Kazakh Segmentation System of Inflectional Affixes

Gulila.Altenbek
1.Information Science and Engineering Colleges
Xinjiang University,
Xinjiang Lab. of Multilanguage Information Technology ,
830046，P.R. China.
2．Harbin Institute of Technology, Harbin
gla@xju.edu.cn

WANG Xiao-long
Institute of Computer Science and Technology,
Harbin Institute of Technology, Harbin,
150001, P.R. China.
wangxl@insun.hit.edu.cn

Abstract

This paper focuses on the automatic segmentation of inflectional affixes of the Kazakh Language (KL) on the basis of studying the corpus of KL. Kazakh is an agglutinative language with word structures formed by productive affixation of derivational and inflectional suffixes to stems. Based on the analysis of the configuration of inflectional affixes, it firstly constructs the Finite-State Automation and the segmentation of inflectional affixes. Secondly it targets at specially constructing the Finite-State Automations of nouns and verbs, which are the most changeable and complex part of speech of KL. And thirdly it adopts the methods of Bidirectional Omni-Word Segmentation and lexical analysis to achieve the goal of stemming and fine segmentation of inflectional affixes of KL. And finally it gives an additional account of studying the segmentation of ambiguous inflectional affixes. The paper intends to improve the accuracy and the quickness of stemming the inflectional affixes of KL.

1 Introduction

Lexical or morphemic analysis is to turn the character string of natural language into “the word string”. During the process, at first it takes “the word” out,, and then conducts the morphological analysis of the internal components of “the word”, and finally it ends up with the tagging. Many language processing tasks, including parsing, semantic analysis, information retrieval, and machine translation usually require a morphological analysis of the language beforehand.

As we know, Kazakh Language belongs to Turkish Language group of Altaic Language Family, whose unique language features decide that we should focus on its Inflectional Morphology is inflectionally changed. Kazakh language is written right-to-left in the Arabic alphabet with some modifications.

This paper attaches importance to analyzing the nouns and the verbs, which have great difficulties in affixes segmentation. And this paper will definitely contribute to the further study of lexical analysis of KL.

2 Related works

There have been some related studies, such as, Martin Porter has proposed ”English Stemming Processor” (1980), which is most widely used; The Longest-March put forward by Kut is a type of word Segmentation algorithm based on the Turkish Lexicon(1995).Beihang University has finished its CDWS Chinese Word Segmentation System (nan-yuan.Liang, 1987); Tsinghua University has also completed its SEG Chinese Word Segmentation System(Da-yang Shen et al.,1997); and <The Grammatical Knowledge-base of Contemporary Chinese> edited by Peking University was also published(Shi-Wen,Yu, 2003).

And the study of lexical analysis of minority languages has also achieved a lot in China, Some researches(A.Gulila and A.M i j i t，2004, K.Aykiz et al.,2006, YuSufu, 2005) have been done in the lexical analysis of Uighur Language conducted Xinjiang; the Automatic Segmentation System of Mongolian language conducted by Inner Mongolia University (U.Nasun, 1997)；And the lexical analysis of kazakh Language conducted by our project is in progress (A.Gulila and A.Dawel,2007) and so on.

There have been several main approaches or algorithms to segment inflectional affixes, including maximum matching algorithm based on mechanic matching of character strings, rules-based algorithm, statistics-based algorithm, and the combination of both rules-based and statistics-based algorithms.

3 Kazakh Morphology

3.1 Kazakh Morphology

Kazakh is an agglutinative language with word structures formed by affixes to grammatically or meaningfully change the words.Kazakh morphology is an affixal system consisting mainly of suffixes and a few prefixes. According to linguistic theory, the word of the text consists of the root or the stem and the affix.(Milat etc. 2003, ding-jin Zhang. 2004).

• *Word root* is the core of the whole word structure, which is the essential morpheme to convey the basic content of its meaning.

