International Journal of Control and Automation
Vol. 9, No. 11 (2016), pp.143-158
http//dx.doi.org/10.14257/ijca.2016.9.11.13

A Survey on the Methods for Uyghur Stemming

Abdurahim Mahmoud[1], Abdusalam Dawut[2], Peride Tursun[2]
and Askar Hamdulla[1*]

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

Abstract

Stemming is a very important natural language processing task that to remove all inflectional affixes from a word, and to lemmatize the remaining part of the word. It is very important in most of the Information Retrieval systems. In this paper we have introduced different stemming algorithms that commonly used in English and other English-Like European languages, and also introduced methods used in Chinese word segmentation, finally we have carried out a more detailed discussions about the methods for Uyghur stemming.

Keywords*: Stemming, Word segmentation, Uyghur Language*

1. Introduction

Improving recall rates is the most important problem in information retrieval systems. A simple recall rate improving method is stemming. An algorithm which maps different morphological variants to their base form (stem) is called a stemming algorithm [1]. In Information Retrieval systems stemming is usually done by removing any attached suffixes and prefixes from index terms before the actual assignment of the term to the index [2].

Chinese has no stemming process, but word segmentation in Chinese is a very important task like a stemming in other languages. While implementing Chinese word segmentation we can use some useful methods used in stemming process of other languages and we can also find some special word segmentation methods based on the characteristics of the Chinese.

Uyghur (refers to the Uyghur Language) is an agglutinative language with most complex morphological structure, so the stemming process of Uyghur is more complicated than other languages. Just using a single stemming method is no effective to Uyghur, we must master a lot of linguistic knowledge, and must find most effective stemming methods according to these knowledge.

In this paper, we will provide a brief review of the evolution of stemming algorithms in the last decades in English and other English-Like European languages, and introduce methods used in Chinese word segmentation, finally we will introduce the related methods in Uyghur stemming in more detail.

2. Stemming Algorithms Used in English and Other English like European Languages

According to the stemming principle, stemming algorithms can be divided into four categories, respectively are rule based (truncating, affix removal) method, dictionary look-up (table lookup) method, statistical method and mixed method.

* Corresponding Author

ISSN: 2005-4297 IJCA

2.1. Rule based Method

Rule based method extracts stems or removes suffixes according to morphological variation of specific language. For example, in English, most of countable nouns change their plural form by adding suffix "s" at the end of these words, and nouns ending with “s”, “z”, “x”, “ch”, “sh” change their plural form by adding "es" at the end of these words. Stemmers can remove suffixes and restore stems according to these rules. Rule based methods can be subdivided into simple deletion method and the longest matching method. Simple deletion method is simple and fast, but the accuracy is low. The longest matching method matches word and suffixes, and deletes the longest matched suffix to obtain stem. The longest matching algorithm has a high accuracy, and widely used in English and other languages similar to English. The most representative algorithms are Lovins Algorithm [3], Dawson algorithm [4], Porter algorithm [5] and Paice/Husk algorithm[6].

2.1.1. Lovins Algorithm

This algorithm was first proposed by Julie Beth Lovins in 1968, it is the first stemming algorithm mentioned in literature. It is a one-way, context sensitive algorithm. It consists of `two steps: suffix stripping and treatment of the remaining stem. It performs a lookup on a list of 294 endings, 29 conditions and 35 transforming rules [3]. It removes the longest suffix from a word.

This algorithm can handle removal of double letters in words, for example, words like ‘getting’ can be transformed to ‘get’ and also can handle many irregular plurals like index and indices, matrix and matrices *etc*. But the main shortcomings of this algorithm are it is time and data consuming, and many suffixes are not available in the endings.

2.1.2. Dawson Algorithm

Dawson algorithm was proposed by John Dawson in 1974[4], it improved the rule and matching method of Lovins’ algorithm. Dawson combined all the plural and simple affixes and increased the endings list. This algorithm also combined all word variants according to suffix list and matching rule.

2.1.3. Porter Algorithm

Porter stemming algorithm is one of the most popular stemming methods proposed by M.F.Porter in 1980[5]. It is widely used in information retrieval systems, such as Lucene, Solr, Google, and so on, mainly used in term standardization [7]. This algorithm comprises of a set of conditional rules. These conditions are either applied on the stem or on the suffix or on the stated rules. It has five steps, and within each step, rules are applied until one of them passes the conditions. If a rule is accepted, the suffix is removed and the next step is performed. The resultant stem at the end of the fifth step returned.

In Porter algorithm, any word can be expressed as:

$[C](VC)^m[V]$

Where C represents a list of consonants, V represents a list of vowels and m represents the measure of any word. For example:

m=0 TR, EE, TREE, Y, BY

m=1 TROUBLE, OATS, OATS, TREES, IVY

m=2 TROUBLES, PRIVATE, OATEN, ORDERY

The rule looks like the following:

<condition> <suffix> → <new suffix>

For example, a rule (m>0) TIONAL → TION means “if the word has at least one vowel and consonant plus TIONAL ending, change the ending to TION”. So “national” becomes “nation”.

2.1.4. Paice/Husk Algorithm

The Paice/Husk algorithm [6] is a simple iterative algorithm. It has 120 rules indexed by the last letter of a suffix. It uses a rule file, and extracts the last character of the word to find an applicable rule from this rule file. If find a rule, delete or replace the character according to the rule.

2.2. Dictionary Look-Up (Table Lookup) Method

In this method, words and their corresponding stems can be stored in a dictionary (table). Stemming is done by looking up the dictionary. Using B-tree or Hash table, the speed of looking up would be very fast.

