Yuan-ping Nie, Yi Han, Jiu-ming Huang, Bo Jiao, Ai-ping Li, 2017. Attention-based encoder-decoder model for answer selection in question answering. *Frontiers of Information Technology & Electronic Engineering*, **18**(4): 535-544.
http://dx.doi.org/10.1631/FITEE.1601232

Attention based encoder-decoder model for answer selection in question answering

Key words: Question answering; Answer selection; Attention; Deep learning

Contact: Yuan-ping Nie
E-mail: yuanpingnie@nudt.edu.cn
ORCID: http://orcid.org/0000-0002-8351-4108

Introduction

- One of the key challenges for question answering is to bridge the lexical gap between questions and answers

- Lexical gap becomes a major barrier to prevent traditional IR models, such as bm25.

- Using a deep neural network-based machine translation model to bridge the lexical gap between a question and its answer.

- To address the noise-bearing problem, we introduce a attention based encoder-decoder model for answer selection.

Encoder-Decoder Model

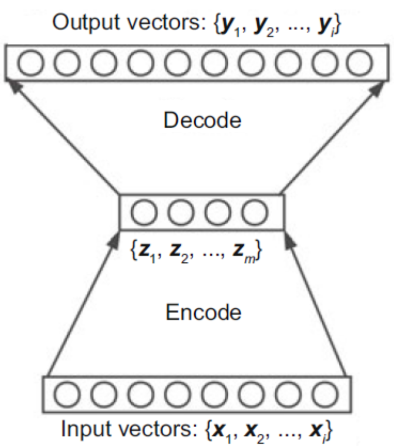

Fig. 1 The structure of an encoder-decoder

An encoder maps an input sequence into **a fixed length latent representation via a deterministic mapping function**

$$r_l = f_e(h_1, h_2, \cdots, h_i),\ h_t = q(x_t, h_{t-1})$$

The decoder is often trained to predict the next word given the latent representation ***r***l and all the previously predicted words

$$p(y) = \prod_{t=1}^{T} p(y_t | \{y_1, \cdots, y_{t-1}\}, c),$$

$$r_l = f_e(x_i) = S_e(W_e x_i + b_e),$$

Convolutional Filter

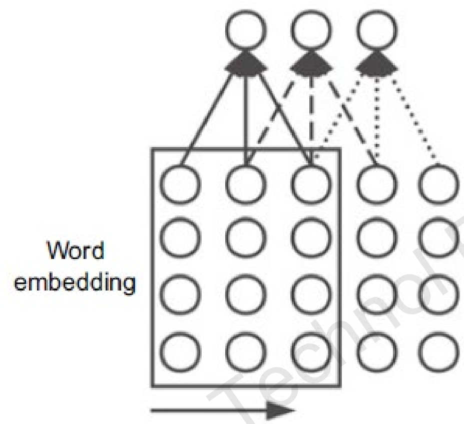

Fig. 5 Convolutional fitler

The convolution operation * between two vectors a and f can be defined as:

$$c_i = (a * f)_i = \sum_{k=i}^{i+m-1} a_k f_k$$

Our model with step attention mechanism

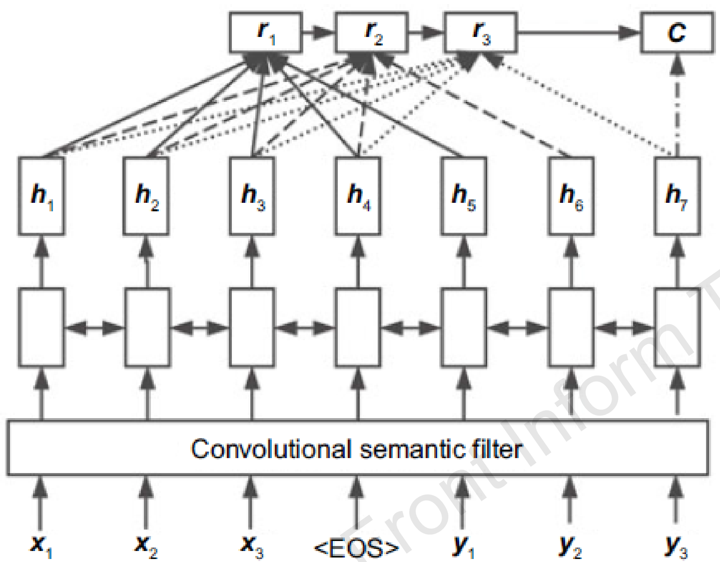

Fig. 7 The step attention model

the attention mechanism will produce a weighted representation ***r*** of the candidate passages and a vector ***s*** of the normalized attention weights via:

