# Zero-Shot Question Classification Using Synthetic Samples

**Hao Fu[1], Caixia Yuan[1], Xiaojie Wang[1], Zhijie Sang[1], Shuo Hu[2], Yuanyuan Shi[2]**

[1]Beijing University of Posts and Telecommunications, Beijing, China
[2]Beijing Samsung Telecom R&D Center, Beijing, China
{haofu,yuancx,xjwang,woailaosang}@bupt.edu.cn, {shuo.hu,yy.shi}@samsung.com

**Abstract:** The quality of question classification is vital for a practical question-answering system. This paper proposes a transfer learning method based on generating virtual data for zero-shot questions. The basic idea is to exploit the commonality and difference between zero annotated questions and large enough annotated questions to generate virtual training data for zero annotated questions, thereby relieving the problem of data imbalance and improving performance of question classifier. Concretely, we first apply a template-based generator to generate basic virtual samples, then use them to train an encoder-decoder based generator to generate large enough virtual data. Finally, the real samples and virtual ones are used to train a supervised question classifier. Experiments show that the proposed method improves the overall classification performance both for English and Chinese data sets. Especially, the classification performance of zero annotated questions increased significantly, from 7.46% to 59.34% for English and from 1.96% to 42.67% for Chinese, and the generated virtual data has minute impact on the performance of large annotated question test set.

**Keywords:** Question Classification; Transfer Learning; Virtual Data Generator; Encoder-Decoder

## 1 Introduction

Question answering (QA) system [1] has made significant progress due to the advance of the widespread use of deep learning that are optimized on large-scale dataset of human labeled frequently asked question (FAQ) [2][3]. Despite the exciting advances, to recognize user questions which have been seldom or never asked remains a daunting challenge [4]. In practical applications, however, this kind of questions occupies a long tail of the whole question distribution even in the "big data" setting.

For example, an operator of a mobile phone customer service can easily recognize the type of a question about "screen" and "battery", while will be puzzled by a question about "root". A FAQ based automatic QA system usually includes a set of standard questions (SQs), their extended questions (EQs, the manually created alternatives of SQs) and corresponding standard answers. EQs play critic roles to train a supervised question classifier for categorizing a real user question to a SQ. However, typical machine learning models require hundreds or thousands of training examples, it is far from enough to directly build a high-quality classifier for SQs with zero EQs, i.e., zero-shot questions.

Transfer learning [5-8] is a learning framework that expands knowledge gained while classifying one question to a different but related question. Usually, the large enough annotated problems are regarded as source domain, while few or zero annotated problem are regarded as target domain [9]. We conduct transfer learning by generating synthesized questions for target domain via using knowledge gained from the source domain, ultimately, to improve the accuracy of question classifier, especially its performance for target zero-shot questions. To this end, we need to address a key problems: (1) for a zero-shot question, how to find the related frequently asked questions that have plenty of EQs and (2) how to synthesize high-quality virtual samples for zero-shot questions.

To address the first problem, we use an unsupervised Word Edit Mover's Distance (WEMD) to measure the similarity between two questions. To address the second problem, we first collect training set for generating virtual samples. For two SQs with small enough WEMD, we use a word level replacement to generate basic virtual samples, then the genuine similar questions are taken as input, and corresponding template-generated virtual sample is taken as standard output to train an automatic virtual sample generator using an encoder-decoder framework. The encoder is a Bi-Directional Attention Flow (BiDAF) [10] comprehensive network which is designed to infer the commonality and difference among similar questions, and a RNN decoder takes as input hidden state of the encoder and outputs a word sequence as a virtual question.

Our main contribution is a novel idea to translate the challenging problem of classify zero-shot questions into generating high quality virtual samples for them. This idea can not only be used to construct an effective question answering system, but also be extended to general tasks with seriously imbalanced samples. Our empirical studies extensively test the proposed method on both English and Chinese QA datasets and yield encouraging experimental results when compared with strong baseline models.

