Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X

IJCSMC, Vol. 3, Issue. 8, August 2014, pg.491 – 496

RESEARCH ARTICLE

Generating Multimodal Grammar using CYK and e-GRIDS Algorithm

Pankaj Devre[1], Madhuri Bhalekar[2]

[1]M.E. Student, Department of Computer Engineering, MET's Institute of Engg., Nashik, Savitribai Phule Pune University, India

[2]Assistant Professor, Department of Computer Engineering, MAEERS MIT, Pune, Savitribai Phule Pune University, India

[1] pbdevre@gmail.com

[2] madhuri.chatur@gmail.com

Abstract - To communicate with each other human being uses multiple modes like text, speech, hand gesture, facial expressions. The use of theses modes makes human communication more fast and flexible. In previous years several methods are used to bring human computer communication more closely. It makes high costs for developing and maintaining multimodal grammar. Integrating and understanding multiple input interfaces leads to discover the novel algorithm which provides a way to automate the grammar and update the grammar. In this paper, we propose a grammar algorithm that uses multimodal grammar from positive samples of multimodal sentences from multiple inputs. The first phase of the algorithm generates multimodal grammar which is able to parse positive samples of sentences collected from multiple inputs and second phase updates grammar by using learning operators and minimum length metrics for avoiding over generalization problem.

Keywords - Multimodal Grammar, Novel Algorithm, Multimodal Sentence, Learning Operator, Over Generalization

I. INTRODUCTION

HUMAN BEING uses multiple modes of signs, speech, touch, gesture etc. to interact with each other. Interaction amongst people is carried through several communication channels i.e. multimodal communication. In last years several efforts have been made to close the human computer interface by means of simplicity, naturalness and robustness. The objective is towards making human computer communication closer. Therefore, multimodal interfaces allow us to communicate with computers through use of several modes. Multimodal interaction provides the user with a way to interface with a system in both input and output, enabling users to communicate more easily with automated systems. The human-computer communication depends on the possibility of exchanging information through the different communication channels. Therefore, multimodal interfaces, which allow us to communicate with the computer through the simultaneous use of several channels of input/output at a single time, have gained increasing importance in human-computer interaction research. Multimodal grammars provide methodology [1]-[4] for integrating inputs by using multimodal interfaces. In this methodology, the outcomes of each unimodal channels are considered as terminal symbols of the grammar, and consequently, they are recognized by the parser as a unique multimodal sentence. Therefore, in the interpretation phase, the parser uses the specified grammar (production rules) to interpret each multimodal sentence in the input.

II. BACKGROUND

The use of grammatical inference exists 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 context free grammar focused by algorithms for NL grammar inference. There are three existing grammatical inference algorithm discussed here.

The *inductive CYK algorithm [4]* produces CFGs from positive and negative sample strings and generates the minimum production rule, which derives positive strings only. The main advantages of the inductive CYK algorithm is based on the generation of simple sets of rules and less computational time.

The *learning by version space algorithm* [14] needs positive and negative examples. The algorithm applies version space strategy, which is based on a compact way of representing the version grammars and other processes have to choose among them.

The *e-GRIDS algorithm* [15] is a grammar inference method that uses positive sentences in order to construct an initial grammar by converting each one of the training examples into a grammatical rule. Subsequently, the learning process, which is organized as a beam search, takes place. Having an initial hypothesis (the initial grammar) in the beam, e-GRIDS uses three learning operators in order to explore the space of CFGs: the *MergeNT*, *CreateNT*, and *Create Optional NT* operators. The main advantage of the e-GRIDS algorithm is its computational efficiency and scalability to large example sets. Also this algorithm is able to generate grammars, based on relatively small sets of training examples and handles large example sets in a significantly reduced amount of time. The algorithm proposed in this paper combines the capabilities of CYK and e-GRIDS algorithm.

III. GRAMMATICAL REPRESENTATION

In the proposed grammar inference algorithm, multimodal attribute grammars (MAGs) are used, A MAG is a triple $G = (G, A, R)$ Where,

G CFG (T,N,P,S), with T as a set of terminal symbols, N as a set of nonterminal symbols, P as a set of production rules of the form.

