Turkish Journal of Electrical Engineering & Computer Sciences

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

Research Article

Turk J Elec Eng & Comp Sci
() : –

doi:10.3906/elk-2102-57

Evaluating the English-Turkish parallel treebank for machine translation

Onur GORGUN[1,2,*], **Olcay Taner YILDIZ**[3]

[1]Computer Science and Engineering, Faculty of Engineering, Işık University, İstanbul, Turkey
[2]Research and Development Center, Nokia, İstanbul, Turkey
[3]Computer Science Department, Faculty of Engineering, Özyeğin University, İstanbul, Turkey

Received: 11.02.2021 • **Accepted/Published Online:** 02.09.2021 • **Final Version:** ..2021

Abstract: This study extends our initial efforts in building an English-Turkish parallel treebank corpus for statistical machine translation tasks. We manually generated parallel trees for about 17K sentences selected from the Penn Treebank corpus. English sentences vary in length: 15 to 50 tokens including punctuation. We constrained the translation of trees by (i) reordering of leaf nodes based on suffixation rules in Turkish, and (ii) gloss replacement. We aim to mimic human annotator's behavior in real translation task. In order to fill the morphological and syntactic gap between languages, we do morphological annotation and disambiguation. We also apply our heuristics by creating Nokia English-Turkish Treebank (NTB) to address technical document translation tasks. NTB also includes 8.3K sentences in varying lengths. We validate the corpus both extrinsically and intrinsically, and report our evaluation results regarding perplexity analysis and translation task results. Results prove that our heuristics yield promising results in terms of perplexity and are suitable for translation tasks in terms of BLEU scores.

Key words: Parallel treebank, parallel corpora, Turkish, English, syntax-based

1. Introduction

Treebanks are crucial to develop cutting-edge statistical natural language processing (NLP) systems like machine translation systems, syntactic parsers, and part-of-speech (POS) taggers. These NLP system models, especially machine translation systems, depend on the high quality and large scale of semantically or syntactically annotated corpora.

There are two types of syntactically annotated treebanks: Constituency (phrase structure) treebanks and dependency treebanks. Based on Chomsky's work [1], constituency structure dominates the major NLP tasks since it supports coheads or multinode dependencies. Penn Treebank (PTB) is the first large scale constitiuency structure treebank for English [2]. Other treebanks for major languages; German [3], French [4], agglutinative languages; Finnish [5] and Hungarian [6], the central semitic language Arabic [7], and the Sino-Tibetan language Chinese [8] follow PTB.

Turkish is an agglutinative and morphologically rich language with free word order. Depending on the discourse, constituents are reordered to emphasize certain elements in the sentence. Moreover, syntactic functions of the constituents are determined by case markings (e.g., dative, locative) [9]. Although statistical NLP research on Turkish has shown advances in recent years, Turkish is still behind its other agglutinative languages Finnish, Czech, and Hungarian.

*Correspondence: onur.gorgun@nokia.com

Parallel treebanks are common resources for statistical machine translation systems. Researchers have put much effort into building parallel treebanks and annotate them with syntactic features. In general, syntactic annotation purely depends on dependency and/or constituency order. There exist a lot of parallel treebanks in the literature. EuroParl [10] is one of the biggest parallel corpora, which contains 22 languages. English-German parallel treebank [11] contains sentences with constituency structure, predicate-argument structures, and all sentences are annotated with POS tags.

Linköping English-Swedish parallel treebank contains 1.2K sentences annotated with dependency structure [12]. Stockholm treebank is an English-German-Swedish multilingual corpora with 1K sentences annotated with constituency structure [13]. Prague treebank [14] is a parallel treebank for Czech-English language pair with dependency structure. Turkish has been subjected to parallel corpora generation studies in recent years, but these efforts have been limited to dependency structure and are only useful for phrase-based machine translation [15, 16].

In this study, we extend our initial efforts [17] for building the first Turkish treebank annotated with phrase structure and report these efforts in constructing an English-Turkish parallel treebank corpus for statistical machine translation. Our approach converts English parse trees into their Turkish equivalent parse trees by manual translation (leaf replacement), applying several transformation heuristics based on syntactic features of Turkish and the order of suffixation (tree permutation), morphological annotation (morphological analysis and disambiguation), and part-of-speech movement (POS movement). With this study, we increase the initial number of sentences by almost 13K by covering 15 to 50 token-long sentences. We expand our efforts on open domain to closed domain as well by applying our translation methodology on Nokia Treebank (translation of technical texts in telecommunication domain). We primarily aim to verify our translation model by applying perplexity analysis and performance evaluation on machine translation tasks, and to report our findings.

This paper is organized as follows: Section 2 introduces related works on parallel treebank construction efforts in Turkish. In Section 3, we briefly compare Turkish and English in terms of syntax and morphology. In Section 4, we demonstrate our corpus construction strategy through transformation heuristics. In Section 5, we explain how our corpus construction strategy was applied to build a closed-domain parallel corpus, namely Nokia Treebank (NTB). Experimental results are explained in Section 6. In Section 7, we conclude with discussions and future work.

2. Related work

In Turkish, treebank creation efforts date back to METU-Sabancı Turkish Treebank, which is a baseline and valuable resource for Turkish Dependency Parser. This treebank is expanded by 7K new sentences for advanced NLP tasks. The original treebank and expanded one are used in various Turkish NLP studies [18–24]. To addess the annotation inconsistencies, improvements are offered on the frequent cases in the existing treebank. Recently, there exists different treebanks in the literature which follow the footprints of METU-Sabancı Treebank; first semantically annotated corpus for Turkish [42], canonically annotated from different web resources (ITU) [41].

We see different types of treebanks supporting various NLP tasks. Many of those treebanks are based on the structure that is proposed in the METU-Sabancı Treebank. Those extended treebanks tend to extract combinatorial categorical grammars (CCG) with lexical annotation [25], and use lexical functional grammar formalism (LFG) to investigate the sublexical structures of lexical annotations [26, 27].

Beside the single-language treebanks in Turkish, there have been many studies that aim to build parallel treebanks. Parallel treebanks introduce part-of-speech tags and morphology to parsed sentences. Parallel

treebank efforts for Turkish starts with the introduction of Swedish-Turkish parallel treebank [28]. The corpus contains 160K and 145K tokens with dependency annotation in Swedish and Turkish, respectively. For syntactically and morphologically different languages, data sparsity is a recognized problem, and building parallel treebanks with a transitional middle language is a common approach to address this issue. Hence, for English-Turkish language pair, Megyesi et al. [29] expanded the Swedish-Turkish treebank to cover English by using Swedish as a transition language. This corpus contains 300K tokens in Swedish, 160K tokens in Turkish, and 150K tokens in English. Swedish is the primary language, and sentences in Swedish are translated into Turkish and English. Tokens are annotated based on dependency structure. For dependency annotation, MaltParser [30] used for Penn Treebank on English, Talbanken05 [31] is used on Swedish, and METU-Sabancı Turkish Treebank [18] is used on Turkish.

