Dependency Parsing of Turkish

Gülşen Eryiğit, Joakim Nivre, and Kemal Oflazer

1. Introduction

Concerns

- Are models and algorithms tailored to properties of specific language groups?
- Are different kinds of syntactic representations suitable for different languages?

Why Turkish?

- Turkish is a language characterized by agglutinative morphology, free constituent order, and predominantly head-final structures.

- Shares these characteristics with languages such as Basque, Estonian, Finnish, Hungarian, Japanese and Korean.

2. Turkish Morphology and Dependency Relations

Morphological Structure

- Words are split into inflectional groups (IG).
- The root and derivational elements are represented by different IGs separated by derivational boundaries.

arabanızdaydı

(‘it was in your car’)

arabanızda DB *ydı*

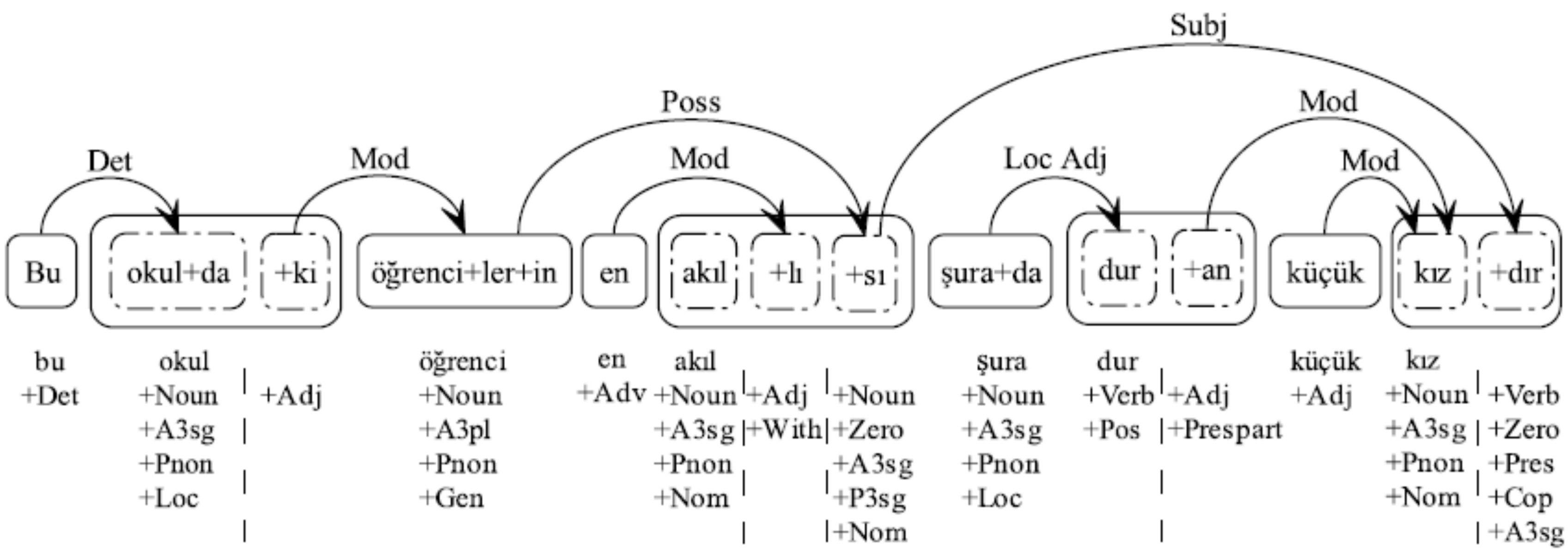
$\underbrace{\textit{araba}\text{+Noun+A3sg+P2pl+Loc}}_{IG_1}$ DB $\underbrace{\text{+Verb+Zero+Past+A3sg}}_{IG_2}$

‘in your car’ ‘it was’

Dependency Relations

- A parser has to show that one word is a dependent of another and state which IGs of the words in question are involved in the syntactic relation.

- Dependency links emanate from the last IG of a word which determines the role of the word as a dependent, and land on one of the IGs of a head word.



