International Journal of Database Theory and Application
Vol.9, No.11 (2016), pp.207-216
http://dx.doi.org/10.14257/ijdta.2016.9.11.19

Uyghur Stemming and Lemmatization Approach based on Multi-Morphological Features

Abdurahim Mahmoud[1], Sediyegvl Enwer[1], Abdusalam Dawut[2], Palidan Tuerxun[2], and Askar Hamdulla[1*]

1Institute of Information Science and Engineering, Xinjiang University, China
2School of Software, Xinjiang University, Urumqi, China, 830046
[*] *askarhamdulla@sina.com*

Abstract

This paper describes a stemming and lemmatization approach for Uyghur using Conditional Random Fields (CRFs). In the proposed approach, we used syllable-level training and test corpus with the combination of some automatically tagged positional and morphological feature tags. The training and test corpus has been manually tagged with a stemming tag set which includes eight kinds of tags which fully reflect the morphological feature of Uyghur word. It has been observed that some morphological features are very helpful for improving the evaluating results. The syllable-level Precision, Recall and F-score of the best evaluation result respectively are 98.79%, 98.71% and 98.75% respectively, and the word-level accuracy we achieved is 95.9%.The experimental results show that the efficiency of this approach is very ideal.

Keywords*: Stemming, Lemmatization, Morphological Feature, Syllable-Level Tagging*

1. Introduction

As a basic Natural Language Processing (NLP) task, stemming plays very important role for several NLP areas, including Information Retrieval, Machine Translation, Search Engine and Language Understanding. To extract stem, it is possible to use rule based approaches such as affix stripping approach [1-3] that uses prior knowledge on a certain language morphology, or use statistical approaches that employ statistical information from a large corpus of a given language to learn morphology of words, such as N-Gram Models [4], Successor Varieties [5], Hidden Markov Models(HMMs)[6], Maximum Entropy Models (MEMs) [7], or Conditional Random Fields(CRFs)[8].

Uyghur is an agglutinative language with rich and complex morphology [9], a Uyghur word consists of some smaller morphological units without any splitter between them [10]. Because of the complexity of Uyghur's morphology, if purely use a rule based approach, cannot get ideal stemming results.

In the statistical methods, CRFs [11] is the better choice than HMMs and MEMs. The primary advantage of CRFs over HMMs is their conditional nature, resulting in the relaxation of the independence assumptions required by HMMs in order to ensure tractable inference. Additionally, CRFs avoid the label bias problem [11], a weakness exhibited by Maximum Entropy Markov Models (MEMMs) [12] and other Conditional Markov Models based on directed graphical models.

2. Syllable-Level Tagging Approach

When a suffix is added behind a stem to form a Uyghur word, the addition of suffix always causes some changes on the dictionary form of stem, therefore stemming and

[*] Corresponding Author

ISSN: 2005-4270 IJDTA
Copyright ⓒ 2016 SERSC

lemmatization must be carried out simultaneously. To implement stemming and lemmatization by using CRFs approach, we need to prepare a certain size, tagged training corpus and test corpus, the size and quality of the training and test corpus, and the selection of tagging level directly affects the stemming and lemmatizing results. Tagging approaches can be divided into three levels, that is, word-level, syllable-level and letter-level tagging.

Word-Level Tagging: In this approach we add tags after each word. The morphological structure of Uyghur word is more complex, it is very difficult to determine the boundaries between stem and suffix and to describe the change of stem form by using word-level tagged corpus, it will directly affect the stemming result.

Letter-Level Tagging: In this approach, we divide each word to letters firstly, then add tags after each letter. Letter is the smallest unit that forms word, so letter-level tagging is very time-consuming, and the size of letter-level tagged corpus is very large, at least ten times larger than word-level tagged corpus so that the corpus training is also very time consuming.

Syllable-Level Tagging: In this approach, we convert each word to syllable sequences first, and then add the tags after each syllable. The Uyghur syllable is very regular, so the syllable segmentation of Uyghur word is very easy. Syllable-level tagging is a compromise approach between word-level tagging and letter-level tagging. It is a good solution of these problems which caused by word-level tagging and letter-level tagging. At the same time, we can use some additional tags that can describe more morphological features. The flow chart of our proposed stemming method is shown in Figure 1.

