New Probabilistic Model of Stochastic Context-Free Grammars in Greibach Normal Form

Yevhen Hrubiian
Student, group FI-33, Institute of Physics and Technology
NTUU "Igor Sikorsky Kyiv Polytechnic Institute"
Kyiv, Ukraine
grubian.euhen@gmail.com

Імовірнісна Модель Стохастичних Контекстно-Вільних Граматик в Нормальній Формі Грейбах

Євген Грубіян
Студент групи ФІ-33, Фізико-технічний інститут
НТУУ "Київський політехнічний інститут ім. Ігоря Сікорського"
Київ, Україна
grubian.euhen@gmail.com

***Abstract*—In this paper we propose new probabilistic model of stochastic context-free grammars in Greibach normal form. This model might be perspective for future developments and researching stochastic context-free grammars.**

***Анотація*—В роботі запропоновано нову імовірнісну модель стохастичних контекстно-вільних граматик в нормальній формі Грейбах. Ця модель може бути перспективною для подальших досліджень стохастичних контекстно–вільних граматик.**

Keywords—stochastic context-free grammar; Greibach normal form; hidden Markov models

Ключові слова—стохастична контекстно-вільна граматика; нормальна форма Грейбах; приховані моделі Маркова.

I. INTRODUCTION

Today, there is a huge interest in developing new methods of natural language processing due to a lot of issues from cryptanalysis, biology and artificial intelligence. Several ways for investigation had been proposed since works of Noam Chomsky in the mid 50s that originated development of modern formal theory of syntax. His conception of generative grammars and context-free grammars became one of the most discussing and controversy in linguistics but these ideas lay behind development of formal theory of programming languages and compilers later. Context-free grammars were proposed firstly for describing grammar of English or any other language but this approach does not cover all features of language such as semantics. Later, head-driven grammars were introduced to describe some valuable properties of languages such as semantic dependencies between words. Probabilistic models of languages were proposed by Claude Shannon to describe informational aspects of languages such as entropy but they became very relevant and actual in modern linguistics. N-gram model is one of the most accurate statistical models of language but the main drawback is that for precious learning this model for high values of N one needs a huge amount of texts in different thematic, but unfortunately, even if we have got a lot of texts it turns out to be harmful for accurate and precise modeling whole language. Because of this reason stochastic context-free grammars were introduced for more precious description of languages because context-freeness guarantees grammatical pattern of language and probability distributions on grammatical rules approach semantic structure of phrases. Stochastic context-free grammars are also useful in modeling DNA sequences, image recognition and modeling plain texts in cryptography so that investigation of stochastic context-free grammars in very important and actual in today's science.

In this paper we consider stochastic context-free grammars in Greibach normal form although significant contributions in natural language processing was associated with algorithms of learning stochastic context-free grammars in Chomsky normal form but complexity of these methods is quite high. We propose new idea of investigation of stochastic context-free grammars based on our conception of hidden Markov models with stack.

II. STOCHASTIC CONTEXT-FREE GRAMMARS

Conception of stochastic context-free grammars (SCFG) came from several problems in natural language processing such as finding the most probable sequence of the words,

learning unknown grammar from corpora of texts etc. But NLP is not only sphere of applications of SCFG, several biologists and geneticists proposed usage SCFG for modeling DNA sequences, modern pattern recognition, speech processing, image processing also use this conception.

We say that G = <N, Σ, R, S, P> is *stochastic context-free grammar* if and only if Q = <N, Σ, R, S> is context-free grammar (N – alphabet of nonterminal symbols or variables, Σ – alphabet of terminal symbols, R – set of production rules, S – start symbol) and one defines family of probability distributions P on R so that for every production in R there is probability of applying this production rule when forming some sequence of symbols such that sum of probabilities of productions starting with same symbol must be equal to 1.

There are several normal forms of grammars. The most popular are Chomsky normal form and Greibach normal form.

We say that context free grammar G = <N, Σ, R, S> is in *Chomsky normal form* (CNF) if and only if all production rules have form:

$$A \rightarrow BC$$

$$A \rightarrow a$$

Our major interest forms Greibach normal form.

We say that context free grammar G = <N, Σ, R, S> is in *Greibach normal form* (GNF) if and only if all production rules have form:

$$A \rightarrow aA_1A_2...A_n$$

Where A is nonterminal, a is terminal symbol and $A_1A_2...A_n$ is a probable empty sequence of nonterminal symbols except S.

