A Novel Machine Translation Method Based on Stochastic Finite Automata Model for Spoken English

https://doi.org/10.3991/ijet.v14i06.10161

Huiyan Li
Ganzhou Teachers College, Ganzhou, China
lhyenglish@163.com

Abstract—Stochastic finite automata have been applied to a variety of fields, machine translation is one of them. It can learn from data and build model automatically from training sets. Stochastic finite automata are well adapted for the constrained integration of pairs of sentences for language processing. In this paper, a novel machine translation method based on stochastic finite automata is proposed. The method formalized rational grammars by using stochastic finite automata. Through given pairs of source and target utterances, our proposed method will produce a series of conventional rules from which a stochastic rational grammar would be inferred, and the grammar is finally converted into a finite state automaton. The efficacy and accuracy of our proposed method is evaluated by a large number of English-Chinese and Chinese-English machine translation experiments.

Keywords—Finite state models, machine translation, rational grammars, stochastic finite automata

1 Introduction

In mathematics and computer science, stochastic finite automata had been introduced to resolve problems of pattern recognition, language processing and so on. Stochastic finite automata have become a useful mechanism for language processing because of its several advantages. There were some algorithms proposed and used in grammatical inference based on stochastic finite state automata [1-6].

The most important reason of using stochastic finite machine for language translation is because that can be learned automatically from training datasets. So there are many regular grammars from finite state sets. Some of them are based on formal language theory which can be built by inferring simple grammars that recognize languages [7-9].

In this paper, a method using stochastic finite automata to learn rational grammar for language translation is proposed. The method can produce inferring finite-state transducer, which can perform the sentence-level and phrase-level language translation very well.

The rest organization of the paper is as follow. The basic definitions and notations are presented in Section 2. In Section 3, a novel method based on stochastic finite automata model for spoken English translation is proposed. The experiments and analysis are given in Section 4. Finally, Section 5 is devoted to conclusions.

2 Definition and Notations

2.1 Stochastic finite automata

Stochastic Finite Automata (SFA) can be represented by a tuple $(Q, q_0, \Sigma, \Delta, F, \delta)$, where Q represents a finite set of states; q_0 represents the initial state of Q; Σ and Δ is the source alphabet and target alphabet respectively. $F \subseteq Q$ represents the set of final states; $\delta: Q \times \Sigma \rightarrow Q$ represents a transition function, and $\delta \subseteq Q \times \Sigma \times \Delta^* \times Q$ represents a set of transitions, where Δ^* represents the set of finite-length strings on Δ, Σ^* is the same. An example of transition function δ is $\delta(q, \gamma) = q$, $\delta(q, aw) = \delta(\delta(q, a), w)$

A translation establishes Φ of length L in SFA can be defined as a sequence of transitions as follows:

$$\Phi = (q_0^{\Phi}, s_1^{\Phi}, t_1^{-\Phi}, q_1^{\Phi})(q_1^{\Phi}, s_2^{\Phi}, t_2^{-\Phi}, q_2^{\Phi})\ldots(q_{L-1}^{\Phi}, s_L^{\Phi}, t_L^{-\Phi}, q_L^{\Phi}) \tag{1}$$

where $(q_{i-1}^{\Phi}, s_i^{\Phi}, t_i^{-\Phi}, q_i^{\Phi}) \in \delta$, $q_0^{\Phi} = q^0$ and $q_L^{\Phi} = F$.

A pair $(s, t) \in \Delta^* \times \Sigma^*$ can be called translation pair if and only if there is a translation form Φ of length L in SFA. $d(s, t)$ represents translation set correlated with the pair (s, t) in SFA.

Function δ can denote transition probabilities, and it is the embodiment of stochastic. A set of strings can be defined as X, $p(X) = \sum_{x \in X} p(x)$. The probability of a string x can be defined $p(x) = p(q_0, x)$, the probability of a string x beginning from the state q is $p(x) = \delta(q, x)$.

