Investigation and use of methods for defining the extends of similarity of Kazakh language sentences

Unzila Kamanur, Altynbek Sharipbay,Gulila Altnbek, Gulmira Bekmanova, Lena Zhetkenbay

L.N.Gumilyov Eurasian National University, Astana
unzila.88@mail.ru, sharalt@mail.ru, glaxd2014@163.com gulmira-r@yandex.ru, jetlen_7@mail.ru

Abstract. Finding similarity degree is one of the significant technologies used in the sample-based machine translation. It works in the following principle, first matching the input sentences with a sentence in the sample database, after that it is necessary to pick up parts of the similar sentences for the sentence which is aimed to translate; it is finished by correcting the structure or paraphrasing it with a relevant meaning. For that reason, the degree of similarity of two samples highly affects on the results of translation. Thus, there are dependence between quality of the outputs and the similarity degree.

Keywords: EBMT, Synonym replacement, Kazakh sentence, similarity sentence similarity, machine translation, natural language processing.

1 Introduction

The translation from one language to another is considered that it was introduced as a separate science for the first time in 1947in the letter written by Warren Wiver for Nobert Viner [1]. After that, it can be seen that in the last 50 years the quick development of this study has occurred. Nowadays, a number of machine translation tools became available for users, various types of machine translation systems are widely used all over the world. Their further development and the extent of use highly depend on the presence of natural language corpus and the degree of difficulty of formalizing the natural languages.

In 1980s a Japanese researcher Makoto Nagao introduced a new method of translation. In the work published by him in 1984 "A framework of a mechanical translation between Japanese and English by analogy principle", he showed the translation of simple sentences done without any grammatical analysis. Instead of undertaking an analysis, he divided the whole sentence into small fragments(sentences). Only after dividing them, he translated the sentence. As a result, he was able to create a whole sentence from the short fragments. By taking the samples and matching them by similarity degree he managed to find a new approach on language translation. This method was called Example based Machine Translation [2].

The method of defining the similarity degree in sentences is widely used to make the natural languages as a separate science. For instance, there are a number of available functions in the systems of question-answer such as, to find answers on the ques-

adfa, p. 1, 2011.

tions given by an user, to match this question with questions in a database by finding a similarity, to filter the unmatched phrases from the user's answers by using the information filter technology and etc.

2 Related Work

All the methods used to find solutions for the artificial intelligence issues contain the steps of matching the knowledge fragments(requested fragments) with a prepared database of samples. This process, generally, include the process of comparing two fragments and the process of filtering in order to make a comparison between them. The process of comparing could be divided to the following methods of comparison:

- Syntactic method;
- Parametrical method;
- Semantical method ;

During undertaking syntactical analysis also known as parsing the full balance and structure of two data are used to describe the fragment are compared. The example of these methods can be the unification of predicates in the Prolog language, where to take balance first of all the predicate's names are compared, after that the arguments pairs are compared with each other.

Parametrical analysis checks the structures of data which is used to describes the fragment for its partly balance(not full).

Semantic analysis is used to check the semantic(meaning) of knowledge fragment. During this process, usually it is necessary to study a structure of the known fragment. Also, in this process the term called "semantic similarity of words" is widely used, i.e. the idea of distance between terms based on their likeness by the meaning or semantic content and it is estimated by their syntactical representation. In some cases, the semantic similarity can be calculated directly. For instance, to compare two linguistic variables which are described in the same metric scales. The semantic algorithm haven't been developed yet which compares by universal and mathematic methods.

In addition, another method which is worth to mention in this paper is the search of relevant data in neuron systems. They are also known as associative search. According to the comparisons shown above, in associative search the known data is described not in symbols but by digits and signals used in neuron conditions.

The determination of similarity degrees of texts started from the early periods in 1963 when Gerard Salten founded the *Vector Space Model* (VSM) [3]. This method is still considered as the most popular and the best developed among existing ones. At the beginning it was used to define similarity degree in documents, but later it started to be used for texts. The main idea behind the Vector Space Model is by comparing the deviation of angles between each text vector and the original query(requested input text) vector where input is represented as the same kind of vector as the texts.

3 Defining the similarity degrees of sentences written in Kazakh language

Defining the similarity degrees of sentences written in Kazakh language is basically based on Vector Space model. It includes two modules: quick search module and similarity calculation module. In the translation process first quick search module searches the samples from database which suits more than others, after that this set of matches are send to the second module for defining the similarity degree. In the similarity calculation module, both modules are used at the same time to define the similarity based on combined vectors calculated from the similarity.