The definition of SCFG in GNF might get stronger if we set number of nonterminal symbols in right hand side of any production to be not larger than 2. Productions of such grammar will be in any of the following forms:

$$A \rightarrow aA_1A_2$$

$$A \rightarrow aA$$

$$A \rightarrow a$$

It could be shown that every SCFG in GNF might be transformed into equivalent grammar in the stronger GNF. For this reason in this paper we suppose that every Greibach normal form will be in the form described above.

Every context-free grammar could be transformed to equivalent grammar in Greibach normal form in polynomial time such that languages they form are equal. The same idea is behind transforming SCFG to SCFG in Greibach normal form. We obtained algorithm for this transformation based on existing algorithm of transforming an ordinal CFG in Chomsky Normal form to context-free grammar in Greibach normal form adding several steps of rebalancing probability distributions P.

Here is the algorithm:

1. Eliminate null productions, unit productions and useless symbols from the grammar G and then construct a $G_0 = (V_0, T, R_0, S)$ in Chomsky Normal Form (CNF) generating the language $L(G_0) = L(G) - \{\varepsilon\}$.
2. Rename the variables like $A_1, A_2, \ldots A_n$ starting with $S = A_1$.
3. Modify the rules in R so that if $A_i \rightarrow A_j\gamma \in R_0$ then $j > i$
4. Starting with A_1 and proceeding to An this is done as follows:
   a) Assume that productions have been modified so that for $1 \le i \le k$, $A_i \rightarrow A_j\gamma \in R_0$ only if $j > i$
   b) If $A_k \rightarrow A_j\gamma$ is a production with $j < k$, generate a new set of productions substituting for the A_j the body of each A_j production. The transfer probabilities should split uniformly from base probability.
   c) Repeating (b) at most $k - 1$ times we obtain rules of the form $A_k \rightarrow A_p\gamma$, $p \ge k$
   d) Replace rules $A_j \rightarrow A_k\gamma$ by removing left-recursion as stated above. Transfer probabilities should also be splitted.
5. Modify the $A_i \rightarrow A_j\gamma$ for $i = n - 1, n - 2, ., 1$ in desired form at the same time change the Z production rules.

III. Scoring problem for regural grammars

Suppose we have to find probability of sequence of symbols GNF generated: $Pr(w_{1:n} \mid G)$ i.e. solve the scoring problem.

Trivial situation come up when our grammar in GNF is regular, i.e. all rules have form:

$$A \rightarrow aB$$

Regular grammars are equivalent to finite state machines (FSM). This equivalence could be set if we assume that set of states of FSM is equal to set of nonterminal symbols and every transition is described by corresponding rule of grammar with emission of terminal symbol.

We say that bivariate stochastic process $\{X_i, O_i, i=1,2,\ldots\}$ is *hidden Markov model of order n by emissions (HMMn)* if

$$Pr(X_k \mid X_{1:k-1}) = Pr(X_k \mid X_{k-1})$$

$$Pr(O_k \mid X_{1:k}) = Pr(O_k \mid X_{k-n+1:k})$$

X_k are called latent variables or hidden states of HMM and O_k are called observations.

Stochastic regular grammars are equivalent to hidden Markov models of order 2 by emissions. It follows from the fact that set of observed values corresponds to set of terminal symbols and set of latent variables corresponds to set nonterminal symbols of grammar. Second order of models is followed from fact that observed word depends on previous 2 states, however order by transitions is still first. This equivalence also requires independence of appliance every grammatical rule. Because of this fact we could apply *forward* algorithm for solving scoring problem for stochastic regular grammar. Complexity of this algorithm is $\Theta(nm^2)$, where n is

length of observed sequence and m is number of nonterminal symbols.

IV. MARKOV CHAINS WITH STACK

The case of general SCFG in GNF is much more complicated. If we try to build HMM out of SCFG in GNF we will face with some uncertainty with description of latent variables set. Naïve approach is based on setting equivalence between set of all possible sequences of nonterminal symbols that grammar could produce during inference all possible sentences. But in that case number of latent states could be very large and even unbounded because of all recursions in grammatical rules.

Let's try to overcome these difficulties. It is known fact that every context-free grammar is equivalent to some finite state machine with stack (pushdown automaton). Hence we could propose conception of *hidden Markov model with stack* for emulation SCFG in GNF.