A rational translation is the set of all translation pairs of SFA. So $T \subseteq \Delta^* \times \Sigma^*$ is a rational translation if and only if there is an alphabet Ω, a regular language $\psi \subset \Omega^*$, and two morphisms $h_\Sigma: \Omega^* \rightarrow \Sigma^*$ and $h_\Delta: \Omega^* \rightarrow \Delta^*$. T can be expressed as $\{(h_\Sigma(x), h_\Delta(x)) | x \in \psi\}$.

2.2 Statistical machine translation

Statistical machine translation (SMT) is a machine translation paradigm where translations are generated on the basis of statistical models whose parameters are derived from the analysis of bilingual text corpora [10].

According to the probability distribution $p(f|e)$ which can be modeled by the SFA, a string e in the target language (for example, English) can be translated to a string f in the source language (for example, China). So the language translation problem changed into pick up the one that gives the highest probability. Finding the best translation can be defined as follows:

$$\hat{e} = \arg\max_{e \in \Delta^*} p(e \mid f) = \arg\max_{e \in \Delta^*} p(f \mid e)\, p(e) \tag{2}$$

where $\hat{e} \in \Delta^*$ denotes a target string which is translation of a source string f in Σ^*.

A stochastic finite-state language transducer L_p can be defined as a tuple $(Q, q_0, \Sigma, \Delta, p, f)$, where the definitions of Q, q_0, Σ, Δ are the same as those in stochastic finite automata, function $p: Q \times \Sigma \times \Delta^* \times Q \to [0,1]$ and function $f: Q \to [0,1]$ which must meet the following requirements:

$$f(q) + \sum_{(a,w,q') \in \Sigma \times \Delta^* \times Q} p(q,a,w,q') = 1 \tag{3}$$

The transitions set of SFA is the set of tuples (q, s, t, q') in L_p whose probabilities are greater than zero, and the final states set are those whose final-state probabilities are nonzero.

The sum of the probabilities of total translations in L_p can be expressed as follows:

$$P_{L_p}(s,t) = \sum_{\Phi \in d(s,t)} P_{L_p}(\Phi) \tag{4}$$

where the probability of a translation $P_{L_p}(\Phi)$ can be defined as follows

$$P_{L_p}(\Phi) = \prod_{i=0}^{L} p(q_{i-1}, s_i, t_i, q_i) \cdot f(q_L) \tag{5}$$

If $P_{L_p}(s,t) = 0$, it means that there is no translation for (s,t) in SFA, the following equation can be gotten:

$$\sum_{(s,t) \in \Sigma^* \times \Delta^*} P_{L_p}(s,t) = 1 \tag{6}$$

Through stochastic finite-state language transducer L_p, the translation of a source string s can be expressed as follows:

$$\hat{e} = \arg\max_{e \in \Delta^*} P_{L_p}(s,t) \tag{7}$$

The source and target regular languages can be denoted by P_s and P_t ,respectively.

$$P_s(s) = \sum_{t \in \Delta^*} P_{L\,p}(s,t) \tag{8}$$

$$P_t(t) = \sum_{s \in \Sigma^*} P_{L\,p}(s,t) \tag{9}$$

It is a difficult to find an optimal solution for Equation (7), the approximate solution can be achieved on the basis of the following approximation [12]:

$$P_{L_P}(s,t) \approx V_{L_p}(s,t) = \max_{\Phi \in d(s,t)} P_{L_p}(\Phi) \tag{10}$$

So the approximate translation could be expressed as:

$$\hat{e} = \arg\max_{t \in \Delta^*} V_{L_p}(s,t) = \arg\max_{t \in \Delta^*} \max_{\Phi \in d(s,t)} P_{L_p}(\Phi) \tag{11}$$

