Discernibility Matrix and Rules Acquisition Based Chinese Question Answering System

Zhao Han[1,2,3], Duoqian Miao[1,3(✉)], Fuji Ren[2], and Hongyun Zhang[1]

[1] The College of Electronics and Information Engineering, Tongji University, Shanghai 201804, China
dqmiao@tongji.edu.cn

[2] The Faculty of Engineering, Tokushima University, Tokushima 770-8506, Japan

[3] The Key Laboratory of Embedded System and Service Computing, Ministry of Education, Tongji University, Shanghai 200092, China

Abstract. Different from English processing, Chinese text processing starts from word segmentation, and the results of word segmentation will influence the outcomes of subsequent processing especially in short text processing. In this paper, we introduce a novel method for Short Text Information Retrieval based Chinese Question Answering. It is developed from the Discernibility Matrix based Rules Acquisition method. Based on the acquired rules, the matching patterns of the training QA pairs can be represented by the reduced attribute words, and the words can also be represented by the QA patterns. Then the attribute words in the test QA pairs can be used to calculate the matching scores. The experimental results show that the proposed representation method of QA patterns has good flexibility to deal with the uncertainty caused by the Chinese word segmentation, and the proposed method has good performance at both MAP and MRR on the test data.

Keywords: Question Answering · Information Retrieval · Rough Set · Discernibility matrix · Rules acquisition · Short Text Similarity

1 Introduction

Question Answering System (QA System) is one of the most recent research topics in Natural Language Processing (NLP). Most of the existing QA systems are based on one of these two architectures: one is Knowledge based Question Answering (KBQA), and the other is Information Retrieval (IR) based Question Answering (IRBQA). KBQA system generates answering based on the given knowledgebase, while IRBQA searches for the best matching sentence or document from a given list of sentences or documents and returns the matched item as the answering. Because of the difficulty of both knowledgebase construction and text generation, IRBQA is more widely used than KBQA [1,2].

The technology of English Question Answering has been developed well, while the research on Chinese Question Answering still faces a lot of difficulties, especially in Chinese Short Text Question Answering. One of the reasons is that

L. Polkowski et al. (Eds.): IJCRS 2017, Part I, LNAI 10313, pp. 239–248, 2017.
DOI: 10.1007/978-3-319-60837-2_20

Chinese text processing starts from word segmentation, and the results of word segmentation will influence the outcomes of subsequent processing. One of the influences is that the wrong segmentation will reduce the count of the similar words between the question and candidate items, and then reduce the similarity between them. Another is that different word segmentation principles and Chinese abbreviation will also cause the decreasing of the similar words, for example, in some context, the Chinese proper noun *"People's Square"* is not similar with the word *"People"*, while the proper noun *"Baidu Company"* is similar with *"Baidu"*. In English, sometimes this kind of proper nouns can be recognized by capital letters, while in Chinese, all the Chinese characters are without grammatical marker. Since each word of a short sentence text takes a large proportion, the semantic representation of the uncertain word segmentation parts plays an important role in the process of QA matching.

In this paper we will introduce a novel method for Short Text and Information Retrieval based Chinese Question Answering. Based on the Rough Set Theory and Discernibility Matrix based Rules Acquisition method, the matching patterns of the training QA pairs can be represented as rules by the reduced attribute words, and the words can also be represented by the QA patterns. Then the attribute words in the test QA pairs can be used to calculate the matching scores. The experimental results show that the proposed representation method of QA patterns has good flexibility to deal with the uncertainty caused by the Chinese word segmentation, and the proposed method has good performance at both MAP and MRR on the test data.

The remainder of the paper is represented as follows: Sect. 2 introduces related works, and Sect. 3 introduces the training processing of system, such like rules acquisition and attribute vector representation. Section 4 introduces the method of matching QA patterns by a trained QA system. Section 5 describes the experiment details and presents the experimental results and analysis. Section 6 is the conclusion and future work.

