BMC Medical Informatics and Decision Making

RESEARCH Open Access

CapsTM: capsule network for Chinese medical text matching

Xiaoming Yu[1†], Yedan Shen[2†], Yuan Ni[3], Xiaowei Huang[2], Xiaolong Wang[2], Qingcai Chen[2,4] and Buzhou Tang[2,4*]

From International Conference on Health Big Data and Artificial Intelligence 2020 Guangzhou, China. 29 October - 1 November 2020

Abstract

Background: Text Matching (TM) is a fundamental task of natural language processing widely used in many application systems such as information retrieval, automatic question answering, machine translation, dialogue system, reading comprehension, etc. In recent years, a large number of deep learning neural networks have been applied to TM, and have refreshed benchmarks of TM repeatedly. Among the deep learning neural networks, convolutional neural network (CNN) is one of the most popular networks, which suffers from difficulties in dealing with small samples and keeping relative structures of features. In this paper, we propose a novel deep learning architecture based on capsule network for TM, called CapsTM, where capsule network is a new type of neural network architecture proposed to address some of the short comings of CNN and shows great potential in many tasks.

Methods: CapsTM is a five-layer neural network, including an input layer, a representation layer, an aggregation layer, a capsule layer and a prediction layer. In CapsTM, two pieces of text are first individually converted into sequences of embeddings and are further transformed by a highway network in the input layer. Then, Bidirectional Long Short-Term Memory (BiLSTM) is used to represent each piece of text and attention-based interaction matrix is used to represent interactive information of the two pieces of text in the representation layer. Subsequently, the two kinds of representations are fused together by BiLSTM in the aggregation layer, and are further represented with capsules (vectors) in the capsule layer. Finally, the prediction layer is a connected network used for classification. CapsTM is an extension of ESIM by adding a capsule layer before the prediction layer.

Results: We construct a corpus of Chinese medical question matching, which contains 36,360 question pairs. This corpus is randomly split into three parts: a training set of 32,360 question pairs, a development set of 2000 question pairs and a test set of 2000 question pairs. On this corpus, we conduct a series of experiments to evaluate the proposed CapsTM and compare it with other state-of-the-art methods. CapsTM achieves the highest F-score of 0.8666.

Conclusion: The experimental results demonstrate that CapsTM is effective for Chinese medical question matching and outperforms other state-of-the-art methods for comparison.

Keywords: Text matching, Deep learning, Capsule network, Chinese medical question matching

Background

Text matching (TM), which aims to judge whether two pieces of text, including sentences, questions, etc., are equal or match in semantic space, is a key component of

*Correspondence: tangbuzhou@gmail.com
†Xiaoming Yu and Yedan Shen contributed equally
[2] Department of Computer Science, Harbin Institute of Technology, Shenzhen, China
Full list of author information is available at the end of the article

many application systems such as information retrieval, automatic question answering, machine translation, dialogue system and reading comprehension. It is usually recognized as a classification problem where the input is a pair of pieces of text and the output is a label to indicate the two pieces of text match (denoted by 1) or not (denoted by 0).

In recent years, a large number of deep learning neural networks, such as Enhanced Sequential Inference Model (ESIM) [1], Attention-based Convolutional Neural Network (ABCNN) [2], Bilateral Multi-Perspective Matching (BIMPM) [3], Directional Self-Attention Network (DISAN) [4], Densely-connected co-attentive Recurrent Neural Network (DRCN) [5], Decomposable Attention Model (DECOMP) [6] and Bidirectional Encoder Representations from Transformers (BERT) [7], have been proposed for TM, and have achieved state-of-the-art performance on lots of benchmark datasets. Therefore, deep learning neural networks have become the mainstream machine learning methods for TM. Among these deep learning neural networks, convolutional neural network (CNN) is one of the most popular basic networks for TM. However, it suffers from difficulties in dealing with small samples and keeping relative structures of features. In this paper, we propose a novel deep learning architecture based on capsule network for TM, called CapsTM, where capsule network [8] is a new type of neural network architecture proposed to address some of the short comings of CNN. CapsTM is a five-layer neural network composed of an input layer, a representation layer, an aggregation layer, a capsule layer and a prediction layer. In this neural network, two pieces of text are first individually converted into embeddings sequences and are further transformed by a highway network in the input layer. Then, Bidirectional Long Short-Term Memory (BiLSTM) is used to represent each piece of text and attention-based interaction matrix is used to represent interactive information of the two pieces of text in the representation layer. Subsequently, the two kinds of representations are fused together by BiLSTM in the aggregation layer, and are further represented with capsules (vectors) in the capsule layer. Finally, the prediction layer is a connected network used for classification. CapsTM is an extension of ESIM by adding a capsule layer before the prediction layer. We apply CapsTM to Chinese medical question matching and achieve considerable performance. Experiments conducted on a manually annotated corpus regarding Chinese question matching show that CapsTM outperforms six state-of-the-art neural networks, that is, ESIM [1], ABCNN [2], BIMPM [3], DISAN [4], DRCN [5], DECOMP [6] and BERT [7].

