Supervised Mover's Distance: A simple model for sentence comparison

Muktabh Mayank Srivastava

ParallelDots, Inc.
muktabh@paralleldots.com,
website: https://paralleldots.xyz

Abstract. We propose a simple neural network model which can learn relation between sentences by passing their representations obtained from Long Short Term Memory(LSTM) through a Relation Network. The Relation Network module tries to extract similarity between multiple contextual representations obtained from LSTM. Our model is simple to implement, light in terms of parameters and works across multiple supervised sentence comparison tasks. We show good results for the model on two sentence comparison datasets.

Keywords: Supervised Mover's Distance, Sentence Comparison, Paraphrase Detection, Natural Language Inference

1 Introduction

Sentence Comparison is a common NLP task which comes up in multiple domains. Sentence comparison measure might be needed to check redundant data [1] or check sentences for being paraphrases [2]. We propose a new method to compare sentences for both these tasks, which uses Relation Networks(RN) module [3] in combination with a Long Short Term Memory (LSTM) [4]. To compare two sentences, all possible pairs of dense vectors, one from each sentence in a pair, are passed through a Relation Network module to decipher relationship information between sentences. To make sure the dense vectors passed to Relation Network have contextual information, sentences are individually passed through a LSTM and the hidden units obtained for each sentence are used as dense vectors. The combination of both models was done following the intuition of supervised Earth Mover's Distance[5] where LSTM aims to model word importance and relation networks help optimize the minimum flow, hence the name Supervised Mover's Distance.

2 Previous Work

In our experiments, we focus on two sentence comparison tasks: 1. Duplication detection between questions [1] and 2. Paraphrase detection[2]. Duplication detection task aims to check whether two questions intend to ask about the same topic. Paraphrase Detection task aims to classify sentences according to whether they have a paraphrase/semantic equivalence relationship. Deep Neural Networks networks have shown state of the art performance in sentence comparison tasks. Most top methods for paraphrase detection are based on Deep Neural Networks[6, 7]. BiMPM model [8] combines a custom matching layer with LSTMs [4] for question duplication detection.

Relation Networks(RN) [3] was introduced as a simple module for relational reasoning. The module has been used for spatial relational reasoning in images earlier, but we try to use it for deciphering relationships in text by combining it with an LSTM. RNs operate on a set of objects without regard to the objects' order, so we use LSTMs to extract out temporal information containing word importances and use RNs on top for reasoning. RN module has a g-layer which models relation between all possible pairs of objects and a f-layer which models the final output looking at the relation between objects.

Another set of models which use pairwise relationships to model document similarity are Word Mover's Distance (WMD) [9] and its supervised variant (SWMD) [10]. They both are methods to calculate Earth Mover's Distance (EMD)[5] between documents for document calculation. Both these methods calculate flows (weightages) to be given distances between each possible pair of words to calculate document distance. WMD is an unsupervised distance measure between documents. The SWMD architecture works on longer documents(with more than 40 words) and uses a complex optimization procedure to optimize EMD. SWMD uses a cascaded loss where the inner loss optimizes word importance and outer loss optimizes EMD flow. The intuition for our model was that sentence similarity being a simpler task, the combination of a LSTM and RN can be used to approximate the supervised EMD, where LSTM models word importance and RN optimizes EMD.

3 Method

As Supervised Mover's Distance, we propose a baseline that generalizes well across different tasks. Our network combines LSTM layers [4] with a RN module modeling semantic relationship between the sentences. The neural network architecture we propose is trained on pair of sentences to predict one of various classes the pair might fall into. For redundancy detection and paraphrase detection the labels are positive or negative, but might be different for any other tasks. The architecture has two basic parts: 1. LSTM layers and 2. RN layer. The LSTM layers can have depth of one or higher which take both sentences as input individually and produce hidden layers as output for each of the words in the sentences. This would yield two series of output hidden states, one hidden state for each time step of each sentence. To clarify again, there is one common LSTM which runs on both sentences separately to produce respective hidden states. In the RN, all possible pairs of hidden states across both sentences are taken as concatenated vectors and passed through a fully connected (or Dense) layer. Aforementioned fully connected layer is the g-layer of the RN. This yields an

PeerJ Preprints | https://doi.org/10.7287/peerj.preprints.26847v2 | CC BY 4.0 Open Access | rec: 23 May 2018, publ: 23 May 2018

embedding for each possible pair of hidden state outputs from the LSTM. These embeddings are averaged and passed through another fully connected layer to predict the output. This fully connected layer is the f-layer of the RN. By taking all pairs of hidden states and using them to model sentence comparison task, we hypothesize that the RN is able to model flow optimization of EMD between the sentences, while LSTM models the word importance before they are fed into the flow optimization task.