There are two problems with this approach. The first is that to make these dictionaries we need a large amount of work on a specific language and these dictionaries can't include all words. The second problem is that the storage overhead of these dictionaries is particularly large. The representative dictionary searching method is Krovetz algorithm [8].

Krovetz algorithm was proposed by Robert Krovetz in 1993. In this method, dictionary constituted by several lists such as basic word list, additional word list, exceptional word list, proper noun list and so on. In Krovetz algorithm, suffix would be removed firstly, Then, the dictionary would be detected, finally return the stem. It is an easy modifying, more flexible method, but it is relatively more dependent on dictionary coverage.

2.3. Statistical Methods

2.3.1. Successor Variety Method

This method proposed by Hafer and Weiss [9]. It is different from the affix removal method. It identifies the boundary of stem and suffix by the character distribution in text set, and directly extract stem. The successor variety of BEAT is shown in Table 1. The successor variety of a string is the number of different characters that follow it in words in a text set. Consider a text set consisting of the following words, for example.

BEATABLE, BATTLE, BEACH, BOY, BODY

Table 1. Successor Variety of BEAT

Prefix	The number of successor variety	Characters
B	3	E,A,O
BE	1	A
BEA	2	T,C
BEAT	1	A

When there are a large text set, the successor variety of substrings of a term will decrease as more characters are added until a segment boundary is reached. If the successor variety of a substring is very low, it may be a stem.

2.3.2. Text Set Method

Xu proposed this method [10]. There is an assumption that the merged word forms will co-occurrence in the text set. Based on this assumption, we can apply the co-occurrence distribution measurement, and perform an equivalence partitioning.

Paik also proposed a new stem extracting method [11] based on the co-occurrence statistics. The algorithm is evaluated by the retrieval results, found that it is based on statistic rule, can handle different languages.

2.3.3. N-Gram Method

Adamson and Boreham proposed a stemming method called the shared digram method in 1974 [12]. A digram is a string composed of adjacent 2 letters extracted from a text. We can use trigrams and hence it is called n-gram method in general. This is a language independent method. For example, the term production and productive can be broken into digrams as follows.

productive ---> pr ro od du uc ct ti iv ve

production ---> pr ro od du uc ct ti io on

Each word has 9 digrams, and the two words share 7 unique digrams:pr, ro, od, du, uc, ct and ti.

Once the digrams of all words in a corpus have been identified and counted, we can compute their similarity measure. Adamson and Boreham used Dice's coefficient to compute similarity measure.

Dice's coeffient of a pair of words defined as:

$$S = \frac{2C}{A+B} \quad (1)$$

where A is the number of unique digrams in the first word, B is the number of unique digrams in the second, and C is the number of unique digrams shared by A and B. For the two term above(production and productive), Dice's coeffient equal to (2*7)/(9+9)=0.78. According to similarity, all the word pairs can be clustered as groups. From the value of Dice coefficient we can extract the first unique seven digrams as stem.

Mayfield and McNamee also proposed a n-gram based method in 2003 [13]. They analyze all n-gram distribution in a textual data, and came to such a conclusion that a word root(stem) generally occurs less frequently than its morphological form. A typical statistical analysis based on the inverse document frequency (IDF) can be used to identify them.

The advantage of n-gram method is language independent and so it can be used in many languages, but these is a disadvantage that it requires a large amount of memory and storage for creating and storing the n-grams and hence is not a very practical approach.

2.3.4. HMM Method

This stemming method is based on the concept of the Hidden Markov Model(HMMs) [14] which are finite state automata where transitions between states are ruled by probability functions. At each transition, the new state emits a symbol with a given probability. Melucci and Orio proposed a stemming method based on HMMs in 2003 [15].

In this method each letter of a word is considered to be a state, and the states are divided in two disjoint sets (stems and suffixes): initial can be the stems only and the later can be the stems or suffixes. Transitions between states define word building process. For any given word, there are many probable paths, the most probable path is found using the Viterbi algorithm [16]. The most probable path from initial to final states will produce the split point (a transition from roots to suffixes). Then the sequence of characters before this point can be considered as a stem.

The advantage of this method is it is based on unsupervised learning and does not need a prior linguistic knowledge of the dataset. The evaluation result of many European languages stemming shows that it is effective in the treatment of multiple languages. But this algorithm is very complex and sometimes over stemming may occur.

2.4. Mixed Method

Mixed method mainly adopts two or more methods to solve the stem extraction problem. For example, It could be a combination of n-gram method and table lookup method. Silva and Oliveira proposed a mixed method to solve the stemming problem of Portuguese language [17], this method adopted both table lookup method and affix removal method.

There are also some improved mixed methods based on the existing methods. E.T.Alshammari reduced the error rate by adding syntactical knowledge in his English stemming algorithm [18]. Alshammari found the role of stop words on stemming, and divided the stop words into two categories: useful and useless stop words. This algorithm used useful stop words to identify POS, built three dictionaries based on different syntactical knowledge: nouns dictionary, verbs dictionary and adjectives dictionary. Then by deleting stop words, deleted affixes through longest matching method. Compared with the Porter algorithm and the Lovins algorithm, it achieves a relatively good effect.

Recent years, a lot of non-English mixed stemming methods are proposed, for example, a hybrid algorithm for Polish proposed by Dawid Weis [19], hybrid algorithm for Nepali stemming proposed by Chiranjibi Sitaula [20], hybrid inflectional stemmer for Gujarati proposed by Suba, Jiandani and Bhattacharyya [21], hybrid stemmer for Arabic proposed by Hadni, Ouatik and Lachkar [22].