The Equation (11) can be computed efficiently by solving the following recurrence:

$$\max_{t \in \Delta^*} V_{L_p}(s,t) = \max_{q \in Q}(V(|s|,q) \cdot f(q)) \tag{12}$$

$$V(i,q) = \max_{q' \in Q, w \in \Delta^*}(V(i-1,q') \cdot p(q',s_i,w,q)) \quad \mathit{if}\ i \neq 0, q \neq q_0 \tag{13}$$

$$V(0,q_0) = 1 \tag{14}$$

The final approximate translation $\hat{e}$ will be expressed with the following translation form, which is a series of target strings.

$$\tilde{\Phi} = (q_0,s_1,t_1,q_1)(q_1,s_2,t_2,q_2)...(q_{L-1},s_{L-1},t_L,q_L) \tag{15}$$

An example of translation based on statistical machine is shown in Figure 1.

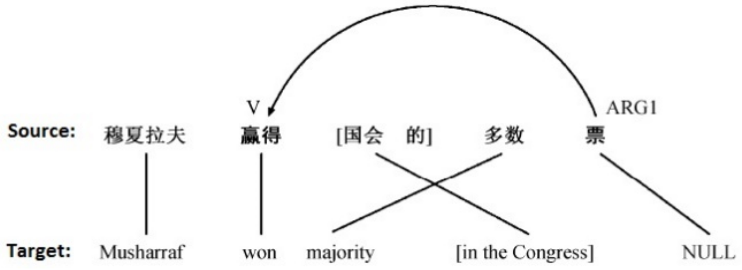


Fig. 1. An example of translation based on statistical machine

The corresponding translation with finite-state transducers is shown in Figure 2.

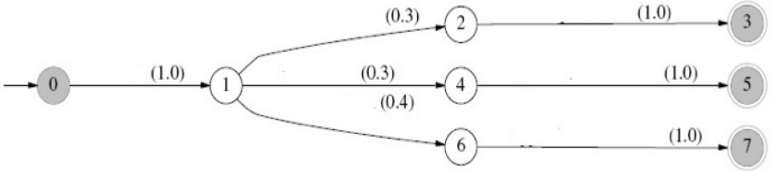


Fig. 2. The corresponding translation with finite-state transducers

3 A Machine Translation Method Based on Stochastic Finite Automata

In this paper, a machine translation method based on verb choice bias is put forward to introduce the preference of verbs to the object in the process of translation, which helps translation system to improve the accuracy of the selection of object candidates. We use conditional probability method based on stochastic finite automata to learn automatically from corpus and get the preference of verb to object.

We suggest the following steps for learning a stochastic finite automata transducer. Firstly, a finite set of string pairs $(s,t) \in \Delta^* \times \Sigma^*$ is given, each string pairs (s,t) will be transformed into a string z by alphabet Γ, which yield a set of strings $S \subset \Gamma^*$. Then a stochastic regular grammar can be inferred from S. Next, the phrase pairs of the target verb phrase pair and the source verb and the target object are extracted respectively. The grammar rules will be transformed back into pairs of the form of symbols-strings again. Finally, based on the conditional probability method, the choice bias and cross semantic choice bias of the verbs is trained respectively, and the preference of the verb is added into the decoding process. The whole scheme can be shown in Figure 3.

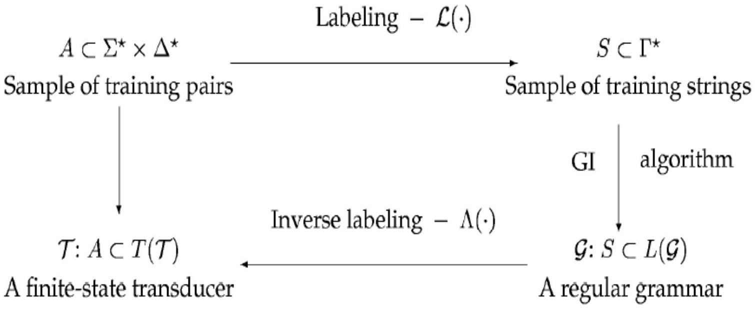

Labeling – $\mathcal{L}(\cdot)$
$A \subset \Sigma^\star \times \Delta^\star$
Sample of training pairs
$S \subset \Gamma^\star$
Sample of training strings
GI
algorithm
Inverse labeling – $\Lambda(\cdot)$
$\mathcal{T}: A \subset T(\mathcal{T})$
A finite-state transducer
$\mathcal{G}: S \subset L(\mathcal{G})$
A regular grammar

