LCQMC: A Large-scale Chinese Question Matching Corpus

Xin Liu[†], **Qingcai Chen**[†*], **Chong Deng**[‡],
Huajun Zeng[#], **Jing Chen**[†], **Dongfang Li**[†], **Buzhou Tang**[†]
†Shenzhen Calligraphy Digital Simulation Technology Lab, Harbin Institute of Technology, China
‡Machine Intelligence Technology, Alibaba Group
†{hit.liuxin, qingcai.chen, mcdh.chenjing, crazyofapple, tangbuzhou}@gmail.com,
‡dengchong.d@alibaba-inc.com, #aaahchi@hotmail.com

Abstract

The lack of large-scale question matching corpora greatly limits the development of matching methods in question answering (QA) system, especially for non-English languages. To ameliorate this situation, in this paper, we introduce a large-scale Chinese question matching corpus (named LCQMC), which is released to the public[1]. LCQMC is more general than paraphrase corpus as it focuses on intent matching rather than paraphrase. How to collect a large number of question pairs in variant linguistic forms, which may present the same intent, is the key point for such corpus construction. In this paper, we first use a search engine to collect large-scale question pairs related to high-frequency words from various domains, then filter irrelevant pairs by the Wasserstein distance, and finally recruit three annotators to manually check the left pairs. After this process, a question matching corpus that contains 260,068 question pairs is constructed. In order to verify the LCQMC corpus, we split it into three parts, i.e., a training set containing 238,766 question pairs, a development set with 8,802 question pairs, and a test set with 12,500 question pairs, and test several well-known sentence matching methods on it. The experimental results not only demonstrate the good quality of LCQMC, but also provide solid baseline performance for further researches on this corpus.

1 Introduction

Question matching is a fundamental task of QA, which is usually recognized as a semantic matching task, sometimes a paraphrase identification task. The goal of the task is to search questions that have similar intent as the input question from an existing database. QA system learns from a large and noisy question matching corpus, and can span a diverse set of topics (Fader et al., 2013). Semantic matching is often regarded as a binary classification problem. Given a pair of sentences, the system is asked to judge whether two sentences express the same meaning. Semantic matching algorithms are widely used in NLP applications, such as information retrieval, machine translation and knowledge-based question answering. Emerging research also shows that first story detection (Petrovic et al., 2012) and text normalization (Xu et al., 2013; Ling et al., 2013) can also benefit from semantic matching techniques.

Chinese is one of the most widely used languages in the world, and the proportion of Chinese using Internet is increasing rapidly (Qiu et al., 2013). Just the same as the English sentence matching task, many Chinese applications can also benefit from Chinese sentence matching. Though some English paraphrase corpora were published for sentence-level matching tasks, the scales are still not enough to meet the data requirements of deep learning techniques. The problem is even worse in Chinese. Researchers can rarely find large-scale Chinese semantic matching corpora (Hu et al., 2015).

One of the critical challenges in constructing sentence-level matching corpus is how to collect sentence pairs that may have the same meaning. The works in paraphrase corpora can provide some references. Some existing paraphrase corpora over the last decade try to overcome this issue with various machine

[*] Corresponding author

[1] The copurs can be found at http://icrc.hitsz.edu.cn/Article/show/171.html.

Proceedings of the 27th International Conference on Computational Linguistics, pages 1952–1962
Santa Fe, New Mexico, USA, August 20-26, 2018.

learning techniques. The Quora question corpus (Iyer et al., 2017) contains sentences from the online question answering forum where the sentences are mainly questions. In the Microsoft Research Paraphrase corpus (MSRP) (Dolan et al., 2004; B. Dolan and Brockett, 2005) the sentences are distilled from a corpus of news articles gathered from thousands of news sources over an extended period (Rus et al., 2014). Multilingual corpus has also been used to construct the Paraphrase Database(PPDB, (Ganitkevitch et al., 2013; Ganitkevitch and Callison-Burch, 2014)) with machine translation methods. Twitter is also used as a source of paraphrase sentences (Xu et al., 2015).

The collection of general purpose sentence pairs with the same intent is quite difficult, but there are plenty of duplicate questions but with different forms on community question answering websites like Baidu Knows(a popular Chinese community question answering website). Some researchers (Zhang et al., 2016; Zhou and Huang, 2017) had proposed to use the resources on BaiduKnows for question retrieval repectively. However, the main difference is that these researches focused on the judgment of relevance between questions or between the candidate questions and queries instead of semantic equivalence. It naturally inspires a question that could we construct a large-scale question matching corpus based on such websites? To answer this question, in this paper, we tried to construct an open-domain large-scale Chinese question matching corpus.