This school-at+that-is student-s-' most intelligence+with+of there stand+ing little girl+is

The most intelligent of the students in this school is the little girl standing there.

Turkish Treebank

- The METU-Sabancı Turkish Treebank comprises 5635 sentences.
- Although the number of sentences in the treebank is comparable to that of other available treebanks, the number of words is considerably smaller.
- The average sentence has 8.6 words.

Morphological Disambiguation

- Assigns the main POS category and correct morphological information.
- The number of potential tag combinations in Turkish is very large.

  e.g. surface form: ***kalemi***

    kale +Noun+A3sg+P1sg+Acc ('my castle' in accusative form)

    kalem +Noun+A3sg+P3sg+Nom ('his pencil')

    kalem +Noun+A3sg+Pnon+Acc ('the pencil' in accusative form)

- Disambiguation involves choosing one of the morphological analyses.

- The accuracy of the morphological disambiguator is 88.4%, including punctuation and using a lookup table for words not recognized by the morphological analyzer.

- Errors in POS categories can prevent the parser from finding the correct head.

3. Dependency Parsers

Parsers Tested

- Baseline parsers:
  - Two naive parsers linking dependents to an IG in the next word
  - A rule based parser

- Data-driven parsers:
  - A probabilistic parser
  - A classifier-based parser

- Data sets:
  - Experiments were carried out on the entire treebank. The treebank was divided into 10 sets: 9 were used for training and 1 for testing.

- Evaluation metrics:
  - AS_U = unlabeled attachment score
  - AS_L = labeled attachment score
  - WW_U = word to word score

3.1 Baseline Parsers

- Parser 1:
  - Attaches the last IG of a word to the first IG of the next word.

- Parser 2:
  - Attaches the last IG of a word to the last IG of the next word.

- Parser 3:
  - Uses a linear-time algorithm to derive a dependency graph in one left-to-right pass over the input. The next parsing action is determined according to 31 predefined hand-written rules.

3.2 Probabilistic Parser

- This approach takes a morphologically tagged and disambiguated sentence as input, and outputs the most probable dependency tree based on the probabilities computed with the training data.

- This approach consists of:
  1) A parsing algorithm
  2) A conditional probability model
  3) Maximum likelihood estimation

Methodology

- Assign a probability to each candidate dependency link based on frequencies computed during training, and find the most probable dependency tree.
- The probability of a tree is the product of the dependency links it contains.

$$T^* = \underset{T}{\operatorname{argmax}}\, P(T|S) = \underset{T}{\operatorname{argmax}} \prod_{i=1}^{n-1} P(dep\,(u_i, u_{\mathcal{H}(i)})\,|\,S)$$

Backward Beam Search Parsing

- Parses a sentence starting from the end, and tries to link dependents to a unit to the right at each step.
- A beam keeps track of the most probable structures.
- Head-initial dependencies are handled using three predefined lexicalized rules to construct the links.

Probability Model

- The probability of a dependency link linking u_i to u_{Hi} is approximated with the product of the probability of seeing the same dependency within a similar context and the probability of seeing the dependent linking to some head some distance away.

- Data sparseness is dealt with by interpolating other estimates while calculating the above probabilities.

Additional Parameters

- The parser is given the following parameters:
  - the number of left and right neighbours of the dependent (D_l,D_r),
  - the number of left and right neighbours of the head (H_l,H_r),
  - the size of the beam (beamsize) set to 3,
  - the distance threshold value set to 6.

Parsing Units and Experiments

- **Word-based model 1:** uses actual words as parsing units and each word is represented by a concatenation of its inner IGs.
- **Word-based model 2:** uses actual words as parsing units and each word is represented by its final IG.
- **IG-based model:** uses IGs as parsing units.

Experimental Results

- The performance of the word-based models is lower than the rule-based baseline parser.
- The IG-based parser outperforms all other models: it recovers the relations between correct IGs and finds the correct head word.
- Running experiments on the IG model with different morphological features for the IG representations does not improve performance.

Unlabeled attachment scores and unlabeled word-to-word scores for the probabilistic parser.