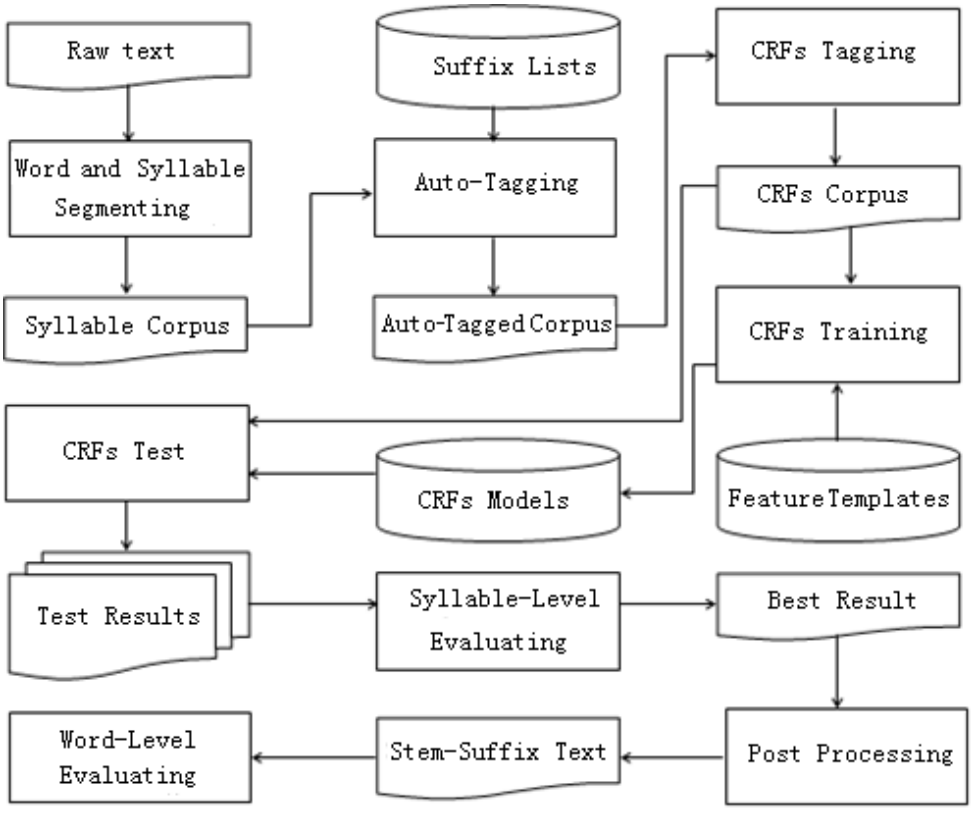


Figure 1. Uyghur Stemming Process

In the proposed method, we carried out word segmentation and syllable segmentation to prepare syllable-level training and test corpus firstly, and automatically added five different tags behind each syllable in both training and test corpus to describe the position information of each syllable in current word, and to describe some useful morphological

 Copyright ⓒ 2016 SERSC

features such as derived suffix, number suffix, ownership-dependent suffix and case suffix, then the last correct tags(CRFs tags) are manually added. To effectively describe the boundary between stem and suffix and the change of stem form, we proposed some special tags, the detailed description of these tags is given in next section.

3. Data Preparation

The process of preparing training and test corpus is shown in Figure 1. Each line of the training corpus and test corpus consists of seven columns that are separated by tabs, the first column is context syllable, the last column is the manually added correct CRFs tag, we can call it stemming tag, and other columns are the automatically added feature tags, shown in Table 1. This paper used 6400 sentences provided by our research team, in which 5900 sentences were used for training, that include 70300 words and 192400 syllables. The remaining 500 sentences were used for testing, that include 7596 words and 25788 syllables.

Table 1. Structure of Training and Test Corpus

Column	Column Name	Symbol	Values
1	Syllable	*syl*	a syllable
2	Syllable Position Tag	*sp*	s, 1, 2, ….
3	Derived Suffix Tag	*ds*	0-15
4	Number Suffix Tag	*ns*	0 or 1
5	Ownership-Dependent Suffix Tag	*os*	0-7
6	Case Suffix Tag	*cs*	0-9
7	Stemming Tag	*st*	see Table 5

3.1. Syllable Corpus

According to spaces and punctuations between words, all Uyghur words and non-word contents from raw document were extracted, and each word was segmented into a syllable sequence by using syllable segmentation algorithm [8] and saved into syllable corpus.

