Detection of Loan Words in Uyghur Texts

Chenggang Mi[1,2], Yating Yang[1], Lei Wang[1], Xiao Li[1], and Kamali Dalielihan[1]

[1] Xinjiang Technical Institute of Physics & Chemistry of Chinese Academy of Sciences,
Urumqi, Xinjiang 830011, China
[2] University of Chinese Academy of Sciences, Beijing, 100049, China
michenggang@gmail.com,
{yangyt,wanglei,xiaoli}@ms.xjb.ac.cn,
kamaly330@gmail.com

Abstract. For low-resource languages like Uyghur, data sparseness is always a serious problem in related information processing, especially in some tasks based on parallel texts. To enrich bilingual resources, we detect Chinese and Russian loan words from Uyghur texts according to phonetic similarities between a loan word and its corresponding donor language word. In this paper, we propose a novel approach based on perceptron model to discover loan words from Uyghur texts, which consider the detection of loan words in Uyghur as a classification procedure. The experimental results show that our method is capable of detecting the Chinese and Russian loan words in Uyghur Texts effectively.

Keywords: loan words detection, phonetic similarity, Uyghur, perceptron-based model.

1 Introduction

Statistical methods are commonly used in recent NLP tasks [1], which rely on corpora heavily. For tasks like SMT (Statistical Machine Translation), there always exist data sparseness during training of translation models because lack of bilingual texts [2]. This situation may get worse in under-resources languages' (like Uyghur) processing.

We find that there are many loan words in Uyghur, which are mainly borrowed from Chinese and Russian (Table 1), and a loan word always pronounces similarly with its corresponding donor language word. This might be an interesting clue to discover loan words in Uyghur texts. And further enrich Uyghur related bilingual resources.

To detection loan words in Uyghur, we consider it as a string similarity problem and transform phonetic similarity into string similarity firstly. Intuition might suggest that common used string similarity algorithms can solve this problem easily. However, spelling of loan words may change when borrowed from the donor language, also, the characters asymmetrical transformation affect the performance of detection model. Additionally, Uyghur words are forming as adding suffixes after a certain word stem, how to deal with these suffixes properly when compute string similarity also should be considered carefully. In this paper, we suggest a novel method to detect Chinese and Russian loan words in Uyghur texts, which can be described as following two parts: Characters alignment and Detection of Chinese and Russian loan words in Uyghur texts.

C. Zong et al. (Eds.): NLPCC 2014, CCIS 496, pp. 103–112, 2014.

The rest of this paper is organized as follows. Section 2 describes previous work on loan words research. In section 3, we give an overview of loan words in Uyghur. The character alignment model and loan words detection approach are presented in section 4. Section 5 illustrates the corpus and gives the experimental results. Conclusions and future work are summarized in section 6.

2 Related Work

Previous works on loan words are mainly focused by linguists. [3] looks specifically at the language contact of English and Chinese and details the resultant language change when words from International English are borrowed into standard Chinese; [4] outlines the historical and cultural contexts of borrowing from English into Japanese, processes of nativization, and functions served by English loan words; [5] studied Chinese loan words in English; [6] concerned about loan words in English and Chinese, and the characteristics of their language contact. For loan words in Uyghur, [7] compared different methods that words borrowed by Uyghur and Chinese; [8] analyzed influence to Uyghur words made by loan words; [9] explained the historical process of the Chinese words borrowed into the Uyghur language, the characteristics of the Chinese borrowed words, their development and certain troubles of them in the course practical usage. In NLP field, some related works also proposed by researchers. [10] presented a string similarity based method to discover Chinese loan words in Uyghur, which combine two basic string similarity algorithms as a recognize model.

In this paper, we propose a novel method to detect Chinese and Russian loan words from Uyghur texts, which extend previous work, and consider the detection as a binary classification based on perceptron model. For minimum differences between donor languages (Chinese and Russian) and Uyghur, we obtain character mapping rules by characters aligning; features used during the perceptron model' s training are taken from five string similarity algorithms.