2 Related Works

Rough Set Theory [10,11] is one of the most popular Granular Computing [13,14] models and can be used to deal with uncertainty problems. In Rough Set Theory, a decision table [12] is defined as *Formula* (1).

$$DecisionTable = \{U, A = C \cup D, V, f\} \tag{1}$$

In a decision table, U is a finite nonempty set of objects, and A is a finite nonempty set of attributes of the objects. A is divided into two subsets, where one is the set of condition attributes and the other is the set of decision attributes. V is a nonempty set of values of all the attributes, and $f : U \times A \rightarrow V$ is the function that maps an object of U by a attribute of A to a value of V. If there are two objects having the same values of all the condition attributes but their decision attribute values are different, the decision table is inconsistent; otherwise it is consistent.

Based on a decision table, we can get its $POS_c(D)$ by $Formula\,(2)$ and (3). $POS_C(D)$ is called a positive region of the partition U/D with respective to C, and is a set of all elements of U that can be uniquely classified to blocks of the partition U/D, by means of C. C_*X is called the $C-lower$ region of X, and $C(x)$ is the equivalence class containing an element x.

$$POS_C(D) = \bigcup_{X \in U/D} C_*X \tag{2}$$

$$C_*X = \{x \in U | C(x) \subseteq X\} \tag{3}$$

Sometimes not all the condition attribute are necessary. If a condition attribute $c \in C$ satisfies $Formula\,(4)$, c is not necessary and can be reduced.

$$POS_{\{C-c\}}(D) = POS_C(D) \tag{4}$$

A lot of Rough Set Theory based methods have been proposed for attribute reduction [3–5]. Our proposed method for QA system is developed from the discernibility matrix theory [6,7]. The classical discernibility matrix is a $|U| \times |U|$ matrix, and its element $M(x, y)$ defined as $Formula\,(5)$. Based on the discernibility matrix, we can get the discernibility function by $Formula\,(6)$.

$$M(x, y) = \{a | a \in A, f(x, a) \neq f(y, a)\} \tag{5}$$

$$df(M) = \wedge\{\vee(M(x, y)) | M(x, y) \neq \emptyset\} \tag{6}$$

In traditional IR, the key words are input by users. Different from IR, the input of IRBQA is natural language sentence. The QA system must abstract the key words from the sentence and then match the most related answers. A lot of works have been done for different English and Chinese QA applications [15,16]. Because of different applications and its corpus or knowledgebase, the method of generating answers is also different, but commonly the QA process is matching the QA pairs by topic similarity. The topic similarity can be measured using the cosine similarity method of word vector representation. In recent years, many word vectorization have been proposed such like VSM [17], LSI [19], LDA [18], Word2Vec [20], and there are also some text similarity related works based on Rough Set method [22,23].

3 Rules Acquisition and Attribute Vectorization

In this section, we will introduce the training processing of our method, including rules acquisition of Chinese QA sentences and vector representations of the attribute words. The attribute word representations are based on the rules, and the representations will be used for matching QA patterns in the testing processing.

3.1 Rules Acquisition of Chinese QA Sentences

Given one question and m labeled candidate items (all the sentences have been segmented into words, the label means whether the item can be used as a answer of the question or not), we first construct a dictionary of all the words of the question and the items. For convenience, we name the item which can match the question as Positive Sentence (PS), and the other Negative Sentence (NS). We name the set of all the PS as Positive Sentence Set (PSS) and the other Negative Sentence Set (NSS). After we get the dictionary, we first remove the words which appear only in the NSS, and also remove some Chinese stopwords. This pre-filtering step will help reduce the dimension and accelerate the following attribute reduction and rules acquisition, and can also make each of the final rule attribute words appear at least once in a PS or the question.

Table 1. Question Answering Matching System (QAMS)

Item/question	w_1	w_2		w_n	Decision label
I_1	v_{11}	v_{12}	...	v_{1n}	v_{1l}
I_2	v_{21}	v_{22}	...	v_{2n}	v_{2l}
...	...	...	...	...	...
I_m	v_{m1}	v_{m2}	...	v_{mn}	v_{ml}
Question	v_{q1}	v_{q2}	...	v_{qn}	v_{ql}