• *Word stem* is a new word generated by adding various affixes to the root, which is also called a derivative word. It expresses the complete and full meaning.

• *Affixes* are divided into inflectional affixes and derivational affixes. The study of derivational affixes focuses on the derivational words, which can be formed by adding prefixes or suffixes or prefixes plus suffixes. Meanwhile the meanings of the derivational words will be changed. While the study of inflectional affixes pays attention to the Inflectional Morphology, which shows grammatical changes between words but does not change word meanings.

3.2 The Analysis of Inflectional Affixes

We focus on most general morphological rules which are common rules related to morpheme segmentation. The inflectional affixes in Kazakhh language are divided into the following four types:

1) *Plural:* KL has six various affixes to express the plural form of words, which usually are directly linked to the general nouns, pronouns and numerals.

2) *Personal pronoun possessive*: KL has six various affixes to express the possessive forms of personal pronouns.

3) *Case*: KL has seven various affixes to express the different cases. So KL has seven cases. Case endings are applied only to the last element of a noun phrase, which are closely linked to the following verbs.

4) *Predicative Person*: The first, second and third personal pronouns are usually followed by the words with additive predicative personal elements.

The above-mentioned four types of inflectional affixes can be used separately or linked together. Suffixes in Kazakh are complex, especially when a stem is linked with many suffixes. There are some rules we can follow to add affixes to word roots. See Figure 1: (Right-to-Left)

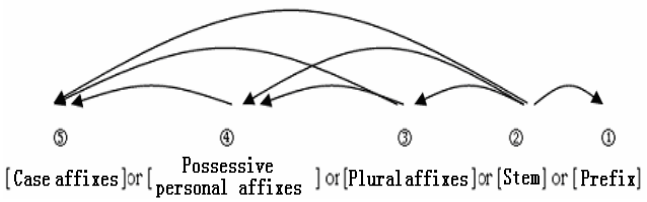

Figure 1. Rules to guide the connections of inflectional affixes

3.3 The Finite-state Automaton model of inflectional affixes of KL

Finite-State Automata (FSA) can be used to describe the possible word forms of a language. We have already applied the model of FSA into the lexical analysis of KL. The following figure shows a FSA model of inflectional changes of a noun. See Figure 2:

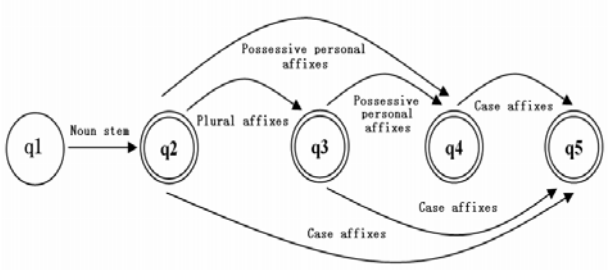

Figure 2. The FSA model of inflectional changes of a noun.

4 The Finite-state Transducer (FST) of Kazakh Words

As a typical agglutinative language, Kazakh words are formed by adding various suffix to word roots. But the Kazakh language itself does not have prefixes with exception of some borrowed or loaned prefixes from foreign words. And there are some rules guiding the usage and connection of various suffixes. Thus we can apply FST to establish a model for Kazakh words. The process of establishing a FST model can be divided into the following steps(E.Gülşen & A.Eşref. 2004):

Step 1: Establish a Right-to-Left FSM.

Step 2: Tag affixes

Step 3: Reverse the Right-to-Left FSM, and get a Non-deterministic Finite-state Automaton (NFA)

Step 4: Convert the NFA to a Deterministic Finite-state Automaton (DFA) and establish a Left-to-Right FSM.

The Kazakh words that can be added affixes to themselves are the followings: nouns, numerals, adjectives, pronouns, verbs, adverbs and so on. Among them, nouns and verbs are the most difficult parts of speech to be segmented. Take these two parts of speech as examples:

4.1 Inflectional Affixes of Nouns

Step 1: Establish a Right-to-Left FSM .