The main contributions of this paper are listed as follows: 1) Based on publicly available community QA website, we construct and manually annotate an open-domain large-scale Chinese question matching corpus that contains 260,068 pairs; 2) We verify the high quality of the corpus through experiments and detailed sample analysis, which prove the feasibility of the proposed corpus constructing method. 3) We present solid baseline performances of several well-known sentences matching methods on the proposed corpus, which is very helpful for its further using in future research works.

This paper is organized as follows. Section 2 reports related works. Section 3 introduces the procedure of constructing and the annotating of the corpus in detail. Section 4 presents evaluation methods and experimental results. Section 5 gives the detailed analysis of the properties and qualities of the corpus. Section 6 makes the conclusion.

2 Related Works

Iyer et al. (2017) released the Quora question corpus, collected from the Quora forum. The corpus contains over 400,000 question pairs with binary labels corresponding to semantic equivalence or not. As we known, the Quora question corpus is by far the largest manually annotated corpus.

Dolan et al. (2004) released Microsoft Research Paraphrase Corpus. MSRP investigate unsupervised techniques to acquire monolingual sentence-level paraphrases from a corpus of temporally and topically clustered news articles collected from thousands of web-based news sources (Dolan et al., 2004). Levenshtein distance and a heuristic strategy are employed on the construction of the corpus. MSRP consists of 5,801 sentence pairs, 3,900 of which were annotated as paraphrases by human annotators with a binary judgment as to whether the pair constitutes a paraphrase. The MSRP corpus is divided into a training set (4,076 pairs) and a test set (1,725 pairs), and paraphrases in both sets are about twice more than that are not (Rus et al., 2014).

The User Language Paraphrase corpus (ULPC, (McCarthy and McNamara, 2011)) comprises about two thousand target-sentence/student response text-pairs, or protocols. Unlike existing paraphrase corpora, these pairs in ULPC have been evaluated by expert human raters along 10 dimensions of paraphrase characteristics, with a six-point scale rather than traditional binary value evaluating (Rus et al., 2014). From a total of 1,998 pairs, 1,436 (71%) are classified by experts as being paraphrases.

The Question Paraphrase corpus (Bernhard and Gurevych, 2008) contains 1,000 questions along with their paraphrases(totally 7,434 question paraphrases). These questions are randomly selected from 100 FAQ files in the education category of the WikiAnswers website. The question paraphrase corpus constitutes paraphrases by retrieving question paraphrases with the input questions from social Q&A sites.

Rus et al. (2012) developed the SIMILAR corpus to foster a deeper and qualitative understanding of word-to-word semantic similarity metrics. They focus on the more general problem of text-to-text semantic similarity (Rus et al., 2012). They reuse the sentences from MSRP with word-to-word semantic

Seeds	**Returned sentences**
学英语，快点 English learning, Efficient	S_1:怎样学习英语才能又快又好？ En:How to be fast and efficient when learning English?
	S_2:怎样快速提高英语的学习能力 En:How to improve the ability of learning English quickly
	S_3: 英语零基础怎么学习才可以很快进步？ En:How to improve on learning English without any foundations?
	S_4:希望你快点学好中文英语怎么说 En:How to say ”I hope you learn Chinese fast” in English
	S_5:有什么软件可以快点学英语 En:Is there any software to help to learn English fast

Table 1: Simplified example of returned questions by Baidu Knows with the seed query ”English learning, Efficient”. Each ”En” labelled sentence is the English translation of corresponding Chinese sentence.

similarity information. There are 700 pairs in the SIMILAR corpus, and the creators relabelled the semantic equivalence of the selected pairs. 63%(442) TRUE paraphrases are yielded for an overall agreement rate with the MSRP annotations.

There are two versions of the Paraphrase database(PPDB, (Ganitkevitch et al., 2013; Ganitkevitch and Callison-Burch, 2014)), either of which contains millions of multilingual sentence pairs. The available Chinese part tends to concentrate on lexical, phrasal and syntactic forms, rather than complete sentences. Since these paraphrase pairs are automatically constructed, there is no manual annotation.

The other paraphrase corpora mainly come from SemEval task, e.g. the semantic textual similarity corpus(SRA, (Dzikovska et al., 2013)), semantic textual similarity (Agirre et al., 2013), and Twitter paraphrase corpus (Xu et al., 2015; Lan et al., 2017).

As can be seen, there is a myriad of data sets with a large variety of distributions, annotation styles, data sources, etc. To the best of our knowledge, there are no more than 50 thousand manually annotated pairs contained in existing corpora except the Quora corpus.