We say that stochastic process $\{S_t,\ t=1,\ldots\}$ is a *Markov chain with stack M* if and only if:

$$Pr(S_t \mid S_{1:t-1},\ M_t) = Pr(S_t \mid S_{t-1},\ M_t)$$

Where M_t is element on the head of the stack at time moment *t*.

Finally, we say that bivariate stochastic process $\{S_i, O_i,\ i=1,2,\ldots\}$ is *hidden Markov model with stack M of order n by emissions* if and only if for any k>0 holds:

$$Pr(S_k \mid S_{1:k-1},\ M_k) = Pr(S_k \mid S_{k-1},\ M_k)$$

$$Pr(O_k \mid M_k,\ S_{1:k}) = Pr(O_k \mid M_k,\ S_{k-n+1:k})$$

As we can see, this definition differs from the origin definition of hidden Markov model because of M_k – head of the stack in conditional part of emission and transition probabilities.

In fig. 1. there is example of *Markov chain with stack*:

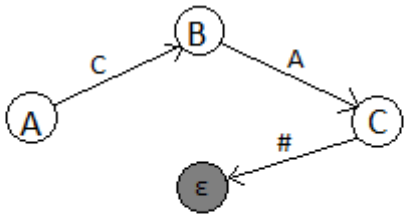

Fig. 1. Markov chain with stack

Here A,B,C denote ordinary states and ε denotes special symbol used for sudden transition to state pointed by top of the stack *M*. Letters above arrows denote states to push into stack. Note that for every transition not more than one state could be pushed into stack. This fact corresponds to conception of the *stronger GNF* described in the second chapter. Symbol # denotes popping out head of stack to which chain jumps after. Transitional probabilities are omitted because the chain in example is deterministic. If start state is A than chain will produce sequence ABCABCABC…

One could notice that stack size in this example is growing unrestrictedly but it is not common case for natural language's grammars. The model terminates when stack is empty and model reaches # symbol.

Our goal than will be setting up equivalency between stochastic context-free grammars in Greibach normal form and hidden Markov models of order 2 with stack (HMM2S).

V. SCFG IN GNF AND HMM2S EQUIVALENCY

In previous paragraph we could see that Markov chains with stack work just similar to the way stochastic context-free grammars in Greibach normal form infers sequences of nonterminal symbols. Indeed, if we consider state space of Markov chain with stack to be the set of nonterminal symbols in SCFG in GNF and transitions between states would show action of the rules of grammar in such way that first nonterminal symbol in right hand side of the rule would be state that the Markov chain with stack jumps to and letters above arrows are the rest of right hand side of the rule than we could negotiate equivalency between nonterminal inferences in stochastic context-free grammars in Greibach normal form and Markov chains with stack.

A bit tricky step is construction of Hidden Markov model with stack that considers terminal productions in each step of inference some sentence. We need to deal with ambiguity of grammatical rules that could produce different terminal symbols but transfer to the same state. This problem could be solved by adding dependence to emitted terminal symbol by current state and previous. Hence we must build Hidden Markov Model of the second order. Also we should take to consideration that in general case of stochastic context-free grammar in Greibach normal form the first letter of each production would not identify uniquely state the system jumps. To solve this we propose to associate leftmost different sequences of nonterminal symbols to states of HMM. For example if we have rules:

$$A \rightarrow aCD$$

$$B \rightarrow bCE$$

We will than have got states {A, B, CD, CE} for our HMM. In this case for each emission there would be no ambiguity to determine which grammatical rule has produced it. Note that in every step of production there is emission of terminal symbol. If some production in the form:

$$A \rightarrow a$$

is applied the system will pop out the head state in the stack and will jump to that state. Size of the stack will reduce than.

Thus we have obtained the hidden Markov model of the second order.

VI. Open problems, ways of solving and motivation for this model

As we could see, stochastic context-free grammars in Greibach normal form are equivalent to Hidden Markov models of the second order with stack. So that we could develop algorithms for finding particular sentence probability, probability of sequence of latent states (grammatical rules applied) and even algorithms for learning stochastic context-free grammars. The main idea why construction proposed in this article might helpful is the fact that there is a lot of algorithms developed for ordinary hidden Markov models and they are quite scalable and easily transformable for appliance for more complex Markov models.

