ISSN (Online) : 2454-4159

International Journal of Advanced Research in Science Management and Technology

Volume 2, Issue 3, March 2016

Multi Way Inputs Algorithm for Grammar Generation and Optimization

Mr.S.R.Gaidhani[1], Prof. Khushboo Sawant[2], Prof.Krishnakant Kishore[3]

PG Scholar, Dept. of Computer science & Engg., J.D.C.T. college, Indore, (M.P.), India[1]

Assistant Professor, Dept. of Computer science & Engg, J.D.C.T. college, Indore, (M.P.), India[2]

Head of Department, Dept. of Computer science & Engg, J.D.C.T. college, Indore, (M.P.), India[3]

ABSTRACT— The high costs of development and maintenance of multimodal grammars in integrating and understanding input in multimodal interfaces lead to the investigation of novel algorithmic solutions in automating grammar generation and in updating processes. Many algorithms for context-free grammar inference have been developed in the natural language processing literature. An extension of these algorithms toward the inference of multimodal grammars is necessary for multimodal input processing. In this paper, we propose a novel grammar inference mechanism that allows us to learn a multimodal grammar from its positive samples of multimodal sentences. The algorithm first generates the multimodal grammar that is able to parse the positive samples of sentences and, afterward, makes use of two learning operators and the minimum description length metrics in improving the grammar description and in avoiding the over-generalization problem. The experimental results highlight the acceptable performances of the algorithm proposed in this paper since it has a very high probability of parsing valid sentences.

I. INTRODUCTION

The USE of the five senses of touch, hearing, sight, smell, and taste allows human beings to perceive the external world. A combination of these senses is also used in all situations of the everyday life during natural human–human communication. Therefore, communication between human beings is multimodal in its nature. In the last few years, the multimodal interaction paradigm has been extensively studied, and it is being applied in human–computer interfaces, with the aim of making computer behavior closer to human–human communication paradigm. Therefore, multimodal interfaces, which allow us to communicate with the computer through the simultaneous or alternative use of several channels of input/output at a time, have gained increasing importance in human–computer interaction research.

II. LITERATURE SURVEY

The studies in grammatical inference exist in several application domains. Such as speech recognition [10],computational linguistics [11], computational biology [12], [4], and machine learning[12][13]. Many of these learning models take as input an initial set of training examples and as output the language description, i.e., the specific grammar that accepts only those examples. Mostly algorithms for NL grammar inference focus on context free grammar. There are three existing grammatical algorithm discussed here. The inductive CYK algorithm [4] synthesizes CFGs from positive and negative sample strings and generate the

ISSN (Online) : 2454-4159

International Journal of Advanced Research in Science Management and Technology

Volume 2, Issue 3, March 2016

minimum production rules, which derive positive strings but do not derive any given negative strings. The main advantages of the extended inductive CYK algorithm rely on the generation of simpler sets of rules and shorter computational time (compared to the other grammatical inference mechanisms) in the inference of CFGs for some simple languages.

III. PROPOSED ALGORITHM

Multimodal Grammar Inference Algorithm-:

First Phase of the MGI Algorithm:

The First step of the MGI algorithm enhances the inductive CYK algorithm in generating the MAG on two main aspects:

Input:

An input sentence x : x1, x2, xk , a set T = { x1, x2…xn} of terminal symbols , a multimodal attributes grammar G = {G, A, R}. A target sentence xt composed of terminal symbols x1 ∈ T

Output:

A CYK matrix C; a set CPR of candidate production rules.

Preconditions:

x is a string that has been parsed by the syntactic analyser yet, Each input element is then associated with a syntactic category n ∈ N0.

Procedure:

(Generate a candidate set of production rules CPR used in step 2)

1. Consider x as the sentence x1, x2….. xn(the multiple combination)

Generate the set P' of production rules that is composed of rules of the form Ai -> xi

2. Continue the following processes for all 1 ≤ i≤ k

i) Initialize a new CYK matrix C(k X k) by

ii) Assign a value

iii)Assign to each cij a set of functions.

3. Iterate the following processes for all 2 ≤ j ≤ k and 1 ≤ i ≤ k-j +1

i) Initialize the element cij = 0

ii) For all q(1≤ q ≤ j - 1)

4. If S ≤ cik then return (success)

Else continue with step 2

Second Phase of the MGI Algorithm-:

During the next phase, whose procedure, the analysis of the structures generated during the first phase is performed, which are the CYK matrix C and set CPR of candidate production rules. In particular, the algorithm selects the candidate derivations with the highest values. Nonterminal symbols, which belong to the set N0, do not need any processing, while those symbols that are created during the first phase for simulating the generation of some productions need to be definitely included into the grammar. Consequently, non-terminals that are part of the production rule inserted into the grammar have to be redefined until all symbols belong to the grammar. Therefore, the output of the MGI algorithm is a new MAG G'' = (G',A',R')

Input:

An input sentence x : x1, x2, xk , A CYK matrix C; A set CPR of candidate production rules; a current multimodal attribute grammar G ={G,A,R} with G = (T',N', P',S')

Output:

A new multimodal attribute grammar G' ={ G',A',R'} with G = (T',N',P',S') and R' = Rp U R'p

Preconditions:

The sentence x does not belong to the language by the current grammar G.

Procedure:

(Generate a candidate set of production rules CPR used in step 2)

1. Select the non-terminal symbol A with the highest weight in the location cin of the CYK matrix.

2. Find the candidate production rule r $\in$ CPR of the form r : A -> BC, containing A in the head, and consider the symbols B and C in the body.