There are parallel treebanks that consist of more languages from different language families and different levels of annotation and alignment. ParGram parallel treebank [32] supports both constituency and dependency structures. This treebank is based on deep LFG and involves ten different languages: English, Georgian, German, Hungarian, Indonesian, Norwegian, Polish, Turkish, Urdu, Wolof.

3. Turkish vs. English: syntactic and morphological comparison

Turkish is an agglutinative language and has a very complex morphology obtained by rule-based suffixation. In some cases, the Turkish word-formation process may end up with an entire English sentence (See Figure 1a). The suffixation process in Turkish relies on different suffix categories and rules that identify the suffixation order. In Turkish, the majority of words are complex and they contain more than one syllable. The suffixation process follows vowel harmony and consonant agreement so that each vowel and consonants in the suffix depend on the morpheme preceeding it. Suffixes may assume different forms (allomorphs) due to vowel harmony and consonant agreement. For example, *plural* has two forms *+lar* and *+ler*. These different suffix forms (surface form) are indicated with their meta representations (lexical forms). Vowels that are subject to change due to vowel or consonant agreement are represented with capital letters (*+lAş* or *+DHr*, etc.).

(1) gitmeyeceksin (*git +me +yecek +sin*)

Turkish affixes are grouped under two classes: (i) derivational affixes and (ii) inflectional affixes. Derivation is the process of creating a new lexical form and achived through suffixation in Turkish. Turkish has a rich derivational affix inventory that allows the transition from one-word class to another. Inflectional affixes express the case, person, and tense relations between the nominals and verbals. Nominal inflectional affixes express number and case (singular 'kitap (*book.NN*)' or plural 'kitap +lAr (*book.PL*)'), possession (+(H)m (1*st singular*), +(H)n (2*nd singular*), +(s)H (3*rd singular*), (H)mHz (1*st plural*),(H)nHz (2*nd plural*), and lArH (3*rd plural*)). The suffixation is in number-possession-case order.

(2) kitap +lar +ın (*book.NN -PL -2SG.POSS*)

Turkish has a flexiable word order, and scrambling of constituents create differences in meaning [45]. The unmarked word order is subject-object-predicate. In general, the major Turkish constituents can appear in sentences in different orders to (i) to emphasize a particular constituent, (ii) deemphasize a particular constituent or constituents, or (iii) to make a particular constituent the pivot of information. Moreover, in Turkish, prepositions follow the noun structures, modal verbs follow the matrix verb, and relative clauses precede the noun they modify. In contrast to Turkish, English strictly follows SVO order.

(3) Ahmet kitabı aldı. (*Ahmed took the* ***book***.)

(4) Kitabı Ahmet aldı. (*It was* ***Ahmet*** *who took the book.*)

(5) Kitabı aldı Ahmet. (*It is the* ***book*** *that Ahmet took.*)

Turkish is a head-final language where the head of the phrase follows its complements, whereas English is a strictly head-initial language. Switching the constituent order leads to ungrammatical sentences in English. English keeps the same head-initial nature in different phrase structures, and the head preceeds its complements.

(6) Dün okuduğun kitap (*The* ***book*** *that you read yesterday*)

There are major differences between Turkish and English in terms of syntax and morphology. These differences make treebank generation a challenging process. English is a right-branching language, whereas Turkish is a left-branching [46]. As a result of the sytactic differences between two languages, syntactic trees show different behaviors. We apply a set of heuristics for different subtree translations to address possible gaps.

4. Corpus construction

In our annotation process, we used sentences from the Penn Treebank [2] in English. We took 17K sentences that vary in length up to 50 tokens to create our parallel treebank [1] We translated these 17K trees using our tree-based translation tool and obtain a total of 17K Turkish equivalents. In Table 1, the number of sentences are categorized based on the sentence length.

Table 1. Number of sentences in the Penn Treebank based on the sentence length.

	≤ 15	≤ 20	≤ 25	≤ 35	≤ 40	≤ 45	≤ 50
No. of sent.	9500	2300	1000	300	1000	2000	1000

We follow a 3-phase approach; tree translation by glossary translation and tree permutations (Phase-1), morphological feature extraction by morphological analysis and disambiguation (Phase-2), and morphological enrichment by mapping morphological features to their English equivalent gloss (Phase-3). We keep the English syntactic tags as it is, and do not introduce sub-tree movements. With this study, we translated 17K sentences to Turkish and applied morphological analysis. We also did morphological enrichment for 9.5K sentences.

4.1. Subtree reordering and glossary replacement

After manual translation, we provide the first annotation level in Turkish trees (Phase-1). We translate our trees by following a two-phase schema [17] to mimic the human-translator. We developed a tree translation tool for this purpose. By the help of the translation tool, annotators permutes the leaf node based on Turkish word ordering and suffixation rules . Then we replace the English word with its Turkish equivalent. We translate functional words, modifiers, and articles in English to Turkish as suffixes.

We permute the children of a node and replace the English word token at the leaf node. In Turkish, the majority of sentences have SOV order, whereas English sentences have SVO order. By permutation, we ensure that we have the correct gloss ordering in Turkish trees. We use the same tags and predicate labels in the leaf

[1]Turkish Annotated TreeBank [online]. Website https://github.com/olcaytaner/TurkishAnnotatedTreeBank-15 [accessed 12 June 2021].

nodes in Turkish trees as in English trees and do not introduce new tags. Addition/deletion of nodes is not allowed as well. If there is no direct gloss for any English token, we mark that leaf as *NONE*.

Alternatively, Yıldız et al. [38] apply an extended approach to build a treebank by structural changes. They apply the same steps which are presented in this study, until the end of Phase-2. There are important differences in morphological enrichment in their approach compared to ours. First, *NONE* leaves are discarded. Next, multiword expressions are branched into multiple leaves based on the POS tag of each seperate word under the same parent. They remove all ancestors until an ancestor that has more than one child is reached. Finally, to compensate for the information lost because of the morphological gaps between English and Turkish, they offer new tags as a combination of root POS tag and suffix's morphological category.

Even if the gloss is marked with *NONE*, we permute the node and keep the word in the same order with suffixation order in Turkish. Note that the POS tag sequence VP-RB-MD-PRP in the Turkish sentence corresponds to the morphological analysis "git-NEG-FUT-2SG" of the verb "gitmeyeceksin" (*You will not go*). In general, we try to permute the nodes to correspond to the inflectional morphemes' order in the chosen gloss.

Turkish and English are syntactically and morphologically different languages. In English, semantic features of language (verb tenses, articles, modifiers, prepositions) are expressed explicitly by words. On the contrary, Turkish provides the same functionality by adding suffixes (morphemes or clitics) to the stem. Hence, we translate a whole subtree (whole phrase, in most cases) as a single word in Turkish. We analyze and reflect the differences between English and Turkish under three major categories; verbals, nominals, and questions. We also apply additional heuristics to handle the uncategorized cases.

We embed pronouns acting as subjects in verbal inflections as a suffix to the root word. We also translate modals and tenses to the root verb as suffixes. We permute the English tree to reflect the Turkish constituent order and mark the English leaves as *NONE*. Hence, we place *NONE* leaves after a single word verb phrase (see Figure 1).