For example for finding most probable latent state sequence there is Viterbi algorithm. For finding observed sequence probability there is forward algorithm. For learning model parameters there is Baum-Welch algorithm based on forward-backward algorithm. Complexity of these algorithms is $O(nm^2)$ which is quite better than complexity of algorithms for stochastic context-free grammar in Chomsky normal form. For example the algorithm inside-outside for learning grammar in Chomsky normal form has complexity $O(n^3m^3)$. This is the reason why equivalence of stochastic context-free grammars in Greibach normal form and hidden Markov models of the second order with stack may be very perspective. However, there might be a lot of work in developing theory of hidden Markov models with stack we believe that this ideas will be helpful for further researching in natural language processing, biology and computer science.

VII. Acknowledgment

Author conveys a big thanks to his supervisor Andriy Fesenko, collective of cathedra of Mathematical methods of informational security in Institute of Physics and Technology and Georgii Riabov, fellow researcher at the Department of the theory of stochastic processes, Institute of Mathematics, NAS of Ukraine.

References

[1] http://www.stat.berkeley.edu/%7Eterry/Classes/s246.2002/Week9/week9b.pdf

[2] Rafael C. Carrasco and Jose Oncina, “Learning deterministic regular grammars from stochastic samples in polynomial time”, Departamento de Lenguajes y Sistemas Inform´aticos, Universidad de Alicante, E-03071 Alicante

[3] Zoubin Ghahramani “An Introduction to hidden Markov models and Bayesian networks”, Gatsby Computational Neuroscience Unit, University College London, London WC1N 3AR, England

[4] K. Lari; S. J. Young, (1990). "The estimation of stochastic context-free grammars using the inside-outside algorithm". Computer Speech and Language.

[5] K. Lari; S. J. Young, (1991). "Applications of stochastic context-free grammars using the inside-outside algorithm". Computer Speech and Language.

[6] J. E. Hopcroft; J. D. Ullman, (1979). Introduction to Automata Theory, Languages, and Computation. Addison-Wesley.

[7] C. de la Higuera. Ten open problems in grammatical inference. In Sakakibara et al. [54], pages 32–44

[8] R.L. Kashyap: Syntactic decision rules for recognition of spoken words and phrases using stochastic automaton. IEEE Trans. on Pattern Analysis and machine Intelligence 1(1979) 154–163

[9] Y. Sakakibara, M. Brown, R. Hughley, I. Mian, K. Sjolander, R. Underwood, D. Haussler: Stochastic context-free grammars for tRNA modeling. Nuclear Acids Res. 22 (1994) 5112–5120

[10] Diego Linares, Jose-Miguel Bened, and Joan-Andreu Sanchez “A Hybrid Language Model based on Stochastic Context-free Grammar“, Pontificia Universidad Javeriana – Cali Calle 18 No. 118-250 Av. Canasgordas, Cali (Colombia), DSIC- Universidad Polítecnica de Valencia Camino de Vera s/n, 46022 Valencia (Spain)

[11] Jose L.Verd Verdu-Mas, Jorge Calera-Rubio, and Rafael C. Carrasco “Offspring-annotated probabilistic context-free grammars”, Departamento de Lenguajes y Sistemas Informaticos Universidad de Alicante, E-03071 Alicante, Spain

[12] http://www.iitg.ernet.in/gkd/ma513/oct/oct18/note.pdf

[13] R. Lawrence, A. Rabiner. “Tutorial on hidden Markov models and selected applications in speech recognition”. Proceeds of the IEEE, vol.77, no.2, February1989.

[14] Noam Chomsky, (1956). "Three models for the description of language.". IRE Transactions on Information Theory.

[15] Baum, L. E.; Petrie, T.; Soules, G.; Weiss, N. (1970). "A Maximization Technique Occurring in the Statistical Analysis of Probabilistic Functions of Markov Chains". The Annals of Mathematical Statistics.

[16] Chomsky, Noam (June 1959). "On certain formal properties of grammars". Information and Control

[17] Sheila Greibach, (January 1965). "A New Normal-Form Theorem for Context-Free Phrase Structure Grammars". Journal of the ACM. 12

[18] Norbert Blum, Robert Koch, (1999). "Greibach Normal Form Transformation Revisited". Information and Computation. 150

[19] Michael A. Harrison (1978). Introduction to Formal Language Theory. Addison-Wesley.

[20] L.E. Baum, T. Petrie, (1966). "Statistical Inference for Probabilistic Functions of Finite State Markov Chains".

[21] D. Jurafsky, J.H. Martin, (2009). Speech and language processing: An introduction to natural language processing, computational linguistics, and speech recognition. Upper Saddle River, N.J: Pearson Prentice Hall.