History-Based Inside-Outside Algorithm

Heshaam Feili and **Gholamreza Ghassem-Sani**[1]

Abstract. Grammar induction is one of the most important research areas of the natural language processing. The lack of a large Treebank, which is required in *supervised* grammar induction, in some natural languages such as Persian encouraged us to focus on *unsupervised* methods. We have found the Inside-Outside algorithm, introduced by Lari and Young, as a suitable platform to work on, and augmented IO with a history notion. The result is an improved unsupervised grammar induction method called History-based IO (HIO). Applying HIO to two very divergent natural languages (i.e., English and Persian) indicates that inducing more conditioned grammars improves the quality of the resultant grammar. Besides, our experiments on ATIS and WSJ show that HIO outperforms most current unsupervised grammar induction methods.

1 INTRODUCTION

With the recent increasing interest in statistical approaches to natural language processing, *corpus* based linguistics has become a hot topic. This is due to the fact that computer based texts are available more than ever before, and easier to use for various data tasks. The success of part-of-speech tagging by using the Hidden Markov Model (HMM) [1, 2] also attracted the attention of computational linguists to the lexical analysis, language modeling, and machine translation by using various statistical methods [3, 4].

Manual design and refinement of a natural language grammar is a difficult and time-consuming task, and requires a large amount of skilled efforts. A hand-crafted grammar is not usually completely satisfactory, and frequently fails to cover many unseen sentences. Automatic acquisition of grammars is a solution to this problem. With the increasing availability of large, machine-readable, parsed corpora such as Penn Treebank [5], there have been numerous attempts to automatically derive a context free grammar by using such corpora [6].

Based on the level of supervision, which is used by different algorithms, grammar induction methods are divided in three main categories: *supervised, semi-supervised, and unsupervised*. Here, we present a novel unsupervised algorithm named **H**istory-Based **I**nside-**O**utside (HIO) as an extension of the well-known unsupervised **I**nside-**O**utside algorithm [7]. The experimental results of applying HIO to both English and Persian languages are also demonstrated and compared with that of several other unsupervised approaches.

[1] Department of Computer Engineering, Sharif University of Technology, Tehran, Iran, emails: {hfaili@mehr.sharif.edu, sani@sharif.edu}

2 PREVIOUS WORKS

In unsupervised grammar induction methods only tagged sentences without any bracketing information or other supervised information are used. Based on the Expectation Maximization (EM) algorithm, Lari and Young, proposed what they called the *Inside-outside* (IO) algorithm, that constructs a grammar from an unbracketed corpus [7]. The algorithm will converge to a local optimum when used to iteratively re-estimate probabilities on a training corpus in a manner, which maximizes the likelihood of the training corpus, given the grammar. This method is so far one of the basic algorithms for unsupervised automatic learning of grammars [8]. Also, Stolcke and Omohundro induced a small and artificial context free grammar with chunk-merge systems [9]. The results of these approaches for completely unsupervised acquisition showed that they are generally ineffective.

There are also other works to improve the quality of the unsupervised induction methods by considering some limitation or additional information. Magerman and Weir use a distituent grammar to eliminate undesirable rules [10]. Carroll and Charniak, restrict the set of non-terminals that may appear on the right hand side of rules with a given left hand side [11].

One of the most promising classes of unsupervised induction algorithms is based on particular distribution of words in sentences, and uses some distributional evidences to identify constituent structure [12]. Here, the main idea is that sequences of words (or tags) generated by the same non-terminal normally appear in similar contexts [13, 14]. Klein and Manning have introduced a new distributional method for inducing a bracketed tree structure of the sentence, with a dependency model to induces a word-to-word dependency structure [12, 14]. By combining these two models, they have achieved the best result in unsupervised grammar induction so far. Other dependency models with weaker results were presented by [11, 15, 16].