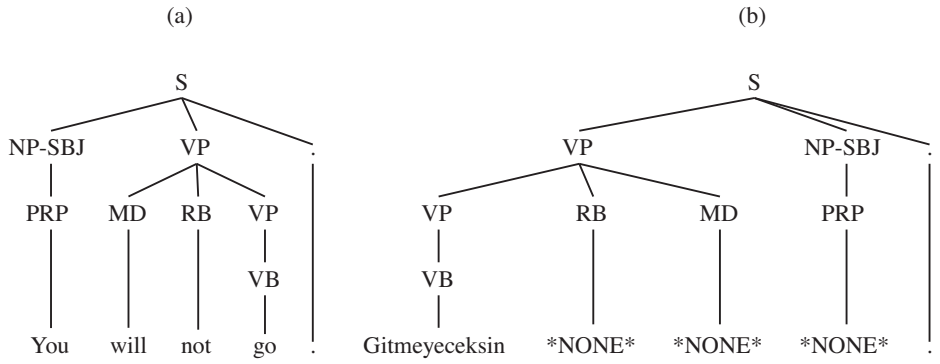

Figure 1. The permutation of the nodes and the replacement of the leaves by the glosses or *NONE*.

Tense ambiguity is a slightly frequent problem when translating tenses from English to Turkish. It creates a real challenge for annotators to find the equivalent tense in Turkish (present vs. present continuous or past vs. past perfect). To simplify the decision-making process, we choose the closest Turkish tense during the translation, which sounds neutral (see Figure 3).

We have many cases where the number agreement between subject and predicate (verbal) is quite loose. In these cases, we leave the decision to the annotator, but we translate plural nouns under the NNS tag in

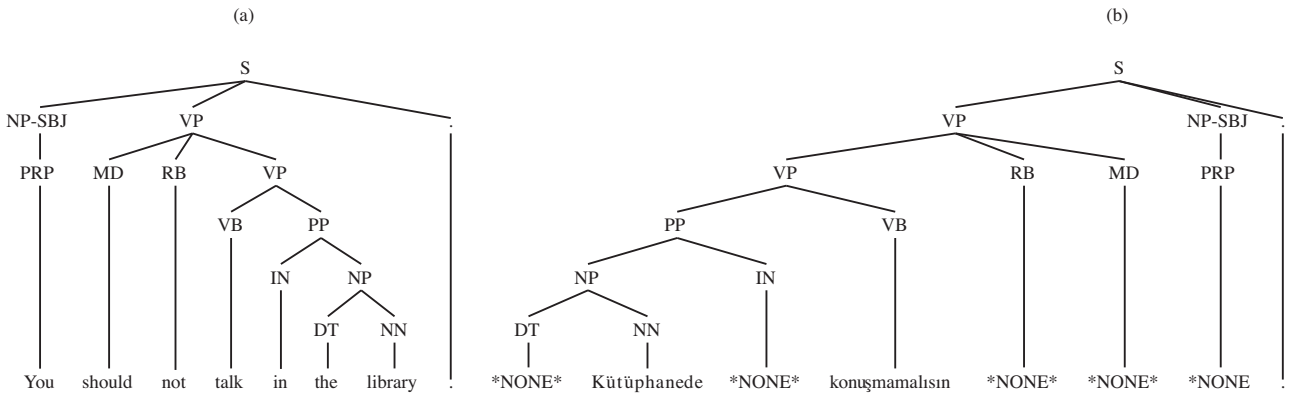

Figure 2. Original and translated trees "Marathon yüksek ham-petrol fiyat-ları-ndan faydalan-dı." (Marathon higher crude-oil price-PL benefit-PAST-3SG)

the English tree to Turkish as singular. In those cases, we keep the original POS tag NNS intact but use the singular gloss (See Figure **??**).

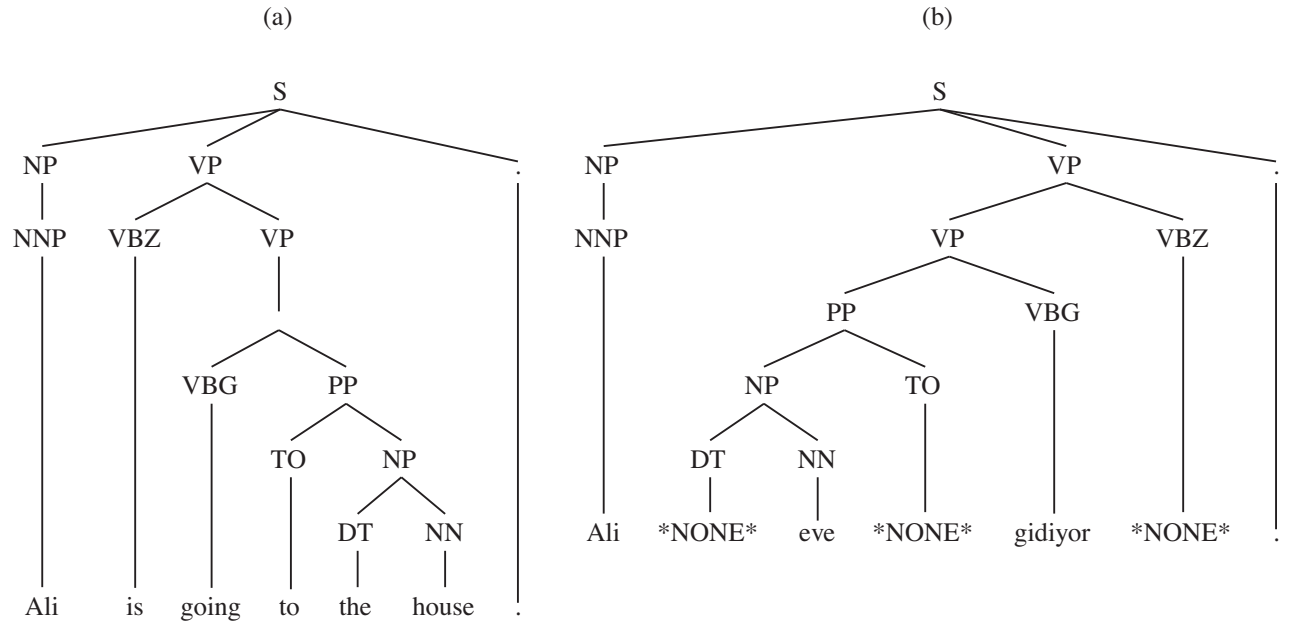

Figure 3. Original and translated trees "Marathon yüksek ham-petrol fiyat-ları-ndan faydalan-dı." (Marathon higher crude-oil price-PL benefit-PAST-3SG)

If we embed a constituent in the morphemes of a Turkish stem, we replace the English constituent leaf with *NONE*. In some cases, the personal pronouns acting as subjects are naturally embedded in the verb inflection. In those cases, pronouns in the original tree are replaced with *NONE* and its subtree is moved after the verb phrase (see Figure 1).

We use the same heuristics for noun phrases as we apply for verbal phrases. Due to the Turkish's complicated nature, case markers denote nouns and noun groups' syntactic functions; the accusative case to be used to mark the direct object of a transitive verb, or the locative case to be used to mark the head of a

prepositional phrase (see Figure 4). In translation from English to Turkish, the prepositions are marked as *NONE*, and the corresponding case marker is attached to the nominal head of the phrase.

