- Assign maximum possibility label for related triplet pattern (Maximum Probability-MaxProb)
- Assign one among 3 maximum possibility label for related triplet pattern (Random Maximum Probability-RndMaxProb)

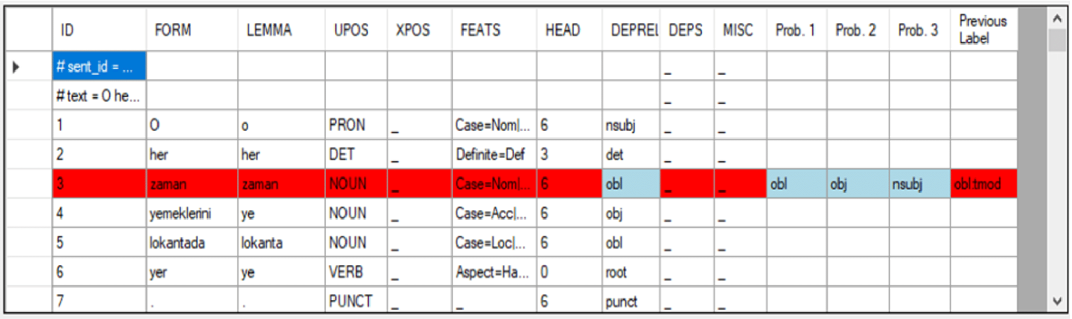
In the MaxProb approach, the label of the triplet with the highest probability corresponding to the Triplet Pattern of the line thought to be faulty is assigned as the new label of the relevant line. The example to this case is presented in Figure 5. In the example, the obl:tmod label of the (NOUN, obl:tmod, VERB) triplet is replaced by obl, the label of the (NOUN, obl, VERB) triplet, which has a higher probability.



ID	FORM	LEMMA	UPOS	XPOS	FEATS	HEAD	DEPREL	DEPS	MISC	Previous Label
# sent_id = GK05...								_	_	
# text = O her za...								_	_	
1	O	o	PRON	_	Case=Nom|Numb...	6	nsubj	_	_	
2	her	her	DET	_	Definite=Def	3	det	_	_	
3	zaman	zaman	NOUN	_	Case=Nom|Numb...	6	obl	_	_	obl:tmod
4	yemeklerini	ye	NOUN	_	Case=Acc|Numb...	6	obj	_	_	
5	lokantada	lokanta	NOUN	_	Case=Loc|Numb...	6	obl	_	_	
6	yer	ye	VERB	_	Aspect=Hab|Evid...	0	root	_	_	
7	.	.	PUNCT	_	_	6	punct	_	_	

Figure 5. Generating the maximsum probable label with MaxProb.

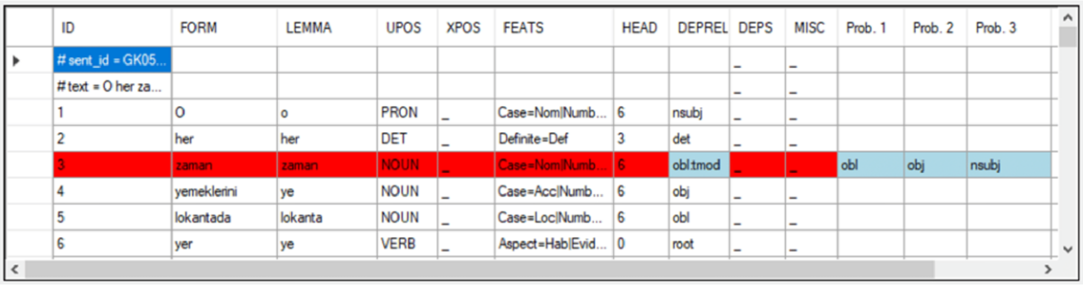
In the RndMaxProb approach, a random one of three triplets with the highest probability corresponding to the triplet pattern of the line thought to be faulty is assigned as the new label of the respective line. The example to this case is presented in Figure 6. In the sample; the label of the obl triplet was replaced by (NOUN, obl:tmod, VERB)'s label of obl:tmod, which was randomly selected from the triplets (NOUN, obl,VERB), (NOUN, obl, VERB) and (NOUN, nsubj, VERB).