3 Loan Words in Uyghur

3.1 Introduction of Loan Words

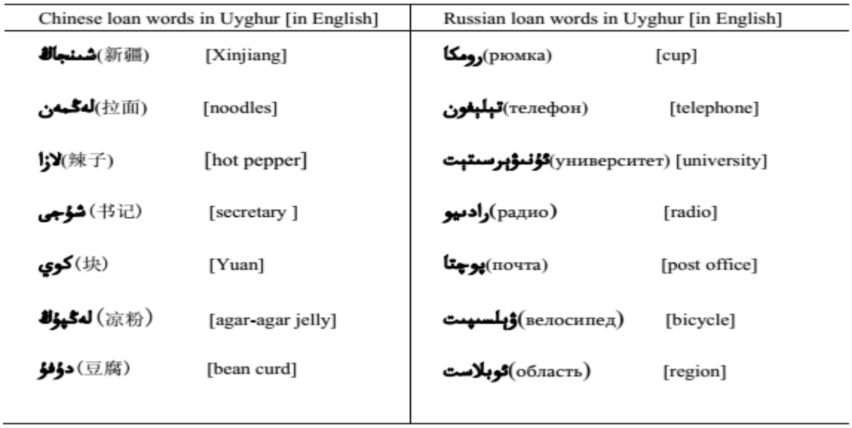
A loanword is a word borrowed from a donor language and incorporated into a recipient language directly, without translation. Loan words may have several changes when loaned: 1) Changes in meaning; 2) Changes in spelling; 3) Changes in pronunciation.

3.2 Loan Words in Uyghur

Uyghur is an official language of the Xinjiang Uyghur Autonomous Region, In addition to influence of other Turkic languages, Uyghur has historically been influenced strongly by Persian and Arabic, and more recently by Mandarin Chinese and Russian.

Except named entity words like names of person and locations, there are also many regular terms borrowed from Chinese and Russian. We give some examples of loan words in Table 1:

Table 1. Examples of loan words in Uyghur



Chinese loan words in Uyghur [in English]	Russian loan words in Uyghur [in English]
شىنجاڭ(新疆) [Xinjiang]	رومكا(рюмка) [cup]
لەڭمەن(拉面) [noodles]	تېلېفون(телефон) [telephone]
لازا(辣子) [hot pepper]	ئۇنىۋېرسىتېت(университет) [university]
شۇجى (书记) [secretary]	رادىيو(радио) [radio]
كوي(块) [Yuan]	پوچتا(почта) [post office]
لەڭپۇڭ (凉粉) [agar-agar jelly]	ۋېلسىپېت(велосипед) [bicycle]
دۇفۇ(豆腐) [bean curd]	ئوبلاست(область) [region]

3.3 Challenges in Loan Words Detection

Challenge One: Spelling change when borrowed from donor languages (Chinese and Russian). The word of Uyghur and Russian can be writing as Latin alphabet, the Chinese word can be presented by Pinyin, for example:

Russian loan words in Uyghur: “رادىيو” (“radyo”) - “радио ” (“radio”)

Chinese loan words in Uyghur: “كوي” (“koi”) - “块” (“kuai”)

Changes of spelling have a great impact on the loan words detection task.

Challenge Two: Suffixes of Uyghur words effect the detection of loan words.
A Uyghur word is composed of a word stem and several suffixes, which can be formally described as:

$$Word = stem + suffix0 + suffix1 + \cdots + suffixN \tag{1}$$

If we use the Edit Distance to measure the string similarity between a word and its original form, the length of the word’s suffixes equal even greater than the original word’ length. Even though the word is a loan word actually, traditional similarity algorithms cannot give a sure result.

For overcome above two challenges, we propose a loan words detection approach, which can be divided into two steps: 1) Characters Alignment (in section 4.1, for overcome Challenge One); 2) Classification-based loan words detection model (in section 4.2, for overcome Challenge Two).