The contributions of this work are: (1) investigating Chinese medical question matching comprehensively from corpus construction to methods; (2) proposing a novel method based on capsule network for Chinese medical question matching, which outperforms other state-of-the-art methods for text matching.

Related work

In recent years, deep learning methods have become mainstream for text matching, and many deep neural networks have been proposed. Most of deep neural networks are based on Siamese network [9] which aims to represent two pieces of text by the same structure. The representative neural networks are DSSM (deep structured semantic models) proposed by Huang et al. [10] and ARC-I/ARC-II proposed by Hu et al. [11]. DSSM first uses multi-layer fully connected neural network to represent two pieces of text, then computes their cosine similarity, and finally makes a prediction. ARC-I/ACR-II uses a CNN-based architecture to model both text semantic information and interactive information between two pieces of text.

By introducing new neural networks and techniques, many variants of DSSM and ARC-I/ARC-II have been proposed. Shen et al. presented CDSSM by replacing multi-layer fully connected neural network by CNN [12]. Palangi et al. developed LSTM-DSSM using LSTM instead of multi-layer fully connected neural network [13]. Yin et al. introduced attention mechanism into ARC-I/ARC-II and proposed attention-based CNN (ABCNN) [2]. Chen et al. proposed an enhanced LSTM for text inference, ESIM, which first used BiLSTM to represent text semantic information and attention matrix to represent interactive information between two pieces of text, and then fused the two kinds of information via BiLSTM and pooling. Wang et al. adopted the same architecture of ESIM with four kinds of attention matrices, called BIMPM [3]. DISAN is a light-weight neural net proposed to learn sentence embedding, based solely on a directional self-attention with temporal order encoded, followed by a multi-dimensional attention without any recurrent neural network/CNN structure [4]. DRCN is a densely-connected co-attentive recurrent neural network proposed by Seonhoon Kim et al. [5], each layer of which uses concatenated information of attentive features as well as hidden features of all the preceding recurrent layers to preserve the original and the co-attentive feature information from the bottommost word embedding layer to the uppermost recurrent layer. Ankur et al. proposed a simple neural architecture for natural language inference, called DECOMP [6], which uses attention to decompose the problem into subproblems that can be solved separately. BERT is a language representation model proposed by Jacob et al. [7], which is designed to pre-train deep bidirectional representations from unlabeled text by jointly conditioning on both left and right context in all layers. Capsule network [8] as a new type of neural network architecture proposed to address some of the short comings of CNN has showed great potential in image classification.

Methods

Formally, the task of TM is to find the most possible label y (0-not match or 1- match) of the given pair of pieces of text (s_1, s_2), where $s_1 = w_{11}w_{12}\ldots w_{1n}$ and $s_2 = w_{21}w_{22}\ldots w_{2n}$ (w_{ij}, the j-th word of s_i for $i = 1$, 2 and $j = 1, 2, \ldots, n$) are two pieces of text of the same length after preprocessing that extends all pieces of text to the same length by appending dummy tokens. Figure 1 shows the overview architecture of CapsTM, which consists of an input layer, a representation layer, an aggregation layer, a capsule layer and a prediction layer. All these layers are presented in the following sections in detail.