Using the dictionary of n words, we can construct a small Question Answering Matching System (QAMS) for the question and its candidate items, like Table 1. We define this small decision system as $QAMS = \{U = I \cup Q, A = W \cup D, V = 1, 0, f\}$. $I = \{I_1, I_2, \ldots, I_m\}$ is the candidate items set, and Q is a set with only one question in it. $W = \{w_1, w_2, \ldots, w_n\}$ is the word attribute set (the dictionary), and D is the decision attribute set with only the matching label attribute in it. The function $f(u, a)$ is defined as $Formula\,(7)$.

$$f(u \in U, a \in A) = \begin{cases} 1, & \text{if } a \in D \text{ and } u \in PSS \cup Q; \\ & \text{or if } a \in W \text{ and } a \in u \\ 0, & \text{the other} \end{cases} \tag{7}$$

The function $f : U \times A \to V$ means that if an attribute word appears in an item or the question, and the attribute value equals 1, or if the item is a PS or the question, its decision attribute value is 1. Then we need to mining the rules in the $QAMS$. Since for QA system, we only need to concern about the rules for question and its PSS. Then the discernibility matrix of the $QAMS$ is a $x \times y$ matrix, $x = |PSS| + |Q|$, $y = |NSS|$. The values of the QA Discernibility Matrix (QADM) is defined as $Fomular\,(8)$ and (9).

$$\begin{aligned} &Dset(u_p, u_n) = \\ &\quad \{a | a \in W, u_n \in NSS, u_p \in PSS \cup Q, f(u_p, a) \neq f(u_n, a)\} \end{aligned} \tag{8}$$

$$QADM(u_p, u_n) = \begin{cases} Dset(u_p, u_n), & \text{if } |Dset(u_p, u_n)| > 0 \\ \{a | a \in u_p\}, & \text{the other} \end{cases} \tag{9}$$

The discernibility function of the $QADM$ is defined as $Fomular$ (10).

$$df(QADM) = \wedge \{\vee(QADM(u_p, u_n)) | u_n \in NSS, u_p \in PSS \cup Q, QADM(u_p, u_n) \neq \emptyset\} \tag{10}$$

In the function expression of $QADM$, $\vee(QADM(u_p, u_n))$ is the disjunction of all attributes in $QADM(u_p, u_n)$ and $\wedge\{\vee(QADM(u_p, u_n))\}$ is the conjunction of all $\vee(QADM(u_p, u_n))$. When u_p and u_n is inconsistent, that is to say, all of their attribute words are the same, we will set the value of $QADM$ by the attributes of u_p. The original corpus of QA system is consistent theoretically. However, there are two reasons for this definition: one is that it can avoid the error case of the mislabeled items in the corpus, and the other is that after the pre-filtering step the consistent $QAMS$ may turn to inconsistent.

Table 2. An example of a QAMS

Item/question	w_1	w_2	w_3	w_4	Decision label
I_1	0	0	1	1	1
I_2	1	0	1	0	1
I_3	0	1	1	1	0
I_4	0	1	0	1	0
Question	1	1	1	1	1

A $QAMS$ example is showed in Table 2 and its $QADM$ is showed in Table 3. The example $QAMS$ is with 4 attribute words and 4 candidate items. 2 of the 4 candidate items are PSs. The $QADM$ of it is a 3×2 matrix. Based on $Formula$ (10) we can get the discernibility function, showed in $Formula$ (11). The result of $Formula$ (11) means that a question and its PSs can be discerned from the NSs by the words w_1 and w_2.

$$\begin{aligned} df(M) &= (w_2) \wedge (w_2 \vee w_3) \\ &\quad \wedge (w_1 \vee w_2 \vee w_4) \wedge (w_1 \vee w_2 \vee w_3 \vee w_4) \\ &\quad \wedge (w_1) \wedge (w_1 \vee w_3) \\ &= (w_1) \wedge (w_2) \end{aligned} \tag{11}$$

If the result is like $(w_1 \vee w_3) \wedge (w_2)$, that means the discernibility rules can be w_1 and w_2, or can be w_3 and w_2.