3.1 The quick search module

The significant issues in the sample based machine translation are grouping into a set the samples from database corresponding to the input and finding the sample most similar to the input . The quick search module is designed to solve these problems.

In order to improve the search speed it is necessary to divide all the words in the database and create an inversed index.To perform it, first of all samples in the database should be included in a table, then each sample formed from the word has to be assigned an ID and another table is created for that. In this case ID is the number of words formed from the samples. This formation is shown on Figure 1 below

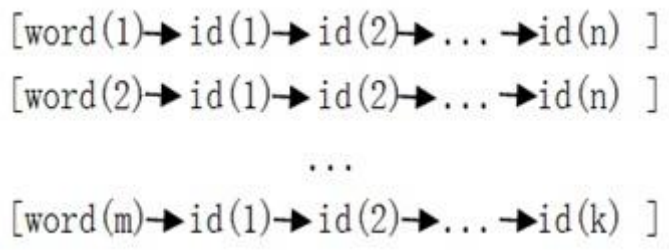

Fig. 1. The scheme of a inversed index based search

The quick search process can be described as following:
Step 1: It uses the separate word's list, where Word is a word itself, id is the allocated special number given for each word.
Step 2.Analyzing the input sentence and produce the table of linked words. By using statistics from the list of frequently appeared separate words it calculates id set of the input sentence.
Step 3. Going back to the list of samples it calculates the degree of similarity, chooses the most similar sentence and analyzes it as a input sentence

The figure 2 shows the example of 2-step analysis of the sentence «Ақпараттық қауіпсіздік жүйелері» in Kazakh language.

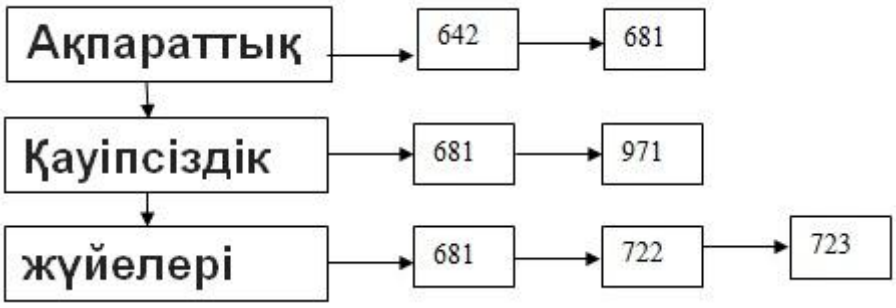

Fig. 2. Analysis of the sentence «Ақпараттық қауіпсіздік жүйелері» in Kazakh language

As the Figure 2 shows 681 the level of linkage and appearing level are the highest among other digits shown in this example. The statistics are done for the each word separately, all id in this example are in ascending order [(681,3),(642,1),(971,1),(722,1),(723,1)]. At the end, it is left the highest frequently id set.

3.2 The similarity calculation module

The calculation of similarity degree for a main word include the following steps:

1. The degree of words' similarity .

Morphological degree of similarity (Word Overlap Measures) is used to calculate the similarity of the two sentences regarding their structure. The calculation of similar words contained there is performed using the following formula:

$$\mathrm{sim}_{\mathrm{Overlap}}(\mathrm{x},\mathrm{y}) = \frac{len(samewc(x,y))}{\max(length(x), length(y))} \quad (1)$$

Where, len(samewc(*x*,*y*)) is the number of matched words from an input sentence*x*and the sample sentence*y*;

$\mathrm{sim}_{\mathrm{overlap}}$ is used to describe a morphological similarity degree;

length(x) is a number of words in the sentence x, whereaslength(y) describes anumber of words in the sentence y(including all the punctuation marks).

Defining the morphological similarity degree is shown in the example below.

Example:

Input sentence: A; ол қалаға қашан келеді?(When will he arrive to the city?)

Sample sentence: B: сабаққа қашан барасың(When will you go to classes)?C: ол ауылдан қашан келеді(When he will arrive from the village)?

Using the following formulae allows to calculate the similarity degree:

$$\mathrm{sim}_{\mathrm{Overlap}}(\mathrm{A},\mathrm{B}) = \frac{len(samewc(A,B))}{\max(length(A),length(B))} = \frac{1}{\max(4,3)} = 0.4$$

$$\mathrm{sim}_{\mathrm{Overlap}}(\mathrm{A},\mathrm{C}) = \frac{len(samewc(A,C))}{\max(length(A),length(C))} = \frac{3}{\max(4,4)} = 0.8$$

$$\mathrm{sim}_{\mathrm{Overlap}}(\mathrm{A},\mathrm{C}) < \mathrm{sim}_{\mathrm{Overlap}}(\mathrm{A},\mathrm{C})$$

In this example we can see that sentence from its structure and context C is better corresponds to the A, whether B.