Fig. 3. The scheme for the proposed method

The first step of the transformation process can be modeled by the labeling function $\zeta: \Delta^* \times \Sigma^* \rightarrow \Gamma^*$, and the inverse transformation is labeled $\Lambda(\cdot)$,which consists of a set of morphisms h_Σ, h_Δ. So for a string $z \in \Gamma^*$, $\Lambda(z) = (h_\Sigma(z), h_\Delta(z))$.

Because Λ is typically the inverse of ζ, the h_Σ and h_Δ will be determined by ζ in our method. So the transformation of corpus is the most crucial step, and a simple inverse labeling mechanism must be designed.

3.1 Semantic choice bias based on conditional probability

First, the choice bias of the monolingual meaning is calculated only at the target side. It can be defined: a verb v in the corpus under semantic relations r, the noun n as the parameter of v would reflect the possibility of selection bias strength. It can be evaluated by $\hat{P}(n|v,r)$:

$$\hat{P}(n \mid v,r)=\frac{f(v,r,n)}{f(v,r)} \tag{16}$$

where $f(v,r)$ represents the frequency of occurrence of verb *v* in the corpus, $f(v,r,n)$ represents the frequency of co-occurrence of verb *v* and noun *n* under semantic relations r.

We can map the semantic relation *r* into verb-object relation. The selection bias of verb to object can be defined as SP_t:

$$SP_t=\frac{f(v_t,n_t)}{f(v_t)} \tag{17}$$

where $f(v_t)$ represents the frequency of occurrence of verb v_t in the corpus, $f(v_t,n_t)$ represents the frequency of co-occurrence of verb v_t and object n_t under the verb-object relationship.

The calculation method of the choice bias eigenvalue in the translation interval *(i, j)* can be expressed as follows:

$$S_{sp}=\log\prod_{i=1}^{N}P_i \tag{18}$$

where *N* represents the number of pairs of phrase (verb, object) in the current translation interval, P_i represents the choice bias of the monolingual meaning which was trained through conditional probability method.

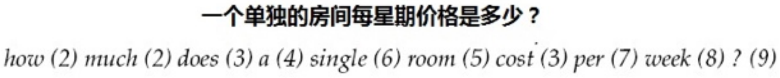
3.2 Statistical alignment

Our machine translation method is combined with the alignments between source and target words. We can define the function $b:\{1,\ldots,|t|\}\rightarrow\{0,\ldots,|s|\}$ as the alignment of a string pair $(s,t)\in\Delta^*\times\Sigma^*$, the $b(j)=0$ denotes that the position *j* of string *t* is not match any position in string s. $B(s,t)$ represents all the string matching between string *t* and string *s*. $P_r(t,b|s)$ represents the probability of translating string s into string t by a given alignment *b* [13].

The alignment between string *s* and string *t* can be obtained as follows:

$$\widehat{b}=\underset{b\in B(s,t)}{\arg\max}\Pr(t,b\mid s) \tag{19}$$

An example of Chinese-English sentence alignment is given as shown in Figure 4.

