An ensemble model for the Stanford Question Answering Dataset (SQuAD)
Report of summer research at CMU's LTI

María Fernanda Mora Alba
Department of Computer Science
Instituto Tecnológico Autónomo de México
México, Distrito Federal
Email: maria.mora@itam.mx

***Abstract*—In this paper we propose a pipelined methodology that uses an ensemble approach to solve a particular instance of the Question Answering's problem called Machine Reading Comprehension. The dataset used to test the proposed model was recently published by [Rajpurkar et al.2016] as a new benchmark for this endeavour. The approach proposed in this paper is modular. It is composed of four pseudo independent modules, sentence ranking, answer extraction, model averaging and evaluation. Its modularity gives it generality since multiple models can be implemented at each stage.**

***Index Terms*—machine reading comprehension, question-answer, question analysis.**

I. Introduction

Machine reading comprehension is the ability to read and understand a natural language documents at a sufficient level where a machine system is capable of answering questions based on the original text. Teaching machines to do this remains an puzzling challenge. First, NLP and QA models are hard on its own and are built and trained using the available data. Second, machine comprehension is commonly evaluated by how good the proposed model answers to questions about a text; but for this evaluation to be meaningful, adequate datasets are crucial. Accordingly, high-quality, real and large datasets play a crucial role to make progress on machine comprehension. The problem is that already existent datasets suffer from shortcomings such as being too small or semi-synthetic. The Stanford Question and Answering Dataset (SQuAD)[1] [Rajpurkar et al.2016] was built in mind to overcome these deficiencies. SQuAD is formed by 100,000+ question-answer pairs based on 500+ Wikipedia articles. The questions and answers were annotated through a mechanical turk. The questions are designed to bring answers which can be defined as a *span*, or segment of the corresponding passage or context.

[Rajpurkar et al.2016] proposed a baseline a logistic regression model over the SQuAD dataset that achieves an F1 score of 51% which outperformed the 20% random baseline but still remains below the human performance of 86.8%.

[1]https://stanford-qa.com/

II. Problem definition

SQuAD provides a challenging dataset for building, testing and evaluating machine comprehension models and systems for three main reasons:

- *No candidate answers are provided:* instead of a predefined list of answer choices such as [Richardson et al.2013b], in SQuAD all the possible spans in the passages are candidate answers.
- *A correct answer to a question can be any sequence of tokens from the given text:* instead of having, for example, a cloze style dataset, in which the answer is a single token, in SQuAD the answers can be composed of sequences of tokens. These sequences can be quite similar, thus making more difficult the recognition of the correct answer. The evaluation of the models is performed with this criteria, so it is more difficult to achieve a good performance.
- *QA in SQuAD were created by humans, hence more realistic*: unlike other datasets such as [Hermann et al.2015], whose questions and answers were created automatically and synthetically, SQuAD's questions and answers were created by humans through crowdsourcing.

An example of a passage, question and answer can be appreciated in the following text:

Passage: It is in the great churches and cathedrals and in a number of civic buildings that the Gothic style was expressed most powerfully, its characteristics lending themselves to appeals to the emotions, whether springing from faith or from civic pride.
Question: What is an example of where the Gothic style is expressed most strongly?

Answer: churches and cathedrals

III. Related Work

Recently there has been two major efforts towards the advance of machine reading comprehension: development of models and creation of datasets.

A. Datasets

We can classify the reading comprehension datasets by how they generate the questions:

- *Cloze style:* these datasets are created by removing certain words from chunks of texts. The reading comprehension ability is assessed by how well the model is able to replace the missing words. For example, [Hermann et al.2015] created a dataset of this type using short summaries of the news articles from CNN and Daily Mail. Another example is the Children's Book Test dataset by [Hill et al.2015] in which 20 sentences from a children's story are used to predict the missing word in the 21th sentence. It is worthwhile to point out that these type of datasets does not have 'real' or factoid questions.
- *Human annotators:* these datasets are created totally or partially by humans making them more realistic. For example, [Richardson et al.2013a] constructed through crowdsourcing the MCTest dataset consisting of short fictional stories, questions and candidate answers. Although the authors claim that its approach is scalable, the dataset is totally generated by humans, thus making real scalability prohibitive. In fact the authors only generated 500 stories and 2000 questions. Another weakness of this dataset is that the stories were written to be understandable by a child in grade school, potentially preventing a model from really performing natural language comprehension.