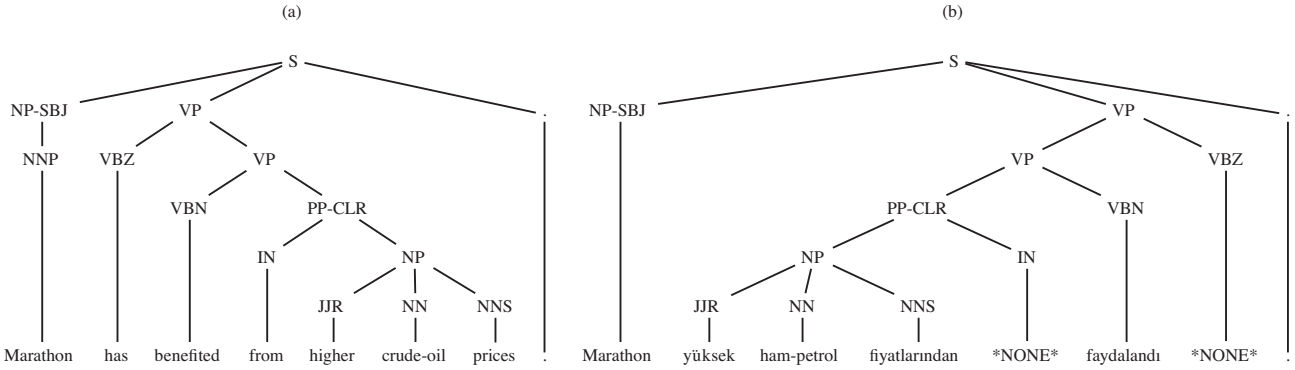

Figure 4. Original and translated trees, "kütüphane-de konuş-ma-malı-sın." (library-LOC talk-NEG-NECES-2SG).

In English syntax, one can form three types of question sentences: yes/no questions, tag questions, and WH-questions. Yes/no questions are formed by introducing an auxiliary verb ("do", "did", "have", etc.) in appropriate tense at the beginning of the sentence. Tag questions require the negative or positive confirmation of the person asked, and they are formed by adding an auxiliary verb to the end of sentences. Tag questions and Yes/no questions can be easily translated by following the heuristics introduced so far. In contrast, WH-questions introduce a corner case in the process and requires subtree movement, which is not allowed in our heuristics. WH-questions are built by introducing interrogative words such as "What" or "Where" at the beginning of the sentence. We stretch our heuristics to support this type of translation accurately. We notice that the Penn Treebank II annotation, wh- constituent and the constituent, which is subject to the question, are bound by the same numeric marker in the syntax tree, "WHADVP-1" and "*T*-1" (see Figure 5). We simply replace the wh-constituent with *NONE* and replace the corresponding leaf with the appropriate question pronoun in Turkish.

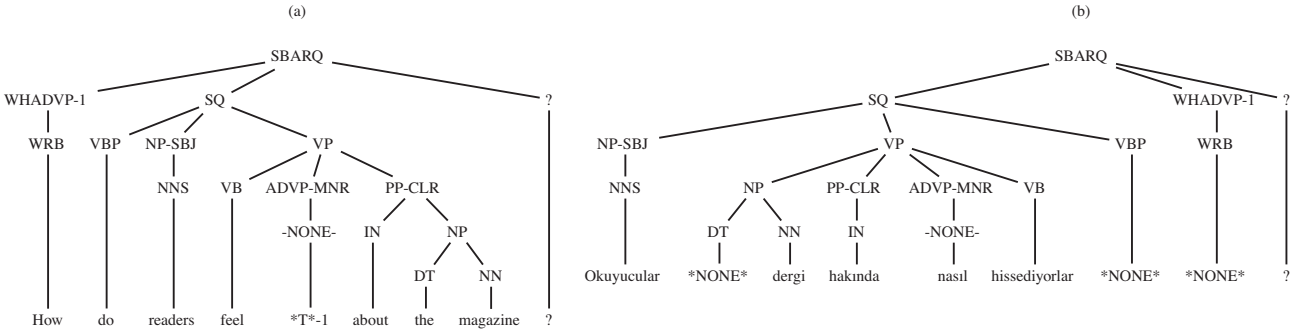

Figure 5. Original and translated trees "Okuyucu-lar dergi hakkında nasıl hisse-iyor-lar?" (reader-PL magazine about how feel-PRES-3SG?).

We also create additional heuristics to address the cases we find during the translation process:

- We drop the copula marker +dHr corresponding to the verb "to be" in most cases.
- We translate proper nouns ("London") to their Turkish equivalent ("Londra").

- We translate conjunctional words such as “IN” or “ON” as suffixes and append the appropriate morpheme to Turkish stem.
- We allow one-to-many and many-to-one as well as many-to-many translations. If multiple English words, “one and only”, correspond to a single gloss in Turkish, ”biricik”, we translate it as a single word in Turkish and mark the other leaves with *NONE*. In contrast, if a single English word matches to a multiword expression in Turkish, ”wheelchair” we translate it as a single word (tekerlekli-sandalye) in Turkish.
- We drop the definite article ”the” in English during translation, and mark the leaf node with a *NONE*, in most cases. Depending on the context, we append the appropriate demonstrative adjective in Turkish (see Figure 4). We follow the same principle for indefinite articles such as ”a” and ”an”.

4.2. Morphological processing

In Phase-1, we apply our heuristics to obtain the correct gloss replacement and ensure that we have glosses in correct order in Turkish. Translated glosses contain morphemes and morphemes are translation for English glosses. These morphemes are kept in their surface forms, but the valuable morphological information is used to enrich the translated trees. This process is achieved by following two steps: automatic morphological analysis (automatic), and morphological disambiguation (semiautomatic). Both steps are integrated into our translation tool. We perform morphological analysis using as FST based morphological analyzer [**?**] for each token in the translated tree. Next, we feed these morphological analyses into a statistical morphological disambiguator [44], and result is used as correct morphological analysis for this token. Otherwise, human annotators select the correct analysis out of all candidate analyses.

4.3. Filling the gaps: morphological enrichment

In Phase-3, we have the translated tree having the correct gloss ordering, words analyzed and annotated with morphological tags (see Figure 6a). We detach the individual morphological tags, e.g., NEG, ABL, analyzed in Phase 2. Meanwhile, these morphological tags are transformed to their canonical equivalents, for example, NEG to +mA or ABL to +DAn. The annotator sees the morphemes in their canonical forms during translation since they find the canonical representation more intuitive. Then, they move these tags to the *NONE* glosses based on the morphotactics of Turkish (see Figure 6b).

Even though, we move these canonical pieces to the right spot; some morphemes do not have any visible canonical forms. In practice, we observe that annotators need to go back and forth to revisit their decisions between phases. This behavior is noticed when they end up with too many *NONE* leaves and, of course, with many morphological gaps between the original and the translated tree. Table 2 shows the total number of *NONE* leaves by POS tags for 10K sentences in which enrichment is completed, currently. As statistics show, the majority of the *NONE* leaves are for Determiner (DT) and Preposition or subordinating conjunction (IN) tags. We basically ignore these and leave them as *NONE* except for some exceptional cases where we feel the translation sounds more natural.