3.2. Auto-Tagged Corpus

Five different tags that describe some extra features of syllable were automatically added behind each syllable in syllable corpus, these tags are as follows:

Syllable Position Tag (*sp*) : This tag indicates the position of the current syllable in the current syllable sequence. It ranges from one to the number of syllables in a word. If a word has only one syllable, in order to indicate it is a single syllable, we use's' as tag value.

Derived Suffix Tag (*ds*): To automatically add this tag, we prepared four derived suffix lists, respectively are noun suffix list, adjective suffix list, verb suffix list and adverb suffix list. A syllable (or a syllable sequence) may appear in two or more lists at the same time, if a syllable (or syllable sequence) appears in a suffix list, we set the corresponding tag weight, otherwise the tag weight is 0, the last derived suffix tag value is equal to the sum of these tag weights, shown in Table 2.

Copyright ⓒ 2016 SERSC

Table 2. Derived Suffixes (*ds*) and Corresponding Tag Weights

Suffix Type	Weight	Symbol	Tag Value (*d_s*)
Noun Suffix	0/1	W_n	$d\_s = W_n + W_{adj} + W_v + W_{adv}$
Adjective	0/2	W_{adj}	
Verb Suffix	0/4	W_v	
Adverb Suffix	0/8	W_{adv}	

For example, syllable "چى " is appeared in noun suffix list, adjective suffix list and verb suffix list, but not appeared in adverb suffix list, so $W_n = 1$, W_{adj}=2, W_{adj}=4, and W_{adv}=0, the derived suffix tag value equal to 7 ($W_n + W_{adj} + W_v + W_{adv} = 1 + 2 + 4 + 0 = 7$) .

Number Suffix Tag (*ns*): The number category of Uyghur nouns indicates the relationship between the object expressed by the noun and its number [13]. Uyghur nouns appear in singular form or in plural form in sentences. The singular form of the noun does not have suffix, but the plural form adds 'لار' and 'لەر'. In syllable sequence may appear some other plural forms such as 'لىرى','لىرىم'. So we stored all possible plural suffix forms in a list. If a syllable (or syllable sequence) appears in the list, the corresponding tag is 1, otherwise 0.

Ownership-Dependent Suffix Tag (*os*): The ownership-dependent category of the noun indicates that the object expressed by the noun is dependent on or belongs to a certain object [13]. In Uyghur, this category is expressed by seven type's ownership-dependent suffixes. So, we prepared seven suffix lists, and according to these lists added corresponding tag value behind each syllable. The ownership-dependent suffix types and their corresponding tag values are shown in Table 3.

Table 3. Ownership-Dependent Suffixes (*os*) and Tag Values

Suffix Type	Suffix	Tag Value
Not a Suffix		0
1st Person Singular	م // ىم // ۇم // ۇم	1
1st Person Plural	مىز // ىمىز	2
2nd Person Singular	ڭ // ىڭ // ۇڭ //ۇڭ	3
2nd Person Plural	ڭلار	4
2nd Person Refined	ڭىز // ىڭىز	5
2nd Person Respectful	لىرى	6
3rd Person	ى // سى	7

Case Suffix Tag (*cs*): The case category of the noun indicates the syntactical relationship between the noun and other words [13]. In Uyghur, this category is expressed by case forms which are made by adding case suffixes. The case of Uyghur nouns is divided into ten types, respectively are nominative case, genitive (possessive) case, accusative case, dative case, ablative case, locative case, limitative case, locative-qualitative case, similitude case and equivalence case, in which nominative case has not suffix. Table 4 shows all case suffix types except nominative case) and their corresponding tag values.

 Copyright ⓒ 2016 SERSC

Table 4. Case Suffixes (*cs*) and Tag Values

Suffix Type	Suffix	Tag Value
not a suffix		0
genitive(possessive) case suffix	نىڭ	1
accusative case suffix	نى	2
dative case suffix	غا // قا // گە // كە	3
ablative case suffix	دىن // تىن	4
locative case suffix	دا // تا // دە // تە	5
limitative case suffix	غىچە // قىچە // گىچە // كىچە	6
locative-qualitative case suffix	دىكى // تىكى	7
similitude case suffix	دەك // تەك	8
equivalence case suffix	چىلىك // چە	9

The suffix tag values of the first syllable can be changed to 0, because the first syllable of the word must be a part of a stem, Figure 2 shows the result of auto-tagging.