3. Chinese Word Segmentation

Unlike some languages that have stem-affix structure such as English, Turkish, Uyghur *etc*, there is no affix before and after the Chinese word, but in Chinese, there are no spaces between words. Therefore, in order to further analyze the Chinese, the Chinese text must be carried on word segmentation.

Chinese word segmentation is a basic work of Chinese information processing field. It is very important for information retrieval, speech recognition, automatic translation, and data mining and so on. The accuracy of Chinese word segmentation directly affects the development of Chinese information processing technology.

3.1. Classification of Chinese Word Segmentation Algorithms

Chinese word segmentation research can be divided into these sub-researches as follows: segmentation algorithm, ambiguity elimination, unknown words recognition and word segmentation and word segmenting evaluation.

The main methods of Chinese word segmentation can be categorized to four types such as dictionary based method, understanding based method, statistical based method and mixed method.

3.1.1. Dictionary Based Method

In this method, according to a certain strategy, the Chinese character string is matched with the entries in the dictionary, if a string is found in the dictionary, the matching is successful. There must be considering three essential factors: dictionary structure, lookup direction and matching rule. According to different lookup direction and matching rule, dictionary based method divided into four type's namely forward maximum matching method, reverse maximum matching method, bidirectional maximum matching method and minimum segmentation method.

The quality of the dictionary structure directly affects the performance of the algorithm. There are three factors which affects dictionary performance [23]: 1) Word query speed; 2) Dictionary space utilization; 3) Dictionary maintenance performance.

Sun Maosong, Zuo Zhengping and HuangChangning [24] designed three typical word dictionary mechanism: the entire word dichotomy, trie index tree and verbatim

dichotomy, and compared their time and space efficiency. Chen Guilin *et al* improved segmentation effect by using a high effect Chinese electronic thesaurus data structure [25].

3.1.2. Understanding Based Method

In this method, syntactic, semantic analysis are carried on, and syntactic and semantic information are used to deal with ambiguity. This method requires a large number of language knowledge and information.

He Kekang, Xu Hui and Sun Bo [26] analyzed the causes of ambiguous segmenting, divided the ambiguity field into four categories, and gave effective methods to eliminate each kind of ambiguities. Lin Yaping [27] and Yi feng [28] used BP neural network to design word segmentation system. They carried out a large amount simulation experiments and achieved good segmentation effect.

3.1.3. Statistical Based Method

The main idea of this method is that a Chinese word is a stable combination of some Chinese characters. Therefore the more times a combination of Chinese characters, the more likely it is to form a word. That is to say, the occurrence frequency of Chinese characters combination can reflect word forming credibility. Therefore, through the frequency statistics, the statistical information can be calculated and can be the basis of word segmentation. There are some kind's word segmentation models such as n-gram model, maximum entropy model (MEM), hidden Markov model (HMM), Sun Maosong *et al* proposed a method [29] that could solve crossing ambiguity segmentation field by using mutual information and t-test difference between adjacent Chinese characters in sentences. Sun Xiao anf Huang Degen proposed a directed- graph model based on the maximum and second-maximum matching method [30].

3.1.4. Mixed Method

There are two commonly used mixed approaches, first is dictionary and statistics combination, and second is segmentation and POS tagging combination.

Zhai Fengwen, He Fengling and Zuo Wanli proposed a dictionary and statistics combination approach [31]. Firstly, they carried on word segmentation through dictionary based method, and then used statistical method to deal with the ambiguity and OOV.

Baishuanhu [32] combined automatic word segmentation and HMM POS tagging, He used bigram POS statistical rule extracted from manually annotated corpus to eliminate segmentation ambiguity.

3.2. Chinese Word Segmentation Systems

SCWS(Simple Chinese Word Segmentation) [33] is a mechanical Chinese word segmentation engine based on word frequency dictionary, its accuracy rate reached 90%-95%, can be used in some applications such as small search engine, keyword extraction and so on.

ICTCLAS (Institute of Computing Technology, Chinese Lexical Analysis System)[34] is one of the earliest open source segmentation project , its key features include Chinese word segmentation, POS tagging , named entity recognition and new word recognition. It can segment 996KB words per second, and its accuracy reached 98.45%.

Hylanda intelligent Chinese word segmentation system[35] solved the ambiguity segmentation and new word recognition problem, its segmentation accuracy reached 99.7%.

4. Uyghur Stemming

Uyghur belongs to the Eastern branch of the Turkic group of the Altaic language family [36]. It is an agglutinative language with rich and complex morphology [37].

4.1. Features of Uyghur Word

4.1.1. Uyghur Alphabets

There are 32 letters in Uyghur, including 8 vowels and 24 consonants. Sentences in Uyghur consist of words, which are separated by space or punctuation marks, therefore Uyghur word segmentation is very simple. Uyghur words consist of some smaller morphological units without any splitter between them.

At present, Uyghur is written in Arabic script with some modifications [38], but for convenience, we sometimes use the Latin script to write Uyghur words. The comparison between the two scripts is shown in Table 2. In this paper, we will use Latin script.

Table 2. Comparison between Arabic and Latin Scripts

Arabic	ئا	ئە	ب	پ	ت	ج	چ	خ	د	ر	ز	ژ	س	ش	غ	ف
Latin	A a	E e	B b	P p	T t	J j	CH ch	X x	D d	R r	Z z	Zh zh	S s	SH sh	GH gh	F f
Arabic	ق	ك	گ	ڭ	ل	م	ن	ھ	و	ۇ	ۆ	ۈ	ۇ	ې	ى	ي
Latin	Q q	K k	G g	NG ng	L l	M m	N n	H h	O o	U u	Ö ö	Ü ü	W w	Ë ë	I i	Y y

4.1.2. Syllabic Structure of Uyghur Word

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) [39].