4 Recognition of Loan Words from Uyghur Texts

In this paper, we propose a novel method to discover loan words from Uyghur texts, which combines several string similarity algorithms as feature functions of a perceptron-based model, and consider the loan words detection in Uyghur texts as a binary classification problem. Rather than transforming Uyghur characters according to traditional rules, associations between Uyghur characters and donor language (Chinese(pinyin) and Russian) characters are obtained by aligning exist words (denote as character sequences).

4.1 Characters Alignment

To obtain characters mapping rules, we consider it as a word alignment problem, which take donor language characters as source language words, recipient language characters as target language words. The IBM Model 1 [11] and HMM [12] are used here as word alignment models, parameters of these models are estimated by EM (Expectation Maximum) algorithm [13].

Two alignment models can be formally described as follows:

IBM Model 1

$$\mathrm{p(e,a|f)} = \frac{\varepsilon}{(l_f+1)^{l_e}}\prod_{j=1}^{l_e} t(e_j|f_{a(j)})\,, 1 \le j \le l_e\,. \tag{2}$$

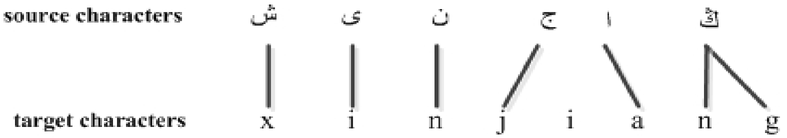
l_f and l_e are the length of donor language characters (Chinese or Russian) and Uyghur characters; a is the word alignment function, $\mathrm{a{:}\,j \rightarrow i}$ means source word f_i is align with target word e_j; $\mathrm{t(e|f)}$ is the translation probability of source word f and target word e. When training the IBM model 1, $\mathrm{t(e|f)}$ can be computed based on word co-occurrence counts:

HMM

$$\mathrm{p(f|e)} = \sum_{a_1^m}\prod_{j=1}^{m}[p(a_j|a_{j-1}, l)\cdot p(f_j|e_{a_j})] \tag{3}$$

Here, alignment probabilities are independent of word position and depend on jump width $(a_j - a_{j-1})$.

Suppose we have one pair of Uyghur-Chinese words, شىنجاڭand "Xinjiang" (which is the Pinyin of Chinese word "新疆". For convenience, in this paper, we denote Chinese words with Pinyin), the associations of characters can be listed as follows:



Additionally, for characters align to null, we assign them target characters according to transform rules (Latin rules for Uyghur).

4.2 Classification-Based Loan Words Detection Model

Features of String Similarity Algorithms

In this part, we investigate features of five string similarity algorithms, and extend some of them (Edit Distance and Common String) to adapt the tasks that detect loan words in Uyghur texts.

Position-Related Edit Distance

Because distinctions of words forming between Uyghur and donor languages, we cannot use the Edit Distance to measure the similarity between a donor language word (Chinese, Russian) and a Uyghur word, directly. Suppose we have a Chinese word "兰州" ("lanzhou") and a Uyghur word "لەنجۇلاردا", due to the suffixes of the Uyghur word, the number of deletions according to the Edit Distance algorithm equal (this example) even greater than the length of Chinese word ("lanzhou"), so the Uyghur word cannot be recognized as a Chinese loan word (which actually is).

Intuition might suggest that the stemming of a Uyghur word firstly can avoid such problems, however, this approach depend on performance of the Uyghur stemmer heavily. In this part, we propose a position-related edit distance (PRED) method, which tracing the procedure of edit distance computing, if continues deletion occurred at the end of words, these deletion number will be subtracted from the edit distance results. Experimental results show that our method (PRED) outperform the stem-based method and the basic edit distance algorithm.

$$PRED_{a,b}(i,j) = \begin{cases} ED_{a,b}(i,j)\ No\ Continue\ Delete\ Occurred, \\ ED_{a,b}(i,j) - times_{delete}(a,b) \quad Otherwise. \end{cases} \tag{4}$$

$times_{delete}(a,b)$ is the times continue deletion occurred at end of words.

