Neural Reasoning For Legal Text Understanding

Kolawole John ADEBAYO [a,1], Guido BOELLA [b] and Luigi DI CARO [b]
[a] *University of Turin, Turin, Italy*
[b] *University of Turin, Turin, Italy*

Abstract. We propose a domain specific Question Answering system. We deviate from approaching this problem as a Textual Entailment task. We implemented a Memory Network-based Question Answering system which test a Machine's understanding of legal text and identifies whether an answer to a question is correct or wrong, given some background knowledge. We also prepared a corpus of real USA MBE Bar exams for this task. We report our initial result and direction for future works.

Keywords. Question Answering, LSTM, LQA, Memory Networks, Neural networks

1. Introduction

Many tasks in Natural Language Processing[2] (NLP) involves reasoning over text and semantic representation for proper text understanding e.g., Question Answering (QA). Researchers have recently employed Deep Neural Network for QA [3,9], though relying on synthetic data for training and evaluation. Our goal is to evaluate how well a Neural reasoner can perform on a real Legal Passage-Question-Answer triples. Using the USA MBE Exams, We introduce a new legal QA corpus (**LQA**) used in our work. A Memory Network (MemN)[10,2] based architecture has been used to encode and decode the Passage-Question-Answer for better semantic representation.

In the next section, we give a short review of related works and the problem we are solving. This is followed by a description of the proposed system, experiment, results and conclusion.

2. Background and Related Works

QA follows the Human learning process, i.e., committing to memory and generalizing on new events. The authors in [9,8] using Deep Neural Networks achieved 100% accu-

[1] Corresponding Author: Kolawole John, Adebayo; E-mail: kolawolejohn.adebayo@unibo.it

[2] Kolawole J. Adebayo has received funding from the Erasmus Mundus Joint International Doctoral (Ph.D.) programme in Law, Science and Technology. Luigi Di Caro and Guido Boella have received funding from the European Union's H2020 research and innovation programme under the grant agreement No 690974 for the project "MIREL: MIning and REasoning with Legal texts".

racy on some tasks[3]. However, synthetic dataset was used and the evaluations tested the ability of the models in providing factoid answers to questions of *where*, *when* and *who* about an entity. Kim et al., [5] propose a Textual Entailment (TE) based Legal Question Answering challenge with data curated from Japanese Bar Exam[4]. However, the work leans toward IR than QA. Biralatei et al., [1] used 100 real multi-choice USA MBE exam questions but also approached as a TE task. Our choice of approach is different. We seek to answer this research question:
RQ: Can we use deep learning techniques to achieve transfer-learning on passage-question-answer (PQA) with similar case templates?

By transfer learning, we mean a generalization procedure whereby our model is able to transfer hidden facts from a scenario to similar scenarios. We employ MemN model to provide a first solution. To train the system, we draw a random sample of 550 passage-questions-answer set from the 1991 MBE-I, 1999-MBE-II, 1998-MBE-III and some text samples obtained from the examiner[5]. Our choice of these sets is because it is publicly available with gold standard answer. Each row of the collection is a 4 tuple (P, Q, A_i, F). Where P is the passage, Q the question, A is the answer and F is a flag which is one for a correct answer and 0 for a wrong answer to Q, given P. We call this data the LQA corpus.

3. Neural Reasoning Over LQA

Deep Networks can autonomously learn semantic representation from text. Recurrent Neural Networks (RNNs) [6] have connections that have loops, adding feedback and memory to the networks over time. However, RNNs memory are small and also not compartmentalized enough for long range information retention [10]. Weston et al.,[10] proposed the MemN as a solution. MenNs are composed of 4 units, i.e., input units I, the Generalization Unit G, output unit O and the response unit R, which generates a representation of the Output in any specified format. We employ LSTM for building a MemN.

The Long Short-Term Memory (LSTM) [4] is a special kind of RNNs that is robust to the vanishing gradient problem. Its transition can be represented as below:

$$\begin{aligned}
i_t &= \sigma\left(W^{(i)}x_t + U^{(i)}h_{t-1} + b^{(i)}\right)\\
f_t &= \sigma\left(W^{(f)}x_t + U^{(f)}h_{t-1} + b^{(f)}\right)\\
o_t &= \sigma\left(W^{(o)}x_t + U^{(o)}h_{t-1} + b^{(o)}\right)\\
u_t &= \tanh\left(W^{(u)}x_t + U^{(u)}h_{t-1} + b^{(u)}\right)\\
c_t &= i_t \odot u_t + f_t \odot c_{t-1},\\
h_t &= o_t \odot \tanh c_t
\end{aligned} \tag{1}$$

Given a set of input from LQA, where s is a representation (P, Q, A, F). We now explain

[3]e.g. the single supporting facts and two supporting facts on BaBi dataset.`https://research.facebook.com/research/babi/`