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 shown in Table 3. Syllable can be considered very important information for stemming research. Therefore syllable segmentation is a very important basic work.

Table 3. Common Syllable Structure

Syllable structure	V	VC	CV	CVC	VCC	CVCC
Example word	u	al	bu	pul	eyt	sort
Meaning in English	He	take	this	money	say	breed

The following is a syllable segmentation example:
mëning akam oqutquchi (My brother is a teacher)
më + ning a + kam o + qut + qu + chi

4.1.3. Morphologic Structur of Uyghur Word

The morphologic structure of Uyghur word is "[prefix] + stem (or root) + [suffix1+suffix2+…]". That is to say, a Uyghur word is composed of three parts namely are prefix, root and suffix. In this, contents in parentheses are optional.

Root can express the basic meaning of a word and can be used independently, and cannot be segmented furthermore. Stem is the part of the word that is common to all its inflected variants.

Prefix is a morphemic part that attached to the front of the stem (or root). Not all words have prefixes, and according to the role in words, prefixes can be divided into two types namely are derivational and inflectional. If the prefix attached to the stem (or root) is a derivational prefix, it will bring a semantic change for the original word, if it is a inflectional prefix, it will enhance the original meaning of the word. Figure 1. is an example of two kinds prefix.

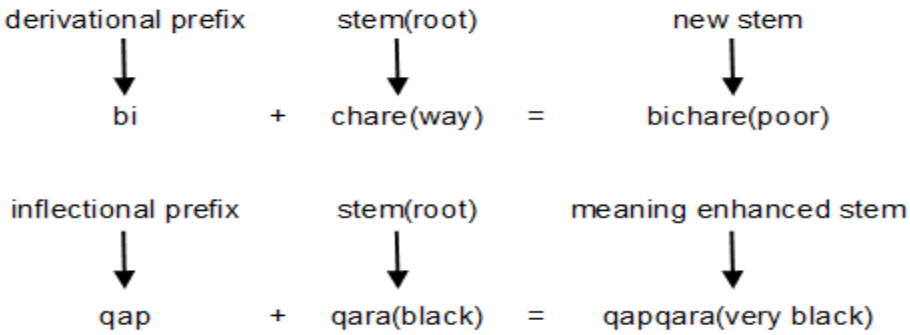

Figure 1. An Example of Derivational and Inflectional Prefixes

The number of prefix in Uygur is very small. Ablimit M only used 7 prefixes in his research [38]. Stems can be attached prefix are rarely. Whether it is a derivational prefix or inflectional prefix, there is only one prefix can be attached before stem, if needed.

Suffix is also a morphemic part attached to the rear of the root by zero to many (longest is about 10 suffixes or more). Suffixes are also divided into two types namely are derivational suffix and inflectional suffix. According to researchers' statistics, there are 532 singular-suffixes, including 243 derivational suffixes [40] and 289 inflectional suffixes [41]. Derivational suffix causes semantic change, that is to say, it forms a new stem. Inflectional suffix causes syntactic change. In most cases, first attach derivational suffixes, and then attach inflectional suffixes or directly attach inflectional suffixes.

There are two steps of stemming, the first step is to find the borders of stem and inflectional affixes (prefix and (or) suffixes), and the second step is lemmatization. Why

has the necessity of lemmatization? The reason is that the attachment of inflectional suffixes may cause some kinds of language phenomenon (phonetic changes) to the last few letters of stem such as insertion, deletion, phonetic harmony, and assimilation [42] [43]. Figure 2 is an example of two kinds of suffix.

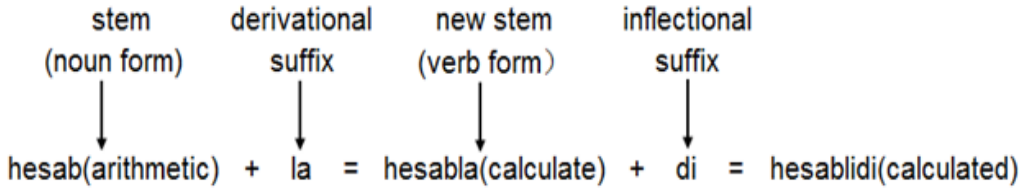

Figure 2. An Example of Derivational and Inflectional Suffixes

4.1.4. Phonetic Changes in Uyghur Word

There are three kinds of Phonetic changes in Uyghur word such as insertion, deletion and Assimilation (weaking), shown in Figure 3, Figure 4 and Figure 5 respectively.

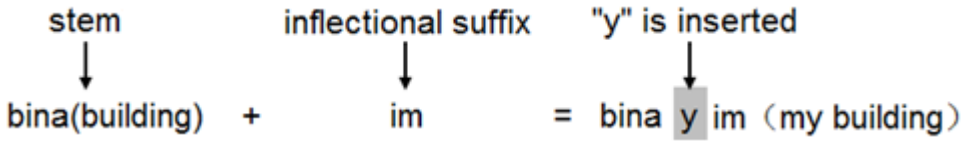

Figure 3. Insertion

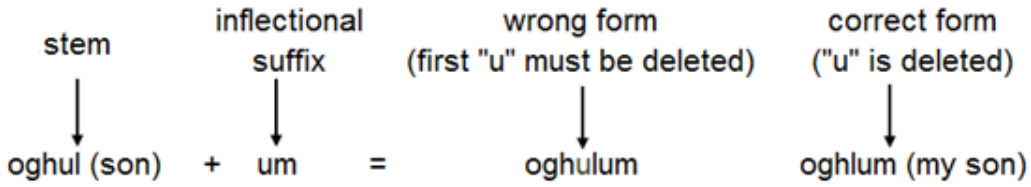

Figure 4. Deletion

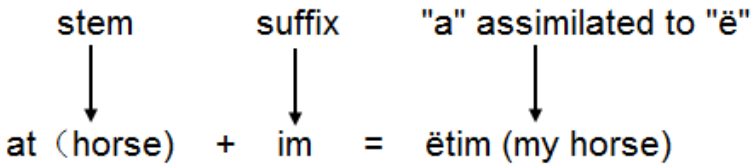

Figure 5. Assimilation (Weakining)