Alignment Based Learning (ABL) is a learning paradigm that can be regarded as a distribution based method. It is based on the principle of substitutability, whereby two constituents are of the same type, and then they could be substituted [17, 18]. Also, Adriaans presents EMILE, which initially used some aspects of supervision, but in later work is modified to be completely unsupervised [19]. Both the ABL and EMILE techniques look for *minimal pairs*; a specific form of distributional learning, where the contexts are the rest of the sentence.

Although supervised methods outperform current unsupervised induction algorithms with a relatively large gap, there are still compelling motivations for working on unsupervised methods [20]. Building supervised training data requires considerable resources, including time and linguistic expertise. This problem is more complicated when we deal with languages other than English,

which usually lack the necessary resources. Furthermore, the resulting hand-crafted treebank may be too susceptible to restriction to a particular domain, application, or genre [21].

On the other hand, applying probabilistic model to natural languages has been investigated in several works where the independence of the input sentence and its context is assumed in parsing [1, 22]. In fact, most works have used even stronger independence assumptions. For instance, the PCFG model assumes the independence of the probability of each constituent and its neighboring constituents [1]. On the other hand, there are some richer models of context that incorporate some additional information with the probability of each constituent and present a way of calculating the probability model more accurately [22, 23].

There have been some promising works adopted the history based grammar induction methods. For instance, *Pearl* is a probabilistic parser that is more sensitive to the model of context [24]. Using supervised learning methods; *Pearl* acquired 88% of bracketing accuracy.

Another important work, which increases the dependencies on the context, is the *history-based parser* that was originally developed by the researchers at IBM [23, 25]. In these models, the parse-tree representation was enriched in a couple of ways: non-terminal labels were augmented by some extra information such as lexical items and head word. An improvement from 59.8% to 74.6% in parsing accuracy was reported by using this model [23].

The idea of adding the parent of each non-terminal as the conditioning information to the grammar rules was also mentioned in [22, 26, 27]. Replacing $P(\alpha \rightarrow \beta \mid \alpha)$ by $P(\alpha \rightarrow \beta \mid \alpha, \text{Parent}(\alpha))$, where $\text{Parent}(\alpha)$ is the non-terminal dominating α, leads to an improvement from 69.6% / 73.5% to 79.3% / 80.1% of the precision/recall metrics. In this paper, we introduce an extension of the IO algorithm augmented by the history notion, and apply it to two very different languages (i.e., English and Persian).

3 HIO ALGORITHM

In this section, we explain HIO, a new estimation model to induce an extended form of a PCFG by using the traditional IO algorithm. Here, the output of IO algorithm and other partial information such as inner and outer probabilities are used as the input data. This information is necessary to calculate the history-based model. The main idea of HIO is to decompose the output probabilities of rules extracted from IO, over their parent non-terminal. This decomposition can guide the parser toward better decisions in analyzing its input sentences.

In order to incorporate the parent non-terminal into the Chomsky normal form, we use the following notation to state the probability of using rule $i \rightarrow jk$[2]:

$$A[C,i,j,k] = P(i \rightarrow jk \mid i \text{ used in derivation}, C = \text{Parent}(i), C \text{ used in derivation}) \quad (1)$$

Considering the conditional probability:

$$p(i \rightarrow jk \mid i-used, C = parent(i), C-used) = \frac{p(i \rightarrow jk, C = Parent(i) \mid i-used, C-used)}{p(C = parent(i))} \quad (2)$$

and $C = Parent(i)$, we can infer:

$$\forall U\ (C \rightarrow U\ i \text{ or } C \rightarrow i\ U) \quad (3)$$

$$p(C = Parent(i)) = P(\forall U, C \rightarrow U\ i) + P(\forall U, C \rightarrow i\ U) - P(C \rightarrow i\ i) \quad (4)$$

In the original IO algorithm [7], matrix "*a*" is defined as follows:

$$a[i,j,k] = p(i \rightarrow jk \mid i\text{-used}) \quad (5)$$

From (3) and (4), we can derive:

$$\forall C, \forall i,\ p(C=parent(i)) = \Sigma_u a[c,u,i] + \Sigma_u a[c,i,u] - a[c,i,i] \quad (6)$$

By the assumption of having observation *O*, we obtain:

$$p(C \Rightarrow^* O(s)\ldots O(t) \mid S \Rightarrow^* O, G) = e(s,t,C) \cdot f(s,t,C) / P \quad (7)$$

where $P = p(S \Rightarrow^* O \mid G)$, and $e(s,t,C)$ *and* $f(s,t,C)$ are the inner and outer probabilities, respectively.