Even though the coverage of *NONE* leaves are quite good, there are still morphological units that the annotators are not able to match with any suitable gloss in English. Hence, those morphological features stay as suffixes attached to the root word. This morphological gap is the result of using the English syntactic tree as the base for Turkish tree. We also do not introduce any structural changes in the Turkish tree such as subtree movement.

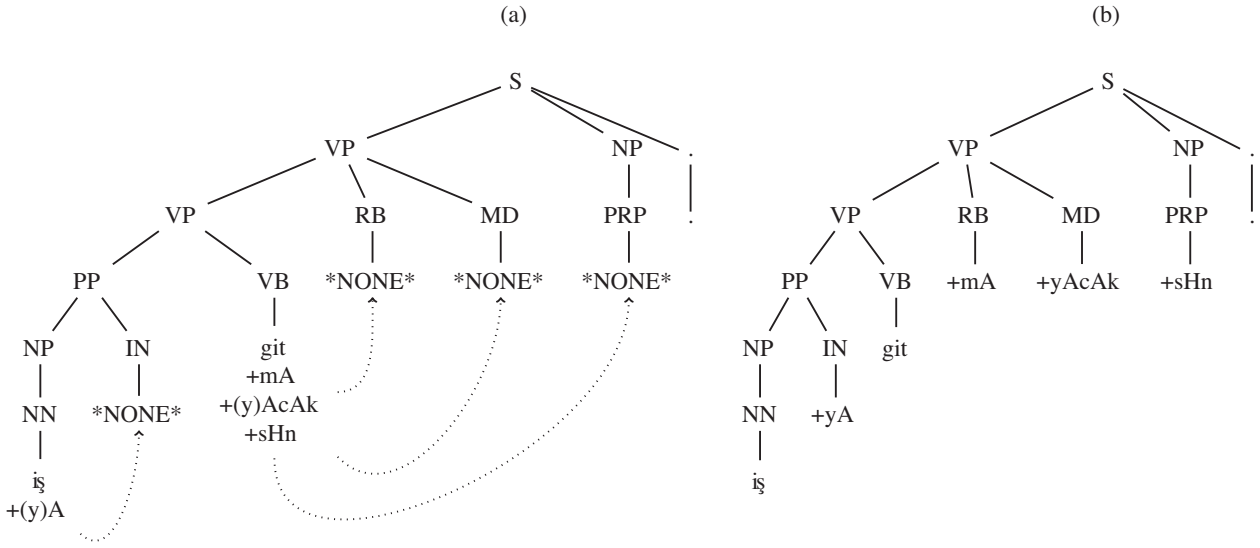

Figure 6. Morphological features extracted in Phase-2 and movement of morphemes to empty slots in Phase-3.

Table 2 shows the top 3 Turkish morphemes for each POS tag in the corpus with their occurences. As our transformation heuristic suggests, Determiner (DT) and Existential there (EX) are not translated. There is no *NONE* movement for nouns (NN, NNS, NNP or NNPS) as expected. Preposition and subordinating conjunction (IN) category is mapped by nouns and numerals (CD) with Case markers. Verbal phrases (VB, VBN, VBP, VBD, VBG, VBZ) have complex morphotactics and they mostly require more than one *NONE* tags to be replaced by a Turkish morpheme. Statistics for Present participle (VBP) and gerunds (VBG) reveal the tense ambiguity issue which is described in our heuristics. TO tag is mapped with both base verbal forms (VB, +mAk) and noun gloss as case markers.

5. Building a closed-domain treebank

We apply our transformation heuristics in technical domain (telecommunication), and build an English-Turkish parallel treebank, namely Nokia Treebank (NTB). The aim of the whole work is to test the applicability of our heuristics in closed-domain. Morover, we created a pretranslation tool to perform technical document translation. We also created a domain specific vocabulary for further NLP studies.

Technical document translation is a closed-domain translation task, and helps companies to save work force, time and cash. Unlike open-domain translation tasks, in technical document translation, sentences have simple present tense and slightly less complex syntactic and morphological structures. However, treebank creation process requires extended domain specific vocabulary. Vocabulary has a crucial role in closed-domain treebank creation and needs supervision from domain experts to keep the manual translation efforts intact. We gradually obtain this vocabulary during the translation phase.

In NTB, we translate 8327 sentences in varying lengths. Table 3 shows the number of sentences in different lenghts. Sentences are extracted from noncustomer technical documentations. These flat sentences are converted to Penn Treebank style (bracketed) syntactic trees. We employ Stanford Parser to perform the conversion. Stanford Parser is a Java based collection of probabilistic natural language parsers supporting both highly optimized PCFG and lexicalized dependency parser. Stanford Parser supports English, German, French,

Table 2. Top-3 frequent Turkish morphemes with occurences of *NONE* tuples, and total number of *NONE* counts for each POS tag.

POS tag	Morpheme and frequency	*NONE* count
DT	+sH (1)	4783
EX	N/A	110
IN	+DA (878), +nHn (735), +DAn (420)	4736
JJ	+yAbil (1), +yAmA (1), +yAmA+Hyor (1)	58
MD	+yAcAk (317), +yAbil+Hr (192), +yAcAk+DH (39)	1040
POS	+nHn (469), +Hn (12), +DAn (6)	701
PRP	+lAr (116), +yHm (80), +yHz (65)	1598
PRP$	+sH (100), +sH+nH (81), +lArH+nH (46)	562
RB	+mA (389), +yAmA (52), +yAmA+DH (15)	871
TO	+yA (715), +nA (109), +mAk (75)	1578
VB	+Hl (57), +n (10), +DH (4)	233
VBD	+yDH (539), +DH (270), +mHs+yDH (27)	1060
VBG	+yAcAk (13), +Hyor (7), +Hl (4)	55
VBN	+mHs (18), +Hyor (13), +Hl (12)	148
VBP	+DHr (179), +Hyor (169),+DH (105)	1027
VBZ	+DHr (643), +Hyor (198), +DH (156)	1701
WDT	+yA (22), +yHncA (1)	77
WP	+yAn (13), +SH (3), +nA (1)	112

Table 3. Number of sentences in Nokia Treebank based on the sentence length.

	≤ 15	≤ 20	≤ 25	≤ 30	≤ 35	≤ 40	≤ 45	≤ 50	> 50
No. of sent.	4928	1476	881	539	274	134	60	19	16

Spanish, Chinese, and Arabic languages with a variety of parser options. For English, collection also offers factored model, and recurrent neural network based syntactic parsing, and based on the selected model, parser produces slightly different output. Hence, it requires some level of verification, even if the given sentences are in English, and they are not syntactically complex. We use PCFG Parser [33] for syntactic parser which yields better accuracy in the tagging of OOV (out of vocabulary) tokens, where parsers tend to mark the gloss as NNP. A sample sentence from NTB with its Turkish transformed tree is illustrated in Figure 7.