	ID	FORM	LEMMA	UPOS	XPOS	FEATS	HEAD	DEPREL	DEPS	MISC	Prob. 1	Prob. 2	Prob. 3	Previous Label
▶	# sent_id = ...								_	_				
	# text = O he...								_	_				
	1	O	o	PRON	_	Case=Nom|...	6	nsubj	_	_				
	2	her	her	DET	_	Definite=Def	3	det	_	_				
	3	zaman	zaman	NOUN	_	Case=Nom|...	6	obl	_	_	obl	obj	nsubj	obl:tmod
	4	yemeklerini	ye	NOUN	_	Case=Acc|...	6	obj	_	_				
	5	lokantada	lokanta	NOUN	_	Case=Loc|...	6	obl	_	_				
	6	yer	ye	VERB	_	Aspect=Ha...	0	root	_	_				
	7	.	.	PUNCT	_	_	6	punct	_	_				

Figure 6. Generating the random maximum probable label with RndMaxProb.

An expert process has been designed to investigate the accuracy of the new approaches that we have developed. In this step, it is provided that labels with the highest 3 probabilities for the label considered to be incorrect are automatically recommended to a specialist and the specialist can manually define the label that should be for the label. The expert can choose the selection of correction from the 3 recommended labels, as well as suggesting that the existing label remains the same or suggest a new label. This step is about investigating the accuracy of the corrections made. The example to this case is presented in Figure 7.



	ID	FORM	LEMMA	UPOS	XPOS	FEATS	HEAD	DEPREL	DEPS	MISC	Prob. 1	Prob. 2	Prob. 3
▶	# sent_id = GK05...								_	_			
	# text = O her za...								_	_			
	1	O	o	PRON	_	Case=Nom|Numb...	6	nsubj	_	_			
	2	her	her	DET	_	Definite=Def	3	det	_	_			
	3	zaman	zaman	NOUN	_	Case=Nom|Numb...	6	obl:tmod	_	_	obl	obj	nsubj
	4	yemeklerini	ye	NOUN	_	Case=Acc|Numb...	6	obj	_	_			
	5	lokantada	lokanta	NOUN	_	Case=Loc|Numb...	6	obl	_	_			
	6	yer	ye	VERB	_	Aspect=Hab|Evid...	0	root	_	_			

Figure 7. The Validation stage by experts.

4. Experimental Results

In order to detect errors on treebank, 5 treebanks in 5 different languages were used to be detected in the training stage. Triplet numbers formed after the training stage are Turkish (IMST)-759, Turkish (IMST Train+Dev)-729, Swedish-702, German-1029, Norwegian-915, and Portuguese-904. After creating triplets for relevant language, Test Treebank is selected and lines with potential errors are marked. Then, according to the chosen approach, changes are made on the relevant test treebank. The numbers of changes made on the test treebank are Turkish (GB)-1630, Turkish (PUD)-3004, Swedish (PUD)-165, Swedish (LINES)-1119, German-835, Norwegian-2343, and Portuguese-2371.

In order to investigate the effect of the changes made on the treebank, dependency parsing was carried out with the UD Pipe program. In the study, besides the 3 metrics frequently used to evaluate the results of DP problems in the literature, the Kappa metric, which measures how well the classifier actually performs, was used.

The study results are introduced in 3 different metrics. These;

- Unlabeled Attachment Score (UAS) [21]: percentage of words that get the correct head.
- Labeled Attachment Score (LAS) [27]: percentage of words that get the correct head and label (DEPREL).
- Labeled Accuracy (LA) [21]: percentage of words that get the correct label (DEPREL).

Besides these metrics, the Kappa metric is used for Turkish-GB treebank.

- Kappa: this metric is to present a coefficient to measure the degree of agreement in nominal scales, and to provide means of testing hypotheses and setting confidence limits for this coefficient [3, 7].

Kappa metric values for GB treebank were measured as 55.21%, 67.56%, 64.49% (respectively Raw Data, MaxProb ve RndMaxProb). Kappa metric is presented in Equation (1). Where P_o is the relative observed agreement among raters, P_e is the hypothetical probability of chance agreement.

$$Kappa = \frac{P_o - P_e}{1 - P_e} \quad (1)$$

Since the offered approaches do not affect the UAS metric, only the value on the raw treebank has been calculated (no change for other cases). LAS and LA metrics are calculated both in Turkish and in the other 4. The results obtained are given in Table 2. Also, the LA metric is presented in Equation (2).

