History Based Unsupervised Data Oriented Parsing

Mohsen Mesgar
Department of Computer Engineering,
Sharif University of technology,
Tehran, Iran
mmesgar@ce.sharif.ir

Gholamreza Ghasem-Sani
Department of Computer Engineering,
Sharif University of technology,
Tehran, Iran
sani@sharif.edu

Abstract

Grammar induction is a basic step in natural language processing. Based on the volume of information that is used by different methods, we can distinguish three types of grammar induction method: supervised, unsupervised, and semi-supervised. Supervised and semi-supervised methods require large tree banks, which may not currently exist for many languages. Accordingly, many researchers have focused on unsupervised methods. Unsupervised Data Oriented Parsing (UDOP) is currently the state of the art in unsupervised grammar induction. In this paper, we show that the performance of UDOP in free word order languages such as Persian is inferior to that of fixed order languages such as English. We also introduce a novel approach called History-based unsupervised data oriented Parsing, and show that the performance of UDOP can be significantly improved by using some history information, especially in dealing with free word order languages.

1 Introduction

Statistical methods of natural language processing have shown to be very successful in corpus based linguistics. One reason is that electronic based texts are now available more than ever (Charniak, 1997; Church, 1998). The success of statistical Part Of Speech (POS) tagger systems has caused the trend of research in lexical analysis, language modeling, and machine translation to be changed towards using various statistical methods (Feili and Ghassem-Sani, 2004; Charniak, 1996).

Grammar is an essential tool in many applications of natural language processing (Feili and Ghassem-Sani, 2004). Writing a natural language grammar by hand is not only a time-consuming and difficult task, but also it needs a large amount of skilled efforts. Availability of large parsed corpus such as Penn Treebank (Marcus et al., 1993) has facilitated the development of automatic methods of grammar induction.

Based on the level of supervision information that is used by the different grammar induction methods, they are divided in to three major groups (i.e., supervised, semi-supervised, and unsupervised).

Supervised and semi-supervised methods require large treebanks, which may not exist for many languages. Therefore, many researchers have focused on unsupervised methods. Unsupervised Data Oriented Parsing (UDOP) is currently the state of the art in unsupervised grammar induction. But in the case of free word order languages such as Persian, its performance is inferior to that of fixed order languages like English.

In this paper, we present a novel unsupervised algorithm, named History-Based Unsupervised Data Oriented Parsing (HUDOP), and show, how to improve the performance of UDOP by using history information.

In section 2, we discuss about different methods of grammar induction. In section 3, UDOP is explained. In section 4, the details of HUDOP are introduced. Section 5 presents our experimental results on English and Persian. Finally, we conclude the paper in section 6.

2 Grammar induction methods

As it was mentioned above, based on the level of information, there are three types of grammar inductions: supervised, semi-supervised and unsupervised.

Proceedings of Recent Advances in Natural Language Processing, pages 453–459,
Hissar, Bulgaria, 7-13 September 2013.

Supervised methods need fully-parsed and tagged corpora such as Penn Treebank (Marcus et al., 1993; Charniak, 1997; Collins, 1997; Charniak, 2000; Magerman, 1995; BoonkWan and Steedman, 2011). There are also some semi-supervised methods (Pereira and Schabes, 1992; Schabes et al., 1993; Koo et al., 2008), which use less information than their supervised counterparts. Also, semi-supervised methods need a rich corpus that for some natural language (e.g., Persian) does not currently exist. Thus, we have focused our attention on unsupervised methods. Unsupervised methods do not need to pars tree of sentences in training corpus.

Inside-Outside (IO) was introduced by Baker (1979) as an unsupervised algorithm. IO uses Expectation-Maximization (EM) to construct a grammar based on an un-bracketed corpus. The algorithm re-estimates rule probabilities toward some maximization on the training corpus. The algorithm may converge to local optima in different runs. This method is regarded as one of the basic algorithms of unsupervised grammar induction (Pereira and Schabes, 1992; Amaya et al., 1999; Casacuberta, 1996).

Alignment based Learning (ABL) is a learning method based on a linguistic principle: two constituents that belong to one family can be used instead of each other (Van Zaanen, 2000; Van Zaanen, 2002; Van Zaanen and Adriaans, 2001). EMILE, another grammar induction system based on this principle, initially used some levels of supervision, but later was modified to be a completely unsupervised system (Adriaans, 2001).