PTB and NTB show some structural differences and gaps in terms of tagging. First, in NTB, we do not seek complete sentences with proper SOV order, and use phrase-like sentences to achive a good vocabulary coverage. Second, sentences are generally in base form (VB), simple present forms (VBP, VBZ) or are presented as Copular (+DHr) form. Most importantly, Phase-3 is almost not needed, since only a minority of sentences require morphological enrichment.

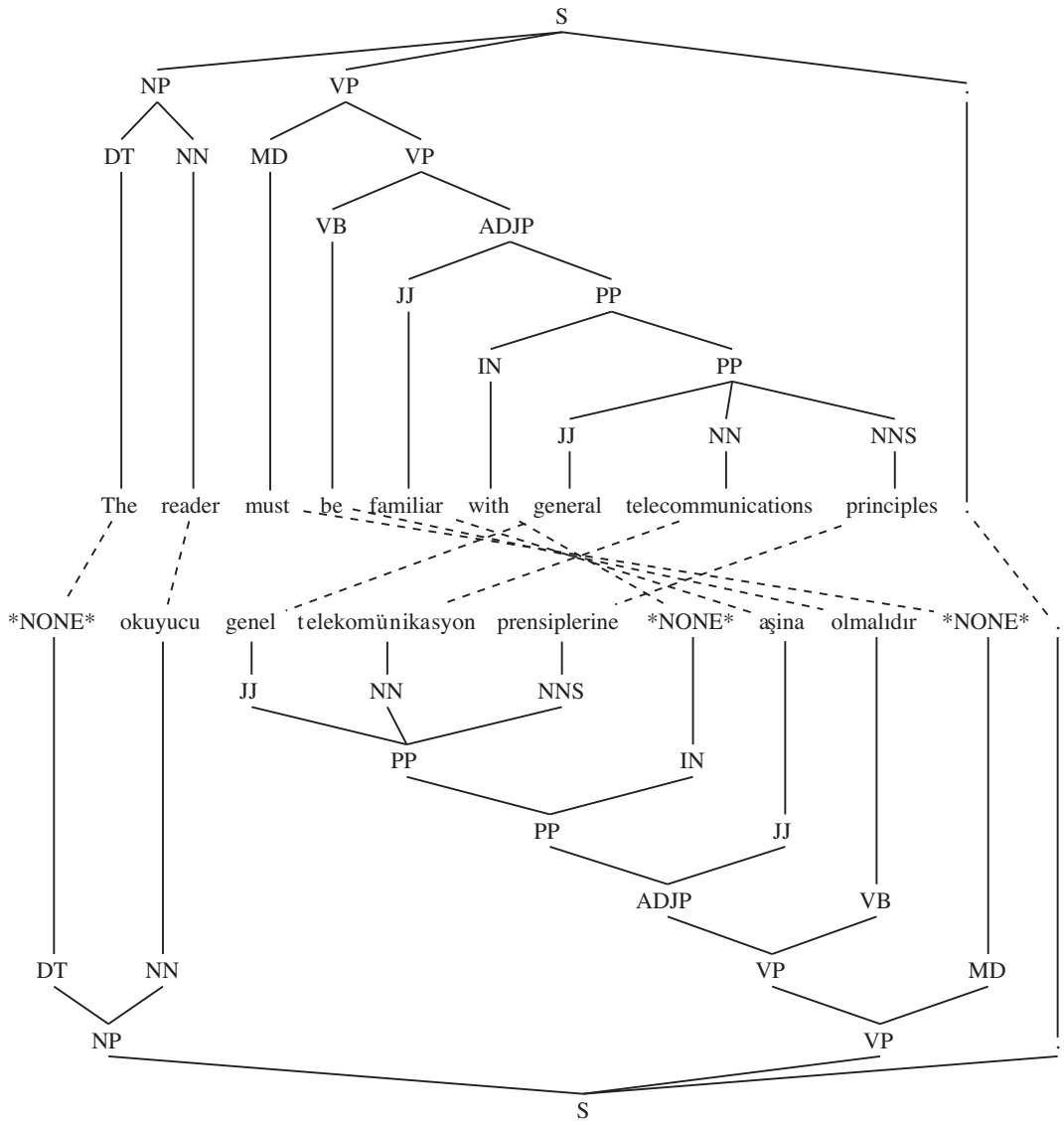

Figure 7. Sample sentence from NTB and its Turkish equivalent.

6. Corpus evaluation and results

Parallel treebank data validation is achieved by creating a language model using the target language and evaluate the performance of language model through validation metrics. In the literature, perplexity is the most commonly used evaluation metric [34]. Perplexity is an *n-gram* based intrinsic evaluation metric that assigns probabilities to words or sentences [35]. We utilize perplexity and measure our treebank quality on an unseen data .

Perplexity as predictive likelihood is considered to be anticorrelated to human-judgement [36], and not a definite way of determining the usefulness of a language model. However, in the absence of efficient means to evaluate a language model, perplexity is a useful metric for comparing language models. Perplexity PP_{p_M} is

considered as the inverse probability of the test set $T = \{w_1, ..., w_n\}$, normalized by the word counts (1).

$$PP_{p_M} = \sqrt[N]{\prod_{i=1}^{N} \frac{1}{P_M(w_i|w_1, ..., w_{i-1})}} \tag{1}$$

In order to deal with out of vocabulary (OOV) words and avoid zero probability issues, we apply *n-gram* smoothing techniques. We apply Kneser-Ney [37] interpolated smooting technique. Kneser-Ney is considered to be the most effective smoothing technique in the literature. The algorithm omits lower *n-gram* frequencies by interpolating the higher and the lower order language models.

We evaluate the perlexity scores of both parallel treebanks. First, we flatten our translated trees in surface forms without any morphological and syntactical annotation. Then, we flatten all trees to sentences. We apply k-fold (*k=10*) cross validation to generate train and test sets. Next, we train the language model by the i^{th}-fold train set for different *n-gram* counts and test the model on the i^{th}-fold test set. k-fold cross validated perplexity scores for both parallel treebanks are given in Table 4 and are plotted (see Figure 8).

Table 4. Perplexity scores of TPTB and NTB train-test data with smoothing.

	TPTB		NTB	
	Train	Test	Train	Test
1-gram	2299.24	1503.61	1395.39	1227.50
2-gram	37.54	570.12	39.58	585.39
3-gram	17.90	539.28	8.10	524.53
4-gram	13.36	509.64	4.82	507.01
5-gram	11.94	508.74	4.14	273.74

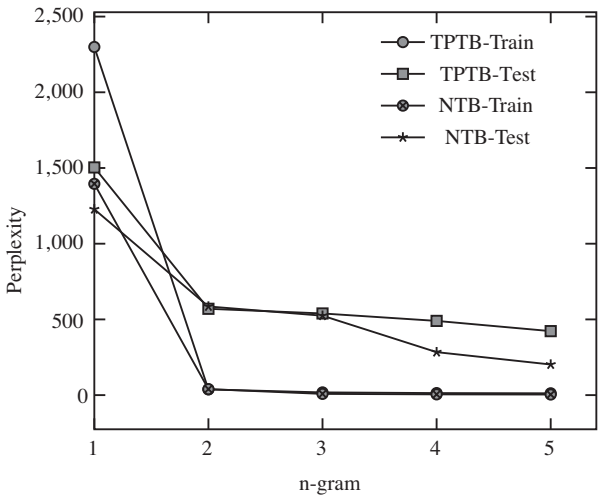

Figure 8. Evaluation of TPTB and NTB train-test data through perplexity analysis.