But:

$$e(s,t,i) = \sum_{j,k}[\sum_{r=s}^{t-1} a[i,j,k].e[s,r,j].e(r+1,t,k)] \quad (8)$$

Thus, if at the first step of derivation starting from non-terminal *C*, rule "$C \rightarrow i\ U$" is used, and by considering definitions of the inner probability (i.e., eq. 8) and matrix *a* (i.e., eq. 5) , we can derive:

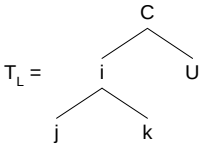
$$p(C \rightarrow iU \overset{*}{\Rightarrow} O(s)\ldots O(t) \mid S \overset{*}{\Rightarrow} O, G) = \frac{1}{P}\sum_{r=s}^{t-1} a[C,i,U].e(s,r,i).e(r+1,t,U).f(s,t,C) \quad \forall i,U,t>s \quad (9)$$

Supposing that in the next step of the derivation, rule "$i \rightarrow jk$" is used, the following equation holds:

$$p(C \rightarrow iU \rightarrow (jk)U \overset{*}{\Rightarrow} O(s)\ldots O(t) \mid S \overset{*}{\Rightarrow} O, G) = \frac{1}{P}\sum_{r=s}^{t-1}\sum_{v=s}^{r-1} a[C,i,U].a[i,j,k].e(s,v,j).e(v+1,r,k).e(r+1,t,U).f(s,t,C)\ \forall i,j,U,t>s \quad (10)$$

By naming the next left branching tree as T_L, we can rewrite equation (10) as follows:

$T_L =$



$$p(T_L, i\text{-used}, C\text{-used}) = \text{equation (10)} \quad (11)$$

Therefore:

$$p(T_L \mid i\text{-used}, C\text{-used}) = \text{equation (10)} / P(i\text{-used}, C\text{-used}) \quad (12)$$

By assuming the independence of using non-terminals *i* and *C*, we have:

$$p(i\text{-used}, C\text{-used}) = p(i\text{-used}) \cdot p(C\text{-used}) \quad (13)$$

And from (12) and (13), we get the following equation:

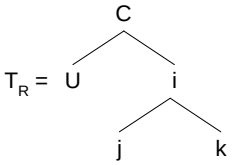
$$p(T_L \mid i\text{-used}, C\text{-used}) = \text{equation(10)} / (p(i\text{-used}) \cdot p(C\text{-used})) \quad (14)$$

Equation (15) can be derived in a way similar to that of (10):

$$p(C \rightarrow Ui \rightarrow U(jk) \overset{*}{\Rightarrow} O(s)\ldots O(t) \mid S \overset{*}{\Rightarrow} O, G) = \frac{1}{P}\sum_{r=s}^{t-1} a[C,U,i].\sum_{v=r+1}^{t-1} a[i,j,k].e(r,v,j).e(v+1,t,k).e(s,r,U).f(s,t,C)\ \forall i,j,U,t>s \quad (15)$$

And by defining T_R, as follows:

[2] For the sake of simplicity, we ignore the assumption of using grammar G in all probabilities. Also, in the remainder of the paper, we use "i-used" to show that non-terminal *i* has been used in the derivation process.