The first approach is scalable but synthetic, and the second approach is more realistic but not scalable. This is where SQuAD stands creating a real and reasonably scalable dataset.

B. Models

[Hermann et al.2015] used recurrent neural networks together with attention based mechanisms to predict a single token. But the answers in SQuAD contain multiple tokens, so this approach is infeasible for the problem.

End-to-end training with Sequence-to-Sequence neural models has been successfully applied to many NLP tasks [Yang et al.2016] and it is possible to generate multiple-token answers. Although SQuAD is larger than most currently available reading comprehension datasets, sequence-to-sequence models require datasets with a greater scale than the one provided by SQuAD. Memory networks [Weston et al.2014] is an alternative approach but these models suffer from lack of scalability on large datasets.

[Rajpurkar et al.2016] proposed the first model over SQuAD but it is below human's performance by more than 35 percentual points (F1 score of 51% vs 86.8 %). The proposed model is a logistic regression built with handcrafted features. [Rajpurkar et al.2016] found that the model performance is very sensitive to the following features:

- *Lexicalized and dependency tree path features:* these are the features that contribute in greater proportion to the performance of the model.
- *Answer types:* the model performs better on answers regarding number and entities, while human performance is more uniform.
- *Syntactic divergence between the question and the sentence containing the answer:* the performance of the model decreases with with increasing divergence while human's performance remains almost constant.

In the following, it is proposed a model capable of performing reading comprehension over SQuAD that tries to outperform the current state of the art.

IV. Methodology

A modular approach was adopted to circumvent the problematic raised by the lack of enough data while incorporating specificity and generality. The problem was tackled through a top-down approach formed by two main phases: *sentence ranking*, *answer extraction*, followed by a *learning* phase to come out with a final answer that is evaluated in the *evaluation* phase. The complete high-level pipeline can be visualized in the following Fig. 1.

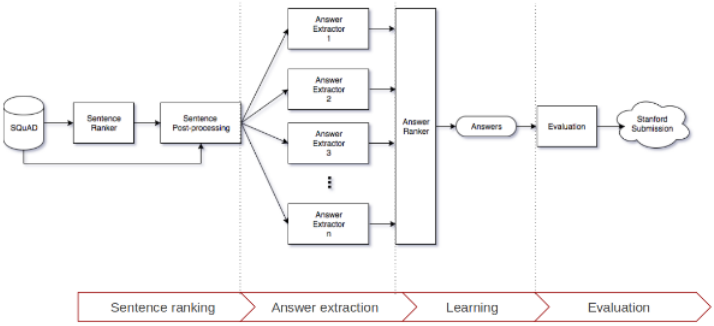

Fig. 1. High-level proposed pipeline

A. Sentence ranking

The idea of *sentence ranking* is to exploit lexical and syntactical similarities between the question and the answer passage to obtain the sentence with the highest likelihood of being the answer. The sentences of the passages are ranked according to a specific question. In this stage the model attempts to maximize the probability of the sentence that includes the answer.

For this phase the following alternative models were tried so far:

- BM25 and Jaccard similarity that only consider lexical similarity, under the bag of words approach.
- Convolutional neural network model for reranking pairs of short texts from [Severyn and Moschitti2015] that performs a two-stage learning: learn an optimal vector representation of question and passage and then learn a similarity function between question and passage vectors.

B. Answer extraction

The idea of *answer extraction* is that given a candidate answer sentence from the *sentence ranking* phase, the tokens that make the answer are extracted. For this phase we have used so far features that extract lexical, syntactical and semantical structure of sentence, question and answer with the aim of training a classifier (random forest so far). For now, the features for *each word* considered are the following: *indicator of right neighbor in question, right neighbor NER, right neighbor POS, word Animacy, word Gender, word NER, word Number, word POS, word type, dependency with father, indicator father in question, father NER, father POS, indicator of word in question, indicator of left neighbor in question, left neighbor NER, left neighbor POS, question type (what, which, etc).*

For example, for the word **it** we will have the following vector of features: (False, u'It', u'PRP', u'O', False, 'whom', '', '', '', '', u'is', u'VBZ', u'O', False, False, u'replica', u'NN', u'O', u'nsubj', False, False, u'INANIMATE', u'SINGULAR', u'NEUTRAL', u'PRONOMINAL').

