Which is the Effective Way for Gaokao: Information Retrieval or Neural Network?

By Shangmin Guo, Xiangrong, Shizhu He, Kang Liu and Jun Zhao

Reference: EACL 2017 Paper

Abstract

National Higher Education Entrance Examination, which is commonly knowns as Gaokao, is designed to be difficult enough to distinguish excellent high school students in China. The paper described Gaokao History Multiple Choice Questions(GKHMC) and proposed three approaches to address them. Approaches include **Information Retrieval(IR)**, **Neural Network(NN)** and the **combination of IR and NN**. Results show that IR performs better at Entity Questions(EQs), while NN performs better at Sentence Questions(SQs). The combination method achieves state-of-art performance, showing the necessity to apply hybrid method when encountering real-world scenarios.

Motivation: Difficult History Multiple Choice Questions

Answering real world questions in various subjects it increasingly getting attentions. An ambitious Project Halo was proposed to create a “digital” Aristotle which can encompass most of the worlds’ scientific knowledge as well as solve hard problems. Important trials include solving mathematic and chemistry questions. In terms of the history questions, there are some NLP attempts for yes-no questions: determining the correctness of the original position (Kanayama et al., 2012) and recognizing textual entailment between a description in Wikipedia and each options(Miyao et al., 2012). Nevertheless, none of these approaches can solve difficult history multiple choice questions as shown in Figure 1, which require a huge amount of background knowledge.

After the World War II, U.S. and Soviet Union are fighting against each other in politics, economics and military. To promote the development of economics in Socialist Countries, Soviet Union establish The Council for Mutual Economic Assistance. This is against

A. Truman Doctrine B. Marshall Plan

C. NATO D. Federal Republic of Germany

Entity Question

From Qin and Han Dynasties to Ming Dynasty, businessmen are always at the bottom of hierarchy. One reason for this is that the ruling class thought the businessmen

A. are not engaged in production B. do not respect Confucianism

C. do not respect the clan D. do not pay tax

Sentence Question

Figure 1: Examples of history questions. EQ means all of the candidates are entities, which SQ means candidates are parts of the sentence.

This paper is not the first to solve Gaokao questions, but the former approaches based on information retrieval did not fit well and suffer from limited knowledge resources in their systems. Therefore, works introduced in this paper are mainly focus on solving difficult GKHMC and proposing new approaches to improve accuracy.

Datasets: Questions and Resources

All of the questions are from Gaokao all over the country in 2011 - 2015. Questions with graphs are filtered out since solving them requires techniques beyond NLP. The remaining questions are manually tagged as EQs or SQs. The question dataset is in XML format and available here. Numbers of different kinds of collected multiple-choice questions are listed as below.

Question Type	Candidate Type	Count
EQ	Entity Question	160
SQ	Sentence Question	584
ALL	Whatever	744

Wide diversity of resources including Baidu Encyclopedia, textbooks and over 50,000 practice questions are also collected, which is in XML format as well and available here.

Approaches and Results: IR, NN and Combination

Information Retrieval (IR)

Since GKHMC questions require finding the most relevant candidate to the question stem from 4 choices, IR approach (source code) is applicable by following the pipeline below:

1. Use **Naive Bayes classifier** to classify questions.

   - Features include length, entity number and verb number of candidates.
   - Do 10-folder cross validation on question dataset.

2. Calculate **relevance scores** for each candidate and combine them with **specific weights** (3 different method with 7 score functions on different indices are provided for the calculation).

   - **Lexical Matching Score**: $Score_{lexical}$, calculated as below. ($score_{top_i}$ is calculated by Lucene's TFIDFSimilarity function, denoting the score of the top i-th returned documents.)

   $$Score_{lexical}(candidate_k) = \sum_{i=1}^{3}(score_{top_i})$$

   - **Entity Co-Occurrence Score**: $Score_{co}$, calculated by normalized google distance.
   - **Page Link Score**: $Score_{link}(candidate_k)$, inspired by PageRank algorithm, calculated as below., Where $e_i \in E_{stem}, e_j \in E_{candidate_k}$.

   $$Score_{link}(candidate_k) = max(Link(e_i, e_j))$$

   - **Training weights and loss function** For each candidate, the score can be calculated as:

   $$score_{candidate_k} = \sum_{i=1}^{7} w_i * f_i(candidate_k)$$

   Then normalize the scores of all candidates:

   $$score_k = \frac{score_{candidate_k}}{\sum_{i=1}^{4}(score_{candidate_i})}$$

   The loss function of it is:

   $$loss_{questions} = -log(1 - score_n)$$

> As all operations are derivable, gradient descent algorithm can be used to train weights.
>
> 3. Candidate with highest score will be chosen as right answer.