$p(T_R \mid i\text{-used}, C\text{-used}) = \text{equation(15)} / (P(i\text{-used}) \cdot P(C\text{-used}))$ (16)

Now, we can evaluate the main equation:

$p(i \rightarrow jk, C = Parent(i) \mid i\text{-used}, C\text{-used}) =$
$\Sigma_U\ p(i \rightarrow jk, (C \rightarrow Ui \text{ or } C \rightarrow iU) \mid i\text{-used}, C\text{-used}) =$
$\Sigma_U\ p(i \rightarrow jk, C \rightarrow Ui \mid i\text{-used}, C\text{-used}) +$
$P(i \rightarrow jk, C \rightarrow iU \mid i\text{-used}, C\text{-used}) -$
$p(i \rightarrow jk, C \rightarrow ii \mid i\text{-used}, C\text{-used})$ (17)

T_L and T_R are equal to:

$T_L = \{ C \rightarrow i\ U, i \rightarrow jk \}, T_R = \{ C \rightarrow U\ i, i \rightarrow jk \}$ (18)

By replacing T_L and T_R in (17), one gets:

$p(i \rightarrow jk, C = Parent(i) \mid i\text{-used}, C\text{-used}) =$
$eq.(14) + eq.(16) -$
$p(C \rightarrow i\ i \rightarrow (jk)\ i \rightarrow (jk)(jk) \mid i\text{-used}, C\text{-used})$ (19)

The last term of (19), can be computed by using (10), where *U* is initialized to *i* and later expanded by i→jk. Thus:

$$p(C \rightarrow ii \rightarrow (jk)i \rightarrow (jk)(jk) \overset{*}{\Rightarrow} O(s)...O(t) \mid S \overset{*}{\Rightarrow} O, G) =$$
$$\frac{1}{P}\sum_{r=s}^{t-1} a[C,i,i].f(s,t,C). \quad (20)$$
$$\sum_{v=s}^{r-1} a[i,j,k].e(s,v,j).e(v+1,r,k).$$
$$\sum_{w=r+1}^{t-1} a[i,j,k].e(r+1,w,j).e(w+1,t,k) \quad \forall i,j,t > s$$

The third line of equation (20) is equal to e(s,r,j) and the fourth line is e(r+1,t, i) (see [7]).

The probability of using any non-terminal *i* in a derivation, is computed as in [7]:

$$p(i\ used) = \sum_{s=1}^{T}\sum_{t=1}^{T} \frac{1}{P} e(s,t,i).f(s,t,i) \quad (21)$$

Considering equations (2), (5), (9), (20) and (21), we can compute the probability mentioned in (1), which is the main parameter of the history-based model:

$p(i \rightarrow jk \mid i\text{-used}, C = Parent(i), C\text{-used}) =$
$(eq.(14) + eq.(16) - eq.(20)) / eq.(21)$ (22)

The last equation, denoted by *A[C, i, j, k]*, is the main estimation formula used by HIO for supporting the history-based model.

In a similar manner, the second form of Chomsky rules ($i \rightarrow m$), can be extended to support probability of using such rules, considering parent non-terminals as well.

After evaluation of parameter matrices *a* and *b* in the traditional IO algorithm, we evaluate matrices *A* and *B*. The main iteration of inside-outside algorithm will be terminated when changes in the overall probability of observations is less than a pre-defined threshold.

The HIO model induced by the mentioned approach assumes parent non-terminals in the parsing. Therefore, the algorithm which is used during the parsing time, should considers the parent non-terminal too. An extension of PCYK algorithm described in [27] is used for this purpose.

4 EXPERIMENTAL RESULTS

Two kinds of experiments are presented in this section. At first the results of evaluating HIO on a number of English data sets are demonstrated. Then the results of applying HIO to Persian, which is essentially very different from English, are also discussed. HIO was tested on both ATIS [28] and Wall Street Journal [6]. In order to be able to compare with others' work, we selected these data sets for testing the induced grammar. We used only POS tag sequences as the lexical information of the training and testing data sets.