3. Initialize P' <- P0

4. Add the production rules t: S->BC to the set P'

5. Add the production rule t : S -> BC to the set P' Else proceed with step 2

6. Iterate the following processes for all symbols in the body of a production rule: If B(C) is contained in the head of any rule of CPR.

IV. EXPECTED RESULTS

Evaluating grammar inference systems is a critical task for several reasons. First of all, in order to evaluate an inferred grammar, it is necessary to compare it with a "correct" grammar, which is difficult to identify? Second, the ambiguity represents an obstacle as there is no obvious single correct grammar that represents a given set of training examples. These issues have been largely addressed in the literature, and several evaluation metrics have been defined in measuring the correctness of the induced grammar. A brief description of the three main evaluation methods used in NL grammar inference is provided in Section V-A.

ISSN (Online) : 2454-4159
International Journal of Advanced Research in Science Management and Technology
Volume 2, Issue 3, March 2016

The analysis of the advantages and drawbacks of these methods has led to choosing the rebuilding known grammars method in evaluating the proposed grammar inference algorithm due to the simplicity and objectivity of the evaluation. The application of this method and its results are illustrated in Sections V-B and C.

TABLE
EFFECT OF THE MERGE OPERATOR

P1) Sentence → NP AP1		P1) Sentence → NP AP3
P1.1) Sentence.val ← NP.val + AP1.val		P1.1) Sentence.val ← NP.val + AP3.val
P1.2) Sentence.mod← NP.mod+ AP1.mod		P1.2) Sentence.mod← NP.mod+ AP3.mod
P2) Sentence → NP AP2		P2) AP3 → prep NN
P2.1) Sentence.val ← NP.val + AP2.val		P2.1) AP3.val ← prep.val + NN.val
P2.2) Sentence.mod← NP.mod+ AP2.mod	⟹	P2.2) AP3.mod← prep.mod+ NN.mod
P3) AP1 → prep NN		P3) AP3 → CC NP
P3.1) AP1.val ← prep.val + NN.val		P3.1) AP2.val ← CC.val + NP.val
P3.2) AP1.mod← prep.mod+ NN.mod		P3.2) AP2.mod← CC.mod+ NP.mod
P4) AP2 → CC NP		
P4.1) AP2.val ← CC.val + NP.val		
P4.2) AP2.mod← CC.mod+ NP.mod		

VI. CONCLUSION AND FUTURE SCOPE

This paper has proposed a computationally efficient algorithm for grammatical inference, which joins together the strengths of the inductive CYK and e-GRIDS algorithms, adapting them to multimodal sentences. The strength of this algorithm relies on its efficiency, simplicity, and capability of avoiding the over-generalization problem through the introduction of a heuristic based on the minimum description length of the grammar description. The performance of the grammar inference algorithm has been validated through several experiments, which aim to examine its ability to infer a "correct" MAG. The main outcome of these experiments is that the algorithm has an acceptable performance since the inferred grammar has a very high probability (i.e., > 0.97) of parsing valid sentences. A promising research direction for future work can be toward finding new heuristics to avoid the over-generalization problem (without the use of negative examples) or toward improving the developed one. Moreover, the introduction of new learning operators, in addition to the "merge" and "create" operators, can be examined in order to enhance the way the algorithm improves the grammar description.

REFERENCES

[1] H. Shiraz and Y. Takashima, "Multimodal definite clause grammar," Syst. Compute. Jpn., vol. 26, no. 3, pp. 93–102, 1995.

[2] M. Johnston and S. Bangalore, "Finite-state multimodal integration and understanding," Nat. Lang. Eng., vol. 11, no. 2, pp. 159–187, Jun. 2005.

[3] D. Reitter, E. M. Panttaja, and F. Cummins, "UI on the fly: Generating a multimodal user interface," in Proc. HLT-NAACL, Boston, MA, 2004.

[4] Y. Sun, Y. Shi, F. Chen, and V. Chung, "An efficient multimodal language processor for parallel input strings in multimodal input fusion," in Proc. Int. Conf. Semantic Comput., 2007, pp. 389–396.

[5] D. Angluin, "Inference of reversible languages," J. ACM, vol. 29, no. 3, pp. 741–765, Jul. 1982.

[6] P. Garcia and E. Vidal, "Inference of k-testable languages in the strict sense and applications to syntactic pattern recognition," IEEE Trans. Pattern Anal. Mach. Intell., vol. 12, no. 9, pp. 920–925, Sep. 1990.

ISSN (Online) : 2454-4159

International Journal of Advanced Research in Science Management and Technology

Volume 2, Issue 3, March 2016

[7] J. D. Emerald, K. G. Subramanian, and D. G. Thomas, "Learning code regular and code linear languages," in Proc. ICGI, 1996, pp. 211–221.

[8] P. Langley and S. Stromsten, "Learning context-free grammars with a simplicity bias," in Proc. 11th ECML, 2000, pp. 220–228.

[9] K. Nakamura and T. Ishiwata, "Synthesizing context free grammars from sample strings based on inductive CYK algorithm," in Proc. 5th Int. Colloq. Grammatical Inference, 2000, pp. 186–195.

[10] K. Nakamura and M. Matsumoto, "Incremental learning of context free grammars," in Proc. 6th Int. Colloq. Grammatical Inference, 2002, pp. 174–184. [13] K. Nakamura, "Incremental learning of context free grammars by extended inductive CYK algorithm," in Proc. ECML Workshop Learn.

[11] Contex-Free Grammars, 2003, pp. 53–64