International Journal of Signal Processing, Image Processing and Pattern Recognition
Vol.8, No.8 (2015), pp.43-50
http://dx.doi.org/10.14257/ijsip.2015.8.8.05

Uyghur Stemming Using Conditional Random Fields

Abdurahim Mahmoud[1], Akbar Pattar[1] and Askar Hamdulla[2]

[1]*Institute of Information Science and Engineering, Xinjiang University, Urumqi, China*
[2]*School of Software, Xinjiang University, Urumqi, China*
[1]*{abdurahim.mahmoud, pattarakbar}@gmail.com*
Corresponding author: [2]askarhamdulla@sina.com

Abstract

Stemming is a natural language processing task that to remove all derivational affixes from a word. This task proved to be harder for languages with complex morphology such as the Uyghur language. This paper presents a new stemming method for Uyghur words based on CRFs (Conditional Random Fields). In the proposed method all words in the training corpus are segmented into syllables and each syllable are tagged as a part of stem or as a part of affix. We experimentally evaluated this method with five test files each includes 100 sentences , results have shown that our method gets good performance, average stemming precision, recall and F-score in open test reached 98.42%, 98.34% and 98.38% respectively.

Keywords: *Uyghur; syllable segmenting; syllable tagging; stemming; CRFs*

1. Introduction

Stemming is a process of normalizing word variations by removing inflectional affixes. In agglutinative languages, stemming is as important as word segmenting in Chinese [1]. To design a stemming algorithm, it is possible to use a linguistic approach [2-4] that uses prior knowledge of the morphology on the specific language, or a statistical approach [5-7] that uses some methods based on statistical principles.

The linguistic approach is likely to be more effective, but it implies manual labor that has to be done linguists, the workload is bigger, needs to spend a lot of time, manpower and financial resources. The statistical approach takes little effort, but for some languages that have complex morphological system, such as Uyghur, its effect is not very ideal.

2. Features of Uyghur Word

2.1. Morphologic Structure

Uyghur is an agglutinative language in which words are formed by affixes attaching to a stem (or root). The structure of Uyghur word is "prefix + stem (or root) +suffix1+suffix2+…"[8]. Stem is the part of the word that is common to all its inflected variants. It is preceded by zero or a prefix and zero or many (longest is about ten or more) suffixes.

2.2. Syllabic Structure

A syllable is a unit of organization for a sequence of speech sounds. A Uyghur word consists at least one syllable and a syllable in Uyghur contains only one vowel (except some syllables imported from other languages).

ISSN: 2005-4254 IJSIP

Syllables in Uyghur language is regular, and the general format is "[C] V [CC]"(C stands for consonant, V stands for vowel), there are six basic syllable structures such as V, VC, CV, CVC, VCC, CVCC.

2.3. Phonetic Changes

There are mainly four types of phonetic changes in Uyghur: phonetic harmony, phonetic assimilation, phonetic dropping and phonetic insertion [9]. In this paper we mainly focused on the phonetic changes while adding an inflectional affix behind stem, because at this time the stem prototype will be changed, and our aim is to recover the original stem.

3. Data Preparation

3.1. Syllable Segmentation

There is no large-scale standard corpus in Uyghur, so this research used ten thousand sentences provided by our research group. First, we took 5900 sentences as our initial training corpus and carried out word segmentation and syllable segmentation to it, thus converted it into syllabic corpus. Uyghur segmentation is very easy, because words separated by spaces or punctuation. In this paper we used syllable tagging corpus for training and testing, so our first work is to cut all words into syllables by using the following syllable segmentation algorithm.

Syllable segmentation algorithm:

Input: A Uyghur word W.

Output: Syllable string S, in which each syllable separated by line break.

(1) Initialize S is empty, set W_L for length of W, turn to (2).

(2) If W_L <4 let S equal to W and finish this algorithm, else turn to (3).

(3) Initialize current letter L is empty, i is 0, turn to (4).

(4) Fetch a letter from W, and assign it to L, if L is hamza turn to (5), if L is vowel turn to (6), if L is consonant turn to (7).

(5) If i >0 add a line break to S, turn to (6).

(6) Add L to S, and turn to (11).

(7) If i = = 0 or i = = 1 or i = = W_L-1 add L to S and turn to (11), if i > 1 and i < W_L-1 turn to (8).

(8) If W[i+1] is consonant or hamza add L to S and turn to (11), if W[i+1] is vowel turn to (9) .

(9) add line break to S and turn to (10).