$X_0 \rightarrow X_1\, X_2 \ldots Xn$

Where,

$n \geq 1$, $X_0 \in N$ and $X_k \in N \cup T$ for $1 \leq k \leq n$ and $S \in N$ as a start symbol (or axiom);

A collection $(A(X))_{X \in N \cup T}$ of the attributes of the nonterminal and terminal symbols, such that, for each $X \in N \cup T$, $A(X)$ is split in two finite disjoint subsets, namely, $I(X)$ (the set of inherited attributes of X) and $S(X)$ (the set of synthesized attributes). The set $S(X)$, with $X \in T$, includes a set of attributes $MS(X)$, called as a set of multimodal synthesized attributes, composed of the following four attributes:

$MS(X) = \{val, mod, synrole, coop\}$;

R collection $(Rp)p \in P$ of semantic functions (or rules).

1) *val* that expresses the current value (concept) of the terminal symbol. The domain of the attribute is the set of terminal symbols: $D_{val} = T$.
2) *mod* that represents the modality. The domain of the attribute is the set of modalities. $D_{mod} = \{speech, handwriting, gesture, sketch\}$.
3) *synrole* that conveys information about the syntactic role. The domain of the attribute is $D_{synrole} = \{noun\ phrase, verb\ phrase, determiner, verb, noun, adjective, preposition, deictic, conjunction\}$.
4) *coop* that expresses the modality cooperation type with other terminal symbols. The domain of the attribute is $D_{coop} = \{complementary, redundant\}$.

IV. GRAMMAR INFERENCE ALGORITHM

As shown in Fig. 1 proposed algorithm consist of two main phase. The first phase generates production rules and associated semantic functions for multimodal grammar and second phase uses minimum description length method for improving grammar description and avoiding over generalization.

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 : x_1, x_2, x_k , a set T = { x_1, x_2…x_n} of terminal symbols ,
a multimodal attributes grammar G = {G, A, R}. A target sentence x_t composed of terminal symbols $X_1 \in T$

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

Procedure:

1. Consider x as the sentence x_1, $x_{2.....}$ x_k.
   Generate the set P' of production rules that is composed of rules of the form A_i -> x_i
2. Iterate the following processes for all $1 \leq i \leq k$
   i) Initialize a new CYK matrix C(k X k) by
   ii) Assign a weight
   iii) Assign to each c_{ij} a set of semantic functions.
3. Iterate the following processes for all $2 \leq j \leq k$ and $1 \leq i \leq k-j+1$
   i) Initialize the element $C_{ij} = 0$
   ii) For all $q(1 \leq q \leq j - 1)$
4. If $S \leq C_{ik}$ then return (success)
   Else proceed with step 2

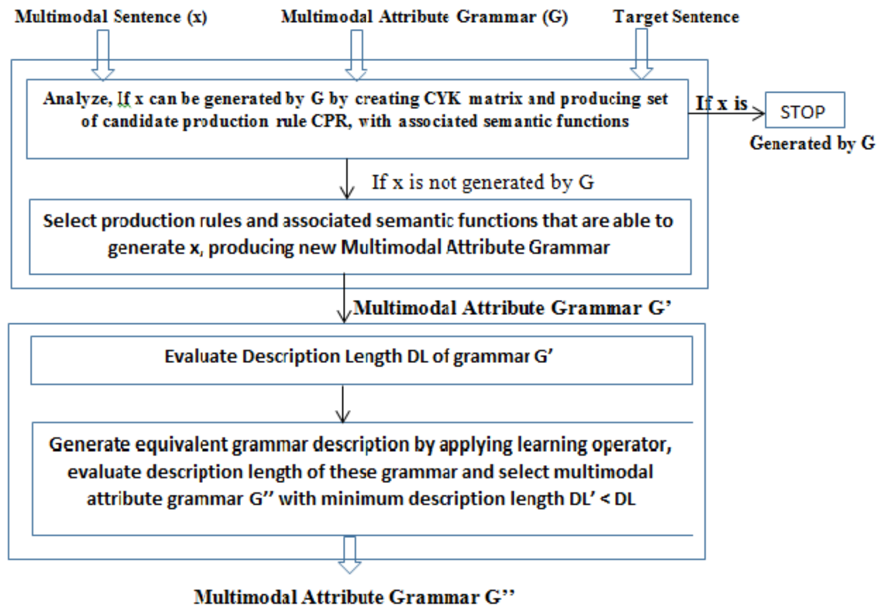

Fig. 1 Workflow of the algorithm

Second Phase of the MGI Algorithm:

During the second phase, the analysis of the generated CYK matrix C and set CPR of candidate production rules performed and new multimodal grammar is generated. Therefore, the output of the MGI algorithm is a new MAG *G'' = (G', A' ,R'),*
Where,

$G' = (T', N', P', S')$
$A' = (A(X))X \in N \cup T$,
R' is the set of semantic functions for evaluating the attributes of $X \in N \cup T$.