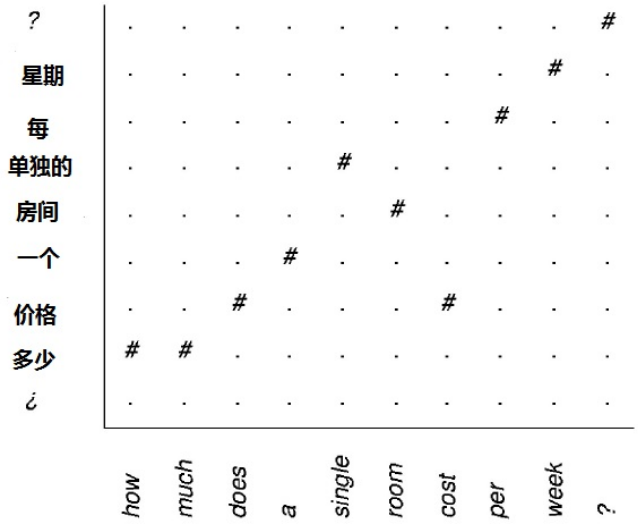


一个单独的房间每星期价格是多少？

how (2) much (2) does (3) a (4) single (6) room (5) cost (3) per (7) week (8) ? (9)

Fig. 4. An example of Chinese-English sentence alignment

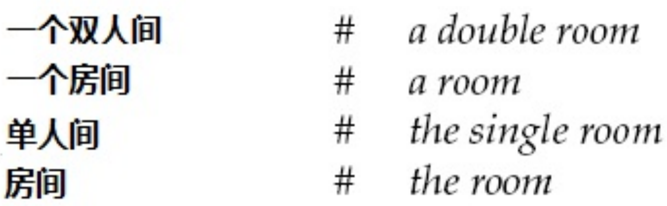
Each number in parentheses of the example denotes the position in the source string which is matched the position of target string. The graphical representation of the alignment is illustrated in Figure 5.



	how	much	does	a	single	room	cost	per	week	?
?	.	.	.	.	.	.	.	.	.	#
星期	.	.	.	.	.	.	.	.	#	.
每	.	.	.	.	.	.	.	#	.	.
单独的	.	.	.	.	#	.	.	.	.	.
房间	.	.	.	.	.	#	.	.	.	.
一个	.	.	.	#	.	.	.	.	.	.
价格	.	.	#	.	.	.	#	.	.	.
多少	#	#	.	.	.	.	.	.	.	.
¿	.	.	.	.	.	.	.	.	.	.

Fig. 5. Graphical representation of the alignment between a Chinese sentence and an English sentence

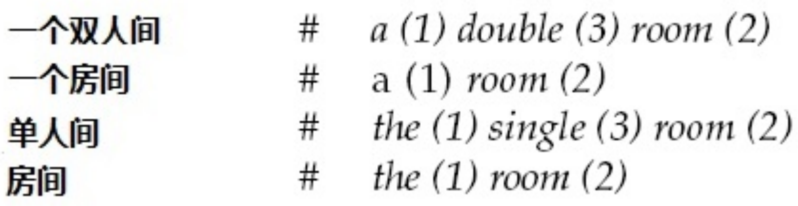
Another example which is a training sample composed by Chinese-English sentence is shown in Figure 6.



一个双人间	#	*a double room*
一个房间	#	*a room*
单人间	#	*the single room*
房间	#	*the room*

Fig. 6. An example of a Chinese-English training sample

Corresponding alignments for pairs of the training sample are shown in Figure 7.



一个双人间 # *a (1) double (3) room (2)*
一个房间 # a (1) *room (2)*
单人间 # *the (1) single (3) room (2)*
房间 # *the (1) room (2)*

Fig. 7. Corresponding alignments for the training sample

In the example, the English word "double" could be matched to the second Chinese word and the English word "room" to the third Chinese word.

Given source string *s*, target string *t*, and alignment *b*, the possible transformation z_i can be defined as follows:

$$z_i = \begin{cases} (s_i, t_j t_{j+1} \ldots t_{j+l}) & \text{if } \exists j : a(j) = i \text{ and } j' < j : a(j') > a(j) \\ & \text{and } for\ j'' : j \le j'' \le j+l, a(j'') \le a(j) \\ (s_i, \lambda) & \text{otherwise} \end{cases} \tag{20}$$

If the order of target string is not violated, each word from string *t* would be joined with the corresponding word from string *s* by the alignment *b*. Otherwise, the target word would be joined with the first word of source string which doesn't violate the order of target string.