(10) Add L to S and turn to (11).

(11) Let i increase 1 and turn to (12).

(12) If i < W_L turn to (4), else finish this algorithm.

3.2. Syllable Tagging

According to the relationship between words and syllables, all syllables were added corresponding tags manually. In tagging process, all kind of phonetic changes were also considered. Tags and meanings which used in this research is shown in table 1. The data preparation process and result is shown in Figure 1. Word and syllable segmentation results show that the 5900 sentences which used for training contains 70,300 words, 192,400 syllables.

Table 1. Tags and Meanings

Tag	Meaning	Tag	Meaning	Tag	Meaning
pr	Prefix	+i	"ى" dropped	sap	"سېلىپ" written "ساپ"
st	A part of stem	+u	"ۇ" dropped	qap	"قېلىپ" written "قاپ"
su	A part of suffix	+v	"ۇ" dropped	Ep	"ئېلىپ" written"ئېپ"
ie	"ە"weakened into"ى"	1	One letter belongs to stem	ep	"ئېلىپ" written "ئەپ"
ia	"ا" weakened into "ى"	2	Two letters belong to stem	qEp	"قېلىپ" written "قېپ"
Ee	"ە" weakened into"ې"	3	Three letters belong to stem	chEp	"چېلىپ" written "چېپ"
Ea	"ا" weakened into "ې"	men	"مەن" written "مې"	kEp	"كېلىپ" written "كېپ"
ea	"ا"written"ە"	sen	"سەن" written"سې"	chap	"چېلىپ" written "چاپ"
uo	"و" written "ۇ"	bop	"بولۇپ" written "بوپ"	qip	"قىلىپ" written "قىپ"
vu	"ۇ" written "ۇ"	kep	"كېلىپ" written "كەپ"	O	is not syllable
ve	"ە" written "ۇ"	ap	"ئېلىپ" written "ئاپ"		

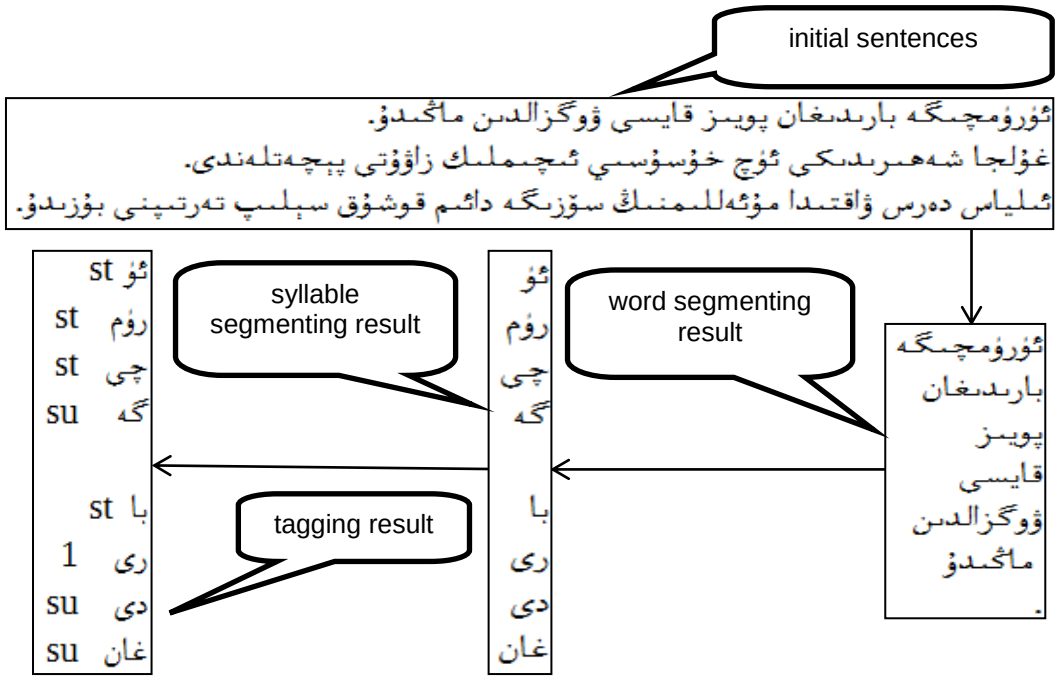

Figure 1. Data Preparing Results

4. CRFs Stemming

CRFs model is an undirected graph model based on statistics, currently in the natural language processing fields is mainly used to segment and mark the serialized data [10]. It has no independence assumption, features can be chosen arbitrarily and all the features can take global normalization to obtain global optimal solution.