## 2 Related work

There is a large body of research on question classification in FAQ based question answering system [11-14]. Although most of them achieve high-quality

**978-1-5386-6005-8/18/$31.00 ©2018 IEEE**

performance on huge annotated dataset, the classification for few or zero annotated questions is far beyond a trivial problem. A typical trouble is a zero-shot question is often easily misclassified into questions with large annotated samples.

Initial approaches turn to utilize heuristic rules to enrich small samples. Chen et. Al [15] use the semantic framework - FrameNet to automatically induce and fill semantic slots in an unsupervised way, and improve the language understanding ability in the target domain. Ferreira et. Al [16] use a framework-like approach to generate training samples for target domain using ontology and external word vectors. Zhu et. Al [17] extract the template from the source domain sample to generate samples for the target domain.

Another class of approaches relies on knowledge transferring by modeling common features or hyper-parameters that are shared across domains. Dai et. Al [18] propose co-clustering based classification (CoCC) to cluster document and word features, and transfer knowledge by sharing word features in different domains. Yazdani and Henderson [19] establish a statistical model that can have good generalization ability for samples that have not appeared in the training data and generate small sample data through this method.

Different from the previous work, we propose a plausible alternative for transferring inter-class semantic structures from a reading and comprehensive view. In particular, the target question is viewed as a query, the similar source question and its EQs are viewed as passages, a copying-generating network derives answers from the passages. Finally the derived answers are used as virtual samples for the target question to expand training data for a supervised Convolutional Neural Network (CNN) question classifier.

## 3 Virtual Sample Generator

We regard a SQ with zero EQ as a target question $std^{(T)}$, $std^{(T)} \in Q^{(T)}$, and a SQ with large EQs as a source question $std^{(S)}$, $std^{(S)} \in Q^{(S)}$, with $ext^{(S)}$, $ext^{(S)} \in Q^{(S)}$ as one of its EQs. Given a triples $<std^{(S)}, std^{(T)}, ext^{(S)}>$, the goal of virtual sample generator is to generate a virtual samples $ext^{(T)}$ for $std^{(T)}$ using $std^{(S)}$ and $ext^{(S)}$**.** Without loss of generality, we assume $std^{(S)}$ is a similar question of $std^{(T)}$.

### 3.1 Template-based Generator

In our work, we use a modified Word Mover's Distance (WMD) [20] to measure the similarity between two questions. In WMD, each text has a dimension with the number of vocabulary of the whole text set, therefore the complexity of the calculation is great. To alleviate this problem, we add an Edit Distance to derive the so-called WEMD (Word Edit Mover's Distance).

We think that when distance $D(std^{(S)}, std^{(T)})$ between $std^{(S)}$ and $std^{(T)}$ is less than $\alpha$, $std^{(T)}$ can be used as the target domain of $std^{(S)}$. We will select the nearest $N$ standard question for each Target SQ.

$$D(std^{(S)} \to std^{(T)}) = \sum_{k=1}^{j} \min(d_{k1}, d_{k2}, \dots, d_{ki})$$
$$D(std^{(T)} \to std^{(S)}) = \sum_{k=1}^{i} \min(d_{1k}, d_{2k}, \dots, d_{jk}) \quad (1)$$

$$D(std^{(S)}, std^{(T)}) = \frac{D(std^{(S)} \to std^{(T)}) + D(std^{(T)} \to std^{(S)})}{2(i+j)} \quad (2)$$

Where $d_{ij}$ is the cosine distance between two words, $D(std^{(S)}, std^{(T)})$ is a symmetric distance of $std^{(S)}$ and $std^{(T)}$. If $D(std^{(S)}, std^{(T)})$ is less than a small threshold $\alpha$ (e.g., $\alpha=0.2$), we use a template with word-level replacement to generate virtual samples for $std^{(T)}$ from $std^{(S)}$ and its EQs. As an illustrating instance, consider a target question $std_i$="iPhone 怎么开机" and a source question $std_j$="Note4 怎么开机". There is only one word that is different from each other, from $ext_j$="iPhone 如何打开", we can get for $std_i$ a virtual sample $ext_i$="Note4 如何打开".