We illustrate the architecture in 1. Our model is light in terms of parameters as it has only a LSTM layer and two dense (fully connected) layers in RN. A limiting case of the architecture can be when the number of LSTM layers is zero, and word embeddings are passed as inputs directly to RN.

The network is trained with common hyperparameters for both the tasks. Pretrained word embeddings are used to initialize the word embedding layer which are finetuned by backpropogation. We use the publicly available 6 Billion token 100 dimensional version of GloVe embeddings [11]. The hidden state output from the LSTM is 100 dimensions and the size of embedding generated in the relational layers is 100 dimensions too. The network is trained with simple Stochastic Gradient Descent with momentum (common values for training across both datasets, learning rate = 0.001, momentum=0.9).

4 Results

As stated we test our model on two datasets. Model is compared to state of the art methods and baselines for each dataset in this section.

Microsoft Research Paraphrase Corpus Microsoft paraphrase corpus [2] is a corpus of sentence pairs classified as paraphrases or non-paraphrases. The dataset has 4076 sentences in training set and 1725 sentences in test set. Our model was trained on the training set with the standard set of hyper parameters mentioned above and evaluated on the test set. The accuracy numbers of different models were taken from this url[1]. Our model gets an accuracy of 80.2% on the dataset as compared to state of the art accuracy of 80.4% [12].

Quora Questions' Pair Dataset Quora Questions' Pair Dataset contains question pairs from the Q&A website[2] tagged as similar or not. A random 90%-10% train-test split is performed as is customary for other methods and the model is trained on the train set and evaluated on the test set. As in case of other datasets, the hyperparameters are fixed as the standard values specified earlier while training. Our model gets an accuracy of 81.2% on the dataset. List of state of the art models on the dataset is available on this url[3]. The best accuracy a model gets on the dataset is 88% [8]. Although our model doesn't get

[1] https://aclweb.org/aclwiki/Paraphrase_Identification_(State_of_the_art)
[2] quora.com
[3] https://github.com/bradleypallen/keras-quora-question-pairs

PeerJ Preprints | https://doi.org/10.7287/peerj.preprints.26847v2 | CC BY 4.0 Open Access | rec: 23 May 2018, publ: 23 May 2018