Another important category of unsupervised induction method is based on the distribution of words in sentences. It usually uses some distributional evidence to identify the constituents' structures (Klein and Manning, 2001). The main idea is that "the same constituents appear in the same contexts" (Clark, 2001; Klein and Manning, 2005). The so-called Context-Constituent Model (CCM) is based on this idea and works on the basis of a weakened version of the classic linguistic constituency test (Radford, 1988): constituents occur in their contexts.

The independence of the input sentence and its surrounding context are usually assumed in parsing. For instance in a Probabilistic Context Free Grammar (PCFG) model, each constituent is assumed to be independent of its surrounding constituents (Charniak, 1997). Such assumptions are not in fact valid in many cases. For instance, in English a noun phrase is more likely to be a pronoun when it is a subject of the sentence than when the noun phrase is in an object position (Allen, 1995). Similar condition exists in Persian, too. For instance, in Persian a pronoun subject can be dropped whereas pronouns in object positions cannot be dropped (Bijankhan, 2003; Bateni, 1995).

We can reduce the impact of this invalid independence assumption by using some form of history in parsing. For instance, the information about parent non-terminals can be utilized as a history of parsing. More specifically, P(NP→Pronoun| Parent=SUBJ) is higher than P(NP→Pronoun | Parent = VP). Therefore, some of the parsing dependencies between constituents can be modeled by history based parsing. History based models were initially developed at IBM (Black et al., 1992; Jelinek et al., 1992; Jelinek et al., 1994).

Increasing the dependencies on the context is the main feature of history based models. For instance, Johnson (1998) used the parent information of each non-terminal as the history information in the condition part of each rule. He showed that, instead of P(A→B|A), which is used in ordinary PCFG based parsing, using P(A→B|A, parent(A)), where parent(A) is the nonterminal immediately dominating A, has a major positive impact on the accuracy of the parsing.

Based on the idea proposed by Johnson (1998), the so-called History based IO (HIO), improved the performance of IO especially in Persian (Feili and Ghassem-Sani, 2004). Parent based CCM (PCCM) is another history based method, which improved CCM (Mirroshandel and Ghassem-Sani, 2008). PCCM employs the parent's information of each context and constituent to prevent from divergence in the likelihood space.

There are also other techniques for improving the quality of an unsupervised grammar induction algorithm by considering some limitations, or additional information. For instance, Carroll and Charniak (1992) limit the set of non-terminals of the right hand side of rules with a given left-hand side.

3 Unsupervised Data Oriented Parsing

Unsupervised Data Oriented Parsing (UDOP) was introduced in (Bod, 2006a; Bod 2006b; Bod, 2007). In the first step, it generates all possible binary trees for each sentence of the corpus. This is followed by extracting all possible binary subtrees for parsing new sentences. In some methods, they convert each subtree to parsing rules. Number of rules will be increased exponentially. So these methods use Goodman reduction algorithm but we use subtree originally due to we want use Hidden Markov Model (HMM) for finding best parse tree for input sentence (Goodman, 2003).

UDOP uses a combination operator between the sub-trees for parsing a new sentence. We use “$\circ$” as the symbol of the combination operator.

Two sub-trees can be combined if the root of the right operand is equal to the leftmost non-terminal of the left operand. For example, let t_1 and t_2 be two sub-trees. Figure 1 shows t_1 and t_2 and the tree resulted from combining t_1 and t_2.

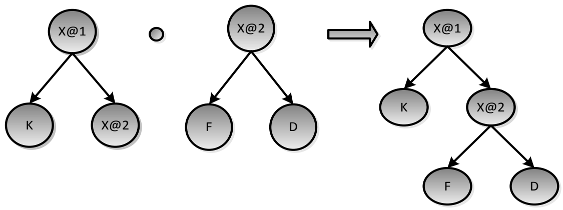

Figure 1. An example of the combination operator.

Let T be a parse tree for an input sentence resulted from combining sub-trees t_1, t_2, … , t_n (i.e., $t_1 \circ t_2 \circ \ldots \circ t_n$), then $t_1 \circ t_2 \circ \ldots \circ t_n$ is said to be a derivation of T (Rankin, 2007).

