ELN - Project Report

Michele Colombo

October 5, 2017

Abstract

With this project we faced the task of understanding whether two questions can be considered as duplicate or not. We exploited the Quora Question Pairs dataset and pretrained word vectors to build and assess two different deep neural network architectures.

Question 1	Question 2	Y
Can I make 800,000 a month betting on horses?	Can I make 400,000 a month betting on horses?	yes
What should I do to improve my English ?	How do I improve my overall native English language?	yes
How do I forget my girlfriend?	How can I forget my first girlfriend?	no
Which is the best laptop below Rs 30000?	Which is the best laptop model to buy within 30k?	yes
What is a good solar panel installation provider in Bell, California CA?	What is a good solar panel installation provider in Calexico, California CA?	no

Table 1: Some examples of question pairs and labels sampled from the dataset

1 Introduction

We faced the task posed by the Quora Question Pairs dataset, pubblicated along with the Kaggle competion[1]. The goal is to build a classifier ables to classify whether two questions, written in english, have the same meaning. Our approach relies on pre-trained word vectors, specifically GloVe[2], and deep neural network models. Two architecture have been developed and assessed side-by-side, the first in the flavour of a sentence encoder, the second more inspired by neural reasoner systems. All the needed code is available on Github[3]

2 Data

The public dataset provides 404K hand-labelled pairs of actual Quora english questions, in Table 1 are shown some samples randomly extracted from the training set. A blind test-set of 2.3M pairs is available as well, cross-entropy performance can be computed submitting predictions to Kaggle, unfortunately perfomance comparison is not trivial since as an anti-cheating measure the blind test is augmented with synthethic samples in such a way that labels distribution differs from the training set. Specifically, there are 36.92% positive samples in the training set and 17.46% in the blind test set[4].
The inherent symmetry of the task was exploited in the models design and to augment the dataset as well. For each pair $< q_1, q_2, y >$ also the symmetric $< q_2, q_1, y >$ was added to the dataset, both pairs are kept in the same data partition to prevent overfitting issues.

[1] `https://www.kaggle.com/c/quora-question-pairs`
[2] `https://nlp.stanford.edu/projects/glove/`
[3] `https://github.com/m-colombo/nlp-quora-duplicate-question`
[4] This can be estimated submitting a constant predictor output

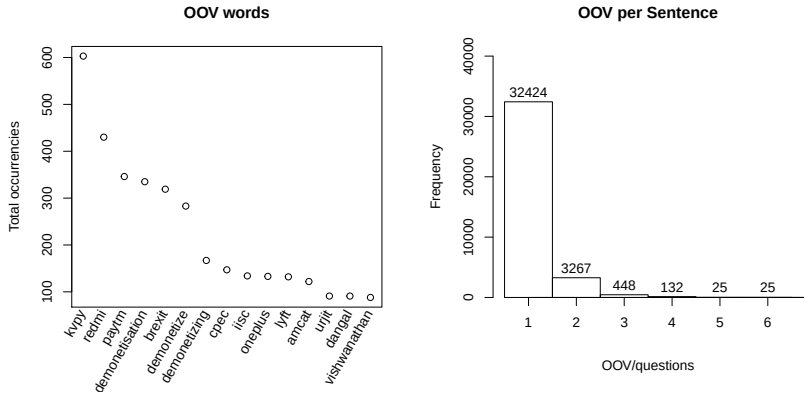

Figure 1: Most common out-of-vocabulary words occurrencies counts (on the left). Distribution of the number of out-of-vocabulary words per question, bucket 0 is ignored for readability (on the right).

Preprocessing In the preprocessing phase questions are lower-cased, non alphanumerical characters are replaced by spaces and some rather simple substitution are performed, such as expanding common language contractions and fixing most frequent mispelled words.
The cleaned text is split along white spaces and tokens are then mapped to their respective embeddings in glove 840B.300d vocabulary, out-of-vocabulary words are simply skipped. Figure 1 shows the most common out-of-vocabulary words on the left and the number of out-of-vocabulary words per questions distribution on the right. In some rare case questions are found to have no word vectors at all, this samples had been removed from the set, a further investigation reveiled that are degenerate sentences (ie. empty, only symbols, no words, random letters).
Along with question embeddings some hand-crafted features are computed, such as Jaccard similarity of questions out-of-vocabulary words.