3 Constructing LCQMC Corpus

In this paper, we use Baidu Knows as the original data source to collect the large-scale sentence pairs. On Baidu Knows, a registered user puts a question and motivates other members to answer it. Each time we feed a query into Baidu Knows, the search engine returns a list of ranked links that point to the question pages with relevant results. Question pages are made up of question sentences, question descriptions, and their corresponding answers. We do this for two reasons: 1) there are nearly one billion real Chinese questions asked by substantial users; 2) there are plenty of duplicate questions with the great variant of expressions, which provide the possibility of constructing large-scale sentence-level pairs for question matching.

3.1 Data Collection

Table 1 illustrates a simplified example of returned questions for a given seed query. It shows that the returned sentences are most relevant, and some sentence pairs that have the same intent can be treated as matching pairs (e.g. the sentences that have the same intent as each other, i.e., S_1, S_2 and S_3). Meanwhile, we should also note that many relevant sentences imply different intents and should be non-matching. (e.g. the pairs S_4 and S_5). So it requires us to identify the matching pairs among these returned sentences, either manually or automatically.

We collect high-frequency words from a wide range of domains, including daily life, education, entertainment, computer games, social, natural science, and sports etc. About 50 unique words of each domain are selected as initial seeds. These initial seeds are feed into Baidu Knows and the top 100 returned pages of each seed are collected. Each page contains a certain number of sentences(e.g. question sentences, question descriptions and their corresponding answers). The purpose of this step is to generate more seeds. We apply term frequency and inverse document frequency(tf.idf) weight on the sentences retrieved by initial seeds. The words in each sentence are calculated and ranked by tf.idf in descending order. Words with higher ranks are selected as the additional seeds and feed into Baidu Knows again. Top 50 pages of each seed are retained, and the question sentences are extracted to compose candidate pairs. The pairs that are identical or differ only by punctuation were discarded. By this way, more than five million question pairs are created.

3.2 Wasserstein Distance based Filtering

Candidate pairs created in the above step are extremely redundant. It's a waste of efforts to annotate all the pairs. To further filter the candidate question pairs, a Wasserstein distance based algorithm called word mover distance(WMD) (Kusner et al., 2015) is employed.

WMD can be cast as an instance of the Wasserstein distance, a well studied transportation problem for which several highly efficient solvers have been developed. WMD relies on word embeddings. It measures the dissimilarity between two sentences as the minimum amount of distance that the embedded words of one sentence need to "travel" to reach the embedded words of another sentence. The more similar the pair is, the smaller the distance value between two sentences is.

We calculate the WMD of each pair and find that almost all the pairs with the distance falling into the interval between 0 and 0.15 are matching, but their expressions do not show big differences. At the same time, all the pairs falling into the interval between 0.45 and 1 are obviously not matching. The rest pairs with WMD values between 0.15 and 0.45 contain both matching and non-matching pairs. It is very hard for WMD to distinguish these pairs. Then we randomly select 10% pairs of this part (that is 260,068 pairs) for manually annotation.

3.3 Annotating Corpus

The main difference between annotating question pairs and paraphrases is the definition of "matching". A paraphrase is a restatement of a text, passage giving the meaning in another form. Though the definition of matching in LCQMC is to some extent similar, it takes the intent of questions into consideration. So, LCQMC will not only focus on the form of words or phrases, which means that even some pairs are different on the semantic meaning, they may still be regarded as matching pairs. The pair " S_6:*After a tense stand-off, the battlewagon turned back.*" and " S_7:*After the French threatened to open fire, the battlewagon turned back*" from MSRP is a typical case in paraphrase definition, sentence S_6 is a restatement of sentence S_7 with words or phrases in another form. While the pair "S_8:我手机丢了，我想换个手机(*My cell phone was lost, now I will have to change another one.*)" and "S_9:我想买个新手机，求推荐(*I want to buy a new cell phone, any advice?*)" is the case that satisfies the definition of question matching with the same intent. The meaning in a single sentence is clear and seems to be different with the other, but if we consider the intent in the questions, we know that both sentences are asked for the same answer: *suggestion for cell phone*.