UDOP takes the shortest derivation as the best derivation. However, there may exist several shortest derivations. In such cases, in order to select the best derivation, UDOP uses probability.

The probability of any construction C is calculated by dividing the number of times C appears in the corpus by the number of times that any tree t with the same root appears in the corpus.

$$P(C) = \frac{|C|}{\sum_{\mathrm{t:\,root(C)=root(t)}} |t|} \qquad (1)$$

The probability of a derivation is calculated by the product of probabilities of all the constructions in the derivation:

$$\mathrm{P}(\mathrm{t}_1 \circ \mathrm{t}_2 \circ \ldots \circ \mathrm{t}_n) = \prod_j P(t_j) \qquad (2)$$

Note that, there is an implicit assumption that, given root node root(t_i), each t_i is independent of every other t_j where j<>i. The probability of a parse tree T is calculated by the sum the probabilities of all the possible derivations of T.

$$P(T) = \sum_{d \in D(T)} P(d) \qquad (3)$$

D(T) is the set of all possible derivations of T. Let T_j be a member in the set of all possible parse trees of a given sentence s. Then the preferred parse tree of s is the one that maximizes $P(T_i|s)$ in:

$$P(T_i \mid s) = \frac{P(T_i)}{\sum_j P(T_j)} \qquad (4)$$

4 History-based UDOP

For computing all possible derivations of a new sentence, we can use the HMM, where each state corresponds to a sub-tree. The probability of each state is equal to the frequency of the subtree of that state. It means, the probability of the state that contains the sub-tree t_i is calculated similar to UDOP as follows:

$$P(state_i) = \frac{|t_i|}{\sum_{t:root(t_i)=root(t)} |t|} \qquad (5)$$

where $state_i$ corresponds to sub-tree t_i.

States in each step of HMM produce states in the next step, using the combination operator. Note that not all states can be combined. This is due to the definition of the combination operator. The transition probability between those states that cannot be combined will be set to zero. It means that if t_i and t_j cannot be combined, then $P(t_i \rightarrow t_{ij})$ and $P(t_j \rightarrow t_{ij})$, where $t_i \rightarrow t_{ij}$ to presents the transition between $state_i$ and $state_{ij}$, are set to zero. On the other hand, let t_x be a sub-tree with root X. Assume t_y is any other sub-tree that can be combined with t_x at node X. Also suppose that in tree t_y, there is a node P(x,y) that immediately dominates X (i.e., P(x,y) is parent of node X in tree t_y). In this case, there is a transition between t_x and

t_{xy} (i.e., $t_x \rightarrow t_{xy}$). The probability of $t_x \rightarrow t_{xy}$ is calculated as follows:

$$(6)\, P(t_x \rightarrow t_{xy} \mid Parent_{\forall i;ix}(t_x) = p_{(x,y)}) = \frac{|t_{xi} : parent(t_x) = p_{(x,y)}|}{|t_{xi}|}$$

We used top-down generative process to generate the HMM. By using parent information, the transition probabilities of HMM is calculated more accurately than in the case of UDOP. In HUDOP, the calculation of other probabilities, such as that of derivations and parse trees, is the same as UDOP.

Finally, in HUDOP, similar to UDOP, in order to find the most probable parse tree, we have used the Viterbi 100-best method, which uses 100 most probable states (sub-trees) in each step of HMM (Bod, 2006b).

5 Experimental results

Two kinds of experiments are presented in this section. At first, the result of applying HUDOP to two different English data sets are demonstrated and compared with that of related work. Then, we show the results of applying HUDOP to Persian, as a free-word order language.

5.1 Experimental result in English

HUDOP was tested on both ATIS (Hemphill et al., 1990) and WSJ-10 (Schabes et al., 1993). We used PARSEVAL to evaluate the quality of the output grammars. Part of speech tag sequences were used as the only lexical information of the training sets.

We executed two different experiments on the English sentences. At first, ATIS was divided in two distinct sets: the training set with almost 90% of the data and the test set including the rest. Although, HUDOP is an unsupervised approach and does not require any bracketing data set, we need the tree style syntactic information of the test data set for the evaluation purpose. We evaluated HUDOP using the ten-fold cross validation method. Similar to the original UDOP, we selected sentences with the length shorter than ten.

In the first experiment, we selected the spoken-language transcription of the Texas Instruments subset of ATIS (Hemphill et al., 1990), which is a part of Penn Treebank.