The four types of inflectional affixes can be added to stems under the guidance of some rules which also decide the FSM. We can apply the FSM to segment stens and we can analyze the FSM from right to left. See Figure 3:

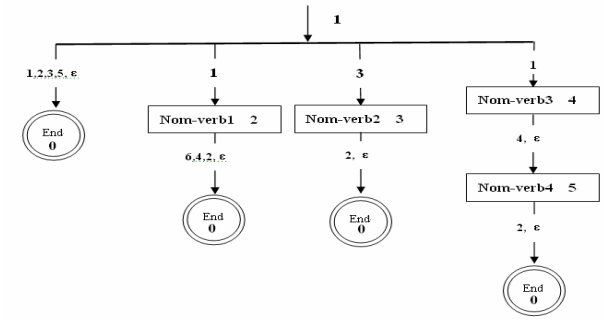

Figure 3. Right-to-Left FSM of inflectional affixes.

Step 2: Tag affixes

How to tag depends on the types of inflectional affixes, in which each type is given a value as its expressing value. Those affixes will be stored in the database as well as those expressing values. See Table 1

Table 1. Types and Expressing Values of Inflectional Affixes of Nouns.

inflectional affixes Type	value	Inflectional affixes type	value
Plural	1	Personal pronoun possessive: plural	4
Case	2	predicative person: singular	5
Personal pronoun possessive: singular	3	predicative person: plural	6

Step 3: Reverse the Right-to-Left FSM to form a Left-to-Right FSM

Reverse the Right-to-Left FSM, and form a Non-deterministic Finite-state Automaton (NFA) (See Figure 4). The number in each circle of Figure 4 represents the state value consistent with the state value of Figure 3. The types and expressing values are also marked above the lines in Figure 4.

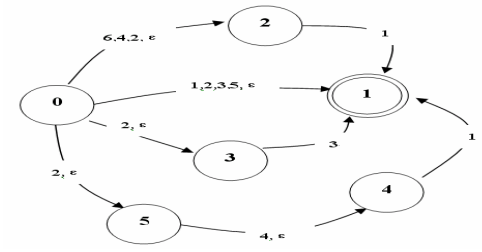

Figure 4. Left-to-Right NFA of inflectional affixes.

Step 4: Convert NFA to DFA

The multi-switches and "ε" switches of an expressing value of NFA makes the realization of NFA on computer very complex. Therefore, we should convert

the NFA to be a DFA with the purpose of making each inputted expressing value facing one switch and making "ε" switch nonexistent. We adopt “subset construction algorithm”[A.V.Aho et al. ,1986] to conduct the operation. We make each state of the new DFA correspondent to a subset of the NFA. As Table 2 shows, the start state (A)of DFA contains one element “0” and the start state of NFA. We know that all other states can be achieved from the state “0” through "ε" switches. Thus, the start state of DFA could be A={0， 1， 2， 3， 4， 5}. The numbers in the brackets represent inputted expressing values or types of affixes. The next state of DFA should be started with A and “1， 2， 3， 4， 5”can be separately inputted as expressing values. The FAS can thus be established.

Table 2. The Conversion from NFA to DFA of Inflectional Affixes.

E-closure({0})={0,1,2,3,4,5}*A E-closure(A,1)={1} *B	E-closure(C,1)={1} B
E-closure(A,2)={1,2,3} *C E-closure(A,4)={2,4} *D	E-closure(D,1)={1} B
E-closure(A,6)={2} *E	E-closure(E,1)={1} B

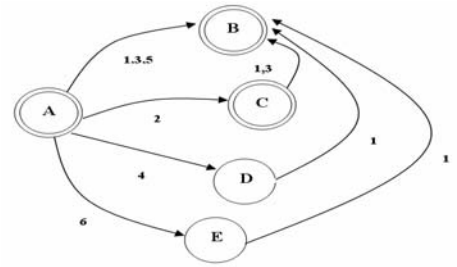

Figure 5. FSM of Inflectional Affixes of Nouns.
inflectional affixes of verbs.