$$\boldsymbol{m}(t) = \tanh(\boldsymbol{W}_{\mathrm{y}}\boldsymbol{y}_{\mathrm{p}} + \boldsymbol{W}_{\mathrm{r}}\boldsymbol{r}(t-1) + \boldsymbol{W}_{\mathrm{u}}\boldsymbol{y}_{d}(t)), \tag{14}$$

$$\boldsymbol{s}(t) = \mathrm{softmax}(\boldsymbol{w}^{\mathrm{T}}\boldsymbol{m}(t)), \tag{15}$$

$$\boldsymbol{r}(t) = \boldsymbol{y}_{d}\boldsymbol{s} + \tanh(\boldsymbol{W}_{\mathrm{r}}\boldsymbol{r}(t-1)), \tag{16}$$

the last output vector ***C*** is:

$$C = \tanh(\boldsymbol{W}_{\mathrm{p}}\boldsymbol{r} + \boldsymbol{W}_{\mathrm{u}}\boldsymbol{h}_{\mathrm{n}})$$

Experimental results (1)

Dataset：

Table 1 Summary of the TREC QA datasets for answer reranking

Dataset	Number of questions	Number of QA pairs	Correctness of dataset
Train-ALL	1229	53 417	12.0%
Train	94	4718	7.4%
Dev	82	1148	19.3%
Test	100	1517	18.7%

Evaluation metric：

$$\mathrm{MAP} = \frac{1}{n}\sum_{q=1}^{n} \mathrm{avg}(P(q))$$

$$\mathrm{MRR} = \frac{1}{n}\sum_{q=1}^{n} \frac{1}{\mathrm{rank}(q)}$$

Results：

Table 2 The results using the TREC-QA dataset in Train-ALL

Model	MAP	MRR
Wang *et al.* (2007)	0.6029	0.6852
Wang and Manning (2010)	0.5951	0.6951
Yao *et al.* (2013a)	0.6307	0.7477
BDT (Yih *et al.*, 2013)	0.6940	0.7894
LCLR (Yih *et al.*, 2013)	0.7092	0.7700
Unigram (Yu *et al.*, 2014)	0.5387	0.6284
Bigram (Yu *et al.*, 2014)	0.5693	0.6613
Three-layer BLSTM (Wang and Nyberg, 2015)	0.5928	0.6721
BLSTM+BM25 (Wang and Nyberg, 2015)	0.7134	0.7913
RNN	0.6198	0.6831
LSTM	0.6428	0.7115
Our model without attention	**0.6871**	**0.7350**
Our model with step attention	**0.7261**	**0.8018**

Experimental results (2)

Dataset： TREC LiveQA 2015 track 1087 questions and each question 20 candidate answers

Evaluation metric：

The answer score is set to 4 levels, shown below:

1. Bad: contains no useful information for the question.
2. Fair: marginally useful information.
3. Good: partially answers the question.
4. Excellent: a significant amount of useful information, fully answers the question.

Results：

Table 3 The performance in the LiveQA track

Model	Average score	Succ@2+	Succ@4+
CMUOAQA	1.081	0.532	0.190
Ecnucs	0.677	0.367	0.086
Our pre-model	0.670	0.353	0.107
Monash	0.666	0.364	0.082
Yahoo	0.626	0.320	0.095
Average	0.467	0.262	0.060
Our model	**1.103**	**0.546**	**0.204**

Succ@i+ ($i = 1, 2, 3, 4$) means the number of questions with scores larger than i divided by the number of all the questions

Conclusions

- The proposed model employs a bidirectional LSTM based encoder-decoder architecture, which can effectively bridge the lexical gap between questions and answers.
- The results show that our model is more effective compared to the baseline approaches.