However, the simple template based generator can only generate virtual samples for a small number of zero-shot questions. Designing universal templates for the whole zero-shot questions is non-trivial problem. Therefore, we solve this problem using a more versatile comprehension-based approach.

### 3.2 Comprehension-based Generator

Given quadruples $<std^{(S)}, std^{(T)}, ext^{(S)}, ext^{(T)}>$ where $ext^{(T)}$ is generated through the above template-based generation, $std^{(T)}$ is regarded as a query, $std^{(S)}$ and $ext^{(S)}$ constitute a context, the comprehension-based generator is trained using triples $<std^{(S)}, std^{(T)}, ext^{(S)}>$ and corresponding $ext^{(T)}$ as standard answers.

We use a BiDAF-based Pointer-Generator (BDPG) which combines the BiDAF [10] and the Pointer-Generator [21] to construct a comprehension-based generator with architecture shown in Figure 1.

**Encoder Layer**: In BiDAF, the input includes *Query* and *Context*. use *Query2Context* and *Context2Query* attention to get the intermediate state $\{m_1, m_2, \dots, m_G\}$ and then get the *ContextVector* as the input to the decoder layer:

$$ContextVector = m_1 : m_2 : \dots : m_G \quad (3)$$

**Decoder Layer**: In Pointer-Generator, the output probability $P(w)$ at each moment consists of generation and copy, where the probability of producing a word out of the context is $p_{gen}$, and the probability of copying a word $w$ is $P_{vocab}(w)$, then the output probability of each word is:

$$P(w) = p_{gen} P_{vocab}(w) + (1 - p_{gen}) \sum_{G:w_G=w} m_G \quad (4)$$

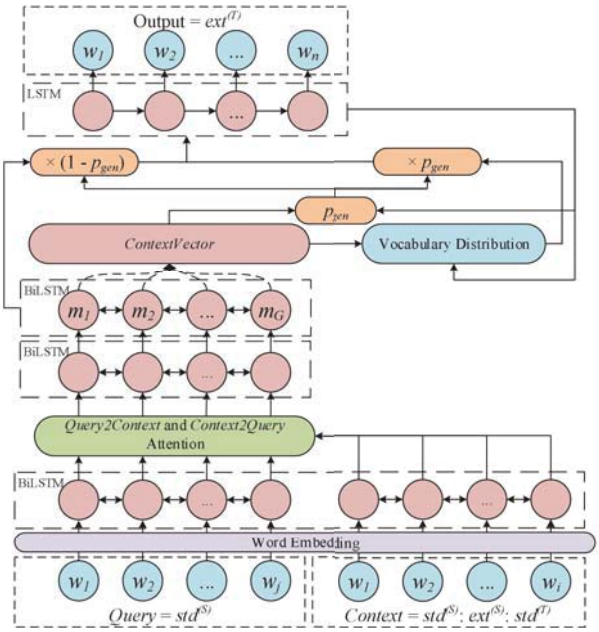


**Figure 1** Structure of Comprehension-based Generator

The generated word sequence is the synthetic samples for zero-shot questions. Finally, the original data and synthetic ones are used to train a CNN classifier.

### 3.3 Variants of BDPG

On the basis of BDPG, we studied several interesting and thought-invoking variants, including:

**Multi-BiDAF-Pointer-Generator (MBDPG)**: As shown in Figure 2(a), We use the source $std^{(S)}$ as *Query*, the target $std^{(T)}$ as the *Context*, and the latter's unique text as the output. For example, $std_i$="iPhone 怎么开机" and $std_j$="Note4 怎么开机", then the latter's unique text is "Note4". At the same time, we use the source SQ as *Query*, the source EQ as the *Context*, and the latter's unique text as the output. Finally, the two outputs are combined for final output.