Dice Coefficient

The coefficient may be calculated for two strings, ***a*** and ***b*** using bigrams as follows:

$$DC_{a,b} = \frac{2n_t}{n_a + n_b} \tag{5}$$

Where $n_t$is the number of character bigrams is found in both strings,n_a is the number of bigrams in string a and n_b is the number of bigrams in string b.

Weighted Common Subsequence

Rather than using the Longest Common Subsequence, we present a Weighted Common Subsequence (WCS) to measure similarity between two words, which assign a weight to each common subsequence according to its length. Finally, we sum up these results as WCS of two words (a and b).

$$WCS_{a,b} = \sum_{i=2}^{min(La,Lb)} LEN_i \cdot NUM_i \tag{6}$$

La and Lb are length of word a and word b, respectively. LEN_i is the length of the ith common string, NUM_i is the number of these common strings appeared.

*Jaccard Similarity Coefficien*t and *Overlap Similarity* ($JSC_{a,b}$, $OLS_{a,b}$) are also used as basic features.

Accordingly, Jaccard Similarity Coefficient, Overlap Similarity and Dice Coefficient are mainly focus on discrete string similarity; Position-Related Edit Distance can measure global similarity of two strings, which also overcome the shortage of basic Edit Distance in loan words detection task. The Weighted Common String algorithm measures string similarity of two words according to number of common strings and length of common strings. For detect loan words in Uyghur texts effectively, we consider these five string similarity algorithms as five feature functions of a binary classification model.

Perceptron-Based Loan Words Detection
The perceptron [14, 15, 16] is an algorithm for learning a binary classifier: a function that maps its input **x** (which is a real-value vector) to an output value $f(x)$ (which is a single binary value)

$$\mathrm{f(x)} = \begin{cases} 1 & if\ w \cdot x + b > 0, \\ 0 & otherwise. \end{cases} \tag{7}$$

Where w is a vector of real-valued weights, **w • x** is the dot product (which here computes a weighted sum), and b is the “bias”, which is a constant term that does not depend on any input value. The value of $f(x)$ (0 or 1) is used to classify x as either a positive or a negative instance, in the case of a binary classification problem. If b is negative, then the weighted combination of inputs must produce a positive value greater than $|\mathrm{b}|$in order to push the classifier neuron over the 0 threshold. Spatially, the bias alters the position of the decision boundary. The perceptron learning algorithm does not terminate if the learning set is not linearly separable. If the vectors are not linearly separable will never reach a point where all vectors are classified properly.

In this paper, we consider detection of loan words in Uyghur texts as a perceptron-based classification problem. Here, x is a real-value vector which contains string similarities of two words, and compute by algorithms described in 4.2. The output of this model is a loan word label (0: no, 1: yes).

The vector x used in the loan words detection task can be formally described as follows:

$$< PRED_{a,b}, DC_{a,b}, WCS_{a,b}, JSC_{a,b}, OLS_{a,b} > \tag{8}$$

$PRED_{a,b}, DC_{a,b}, WCS_{a,b}, JSC_{a,b}$ and $OLS_{a,b}$ are string similarity scores of two words (a donor language word and a Uyghur word). The computation methods of these scores are presented in section 4.2.

5 Experiments

In this section, we evaluate our method by detect Chinese and Russian loan words in Uyghur texts.

5.1 Set Up

Character transformation rules are obtained by GIZA++[1], which is widely used in word alignment, and implemented IBM models and HMM. We implement five string similarity algorithms, respectively. The classification model we use here is **XPerceptron**, which is a C++ implementation of the perceptron model.