We executed two different experiments on English sentences. At first, as in other works, ATIS was divided into two distinct sets: the *training* set with approximately 90% of data and the *test* set (i.e., remaining 10%). Note that although our approach is unsupervised and does not need a bracketing data set, we need the tree style of syntactic information of the test data set for the evaluation purpose. The results were computed using the so-called ten fold cross validation method. In the second experiment, 7400 sentences shorter than 11 words of the Wall Street Journal (WSJ-10) were selected. This data set was the main corpus used in training of HIO, and the induced grammar was tested on both ATIS and WSJ, too.

The initial grammar of IO is a full grammar, which contains all possible CNF rules, with random probabilities assigned to each rule. To alleviate the algorithm from any possible bias, the process was repeated one hundred times with different initial probabilities.

The outputs of all experiments were evaluated on a test corpus by using the extended PCYK parsing algorithm. This algorithm gets the extended model of PCFG and a test sentence, generates all possible parses of the input sentence in a dynamic programming manner, and selects the most probable parse. Then the parsed sentences were evaluated by comparison against the corresponding treebank of the same corpus.

In the first experiment, we selected spoken-language transcription of the Texas Instruments subset of the Air Travel Information System (ATIS) corpus [28]. This corpus, which has been automatically labeled, analyzed and manually checked, is included in Penn Treebank II [5]. There are two different pieces of labeling information in the Treebank: a part of speech tag and a syntactic labeling. We used 577 sentences of the corpus with 4609 words. We ran the experiments with both 5 and 15 non-terminals[3], and every experiment was also repeated one hundred times. The means of the results from both IO and HIO, and the standard deviations of the latter are shown in table 1.

Table 1. the results of IO and HIO on ATIS data set

No. of non-terminals = 5			
	UP	UR	F1
IO	30.03	25.27	27.45
HIO	42.54(3.15)	38.95(5.8)	40.67(6.8)
No. of non-terminals = 15			
	UP	UR	F1
IO	42.19	35.51	38.56
HIO	46.85(3.2)	40.9(5.9)	43.67(3.9)

As it is shown, HIO significantly outperforms IO on the mentioned corpus. Moreover, the ratio of improvement of IO is

[3] By considering the NULL non-terminal, the numbers of used non-terminal in these experiments are 6 and 16 respectively.

very sensitive to the number of non-terminals. In other words, the number of non-terminals used in the IO algorithm has a remarkable impact on the accuracy of the induced grammar; while HIO's accuracy improvement is much less dependent on this factor. This is due to the structure of the richer model used by HIO[4].

To compare HIO with other unsupervised methods, we collected the results of testing several different approaches on ATIS data set in table 2: EMILE [19], ABL [17], CDC with 40 iterations [13], and CCM [14]. LEFT and RIGHT are left- and right-branching baselines applied to ATIS. The results of LEFT and RIGHT baselines have been taken from [14].

HIO was trained on two different data sets. It was trained on WSJ-10, and tested on ATIS data set. Then it was also trained on 90% of ATIS, and tested on the remaining 10%. The second experiment was evaluated by the ten fold cross validation method. HIO-WSJ and HIO-ATIS respectively correspond to these two experiments.

Table 2. the results of different approaches on ATIS data set

Method	***UP***	**UR**	**F1**
EMILE	51.59 (2.70)	16.81 (0.69)	25.35 (1.00)
ABL	43.64 (0.02)	35.56 (0.02)	39.19 (0.02)
CDC-40	53.4	34.6	42.0
CCM	55.4	47.6	**51.2**
LEFT	19.89	16.74	18.18
RIGHT	39.9	46.4	42.9
IO	42.19	35.51	38.56
HIO-WSJ	45.2 (2.6)	41.6 (1.5)	43.32 (1.9)
HIO-ATIS	46.85 (3.2)	40.9 (5.9)	43.67 (4.15)

As it's shown, HIO's output is superior to that of all other mentioned works except CCM, which seems to be the current state of the art in unsupervised grammar induction. Also, results of HIO-ATIS are better than that of HIO-WSJ because the test and training data set have been selected from the same corpus.