Table 2. The results (Treebank (TB), Label Count (LC),Correct Label Counts (CLC), Difference (Diff.).

			Raw Data				**MaxProb**			**RndMaxProb**		
Language	Test TB	**LC**	**CLC**	**UAS**	**LAS**	**LA**	**Diff.**	**LAS**	**LA**	**Diff.**	**LAS**	**LA**
Turkish	GB	16881	10859	75.35	57.39	64.32	+773	**59.63**	**68.90**	+385	58.42	66.61
	PUD	15247	8273	56.58	35.27	54.26	+577	**37.33**	**58.04**	+287	36.21	56.14
	IMST (Test)	10254	7492	61.23	54.53	73.06	+3	**54.53**	**73.09**	+2	54.53	73.08
Swedish	LINES	90960	76112	78.93	72.80	83.67	+226	**72.83**	**84.58**	+90	72.81	83.78
	PUD	19085	16290	79.86	74.54	85.35	+106	**75.08**	**85.91**	+34	74.79	85.53
German	PUD	21657	17269	77.77	70.02	79.74	+283	**70.36**	**81.05**	+79	70.16	80.10
Norwegian	NynorskLIA	55408	42430	65.02	58.90	76.58	+293	**59.01**	**77.10**	+74	58.95	76.71
Portuguese	PUD	22301	14180	34.99	28.61	63.58	+352	**28.62**	**65.16**	+124	28.61	64.14

The validation stage was only possible for Turkish due to the problem of an unavailable expert. In this study, IMST (dev and train) treebank was used during the training stage and IMST-Test corpus was used as the test set. In the study, 28 changes were made on the test set with the MaxProb approach. In another experiment for this stage, IMST treebank was used during the training stage and the GB corpus was used as the test set. In this study, 1630 changes were made on the test set with the MaxPRob approach. In order to evaluate the results, studies were conducted on IMSTTest (28 changes) and GB (20 randomly selected changes) with the support of ten different language experts from Uppsala University, Tübingen University, and Bursa Uludağ University. The procedures for the old label to remain the same, the assignment of one of the labels offered by the MaxProb approach, and the assignment of a new label were conducted by the expert. It is marked as a contradictory situation on which the experts cannot agree. The evaluation results obtained are given in Tables 3 and 4.

$$Label\ Accuracy\ (LA) = \frac{True\ Positive + True\ Negative}{True\ Positive + True\ Negative + False\ Positive + False\ Negative} \qquad (2)$$

Table 3. MaxProb Results for Validation Stage (Treebank (TB), Label Count (LC), Previous Label-PL, New Label (NL), Conflicting State (CS).

Training TB	**Test TB**	**LC**	**MaxProb**	**PL**	**NL**	**CS**	**LA**
IMST (Training+Dev)	IMST (Test)	28	13	3	7	5	**46.42**
IMST (All)	GB	20	12	-	8	-	**60**

Table 4. Samples of editing rows Sentence (S)., Probability (Prob)., Expert (Exp).

TB	**S.ID**	**ID**	**Deprel**	**Prob.1**	**Prob.2**	**Prob.3**	**Exp.1**	**Exp.2**	**Exp.3**	**Exp.4**	**Exp.5**	**Exp.6**	**Exp.7**	**Exp.8**	**Exp.9**	**Exp.10**	**Result**
IMST	0862	8	obj	nummod	det	flat	nummod	nummod	nummod	nummod	nummod	nummod	nummod	nummod	det	nummod	**MaxProb**
	1273	7	cc	obl	csubj	nsubj	obl	obl	obl	obl	obl	obl	obl	cc	csubj	obl	**MaxProb**
	0913	1	acl	csubj	conj	nmod	acl	acl	acl	csubj	acl	acl	acl	acl	acl	csubj	Unchanged
	4600	2	amod	compound	acl	csubj	advcl	advcl	compound	advcl	csubj	advcl	advcl	advcl	advcl	advcl	New Label
	3723	2	compound	advmod	mark	cc	fixed	goeswith	fixed	fixed	cc	cc	fixed	fixed	fixed	cc	Conflicting
GB	0153	1	nmod	amod	nmod: poss	det	amod	amod	amod	amod	nmod	amod	nmod:poss	amod	amod	amod	**MaxProb**
	0170	1	amod	flat	amod	compound	nmod	nmod	amod	nmod	nmod	nmod	amod	amod	nmod	nmod	New Label

5. Discussion

It is known that most of the data sets prepared for commitment parsing are prepared manually and this causes incorrect labeling on the datasets. Since manual marking of data sets causes both time-consuming and incorrect labeling, it has become necessary to develop systems supported by computers. In this study, 2 new approaches are presented in order to automatically solve the problem of wrong labeling and experimental studies have been carried out on datasets belonging to 5 different languages.