Table 3. The QADM of the QAMS in Table 2

	I_3	I_4
I_1	$\{w_2\}$	$\{w_2, w_3\}$
I_2	$\{w_1, w_2, w_4\}$	$\{w_1, w_2, w_3, w_4\}$
Question	$\{w_1\}$	$\{w_1, w_3\}$

3.2 Vector Representation of Attribute Word

Given a set of questions and their labeled candidate items, we can get all of their $QAMS$s, reduced attributed words and rules. Based on the reduced attribute words and the acquired rules, each of the attribute words can be represented as list of vectors. The vector unit v is defined as $Formula\,(12)$. $NO.(QADM)$ is the number label of the $QADM$, $Len(df_{QADM})$ is the sum count of all conjunction elements in the final result of the discernibility function, and $NO.(w_{df})$ is the number label of the conjuncted element of the final result in which the word appears. $T(w_{df})$ is the tag whether the word is appeared in the question or candidate items or both of them.

$$v = [NO.(QADM), \quad Len(df_{QADM}), \quad NO.(w_{df}), \quad T(w_{df})] \tag{12}$$

After we trained a set of questions and its labeled candidate items, all the attribute words can be represented like $Formula\,(13)$. In this Formula, θ is the appearance times of the attribute words in all the $QADM$ of the corpus.

$$WV = [v_1, v_2, \ldots, v_\theta] \tag{13}$$

For example, if the $QAMS$ is the second one of the whole training corpus and the word w_1 and w_2 does not appears in other $QAMS$s, based on $Formula\,(11)$ the word w_1 can be represented as $Formula\,(14)$ and the word w_1 can be represented as $Formula\,(15)$. The ellipsis is the cases of the word appearance vectors in other $QAMS$s.

$$WV_{w_1} = [\quad [2, 2, 1, \{\text{'Q','PSS'}\}] \quad , \ldots] \tag{14}$$

$$WV_{w_2} = [\quad [2, 2, 2, \{\text{'Q'}\}] \quad , \ldots] \tag{15}$$

The attribute words and the acquired rules can be treated as a kind of QA sentence patterns, and $NO.(QADM)$ can be treated as the QA pattern number. However, the model lacks the topic information of the QA. So when it comes to practical application, it must be used at the same time with some topic similarity model.

4 Method of Matching QA Patterns

We can get a dictionary with all the attribute words represented by $Formula\,(13)$. Then when a test question and an unlabeled candidate item are

given, we can get two list of word vector elements from the attribute words appears in the two word sequence: $VL_q = [v_1, v_2, \ldots]$ and $VL_{I_i} = [v_1, v_2, \ldots]$. The next step is to count up the QA pattens and measure their completeness. But before that we must do some preliminary reduction.

At the reduction step, there are two kinds of processing choices. One is that we need to concern the word vector element tag $T(w_{df})$, that means, for example, if a word appears only in the question, and one of its vector tag means it appears only in the NSS in a QA pattern of the train corpus, we must remove it from VL_q. That means we treat strictly that in one QA pattern, the word role of it should not be exchanged. The other processing choice is that we just ignore the tags and we consider that sometimes the words among question and candidate items can be exchanged and will not change the semantic too much.

Table 4. An example of the middle dictionary of the patterns

$NO.(QADM)$	vlist	$Len(df_{QADM})$	vlistlength	C_{QADM}
36	{[36, 4,2, {'Q'}], [36, 4,1, {'PSS'}]}	4	2	0.5
53	{ [53, 1,1, {'Q', 'PSS'}] }	1	1	1
...	...	...	...	...
182	{[182, 4,2, {'Q'}] }	4	1	0

Then based on the $NO.(QADM)$ we count up the pattern and its vector elements (the same elements are counted only once). An example of the middle dictionary of the patterns is illustrated in Table 4. Here we define the completeness of a pattern (QADM) as *Formula* (16).

$$C_{QADM} = \begin{cases} 0, & \text{if } vlistlength = 1 \text{ and } Len(df_{QADM}) \neq 1 \\ \dfrac{vlistlength}{Len(df_{QADM})}, & \text{the other} \end{cases} \tag{16}$$

and the final completeness of the QA pairs is calculated by *Formula* (17).

$$C(q, I_i) = \sum_{\bigcup QADM|q,I_i} C_{QADM} \tag{17}$$

5 Experiment

The experiment is divided into two parts: one is on the sentence pattern similarity and the other is on the text retrieval. As there are two choice at the reduction step of the Matching method (with vector tags and without tags), we evaluate both in the experiment. The first experiment is comparing the proposed method with the word2vec pattern similarity method, and in the second experiment it is compared with cosine similarity of LDA and LSI model. In the second

experiment, the text similarity matching part of our method is the same as LDA baseline.