We also tested HIO on WSJ in order to compare it with other published works, especially with CCM [14]. The mean value of F1 score for HIO and other different methods on WSJ-10 are shown in Figure 1[5]. RANDOM selects a tree uniformly at random from the set of binary trees. DEP-PCFG is the result of duplicating the experiments of [11], using EM to train a dependency structured PCFG. SUP-PCFG is a supervised PCFG parser trained on a 90-10 split of WSJ-10 data set, using the treebank grammar with the Viterbi parse right-binarized. UBOUND is the upper bound of how well a binary system can cope with the treebank sentences that are generally flatter rather than being binary, limiting the maximum achievable precision. As it is shown, although HIO outperforms the right-branching baseline, CCM still has the best rank among unsupervised methods.

We have also tested HIO on Persian, which is linguistically very divergent from English [3]. In order to test HIO on Persian, we manually produced a treebank adopting different notions of the Penn Treebank. We chose a data set called *Peykareh* as the initial corpus for our experiments [29]. This corpus has more than 7000 sentences collected from formal newsletters in Persian. The tag set that was used in the corpus is the same as one used by [30, 31]; however some of the tags were merged in order to be used in the syntactical analysis[6], remaining only 18 POS tags. We selected 2200 sentences shorter than 11 words from Peykareh for both HIO and IO. Table 3, shows the results of these experiments.

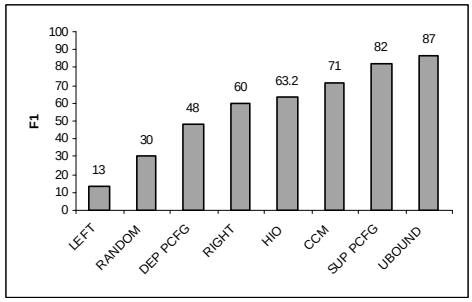


Figure 1. F1 score for various models on WSJ-10

Table 3. the results of IO and HIO on Persian corpus with sentences shorter than 11 words

No. of non-terminals = 5			
	UP	UR	F1
IO	33.25(0.64)	31.93(0.87)	32.58(0.74)
HIO	41.26(1.16)	38.59(2.3)	39.88(1.54)
No. of non-terminals = 15			
	UP	UR	F1
IO	44.35(0.5)	40.1(0.68)	42.19(0.58)
HIO	52.5(1.29)	50.28(1.7)	**51.37(1.5)**

As it is shown, in Persian too, HIO outperforms IO by more than 20% with respect to the F1 measure. Figure 2 compares the effect of the grammar size (i.e., the number of its non-terminals) on the quality of the induced grammar (again based on F1) in Persian with that of English. This comparison implies that the grammar size affects Persian more than English. In other words, a larger grammar is required to model Persian. That is mainly because Persian is a free-word-order language, and thus harder to model. Therefore, HIO can achieve even better results on free-word order languages by using larger grammars.

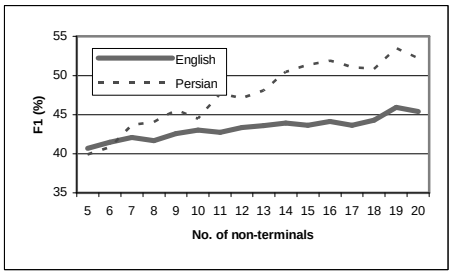


Figure 2. The effect of the grammar size on F1 measure in English and Persian

5 CONCLUSION

We showed that adding the parent non-terminal label to the rule assumptions increases the quality of output grammar. In other words, by relaxing the flawed independence assumption, we can construct a richer model of unsupervised grammar induction method. It might be the case that augmenting the model with some

[4] Note that HIO uses an extended PCFG model in which each rule is associated with the parent of its left-hand side.

[5] The higher results on WSJ, compared with that of ATIS, are due to the tendency of ATIS to have shorter constituents, especially with one word. HIO and IO do not bracket single words, and that is why the F1 measure (especially recall) decreases.