The annotation is conducted via crowdsourcing. The selected pairs are scored by three annotators. All annotators have professional Chinese background and have been trained on making the judgment for ambiguous pairs. The annotators follow the 3-point Likert scale to measure the degree of semantic similarity between sentences with the score "1", "0" and "0.5". The measurement is the same as defined by Agirre et al. (2012). According to our statistics, in our constructing process, there are about 15% pairs completely inconsistent among three annotators and about 20% pairs uncertain annotations by the annotators. For these pairs, we make a second annotation by other annotators until at least 2/3 annotators give the consistent and certain decisions. Besides, after annotating all pairs, the proportion of positive examples and negative examples is about 7:3, in order to balance the distribution, we first discard some positive pairs that show low quality and supplement some random negative pairs, then we select positive and negative pairs to construct LCQMC. Finally, the annotation pairs are divided into three parts: the training set, the validation set, and the test set. The validation set and test set are further reviewed by expert annotations. The distribution of LCQMC corpus is given in Table 2.

Data	Total	Positive	Negative
training	238,766	138,574	100,192
validation	8,802	4,402	4,400
test	12,500	6,250	6,250

Table 2: The distribution of different data sets in LCQMC corpus.

4 Evaluation and Experimental Results

We evaluate the LCQMC corpus with unsupervised and supervised methods respectively. These methods have been proven effective on semantic matching. Unsupervised methods simply make the matching judgment by using two types of similarity computing methods: string similarity methods and vector space methods. Supervised methods, those algorithms have been proven effective on sentence matching tasks conducted on English corpus(e.g. the Quora question corpus, MSRP, SemEval, PPDB etc.).

For each evaluation data set, we compute the precision(P), recall(R) and F1 score of matching. We also compute the main summary metric (Dzikovska et al., 2013): accuracy(Acc). Accuracy is the overall percentage of correctly classified examples.

4.1 Evaluation Methods

4.1.1 Unsupervised Question Matching

To measure the difficulties of the corpus, we firstly use unsupervised matching methods based on word overlap, n-gram overlap, edit distance and cosine similarity respectively. Word overlap coefficient is the average of simple word overlap, that is the number of common words divided by the average length of the two sentences (Dolan et al., 2004). It is computed by the following formula:

$$C_{wo} = \frac{|S_1 \cap S_2|}{avg(|S_1|, |S_2|)} \tag{1}$$

Similarly, the n-gram overlap coefficient is computed by the following formula:

$$C_{ngram} = \frac{1}{N} \sum_{n=1}^{N} \frac{|G_n(S_1) \cap G_n(S_2)|}{avg(|G_n(S_1)|, |G_n(S_2)|)} \tag{2}$$

where $G_n(S)$ is the set of n-grams in sentence S, and N is usually set to 4 (Barzilay and Lee, 2003).

The edit distance(D_{edt}) of two sentences is the number of words that need to be substituted, inserted, or deleted, to transform one sentence into the other (Bernhard and Gurevych, 2008).

The cosine similarity S_{cos} is computed based on the tf.idf coefficient.

4.1.2 Supervised Question Matching

For supervised question matching, we compare the continuous bag of words and deep neural network methods that have been proven useful for paraphrase task. To provide baseline performance on LCQMC, we run several methods on this corpus in the following sections. The training set is used to train each model, validation set is for parameters selecting and the results are reported on the test set. These baseline methods are introduced below.

- Continuous bag of words(CBOW): first we represent each character or word in one sentence with embeddings orderly. The embeddings are pre-trained with the original sentences. Second, each sentence is represented as the sum of the embedding representations. Third, the output is predicted by feeding concatenated representation of both sentence into a softmax classifier (Blacoe and Lapata, 2012; Yin and Schütze, 2015).

Methods	Emb	P	R	F1	Acc
1.baseline	c	67.0	81.2	73.4	70.6
(WMD)	w	64.4	78.6	70.8	60.0
2.C$_{wo}$	-	61.1	83.6	70.6	70.7
3.C$_{ngram}$	-	52.3	89.3	66.0	61.2
4.D$_{edt}$	-	46.5	86.4	60.5	52.3
5.S$_{cos}$	-	60.1	88.7	71.6	70.3
6.CBOW	c	66.5	82.8	73.8	70.6
	w	67.9	89.9	77.4	73.7
7.CNN	c	67.1	85.6	75.2	71.8
	w	68.4	84.6	75.7	72.8
8.BiLSTM	c	67.4	91.0	77.5	73.5
	w	70.6	89.3	78.92	76.1
9.BiMPM	c	77.6	**93.9**	**85.0**	**83.4**
	w	**77.7**	93.5	84.9	83.3

Table 3: The performance of different methods on LCQMC test set. 'c' means embeddings are character-based and 'w' means word-based.