The proposed approaches for all studied language treebanks have provided more or less improvement. As a result of the study, the biggest change was in Turkish-GB with 4.58% for LA, while the smallest change was in Swedish-PUD with 0.36% for LA.

There may be several reasons for these changes. The size of the selected training set and the number of different pattern numbers it contains may be the reasons for this case.

As a result of the study for the Turkish treebank, more correct changes were made to the GB treebank.

The reason for this may be that GB is a larger treebank than PUD. It may not be correct to say that much number of training and test text corpus always increases the number of corrections. Examples of this case are Swedish-PUD and German-PUD. Although they are large treebanks in structure, the number of changes produced by the system remains limited.

Considering this case, the greater the similarity between the treebanks of the same language, the lower the number of changes produced by our system will be. When the obtained results are evaluated, it can be said that the Swedish-Talkanben/LINES compilations and the Swedish-Talbanken/PUD compilations are more consistent with each other. As a result of the study, Turkish-PUD and Portuguese-PUD treebanks reached the lowest LA value.

When the results obtained in the validation stage are evaluated, 13 (accuracy about 46.42%) of the 28 changes suggested by the proposed approach on the IMST-Test treebank were evaluated by the experts as correct statements. Similarly, our system has succeeded in making 12 (accuracy 60%) correct proposals for 20 proposed changes on the GB treebank. It was decided to keep 4 labels as they are for IMST-Test and to suggest the new label for 7 labels and to mark 6 cases as contradictory. No unchanged or contradictory case has been found for GB. In addition, the new label suggestion was made for 8 labels in GB.

Since the UAS metric is not related to the label, it was not expected to be affected by the experimental studies performed and the results obtained confirm that there is no change. Label corrections made for all treebanks have resulted in a positive increase on LAS and Kappa metric as well as LA. With these positive results, it is ensured that the approach is not valid. Kappa metric calculated for Turkish-GB reached thehighest value for MaxProb, followed by RndMaxProb and Raw Data values. When the Kappa metric values were examined, it was parallel to the LAS and LA metrics. The Kappa metric, used to measure the accuracy of the classifier, proves the accuracy of the results we get for other metrics.

6. Conclusions

This study is about finding and correcting faulty labels that are likely to exist in the treebank developed under the UD. Although data in CoNLL-U format is used, a structure that can be used for data in CoNLL-X format is presented. It can be said that the proposed system is usable after the experimental studies and the results obtained were controlled by an expert. Treebank to be chosen at the training stage, which is the most important step of the system, is vital. If the selected treebank itself is filled with errors, it is possible to transfer it to the test set. In addition, the Training treebank, which will be chosen correctly, will positively affect the overall performance of the system.

When the existing studies are examined, it is seen that manual or rule-based approaches are offered to correct defective labels. The designed systems can be designed like manual systems in which all labels are controlled by an expert, as well as re-labeling the labels that are thought to be faulty with various approaches by an expert. Studies on automatic systems are very limited. In rule-based systems, there are difficulties in creating a rule suitable for every situation. Also, the approaches are generally word-based and specific to the developed language. Our approach offers a language-independent structure based on context rather than words. It has unique features because it is an approach that automatically extracts syntactic relationships in the existing treebank, automatically marks possible incorrect lines, and offers correction suggestions (can be done automatically if desired). Thus, it can be said that the approach saves labor and time.

Acknowledgments

This work was supported by The Scientific and Technological Research Council of Turkey (TUBITAK). International post-doctoral research fellowship #2219.

I would like to thank Professor Joakim Nivre for his expert advice, encouragement, and also research opportunity at Uppsala University throughout this study.

References

[1] Ambati B., Gupta M., Husain S., and Sharma D., "A High Recall Error Identification Tool for Hindi Treebank Validation," *in Proceedings of the International Conference on Language Resources and Evaluation*, Valletta, pp. 682-686, 2010.