4.2. A Review of Uyghur Stemming Research

Researches on Uyghur stemming started relatively late than English and Chinese. Uyghur stemming was also used some of the methods which used in English stemming and Chinese word segmentation, but the direct usage of these methods is not suitable for Uyghur stemming, because the morphological structure of Uyghur word is not same to English and Chinese. It is more complex than the structure of English words. Words in Uyghur are separated by space or punctuation marks, so word segmentation in Uyghur is not complex like in Chinese. But as an agglutinative language with rich morphologic changes, Uyghur stem extraction must find a proper way according to its structural feature. In the following content, we will expound some published papers about Uyghur stemming.

In the following content, we will expound some published papers on Uyghur stemming according to the time sequence.

Gulila Adongbieke and Mijit Ablimit [42] proposed methods of handling with the basic phonetic features of Uighur words, such as the final vowel change, rules of vowel and consonant harmony, and syllable segmentation. They established a Uyghur word dictionary, affix dictionary and two kinds lexical rule dictionaries based on regular words and irregular words, respectively. Then extracted stems according to the "prefix + root" and "root + suffix" structure by using the Boyer-Moore algorithm and Forward maximum matching algorithm respectively. After stemming process, they also carried out lemmatization according to the phonetic assimilation rule. They carried on some experiments on regular words from scientific publishing on Xinjiang, and achieved 95% accuracy.

Chen Peng solved the stemming problem in his Master's thesis by combining the bidirectional matching algorithm and Omni word segmentation algorithm [44]. Compared with the maximum matching algorithm, this method improved the precision of the stem segmentation. In his thesis, the improved binary-seek-by-character dictionary query mechanism is employed in the application of Uyghur stem segmentation and it can improve the efficiency. The best stemming accuracy Chen Peng reached is 91%. Chen Peng also listed 5 types of stemming ambiguities and proposed different ambiguity elimination methods, and achieved 88.6% ambiguity excluding rate.

Aykiz.Kadir *et al* introduced the formalized morphological description and analysis of Uyghur noun (number, person and case *etc*.) [45]. They pointed out the essential morphological parameters of Uyghur noun, summarized the rule of its composition, statistical type and gave a method for paring suffixes, this approach provided an effective way for noun analysis in Uyghur language information processing. They gave a more detailed description on noun's number, person and case categories. They found 120 different noun forms and listed some of them in a table. These analyses provided important theoretical basis for rule based Uygur stemming.

Mijit Ablimit *et al* introduced a Uyghur word analyzer [46]. They built four tables in the form of a database table namely are root table, combined suffix table, singular suffix table and syllable table and analyzed syllable rule, harmony rule and assimilation rule of Uyghur word, provided the corresponding processing algorithms. They proposed two methods for root-suffix segmentation namely are forward search and reverse search. They used minimum edit distance algorithm to calculate the similarity between misspelled word and candidate word, and according to this value chose a candidate word for the misspelled word.

Mijit Ablimit *et al* proposed a partly supervised morpheme segmentation method [43], and implemented a morpheme Segmentation based on this rule based and n-gram method. In their study, they built a corpus with 38500 stems, 325 singular suffixes, and 5880 compound suffixes. They carried out an evaluation with 18400 words, and the accuracy of stem-suffix boundary detection reached 96%, the accuracy of all stem/suffix segmentation reached 92%.

Arzugul.Xerip studied the complex features of verbal suffixes in linguistic point [47], she divided the characteristics of verbal suffixes into 6 kinds namely are classification, grammatical form, aspect, tense, number, person, additive and position. She counted 19 kinds derivational suffixes and their 56 variants, 8 kinds of inflectional suffixes and their 71 variants, 22 kinds of derivational-inflectional suffixes and their 79 variants. This study is very useful rule based Uyghur stemming.

Aishan.Wumaier *et al* proposed Uyghur noun stemming method [48]. This method was based on Finite State Machine (FSM) and Dictionary Look-up algorithm. They counted 49 noun suffixes, these suffixes can be divided into three category namely are number, person and case. Their attached sequence is:

Stem+[suffix in number category]+[suffix in person category]+[suffix in case category]

Noun suffixes in Uyghur have their attachment rules. They listed these rules and built a FSM according to these rules. Then they implemented a stemming algorithm by using the FSM. They tested 16480 words, and reached 78% accuracy.

Batuer Aisha and Maosong Sun proposed a statistical tokenization [37], they constructed a raw Uyghur corpus denoted UC, with 594172 words, and further constructed two corpus from UC, one was manually stemmed corpus denoted UCS, another one was manually lemmatized corpus denoted UCL.