As illustrated in Figure 8, perplexity scores for both treebanks decrease as the number of *n-grams* increases as expected. The best result is achieved using a 5-gram language model in both treebanks. NTB is a closed-domain parallel treebank and outperforms TPTB. Word-based accuracy is affected by the domain vocabulary,

and closed-domain treebanks benefit from it, especially in multiword NNP structures. In terms of unique words, NTB has 127K number of tokens, even though only 10% of it is unique. In contrast, uniqueness level in TPTB is 20% out of 230K tokens. Since we use the surface form glosses, perplexity is higher in TPTB. From a morphotactics point of view, TPTB has higher complexity in verbal structures than NTB in terms of morpheme variations due to complex morphotactics.

We employ our methodology to create different corpora and apply intrinsic analysis for TPTB in machine translation task using statistical approach with limited data. We also utilize NTB in translation tasks. We follow the statistical approach presented in [39]. Moreover, we introduce the semiautomatically generated domain specific corpus to the translation model. The dictionary was extracted from parallel corpus automatically and improved during the consecutive iterations. According to the experiments, our approach with 10-fold cross validation yields a $26.83 \pm .03$ BLEU score in NTB, the best we have obtained so far in machine translation.

7. Conclusion

In this paper, we have presented our efforts towards extending our English-Turkish treebank by (i) increasing number of sentences in the corpus, (ii) morphologically and syntactically annotating the corpus to fill the gaps between the language pair, (iii) appyling our corpus construction methodology to closed-domain data, and (iv) evaluating the applicability of corpus construction strategy and the resulting corpus intrinsically and extrinsically.

English is not syntactically and morphologically rich. Translation tasks between languages such as Turkish and English is hard to tackle. Lack of large and high quality data sets that emphasize languages' syntactic features is the main reason for failure. We start building TPTB to fill that gap especially in statistical machine translation tasks.

Currently, TPTB has 17K parallel sentences and NTB has 8327 parallel sentences. However, NTB translation efforts do not continue, since the parent project has ended by the end of 2017. However, we still work on improvements for the current NTB data set. In TPTB, we still provide improvements in translation quality and morphological enrichment. Moreover, we provide different levels of annotation to support different NLP tasks. As future work, we plan to expand morphological enrichment and increase the treebank size.

As future work, we plan to invest in intrinsic and extrinsic evaluation, and toolset development in different ways. For intrinsic evaluation, we currently measure the performance of both treebanks by word-based language perplexity. This study shows that as the corpus size grows, the number of low-frequency words also increase in Turkish. In morphologically rich languages, morpheme plays important role and they are treated as immediate glosses. Subword dependencies between stems and suffixes are needed to reflect sytactic dependencies in language models [40]. Our goal in applying morphological embedding is to solve the data sparsity problem by reducing low frequency words. Building and integrating stem and morpheme-based language models will offer a valuable asset, and better translation results as well.

For extrinsic evaluation, we plan to employ our expanded treebank on translation tasks. We utilize TPTB in statistical machine translation tasks by utilizing simple log likelihood approach. Our approach works fine to detect the correct tree permutations. However, we are not able to replace the correct gloss in translated tree, even though we have the domain specific dictionary. In a closed-domain, we can achieve better BLEU scores, if we have a good dictionary coverage. This leads us to invest for expanding the domain dictionary, at all.

Acknowledgments

Nokia Parallel Treebank (NTB) creation efforts presented in this paper is supported by the Scientific and Technological Research Council of Turkey (TÜBİTAK) TEYDEB (Technology and Innovation Grant Programs Directorate) 1501 Industrial R&D Projects Grant Program (3140986). We thank Oğuzhan Kuyrukçu, Arife Betül Yenice, Büşra Marşan, Aslı Kuzgun, Ezgi Sanıyar, and Neslihan Cesur for their valuable support.

References

[1] Chomsky N. Syntactic Structures. The Hague: Mouton and Co., 1957.

[2] Marcus M, Marcinkiewicz M, Santorini B. Building a large annotated corpus of English: the penn treebank. Computational Linguistics 1993; 19 (2): 313-330. doi: 10.21236/ada273556

[3] Brants S, Dipper S, Hansen S, Lezius W, Smith G. The TIGER treebank. In: Workshop on treebanks and linguistic theories; Sozopol, Bulgaria; 2002. pp. 24-41.

[4] Abeillé A, Clément L, Kinyon A. Building a treebank for French. In: Second International Conference on Language Resources and Evaluation (LREC 2000); Athens, Greece; 2000. pp. 165-187

[5] Haverinen K, Nyblom J, Viljanen T, Laippala V, Kohonen S et al. Building the essential resources for Finnish: the Turku Dependency Treebank. Language Resources and Evaluation 2014; 48 (3): 493–531. doi: 10.1007/s10579-013-9244-1

[6] Csendes D, Csirik J, Gyimóthy T, Kocsor A. The Szeged Treebank. In: Text, Speech and Dialogue, 8th International Conference (TSD 2005); Karlovy Vary, Czech Republic. pp. 123-131

[7] Maamouri M, Bies A, Buckwalter T, Mekki W. The penn Arabic treebank: Building a large-scale annotated Arabic corpus. In: NEMLAR Conference on Arabic Language Resources and Tools; Cairo, Egypt; 2004. pp. 102-109.

[8] Kornfilt J. Turkish (Descriptive Grammars). London, UK: Routledge, 1997.

[9] Xue N, Xia F, Chiou F-D, Palmer M. The Penn Chinese TreeBank: Phrase structure annotation of a large corpus. Natural Language Engineering 2005; 11 (2): 207–238. doi: 10.1017/S135132490400364X

[10] Koehn P. Europarl: A multilingual corpus for evaluation of machine translation. Tenth Machine Translation Summit; Phuket, Thailand; 2005. pp. 79-86.

[11] Cyrus L, Feddes H, Schumacher F. FuSe - A Multi-Layered Parallel Treebank. In: Second Workshop on Treebanks and Linguistic Theories; Växjö, Sweden; 2003. pp. 213-216.

[12] Ahrenberg L. LinES: An English-Swedish Parallel Treebank. In: 16th Nordic Conference of Computational Linguistics (NODALIDA 2007); Tartu, Estonia; 2007. pp. 270–273.

[13] Gustafson-Čapková S, Samuelsson Y, Volk M. SMULTRON (version 1.0) - The Stockholm MUltilingual parallel TReebank. An English-German-Swedish parallel Treebank with subsentential alignment; 2007.