V. Results

Before diving into the results of the *sentence ranking* and *answer extraction* phases we present an analysis over SQuAD to gain a better understanding of the dataset.

A. Dataset analysis

1) General statistics: SQuAD is formed by articles containing passages, each with 5 questions with its corresponding answers. The *complete dataset* contains 536 Wikipedia articles with 108K QA pairs. This dataset is further divided in training, dev and test sets; test set is not publicly available, this test is used by Stanford to evaluate a submitted model. The *training dataset* contains 378 Wikipedia articles with approximately 42 passages per article, 5 questions per passage and 1 answer per question yielding a total of 80K question-answer observations. This is the dataset we used to train the models.

As of the vocabulary size in number of words:

- Passages: approx. 88K (98% without stop words)
- Questions: approx. 1K (93% without stop words)
- Answers: approx. 0.5K (93% without stop words)

The input for a model is a question and a context and the output is a proposed answer (sequence of tokens). The models are evaluated using two metrics: exact match and the F1 score.

The analysis showed the following findings:

- >99% of the questions are factoid questions and that >50% of the questions are *what* questions
- Questions length is similar; answers to why and other questions show length variation. See Fig. 2.
- Questions are larger (in number of words than answers); why questions have the largest answers but represent <5%. See Fig. 2.

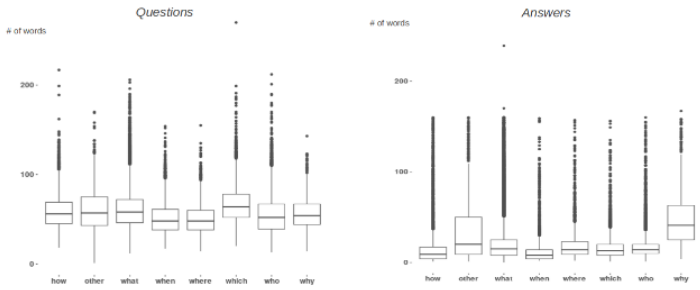

Fig. 2. Box-plot of questions and answers length

2) Lexical analysis:

- There exists a lexical similarity (cosine similarity) 0.3-0.4 between passages of the same article. This similarity is independent of the length of the passage. See Fig. 3.
- We performed LDA analysis to understand the underlyting topics. We varied the number of words and the number of topics and we found the following persistent topics: history, government, nation-state, sports and art.

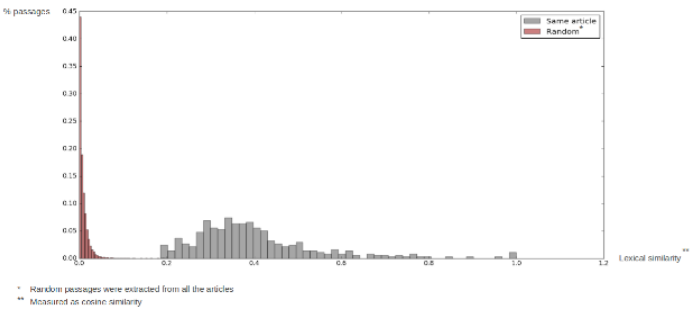

Fig. 3. Lexical similarity between passages of the same article

3) Syntactic and semantic analysis: To understand the syntactic -and possibly semantic- relationships we performed **embeddings** at three aggregation levels: word, sentence and paragraph. The findings are the following:

- **Word:** In order to extract the semantic and syntactical structure of the text, we represented each word as a continuously distributed vector by trying different methods [Mikolov and Dean2013], [Liu et al.2015], [Pennington et al.2014] considering linear, syntactic, topical and ensemble embeddings. The best performing embeddings overall were obtained with GloVe. See Fig. 4 for a visualization of GloVe embeddings wirh window size of 15 and vector size of 100. Cluster 2 was able to identify dates, numbers and months. Cluster 4 sports and music. Cluster 5 identified politics, kings and genders.