Input layer

For the given pair of pieces of text (s_1, s_2), the input layer first converts each piece of text into embeddings leant from large-scale unlabeled data by word2vec [14] or BERT [7], denoted by e_1 for s_1 and e_2 for s_2, and then further makes a transformation to the embeddings using highway network as follows:

$$\widehat{e_i} = \tan\ h(w_f e_i + b_f), \tag{1}$$

$$g = sigmoid\left(w_g \widehat{e_i} + b_g\right), \tag{2}$$

$$e_i' = g\widehat{e_i} + (1 - g)e_i, \tag{3}$$

where $i = 1{,}2$, w_f and w_g are weight vectors, and b_f and b_g are bias vectors.

Representation layer

In the representation layer, two types of information are extracted: (1) information of each piece of text; (2) interactive information of the two pieces of text. We utilize BiLSTM to extract the first type of information (Eq. 4) and attention-based interaction matrix to extract the second type of information (Eqs. 5–8) as follows:

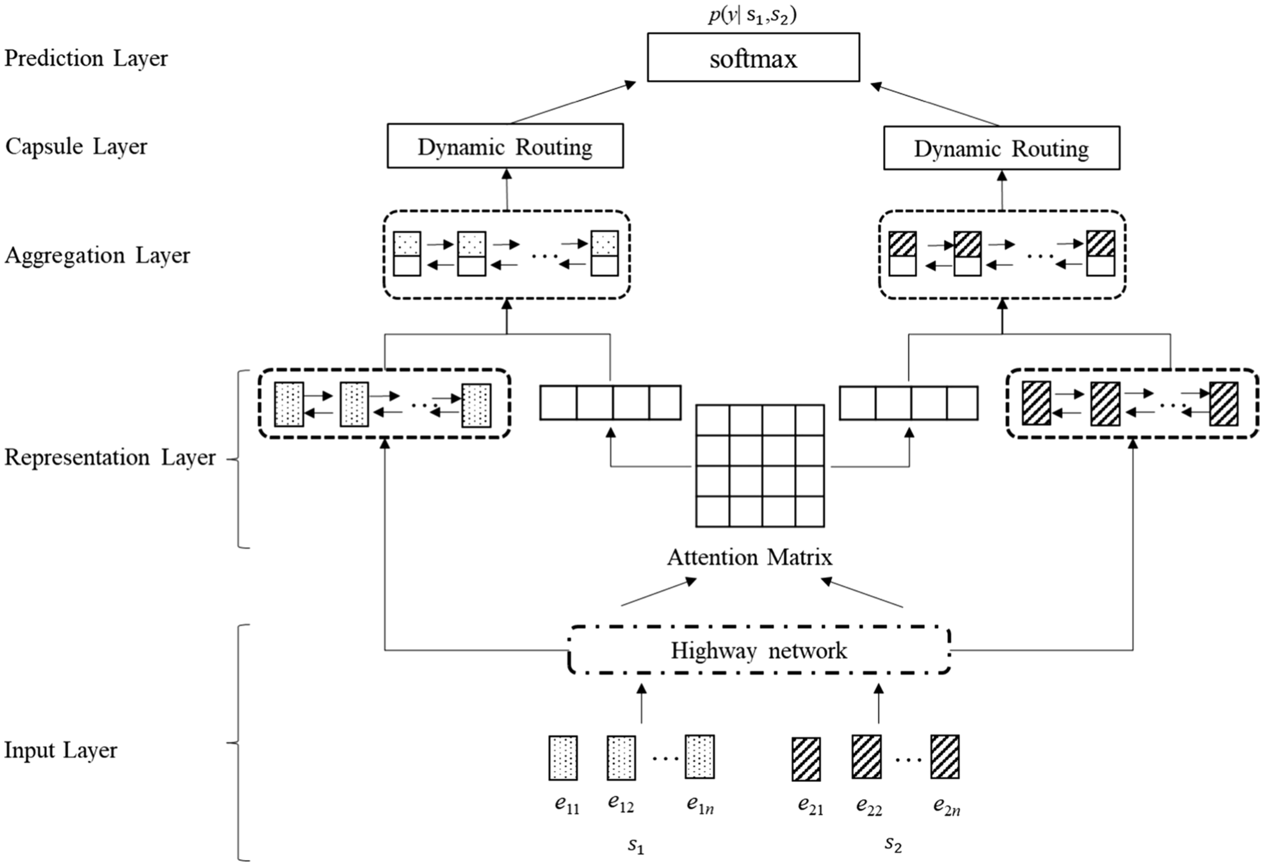


Fig. 1 Architecture of CapsTM

$$h_i = BiLSTM\left(e_i'\right), \tag{4}$$

$$sim_{ks} = e_{1k}'.e_{2s}', \tag{5}$$

$$a_{ks} = \frac{sim_{ks}}{\sum_s sim_{ks}}, \tag{6}$$

$$\widehat{a_{1k}} = \sum_s a_{ks} e_s, \tag{7}$$

$$\widehat{a_{2k}} = \sum_s a_{sk} e_s, \tag{8}$$