[2] Ambati B., Agarwal R., Gupta M., Husain S., and Sharma D., "Error Detection for Treebank Validation," *in Proceedings of 9th International Workshop on Asian Language Resources*, Chiang, pp. 23-30, 2011.

[3] Bilgin M. and Köktaş H., "Sentiment Analysis with Term Weighting and Word Vectors," *The International Arab Journal of Information Technology*, vol. 16, no. 5, pp. 953-959, 2019.

[4] Boyd A., Dickinson M., and Meurers W., "On Detecting Errors in Dependency Treebanks," *Research on Language and Computation*, vol. 6, no. 2, pp. 113-137, 2008.

[5] Bryant C., Felice M., and Briscoe T., "Automatic Annotation And Evaluation of Error Types for Grammatical Error Correction," *in Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, Vancouver, pp. 793-805, 2017.

[6] Chun J., Han N., Hwang J., and Choi J., "Building Universal Dependency Treebanks in Korean," *in Proceedings of the 11th International Conference on Language Resources and Evaluation*, Miyazaki, pp. 2194-2202, 2018.

[7] Cohen J., "A Coefficient of Agreement for Nominal Scales," *Educational and Psychological Measurement*, vol. 20, no. 1, pp. 37-46, 1960.

[8] Çöltekin Ç., "A Grammar-Book Treebank of Turkish," *in Proceedings of the 14th workshop on Treebanks and Linguistic Theories*, Warsaw, pp. 35-49, 2015.

[9] Dale R. and Kilgarriff A., "Helping our Own: The HOO 2011 Pilot Shared Task," *in Proceedings of the Generation Challenges Session at the 13th European Workshop on Natural Language Generation*, Nancy, pp. 242-249, 2011.

[10] Dale R., Anisimoff I., and Narroway G., "A Report on The Preposition and Determiner Error

Correction Shared Task," *in Proceedings of the 7th Workshop on the Innovative Use of NLP for Building Educational Applications*, Montreal, pp. 54-62, 2012.

[11] De Marneffe M., MacCartney B., and Manning C., "Generating Typed Dependency Parses from Phrase Structure Parses," *in Proceedings of the 5th International Conference on Language Resources and Evaluation*, Genoa, pp. 449-454, 2006.

[12] De Marneffe M. and Manning C., "The Stanford Typed Dependencies Representation," *in Proceedings of the workshop on Cross-Framework and Cross-Domain Parser Evaluation Association for Computational Linguistic*, Manchester, pp. 1-8, 2008.

[13] De Marneffe M., Dozat T., Silveira N., Haverinen K., Nivre J., and Manning C., "Universal Stanford Dependencies: A Cross-Linguistic Typology," *in Proceedings of the 9th International Conference on Language Resources and Evaluation*, Reykjavik, pp. 4585-4592, 2014.

[14] Del Río I., Antunes S., Mendes A., and Janssen M., "Towards Error Annotation in a Learner Corpus of Portuguese," *in Proceedings of the Joint Workshop on NLP for Computer Assisted Language Learning and NLP for Language Acquisition*, Umea, pp. 8-17, 2016.

[15] Diaz-Negrillo A. and Fernández-Domíguez J., "Error Tagging Systems for Learner Corpora," *Revista Española De Lingüística Aplicada*, vol. 19, no. 83, pp. 83-102, 2006.

[16] Dickinson M., "Detection of Annotation Errors in Corpora," *Language and Linguistics Compass* vol. 9, no. 3, pp. 119-138, 2015.

[17] Dickinson M. and Meurers W., "Detecting Inconsistencies in Treebank," *in Proceedings of the 2nd Workshop on Treebanks and Linguistic Theories*, Växjö, pp. 45-56, 2003.

[18] Dickinson M. and Meurers W., "Detecting Errors in Discontinuous Structural Annotation," *in Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics*, Ann Arbor, pp. 322-329, 2005.

[19] Dickinson M. and Smith A., "Simulating Dependencies to Improve Parse Error Detection," *in Proceedings of the 15th International Workshop on Treebanks and Linguistic Theories*, Bloomington, pp. 76-88, 2017.

[20] Droganova K., Lyashevskaya O., and Zeman D., "Data Conversion and Consistency of Monolingual Corpora: Russian UD Treebanks," *in Proceedings of the 17th International Workshop on Treebanks and Linguistic Theories*, Oslo, pp. 52-65, 2018.