[6] The tag set presented by [29, 31] was used only for morphological analysis tasks; the set needed to be re-arranged and merged in a syntactical analysis view.

other conditions further improves the accuracy of the results. However, adding extra conditions to the model may also cause an exponential increase in the size of the output grammar, which hampers the parsing process.

We introduced HIO, a new approach based on the well-known IO algorithm for inferring stochastic context-free grammars. We relaxed the independence assumptions often used with the PCFG rules in the parsing process by adding some extra information about the context to the derivation process. Here, the parent non-terminal, which dominates the PCFG rule, was chosen as the contextual information.

The new method was applied to both English and Persian languages. The results showed a significant superiority over almost all other unsupervised methods. The results on Persian was even more prominent though it is a free-word-order language and harder to model. CCM, a distributional algorithm based on the idea that similar words occur in similar contexts [14], is the only unsupervised method that outperforms HIO on English. However, as it was also pointed out by Clark [11], we think that distributional methods might not be applicable to free-word-order languages, and intend to verify that.

ACKNOWLEDGMENTS

We would like to thank the anonymous reviewers of the paper for their invaluable suggestions and comments. This work has been partially supported by the Iranian Telecommunication Research Center (ITRC).

REFERENCES

[1] Charniak, E., "Statistical techniques for natural language parsing", AI Magazine, Vo. 18, No. 4, pp. 33-44, Winter 1997.

[2] Church, K., "A stochastic parts program and noun phrase parser for unrestricted text", In the Proceedings of the Second Conference on Applied Natural Language Processing, pp. 136-143, 1988.

[3] Feili, H. and Ghassem-Sani, G., "An Application of Lexicalized Grammars in English-Persian Translation", Proceedings of the 16th European Conference on Artificial Intelligence (ECAI 2004), Universidad Politecnica de Valencia, Spain, pp. 596-600, 2004.

[4] Charniak, E., "Statistical Language Learning", Cambridge, London, UK, MIT Press, 1996.

[5] Mitchell P., Marcus, B., Santorini, ce., and Marcinkiewicz, M. A., "Building a large annotated corpus of English: the Penn Treebank", Computational Linguistics, Vol. 19, pp. 313-330, 1993.

[6] Pereira, F. and Y. Schabes, "Inside-Outside reestimation from partially bracketed corpora ", In proceeding of 30th annual Meeting of the ACL, pp. 128-135, 1992.

[7] Lari, K. and Young, S.J., "The estimation of stochastic context-free grammar using the inside-outside algorithm", Computer Speech and Language, Vol. 4, pp. 35-56, 1990.

[8] Briscoe, T., and Waegner, N., "Robust stochastic parsing using the inside-outside algorithm". In AAAI-92 Workshop on Statistically Based NLP Techniques, 1992.

[9] Stolcke, A., and Omohundro, S. M., "Inducing probabilistic grammars by Bayesian model merging". In Grammatical Inference and Applications: Proceedings of the Second International Colloquium on Grammatical Inference. Springer Verlag, 1994.

[10] Magerman, D. M. and Marcus, M. P., "Parsing a natural language using mutual information statistics", In Proceedings of the Eighth National conference on Artificial Intelligence, August, 1990.

[11] Carroll, G. and Charniak, E., "Two experiments on learning probabilistic dependency grammars from corpora". In C. Weir, S. Abney, R. Grishman, and R. Weischedel, editors, Working Notes of the Workshop Statistically Based NLP Techniques, pages 1–13. AAAI Press, 1992.

[12] Klein, D., and Manning, C. D., "Natural language grammar induction using a constituent-context model", In T. G. Dietterich, S. Becker, and Z. Ghahramani (Eds.), Advances in Neural Information Processing Systems 14 (NIPS 2001), Vol. 1, 35–42, MIT Press, 2001.

[13] Clark, A., "Unsupervised Language Acquisition: Theory and Practice", PhD thesis, University of Sussex, 2001.