Parsing Model	**Parameters**	**AS_U**	**WW_U**
Word-based model 1	(D_l=1, D_r=1, H_l=1, H_r=1)	68.1±0.4	77.1±0.7
Word-based model 2	(D_l=1, D_r=1, H_l=1, H_r=1)	68.3±0.3	77.6±0.5
IG-based model	(D_l=1, D_r=1, H_l=0, H_r=1)	72.1±0.3	79.0±0.7

3.3 Classifier-based Parser

- This approach has achieved high accuracy results across different languages. It does not employ a grammar, but relies solely on inductive learning from the treebank to analyze new sentences, and on deterministic parsing to disambiguate.

- This approach consists of:
  1) A deterministic parsing algorithm
  2) A history-based model
  3) Discriminative classifiers

Methodology

- Use a deterministic linear-time algorithm to derive labeled dependency graphs in one left-to-right pass over the input, where a stack σ stores partially processed tokens and a list τ stores the remaining tokens.
- This algorithm is restricted to projective dependency graphs.

Linear-time Parsing

- The parser is initialized with an empty stack and all sentence tokens in the input list. Target tokens σ_0 and τ_0 are candidates for a dependency relation.

- Parsing actions:
  - **Shift:** Push the next token onto the stack.
  - **Left-Arc$_r$:** Add a dependency arc *r* from the next token to the top token. Pop the stack.
  - **Right-Arc$_r$:** Add a dependency arc *r* from the top token to the next token. Replace next token by top token in the input list.

History-based Model

- Token histories are represented as feature vectors, where the features are based on the target tokens, the neighbouring tokens, or the dependent and head tokens.
- Available features: lexical form (root), part-of-speech (POS), inflections (INF), dependency type to the head (DEP).
- Support vector machines (SVM) predict the parser’s actions from histories.

Parsing Units and Experiments

- **Word-based model:** each word is a concatenation of its IGs.
- **IG-based model:** each unit is an IG.
- **Feature model 1:** use unlexicalized features with only the minor POS and DEP features for comparison with the probabilistic parser.

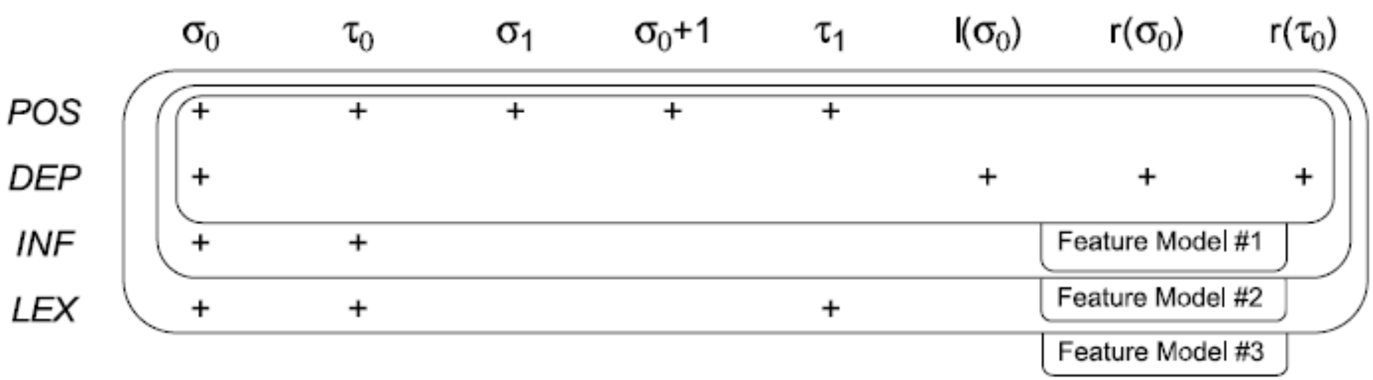
Experimental Results

- The IG-based model outperforms the word-based model.
- The AS_U scores are not better than the scores obtained from the probabilistic parser.

Parsing Model	$\mathbf{AS_U}$	$\mathbf{AS_L}$
Word-based	67.1±0.3	57.8±0.3
IG-based	70.6±0.2	60.9±0.3

Improvements

- **Feature model 2:** adds inflectional features to Feature model 1.
- **Feature model 3:** lexicalizes Feature model 2, first using the root information and then using the complete surface form as lexical features.