5 Approaches to the Segmentation of Inflectional Affixes of Kazakh Words

Some mathematical frameworks or modeling methodologies can be used for morphology learning and word segmentation: maximum likelihood (ML) modeling, probabilistic maximum a posteriori (MAP) models, finite state automata (FSA), etc.

The algorithms suitable for the segmentation of inflectional affixes of Kazakh words include the followings: Bidirectional Maximum Matching and Omni-word Segmentation.

5.1 Bidirectional Matching Algorithm

Forward and backward algorithm is applied for the segmentation of a given word is examined for the words whose surface forms change after concatenation. The basic idea of this approach is to conduct the segmentation of inflectional affixes from left side of a character string to its right side and vice versa. But during the process, the critical issue is to determine the border between stens and inflectional affixes. Under many situations, vague borders between stens and inflectional affixes cause the inaccurate stemming. Thus, this algorithm can solve this problem.

5.2 Omni-word Segmentation Algorithm

The basic idea of this algorithm is to find all the segmentation forms of character strings waited for the segmentation starting from position “i”. For Kazakh text, we should find all the segmentation forms of a word. We just leave the issue of ambiguity for later discussion.

5.3 The Combination of Bidirectional Omni-word Segmentation Algorithm and The Lexical Analysis

1) The segmentation of inflectional affixes is conducted from the left side of a Kazakh word and then matched with the table of inflectional affixes. In general the inflectional affixes are formed by short character strings. Therefore some inflectional affixes maybe become sub-strings of other inflectional affixes. It is very possible that there are many successful matches of inflectional affixes, that is to say, there will be various segmentations of inflectional affixes of a word, But only one of them is accurate. So we need to classify the inflectional affixes and enact the rules to guide their connection order. According to those rules we just search one type of inflectional affixes and adopt Maximum matching algorithm to avoid the problem of many segmentations of an affix. When conducting the segmentation of inflectional affixes

and the stem extraction, we call the far right side of segmentation border “candidate border” .

2) The extract stems is conducted from the right side of a Kazakh word and then matched with the lexicon in order to find the candidate border of the sten. The ability to form new words for some affixes is very strong, so many stems which are added to various derivational affixes become new stems of other words. So the situation is the same with the segmentation of inflectional affixes. We should conduct Omni-word Segmentation and list all the possible segmentation forms of affixes.

We should deal with some special problems when conducting segmentation of inflectional affixes. Borders of stens will be changed somewhat when some inflectional affixes are added to the stens. Changes would occur like vowel deletion and lenition reduction. Sometimes it is impossible to find the complete match in the lexicon. Under such situation, we should apply orthographic rule of Kazakh language to deal with it.

6 The Analysis of the Ambiguity of Inflectional Affixes

6.1 Rule-based analysis of Ambiguity

To prevent over-segmentation and secure the semantic identity of a word, stem and suffix boundary is chosen as the primary target of segmentation.

Suppose the right border of inflectional affix is indicated as S1 while the left border of sten as S2. The ambiguity probably occurred in the segmentation is listed below as well as its solution.

1) S1=S2 (Under various situations, it is possible that S1 is S2): Under this situation we should segment the longest sten.

2) S1 ≠S2 (Under this situation, we also consider the following two cases see Figure 8)

Case 1:

(1) The sub-string on the right side of S1 will be regarded as the candidate stem. And then we should apply orthographic rule to make a choice of the candidate stem and the sub-string on the left side of S1.

(2) If some variants of the words do exist, a new sten is formed; otherwise, go to Step 5.

(3) We should search the new sten in the lexicon

(4) If we succeed to find the new sten in the lexicon, please tag the stem and the inflectional affix (the left sub-string of S1)

(5) End

Case 2:We should apply probability statistics to solve the problem.