When finishing the above steps, successive isolated source words can be joined to the first extended word which has target word assigned. Assumed that *z* is a transformed string obtained from the above step, and $(s_{k-1}, t_j t_{j+1} \ldots t_{j+m})(s_k, \lambda) \ldots (s_{k+l-1}, \lambda)(s_{k+l}, t_{j+m+1} \ldots t_{j+n})$ is a subsequence of z. Then the subsequence $(s_k, \lambda) \ldots (s_{k+l-1}, \lambda)(s_{k+l}, t_{j+m+1} \ldots t_{j+n})$ would be transformed into $(s_k, \ldots s_{k+l-1} s_{k+l}, t_{j+m+1} \ldots t_{j+n})$.

3.3 Stochastic regular grammar inference

In this paper, we used n-grams to infer regular grammar. The state transitions inferred from above examples are shown in Figures 8 and 9 [14].

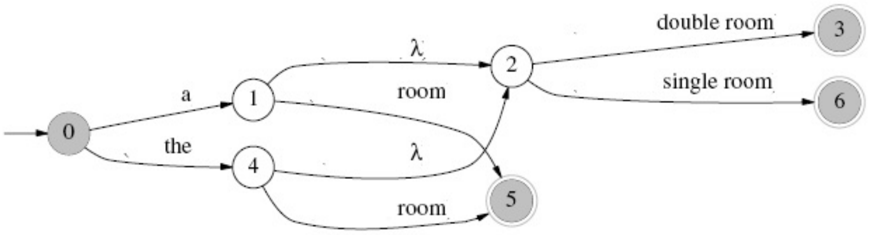


Fig. 8. State transitions inferred from example 1

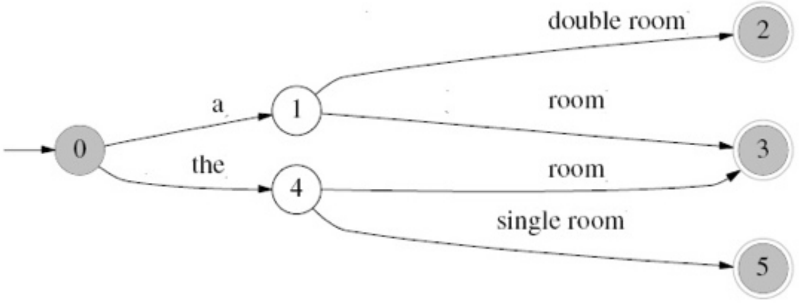


Fig. 9. State transitions inferred from example 2

The probabilities of the *n*-grams can be computed according to the counts of string in the training set. According to the sequence of words $z_{i-n+1}, \dots, z_{i-1} = (s_{i-n+1}, \bar{t}_{i-n+1}) \dots (s_{i-1}, \bar{t}_{i-1})$, the probability of word $z_j = (s_i, \bar{t}_i)$ can be evaluated as follows:

$$p_n(z_i \mid z_{i-n+1} \dots z_{i-1}) = \frac{c(z_{i-n+1}, \dots, z_{i-1}, z_i)}{c(z_{i-n+1}, \dots, z_{i-1})} \tag{21}$$

where $c(\cdot)$ represents the number of string occurences in the training set.

The *n*-gram model can be represented as a stochastic finite automaton. For the *n*-gram $(z_{i-n+1} \dots z_i)$, there exists a transition from state $(z_{i-n+1} \dots z_{i-1})$ to state $(z_{i-n+2} \dots z_i)$ according to Equation (21) [15].

In order to transform the grammar symbols to a finite-state automaton, $\Lambda(\cdot)$ based on two morphisms is used: $h_\Sigma((a, b_1 b_2 \dots b_k)) = a$, $h_\Delta((a, b_1 b_2 \dots b_k)) = b_1 b_2 \dots b_k$.