syl	sp	ds	ns	os	cs
تاغ	s	0	0	0	0
دەر	1	0	0	0	0
يا	2	0	0	0	0
لار	3	0	1	0	0
نى	4	7	0	7	2
ئۆز	1	0	0	0	0
گەر	2	5	0	0	0
تى	3	4	0	7	0
شى	4	5	0	2	0
مز	5	0	0	2	0

Figure 2. The Format of Auto-Tagged Corpus

3.3. CRFs Corpus

The CRFs corpus is the last corpus that used for training and testing. In order to generate it, we added the last stemming tag set (s_t) at the end of each non-empty line of the auto-tagged corpus manually. These tags should be added manually, because they are the correct tags in the last column of the CRF corpus. To add it, one must have sufficient patience and certain knowledge of Uyghur morphology. We carefully selected 31 different tags, these tags reflected different morphological and phonetically features of the Uygur word structure, and these tags also fully reflected the boundary of stem and suffix, phonetic changes of Uyghur words that often appear in stem such as weakening, inserting and dropping. We also considered the tagging of abbreviated verbs that often appear in oral language. The explanations of these tags are shown in Table 5.

Copyright ⓒ 2016 SERSC

Table 5. Tag Values of Stemming Tag (*St*)

Tag Type	Tag Value	Meaning
Stem Tag	st	Current syllable is a stem
Suffix Tag	su	Current syllable is a suffix
Boundary Tag	1 / 2 / 3	The first one, two or three letter(s) belong(s) to stem, other letter(s) belong(s) to suffix
Weakened Tag	ie/ia/ea/Ea/Ee /uo/vu/ve	Current syllable belongs to stem, but occurred vowel weakening.
Dropped Tag	i+/u+/v+	Current syllable belongs to stem, but occurred vowel dropping.
Personal Pronoun Tag	men/sen	Current syllable is a personal pronoun, but affected by suffix and changed its form.
Abbreviated Tag	bop/kep/up/sa p/qap…	Two syllables are abbreviated to one.
Non-Syllable Tag	O	This is not a Uyghur syllable.

4. Feature Templates and Training

We use CRFs model to carry out Uyghur stemming. In this model, the selection of language features is very important. It will directly affect the quality of the CRFs training model, finally affects the testing result. In CRFs model, the selection of features is controlled by feature templates.

Generally, in syllable-level stemming, the context syllables can be selected as basic features. In order to improve the stemming efficiency, this paper used five kinds of positional and morphological features which mentioned above.

Context is an observation window centered on the current feature; its length can be defined as the distance from the top feature to bottom feature in the context. The bigger the observation window length, the more context information can be used, the more helpful to stemming. But, if the window length is too large, it will produce over-stemming, if it is too small, then the added features are not sufficient, the information contained in it will be limited, thus a lot of important context information will be lost.

In order to determine the stemming effect of various features in different window length, we prepared a group of feature templates, which contains different features with different window length, shown in Table 6. We carried out CRFs model training with all of these feature templates and generated different CRFs training models.

Table 6. Feature Templates

Template Type	Selected Features	Max Length of Observation Window
Basic Feature	*syl*	*syl:*10 other features:8
Two Features	*syl*+another one feature	
Three Features	*syl*+another two features	
Four Features	*syl*+another three features	
Five Features	*syl*+another four features	
All Features syl,sp,ds,ns,os,cs		

 Copyright ⓒ 2016 SERSC

5. Testing and Evaluating

5.1. Testing

The test corpus used in this paper is taken from the CRFs corpus. It consists of 500 sentences, which contain 7596 words and 25788 syllables. The proportion of training data and test data is 12:1.

We carried out CRFs testing experiment on the test corpus by using different training models mentioned above and got different testing results.

5.2. Syllable-Level Evaluating

To find the best observation window length of feature templates which can generate best training model, we carried out evaluation experiments on each test result, and compared the experimental results (F scores) each other, shown in Table 7. We can make a conclusion that, in these results, the window length which produces the largest F-score is the best window length.