Both the two experiments use the opensource corpus and toolkits of NLPCC-ICCPOL2016 Shared Task (Evaluation Competition) [8]. The corpus contains a train subset and test subset. The train set contains 8772 question texts, and the test set contains 5997 questions. Each of the question is given a list of candidate items and some of the items can be used as answers to the question. The train set contains 181882 items and the test set contains 122531 items. The baseline models of the experiments are constructed by Gensim Toolkit [9], and the word segmentation of all the Chinese text is completed by the NLPIR (also named as ICTCLAS) tool [21].

In our experiment, the evaluation metrics is the same with the competition: Mean Average Precision (MAP) (see *Formula* (18) and (19)) and Mean Reciprocal Rank (MRR) (see *Formula* (20)).

$$MAP = \frac{1}{|Q|}\sum_{i=1}^{|Q|} AveP(C_i, A_i) \tag{18}$$

$$AveP(C_i, A_i) = \begin{cases} 0, & \text{if } min(m,n) = 0 \\ \frac{\sum_{k=1}^{n}(P(k)\cdot rel(k))}{min(m,n)}, & \text{the other} \end{cases} \tag{19}$$

$$MRR = \frac{1}{|Q|}\sum_{i=1}^{|Q|} \frac{1}{rank_i} \tag{20}$$

Here is the explanation of MAP and MRR from the official document [8]: In MAP formula, k is the rank in the sequence of retrieved answer sentences, m is the number of correct answer sentences, and n is the number of retrieved answer sentences. $P(k)$ is the precision at cut-off k in the list. $rel(k)$ equals 1 if the item at rank k is an answer sentence, otherwise it equals 0. In MRR formula, $rank_i$ is the position of the first correct answer in the generated answer set C_i for the Q_i, and if C_i doesn't overlap with the golden answer A_i for Q_i, $\frac{1}{rank_i}$ equals 0.

The experimental results are in Tables 5 and 6. In Table 5, the withtags version of our method has best performance, but the withouttags version is not unsatisfactory. In Table 6, both the two version of our method have improve the performance of LDA baseline, and they all have better performance that LSI baseline model.

Table 5. Results of sentence patterns similarity experiment

	MAP	MRR
W2Vcosine	0.4075	0.4081
DM (withtags)	**0.4520**	**0.4525**
DM (withouttags)	0.2923	0.2924

Table 6. Results of QA retrieval experiment

	MAP	MRR
LDAcosine	0.6386	0.6392
LSIcosine	0.5372	0.5376
DM (withtags)	**0.6464**	**0.6469**
DM (withouttags)	**0.6436**	**0.6440**

The MAP and MRR results of the withtags version of our method are higher than the withouttags version at both of the two experiments. It shows that at this QA corpus, most of the attribute words have fixed roles in QA patterns. So the final rule expressions acquired by the withtags version method can represent more information of the QA patterns.

6 Conclusion

In this paper a novel method for short text and Information Retrieval based Chinese Question Answering is proposed. It has good flexibility to deal with the Chinese QA uncertainty by mining and representing QA pattern, and the proposed method has good performance at both MAP and MRR on the test data. The future work will focus on more QA experiments by other kinds of feature selection and attribute reduction method based on Rough Sets and on other Chinese and English QA corpus.

Acknowledgments. This work is supported by the National Natural Science Foundation of China (61273304, 61673301, 61573255) and the Specialized Research Fund for the Doctoral Program of Higher Education of China (20130072130004).