For z_i which represents a transition of the inferred grammar, its definition is $z_i = (a, b_1 b_2 \dots b_k) \in \Gamma$, the corresponding stochastic finite automaton can be expressed as $(q, a, b_1 b_2 \dots b_k, q')$ .If the states q and q' are $(z_{i-n+1} \dots z_{i-1})$ and $(z_{i-n+2} \dots z_i)$, the probability of the transition will be $p_n(z_i | z_{i-n+1} \dots z_{i-1})$ when $z_i = (a, b_1 b_2 \dots b_k)$.

4 Experiments and Analysis

The bilingual training corpus is composed of partial subsets of the LDC corpus, there are a total of about 4 million pairs of Chinese and English aligned sentences. Table 1 illustrates the summary of this corpus [16].

Firstly, we use our proposed method to translate Chinese to English, the example of our experimental sentence shown as Figures 10.

Results for the Train Set 1 and Train Set 2 corpora from Chinese-English translation are shown in Tables 2 and 3.

Table 1. Summary of the Chinese-English corpus

		Chinese	**English**
Train Set 1	Sentence pairs	3980000	
	Distinct pairs	1676269	
	Running words	3657000	3875500
	Vocabulary	984	531
Train Set 2	Sentence pairs	1500000	
	Distinct pairs	60860	
	Running words	97436	99821
	Vocabulary	972	531
Test	Sentences	2986	
	Running words	35911	35861

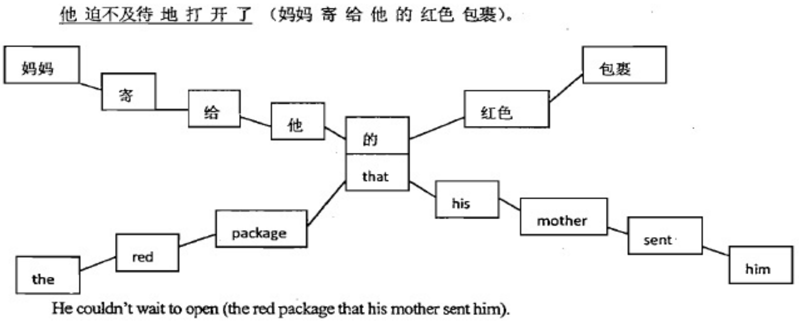

他 迫不及待 地 打 开 了 （妈妈 寄 给 他 的 红色 包裹）。
妈妈
寄
给
他
的
红色
包裹
that
his
mother
sent
him
package
red
the
He couldn't wait to open (the red package that his mother sent him).

Fig. 10. An example of our experimental sentence

Table 2. Results with the corpus TrainSet1

n-grams	**States**	**Transitions**	**WER (%)**	**SER (%)**	**BLEU**
2	4050	68135	8.7	50.1	0.85
3	33621	172506	4.9	27.5	0.93
4	110311	362376	4.1	23.9	0.94
5	146893	493741	3.9	20.6	0.95
6	200986	652147	3.6	19.1	0.96
7	265087	856976	3.3	18.3	0.96
8	330876	1039949	3.3	17.6	0.96
9	391632	1118468	3.2	17.2	0.96
10	437673	1348238	3.1	16.7	0.96
11	472981	1392127	3.1	16.2	0.96
12	493765	1486270	3.1	16.1	0.96

Table 3. Results with the corpus TrainSet2

n-grams	States	Transitions	WER (%)	SER (%)	BLEU
2	2421	22182	16.3	77.1	0.73
3	10352	42297	11.9	66.5	0.83
4	23558	73419	10.7	62.8	0.84
5	27689	82579	10.7	62.5	0.84
6	30527	92157	10.6	62.3	0.84
7	32511	96853	10.6	62.1	0.84

5 Conclusion

In this paper, a method based on stochastic finite automata has been proposed for inferring stochastic regular grammars. Our proposed method which was trained from source-target pairs, can achieve better effect in Chinese-English and English-Chinese translation. The effectiveness of the machine translation method had been evaluated with sufficient training data. Experimental results show that our method works better than traditional machine translation methods.