Table 7. F-Scores of All Experiments

Window Length of First Feature		3	4	**5**	6	7	8	9	10
Window Length of Last Feature syl		1	2	**3**	4	5	6	7	8
One Feature	*syl*	97.95	98.09	98.07	**98.12**	98.05	97.91	96.15	96.04
Two Features	*syl+sp*	98.04	98.05	98.07	98.09	**98.14**	98.00	97.88	97.90
	syl+ds	98.02	98.07	98.02	**98.11**	98.10	98.09	97.95	97.32
	syl+ns	98.18	**98.23**	98.16	98.21	98.19	98.18	98.13	98.15
	syl+os	**97.36**	97.28	97.29	97.21	97.24	96.26	96.15	96.03
	syl+cs	98.00	98.19	98.37	**98.52**	98.23	97.91	97.94	97.24
Three Features	*syl+os+sp*	97.98	98.08	98.05	**98.20**	98.07	98.00	97.95	97.60
	syl+os+ds	98.05	98.35	**98.75**	98.63	98.15	97.75	97.83	97.46
	syl+os+ns	98.18	**98.23**	98.16	98.21	98.19	98.18	98.13	98.15
	syl+os+cs	98.02	**98.24**	98.11	98.09	98.04	98.00	97.76	97.60
Four Features	*syl+os+ns+sp*	98.02	98.20	98.27	**98.34**	98.10	98.06	98.00	97.62
	syl+os+ns+ds	98.12	**98.27**	98.21	98.21	98.17	98.00	97.97	97.75
	syl+os+ns+cs	98.04	98.15	**98.26**	98.17	98.07	98.00	97.85	97.61
Five Features	*syl+os+ns+ds+sp*	98.06	98.17	**98.24**	98.14	98.03	97.90	97.81	97.67
	syl+os+ns+ds+cs	96.63	97.071	**97.79**	97.66	97.15	97.05	96.77	96.41
All Features		96.63	97.95	98.09	98.07	**98.12**	98.05	97.91	96.15

From Table 7 we can see that the result of any feature group is better than the result of single feature, that is, any additional feature is helpful to improve the stemming efficiency. The feature group that generates the best evaluation result is $syl_{-3\sim2}$, $os_{-1\sim1}$, $ns_{-2\sim1}$, and the best F-score of syllable-level is 98.75% .

6. Post-Processing and Word-Level Evaluating

The purpose of the post processing is to convert each syllable sequence in the test result to a string in “stem+suffix” form according to the result tags corpus. The post-processing algorithm is as follows:

Post-Processing Algorithm:

Input: Test result file .

Output: ”stem+suffix” sequence file .

Copyright ⓒ 2016 SERSC

Variables: st : "syllable+tags" string; s : Syllable string; c-tag : The tag of current syllable; p_tag : The tag of previous syllable; ss : "stem+suffix" string; ss_list : "stem+suffix" list .

Step 1: Initialize all variables to empty .

Step 2: Read a line ("syllable+tags" sequence) from test result file,if this is an empty line turn to Step 11, else save it to st .

Step 3: Split st to syllable and tags, save first item to s and save the last item to c_tag .

Step 4: If c_tag is a stem tag or a non-syllable tag, append s to ss .

Step 5: If c_tag is stem-suffix tag, first append one(two or three) letter(s) in s to ss, then append "+" to ss, and finally add the remaining letters in s to ss .

Step 6: If c_tag is weakened tag or dropped tag, then change s to original form, and append it to ss .

Step 7: If c_tag is personal pronoun tag, then change s to "men" or "sen", and append it to ss .

Step 8: If c_tag is abbreviated tag,then change s to "stem+suffix" form, and append it to ss .

Step 9: If c_tag is suffix tag, p_tag is suffix tag or stem-suffix tag or abbrievated tag, then append s to ss, otherwise append "+" and s to ss .

Step 10: Assign c_tag to p_tag and turn to Step 12 .

Step 11: If ss is not empty, then append ss to ss_list and assign empty string to ss .

Step 12: If this is not the last line,then turn to Step 2 .

Step 13: If ss is not empty, then append ss to ss_list .

Step 14: Write ss_list to "stem+suffix" sequence file .

We selected the best syllable-level testing result, and converted it in "stem+suffix" form by using the post-processing algorithm, finally a word-level evaluation carried out on this result, the word-level accuracy we achieved was 95.9%.

7. Conclusion and Future Work

This paper proposed a syllable-level multi-tagged CRFs Uyghur stemming approach. This approach showed a good stemming efficiency. The evaluation results show that the CRFs models which use multi-morphological features are more efficient than the CRFs models which use single feature. The contribution of this approach is that it applied the syllable-level tag set and integrated the morphological features of Uyghur word into CRFs model.