where h_i is the concatenation of the last hidden states from forward and backward directions of BiLSTM, $e_{ij}^{'}$ is the j-th vector of $e_i^{'}$ (i = 1,2) corresponding to w_{ij}. For the detailed information about BiLSTM, please refer to reference [1].

Finally, h_i and $\widehat{a_i}$ are concatenated to form the representation of s_i for i = 1, 2, that is $c_i = [h_i : s_i]$.

Aggregation layer

This layer is employed to aggregate the representations of the two pieces of text using BiLSTM as follows:

$$\begin{aligned} S_1^c &= BiLSTM(c_{11}, c_{12}, \ldots, c_{1n}), \\ S_2^c &= BiLSTM(c_{21}, c_{22}, \ldots, c_{2n}), \end{aligned} \tag{9}$$

where S_1^c and S_2^c are the concatenations of the last hidden states from forward and backward directions of BiLSTM for the two pieces of text.

Capsule layer

Capsule network (as shown in Fig. 2) adopts the dynamic routing algorithm (as shown in Table 1) to process the text representations from the aggregation layer iteratively.

Table 1 Dynamic routing algorithm

Dynamic routing algorithm
Initialize parameter $b_{i\|j}^0 = 0$
for r from 1 to T **do**
$c_{i\|j}^r = softmax\left(b_{i\|j}^{r-1}\right)$
$s_j^r = \sum_i c_{i\|j}^r \widehat{u_{i\|j}}$
$d_j^r = Squash\left(s_j^r\right) = \frac{\\|s_j\\|^2}{1+\\|s_j\\|} \frac{s_j}{\\|s_j\\|^2}$
$b_{i\|j}^r = b_{i\|j}^{r-1} + d_j^r u_{i\|j}$
return d_j^T