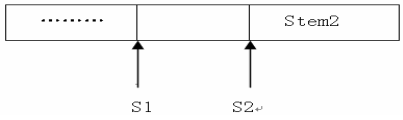

Figure 6(a). Case 1: (S1 ≠S2).

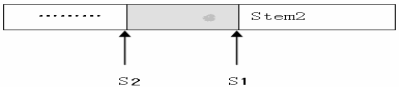

Figure 6(b). Case 2: (S1 ≠S2).

3) Non-matched stems but with matched inflectional affixes

We adopt the same solution to deal with this situation. That is to say, we at first should apply orthographic rule to make a choice of the sub-string on the left side of S1 and the sub-string on the right side of S1.And we also try to find the existence of lenition reduction. If we could not find the new sten in the lexicon, we should change to apply probability statistics to analyze.

4) Non-matched stens with non-matched inflectional affixes

We should search the inflectional affixes from the left side of the word to be segmented. If we could not find the match in the lexicon, we should judge the suffix and the sub-string on the right side by use of orthographic rule. And then we continue to search the candidate stem in the lexicon. If the match does exist, we tag the word as a sten; otherwise we tag it as an unregistered word.

5) Non-matched inflectional affixes with matched stems.

We consider the sub-string on the left side of the stem as the candidate derivational affix and search the match in the table of derivational affixes. If the match could be found in the table and its type is the same with

the stem , we should tag the word as a stem; otherwise we tag it as an unregistered word.

6.2 The ambiguity analysis based on Bayesian classification

The principle of Semantic Bayesian classifier is to consider the information of surrounding words of ambiguous words in a large context. Each practical word contains potentially useful information to imply the possible semantics of the ambiguous words. This Classifier is not a features selection but a combination of all features. The ambiguous words of the corpus should be semantically tagged in advance.Table 3 lists some symbols presented by this paper.

Table 3. Symbols Agreement.

Symbol	Meaning
W	An ambiguous word
s1,…,sk,…,sK	ALL the different segmentations of W
c1,…,ci,…,cI	the context in which W is in the corpus
v1,…,vj,…,vJ	the context features of the Disambiguation

When selecting the types, the Bayesian classifier using Bayesian decision-making rules could be used; those rules can minimize the error probability (R. O.Duda, P. E. Hart. ,1973).

According to simple Bayesian assumption, we have revised the decision making rules, as follows:

Simple Bayesian decision-making rules.

$$\text{Decide } S' \text{ if } S' = \text{argmax}_{sk}[\log P(s_k) + \sum_{vj \text{ in } c} \log P(v_j|s_k)] \quad (1)$$

$P(v_j \mid s_k)$ and $P(s_k)$ in the formula can be calculated using the maximum likelihood estimates from the tagging of training in Corpus:

$$\begin{aligned} s' &= \arg\max_{s_k} P(s_k/c) \\ &= \arg\max_{s_k} \frac{P(c/s_k)}{P(c)} P(s_k) \\ &= \arg\max_{s_k} P(c/s_k) P(s_k) \\ &= \arg\max_{s_k} [\log P(c/s_k) + \log P(s_k)] \end{aligned}$$

$$P(v_j \mid s_k) = \frac{C(v_j, s_k)}{\sum_t C(v_t, s_k)}$$

$$P(s_k) = \frac{C(s_k)}{C(w)} \quad (2)$$

C(vj,sk) in the formula is the number to show how many times sk is to be segmented by vj in the context of training materials; C (sk) is the number to show how many times that sk occur in the training corpus; and C (w) is the total number to show how many times the unambiguous words occur.

```
1   comment: Training
2   for all stemmings of w do
3         for all words vj in the vocabulary do
4               P(v_j | s_k) = C(v_j, s_k) / Σ_t C(v_t, s_k)
5         end
6   end
7   for all stemmings sk of w do
8               P(s_k) = C(s_k) / C(w)
9   end
10  comment:Disambiguation
11  for all stemmings sk of w do
12        score(sk)=logP(sk)
13        for all words vj in the context window c do
14            score(sk)=score(sk) + log P(vj| sk)
15        end
16  end
17  choose s'=argmax Sk score(sk)
```

Figure 7. Bayes Disambiguation.