- Convolutional neural network(CNN): each sentence is represented as an embedding matrix, and the matrix goes through a convolutional neural network (Hu et al., 2014). In this experiment, we make two sentence matrices sharing the same weight of convolution layers, and the convolutional operations follow the convolution in Kim (2014).
- Bi-directional Long Short Term Memory(BiLSTM): first, two sentences go through the same L-STM unit and are encoded into sentence vectors with LSTM encoder in forward and backward direction. Second, we concatenate the representations of both sentences and use softmax to make a classification (Mueller and Thyagarajan, 2016; Tomar et al., 2017).
- Bilateral Multi-Perspective Matching(BiMPM (Wang et al., 2017)): BiMPM uses a character-based LSTM at its input representation layer, a layer of BiLSTMs for computing context information, four different types of multi-perspective matching layers, an additional BiLSTM aggregation layer, followed by a two-layer feedforward network for prediction. BiMPM model has shown state-of-art performance on several NLP task, e.g. paraphrase identification, natural language inference and answer sentence selection.

4.2 Experimental Results

4.2.1 Experimental details

The tool for Chinese word segmentation is jieba[2] and toolkit for computing distance and tf.idf is sklearn[3]. In unsupervised methods, we take the validation set out to choose the threshold for unsupervised methods. The thresholds used to distinguish matching pairs for word overlap coefficient, n-gram overlap coefficient, edit distance, and cosine similarity are 0.65, 0.2, 0.2, 0.7 respectively, the value over thresholds means the pair is matching. In supervised methods, the methods are based on word2vec. We apply gensim[4] to calculate word embeddings on LCQMC. The dimension of embeddings is 200.

4.2.2 Results

Table 3 shows the results of baseline(line 1), unsupervised methods(line 2-5) and supervised paraphrase methods(line 6-9) on the test set. We list WMD performance as the baseline just because it is used to filter the original question pairs. Unsupervised methods try to evaluate the quality of the corpus. The

[2] https://pypi.python.org/pypi/jieba/

[3] http://scikit-learn.org/

[4] http://radimrehurek.com/gensim/index.html

Example	Matching?	Pairs
1	No	S_{10}:飞行员没钱买房怎么办？ En: What can pilots do since they could not **afford** a **house**? S_{11}:父母没钱买房子 En: Parents can not **afford** the **house**.
2	Yes	S_{12}: 聊天室都有哪些好的 En: Are there any better **chat rooms**? S_{13}:聊天室哪个好 En: Which **chat room** is better?
3	Yes	S_{14}:不锈钢上贴的模怎么去除 En: How to wipe off the **film** on **stainless steel** S_{15}:不锈钢上的胶怎么去除 En: How to wipe off the **glue** on **stainless steel**
4	No	S_{16}:动漫人物的口头禅 En: The pet phrase of **character in animation** S_{17}:白羊座的动漫人物。 En: The **characters in animation** who are Aries

Table 4: Examples from the same keywords.

results of each method are listed from line 2 to line 5 in Table 3. All methods show a high recall but with very low precision. This is because the matching pairs in LCQMC indeed share some overlaps, but at the same time, the non-matching pairs also satisfy the overlaps. Obviously, the best performance of unsupervised methods stays around 70% either F1 or Acc. In supervised methods, BiMPM shows the best performance on F1 measure is 85.0% reached by char-based model, which outperforms the char-based WMD 11.6% of F1. The other methods also show significant improvements after being trained with the training set. The performance of CNN model is slightly lower than BiLSTM model, this may be that in CNN model we follow the TextCNN structure while it is not fit for sentence matching, which is actually proposed for classification.

Comparing the experimental results for unsupervised and supervised methods(shown in line 2-5 and line 6-9 of Table 3 respectively), we can see that there is about the average of 10% absolute improvement on both F1 and Acc score, and up to about average of 15% improvement on precision. It not only shows the effectiveness of supervised methods but also proves the effectiveness of the proposed LCQMC corpus. Though through training on LCQMC, we get a big performance gain for the questions matching task, compared to other natural language processing tasks, there is still a big margin left for further research, especially the precision of matching on this corpus is far from satisfaction.

5 Discussion

In this section, we discuss three key issues about the principle of our corpus construction methodology, i.e., the keyword based methodology, the overlap in sentence pairs and the distribution of matching types in the corpus. We make the discussion based on a random selection of 1000 samples form LCQMC.

5.1 Keyword based Methodology

In this paper, to overcome the bottleneck of collecting large-scale real questions that may have the same intent, the keyword-based method is proposed. In general, keyword matching is not a requirement of paraphrase construction and solely relying on keyword matching may limit the quality of the corpus. Taking this issue into consideration, in this paper, the keyword-based searching technology just provides a source for matching instead of the final decision. Keywords give the specific domain information, but the intent of the sentences with the same keywords may differ a lot.