Results of loan words detection are evaluated by R (Recall), P (Precision) and F1 (F-measure), respectively. A indicates a set of loan words output by our method; B is a set of words also output by our method but none of them is loan word and C is a set of loan words included in test set but did not output. Therefore, we can compute F1 as:

$$\mathrm{R} = \frac{A}{A+C}, \qquad P = \frac{A}{A+B}, \qquad F1 = \frac{2*P*R}{P+R} \tag{9}$$

5.2 Introduction of Corpora

In this paper, we use a Uyghur-Chinese city names mapping table and Uyghur-Russian city names table to train the transformation rules (as described in Section 4.1). The test sets of loan words detection are selected from web, which include several domains, such as government documents, news, daily life, etc.

5.3 Experiments

We calculate similarity features of donor language words and Uyghur words according to five string similarity algorithms. Then, features of each word pair and its loan word label will be considered as an input of perceptron-based detection model. For validate the effectiveness of our method, we also conduct experiments on stem-based and traditional transformation rules-based approaches. Results of these experiments are shown in Table2, Table 3, Table 4 and Table 5, respectively.

5.4 Results and Analysis

Table 2 performs results of five basic string similarity algorithms (ED (Edit Distance), DC (Dice Coefficient), CS (Common String), OS (Overlap Similarity) and JSC (Jaccard Similarity Coefficient)), which are based on words. Experiments show that ED and CS outperform other three algorithms, that because DC, OS and JSC mainly focus on discrete characters, ED and CS concern much on characters that continuous.

Results of stem-based methods (Table 3) are slightly better than word-based (in Table 2), one possible reason is that Uyghur stemmer can overcome some situation that caused by suffixes when computing string similarity.

[1] `https://code.google.com/p/giza-pp/`

Table 2. Results of five basic algorithms (word-based)

	Chinese Loan Words			Russian Loan Words		
	R	P	F1	R	P	F1
ED	71.20	62.41	66.52	73.29	67.32	70.18
DC	69.22	60.98	64.84	70.02	62.41	66.00
CS	73.12	61.16	66.61	76.08	68.43	72.05
OS	69.25	60.73	64.71	70.29	63.78	66.88
JSC	69.10	61.98	65.35	71.81	64.59	68.01

Table 3. Results of five basic algorithms (stem-based)

	Chinese Loan Words			Russian Loan Words		
	R	P	F1	R	P	F1
ED	71.38	62.71	66.76	74.32	68.50	71.29
DC	69.30	61.92	65.40	71.43	63.09	67.00
CS	73.15	63.20	67.81	76.13	69.27	72.54
OS	70.02	60.94	65.17	70.82	64.50	67.51
JSC	69.83	62.51	65.97	72.61	65.08	68.64

Table 4. Results of five algorithms (two improved, word-based)

	Chinese Loan Words			Russian Loan Words		
	R	P	F1	R	P	F1
PRED	75.72	64.73	69.80	75.39	70.02	72.61
DC	69.78	62.33	66.35	71.64	63.25	67.18
WCS	74.39	64.36	69.01	78.01	72.34	75.07
OS	71.29	61.72	66.16	71.05	65.20	68.00
JSC	71.32	63.65	67.27	72.89	65.37	68.92

Table 5. Result of perceptron-based detection model

	Chinese Loan Words			Russian Loan Words		
	R	P	F1	R	P	F1
PBDM	78.82	68.30	73.18	81.03	73.22	76.93

In Table 4, ED and CS in five basic string similarity algorithms are improved as PRED (Position-Related Edit Distance) and WCS (Weighted Common String), respectively. Although word-based, performances of PRED and WCS outperform ED and CS both in Table 2 and Table 3, significantly. Besides, results of Table 4 are based on transformation rules obtained by character aligning, which also contribute to the performance of string similarity algorithms.

Table 5 performs results of perceptron-based detection model (PBDM), which is the integration of our method. The PBDM combines five string similarity algorithms (PRED (Position-Related Edit Distance), DC (Dice Coefficient), WCS (Weighted Common String), OS (Overlap Similarity) and JSC (Jaccard Similarity Coefficient)) as five features, and loan word labels as outputs, according to features of the perceptron. In our experiments, PBDM achieved the best performance. The most important reason is that, perceptron-based model integrate advantages of five string similarity algorithms, and the error-driven model much adaptive to our task. Interestingly, the performance of Russian loan words detection is outperforming Chinese loan words detection in all experiments, which may because the spelling method of Russian loan words is much closer with Uyghur.