[4]`http://webdocs.cs.ualberta.ca/~miyoung2/COLIEE2016/`

[5]http://www.ncbex.org/exams/mbe/

the MemN architecture with LSTM.
Input Representation: Assume that each of P, Q and A retains its previous definition and contain words $x_i, x_{i+1}, x_{i+2}, x_{i+3} ... x_n$. We associate each word w in our vocabulary with a vector representation $x_w \in \mathbb{R}^d$. Each x_w is of dimension $d \times V$ of the word embedding matrix W_e, where V is the size of the vocabulary. For each P, Q and A, we generate a representation by performing an element wise concatenation of each embedding $x_w \in$ P and $x_w \in$ Q.We use a special delimeter to show the end of character for the words in P, Q and A.
We encode these input in order to generate a vectorial representation for each, e.g., we encode P and Q which are the input passages and question into their memory vectors. such that
P $\longrightarrow$ I(p) , I(p) $\in \mathbb{R}^{d\times v}$, Q $\longrightarrow$ I(q), I(q) $\in \mathbb{R}^{d\times v}$ A $\longrightarrow$ I(a), I(a) $\in \mathbb{R}^{d\times v}$
F $\longrightarrow$ [0,1]

$$I_{pq} = I_p \bigotimes I_q \tag{2}$$

Generalization Unit: We obtain a representation of the Passage I(p) over I(p)(q) by performing an element-wise sum, where I(p), I(p)(q) $\in \mathbb{R}^{d\times v}$ as given in the equation below.

$$I_m = (I_p \oplus I_{pq}) \circledcirc I_q \tag{3}$$

$$I_o = (I_m \oplus I_q) \bigoplus I_a \tag{4}$$

Output Representation: Each Answer a is also a sequence $x_w \in \mathbb{R}^d$. For each x_w, we obtain the embedding, all concatenated to form a dense feature vector I(a) $\in \mathbb{R}^{d\times v}$ and update the memory with this representation by concatenating their vectors as given in equation 4. We use a 64 layer LSTM for an end-to-end connection and a softmax activation function to output the class probability over the vocabulary and the Flag F.
From the equation, we used the $\oplus$ to represent the element-wise dot product, $\circledcirc$ is used to denote the element-wise sum and $\bigoplus$ denotes a concatenation operation.

4. Experiments

Given a background knowledge, a question and an answer, the goal is to make the model identify whether the answer is right for the question or not. We evaluated our system on the LQA corpus[6] We implemented our adaptable Memory Network following the works in [10]. For neural computation, instead of generating an on-the-fly embedding e.g., by encoding our input as one-hot vectors, we take advantage of the embedding layer offered by Keras[7], we used the 300 dimensional Glove vectors [7] for embedding. We uniformly use dropout of 0.20, batch size of 25, ADAM optimizer and learning rate of 0.01 and 200 epochs. Since we have 4 sets of (P,Q,A,F) for each distinct Passage-Question pair, it is necessary to address instances imbalance as we have 3:1 in terms of wrong to correct answers. To address this, we remove one wrong answer, thus resulting into 2:1 wrong-correct ratio. We also evaluated the system when all the samples were used for training. Table 1 shows the result obtained on the LQA corpus. The full-set column shows the

[6]Full dataset is to be released after publication
[7]`https://github.com/fchollet/keras`

Flag	Full-set	Augmented-Set
Correct	68.50	71.2
Wrong	73.40	75.00
Total	70.90	73.10
Baseline (Random Guessing)	52.00	–

Table 1. Evaluation on LQA dataset

result when all the training samples were used and the augmented-set column otherwise. The Flag column shows the test for both correct or wrong class of passage , question and answer triple, i.e., when (P,Q,A) = 1 or 0. For the purpose of evaluation, we divided the dataset in the ratio 80:20 train/test split. We report our results using only the accuracy metrics.

Our initial result is encouraging, especially since no feature was engineered neither did we use any semantic resource. In comparison to the works of [1] which report an accuracy 0f 63.5%, our average accuracy supersede theirs. However, comparison is not empirical since we use different dataset. The baseline reported in table 1 was obtained from random guessing the Flag [0,1].

5. Conclusion

This paper presented a Legal Question Answering system using LSTM-based MemN. The proposed evaluation or task is different from textual entailment since the goal is to make a machine say whether an answer to a question is correct or not, given some background knowledge. We report encouraging results.

References

[1] Fawei Biralatei, Wyner Adam, and Pan Peng Jeff. Passing a usa national bar exam: a first corpus for experimentation. 2015.

[2] Antoine Bordes, Nicolas Usunier, Sumit Chopra, and Jason Weston. Large-scale simple question answering with memory networks. *arXiv preprint arXiv:1506.02075*, 2015.

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

[4] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[5] Mi-Young Kim, Ying Xu, and Randy Goebel. Legal question answering using ranking svm and syntactic/semantic similarity. In *JSAI International Symposium on Artificial Intelligence*, pages 244–258. Springer, 2014.

[6] LR Medsker and LC Jain. Recurrent neural networks. *Design and Applications*, 2001.

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

[8] Sainbayar Sukhbaatar, Jason Weston, Rob Fergus, et al. End-to-end memory networks. In *Advances in neural information processing systems*, pages 2440–2448, 2015.

[9] Jason Weston, Antoine Bordes, Sumit Chopra, Alexander M Rush, Bart van Merriënboer, Armand Joulin, and Tomas Mikolov. Towards ai-complete question answering: A set of prerequisite toy tasks. *arXiv preprint arXiv:1502.05698*, 2015.

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