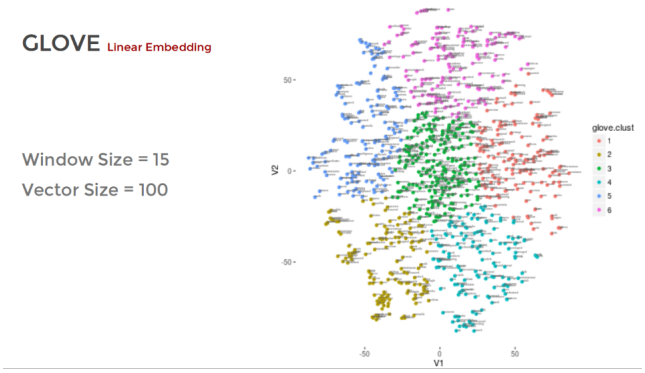

Fig. 4. Clusters with Glove embeddings

- **Sentence:** For the *what* questions (80 % of the questions) the most similar words were *which, where, represent, resemble*, pointing to entities, identities, and places. The similar words to *why* questions does not point out to something very clearly (similar to stepper, absorb, doing, mark, without).
- **Paragraph:** The paragraph embeddings were able to identify *synonyms*: sim([’college’, ’professor’], [’university’, ’teacher’]) = 0.92, sim([’marriage’, ’husband’, ’baby’], [’wife’, ’wedding’, ’children’]) = 0.85, sim([’house’,’residence’,’bed’,’accommodation’,’address’], sim [’shelter’,’mansion’,’home’, ’place’]) = 0.77. Also they identified *non-related terms*: sim(’husband’, ’floor’) = 0.30, sim(’night’, ’chicken’) = 0.29, sim(’computer’, ’city’) = 0.22. Finally *analogies:* woman is to king as man is to...? prince, most similar to “queen”: Madonna-widow-performed, most similar to “man”: said-wrote-god

B. Sentence ranking

The following modifications to the the convolutional model of [Severyn and Moschitti2015] were made:

- In our model, we added an hybrid vector representation that used both, the representation trained over the AQUAINT corpus (to obtain the most general context of each word), and over the SQUAD dataset (to obtain the particular uses of each word).
- We also used Jaccard similarity as a proxy of relevance judgments and to promote focus over the sentences with higher lexical resemblance with the question.
- We added topic information to the x_{joint} representation.

The Mean Reciprocal Rank (MRR) results for the three approaches are shown in the following table :

Model	MRR
Convolutional networks	0.25
BM25	0.71
Jaccard	0.76

We believe that the dissapointing results of the Convolutional networks is due to underfitting of the training data (80K observations are not enough).

C. Answer extraction

We train a Random Forest of 1,000 trees, 5 variables per cut and using the Gini criterion.

The results are shown in the following table:

Set	F1 score	Precision
Training	0.49	0.62
Test	0.47	0.6

It is important to point out that currently there is no *learning* phase after the *answer extraction* phase because we do not have enough models yet.

D. Pipeline implementation

To speed up the *evaluation* phase and analysis of the models developed so far, a pipeline was implemented. The implementation supports:

- An end to end pipelined execution
- *Model training:* the system allows you to choose the number of sentences to be considered as part of the answer as well as the number of instances used on the training phase. See Fig. 5
- *Model testing:* the system also gives you the option to train or test the model. And provides a final evaluation with Stanford's script. See Fig. 6
- *Interactive Mode:* for testing of new models, the system also supports interactive mode. See Figs. 7 and 8

Fig. 5. Model training

Fig. 6. Model testing

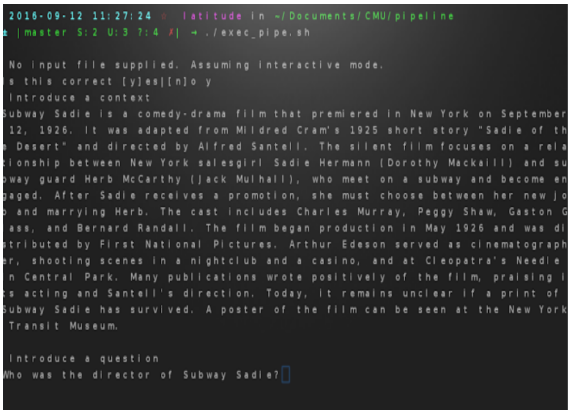

Fig. 7. Interactive mode (I)