2. The reverse order based similarity degree calculation algorithm .

During the match of two sentences in some cases, it can be so that their divided units(parts) seem similar, but we cannot make a constant rule from their similarity based on this context.

Thus, if the orders of two divided sentences are changed, it might lead completely different context than its initial meaning. Therefore, the similarity degree of orders of words has to be calculated as well.

In this case, n(n∈N) is a set of various elements, where first of all the rule of ordering is set for each element. For instance, N is a directly ordered numbers and they are put in ascending order, whereas reverse ordered numbers are put in descending order. The overall number of reverse orders in a string is a number of reverse orders of this string.

The word's order describes the linkage similarity of all the units that initial sentence contains. Using these methods assumes that we put all the containing similar units in reverse order which is located next to each other.

Also we need to following labels for two sentences (*x* and *y*):

Ordoccur (*x*,*y*) – a set of units which appears only once in the sentence;

pfir(*x*,*y*) –the number of a vector which describes the position of units in the x sentence within the set of units *ordoccur*(*x*,*y*) ;

psec(*x*,*y*) – the number of a vector which describes the position given by similarity degree of units in the y sentence within the *pfir*(*x*,*y*) vector of units *pfir*(*x*,*y*;

rew(*x*,*y*) – the inverse order number of element sequences in *psec*(*x*,*y*) vector

$similar_{worder}(x,y)$defines the similarity degrees of words in *x* and *y* sentences:[4]

$$similar_{worder}(x,y) = \begin{cases} 1 - \frac{rew(x,y)}{|ordoccur(x,y)|-1}, & |ordoccur(x,y)| > 1 \\ 1, & |ordoccur(x,y)| = 1 \\ 0, & |ordoccur(x,y)| = 0 \end{cases} \quad (2)$$

The similarity degreeof words for the example we used in above will be as following:

ordoccur(A,C)={“ол”,”қашан”,”келеді”,”?”};When will he come?

pfir(A,C)= (1,3,4, 5);

psec(A,C)=(1,3,4,5);

rew(x,y)= 0, since forpsec(A,C) the reverse order number of unit’s sequence is1<3,3<4,4<5.

Next, by using the formulae (2) the degree of words order’s similarity in the A and C sentences is defined as following:

$similar_{worder}(x,y) = 1 - \frac{0}{4-1} = 1.$

3. Defining the similarity degree of sentence length .

It is important to clarify the morphological similarity when the similarity degree is defined based on the similarity degree of sentence’s length. The length similarity of both sample sentences in database and input sentence affects on the whole sentence similarity degree. The similarity degree of sentence length is calculated using the formula below:

$$similar_{length}(x,y) = 1 - \frac{length(x) - length(y)}{length(x) + length(y)} \quad (3)$$

In order to define the similarity of words of an input sentence x and sample sentence y the following formula is used:

$$similar_{wstn} = \alpha \cdot similar_{overlap} + \beta \cdot similar_{worder} + \gamma \cdot similar_{length}, \quad (4)$$

reα,β, γ–experimental values.

3.3 The Vector Space Model based TF_IDF similarity degree calculation method.

TF_IDF is a statistical measure used to assess the importance of words in the context of being a part of a document or corpus.TF-IDF measure are widely used in text analysys and information search purposes. For instance, when the request comes it is used to match the relevancy of a document and during the cauterization it measures the extend of suitability. This idea was introduced by Karen Spark Jones. It contains two parts:

1. Term Frequency – the ratio of the total number of terms(words) in a document to the number of input words. Thus, by the document we define the importance of a word t_k:

$$TF(t_k, d) = \frac{n_k}{\sum_i n_i}, \quad (4)$$

where n_k– a total number of a terms t_k in a $d$document, and $n_i$is a total number of all words containing in the document.