4.1. Problem Definition

When the words segmented into syllables, stemming can be defined as a sequence tagging problem, that whether the observed syllable belongs to a predefined tagging set.

Suppose that a Uyghur word composed by the syllable sequence $s = (s_1, s_2, \ldots \ldots s_n)$, the given tag set is: $y = (y_1, y_2, \ldots \ldots y_n)$. Wherein, n

represents the length of a given syllable sequence, CRFs model is defined as the following equation (1):

$$p(y|s) = \frac{1}{Z(s)} \exp(\sum_i \sum_k \lambda_k f_k(y_i, y_{i-1}, s)) \quad (1)$$

Wherein, Z is a normalization factor, depends only on the observing sequence, the value of Z can be obtained by the forward backward algorithm, as shown in formula (2):

$$Z(s) = \sum_y \exp[\sum_k \lambda_k f_k(y_i, y_{i-1}, s)] \quad (2)$$

λ_k is the weight coefficient of the K-th feature function; $f_k(y_i, y_{i-1}, s)$ is feature function, its definition given in formula (3), given below.

$$f_{u,v}(y_i, y_{i-1}, s) = \begin{cases} 1 & \text{if } y_i = u \text{ and } y_{i-1} = v \\ 0 & \text{else} \end{cases} \quad (3)$$

4.2. CRFs Training

In a variety of machine learning models, feature selection has an important influence on the training result and in CRFs model, and the feature selection is controlled by the feature template.

This paper selected the context syllables as basic features. 245 different template file were created, and each template file composed by different combinations of features. Finally, we generated 245 training model file by using the CRF++ Toolkit (Taku Kudo).

4.3. Testing and Post-processing

We prepared five test files, two from the initial ten thousand sentences, two from Internet and one from training data, each test file includes 100 sentences, as shown in Table 2.

Table 2. Test Corpus

Test corpus	Number of syllables	Data source
test1.txt	5400	From the Internet(open test)
test 2.txt	5667	From the Internet(open test)
test 3.txt	6102	From initial 10000 sentences except training data(close test)
test 4.txt	5870	From initial 10000 sentences except training data(close test)
test 5.txt	2749	From the training data(close test)

Each test corpus was tested with 245 models, generated 245 test results respectively. Then, we carried out post-processing for the testing results and produced the final stemming results shown in Figure 2.