Like other languages, Uyghur stemming is a challenging research topic, there are a lot of issues that need further study. For example, how to improve the tagging quality of automatic-tagging, how to integrate more morphological features into the corpus, can we add part-of-speech tag to improve stemming efficiency, *etc.* These are the problems we need to further solve in the future research works.

Acknowledgements

This work has been supported by National Natural Science Foundation of China under grant number of 61562081, "thousand talents program" of China and High Technology Research and Development Project of Xinjiang under grant number of 201312103.

References

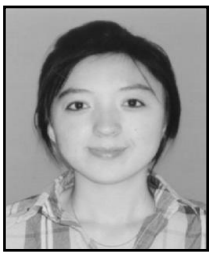
[1] M. F. Porter, "An algorithm for suffix stripping", Program, vol. 14, no. 3, (**1980**).
[2] G. Eryiğit and E. Adalı, "An Affix Stripping Morphological Analyzer for Turkish", Proceedings of the IASTED International Conference on Artificial Intelligence and Applications, Innsbruck, Austria, (**2004**), pp. 299-304.
[3] K. Taghva, R. Elkhoury and J. Cooms, "Arabic Stemming Without A Root Dictionary", Proceedings of the International Conference on Information Technology: Coding and Computing, Las Vegas, NV, USA, vol. 1, **(2005)**, pp. 152-157

 Copyright ⓒ 2016 SERSC

[4] 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, Toronto, Canada, **(2003)**, pp. 415-416.
[5] M. A. Hafer and S. F. Weiss, "Word Segmentation by Letter Successor Varieties", Information Storage and Retrieval, vol. 10, no. 11-12, **(1974)**, pp. 371-385.
[6] 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, New Orleans, LA, USA, **(2003)**.
[7] Z. Kadeer, A. Wumaier, T. Yibulayin, P. Tursun and W. Xiaochuan, "Uyghur noun stemming system based on hybrid method", Computer Engineering and Applications, **(2013)**, vol. 49, no. 1, pp. 171-175.
[8] A. Mahmoud, A. Pattar and A. Hamdulla, "Uyghur Stemming Using Conditional Random Fields", International Journal of Signal Processing, Image Processing and Pattern Recognition, vol. 8, no. 8, **(2015)**, pp. 43-50.
[9] B. Aisha and M. Sun, "A Statistical Method for Uyghur Tokenization", Proceedings of the 5th International Conference on Natural Language Processing and Knowledge Engineering, Dalian, China, **(2009)**, pp. 1-5.
[10] 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, Paderborn, Germany, **(2010)**, pp. 581-584.
[11] 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, Williamstown, MA, USA, **(2001)**.
[12] A. McCallum, D. Freitag, and F. Pereira, "Maximum Entropy Markov Models for Information Extraction and Segmentation", Proceedings of the 17th International Conference on Machine Learning, Stanford, CA, USA, **(2000)**.
[13] H. Tomur, "Modern Uyghur grammar (Morphology)", National Publishing House of China, Beijing, **(1987)**.

Authors

Abdurahim Mahmoud, 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 is natural language processing.



Sediyegvl Enwer, has received her B.E. degree in Information Engineering from Capital Normal University, China, in 2013. Currently, she is a M.S. student in Signal & Information processing in Xinjiang University. Her research interest is Natural language Processing.

Abdusalam Dawut, received M.I.S. degree from Tokyo Denki University, Japan, in 2006, and received Ph.D. degree in Education technology from Tokyo Denki University, Tokyo, Japan, in 2009. He is a lecturer at Institute of Software of Xinjiang University since May2011. Currently, His research interests include Education technicalization and imformatization.

Copyright ⓒ 2016 SERSC

Palidan Tuerxun, received her M. S. degree in 1996 from Liaoning University, China and her Ph.D. degree in 2015 from Northwestern University, China. Since 1992, she has been working as a teacher at Xinjiang University, and since 2004, she was an associate professor in school of software of Xinjiang University. Her research interests are machine learning and Uyghur natural language processing.

Askar Hamdulla, 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 160 technical papers on speech synthesis, natural language processing and image processing. He is a senior member of CCF and an affiliate member of IEEE.

Copyright ⓒ 2016 SERSC