Method	UP	UR	F1
EMILE	51.9	16.81	25.35
ABL	43.64	35.56	39.19
LEFT	19.89	16.74	18.18
RIGHT	39.9	46.4	42.9
IO	42.19	35.51	38.56
HIO	46.85	40.9	43.67
CCM	55.4	47.6	51.2
PCCM	-	-	52.08
UDOP	58.90	58.50	58.70
HUDOP	63.90	62.89	**63.39**

Table 1. The results of HUDOP and other methods on ATIS data set.

The results of comparing HUDOP with other unsupervised methods, including EMILE (Adriaans and Haas, 1999), ABL (Van Zaanen, 2000), and CCM (Klein and Manning, 2005), on ATIS are shown in table 1. LEFT and RIGHT are the left and the right-branching baselines applied to ATIS. The results of left and right baselines have been taken from Klein and Manning (2005). As table 1 shows, the performance of HUDOP is superior to all the mentioned work.

We also tested HUDOP on WSJ-10 and compared its results with a number of related works including the state of the art (i.e., UDOP). The results are shown in Figure 2.

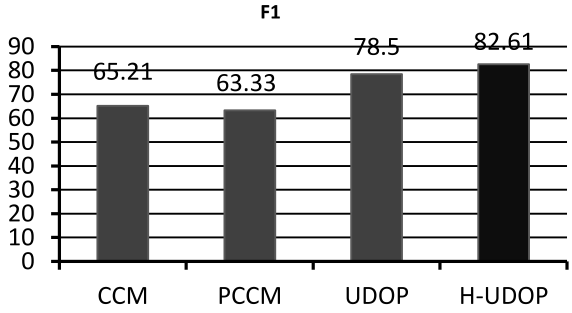

Figure 2. F1 scores for various models on WSJ-10.

5.2 Experimental results in Persian

We have also applied HUDOP to Persian, which is linguistically very different from English. Although many sentences in Persian have the form of SOV, it is generally considered to be a free-word-order language, especially in proposition adjunction and complements. It means that an adverb can be used at the beginning, in the middle, or at the end of sentences. This does not often change the meaning of the sentences.

In order to test HUDOP in Persian, we manually produced two different training corpora. All sentences of these corpuses contain less than 11 words, and have been extracted from a Persian corpus named Peykareh (Bijankhan, 2003; Megerdoomian, 2000). Peykareh has more than 32,255 sentences and uses a tag set similar to the tag set used in Amtrup et al. (2003). The first corpus included 3,000 sentences, which were manually changed in such a way that the structure of "S PP O V" was held. In other words, the common property of the sentences in this corpus was that the order of words were artificially fixed (i.e., they were not free in order). Table 2 shows main properties of the first corpus.

Property	Value
Number of sentence	3,000
Maximum length	10
Minimum length	2
Average Length	7
Number of words	22,153
Number of POS	18

Table 2. Main properties of first corpus.

The second corpus comprised 2,500 sentences with a high degree of free word orderness. Table 3 shows main properties of the second corpus.

Property	Value
Number of sentence	2,500
Maximum Length	10
Minimum Length	2
Average Length	7
Number of Words	18,482
Number of POS tags	18

Table 3. Main Properties of second corpus.

In Persian, we first ran both UDOP and HUDOP on each of the above corpora, separately. We also joined these corpuses to create a third mixed corpus, and repeated the experiments on this corpus, too. The results are shown in figure 3.

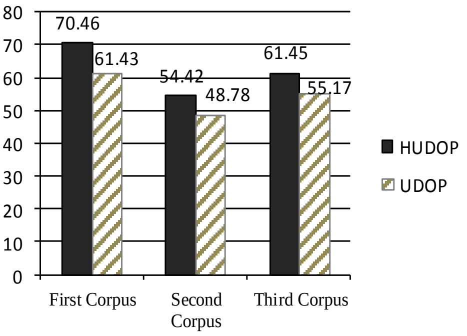

Figure 3. Comparison of UDOP and HUDOP methods in Persian (Based on the F1 measure).

Figure 3 shows the impact of the free word orderness property on the performance of both UDOP and HUDOP. The reduction in the performance of UDOP on the first corpus, in comparison to that of the second corpus, has been 13 percent in F1 score. The results of applying both UDOP and HUDOP to the combined corpus demonstrate little improvement. This shows that the free word orderness property of the input language has a negative effect on these methods.