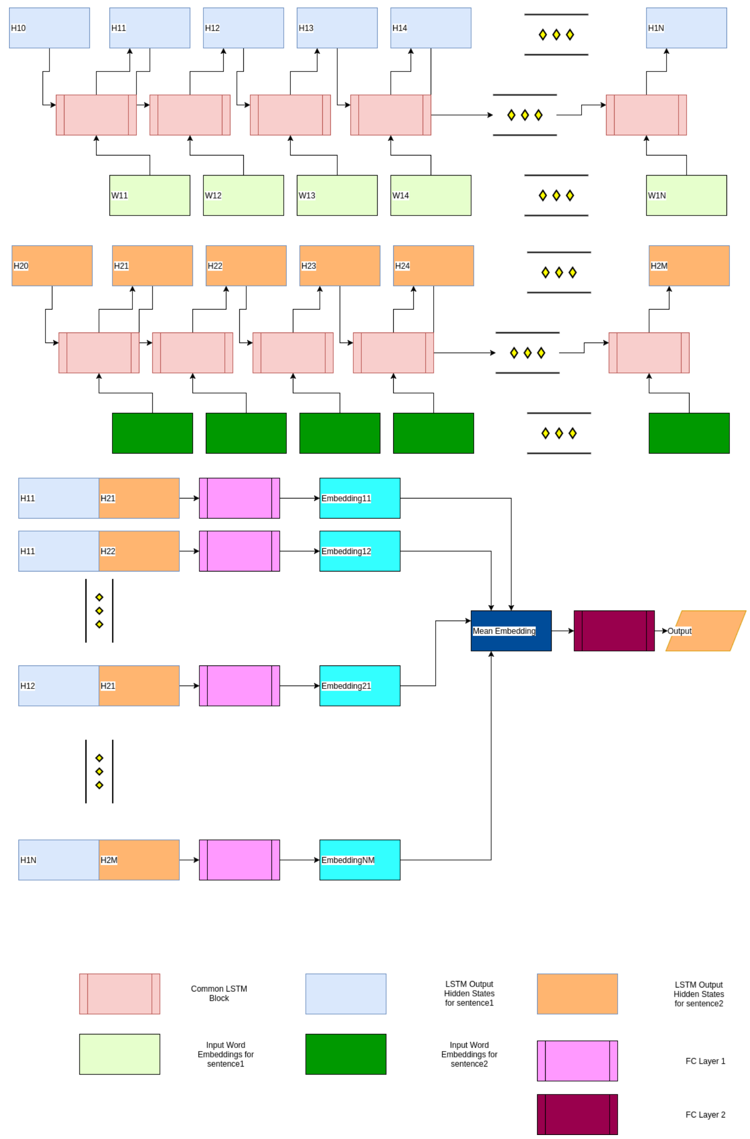

Fig. 1. Illustration of the Neural Network architecture for Supervised Mover's Distance between sentences (user might need to zoom-in to view)

PeerJ Preprints | https://doi.org/10.7287/peerj.preprints.26847v2 | CC BY 4.0 Open Access | rec: 23 May 2018, publ: 23 May 2018

results as good as the state of the art, it is competitive to baselines like siamese Convolutional Neural Networks (79.6%) and siamese LSTMs(82.58%).

It should be noted that in both models, dataset specific hyperparameter tuning was not performed.

5 Discussion

We propose a new method which uses a new and simple neural network model to compare sentences. The model tries to approximate supervised Earth Movers' Distance(EMD) between sentences by splitting the task of calculating word importance calculation and flow optimization between a LSTM and a RN module. Models performance is calculated on two sentence comparison datasets.

References

1. Shankar Iyar, N.D., Csernai., K.: First quora dataset release: Question pairs (January 2016)
2. Dolan, B., Brockett, C.: Automatically constructing a corpus of sentential paraphrases. In: Third International Workshop on Paraphrasing (IWP2005), Asia Federation of Natural Language Processing (January 2005)
3. Santoro, A., Raposo, D., Barrett, D.G., Malinowski, M., Pascanu, R., Battaglia, P., Lillicrap, T.: A simple neural network module for relational reasoning. In: Advances in neural information processing systems. (2017) 4974–4983
4. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural computation **9**(8) (1997) 1735–1780
5. Rubner, Y., Tomasi, C., Guibas, L.J.: A metric for distributions with applications to image databases. In: Computer Vision, 1998. Sixth International Conference on, IEEE (1998) 59–66
6. Cheng, J., Kartsaklis, D.: Syntax-aware multi-sense word embeddings for deep compositional models of meaning. arXiv preprint arXiv:1508.02354 (2015)
7. Croce, D., Moschitti, A., Basili, R.: Structured lexical similarity via convolution kernels on dependency trees. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics (2011) 1034–1046
8. Wang, Z., Hamza, W., Florian, R.: Bilateral multi-perspective matching for natural language sentences. arXiv preprint arXiv:1702.03814 (2017)
9. Kusner, M.J., Sun, Y., Kolkin, N.I., Weinberger, K.Q.: From word embeddings to document distances. In: Proceedings of the 32Nd International Conference on International Conference on Machine Learning - Volume 37. ICML'15, JMLR.org (2015) 957–966
10. Huang, G., Guo, C., Kusner, M.J., Sun, Y., Sha, F., Weinberger, K.Q.: Supervised word mover's distance. In: Advances in Neural Information Processing Systems. (2016) 4862–4870
11. Pennington, J., Socher, R., Manning, C.D.: Glove: Global vectors for word representation. In: Empirical Methods in Natural Language Processing (EMNLP). (2014) 1532–1543
12. Ji, Y., Eisenstein, J.: Discriminative improvements to distributional sentence similarity. In: Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing. (2013) 891–896

PeerJ Preprints | https://doi.org/10.7287/peerj.preprints.26847v2 | CC BY 4.0 Open Access | rec: 23 May 2018, publ: 23 May 2018