Result: Accuracy of EQs and SQs with corresponding best weights. IR works better over EQs than SQs.

-	EQ-W_{EQ}	SQ-W_{SQ}
Accuracy	49.38%	28.60%

Neural Network (NN)

Permanent-Provisional Memory Network(PPMN) is introduced in this paper as NN approach, which is designed to handle the joint inference between background knowledge and question stems in GKHMC. The diagram of PPMN(source code) is shown in Figure 2.

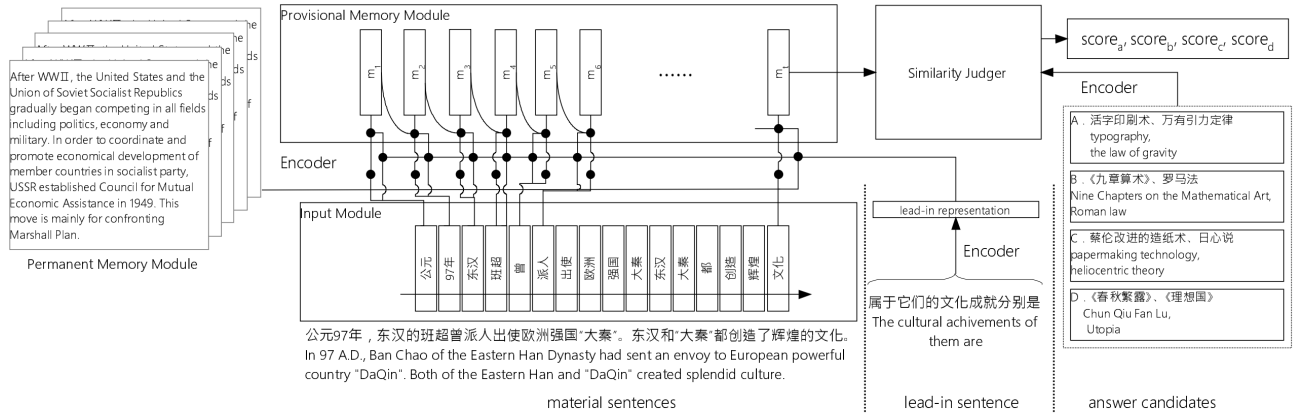

Figure 2: Diagram of PPMN

PPMN is composed of the following 5 modules:

> 1. **Permanent Memory Module** (Knowledge Base): A constant matrix composed of concatenation of representation vectors of sentences $[k_1, k_2, ..., k_K]$, where K is the scale of permanent memory. Only take syllabus of all history courses ($K = 198$) in terms of time complexity here.
>
> 2. **Provisional Memory Module**: First inquires current word of background sentences in Permanent Memory Module, then use an attention vector to decide how to adjust itself.
>
> 3. **Input Module**: takes same weight matrix in sentence encoder and calculates the hidden states of each word sequentially.

4. **Similarity Judger**

   - Input ($[m_K; a]$): the concatenation of the output from provisional memory and representation of the answer candidate.
   - Output: $score$ of input (using a classifier based on logistic regression).

$$\hat{p} = \sigma(W^l[m_k; a] + b^l), score = softmax(\hat{p}) \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

5. **Sentence Encoder**: Gated Recurrent Unit - $GRU(w_t, h_{t-1})$, where w_t is extract from a word embedding matrix W_e initialized by word2vec. A negative log-likelihood loss function is introduced as $L = -log(\hat{p} \begin{bmatrix} 0 \\ 1 \end{bmatrix})$. In this paper, AdaDelta is introduced to minimize L . Back propagation through time is introduced as well to optimize the calculation of intermediate results.

Result: In comparisons among different neural network models (RNN, LSTM, GRU, MemNN, DMN, PPMN, Random), PPMN has the best accuracies in EQs, SQs and ALL. The results are listed as below.