Then they used a two-step statistical process for Uyghur tokenization. In the first step, they used the Maximum Entropy model (ME) to learn knowledge of Uyghur word structure statistically from UCS. They tried a number of feature template schemes by experiments to fix the best one that can achieve the best stemming performance. In the second step, they used the Conditional Random Fields model (CRFs) to learn the "mapping" knowledge between "quasi-words" and real words from the paralleled UCS and UCL. This method was the first work of statistical tokenization method for Uyghur. The F measure of tokenization reached 88.9% in the open test.

Aishan Wumaier *et al* proposed a new Uyghur noun stemming method in 2009[49]. They used two steps to implement Uyghur noun stemming. In the first step, they implemented Uyghur noun stemming by using the Uyghur suffix FSM. They tested the Uyghur noun suffix FSM with 55625 inflected words, and found 6239 wrong stems which was over-stemmed, this problem caused by some ambiguous suffixes, so in the second step, they used CRF to resolve the ambiguity problem caused by the FSM. They built a CRF training corpus with 55625 words which include 17317 ambiguous suffix words, in which 6239 suffixes were not correct suffixes, and 11078 suffixes were correct suffixes. Eventually, they got this result: the recall rate was 88.78% if only use the FSM, and the recall rate was 94.04% if using both FSM and CRF. That's to say, the usage of CRF method brought 5.26% improvement to the recall rate.

In 2012, Azragul, Qixiangwei and Yusup Abaydulla developed a Uyghur stemmer [50] They used a dictionary based method to implement stemming. First, a word was searched in the stem dictionary, if not found, split the word by using suffix dictionary, and the candidate word obtained by splitting was searched in the stem dictionary, if not found, they extracted stem from this word manually, and added the manually extracted stem to stem dictionary.

Azragul *et al* carried out statistics on frequency of different type of stems such as all stems, high-frequency stems (90% coverage), top ten pure stems, and top ten stems with suffix from 9 Uyghur websites. These statistical data was very useful for further research.

Zaokere Kadeer *et al* developed a noun stemming system based on hybrid method [51], this is a four layer hybrid stemming approach. First the system searches a word from dictionary, if find it, then output it as a stem. If cannot find, then use FSM. If FSM encounters a affix ambiguity, then use the Maximum Entropy (ME) to eliminate the ambiguity. Finally, system will use the noisy channel model to resolve the vowel weakening problem. They used a Uyghur noun corpus with 61476 words, their experimental results show the best accuracy of this system is 95.6%.

In 2013, Tayier Abuduwaili *et al* proposed a Uyghur verb stemming method [52], as we all know, as all we know, in Uyghur, the attaching form of verb suffixes is most complex, therefore, the accuracy of the verb stemming mostly affects the entire stemming system. This method which Tayier proposed is based on a tagged dictionary and rules and the accuracy of their method reached 84.15%.In the same year, Arzugul. Xerip *et al* proposed a formalized description of Uyghur verb morphology [53], they described the multi-leveled verbal morphology layers in Uyghur by using Context Free Grammar (CFG), although they didn't implement any actual stemming system, but this description is very useful for the stem algorithm and automatic generation of Uyghur verbs.

In 2015, Abdurahim Mahmoud, Akbar Pattar and Askar Hamdulla proposed a new Uyghur stemmer [39]. In this stemmer, they built a training corpus with 5900 sentences,

in which include 70300 words. They used a syllable segmentation algorithm and obtain a syllable segmented corpus with 192400 syllables, and each syllables are manually tagged as a part of stem or as a part of affix. They used 31 different tags to express the relationship among prefix, stem, suffix and syllable, and also considered all kinds of morphological changes such as insertion, deletion, assimilation *etc*. Thus, they resolved the lemmatization by syllable tagging. Their evaluation result shows that the best F-Scores in syllable level are 99.25% and 98.73% respectively in close test and open test.

Sediyegvl Enwer *et al* proposed a multi-strategy Uyghur stemming approach in 2015[54], they further improved the result of Mijit's work [Ablimit M, Eli M,Kawahara T. Partly Supervised Uyghur Morpheme Segmentation. In:Proc. of the Oriental-COCOSDA Workshop. 2008]. In this method, they combined the POS and context information under the n-gram framework to most effectively eliminate stemming ambiguities. Experimental results show that they achieved 96.6% accuracy, compared with pure n-gram method, the accuracy of this method increased 1.58%.

4.3. The Existing Problems in Uyghur Stemming

From the various methods related to Uyghur stemming mentioned above, we can find some problems in Uyghur stemming.

There are very little more comprehensive studies on Uyghur stemming, some proposed approaches only resolved noun stemming problem, and another approaches only considered the Uyghur verb stemming. Some researchers only analyzed the noun structure or verb structure of Uyghur, but didn’t proposed any actual stemming approaches.

In Uyghur, no any public, large scale standard corpus and dictionary which used for stemming, every one established their own corpus or(and) dictionary with very small amount of vocabulary. Many corpus didn't pass the inspection of language experts.

There are no any public, high-accuracy standard Uyghur stemmer, all stemmers are only used in their labs and the accuracy is not very high.

5. Conclusions

The morphological changes of English and English-Like European languages are relatively simple, and many famous stemming algorithms are proposed for English stemming. It is very difficult to use these methods without any changes for the agglutinative languages with complex languages such as Uyghur. At the same time, there is a big difference between Uyghur and Chinese. Chinese text no any stemming problem, but there is no space between the Chinese words. Therefore, Chinese text must implement word segmentation. So Uyghur stemming also can’t use Chinese word segmentation methods without any changes. We must find better ways that suitable for Uyghur morphological structure. And we should strengthen the construction of large-scale standard corpus.

We hope that this survey will be helpful for other researchers’ further work on Uyghur stemming and other natural language processing works.