7 The Design of the System

In the process of segmenting affixes in Kazakh language, the main task is to segment the prefixes, stems, and inflectional affixes. For this purpose, About 60,000 stems and 438 tables of affixes are collected as the basis of segmentation. The stem list consists of almost all the common stems except from the domain specific words and rarely used words. The realization of the whole system experiences the following steps:

1) Take a Kazakh word from a text.

2) Establish a FSM of a noun or a verb. If possible, directly give the result of segmentation and return to step 1;otherwise turn to the next step.

3) Adopt the combination of Bidirectional Omni-word Segmentation Algorithm and the Lexical Analysis to analyze for the words to be segmented. If possibly segmented, directly give the result of the segmentation and return to step 1; otherwise adopt Bayesian Classification by use of the parameters from the training corpus to select the correct segmentation of ambiguous words.

4) The result of tagging the segmentation of affixes .

The corpus contains 150，992 words, among which 51% is used as training corpus while the rest as test

corpus. The accuracy rates generated from the testes conducted for this paper include Precision 1 and Precision 2. We can define two evaluation functions, as follows:

Definition 1:The accuracy rate of inflectional affixes segmentation of words.

$$precision1 = \frac{\text{numbers of correct extracted stems}}{\text{total words}} \times 100\% \quad (3)$$

Definition 2: The accuracy rate of inflectional affixes segmentation of ambiguous words

$$precision2 = \frac{\text{numbers of correct extracted ambiguous words}}{\text{total number of ambiguous words}} \times 100\% \quad (4)$$

8 Experimental results

8.1 The comparison of the segmentation speeds

Table 5 shows the comparison of the segmentation speeds, in which we compare the system adopting FSM to the system not adopting FSM. We have tested 10，000 words and the result of the comparison is quite obvious, which indicates the high segmentation speed of the system adopting FSM.

Table 4. The comparison of Two segmentation speeds.

Type of segmentation	The number of tested words	Total time used for segmentation (Ms)	average velocity (Ms/words)
not adopting FSM	100,00	24，422	2.4422
adopting FSM	100,00	19，408	1.9408

8.2 The Analysis of the result of inflectional affixes segmentation

This paper adopts the combination of bidirectional omni-segmentation and rule-based segmentation to segment inflectional affixes and extract stems.

Table 5. The contrast of affix segmentations by use of different algorithms.

Algorithms	Precision1 （%）
Omni-word Segmentation	78.1
Maximum matching	74.2
Combination of bidirectional omni-segmentation and lexical analysis	84.0

The tests show that the final one improves the accuracy rate of affix segmentation and realizes the segmentation of inflectional affixes.

8.3 The Analysis of segmentation of ambiguous words

This paper puts forward that we should firstly adopt rule-based approach or algorithm to the segmentation of ambiguous words, if without any result, we should adopt Bayesian Classification to the segmentation of ambiguous words. In the test corpora, among 74,026 words 922 words are ambiguous words. So at first we should adopt rule-based algorithm to deal with those ambiguous words, in which 600 ambiguous words can be correctly dealt with; and then we should adopt Bayesian Classification Algorithm to further improve the accuracy rate of the segmentation of ambiguous words. As a result, the accuracy rate of the segmentation of ambiguous words can be reached to 84.38%. Table 7 shows the analysis of the segmentation of ambiguous words.

Table 6. The analysis of the segmentation of ambiguous words.