[14] Čmejrek M, Cuřín J, Havelka J, Hajič J, Kuboň V. Prague Czech-English dependency treebank: Syntactically annotated resources for machine translation. In: Fourth International Conference on Language Resources and Evaluation (LREC 2004); Lisbon, Portugal; 2004. pp. 1597-1600

[15] Yeniterzi R, Oflazer K. Syntax-tomorphology mapping in factored phrase-based statistical machine translation from English to Turkish. In: 48th Annual Meeting of the Association for Computational Linguistics; Stroudsburg, PA, USA; 2010. pp.454-464.

[16] El-Kahlout ID. Statistical machine translation from English to Turkish. PhD, Sabanci University, Istanbul, Turkey, 2009.

[17] Yıldız OT, Solak E, Görgün O, Ehsani R. Constructing a Turkish-English parallel treebank. In: 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers); Baltimore, MD, USA; 2014. pp. 112-117.

[18] Atalay NB, Oflazer K, Say B. The annotation process in the Turkish treebank. In: 4th International Workshop on Linguistically Interpreted Corpora (LINC 2003) at EACL 2003; Budapest, Hungary; 2003. pp. 33-38.

[19] Eryiğit G, Oflazer K. Statistical dependency parsing of Turkish. In: 11th Conference of the European Chapter of the Association for Computational Linguistics(EACL); Trento, Italy; 2006. pp. 89–96.

[20] Yüret D. Dependency parsing as a classification problem. In: Tenth Conference on Computational Natural Language Learning (CoNLL-X); New York City, NY, USA; 2006. pp. 246-250.

[21] Riedel S, Çakıcı R, Meza-Ruiz I. Multi-lingual dependency parsing with incremental integer linear programming. In: Tenth Conference on Computational Natural Language Learning (CoNLL-X); New York City, NY, USA; 2006. pp. 226-230.

[22] Çakıcı R, Baldridge J. Projective and non-projective Turkish parsing. In: Conference on Treebanks and Linguistic Theories (TLT 2006); Prague, Czech Republic; 2006. pp. 19-30.

[23] Eryiğit G, Adalı E, Oflazer K. Türkçe cümlelerin kural tabanlı bağlılık analizi. In: 15th Turkish Symposium on Artificial Intelligence and Neural Networks (TAINN 2006); Muğla, Turkey; 2006. pp. 17–24 (in Turkish).

[24] Eryiğit G, Nivre J, Oflazer K. Dependency parsing of Turkish. Computational Linguistics 2008; 34 (3): 357-389. doi:10.1162/coli.2008.07-017-R1-06-83

[25] Şahin G, Adalı E. Annotation of semantic roles for the Turkish Proposition Bank. Language Resources and Evaluation 2018; 52 (3): pp. 673–706. doi: 10.1007/s10579-017-9390-y

[26] Sulubacak U, Pamay T, Eryiğit G. IMST: revisited Turkish dependency treebank. In: TurCLing 2016, The First International Conference on Turkic Computational Linguistics at CICLING; Konya, Turkey; 2016. pp. 1-6.

[27] Çakıcı R. Automatic induction of a CCG grammar for Turkish. In: ACL Student Research Workshop; Ann Harbor, MI, USA; 2005. pp. 73-78.

[28] Çetinoğlu Ö, Oflazer K. Morphology-syntax interface for Turkish LFG. In: 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics; Sydney, Australia; 2006. pp.153-160.

[29] Çetinoğlu Ö, Oflazer K. Integrating derivational morphology into syntax. In: Recent Advances in Natural Language Processing (RNLP 2007); Borovets, Bulgaria; 2007. pp. 155-170.

[30] Megyesi B, Dahlqvist B, Pettersson E, Nivre J. Swedish-Turkish parallel treebank. In: Sixth International Conference on Language Resources and Evaluation (LREC 2008); Marrakech, Morocoo; 2008. pp.470-473.

[31] Megyesi B, Dahlqvist B, Csató É, Nivre J. The English-Swedish-Turkish parallel treebank. In: Seventh International Conference on Language Resources and Evaluation (LREC 2010); Valletta, Malta; 2010. pp. 3393-3397.

[32] Nivre J, Hall J, Nilsson J. MaltParser: A data-driven parser-generator for dependency parsing. In: Fifth International Conference on Language Resources and Evaluation; Genoa, Italy; 2006. pp. 2216-2219.

[33] Nivre J, Nilsson J, Hall J. Talbanken05: A Swedish treebank with phrase structure and dependency annotation. In: Fifth International Conference on Language Resources and Evaluation (LREC 2006); Genoa, Italy; 2006. pp.1392-1395.

[34] Sulger S, Butt M, King TH, Meurer P, Laczko T et al. ParGramBank: The ParGram parallel treebank. In: 51st Annual Meeting of the Association for Computational Linguistics; Sofia, Bulgaria; 2013. pp. 550-560.

[35] Erguvanlı ET. The Function of Word Order in Turkish Grammar. Berkeley, CA, USA: University of California Press, 1984.

[36] Dryer MS. The Greenbergian Word Order Correlations Language 1992; 68 (1): 81-138. doi: 10.2307/416370

[37] Yıldız OT, Çandır S, Solak E, Ehsani R, Görgün O. Constructing a Turkish constituency parse treeBank. In: International Conference on Computer and Information Sciences (ISCIS); London, UK; 2015. pp. 339-347.

[38] Yıldız OT, Avar B, Ercan G. An open, extendible, and fast Turkish morphological analyzer. In: International Conference on Recent Advances in Natural Language Processing (RANLP 2019); Varna, Bulgaria. pp. 1364–1372.

[39] Görgün O, Yıldız OT. A novel approach to morphological disambiguation for Turkish. In: Computer and Information Sciences II - 26th International Symposium on Computer and Information Sciences; London, UK; 2011. pp. 77-83.

[40] Klein D, Manning C. Accurate unlexicalized parsing. In: 41st Annual Meeting of the Association for Computational Linguistics; Morristown, NJ, USA; 2003. pp. 423-430.

[41] Chen S, Beeferman D, Rosenfeld R. Evaluation metrics for language models. In: Proceedings of DARPA Broadcast News Transcription and Understanding Workshop; Lansdowne, VA, USA; 1998. pp. 275-280.

[42] Jurafsky D, Martin JH. Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition 2nd ed. Upper Saddle River, NJ: Prentice Hall, 2009.

[43] Chang J, Boyd-Graber J, Wang C, Gerrish S, Blei DM. Reading tea leaves: How humans interpret topic models. In: Advances in Neural Information Processing Systems 21 (NIPS 2009); Vancouver, Canada; 2009. pp. 288-296.

[44] Kneser R, Ney H. Improved backing-off for m-gram language modeling. In: International Conference on Acoustics, Speech, and Signal Processing; Detroit, MI, USA; 1995. pp.181–184.

[45] Görgün O, Yıldız OT, Solak E, Ehsani R. English-Turkish parallel treebank with morphological annotations and its use in tree-based smt. In: 5th International Conference on Pattern Recognition and Methods (ICPRAM); Rome, Italy; 2016. pp. 510-516.

[46] Yüret D, Biçici E. Modeling morphologically rich languages using split words and unstructured dependencies. In: ACL-IJCNLP 2009 Conference Short Papers; Suntec, Singapore; 2009. pp. 345-348.