Unlabeled and labeled attachment scores for enhancements of the IG-based model

Feature Model	**AS_U**	**AS_L**
Feature Model 2	72.4±0.2	63.1±0.3
Feature Model 3 (roots)	76.0±0.2	67.0±0.3
Feature Model 3 (surface forms)	75.7±0.2	66.6±0.3

4. Inflectional Features, Lexicalization, and Training Set Size

Inflectional Features

- The features with the greatest impact were case and possession; and number/person agreement.

- Labeled accuracy is more affected by the usage of inflectional features.

- Inflectional features are crucial towards determining the type of relationship between dependent and head units.

Lexicalization

- Lexicalization only improves the performance of the classifier-based parser.

- Lexicalizing IGs from different parts-of-speech categories does not produce uniform results. Only lexicalizing conjunctions and nouns has an impact on accuracy.

Training Set Size

- The classifier-based lexicalized model shows the most improvement with increased training sets.
- The probabilistic model is less affected by the size of the training data, i.e. cannot be improved by simply increasing the size of the data.

5. Error Analysis

Attachment score (AS_U), labeled precision (P), labeled recall (R) and labeled F-score for each dependency type in the treebank.

Label	*n*	*dist*	AS_U	P	R	F
SENTENCE	7,252	1.5	90.5	87.4	89.2	88.3
DETERMINER	1,952	1.3	90.0	84.6	85.3	85.0
QUESTION.PARTICLE	288	1.3	86.1	80.0	76.4	78.2
INTENSIFIER	903	1.2	85.9	80.7	80.3	80.5
RELATIVIZER	85	1.2	84.7	56.6	50.6	53.4
CLASSIFIER	2,048	1.2	83.7	74.6	71.7	73.1
POSSESSOR	1,516	1.9	79.4	81.6	73.6	77.4
NEGATIVE.PARTICLE	160	1.4	79.4	76.4	68.8	72.4
OBJECT	7,956	1.8	75.9	63.3	62.5	62.9
MODIFIER	11,685	2.6	71.9	66.5	64.8	65.7
DATIVE.ADJUNCT	1,360	2.4	70.8	46.4	50.2	48.2
FOCUS.PARTICLE	23	1.1	69.6	0.0	0.0	0.0
SUBJECT	4,479	4.6	68.6	50.9	56.2	53.4
ABLATIVE.ADJUNCT	523	2.5	68.1	44.0	54.5	48.7
INSTRUMENTAL.ADJUNCT	271	3.0	62.7	29.8	21.8	25.2
ETOL	10	4.2	60.0	0.0	0.0	0.0
LOCATIVE.ADJUNCT	1,142	4.2	56.9	43.3	48.4	45.7
COORDINATION	814	3.4	54.1	53.1	49.8	51.4
S.MODIFIER	594	9.6	50.8	42.2	45.8	43.9
EQU.ADJUNCT	16	3.7	50.0	0.0	0.0	0.0
APPOSITION	187	6.4	49.2	49.2	16.6	24.8
VOCATIVE	241	3.4	42.3	27.2	18.3	21.8
COLLOCATION	51	3.3	41.2	0.0	0.0	0.0
ROOT	16	-	0.0	0.0	0.0	0.0
Total	43,572	2.5	76.0	67.0	67.0	67.0

- Determiners, particles, and nominals have an AS_U over 79% and link to nearby heads.
- Subjects, objects, and adjuncts have an AS_U between 55–79% and a distance of 1.8–4.6 IGs to their head.
- Modifiers, vocatives, and appositions, which are indistinguishable from other nominals, have distant dependencies with a much lower accuracy.

Other Errors

- Head-initial dependencies have an AS_U of 72.2. 87% of errors occur when dependents are linked to the wrong IG of the correct head.
- Head-final dependencies have an AS_U of 76.2.
- Error probability per word does not increase with sentence length.

6. Conclusion

- IG-based models consistently outperform word-based models regardless of the choice of parser and evaluation method.
- Using morphological information increases parsing accuracy substantially.
- The best results were obtained using the IG-based models with the deterministic classifier-based parser.