**Split-BiDAF-Pointer-Generator (SBDPG)**: As shown in Figure 2(b), We use the source SQ as *Query*, the source EQ as the *Context*, the latter's unique text as the output, and finally splicing it directly with the target SQ as the final output.

**Random-Split-BiDAF-Pointer-Generator (RSBDPG)**: We randomly shuffle the output of SBDPG with the word granularity as the final output.

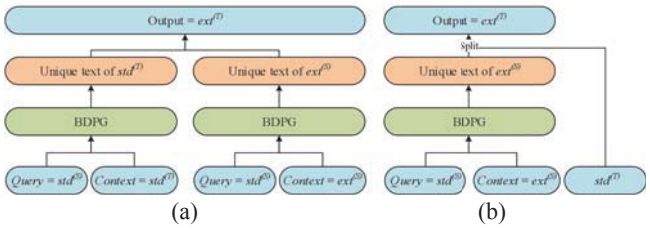


**Figure 2** Structures of MBDPG and SBDPG

The following experiments will discuss the different architectures.

## 4 Experiments

### 4.1 Data and Setting

We use two dataset, the Samsung's internal Chinese QA data and Quora English QA data, to test the effectiveness of the proposed comprehension-based generator in solving imbalanced question classification task.

The Chinese data set is collected from a real online customer service system, including a total of 1620 SQs and 324,402 corresponding EQs, and each SQ has more than 9 EQs.

The English data set comes from processed Quora corpus, including a total of 510 SQs and 8,437 corresponding EQs, and each SQ has more than 9 EQs.

**Table 1** Information of Data Set.

| Type | Samsung Data Set | | Quora Data Set | |
|---|---|---|---|---|
| | Name | Size | Name | Size |
| Train Set | *Source-SQ-Ch* | 1,296 | *Source-SQ-En* | 408 |
| Train Set | *Source-EQ-Ch* | 238,436 | *Source-EQ-En* | 5,888 |
| Train Set | *Target-SQ-Ch* | 324 | *Target-SQ-En* | 102 |
| Test Set | *Source-Test-Ch* | 26,651 | *Source-Test-En* | 847 |
| Test Set | *Target-Test-Ch* | 59,314 | *Target-Test-En* | 1,702 |

For the two data sets, we randomly selected 80% of the whole questions as source samples, the remaining 20% as target samples. 90% of the source samples and all the standard target questions are used as training data for the CNN classifier. While the remaining 10% of source samples and all the target extended questions are used as test sets. That is, in the training set, all target questions are zero-shot questions. The detailed data set is listed in Table 1.

All experiments follow the principle of counterpart parameters. The Chinese and English word vectors are pre-trained using Glove respectively on Samsung and Common Crawl corpus. The word dimension is 300.

### 4.2 The Quantity and Quality of Generators

Using *Source-SQ-Ch*, *Source-EQ-Ch*, and *Target-SQ-Ch* in Table 1, the template-based generator generated a total of 48,125 virtual samples for Chinese QA. 80% of them are used for training the comprehension-based generator, the remaining 20% are used to test the quality of generated samples in BLEU values.

**Table 2** BLEU of comprehension-based generator.

| Model | Name | Size | BLEU-1 | BLEU-4 |
|---|---|---|---|---|
| BDPG | *Training data* | 38,500 | 0.9848 | 0.9513 |
| | *Test data* | 9,625 | 0.9717 | 0.9409 |

As shown in Table 2, the BLEU values of BDPG is at a very high level both for training data and test data. A well-trained comprehension-based generator are used to generate virtue samples for more zero-shot questions that the templated-generator cannot cover. As shown in Table 3, the template-based generator can only capture

the similarity pattern of word-level between similar samples, while the comprehension-based generator is good at capturing similar semantics between similar samples and generate reliable virtual samples accordingly.

### 4.3 Results of Different Models

For the different models used by the comprehension-based generators in this article, we tested their classification enhancements for source and target samples. As shown in Figure 3, because the data quality generated by the BDPG is not good, it will cause large interference to the source sample classification. We added discriminant filtering to the data, namely BDPG(Judge).