[14] Klein, D., "The Unsupervised Learning Of Natural Language Structure", PhD Thesis, Department of Computer Science, Stanford University, 2005.

[15] Yuret, D., "Discovery of Linguistic Relations Using Lexical Attraction", PhD thesis, MIT, 1998.

[16] Paskin, M. A., "Grammatical bigrams", In T. G. Dietterich, S. Becker, and Z. Ghahramani (Eds.), Advances in Neural Information Processing Systems 14, Cambridge, MA. MIT Press, 2002.

[17] Van Zaanen, M., "ABL: Alignment-Based Learning", In COLING 2000, pp. 961-967, 2000.

[18] Van Zaanen, M. and Adriaans, P. W., "Comparing Two unsupervised Grammar Induction Systems: Alignment-Based Learning vs. EMILE", Technical Report: TR2001.05, School of Computing, University of Leeds, 2001.

[19] Adriaans, P., and Haas, E., "Grammar induction as sub-structural inductive logic programming". In J. Cussens (Ed.), Proceedings of the 1stWorkshop on Learning Language in Logic, 117–127, Bled, Slovenia, 1999.

[20] Marcken, C., "Unsupervised Language Acquisition", PhD. thesis, Department of Electrical Engineering and Computer Science, MIT, 1996.

[21] Kehler, A. and Stolcke, A., Preface, In A. Kehler and A. Stolcke, editors, "Unsupervised Learning in Natural Language Processing", Association for Computational Linguistics, Proceedings of the workshop, 1999.

[22] Johnson, M., "The Effect of Alternative Tree Representations on Tree Bank Grammars", In D.M.W. Powers (ed.) NeMLaP3/CoNLL98: New Methods in Language Processing and Computational Natural Language Learning, ACL, pp. 39-48, 1998.

[23] Black, E., Jelinek, F., Lafferty, J., Magerman, D., Mercer, R. and Roukos, S., "Towards History-based Grammars: Using Richer Models for Probabilistic Parsing", In the Proceedings of the 5th DARPA Speech and Natural languages Workshop, Harriman, NY, 1992.

[24] Magerman, D. and Marcus, M., "Pearl: A Probabilistic Chart Parser", In the Proceedings of the 1991 European ACL conference, Berlin, Germany, 1991.

[25] Jelinek, F., Laferty, J. D., Magerman, D., Mercer, R., Ratnaparakhi, A. and Roukos, S., "Decision-Tree Parsing using Hidden Derivation Model", In the Proceedings of the 1994 Human Language Technology Workshop, pp. 272-277, 1994.

[26] Feili, H. and Ghassem-Sani, G., "One Step toward a richer model of unsupervised grammar induction", Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2005), 21-23 September, 2005, Borovets, Bulgaria, pp. 197-203.

[27] Feili, H. and Ghassem-Sani, G., "Unsupervised Grammar Induction Using History Based Approach", to appear in Computer Speech and Language Journal, Elsevier publishing, 2006.

[28] Hemphill, C.T., Godfrey, J., and Doddington, G., "The ATIS spoken language systems pilot corpus", In DARPA Speech and Natural language Workshop, Hidden Valey, Pennsylvania, June 1990.

[29] Bijankhan, M., "Naghshe Peykarehaye Zabani dar Neveshtane Dasture Zaban:Mo'arrefiye yek Narmafzare Rayane'i [the Role of Corpus in generating grammar: Presenting a computational software and Corpus]", Iranian Linguistic Journal, No. 19, Vol. 2, pp. 48-67, 2005 (in Persian).

[30] Amtrup, Jan W., Rad, H. R., Megerdoomian, K., and Zajac, R.,. "Persian-English Machine Translation: An Overview of the Shiraz Project", NMSU, CRL, Memoranda in Computer and Cognitive Science (MCCS-00-319), 2000.

[31] Megerdoomian, Karine, "Persian Computational Morphology: A Unification-Based Approach". NMSU, CRL, Memoranda in Computer and Cognitive Science (MCCS-00-320), 2000.