Acknowledgments

This work has been supported by Innovation Program for Excellent Ph.D. Candidates of Xinjiang University (XJUBSCX-2012010), the National Natural Science Foundation of China under grant of (61562081), and High Technology Research and Development Project of Xinjiang (201312103).

References

[1] W. Kraaij and E. P. Ren, “Porter's stemming algorithm for Dutch”, Informatiewetenschap Wetenschappelijke Bijdragen Aan De Derde Stinfon Conferentie, **(1994)**, pp.167--180.

[2] A. G. Jivani, "A Comparative Study of Stemming Algorithms", International Journal of Computer Technology & Applications, vol. 2, no.6, **(2011)**.
[3] J. B. Lovins, "Development of a Stemming Algorithm", Mechanical Translation & Computational Linguistics, no.11, **(1968)**, pp.22-31.
[4] M. F. Porter, "An algorithm for suffix stripping", Program, vol.14, no.3, **(1980)**, pp.130-137.
[5] J. Dawson., "Suffix removal and word conflation", ALLC Bulletin, vol. 2, no.3, **(1974)**, pp.33-46.
[6] C. D. Paice, "Another Stemmer", ACM SIGIR Forum, **(1990)**, pp.56-61.
[7] S. Wu, Q. Qian, Tiejun, D. Li, J. Li and N. Hong, "Comparative Analysis of Methods and Tools for Word Stemming", Library and Information Service, vol. 56, no.15, **(2012)**, pp.109-115.
[8] R. Krovetz, "Viewing morphology as an inference process", Artificial Intelligence, vol.118, no.99, **(1993)**, pp.277-294.
[9] M. A. Hafer and S. F. Weiss, "Word segmentation by letter successor varieties", Information Storage and Retrieval, **(1974)**, pp.371–385.
[10] J. Xu and W. B. Croft, "Corpus-based stemming using co-occurrence of word", ACM Transactions on Information Systems Journal, vol.16, no.1, **(1998)**, pp. 61-81.
[11] J. H. Paik, D. Pal and S. K. Parui, "A novel corpus-based stemming algorithm using co-occurrence statistics", Ma Weiying, NieJianyun, Baeza-Yates R A. Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval, New York: ACM, **(2011)**, pp.863-872.
[12] G. Adamson and J. Boreham, "The Use of an Association Measure Based on Character Structure to Identify Semantically Related Pairs of Words and Document Titles", Information Storage and Retrieval, no.10, **(1974)**, pp.253-60.
[13] J. Mayfield and P. Mc Namee, "Single n-gram stemming", Efthimia-dis E N, Dumais S T, Hawking D, *et al*., Proceedings of the 26thAnnual International ACM SIGIR Conference on Research and Development in Information Retrieval. New York: ACM, (**2003**), pp. 415-416.
[14] L. E. Baum and T. Petrie, "Statistical Inference for Probabilistic Functions of Finite State Markov Chains", Annals of Mathematical Statistics, vol.37, no.6, **(1966)**, pp.1554-1563.
[15] M. Melucci and N. Orio, "A novel method for stemmer generation based on hidden Markov models", Kraft D, Frieder O, Hammer J, Proceedings of the Twelfth International Conference on Information and Knowledge Management. New York: ACM, **(2003)**, pp. 131-138.
[16] A.J. Viterbi, "Error bounds for convolutional codes and an asymptotically optimal decoding algorithm", IEEE Trans on Inf Proc, no.13, **(1967)**, pp.260-269.
[17] V. M. Orengo and C. Huyck "A stemming algorithm for the Portuguese language", String Processing and Information Retrieval, 2001. SPIRE 2001. Proceedings of Eighth International Symposium on. IEEE, **(2001)**, pp.186-193.
[18] E. T. Al-Shammari, "Towards an Error-Free Stemming", International Association for Development of the Information Society, (**2008**), pp.160-163.
[19] Weiss D. Stempelator, "A Hybrid Stemmer for the Polish Language", Institute of Computing Science, **(2005)**.
[20] C. Sitaula, "A Hybrid Algorithm for Stemming of Nepali Text", Intelligent Information Management, no.5, **(2013)**, pp.136-139.
[21] K. Suba, D. Jiandani and P. Bhattacharyya, "Hybrid Inflectional Stemmer and Rule-based Derivational Stemmer for Gujarati", In proceedings of the 2nd Workshop on South and Southeast Asian Natural Language Processing (WSSANLP), IJCNLP 2011, Chiang Mai, Thailand, **(2011)**, pp.1-8.
[22] M. Hadni, S. A. Ouatik and A Lachkar. "Effective Arabic Stemmer Based Hybrid Approach for Arabic Text Categorization", International Journal of Data Mining & Knowledge Management Proc, vol.3, no.4, **(2013)**, pp.1-14.
[23] M. Duoqian and W. Zhihua, "Principles and Applications of Chinese Text Information Processing", Tsinghua University, **(2007)**.
[24] M. Sun, Z. Zuo and C. Huang, "An Experimental Study on Dictionary Mechanism for Chinese Word Segmentation", Journal of Chinese Information Processing, vol.14, no.1, **(2004)**, pp.1-6.
[25] G. Chen, Y. Wang, K. Han and G. Wang, "AN IMPROVED FAST ALGORITHM FOR CHINESE WORD SEGMENTATION", Journal of Computer Research & Development, vol. 37. no.4, **(2004)**, pp.418-424.
[26] K. He, X. Hui and S. Bo, "Design Principle of Expert System for Automatic Word Segmentation in Written Chinese", Journal of Chinese Information Processing, vol. 2, no.5, **(1991)**, pp.1-14.
[27] Y. Lin, Y. Li, S. Tiao and F. Yin, "Research on Neural Network Technology in Chinese Word Separation", Journal of Hunan University, vol.24, no.6, **(1997)**, pp.95-101.
[28] Y. Feng, "Design and Analysis of Chinese Automatic Segmenting System Based on Neural Network", Journal of The China Society for Scientific and Technical Information, vol.17, no.1, **(1998)**, pp.41-50.
[29] S. Maosong, H. Changning, B. K. Tsou, L. Fang and S. Dayang, "Using Character Bigram For Ambiguity Resolution In Chinese Word Segmentation", Computer Research & Development, vol.35, no.5, **(1997)**, pp.332-339.