Algorithm to deal with ambiguous words	Total number of ambiguous words of test corpora	Number of correctly segmented ambiguous words	Precision2 （%）
Rule-based segmentation of ambiguous words	922	532	57.70%
Bayesian classification	390	246	63.07%

9 Conclusion and Future Study

This paper firstly analyzes the morphemic structure in the corpus of Kazakh Language, and especially studies stem extraction and affix segmentation. It establishes the FSM of inflectional affixes and then conducts the segmentation of inflectional affixes. The process starts with the analysis of FSM of the words to be segmented. If successfully achieved, the result would be considered as the result of segmentation. Otherwise, the algorithm of combining the bidirectional

omni-word segmentation and ruled based segmentation should be adopted to segment the inflectional affixes, which better solves the problem of segmenting inflectional affixes. At last the paper presents that we should apply the method of statistics to disambiguate the segmentation of inflectional affixes of ambiguous words. Compared to other approaches, this approach improves the accuracy rate and the segmentation speed of segmenting inflectional affixes.

But there exist other problems presented in this paper, such as unregistered words. We should continue to make efforts to improve the accuracy rate of the segmentation of inflectional affixes through further enlarging the vocabulary of the dictionary and adopting the method of statistics. And we should be well-trained in obtaining parameters of segmentation models of Kazakh Language, making the language model close to the reality language itself.

Acknowledgment

This work is funded by the Natural Science Foundation of China(NSFC)(No.60763005), And the Project of China Ministry of Education (No.MZ115-92).

References

A.Kut, A. Aplkoçak, E.Özkarahan. 1995.Bilgi bulma sistemleri için otomatic turkçe dizinleme yöntemi. In Bilişim Bildirileri, Dokuz Eylül University, İzmir, Turkey.

A.V.Aho,R.Sethi&J.D.Ullman. 1986. Compilers: principles,techniques,tools[R].Reading,MA:Addison Wesley.

A.Gulila ,A.M i j i t.2004 .Reseach on Uighur Word Segmentation, Journal of Chinese information processing ,l . 18(6):61-65.

A.Gulila, A.Dawel.2007.Study on the Rule-based Kazakh Word Lemmatization, 11TH Symposium national language and information Proceedings ,Xishuangbanna, 109-114.

E.Gülşen , A.Eşref. 2004. An affix stripping morphological analyzer for Turkish, Proceedings of the International Conference on Artificial Intelligence and Application,Austria,299-304.

K.Aykiz,K.Kaysar,I.Turgun,2006.Morphological Analysis of Uighur Noun for Natural Language Information Processing,Journal of Chinese information processing，20(3)：43-48.

Liang nan-yuan. 1987.The Mordern Printed Chinese Distinguishing Word System, Journal of Chinese information processing,l . 1 (2)：44-52.

M.F.Porter. 1980.An algorithm for suffix stripping", Program ,14(3)：130−137.

Milat etc. 2003.Contemporary Kazakh language, Xinjiang People's Publishing House.

R. O.Duda, P. E. Hart. 1973. Pattern Classification and Scene Analysis,John Wiley and Sons, New York,10-43.

Shen Da-yang,Huang Chang-ning,Sun Moa-song, 1997. The approaches of Information integration and bestpath seaching inCWASS, Journal of Chinese information processing, l . 11 (2）：34-47.

U.Nasun, 1997 .The automatic segmentation system of Mongolian roots, stems, word, Journal of Inner Mongolia University ,NO2：53-57.

YuShi-Wen,2003.TheGrammatical Knowledge-base of Contemporary Chinese-A Complete Specification, Tsinghua University Press.

Zhang ding-jin. 2004.Modern Kazakh language practicality grammar，The central University for Nationalities Publishing House.

Yusup Abaidula ， Rezwangul, Abdiryim Sali. 2005.The Research and Development of Computer Aided Contemporary Uighur Language Tagging System. Journal of Chinese Language and Computing.