Example 1 in Table 4 is a pair of sentences selected from the search results of the same keywords (bold words). We can clearly see that the sentences S_{10} and S_{11} in the pair are expressing different meaning.

Example	Matching?	Categories	Pairs	Proportion(%)
5	No	High lexical overlap but not paraphrase	S_{18}:从广州到长沙在哪里定高铁票 En: Where to buy high-speed rail tickets from Guangzhou to Changsha S_{19}:在长沙那里坐高铁回广州？ En: Where to take high-speed rail to Guangzhou in Changsha?	14.4%
6	Yes	Low lexical overlap but paraphrase	S_{20}:请问现在最好用的听音乐软件是什么啊 En: Please tell me what is the best software for listening music now. S_{21}:听歌用什么软件比较好 En: Which software is better for listening popular songs?	1.1%
7	Yes	Intent-based	S_{22}:谁有吃过完美的产品吗？如何？ En: Has anyone ate the product of company ”PERFECT”? How? S_{23}:完美产品好不好 En: How about the product from company ”PERFECT”?	2.8%
8	Yes	Elaboration	S_{24}:朱熹是哪个朝代的诗人 En: What dynasty is poet Zhuxi in? S_{25}:朱熹是宋明理学的集大成者,他生活在哪个朝代 En: Zhuxi is the integration of science in Song and Ming Dynasty? Which one does he live in?	2.4%
9	Yes	Phrasal	S_{26}:这是哪个奥特曼？ En: Which Ultraman is this? S_{27}:这是什么奥特曼… En: What Ultraman is this...	5.3%
10	Yes	Synonymy	S_{28}:网上找工作可靠吗 En: Is job hunting online available ? S_{29}:网上找工作靠谱吗 En: Is job hunting online accessible?	4.7%
11	Yes	Reordering	S_{30}:你们都喜欢火影忍者里的谁啊 En: Who do you like in Naruto? S_{31}:火影忍者里你最喜欢谁 En: In Naruto, who do you like most?	11.7%

Table 5: Categories and the corresponding pairs in LCQMC.

Example 2 in Table 4 are from the search results of the keyword ”chat room”, where the meanings of the sentences S_{12} and S_{13} are not determined by the keyword, but by the rest parts in the sentences. Though the lexicons are little different, we can still make the judgment that these two sentences asked for the recommendation of chat rooms and thus are matching. This kind of examples are quite common in LCQMC, we only list some in Table 4. Here Example 3 are matching while Example 4 is not.

5.2 Lexical Overlap of Questions

Another issue that we are concerning about is the high lexical overlaps between pairs of sentences. We have mentioned it in Section 4.1.1, e.g. the results by unsupervised methods. In fact, the high lexical overlaps have been common in several existing paraphrase data sets, especially those studying sentence-level paraphrases. In MSRP, the average lexical overlap equals about 70% while in the ULPC corpus the average lexical overlap is about 60%. In LCQMC, the average lexical overlap is about 75%. Of course, the problems are brought by the building method, but one main reason is that it is too hard to get such cases with low overlaps but matching in the natural background, especially for manual annotation.

The other reason may be the commonness of users to express the same thing when asking questions in the search engine. Matching pairs with low lexical overlaps are ideal instances. We cannot make the corpus cover all these cases, but the cases indeed contribute to the diversity of matching pairs. Luckily, since we annotate plenty of pairs, there is also a small part falling into the low lexical overlap area. According to our statics, the cases that are matching with less than 50% overlaps are about 2.5% of the whole corpus. Example 6 in Table 5 is an example of matching pairs with low lexical overlaps. In this kind of pairs, the overlaps are far less than 50%.

Besides, non-matching pairs are another aspect to prove the quality of the corpus. The lexical overlaps in non-matching pairs have a great influence. Just opposite to the matching pairs, non-matching pairs need to be lexical overlaps as many as possible. The more lexical overlaps there are, the harder it is for algorithms to identify. In LCQMC, the non-matching pairs satisfy the requirement very well, because the non-matching pairs are selected by WMD and only kept with smaller Wassertein distance. In most cases, the smaller distance shares the views with higher lexical overlaps except those are expressed with synonyms. Example 5 in Table 5 is a non-matching pairs. The differences between the two sentences are only a few words, which is extremely hard for algorithms to identify.

5.3 Question Matching Types

To explore types of matching pairs in LCMQC, we manually examined the random 1000 samples of sentence pairs from the corpus. The categories of intent-based, elaboration, phrasal, synonymy and reordering often appear in matching pairs.