3 Models

3.1 Sentence Encoder

Figure 2 shows the architecture of the Sentence Encoder model. Each question, represented as a sequence of word vectors, is encoded in a single vector as the last output of a Recursive Neural Network built with Long Short Term Memory cells. In order to reduce the number of trainable parameters task symmetry has been exploited encoding both questions with the same network thanks to a weight sharing architecture. Both question embeddings among with hand-crafted features are then fed to a feed-forward layer that outputs the probability of the two question having the same meaning. Model regularization has been

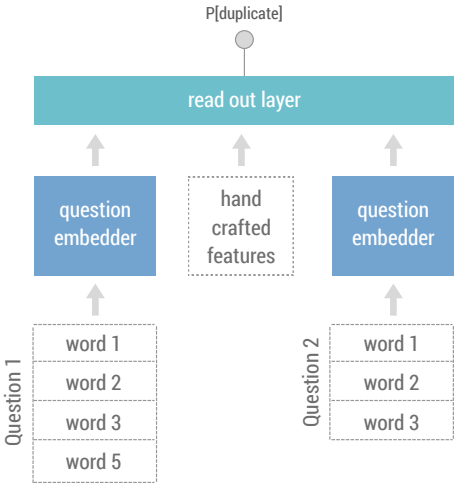

Figure 2: Sentence embedder architecture

achieved by mean of drop-out in the question embedder module and the read-out layer.

3.2 Neural Reasoner

Figure 3 shows an overview of the Neural Reasoner model architecture. The *Reasoner* module realizes the reasoning part, it consists of a LSTM layer that it's dynamically unfolded until the *Answerer* module outputs to stop, or the hard limit it's reached. At each recursive step the reasoner module is fed with the hand-crafted features, the previous reasoner state and the Question Analyzer outputs. *Question Analyzer* module is sketched in Figure 4, each word-vectors is projected by a Bi-Directional LSTM Recursive Neural Network and them merged as a weighted sum of the embeddings by mean of an *Attention* mechanism dependents from the previous reasoner state. Model regularization has been achieved by mean of drop-out in the RNN, specifically the question embedder and the reasoner modules.

4 Experiments

Processing, models and experiments had been implemented using Tensorflow 1.3 [5] and are pubblicly available on Github [6]. To speed up the experiments GPU parallelization of the gradient computation had been implemented allowing to use multiple GPU simultaneously and achieving a speed of thousands of pairs

[5] `https://www.tensorflow.org`
[6] `https://github.com/m-colombo/nlp-quora-duplicate-question`