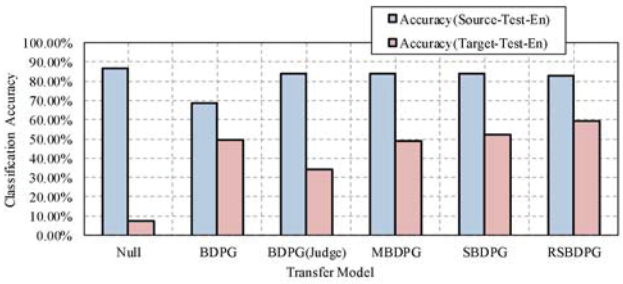


**Figure 3** Classification Accuracy of Different Model

Figure 3 illustrates the results of baseline CNN classifier without virtual samples and that with virtual samples generated by different comprehension-based generators. It can be seen that RSBDPG achieves highest accuracy both on target samples and on source samples.

As shown in Table 4, for the English dataset, the comprehension-based generator improves the accuracy of target sample classification by 39.72% higher than the template-based one, and can generate virtual samples for all target samples (the coverage of zero-shot questions=100%). For the Chinese dataset, the classification accuracy rate for target samples is 21.74% higher than that of the template-based ones, and can generate virtual samples for 95.19% of target samples. Moreover, for the English dataset, the templet combined comprehension-based generator improves the accuracy of target sample classification by 51.88% higher than no generator. For the Chinese dataset, the classification accuracy rate for target samples is 40.71% higher than no generator. It is also worthy to note that, the classification accuracy of source samples is not effected by the virtual samples.

## 5 Conclusion

This paper proposes a transfer learning method for extremely imbalanced question classification task. A comprehension-based model is investigated, which is verified versatile for understanding the semantic correlations of different questions, and can efficiently generate credible virtual samples for zero-shot questions. In the future work, we will investigate methods to improve the diversity of virtual samples to and validate its applicability for larger scale of imbalanced classification tasks.

**Table 3** Examples of Synthetic Samples of Different Generators

| Type of Question | Question Examples | |
|---|---|---|
| | Example 1 | Example 2 |
| $std^{(S)}$ | a8000 需要 优化 内存 | 手机 如何 连接 电脑 传输 文件 |
| $ext^{(S)}$ | a8000 内存 占用 大 怎么 优化 | 手机 照片 导入 电脑 上 |
| $std^{(T)}$ | c7000 需要 优化 内存 | 手机 之间 如何 传输 文件 |
| $ext^{(T)}$ generated by template | **c7000 内存 占用 大 怎么 优化** | ------ （*Template cannot be generated*） |
| $ext^{(T)}$ generated by BDPG | **c7000 内存 占用 大 怎么 优化** | **手机 照片 如何 导入 手机 上 <EOS>** |

**Table 4.** Classification Accuracy of Template and Comprehension(RSBDPG)-based Generator

| Corpus | Generator Type for Virtual Sample | Classification Accuracy | | | The coverage of zero-shot questions |
|---|---|---|---|---|---|
| | | Source Test Set | Target Test Set (Cover) | Target Test Set (All) | |
| Quora | ------ | 86.65% | ------- | 7.46% | ------ |
| Quora | Templet | 86.77% | 39.13% | 17.21% | 27.03% |
| Quora | Comprehension | 83.46% | 56.93% | 56.93% | 100.00% |
| Quora | Templet + Comprehension | 83.01% | 59.34% | 59.34% | 100.00% |
| Samsung | ------ | 76.59% | ------- | 1.96% | ------ |
| Samsung | Templet | 76.25% | 59.48% | 20.93% | 33.60% |
| Samsung | Comprehension ($\alpha$=0.20) | 73.54% | 33.77% | 21.97% | 64.67% |
| Samsung | Templet + Comprehension ($\alpha$=0.20) | 74.69% | 43.21% | 31.10% | 69.07% |
| Samsung | Comprehension ($\alpha$=0.30) | 70.46% | 37.41% | 33.88% | 95.19% |
| Samsung | Templet + Comprehension ($\alpha$=0.30) | 71.32% | 43.54% | 42.67% | 95.44% |

## References


[1] Feng M, Xiang B, Glass M R, et al. Applying Deep Learning to Answer Selection: A Study and An Open Task[J]. 2015:813-820.