- Intent-based: Sentence pairs are matching because of the intent instead of the semantic meaning.
- Elaboration: Sentence pairs are different on text content, with an extra word, phrase or clause in one sentence that has no counterpart in the other.
- Phrasal: A phrase in one sentence is changed with another words or phrases in the other.
- Synonymy: Sentences share a little difference but the differences are synonymy.
- Reordering: Words, phrases or other constituents occur in different orders between two sentences.

Table 5 list some categories and the corresponding pairs in LCQMC, the last column is the proportion of each category in the random 1000 samples. The sum of the proportion listed in the Table is about 44%, this is because that some common cases are not listed, e.g. matching pairs with some unnecessary words and non-matching pairs.

The categories in matching pairs are far from comprehensive, there are still some other types that a matching corpus is expected to contain. Some possible categories(e.g. Requiring world knowledge, Metaphoric, Named entity, etc.) referred from (Bernhard and Gurevych, 2008) are what semantic matching algorithms expect and need to handle.

6 Conclusion

In this paper, a large-scale Chinese question matching corpus is constructed from Baidu Knows and is manually annotated. The evaluation is conducted through unsupervised and supervised sentence matching methods. Experimental results not only show that the proposed corpus is helpful for further research on Chinese question matching and other related tasks, they also present solid baseline performance on the proposed corpus. Additionally, our further discussions show the feasibility of the proposed corpus construction methodology. Different from the word-level paraphrase corpus, the LCQMC is mainly focused on the variance of expressions for the same intent, rather than the variance of vocabularies for the same meaning.

In addition to current work, the application of more bootstrapping techniques could be one of the most fruitful research direction for further increasing the scale of the corpus. Seeking for different types of available resources is also a key work for further improving the quality of the corpus.

Acknowledgements

We would like to thank the anonymous reviewers for their helpful feedback. This work is supported by Natural Science Foundation of China (Grant No. 61473101, 61573118), Strategic E-merging Industry Development Special Funds of Shenzhen(Grant No. JCYJ20170307150528934, J-CYJ20160531192358466) and the joint project foundation of Alibaba Group.

References

Eneko Agirre, Daniel Cer, Mona Diab, and Aitor Gonzalez-Agirre. 2012. Semeval-2012 task 6: A pilot on semantic textual similarity. In **SEM 2012: The First Joint Conference on Lexical and Computational Semantics – Volume 1: Proceedings of the main conference and the shared task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation (SemEval 2012)*, pages 385–393, Montréal, Canada, 7-8 June. Association for Computational Linguistics.

Eneko Agirre, Daniel Cer, Mona Diab, Aitor Gonzalez-Agirre, and Weiwei Guo. 2013. *sem 2013 shared task: Semantic textual similarity. In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 1: Proceedings of the Main Conference and the Shared Task: Semantic Textual Similarity*, pages 32–43. Association for Computational Linguistics.

William B. Dolan and Chris Brockett. 2005. Automatically constructing a corpus of sentential paraphrases. In *Proceedings of the Third International Workshop on Paraphrasing (IWP2005)*.

Regina Barzilay and Lillian Lee. 2003. Learning to paraphrase: An unsupervised approach using multiple-sequence alignment. In *Proceedings of the 2003 Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*.

Delphine Bernhard and Iryna Gurevych, 2008. *Proceedings of the Third Workshop on Innovative Use of NLP for Building Educational Applications*, chapter Answering Learners' Questions by Retrieving Question Paraphrases from Social Q&A Sites, pages 44–52. Association for Computational Linguistics.

William Blacoe and Mirella Lapata. 2012. A comparison of vector-based representations for semantic composition. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 546–556. Association for Computational Linguistics.

Bill Dolan, Chris Quirk, and Chris Brockett. 2004. Unsupervised construction of large paraphrase corpora: Exploiting massively parallel news sources. In *COLING 2004: Proceedings of the 20th International Conference on Computational Linguistics*.

Myroslava Dzikovska, Rodney Nielsen, Chris Brew, Claudia Leacock, Danilo Giampiccolo, Luisa Bentivogli, Peter Clark, Ido Dagan, and Trang Hoa Dang. 2013. Semeval-2013 task 7: The joint student response analysis and 8th recognizing textual entailment challenge. In *Second Joint Conference on Lexical and Computational Semantics (*SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, pages 263–274. Association for Computational Linguistics.