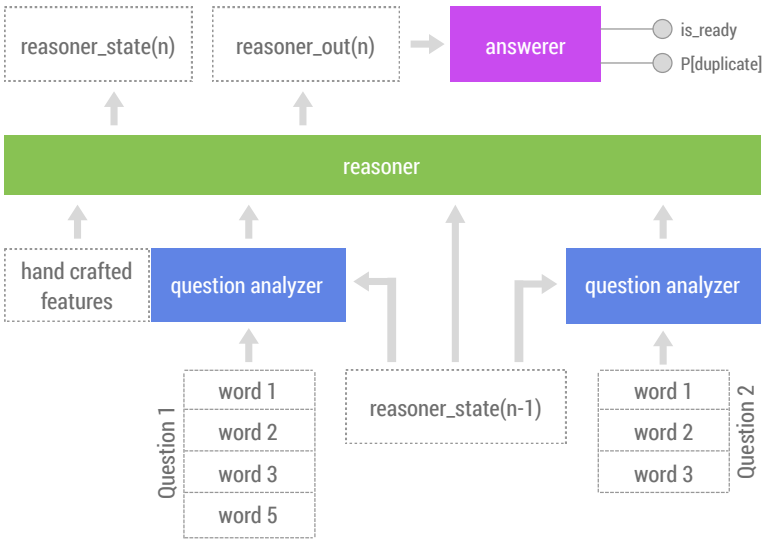

Figure 3: Neural Reasoner model architecture overview

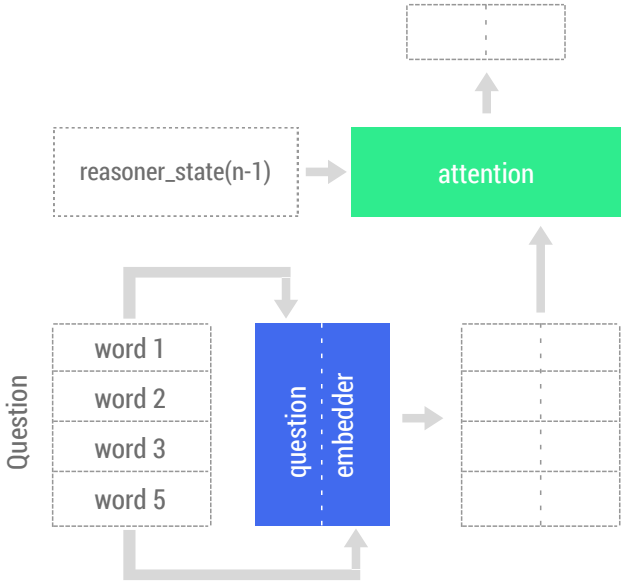

Figure 4: Sketch of the Question Analyzer module

Model	Validation Acc.	Validation wCE	Test CE
Sentence Embedder	80.2%	0.28	0.31
Neural Reasoner	83.4%	0.24	0.30

Table 2: Selected models performance. Acc. refers to Accuracy, wCE to weighted cross-entropy (see Section 4) and CE to cross-entropy.

per seconds during training.
All models had been trained with an Adam optimizer minimizing a weighted Cross-Entropy loss function:

$$-\frac{1}{n}\sum_{k=1}^{n}\omega_p * y_k * Log(\hat{y}_k) + \omega_n * (1 - y_k) * Log(1 - \hat{y}_1)$$

$$\omega_p = \frac{positive\_ratio_{test}}{positive\_ratio_{train}} \qquad \omega_n = \frac{negative\_ratio_{test}}{negative\_ratio_{train}}$$

The weight allowed us to have a rough estimation of the test performance also using the validation set which has a different class distribution (see Section 2), moreover we explicitly target the test set performance. Intuitively the error contribution of each class it's scaled according to the target distribution.
Model hyper-parameters have been tuned with an hold-out cross-validation on 10% of the labelled data, randomly sampled.

Sentence Embedder The best sentence embedder model selected has the embedder module composed of 250 LSTM cells and the readout layer of 100 neurons.

Neural Reasoner The neural reasoner model selected through the model selection phase has a Question Embedder with 300 LSTM cells, Attention hidden layer of 100 units, Reasoner with 300 LSTM cells and 5 maximum iteration allowed.

Performance of the two selected model are shown in Table 2. For the selected Neural Reasoner model in the Table 3 are shown some of the pairs, sampled from the validation set, where the model prediction was most wrong, i.e. the distance between prediction and target is maximal. In Table 4 are instead shown some of the pairs that the model classificated right, but with maximal uncertainity.

5 Conclusion

We trained two rather different kind of deep neural network model to fullfil the task of classificate pairs of english questions with the same meaning. Both architectures ultimately achieved almost the same performance with several different hyparameters configurations showing their ability to cope with the task

Question 1	Question 2	Y
If talent is independent of skin color then why are nations with white people as the majority more developed than Asian nations with brown people which are relatively more developed than African nations with black people?	Why are nations with black people (Ghana, Nigeria) poorer than white nations (Germany, UK)? Is it because white people are more intelligent?	yes
India is home to 70 billionaires, Mukesh Ambani is the richest Indian. As an Indian, what's your view on this?	What is your review of India Home to 70 Billionaires?	yes
I have a MacBook air. I want to buy a printer. Which printer should i buy?	What printer should I buy?	yes
Who is 'they' in Interestellar?	Who are the 'they' in the movie Interstellar?	yes
What can wrestlers do to prevent cauliflower ears?	Why do wrestlers have deformed ears?	yes

Table 3: Some of the pairs that the selected Neural Reasoner model predicted wrong, specifically those where the difference between prediction and target is maximal. Y is the ground truth label, the model predicted the opposite.

Question 1	Question 2	Y
How do I overcome despair and depression?	How do I overcome depression?	no
Why does cutting feel so good?	Why does cutting even make me feel better?	no
What would happen if Olympiaturm in Munich suddenly vanished?	What would happen if Munich suddenly vanished?	no
Which is the best book for contract law?	What are the best books on employment and labor law?	no
What race are the American Indians?	What is the race of Indian people?	no

Table 4: Some of the pairs that the selected Neural Reasoner model predicted right, specifically those where the confidence of the model was minimal.

at hand, even though better perfomance where expected. As possible avenues of further improvement we suggest: a better investigation of the text preprocessing in order to assess the loss of relevant informations; study and test more hand-crafted features since they've proven useful for the Kaggle community; pretrain question embedders on unlabelled data with autoencoder tecniques.