References

1. Yang, Y., Jiang, P., Ren, F., et al.: Classic Chinese automatic question answering system based on pragmatics information. In: 7th Mexican International Conference on Artificial Intelligence, pp. 58–64. IEEE Computer Society (2008)
2. Hu, H., Ren, F., Kuroiwa, S., Zhang, S.: A question answering system on special domain and the implementation of speech interface. In: Gelbukh, A. (ed.) CICLing 2006. LNCS, vol. 3878, pp. 458–469. Springer, Heidelberg (2006). doi:10.1007/11671299_48
3. Yao, Y., Zhao, Y.: Attribute reduction in decision-theoretic rough set models. Inf. Sci. **178**(17), 3356–3373 (2008)
4. Wang, J., Miao, D.: Analysis on attribute reduction strategies of rough set. J. Comput. Sci. Technol. **13**(2), 189–192 (1998)
5. Lang, G., Miao, D., Yang, T., et al.: Knowledge reduction of dynamic covering decision information systems when varying covering cardinalities. Inf. Sci. **346**, 236–260 (2016)

6. Skowron, A., Rauszer, C.: The discernibility matrices and functions in information systems. Theory Decis. Libr. **11**, 331–362 (1992)
7. Miao, D.Q., Zhao, Y., Yao, Y.Y., et al.: Relative reducts in consistent and inconsistent decision tables of the Pawlak rough set model. Inf. Sci. **179**(24), 4140–4150 (2009)
8. Duan, N.: Overview of the NLPCC-ICCPOL 2016 shared task: open domain chinese question answering. In: Lin, C.-Y., Xue, N., Zhao, D., Huang, X., Feng, Y. (eds.) ICCPOL/NLPCC -2016. LNCS, vol. 10102, pp. 942–948. Springer, Cham (2016). doi:10.1007/978-3-319-50496-4_89
9. Rehurek, R., Sojka, P.: Software framework for topic modelling with large corpora. In: Proceedings of LREC 2010 Workshop on New Challenges for NLP Frameworks, pp. 45–50 (2010)
10. Pawlak, Z.: Rough sets. Int. J. Parallel Prog. **11**(5), 341–356 (1982)
11. Pawlak, Z.: Rough Sets: Theoretical Aspects of Reasoning about Data. Springer Science and Business Media, Heidelberg (2012)
12. Pawlakab, Z.: Rough set approach to knowledge-based decision support. Eur. J. Oper. Res. **99**(1), 48–57 (1995)
13. Bargiela, A., Pedrycz, W.: Toward a theory of granular computing for human-centered information processing. IEEE Trans. Fuzzy Syst. **16**(2), 320–330 (2008)
14. Yao, J.T., Vasilakos, A.V., Pedrycz, W.: Granular computing: perspectives and challenges. IEEE Trans. Cybern. **43**(6), 1977–1989 (2013)
15. Sun, A., Jiang, M., He, Y., et al.: Chinese question answering based on syntax analysis and answer classification. Acta Electronica Sinica **36**(5), 833–839 (2008)
16. Dwivedi, S.K., Singh, V.: Research and reviews in question answering system. Proc. Technol. **10**(1), 417–424 (2013)
17. Salton, G.: A vector space model for automatic indexing. Commun. ACM **18**(11), 613–620 (1975)
18. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent Dirichlet allocation. J. Mach. Learn. Res. **3**(1), 993–1022 (2003)
19. Papadimitriou, C.H., Tamaki, H., Raghavan, P., Indexing, L.S., et al.: A probabilistic analysis. In: Proceedings of 17th ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems, pp. 159–168. ACM (1998)
20. Mikolov, T., Sutskever, I., Chen, K., et al.: Distributed representations of words and phrases and their compositionality. In: Advances in Neural Information Processing Systems, vol. 26, pp. 3111–3119 (2013)
21. Zhang, H.P., Yu, H.K., Xiong, D.Y., et al.: HHMM-based Chinese lexical analyzer ICTCLAS. In: Proceedings of 2nd SIGHAN Workshop on Chinese Language Processing, vol. 17, pp. 184–187. Association for Computational Linguistics (2003)
22. Janusz, A., Zak, D., Nguyen, H.S.: Unsupervised similarity learning from textual data. Fundamenta Informaticae **119**(3–4), 319–336 (2012)
23. Janusz, A.: Algorithms for similarity relation learning from high dimensional data. In: Peters, J.F. (ed.) Transactions on Rough Sets XVII, pp. 174–292. Springer, Heidelberg (2014)