Anthony Fader, Luke Zettlemoyer, and Oren Etzioni. 2013. Paraphrase-driven learning for open question answering. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics*, volume 1, pages 1608–1618. Association for Computational Linguistics.

Juri Ganitkevitch and Chris Callison-Burch. 2014. The multilingual paraphrase database. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC-2014)*. European Language Resources Association (ELRA).

Juri Ganitkevitch, Benjamin Van Durme, and Chris Callison-Burch. 2013. Ppdb: The paraphrase database. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 758–764. Association for Computational Linguistics.

Baotian Hu, Zhengdong Lu, Hang Li, and Qingcai Chen. 2014. Convolutional neural network architectures for matching natural language sentences. In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 27*, pages 2042–2050. Curran Associates, Inc.

Baotian Hu, Qingcai Chen, and Fangze Zhu. 2015. Lcsts: A large scale chinese short text summarization dataset. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1967—1972. Association for Computational Linguistics.

Shankar Iyer, Nikhil Dandekar, and Kornél Csernai. 2017. First quora dataset release: Question pairs.

Yoon Kim. 2014. Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1746–1751. Association for Computational Linguistics.

Matt J. Kusner, Yu Sun, Nicholas I. Kolkin, and Kilian Q. Weinberger. 2015. From word embeddings to document distances. In *Proceedings of the 32nd International Conference on Machine Learning(ICML 2015)*, pages 957–966.

Wuwei Lan, Siyu Qiu, Hua He, and Wei Xu. 2017. A continuously growing dataset of sentential paraphrases. In *Proceedings of The 2017 Conference on Empirical Methods on Natural Language Processing (EMNLP)*, pages 1235–1245. Association for Computational Linguistics.

Wang Ling, Chris Dyer, W. Alan Black, and Isabel Trancoso. 2013. Paraphrasing 4 microblog normalization. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 73–84. Association for Computational Linguistics.

Philip M. McCarthy and Danielle S. McNamara, 2011. *The user-language paraphrase corpus*, pages 73–89. IGI Global.

Jonas Mueller and Aditya Thyagarajan. 2016. Siamese recurrent architectures for learning sentence similarity. In *AAAI*, pages 2786–2792.

Sasa Petrovic, Miles Osborne, and Victor Lavrenko. 2012. Using paraphrases for improving first story detection in news and twitter. In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 338–346. Association for Computational Linguistics.

Xipeng Qiu, Qi Zhang, and Xuanjing Huang. 2013. Fudannlp: A toolkit for chinese natural language processing. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 49–54. Association for Computational Linguistics.

Vasile Rus, Mihai Lintean, Cristian Moldovan, William Baggett, Nobal Niraula, and Brent Morgan. 2012. The similar corpus: A resource to foster the qualitative understanding of semantic similarity of texts. In *In Semantic Relations II: Enhancing Resources and Applications, The 8th Language Resources and Evaluation Conference (LREC 2012)*.

Vasile Rus, Rajendra Banjade, and Mihai Lintean. 2014. On paraphrase identification corpora. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC-2014)*. European Language Resources Association (ELRA).

Gaurav Singh Tomar, Thyago Duque, Oscar Täckström, Jakob Uszkoreit, and Dipanjan Das. 2017. Neural paraphrase identification of questions with noisy pretraining. *EMNLP 2017*, page 142.

Zhiguo Wang, Wael Hamza, and Radu Florian. 2017. Bilateral multi-perspective matching for natural language sentences. In *Proceedings of the 26th International Joint Conference on Artificial Intelligence*, pages 4144–4150. AAAI Press.

Wei Xu, Alan Ritter, and Ralph Grishman. 2013. Gathering and generating paraphrases from twitter with application to normalization. In *Proceedings of the Sixth Workshop on Building and Using Comparable Corpora*, pages 121–128. Association for Computational Linguistics.

Wei Xu, Chris Callison-Burch, and Bill Dolan. 2015. Semeval-2015 task 1: Paraphrase and semantic similarity in twitter (pit). In *Proceedings of the 9th International Workshop on Semantic Evaluation (SemEval 2015)*, pages 1–11. Association for Computational Linguistics.

Wenpeng Yin and Hinrich Schütze. 2015. Discriminative phrase embedding for paraphrase identification. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1368–1373. Association for Computational Linguistics.

Kai Zhang, Wei Wu, Fang Wang, Ming Zhou, and Zhoujun Li. 2016. Learning distributed representations of data in community question answering for question retrieval. pages 533–542.

Guangyou Zhou and Jimmy Xiangji Huang. 2017. Modeling and learning distributed word representation with metadata for question retrieval. pages 1226–1239.