Solving MCTest with semantic textual similarity and matching rules??

A. Vlachos, T. Brown, N. Greco, G. Mocanu and E. Smith
Computer Science Department
University College London
{a.vlachos,t.brown,n.greco,g.mocanu,e.smith}@cs.ucl.ac.uk

Abstract

MCTest is a recently developed test for evaluating Machine Comprehension. Our approach to the task is through textual similarity and shallow methods. We build a simple bag-of-words baseline and we enhance it through additional pre-processing (i.e. coreference resolution, hypernym). We build a set of features and train a logistic classifier that we use to score multiple-choice answers. Finally we show how the introduction of simple matching rules system outperform current results on the MCTest.

1 Introduction

2 Previous work

Yu et al. (2014) uses bag-of-words as a baseline combining this shallow method with convolutional neural networks to capture complex semantics of a sentence.

Hirschman et al. (1999) proposed DEEPREAD with shallow methods for story comprehension and combine them with some heuristics to answer questions 'who / what / when".

3 Task description

The task has been presented in (Richardson et al., 2013)

4 Baseline

We propose a baseline system, a simpler version than the originally proposed by Richardson et al. (2013) only using simple lexical feature. This system matches a bag of words constructed from the question and a candidate answer with each sentence in the story. The question-answer pair with the most words in common is considered the best candidate answer. Normalization are applied such as removing stop words stemming to remove affixes from words.

Definitions

Passage P, P_i sentence i in passage P, set of words in question Q, set of words in candidate answer $A_{1..4}$ and set of stop words X.

Algorithm 1 Sentence level bag-of-words

```
for i=1 to 4 do
   S = A_i ∪ Q
end for
return sw_{1...4}
```

Figure 1: Lexical-based baseline algorithm

This semantic overlap approach treats the problem as a textual similarity task and would perform best when the pair is a subset of a sentence. However, this strategy ignores predicate-argument structure and can easily fail in the presence of quantifiers, negations or synonyms. Work on story comprehension using bag-of-words has a long history, Hirschman et al. (1999) proposed DEEPREAD and showed how such systems with some heuristics can achieve high accuracy especially on questions with "who / what / when", which is most part of our questions.

We will build upon this baseline in Section 6.1 and combine it with heuristics in Section 6.3. Results for MC160 and MC500 are shown in Table **??** and Table **??**. The baseline has been authored without seeing both the test sets.

5 Preprocessing

Matching question-answer pairs with the story can be significantly improved by homogenising the format of all stories and question-answer strings. Our matching algorithms operate on raw textual

tokens, which are lemmatised and stripped of all extraneous function words; however, the raw format was generated on-demand, rather than during the pre-processing stage, and we retained the deep grammatical structure of the text in order to dynamically alter the format based on certain question conditions.

We focused on four pre-processing stages that will be discussed in this section: syntactic parsing and coreference resolution, hypernym annotation, sentence selection and combining question and answer.

5.1 Syntactic Analysis

The initial pre-processing stage used the Stanford Parser REFERENCE and Stanford Dependencies REFERENCE to obtain phrase-structure and dependency trees for each story, along with its questions and answers. We also made use of the lemmatization and part-of-speech tagging provided by this system. This toolkit performed well on the given data, due to the intentional linguistic simplicity of the stories. Of the few inconsistencies, the majority were due to incorrect recognition of invented brand names (e.g. Cookies n Crme and Friendly-Os) and the inability to categorize some subordinate dependencies. However, such cases were rare, and the errors introduced at this pre-processing stage were negligible in the final results.

Following this, coreference information was extracted using the Stanford CoreNLP package REFERENCE. The passage text was parsed independently of the question and answer strings, so all coreference chains were local to the story itself. Resolving links between a question and its answer strings proved to be detrimental to performance. We used an out-of-the-box configuration of the coreference rules, as this was deemed to perform adequately on the simplisitic format of the given stories. Some errors emerged when resolving coreferences involving multiple entities, however, but correcting these errors is beyond the scope of this work.

5.2 Hypernym

5.3 Sentence selection

The current baseline chooses the best candidate on the basis of how much it matches one sentence in the corpus. This is a clear disadvantage give that the MCTest has questions whose answer may be in contained in multiple sentences; at the same time, running matching the bag of words between the text entire text with the question-answer pair would give poor results since ().

To improve on answering questions using multiple sentences, we propose to run the bag-of-words on the n most relevant sentences for answering the question. This problem reduces to an information retrieval task. The idea of retrieve the sentences with the highest relevance for question answering has been already proved successful in the literature (Harabagiu et al., 2003; **?**). In order to rank the sentences in the story we will have a scoring function $selectScore(query, document)$, that given a $query$ and a $document$, it will

We will need a scoring function that will score each sentence

For the purpose of this paper we are going to re-use our bag-of-words combined with hypernym and coreference to rank the most relevant sentences (in details in Section 6.2)

(Some is q, some is qa) By tuning on both the MC160 and MC500 training set, $n = 3$ gets the highest results.

Reduces the task to an information retrieval how to score the best answers

5.4 Question answer combiner

6 Strategy

6.1 Bag of words

several features BOWNN BOW Complement Bow ALL

The great 6

6.2 Scoring function

equations of deep selection

scoring question answer pairs the choice of the SVM and gridsearch

6.3 Rule based system

7 Experiments and results

8 Evaluation of strategies

9 Future work and conclusion

Semantic overlap is typically a symmetric relation while textual entailment is clearly not, this is a serious limitation of our baseline and the systems built on top. However, the great results show how really simple methods can achieve great results on the MCTest.

Wordnet for synonym as well in addition to hypernym

Acknowledgments

Thanks to

References

Sanda M. Harabagiu, Steven J. Maiorano, and Marius Pasca. 2003. Open-domain textual question answering techniques. *Natural Language Engineering*, 9(3):231–267.

Lynette Hirschman, Marc Light, Eric Breck, and John D. Burger. 1999. Deep read: A reading comprehension system. In *ACL*, pages 325–332.

Matthew Richardson, Christopher J. C. Burges, and Erin Renshaw. 2013. Mctest: A challenge dataset for the open-domain machine comprehension of text. In *EMNLP-SIGDAT*, pages 193–203.

Lei Yu, Karl Moritz Hermann, Phil Blunsom, and Stephen Pulman. 2014. Deep learning for answer sentence selection. *CoRR*, abs/1412.1632.