[30] S. Xiao and H. Degen, "Chinese Word Segmentation Using Minimal Cost Path Algorithm Based on Dynamic Programming", MINI-MICRO SYSTEMS, vol.27, no.3, **(2006)**, pp.516-519.
[31] F. Zhai, F. He and W. Zuo, "Chinese Word Segmentation Based on Dictionary and Statistics", Journal of Chinese Computer Systems, vol.44, no.10, **(2006)**, pp.144-146.
[32] B. Shuanhu, "An integrated approach of Chinese word segmentation and POS tagging", Chinese Information, no.2, **(1996)**.
[33] SCWS, http://www.xunsearch.com/scws/, **(2013)**.
[34] ICTCLAS Chinese word segmentation system, http://ictclas.org/, **(2010)**.
[35] Hylanda Chinese Intelligent Word Segmentation White Paper, http://wenku.baidu.com.
[36] M. Saimaiti and Z. Feng, "A Syllabification Algorithm and Syllable Statistics of Written Uyghur", Proceedings of the Corpus Linguistics Conference, CL2007.
[37] B. Aisha and M. Sun, "A statistical method for Uyghur tokenization", Natural Language Processing and Knowledge Engineering, NLP-KE 2009. International Conference on. IEEE, **(2009)**, pp.1-5.
[38] M. Ablimit, G. Neubig, M. Mimura, "Uyghur Morpheme-based Language Models and ASR", Ipsj Sig Notes, no.17, (2010), pp.581-584.
[39] Abdurahim Mahmoud, Akbar Pattar and Askar 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.
[40] K. Mahmutjan, Reyisi, A. Litip and Y. Abaydulla, "Modern Uyghur Language", First Edition. Xinjiang People's Press, **(2003)**, (in Uyghur).
[41] "Modern Uyghur Particular Dictionary", Xinjiang People's Press, **(1999)**,(in Uyghur).
[42] G. Adongbieke and M. Ablimit, "Research on Uighur Word Segmentation", Journal of Chinese Information Processing, vol.18, no.6, **(2004)**, pp.61-65.
[43] M. Ablimit, M. Eli and T. Kawahara, "Partly supervised Uyghur morpheme segmentation", In Proc. Oriental-COCOSDA Workshop, **(2008)**, pp.71—76.
[44] C. Peng, "Uyghur Stem Segmentation and POS Tagging based on Corpora", Xinjiang University, Master Thesis, **(2006)**.
[45] A. Kadir, K. Kadir and T. Ibrahim, "Morphological Analysis of Uighur Noun for Natural Language Information Processing", Journal of Chinese Information Processing, vol. 20, no.3, **(2006)**, pp.43-48.
[46] M. Ablimit, A. Hamdulla and T. Tohti, "Reseacrh on Uyghur Word Analyzer", National Language Information Research Technology, Eleventh National Symposium on National Language Information, **(2007)**.
[47] A. Xerip, "On the Complex Feature of Verbal Suffixes in Uyghur Language", Journal of Chinese Information Processing, vol.22, no.3, **(2008)**, pp.105-109.
[48] A. Wumaier, T. Yibulayin and Z. Kadier, "Study of Uyghur Noun Stemming Algorithm", the Fourth National Information Retrieval and Content Security Conference, **(2008)**.
[49] A. Wumaier, T.Yibulayin, Z. Kadeer and S. Tian, "Conditional Random Fields Combined FSM Stemming Method for Uyghur", Computer Science and Information Technology, International Conference on. IEEE, **(2009)**, pp.295-299.
[50] Azragul, X. Qi and A. Yusup, "Website Phrasal Survey Based Modern Uighur Stem Extraction and Application Study", Computer Applications and Software, vol.29, no.3, **(2012)**, pp.32-34.
[51] Z. Kadeer, A. Wumaier, T. Yibulayin, P. Tursun and Wu X., "Uyghur NounStemming System Based on Hybrid Method", Computer Engineering and Application, vol.49, no.1, **(2013)**, pp171-175.
[52] A. Tayier, W. Tuergen, Y. Aishan and J. Zhang, "Uyghur Verb Stemming Method Based on a Tagged Dictionary and Rules", Journal of Xinjiang University(Natural Science Edition), no.1, **(2013)**.
[53] A. Xerip, M. Eli and T. Ebrayim, "Formalized Description of Verbal Morphological Rules of Uyghur Language", Journal of Minzu University of China, vol.40, no.3, **(2013)**, pp.117-123.
[54] E. Sediyegvl, L. Xiang, C. Zong, P. Akbar and H. Askar, "A Multi-strategy Approach to Uygur Stemming", Journal of Chinese Information Processing, vol.29, no.05, **(2015)**, pp.204-210.

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 is natural language processing.

Abdusalam Dawut, he received M.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 May 2011. His research interests include Education technicalization and imformatization.

Palidan Tuerxun, she 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, 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 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.