[21] Eisner J., "Three New Probabilistic Models For Dependency Grammar," *in Proceedings of the 6th International Conference on Computational Linguistics*, Stroudsburg, pp. 340-345, 1996.

[22] Eryiğit G., "Dependency Parsing of Turkish," PhD Thesis, İstanbul Technic University, 2006.

[23] Grundkiewicz R. and Junczys-Dowmunt M., "The WikEd Error Corpus: A Corpus of Corrective Wikipedia Edits and its Application to Grammatical Error Correction," *in Proceedings of in International Conference on Natural Language Processing*, Warsaw, pp. 478-490, 2014.

[24] Hovy D., Berg-Kirkpatrick T., Vaswani A., and Hovy E., "Learning Whom to Trust with MACE," *in Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, Atlanta, pp. 1120-1130, 2013.

[25] Ng H., Wu S., Wu Y., Hadiwinoto C., and Tetreault J., "The CoNLL- 2013 shared task on Grammatical Error Correction," *in Proceedings of the 7th Conference on Computational Natural Language Learning: Shared Task*, Sofia, pp. 1-12, 2013.

[26] Ng H., Wu S., Briscoe T., Hadiwinoto C., Susanto R., and Bryant C., "The CoNLL-2014 Shared Task on Grammatical Error Correction," *in Proceedings of the 8th Conference on Computational Natural Language Learning: Shared Task*, Baltimore, pp. 1-14, 2014.

[27] Nivre J., Hall J., and Nilsson J., "Memory-based Dependency Parsing," *in Proceedings of the 8th Conference on Computational Natural Language Learning*, Boston, pp. 49-56, 2004.

[28] Noh Y., Han J., Oh T., and Kim H., "Enhancing Universal Dependencies for Korean," *in Proceedings of the 2nd Workshop on Universal Dependencies*, Brussels, pp. 108-116, 2018.

[29] Oflazer K., Say B., Hakkani-Tür D., and Tür G., *Treebanks*, Springer Dordrecht, 2003.

[30] Petrov S., Das D., and McDonald R., "A Universal Part-of-Speech Tagset," *in Proceedings of the 8 International Conference on Language Resources and Evaluation*, Istanbul, pp. 2089-2096, 2012.

[31] Rehbein I. and Ruppenhofer J., "Sprucing up the Trees-Error Detection in Treebanks," *in Proceedings of the 27th International Conference on Computational Linguistics*, Santa Fe, pp. 107-118, 2018.

[32] Sulubacak U., Eryiğit G., and Pamay T., "IMST: A Revisited Turkish Dependency Treebank," *in Proceedings of 1st International Conference on Turkic Computational Linguistics*, Konya, pp. 1-6, 2016.

[33] Sulubacak U., Gökırmak M., Tyers F., Çöltekin Ç,, Nivre J., and Eryiğit G., "Universal Dependencies for Turkish," *in Proceedings of the 26th International Conference on Computational*

Linguistics: Technical Papers, Osaka, pp. 3444-3454, 2016.

[34] Sulubacak U. and Eryiğit G., "Implementing Universal Dependency, Morphology and Multiword Expression Annotation Standards for Turkish Language Processing," *Turkish Journal of Electrical Engineering and Computer Sciences*, vol. 26, no. 3, pp. 1662-1672, 2018.

[35] Tesnière L., *Introduction A la SyntaxeStructurale*, Klincksieck Press, 1959.

[36] Tezcan A., Hoste V., and Macken L., "Detecting Grammatical Errors in Machine Translation Output Using Dependency Parsing and Treebank Querying," *Baltic Journal Modern Computing*, vol. 4, no. 2, pp. 203-217, 2016.

[37] Treebanks url {https:// universaldependencies.org/}, Last Visited, 2020.

[38] Zeman D., "Reusable Tagset Conversion Using Tagset Drivers," *in Proceedings of the 6th International Conference on Language Resources and Evaluation*, Marrakech, pp. 213-218, 2008.

Metin Bilgin received the Ph.D. degree in Computer Engineering from Yıldız Technical University in 2015. Also, he did research post-doc in the Computational Linguistic department at Uppsala University for about 10 months.He is currently assistant professor in the Department of Computer Engineering, Bursa Uludağ University, Turkey. His current research interests include machine learning, natural language processing, and text classification.