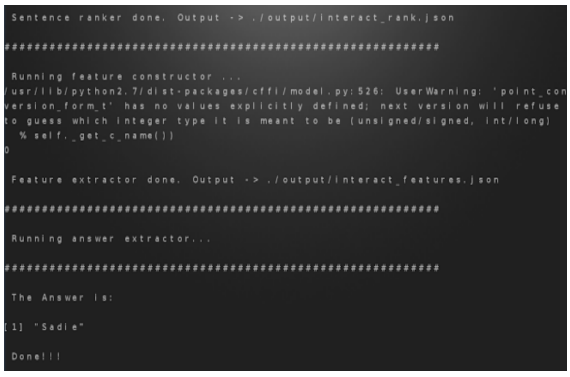

Fig. 8. Interactive mode (II)

The end-to-end execution results evaluated under Stanford's metric are the following:

Metric	Result
F1	0.20368373764600187
exact_match	0.07547169811320754

If we compare our results with [Rajpurkar et al.2016] in Fig. 9 we can see that on the one hand with F1 we are above Random guess, Sliding window and Sliding Window+Dist. but still below Logistic regression. On the other hand, with exact match we are only above random guess.

	Exact Match		F1	
	Dev	Test	Dev	Test
Random Guess	1.1%	1.3%	4.1%	4.3%
Sliding Window	13.2%	12.5%	20.2%	19.7%
Sliding Win. + Dist.	13.3%	13.0%	20.2%	20.0%
Logistic Regression	40.0%	40.4%	51.0%	51.0%
Human	80.3%	77.0%	90.5%	86.8%

Fig. 9. Stanford's results

So, there is still room to improve both scores, specially the $exact\_match$.

VI. Conclusions

Even tough we have not been able to improve Stanford's results, a solid pipeline architecture and baseline has been designed and implemented. An error analysis on the current baseline is mandatory as this will allow us to understand the behavior of the model and potentially to improve it.

References

[Hermann et al.2015] Karl Moritz Hermann, Tomas Kocisky, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. Teaching machines to read and comprehend. In *Advances in Neural Information Processing Systems*, pages 1693–1701.

[Hill et al.2015] Felix Hill, Antoine Bordes, Sumit Chopra, and Jason Weston. 2015. The goldilocks principle: Reading children's books with explicit memory representations. *arXiv preprint arXiv:1511.02301*.

[Liu et al.2015] Yang Liu, Zhiyuan Liu, Tat-Seng Chua, and Maosong Sun. 2015. Topical word embeddings. In *AAAI*, pages 2418–2424.

[Mikolov and Dean2013] T Mikolov and J Dean. 2013. Distributed representations of words and phrases and their compositionality. *Advances in neural information processing systems*.

[Pennington et al.2014] Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global vectors for word representation. In *EMNLP*, volume 14, pages 1532–43.

[Rajpurkar et al.2016] Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. Squad: 100, 000+ questions for machine comprehension of text. *CoRR*, abs/1606.05250.

[Richardson et al.2013a] Matthew Richardson, Christopher J. C. Burges, and Erin Renshaw. 2013a. Mctest: A challenge dataset for the open-domain machine comprehension of text.

[Richardson et al.2013b] Matthew Richardson, Christopher JC Burges, and Erin Renshaw. 2013b. Mctest: A challenge dataset for the open-domain machine comprehension of text. In *EMNLP*, volume 3, page 4.

[Severyn and Moschitti2015] Aliaksei Severyn and Alessandro Moschitti. 2015. Learning to rank short text pairs with convolutional deep neural networks. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 373–382. ACM.

[Weston et al.2014] Jason Weston, Sumit Chopra, and Antoine Bordes. 2014. Memory networks. *arXiv preprint arXiv:1410.3916*.

[Yang et al.2016] Zhilin Yang, Ruslan Salakhutdinov, and William Cohen. 2016. Multi-task cross-lingual sequence tagging from scratch. *arXiv preprint arXiv:1603.06270*.