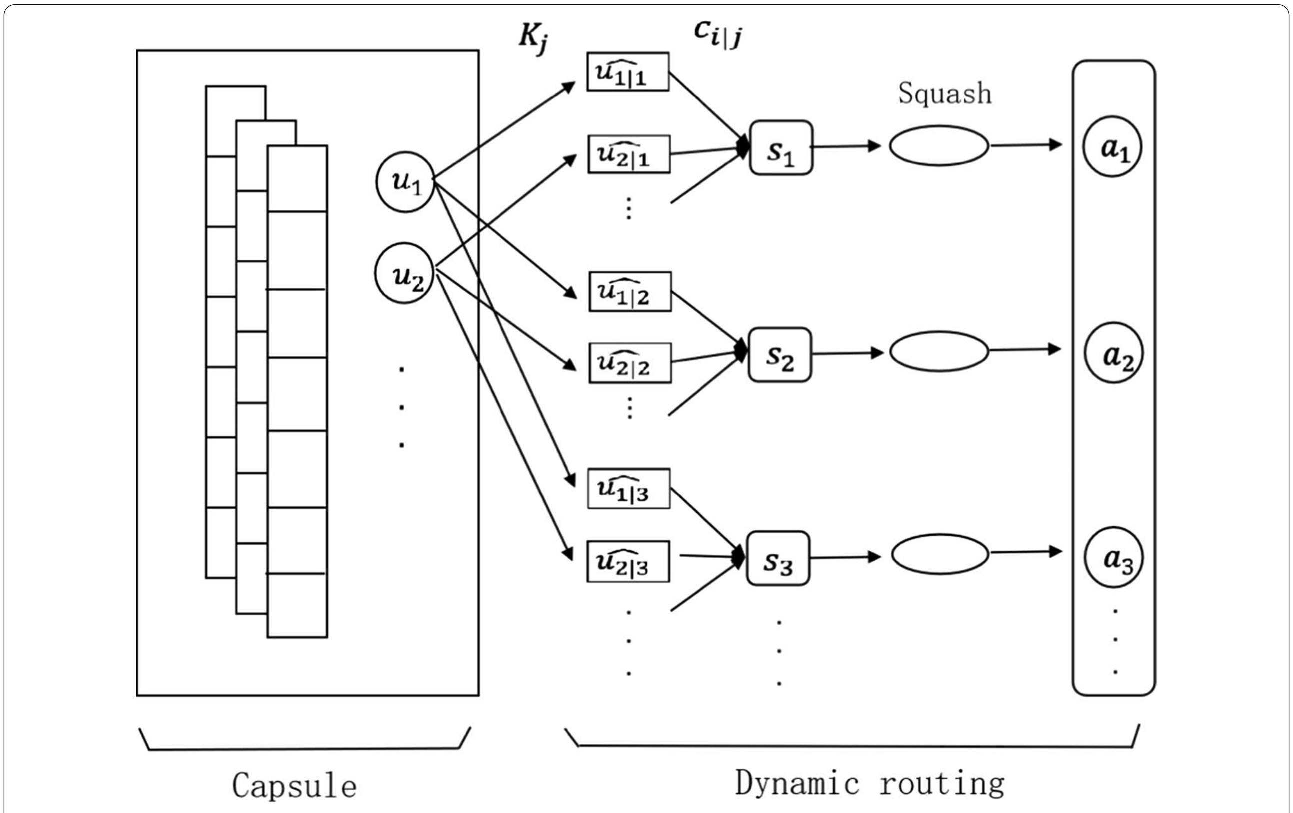


Fig. 2 Architecture of capsule network

Firstly, J d-dimensional capsule networks are initialized. For each capsule, convolution operation is applied to S_1^c and S_2^c:

$$F_{ij} = S_i^c.T_j + b_j(i = 1, 2), \tag{10}$$

where F_{ij} is the feature vector obtain from the j-th convolution kernel T_j for s_i, and b_j is the bias vector for T_j.

Suppose that there are I convolution kernels, we can obtain I-channel feature vectors for s_i:

$$U_i = [F_{i1}, F_{i2}, \ldots, F_{iI}] \tag{11}$$

The generated *feature* vectors are then input into the capsule layer, which uses vector instead of scalar to save the instanced parameters of each feature. It can not only represent the intensity of activation, but also record some details of instanced part in the input. For each channel feature vector u_i in U_1 and U_2 (i.e., u_j=F_{1j} for U_1 and F_{2j} for U_2), convolution kernel K_j (j=1, ..., k) is used to generate $u_{i|j} \in R^d$ for the j-th capsule using the following operation:

$$u_{i|j} = g(K_j.u_i + b), \tag{12}$$

where g is a *nonlinear* activation function and b is a bias vector. The k channels can be reconstituted to $\widehat{u_i}$:

$$\widehat{u_i} = \left[u_{i|1}, u_{i|2}, \ldots, u_{i|k}\right] \tag{13}$$

Then, the dynamic routing algorithm (as shown in Table 1) is applied to generate capsules of the next layer. This process actually *replaces* the pooling operation that discards location information. At the beginning of the dynamic routing algorithm, the same weight is assigned to each location $c_{i|j}^r$ like the average pooling operation. After the first iteration, the weight of each location is updated according to the similarity between $c_{i|j}^r$ and $\widehat{u_{i|j}}$. The weight of each position is stable after iterating T times.