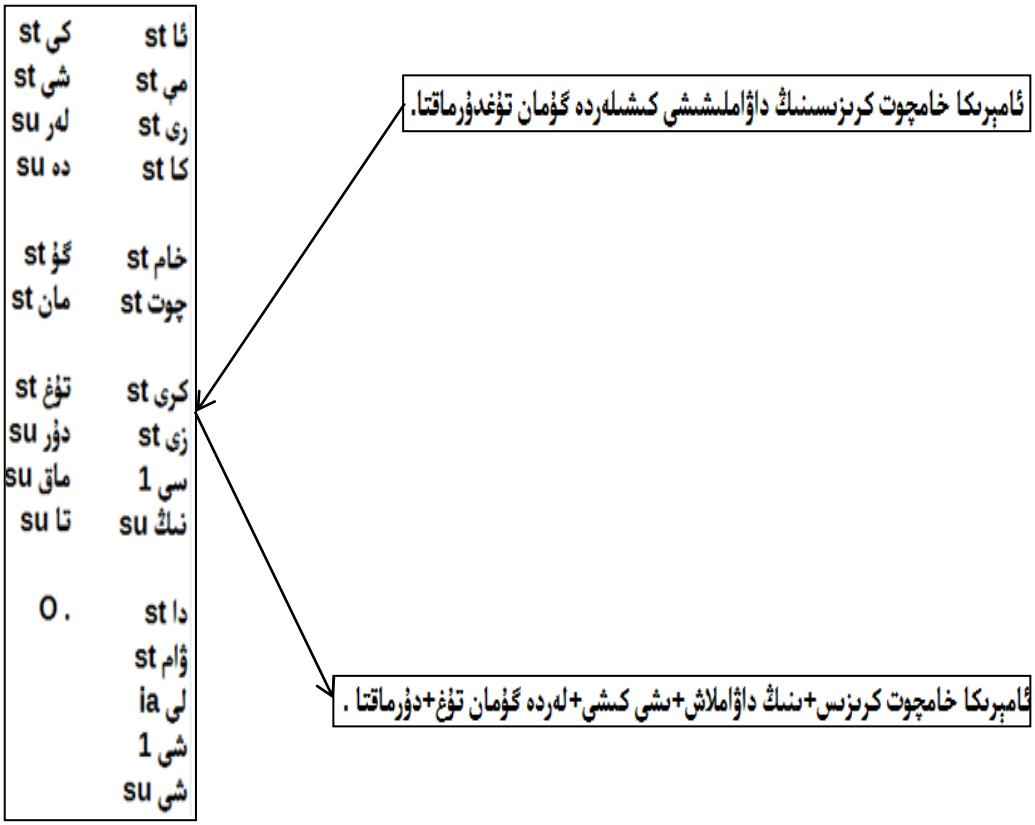

Figure 2. Testing and Post-processing Results

5. Evaluating

We calculated the precision, recall and F-Score of each testing results. The top five evaluation results of all testing results were shown in table 3. From the evaluation result we can see that the best F-Score of these five testing results are 98.03%, 98.73%, 99.25%, 98.95% and 99.89% respectively. the best result of close test is 99.25% and the best result of open test is 98.73%.

The model which generates the best result on different test file is not a same model, we can further improve the evaluation result through the analysis of the reasons that influence evaluation results.

Figure 3 shows the changing trends of all evaluation results, from figure.3 we can see that model 1, from model 5 to model 8, from model 44 to model 68, model 244 and model 245 show relatively poor evaluation results, the result of model 5 is poorest. From model 21 to model 43, from model 73 to model 243 show relatively better evaluation results.

Table 3. Top Five Evaluation Results

	Model No	Precision	Recall	F-Score
test1.txt	38	98.11	97.95	98.03
	34	98.06	97.95	98.00
	35	98.05	97.89	97.97
	39	98.04	97.87	97.95
	27	97.96	97.85	97.91
test2.txt	151	98.73	98.73	98.73
	149	98.73	98.73	98.73
	147	98.73	98.73	98.73

	117	98.73	98.73	98.73
	98	98.71	98.71	98.71
test3.txt	229	99.25	99.25	99.25
	81	99.23	99.23	99.23
	83	99.21	99.21	99.21
	172	99.21	99.21	99.21
	214	99.20	99.20	99.20
test4.txt	38	98.96	98.94	98.95
	148	98.94	98.93	98.94
	27	98.93	98.93	98.93
	90	98.93	98.91	98.92
	192	98.93	98.91	98.92
test5.txt	99	99.89	99.89	99.89
	91	99.89	99.89	99.89
	90	99.89	99.89	99.89
	89	99.89	99.89	99.89
	88	99.89	99.89	99.89