6 Conclusion and Future Work

To detection loan words in Uyghur texts effectively, we transfer the phonetic similarity between donor language (Chinese and Russian in our paper) words and Uyghur words to strings similarity, and consider the detecting procedure as a classification problem. Experimental results show that with our method Chinese and Russian loan words can be recognized efficiently. In future work, we will focus on extraction of bilingual resources based on loan words, and extend this approach to other languages.

Acknowledgements. This work is supported by the Strategic Priority Research Program of the Chinese Academy of Sciences (Grant No.XDA06030400), West Light Foundation of Chinese Academy of Sciences (Grant No.LHXZ201301 XBBS201216), the Xinjiang High-Tech Industrialization Project (Grant No. 201412101) and Young Creative Sci-Tech Talents Cultivation Project of Xinjiang Uyghur Autonomous Region (Grant No. 2013731021).

References

1. Chris, M., Hinrich, S.: Foundations of Statistical Natural Language Processing. MIT Press, Cambridge (1999)
2. Chung, C., Ho, C., Ping, C.: Using Sublexical Translations to Handle the OOV Problem in Machine Translation. ACM Transactions on Asian Language Information Processing 10(3), 1–20 (2011)
3. Lauren, A.H.L.: English Loanwords in Mandarin Chinese. The University of Arizona, Arizona (2002)
4. Gillian, K.: English loanwords in Japanese. World Englishes 14(1), 67–76 (1995)
5. Kui, Z.: On Chinese-English Language Contact through Loanwords. English Language and Literature Studies 1(2), 100–105 (2011)
6. Xuan, L., Lanqin, Z.: On Chinese Loanwords in English. Theory and Practice in Language Studies 1(12), 1816–1819 (2011)
7. Yan, C., Ping, C.: A Comparison on the methods of Uyghur and Chinese Loan Words. Journal of Kashgar Teachers College 32(2), 51–55 (2011)
8. Yan, Z.: Influence of Loan Words on the Words of Uygur Language. Journal of Hubei University of Education 28(1), 37–39 (2011)

9. Shiming, C.: New Research on Chinese Loan Words in the Uygur Language. N.W.Journal of Ethnology 28(1), 176–180 (2011)
10. Mi, C., Yang, Y., Zhou, X., Li, X., Yang, M.: Recognition of Chinese Loan Words in Uyghur Based on String Similarity. Journal of Chinese Information Processing 27(5), 173–178 (2013)
11. Brown, P.E., Pietra, S.A.D., Pietra, V.J.D., Mercer, R.L.: The mathematics of statistical machine translation: Parameter estimation. Computational Linguistics 19(2), 263–311 (1993)
12. Vogel, S., Ney, H., Tillmann, C.: Hmm-based word alignment in statistical translation. In: Proceedings of the 16th Conference on Computational Linguistics, pp. 836–841. Association for Computational Linguistics (1996)
13. Dempster, A., Laird, N., Rubin, D.B.: Maximum-likelihood from incomplete data via the EM algorithm. Journal of the Royal Statistical Society. Series B (Methodological) (39), 1–38 (1977)
14. Gallant, S.I.: Perceptron-based learning algorithms. IEEE Transactions on Neural Networks 1(2), 179–191 (1990)
15. Collins, M.: Discriminative training methods for hidden markov models: Theory and experiments with perceptron algorithms. In: Proceedings of the ACL 2002 Conference on Empirical Methods in Natural Language Processing, vol. 10. Association for Computational Linguistics (2002)
16. Dasgupta, S., Kalai, A.T., Monteleoni, C.: Analysis of perceptron-based active learning. Learning Theory, 249–263 (2005)