The reason for this weakness is that these methods work based on the repetition of subtrees. Since in free word order languages, some words can freely appear in different places of sentences, the mentioned repetition decreases substantially, and as a result, the performance of the parsing is decreased.

The experiments also show that HUDOP outperforms UDOP in both languages.

6 Conclusion

Unsupervised Data Oriented Parsing (UDOP) is currently the state of the art in unsupervised grammar induction. UDOP works based on the repetition of possible sub-trees of parse trees of the input sentences. However, in free word order languages such as Persian, words can grammatically appear in different places of sentences. Thus, occurrence frequency of such sub-trees substantially decreases. In this paper, we proposed a novel approach, called History-based Unsupervised Data Oriented Parsing (HUDOP). We showed how by using parent nodes as a history notion of sub-trees, HUDOP outperforms UDOP. Parent information prevents from probability divergence and parsing will be more informative. To evaluate HUDOP, it was applied to both English and Persian (as a free word order language). The results of applying the new method to several corpuses with different degree of free word orderness showed that using parent information notably improves the performance of UDOP. One possible future work to improve the performance of HUDOP can be usage of other possible forms of history information. We are working on the idea implementing a semi-supervised HUDOP.

References

Adriaans, P. and Haas, E., (1999). Grammar induction as sub structural inductive logic programming. Proceedings of the 1st Workshop on Learning Language in Logic. Bled, Slovenia, pp. 117–127.

Allen, J., (1995). Natural Language Understanding. Benjamin/Cummings Pub.

Amaya, F., Benedi, J.M. and Sanchez, J.A., (1999). Learning of stochastic context-free grammars from bracketed corpora by means of re-estimation algorithms. The VIII Symposium on Pattern Recognition and Image Analysis, vol. 1, pp. 19–126.

Amtrup, J.W, Rad, H. R., Megerdoomian, K. and Zajac, R., (2000). Persian-English Machine Translation. An Overview of the Shiraz Project, NMSU, CRL.

Baker, J.K., (1979). Trainable grammars for speech recognition. Speech communication papers for the 97th Meeting of the Acoustical Society of America, pp. 547–550.

Bateni, M., (1995). Tosif-e Sakhtari Zaban-e Farsi (Describing the Persian Structure). Amir-Kabir Press, Tehran, Iran (in Persian).

Bijankhan, M., (2003). Emkansanji baraye Tarhe Modelsaziye Zabane Farsi (The feasibility study for Persian language modelling). The Journal of Literature. pp. 162–163.

Bijankhan, M., (2005). The role of corpus in generating grammar: presenting a computational software and corpus. Iranian Linguistic Journal 2 (19), pp. 48–67, (in Persian).

Black, E., Lafferty, J. and Roukos, S., (1992). Development and evaluation of a broad-coverage probabilistic grammar of English-language computer manuals. The Proceedings of the 30th Annual Meeting of the association for computational Linguistics, pp. 185–192.

Bod, R., (2006)a., Exemplar-based syntax: How to get productivity from examples? The Linguistic Review 23 (3), Special Issues on Exemplar- Based Models in Linguistics.

Bod, R., (2006)b., An All-subtrees Approach to Unsupervised Parsing. Proceedings ACL-COING, Sydney.

Bod, R., (2007)., A Linguistic Investigation into U-DOP. Proceeding ACL workshop on Cognitive Aspects of Computational Language Acquisition, pp.1-8.

BoonkWan, P. and Steedman. M., (2011)., Grammar Induction from Text Using Small Syntactic Prototypes. Proceedings of the 5th International Joint Conference on Natural language Processing, pp. 438-446.

Carroll, G. and Charniak, E., (1992). Two experiments on learning probabilistic dependency grammars from corpora. Technical Reports Department of Computer Science, Brown University, March.

Casacuberta, F., (1996). Growth transformations for probabilistic functions of stochastic grammars. IJPRAI 10 (3), pp. 183–201.

Charniak, E., (1996). Statistical Language Learning. MIT Press, Cambridge, London, UK.

Charniak, E., (1997). Statistical parsing with a context-free grammar and word statistics. Proceedings of the 14th National Conference on Artificial Intelligence, pp. 598–603.