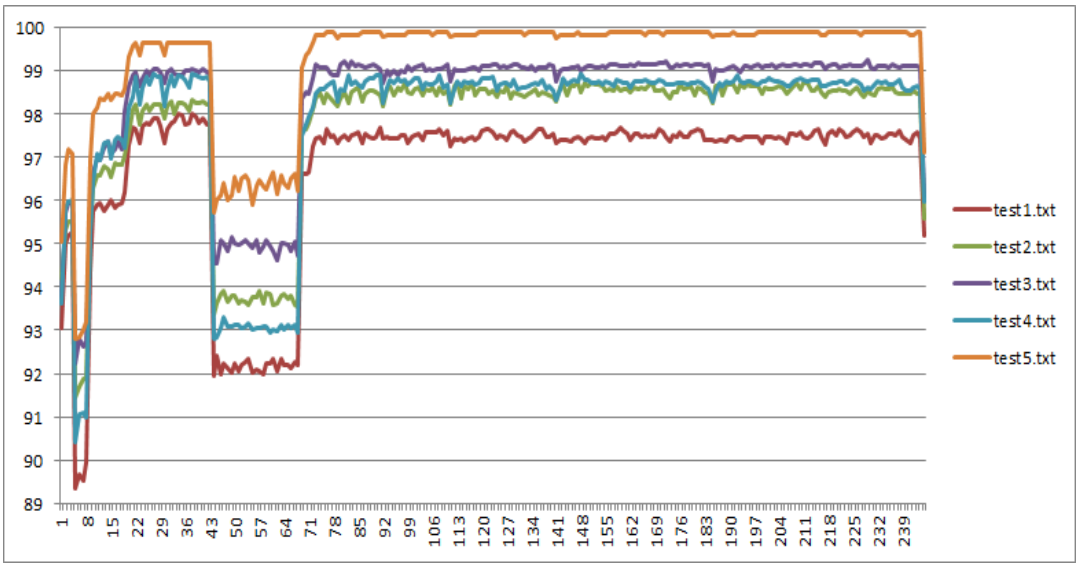

Figure 3. Changing Trends of Evaluation Results

6. Conclusion

We have presented a new and extensible stemming method for Uyghur based on CRFs and has achieved a good stemming result. The main idea of this method is syllable segmenting and tagging. This method can solve common phonetic changing problem in Uyghur such as assimilation, dropping and insertion. Languages like Turkish, Kazakh, Uzbek and Kyrgyz which has the syllablic structure can use this method to extract stem. The system has been tested using Uyghur. Results have been very promising.

Like any other languages, Uyghur stemming is a challenging research topic, there are many issues that need further study such as how to improve the accuracy of syllable segmenting and tagging, how to avoid tagging ambiguity, how to select the best feature template etc. These are our next step is to carry out the work.

Acknowledgements

This work has been supported by Innovation Program for Excellent Ph.D. Candidates of Xinjiang University (XJUBSCX-2012010) and the National Natural Science Foundation of China under grant of 61163032.

References

[1] T. Abuduwaili, A. Wumaier, T. Yibulayin and J. Zhang, "Uyghur Verb Stemming Method Based on a Tagged Dictionary and Rules", Journal of Xinjiang University(Natural Science Edition), vol. 1, no. 30, **(2013).**
[2] M. F. Porter, "An algorithm for suffix stripping", Program, vol. 3, no. 14, **(1980).**
[3] G. Eryiğit and E. Adahı, "An Affix Stripping Morphological Analyzer for Turkish", Proceedings of the International Conference Artificial Intelengce and Applications, **(2004)** February 16-18; Innsbruck, Austria.
[4] K. Taghva, R. Elkhoury, and J. Cooms, "Arabic Stemming Without A Root Dictionary", Proceedings of the International Conference on Information Technology: Coding and Computing, **(2005)** April 4-6; Las Vegas, NV, USA.
[5] M. Melucci and N. Orio, "A novel Method for Stemmer Generation Based on Hidden Markov Models", Proceedings of the twelfth international conference on Information and knowledge management, **(2003)** New Orleans, LA, USA.
[6] J. Mayfield and P. McNamee, "Single N-gram Stemming", Proceedings of the 26th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval. (2003) July 28-August; Toronto, Canada.
[7] M. A. Hafer and S. F. Weiss, "Word Segmentation by Letter Successor Varieties", Information Storage and Retrieval, vol. 10, (**1974**), pp. 371-385
[8] M. Ablimit, G. Neubig, M. Mimura, S. Mori, T. Kawahara and A. Hamdulla, "Uyghur Morpheme-based Language Models and ASR", Proceedings of the International Conference on Software Process, **(2010)** July 8-9; Paderborn, Germany.
[9] H. Tomur, "Modern Uyghur grammar (Morphology)", National publishing house of China, Beijing, **(1987).**
[10] J. Lafferty, A. McCallum and F. Pereira, "Conditional Random Fields: Probabilistic Models For Segmenting and Labeling Sequence Data", Proceedings of the 18th International conference on Machine Learning, **(2001**); Williamstown, MA, USA.

Authors

Abdurahim Mahmoud, he received his BSc degree in Electronics and Information System from Xinjiang University, Xinjiang, China in 1996, and MSc degree in Mechanical Design Theory from Xinjiang University, Xinjiang, China in 2007. He joined Xinjiang University as an assistant teacher in 1996. Currently, he is a doctoral student of computer science, his research direction: natural language processing.

Akbar Pattar, he received his B.E. degree in radio electronics from Xinjiang University, China, in 1983. He has been working as a teacher in School of Information Science and Engineering, Xinjiang University since 1983. His research interests include natural language processing and pattern recognition.

Askar Hamdulla, he received B.E. in 1996, M.E. in 1999, and Ph.D. in 2003, all in Information Science and Engineering, from University of Electronic Science and Technology of China. In 2010, he was a visiting scholar at Center for Signal and Image Processing, Georgia Institute of Technology, GA, USA. Currently, he is a professor in the School of Software Engineering, Xinjiang University. He has published more than 120 technical papers on speech synthesis, natural language processing and image processing. He is a senior member of CCF and an affiliate member of IEEE.