Algorithm

Input: An input sentence x : x_1, x_2, x_k , 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*

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 c_{in} of the CYK matrix.
2. Find the candidate production rule r ∈ 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'* <- P_0
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.

Grammar Updating Steps of the MGI Algorithm:

The next step of the MGI algorithm is to update the MAG G', outputted by the first step, by evaluating its description length and by applying to it the learning operators to produce the equivalent grammar descriptions that are more "compact" with respect to the description length of the grammar.

Algorithm

Input: A current multimodal attribute grammar *G' = {G, A, R}* with *G = {T,N,P,S}*, A contains the sets of synthesized attributes *S(x1)* associated with each terminal symbol x_i ∈ *T; R* contains the semantic functions *Rp* for valuating the attribute of non-terminal in the head of some production rules in P'

Output: A new multimodal attribute grammar is G''.

Procedure:

1. Evaluate the description length DL of *G'*
2. Iterate the following processes
   a. For each production p ∈ P, such that p *: A -> BC*
   b. Evaluate DL of the new grammar *G''*
   c. For each production p ∈ P, such that such that BC belongs to the body of p
   d. Evaluate DL of the new grammar *G''*

Description Length of a MAG:

A general principle of statistics that allows us to seek the shortest possible representation of the data expressed through a representation language.

Following the approach proposed in [6], given a CFG G and a set of positive examples E, the description length DL of G is the sum of two independent lengths

DL = GDL + DDL

Where,

GDL Grammar description length, i.e., the bits required to encode the grammar rules and to transmit them to a recipient who has minimal knowledge of the grammar representation;

DDL derivation description length, i.e., the bits required to encode and transmit all examples in set E, provided that the recipient already knows the grammar G.

V. EXPECTED RESULTS

Running of the First Step of the MGI Algorithm:

Consider the multimodal sentence composed of the speech "call that person" and text "Sam". The set T of terminal symbols is composed of the elements of each unimodal sentence, *T = {Call, that, person, Sam}.*

TABLE I
CYK MATRIX

	Call	*That*	*Person*	*Sam*
	1	*2*	*3*	*4*
1	*VB* *VB.val <- Call* *VB.mod<-speech* *VB.synrole<-verb*	*DT* *DT.val <-That* *DT.mod <-speech* *DT.synrole <-deictic* *DT.coop <-compl.*	*NN* *NN.val <-person* *NN.mod<-speech* *NN.synrole<-noun* *NN.coop<-compl*	*NNS* *NNS.val <-atos* *NNS.mod <-text* *NNS.synrole<-noun* *NNS.coop <-compl*
2	*B* *B.val <-cal* *B.mod<-speech*	*C*	*D* *D.val <-Sam* *D.mod <-text*	
3	*E* *E.val<-call* *E.mod<-speech*	*G* *G.val <-Sam* *G.mod <-text*		
4	*I* *I.val <- call Sam* *I.mod <-SpeechText* *L* *L.val <-call Sam* *L.mos<-speech + text* *M* *M.val <- call Sam* *M.mod<-speech_text*			

The initial set of production rules *P'* contains the following rules:
P' = {VB -> Call; DT ->that; NN->person; NNS>Sam}.
Furthermore, the set of candidate production rules CPR contains the following rules:
CPR = {B -> VB DT; C -> DT NN;D -> NN NNS; E-> VB C; F ->B NN;
G -> DT D;H -> C NNS; I -> VB G;L -> B D; M -> E NNS};

In proposed system to evaluate multimodal grammar algorithm first phase generates CYK matrix as shown in TABLE 1. And set of candidate production rule.