Finally, each piece of text s_i is represented by the outputs of all capsule networks:

$$C_i = \left[d_1^T, d_2^T, \ldots, d_J^T\right] \tag{14}$$

Prediction layer

The prediction layer is a fully connected network using the sigmod activation function for prediction. Following the previous work for TM [1–3], we use the following vector as the input of the prediction layer and the cross-entropy loss as the classification loss:

$$C = [C_1, C_2, C_1 - C_2, cos(C_1, C_2)] \tag{15}$$

Experiments

Dataset

We ask two medical experts to annotate a corpus of Chinese medical question matching, which contains 36,360 question pairs. This corpus is randomly split into three parts: a training set of 32,360 question pairs, a development set of 2000 question pairs and a test set of 2000 question pairs. The distributions of positive samples and negative samples in each dataset are listed in Table 2 in detail. Here, positive samples are the medical question pairs of the same meaning or intent, while negative samples are the medical question pairs of different meaning or intent.

Experiment settings

We compare CapsTM with the following state-of-the-art deep learning neural networks: ESIM [1], ABCNN [2], BIMPM [3], DISAN [4], DRCN [5], DECOMP [6] and BERT [7]. All hyperparameters used in our experiments are shown in Table 3.

All Chinese character embeddings are pretrained by word2vec (https://code.google.com/p/word2vec/) and BERT (https://github.com/google-research/bert) on a large-scaled Chinese medical corpus. All model parameters are optimized on corresponding development sets. All the methods are implemented with Tensorflow 1.10.0,

Table 3 Hyperparameters used in our experiments

Hyperparameter	Value	Hyperparameter (BERT)	Value
Embedding size	300	Embedding size	768
#Hidden states in BiLSTM	100	#Hidden states in BiLSTM	384
#Capsules	6	#Capsules	6
#Dimension of capsules	50	#Dimension of capsules	50
#Iterations	3	#Iterations	3
Learning rate	0.001	Learning rate	0.001
Dropout rate	0.5	Dropout rate	0.9
Activation function	ReLU	Activation function	ReLU
Batch size	32	Batch size	32

Table 2 Distributions of positive samples and negative samples in the corpus used in this study

Dataset	Training		Development		Test	
	Positive	Negative	Positive	Negative	Positive	Negative
Number of smaples	12,610	19,750	801	1199	798	1202

Table 4 Comparison of CapsTM and other-state-of-the-art methods

Model	F-score	Precision	Recall
ESIM	0.8283	0.8156	0.8413
ABCNN	0.8144	0.8165	0.8123
BIMPM	0.8367	0.8190	0.8555
DISAN	0.8326	0.7800	**0.8930**
DRCN	0.8047	0.7877	0.8224
DECOMP	0.7951	0.7410	0.8577
CapsTM (word2vec)	**0.8432**	**0.8364**	0.8501
BERT	0.8646	0.8603	**0.8689**
CapsTM (BERT)	**0.8666**	**0.8655**	0.8677

Table 5 Ablation study on CapsTM

Model	F1	Precision	Recall
CapsTM (word2vec)	**0.8432**	0.8364	**0.8501**
w/o routing (max)	0.8361	**0.8503**	0.8224
w/o routing (mean)	0.8295	0.8380	0.8212
w/o attention	0.8216	0.8005	0.8438
CapsTM (BERT)	**0.8666**	0.8655	0.8677
w/o routing (max)	0.8630	0.8635	0.8624
w/o routing (mean)	0.8646	**0.8680**	0.8613
w/o attention	0.8662	0.8641	**0.8683**

and all models are trained on machines with NVIDIA GeForce GTX 1080ti GPU. The performance of models is measured by precision (P), recall (R) and F-score.

Results and discussion

As shown in Table 4 where all the highest values in each type are highlighted in bold, when using Chinese character embeddings initialized by word2vec, CapsTM(word2vec) achieves an F-score of 0.8432, and outperforms other state-of-the-art neural networks except BERT. The difference ranges from 0.65 to 3.85% in F-score. When using Chinese character embeddings initialized by BERT, the F-score of CapsTM(BERT) increases to 0.8666, which is higher than that of BERT by 0.2%. Compared to ESIM, CapsTM(word2vec) is significantly better with an improvement of 1.49% in F-score, indicating that the capsule layer added is effective.