6 Acknowledgement

The study was supported by "Teaching Reform Project of Higher Education of Jiangxi Province, China (Grant No. JXJG-17-46-1)

7 References

[1] Dean, T., Angluin, D., Basye, K. (1995). Inferring finite automata with stochastic output functions and an application to map learning. Machine Learning, 18(1): 81-108. http://dx.doi. org/10.1007/BF00993822

[2] Casacuberta, F., Higuera, C. D. L. (2000). Computational complexity of problems on probabilistic grammars and transducers. Lecture Notes in Computer Science, 1891(1): 15-24. http://dx.doi. org/10.1007/978-3-540-45257-7_2

[3] Mohri, M. (1997). Finite-state transducers in language and speech processing. Computational Linguistics, 23(1): 269-311. http://dx.doi. org/10.1016/S0169-7439(97)00040-3

[4] Adriaans, P. W., Fernau, H., Zaanen, M. M. V. (2002). Grammatical inference: Algorithms and applications (ICGI). Lecture Notes in Computer Science, 41(3): 11-21. http://dx.doi. org/10.1007/b75249

[5] Ankinakatte, S., Edwards, D. (2015). Modelling discrete longitudinal data using acyclic probabilistic finite automata. Computational Statistics & Data Analysis, 88(2): 40-52. http://dx.doi. org/10.1016/j.csda.2015.02.009

[6] Kearns, M., Mansour, Y., Ron, D. (1994). On the learnability of discrete distributions. Journal of Japanese Society for Artificial Intelligence, 10(2): 273-282. http://dx.doi. org/10.1145/195058.195155

[7] Bangalore, S., Joshi, A. K. (1999). Supertagging: An approach to almost parsing. Computational Linguistics, 25(25): 237-265. http://dx.doi. org/10.1023/A:1008936413435

[8] Alshawi, H., Bangalore, S., Douglas, S. (2000). Head-transducer models for speech translation and their automatic acquisition from bilingual data. Machine Translation, 15(2): 105-124.

[9] Bangalore, S., Riccardi, G. (2002). Stochastic finite-state models for spoken language machine translation. Machine Translation, 17(3): 165-184. http://dx.doi.org/10.1023/b:coat.0000010804.12581.96

[10] Jones, D., Havrilla, R. (1998). Twisted pair grammar: Support for rapid development of machine translation for low density languages. Lecture Notes in Computer Science, 1889(7): 318-332. http://dx.doi. org/10.1007/3-540-49478-2_29

[11] Paliouras, G., Sakakibara, Y. (2002). Grammatical inference: algorithms and applications. Lecture Notes in Computer Science, 41(3): 191-202.

[12] Brown, P. F., Pietra, V. J. D., Pietra, S. A. D. (1993). The mathematics of statistical machine translation: Parameter estimation. Computational Linguistics, 19(2): 263-311.

[13] Li, X., Ouyang, J., Lu, Y. (2015). Group topic model: organizing topics into groups. Information Retrieval Journal, 18(1): 1-25. http://dx.doi. org/10.1007/s10791-014-9244-9

[14] Llorens, D., Vilar, J. M., Casacuberta, F. (2002). Finite state language models smoothed using n-Grams. International Journal of Pattern Recognition & Artificial Intelligence, 16(3): 275-289. http://dx.doi. org/10.1142/S0218001402001666

[15] Sun M., Yao J., Lv, Y. J. (2011). An improving feature extraction algorithm for maximum entropy based phrase reordering model. Journal of Chinese Information Processing, 25(2): 78-82. http://dx.doi. org/10.1007/s00466-010-0527-8

8 Author

Huiyan Li, female, Han nationality, was born in Fuzhou, Jiangxi Province in December, 1981, a lecturer in Zhangzhou Teachers College with a master's degree. Her main research direction: discourse analysis and English teaching research.

Article submitted 2019-01-16. Resubmitted 2019-02-23. Final acceptance 2019-02-24. Final version published as submitted by the authors.