[2] Moreo A, Eisman E M, Castro J L, et al. Learning regular expressions to template-based FAQ retrieval systems[J]. Knowledge-Based Systems, 2013, 53(9):108-128.

[3] Shaikh A D, Jain M, Rawat M, et al. Improving Accuracy of SMS Based FAQ Retrieval System[M]// Multilingual Information Access in South Asian Languages. Springer Berlin Heidelberg, 2013:142-156.

[4] Wang Y X, Hebert M. Learning to Learn: Model Regression Networks for Easy Small Sample Learning[C]// European Conference on Computer Vision. Springer, Cham, 2016:616-634.

[5] He H, Garcia E A. Learning from Imbalanced Data[M]. IEEE Educational Activities Department, 2009.

[6] Kubat M, Matwin S. Addressing the Curse of Imbalanced Training Sets: One-Sided Selection[C]// International Conference on Machine Learning. 2012:179--186.

[7] Pan S J, Yang Q. A Survey on Transfer Learning[J]. IEEE Transactions on Knowledge & Data Engineering, 2010, 22(10):1345-1359.

[8] Weiss K, Khoshgoftaar T M, Wang D D. A survey of transfer learning[J]. Journal of Big Data, 2016, 3(1):9.

[9] Gan C, Yang T, Gong B. Learning Attributes Equals Multi-Source Domain Generalization[J]. 2016:87-97.

[10] Seo M, Kembhavi A, Farhadi A, et al. Bidirectional Attention Flow for Machine Comprehension[J]. 2016.

[11] Quan X, Liu G, Lu Z, et al. Short text similarity based on probabilistic topics[J]. Knowledge & Information Systems, 2010, 25(3):473-491.

[12] Article G T. Measuring semantic similarity between words using web search engines[J]. Computer Science, 2015:757-766.

[13] Mueller J, Thyagarajan A. Siamese recurrent architectures for learning sentence similarity[C]// Thirtieth AAAI Conference on Artificial Intelligence. AAAI Press, 2016:2786-2792.

[14] Neculoiu P, Versteegh M, Rotaru M. Learning Text Similarity with Siamese Recurrent Networks[C]// Repl4nlp Workshop at ACL. 2016.

[15] Chen Y N, Wang W Y, Rudnicky A I. Unsupervised induction and filling of semantic slots for spoken dialogue systems using frame-semantic parsing[C]// Automatic Speech Recognition and Understanding. IEEE, 2014:120-125.

[16] Ferreira E, Jabaian B, Lefèvre F. Online adaptative zero-shot learning spoken language understanding using word-embedding[C]// IEEE International Conference on Acoustics, Speech and Signal Processing. IEEE, 2015:5321-5325.

[17] Zhu S, Chen L, Sun K, et al. Semantic parser enhancement for dialogue domain extension with little data[C]// Spoken Language Technology Workshop. IEEE, 2015:336-341.

[18] Dai W, Xue G R, Yang Q, et al. Co-clustering based classification for out-of-domain documents[C]// ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM, 2007:210-219.

[19] Yazdani M, Henderson J. A Model of Zero-Shot Learning of Spoken Language Understanding[C]// Conference on Empirical Methods in Natural Language Processing. 2015:244-249.

[20] Kusner M J, Sun Y, Kolkin N I, et al. From word embeddings to document distances[C]// International Conference on International Conference on Machine Learning. JMLR.org, 2015:957-966.

[21] See A, Liu P J, Manning C D. Get To The Point: Summarization with Pointer-Generator Networks[J]. 2017:1073-1083.