Charniak, E., (1997). Statistical techniques for natural language parsing. AI Magazine 18 (4), pp. 33–44.

Charniak, E., (2000). A maximum-entropy-inspired parser. NAACL 1, pp. 132–139.

Church, K., (1988). A stochastic parts program and noun phrase parser for unrestricted text. Proceedings of the Second Conference on Applied Natural Language Processing, pp. 136–143.

Clark, A., (2001). Unsupervised induction of stochastic context-free grammars using distributional clustering. The Proceedings of 15th Conference on Natural Language Learning.

Collins, M., (1996). A new statistical parser based on bigram lexical dependencies. The Proceedings of the 34th Annual Meeting of the ACL, Santa Cruz.

Collins, M., (1997). Three generative, lexicalized models for statistical parsing. ACL 35/EACL 8, pp. 16–23.

Feili, H. and Ghassem-Sani G.,(2004). An Application of Lexicalized Grammars in English-Persian Translation. Proceedings of the 16th European Conference on Artificial Intelligence, pp. 596-600.

Goodman, J., (2003). Efficient algorithms for the DOP model. In R. Bod, R. Scha and K. Sima'an (eds.). Data-Oriented Parsing, The University of Chicago Press.

Hemphill, C.T., Godfrey, J. and Doddington, G., (1990). The ATIS spoken language systems pilot corpus. DARPA Speech and Natural language Workshop, Hidden Valey, Pennsylvania, June.

Jelinek, F., Black, E., Lafferty, J., Magerman, D.; Mercer, R. and Roukos, S. (1992). Towards history-based grammars: using richer models for probabilistic parsing. The Proceedings of the 5th DARPA Speech and Natural Languages Workshop, Harriman, NY.

Jelinek, F., Laferty, J.D., Magerman, D., Mercer, R.; Ratnaparakhi, A. and Roukos, S., (1994). Decision-tree parsing using hidden derivation model. The Proceedings of the Human Language Technology Workshop, pp. 272–277.

Johnson, M., (1998). The effect of alternative tree representations on treebank grammars. New Methods in Language Processing and Computational Natural Language Learning, ACL, pp. 39–48.

Klein, D. and Manning, C.D, (2005). The Unsupervised Learning of Natural Language Structure. Ph.D. Thesis, Department of Computer Science, Stanford University.

Klein, D. and Manning, C.D., (2001). Natural language grammar induction using a constituent-context model. Dietterich, T.G., Becker.

Koo, T., Carrera, X., and Collins, M., (2008). Simple Semi-Supervised Dependency Parsing. Proceeding of ACL 2008.

Magerman, D., (1995). Statistical decision-tree models for parsing. The Proceedings of ACL Conference.

Marcus, M.P., Santorini, B. and Marcinkiewicz, M.A., (1993). Building a large annotated corpus of English: the Penn Treebank. Computational Linguistics 19, pp. 313–330.

Megerdoomian, K., (2000). Persian Computational Morphology: A Unification-Based Approach. NMSU, CRL. Memoranda in Computer and Cognitive Science.

Mirroshandel, S. A. and Ghassem-Sani, G., (2008). Unsupervised Grammar Induction Using a Parent Based Constituent Context Model. 18th European Conference on Artificial Intelligence.

Pereira, F. and Schabes, Y., (1992). Inside-outside reestimation from partially bracketed corpora. The Proceeding of 30th Annual Meeting of the ACL, pp. 128–135.

Radford, A., (1988). Transformational Grammar. Cambridge University Press, Cambridge.

Rankin, J., (2007). Data Oriented Parsing, Literature Review, November. pp. 4.

Schabes, Y., Roth, M. and Obsorne, R., (1993). Parsing the Wall Street Journal with the inside-outside algorithm. The Proceedings of the 6th Conference of the European Chapter of the ACL, pp. 341–347.

Van Zaanen, M. and Adriaans, P.W., (2001). Comparing two unsupervised grammar induction systems: alignment-based learning vs. EMILE. Technical Report: TR2001.05, School of Computing, University of Leeds.

Van Zaanen, M., (2000). ABL: alignment-based learning. COLING 2000, pp. 961–967.

Van Zaanen, M., (2002). Bootstrapping structure into language: alignment-based learning. Ph.D. Thesis, School of Computing, University of Leeds.