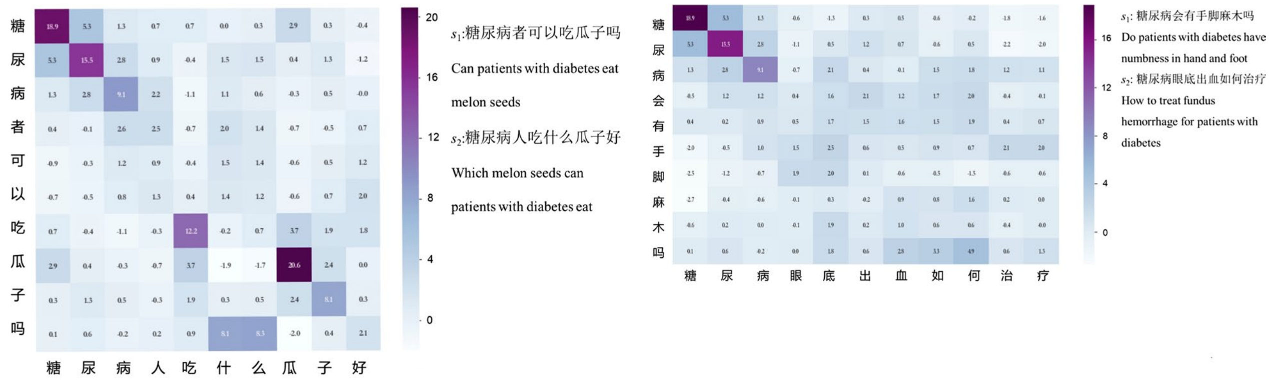
In addition to investigate the effect of the attention mechanism used in the representation layer and the dynamic routing algorithm used in the capsule layer, we conduct ablation study on CapsTM. The results are shown in Table 5, where all the highest values in each type are highlighted in bold and w/o denotes "without". When attention is removed or routing replaced by max pooling or mean pooling, the F-score of CapsTM drops. In the case of CapsTM(word2vec), the F-score decreases by at least 0.71% because of replacing routing by pooling and 2.16% caused by removing attention.

Furthermore, we check the attention matrices of some samples and find that the attention mechanism can depict semantic similarities between words in question pairs. Figure 3 gives examples of a matched question pair and an unmatched question pair, where the darker the color is, the more semantically similar the question pair is.

There are also some errors in CapsTM. These errors mainly fall into the following categories: (1) question pairs of the same type with different topics are usually wrongly classified into 1. For example, "乙肝疫苗有效期为多久 (How long is the validation period of hepatitis B vaccine)" and "乙肝表面抗体能持续多久 (How long does hepatitis B antibody last)" are wrongly classified into 1. (2) the answer to one question covers the answer to another, but they are not the same. For example, the answer to "乙肝高血压如何用药 (How to take medicine for patients with hepatitis B)" should be included in the answer to "有乙肝病要如何控制高血压 (How to control hypertension of patients with hepatitis B)", but we cannot answer the former question using the answer to the latter question directly. It is because that medication is only one type of treatments for hypertension of patients with hepatitis B. If we have a complete clinical knowledge graph, this problem may be solved. Therefore, for further improvement, we will investigate how to integrate clinical knowledge graph into existing state-of-the-art deep neural networks in the future.

Conclusion

In this paper, we propose a novel five-layer neural network based on capsule network for Chinese medical TM, called CapsTM. Experiments on a manually annotated corpus shows that CapsTM outperforms other compared



a Matched sample

b Unmatched sample

Fig. 3 Visualization samples of the attention mechanism in the representation layer

state-of-the-art neural networks. CapsTM can also have potential to be applied to TM in other domains.

Abbreviations

TM: Text Matching; CNN: Convolutional neural network; CapsTM: Capsule Network for Chinese Medical Text Matching; BiLSTM: Bidirectional Long Short-Term Memory; ESIM: Enhanced Sequential Inference Model; ABCNN: Attention-Based Convolutional Neural Network; BIMPM: Bilateral Multi-Perspective Matching; DRCN: Densely-connectied Co-attentive Recurrent Neural Network; DECOMP: Decomposable Attention Model; BERT: Bidirectional Encoder Representations from Transformers; DSSM: Deep Structured Semantic Model.

Acknowledgements

Not applicable.

About this supplement

This article has been published as part of BMC Medical Informatics and Decision Making Volume 21, Supplement 2 2021: Health Big Data and Artificial Intelligence. The full contents of the supplement are available at https://bmcmedinformdecismak.biomedcentral.com/articles/supplements/volume-21-supplement-2.