2. *Inverse Document Frequency*– inverse frequency of word occurance in a document collection. IDF calculation diminishes the weight of terms that occur very frequently (such as articles in English) in the document set and increases the weight of terms that occur rarely.In the collection of documents there is only one IDF value for one separate term.

$$\mathrm{IDF}(t, D) = log\frac{|D|}{|(d_k \supset t_k)|}, \qquad (5)$$

where$|D|$–the number of documents in the corpus
$|(d_k \supset t_k)|$ is –number of documents where the term t appears when $n_k \neq 0$
It is not so important to calculate the base of the logarithm in the formula, because by increasing the base leads to increasing the weight of terms.
Thus, TF-IDF is a measure that comes from multiplying two separate measures:
$\mathrm{TF\_IDF}(t, d, D) = TF(t, d) \cdot IDF(t, D)$.
During using the TF_IDF method in some cases take a high frequency for one document, and lower frequency in another documents.
If it is imagined that both sample sentence and input sentence contain set of words $\mathrm{w}_1, \mathrm{w}_2, \ldots, \mathrm{w}_n$. Then, the input sentence is labeled by n-dimensional vector $\mathrm{t} = (\mathrm{t}_1, \mathrm{t}_2, \ldots, \mathrm{t}_n)$ and sample sentence by vector $\mathrm{q} = (\mathrm{q}_1, \mathrm{q}_2, \ldots, \mathrm{q}_n)$.

After defining the input sentence and sample sentence by n-dimensioanl vectors t and q respectively the similarity degree of sentences can be defined by cosines of two vectors t and q:

$$\mathrm{similar}_{\mathrm{tf\_idf}} = \frac{t \cdot q}{\|t\| \cdot \|q\|} = \frac{\sum_{\mathrm{i}=1}^{n} \mathrm{t_i} \cdot \mathrm{q_i}}{\sqrt{\sum_{\mathrm{i}=1}^{n} (t_i)^2} \cdot \sqrt{\sum_{\mathrm{i}=1}^{n} (q_i)^2}} \qquad (6)$$

After finding the similarity degree of by cosines of two vectors t and q, their total degree of similarity is described by similartotal,

$$\mathrm{similar}_{\mathrm{total}} = \mathrm{a}, \mathrm{similar}_{\mathrm{wctn}} + \mathrm{b}, \mathrm{similar}_{\mathrm{tf\_idf}} \qquad (7)$$

Defining the similarity degree of Kazakh language sentences.

The program implementation intended to define a similarity degree of Kazakh language sentences include two major modules: module of inversed index and module of calculating the similarity degree.

The implementation of inversed index is basically adding a new example word into inversed module database after dividing it, which is shown in the figure 3 below

Fig. 3. Adding a new inversed index to the database

The module that calculates the similarity degree is used when it takes the value of input sentence and examples and organize them in descending order with inversed results. Its interface is shown in Figure 4 below.

Fig. 4. The interface used in defining the similarity degree.

4 Results and discussion

In this comparison example contains 1000 sentences and 3500 inversed indexed words. The results of defining the similarity degree according to this data are shown in the 1 table below

Table 1. The results of defining the similarity degree

A number of sentences in the sample database	A number of sentences about to checked	A number of correct sentences	Correctness degree
400	20	8	0.4
800	20	10	0.5
900	20	11	0.55
1000	20	13	0.65

5 Conclusion and future works

The results of study has shown that the similarity degree defined by distances between the input sentences and the sentences in database is significantly different. Also, the figures of separate words are different. For that reason the interdependence between input sentence and the database sentences will always remain constant. This kind of link between words are sorted based on their order.

It is achievable to speed up the search by dividing the sentences into parts using an inverse index. Thus, it was set as a main goal of this work that is to find the match from the samples databases as much quickly as possible.

It is planned to conduct a study that intended to find the similarity degree first of all by calculating the cosine of the input sentence and a sample, combining two methods and find and a similarity degree. Due to the fact that the current databases of samples are small, it is not possible to obtain satisfactory results from translation outputs. In the future, it is planned to expand the database of samples which can facilitate on speeding up the search, improving the data about semantics and grammar of language as a result the quality of the translation might be improved in a significant manner.

References

1. Hutchins, J. (1997): 'From first conception to first demonstration: the nascent years of machine translation, 1947-1954. A chronology.' Machine Translation12, 195-252.
2. Makoto Nagao．A framework of a mechanical translation between Japanese and English by analogy principle．Proceedings of the international NATO symposium on Artificial and human intelligence，Lyon, France，Elsevier North-Holland, Inc.，1984:173-180
3. SaltonG., WongA.,YangC.S.. "A Vector Space Model for Automatic Indexing," Communications of the ACM,1975.– v. 18, №11, – p. 613-620.
4. Xue-qiang, RENFei-liang, HUANG Zhi-dan,YAO Tian-shun Sentence Similarity Model and the Most Similar Sentence Search Algorithm 1005-3026(2003)06-0531-04
5. Jones K.S. A statistical interpretation of term specificity and its application in retrieval // Journal of Documentation : журнал. – MCB University: MCB University Press, 2004. – v. 60, № 5. – p. 493-502.