Improving the Grammar Description to Avoid the Over-Generalization Problem. The goal of the second step of the MGI algorithm is to update the *MAG G'*, outputted by the first step, by evaluating its description length .The second step of the grammar inference method, named as the grammar-updating step, works in the following way.
It takes as input the *MAG G' = {G', A', R'}* generated during the first step

Multimodal Attribute Grammar (Expected)

Here In proposed system following expected multimodal grammar has been shown by using speech and text modality.

P1) S -> NP VERB
R1.1) S.val<-NP.val +VERB.val
R1.2)S.mod<-NP.mod+VERB.mod
P2) S->VP NP
R2.1)S.val<-VERBT.val
R2.2)S.mod<-VP.mod+NP.mod
P3) VP->VERBT
R3.1) VP.val<-VERBT.val
R3.2) VP.mod<-VERBT.mod
P4) VERBT-> call
R4.1) VERBT.val<- call
R4.2) VERBT.mod <- speech
R4.3) VERBT.synrole<- verb
P5) NOUN -> Person
R5.1) NOUN.val<- person
R5.2) NOUN.mod<- speech
R5.3) NOUN.synrole<- noun
R5.4) NOUN.coop<- complementary
P6) NNP1 -> Sam
R6.1)NNP1.val<-Sam
R6.2)NNP1.mod<-text
R6.3)NNP1.synrole<-Sam
R6.4)NNP1.coop<-complementary

VI. CONCLUSIONS

Multimodal interaction has risen in the last few years as the future of human-computer interaction. This fact is increasingly inferring computer interfaces making computer behavior closer to human communication. Multimodal interaction requires that multiple (more than one) inputs, coming from various input modalities that are combined into a complete sentence. In order to overcome the deficiencies of the grammar based paradigm, this paper proposed an approach of grammar definition that follows the "by example" paradigm, that is, the language designer provides training examples of multimodal sentences that have to be recognized, and a grammar inference algorithm generates the grammar rules to parse those examples. In such a way no expert grammar writers are needed, but even non-expert users can define multimodal grammars.

REFERENCES

[1] Arianna D'Ulizia, Fernando Ferri, Patrizia Grifon, *A Learning Algorithm foR Multimodal Grammar Inference*" SYSTEMS, MAN, AND YBERNETICS VOL. 41,NO. 6, DECEMBER 2011

[2] 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.

[3] K. Nakamura and M. Matsumoto, *"Incremental learning of context free grammars*," in Proc. 6th Int. Colloq. Grammatical Inference, 2002, pp. 174-184.

[4] K. Nakamura, "*Incremental learning of context free grammars by extended inductive CYK algorithm*," in Proc. ECML Workshop Learn. Contex-Free Grammars, 2003, pp. 53-64.

[5] K. Vanlehn and W. Ball, "*A version space approach to learning contextfree grammars*," Mach. Learn., vol. 2, no. 1, pp. 39-74, 1987.

[6] G. Petasis, G. Paliouras, V. Karkaletsis, C. Halatsis, and C. D. Spyropoulos, "*e-GRIDS: Computationally efficient grammatical inference from positive examples,"* GRAMMARS, vol. 7, pp. 69-110, 2004.

[7] J. Rissanen, "*Modeling by shortest data description*," *Automatica, vol. 14,no. 5, pp. 465- 471, Sep. 1978.*

[8] A. D'Ulizia, F. Ferri, and P. Grifoni, "Generating multimodal grammars for multimodal dialogue processing," IEEE Trans. Syst., Man, Cybern. A, Syst., Humans, vol. 40, no. 6, pp. 1130-1145, Nov. 2010.

[9] D. E. Knuth, "*Semantics of context-free languages*," Theory Comput. Syst., vol. 2, no. 2, pp. 127-145, Jun. 1968

[10] J. K. Baker, "*Trainable grammars for speech recognition*," in Proc. Speech Commun. Papers 97th Meeting Acoust. Soc. Amer., 1979, pp. 547–550.

[11] P. Adriaans, "*Language learning from a categorical perspective*," Ph.D. dissertation, Univ. van Amsterdam, Amsterdam, The Netherlands, 992

[12] Y. Sakakibara, "*Recent advances of grammatical inference*," Theor. Comput. Sci., vol. 185, no. 1, pp. 15–45, Oct. 1997.

[13] C. de la Higuera and J. Oncina, "*Identification with probability one of stochastic deterministic linear languages*," in Proc. ALT, 2003, pp. 134 – 148.