Authors' contributions

XY, YS and BT design the experiments, XY and YS write the manuscript, and YN, XH, XW, QC and BT revised the manuscript. All authors check this revised version.

Funding

This paper is supported in part by Grants: NSFCs (National Natural Science Foundations of China) (U1813215, 61876052 and 61573118), National Key Research and Development Program of China (2017YFB0802204), Special Foundation for Technology Research Program of Guangdong Province (2015B010131010), Natural Science Foundation of Guangdong Province (2019A1515011158), Strategic Emerging Industry Development Special Funds of Shenzhen (JCYJ20170307150528934 and JCYJ20180306172232154), Innovation Fund of Harbin Institute of Technology (HIT.NSRIF.2017052). We also thank PingAn Health Technology Ltd. to support this study.

Availability of data and materials

Not applicable.

Ethics approval and consent to participate

Not applicable.

Consent for publication

Not applicable.

Competing interests

The authors declare that they have no competing interests.

Author details

[1] School of Political Science and Public Management, WuHan University, Wuhan, China. [2] Department of Computer Science, Harbin Institute of Technology, Shenzhen, China. [3] PingAn Health Technology Ltd, Shenzhen, China. [4] Peng Cheng Laboratory, Shenzhen, China.

Received: 7 February 2021 Accepted: 15 February 2021
Published: 30 July 2021

References

1. Chen Q, Zhang X, Ling Z, et al. Enhanced LSTM for natural language inference. In: Proceedings of the 55th annual meeting of the association for computational linguistics; 2017. p. 1657–8.
2. Yin W, Hinrich S, Bing X, et al. ABCNN: attention-based convolutional neural network for modeling sentence pairs. In: Proceedings of the 54th association for computational linguistics; 2016. p. 259–72.
3. Wang Z, Wang H, Radu F. Bilateral multi-perspective matching for natural language sentences. In: Proceedings of the conference division and the AI journal division; 2017. p. 4144–50.
4. Shen T, Zhou T, Long G, et al. DiSAN: directional self-attention network for RNN/CNN-free language understanding. In: Proceedings of the thirty-second AAAI conference on artificial intelligence; 2018. p. 5446–55.
5. Kim S, Kang I, Kwak N, et al. Semantic sentence matching with densely-connected recurrent and co-attentive information. In: Proceedings of the thirty-three AAAI conference on artificial intelligence; 2019. p. 6586–93.
6. Parikh AP, et al. A decomposable attention model for natural language inference. In: Proceedings of the 2016 empirical methods in natural language processing; 2016. p. 2249–55.
7. Devlin J, Chang M, Lee K, et al. BERT: pre-training of deep bidirectional transformers for language understanding. In: Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics; 2019. p. 4171–86.
8. Sabour S, Frosst N, Hinton GE, et al. Dynamic routing between capsules. In: Proceedings of the 2017 neural information processing systems; 2017. p. 3856–66.
9. Neculoiu P, Versteegh M, Rotaru M, et al. Learning text similarity with siamese recurrent networks. In: Proceedings of the 2013 conference of the association for computational linguistics; 2016. p. 148–57.

10. Huang P, He X, Gao J, et al. Learning deep structured semantic models for web search using click through data. In: Proceedings of the 2013 conference on information and knowledge management, 2013: 2333–38.
11. Hu B, Lu Z, Li H, Chen Q. Convolutional neural network architectures for matching natural language sentences. In: Proceedings of the 27th international conference on neural information processing systems—volume 2 (NIPS'14). MIT Press, Cambridge, p. 2042–50.
12. Shen Y, He X, Gao J, et al. A latent semantic model with convolutional-pooling structure for information retrieval. In: Proceedings of the 2014 conference on information and knowledge management; 2014. p. 101–10.
13. Palangi H, et al. Semantic modelling with long-short-term memory for information retrieval. arXiv preprint arXiv:1412.6629 (2014).
14. Mikolov T, Chen K, Corrado GS, et al. Efficient estimation of word representations in vector space. In: Proceedings of the 2013 international conference on learning representations; 2013.

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.