Model	EQs	SQs	All
RNN	36.25%	29.74%	31.18%
LSTM	40.63%	40.41%	40.46%
GRU	40.63%	40.24%	40.32%
MemNN	43.75%	36.13%	37.77%
DMN	44.38%	45.38%	45.16%
PPMN	45.63%	45.72%	45.70%
Random	25.00%	25.00%	25.00%

Combination IR and NN Approach

It is obvious that IR and NN approaches are complementary to some extent, which is intuitively as well. In EQs, information given by question stems is usually the description of the key entity, which is the reason why correct answer has the highest relevance score. However, in SQs, the key entity does not appear in any candidate, which means inference is needed. Therefore, though IR works well in EQs, it is not sufficient to find the correct choice in SQs, while NN works better in SQs.

Considering that (1) some of EQs may be more suitable to be handled as SQs and (2) both character and word embedding are more sufficient to cover the lexical meaning, a hybrid approach is proposed. IR and NN approaches are simply combined via s weights matrix as below, where W_i^c denotes the i-th row of W_c.

$$score_{EQ} = W_1^c \begin{bmatrix} score_{IR} \\ score_{NN} \end{bmatrix}, score_{SQ} = W_2^c \begin{bmatrix} score_{IR} \\ score_{NN} \end{bmatrix}$$

The performance of combined model and its comparison to IR and NN approaches are illustrated in Figure 3.

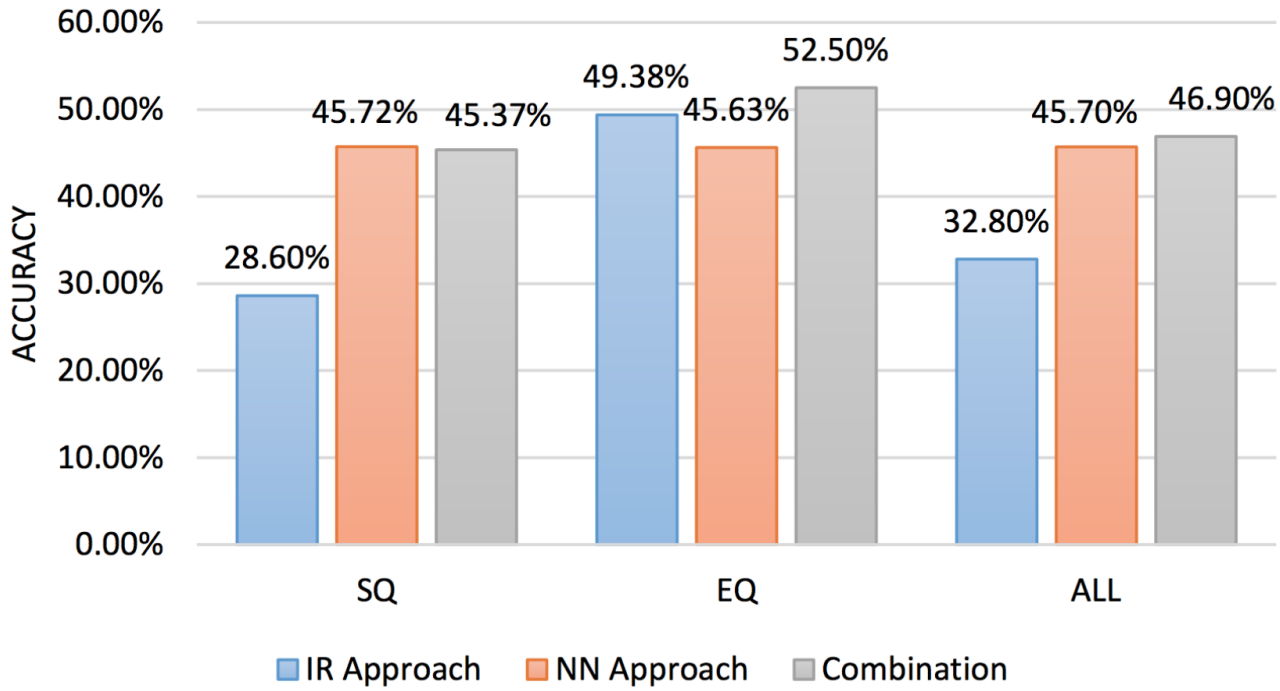

Figure 3: Result of different approaches: IR, NN and Combination

Conclusion

The paper details the GKHMC, presents different approaches to address them and compares their performances. According to the results, IR approach is more suitable for EQa while NN approach is more suitable for SQs. The combination of IR and NN has a state-of-the-art performance on GKHMC, pointing out that hybrid methods may be a better choice in real world scenarios.