24SP Math179 final project

Rui Zhang
Harvey Mudd College
Claremont, CA 91711
ruzhang@hmc.edu

Abstract

This report delves into the result of SciNLI dataset on Natural Language Inference using Enhanced Sequential Inference Model. Enhanced Sequential Inference Model is one of the best models for Natural Language Inference based on Long short-term memory (LSTM) network. For this final project, I trained and test the SciNLI dataset on ESIM model implemented with PyTorch to compare with the result of SciNLI dataset trained on BiLSTM model from the paper *SciNLI: A Corpus for Natural Language Inference on Scientific Text*.

1 ESIM model

1.1 LSTM

Long Short-Term Memory (LSTM) is a type of recurrent neural network (RNN) architecture. LSTM networks are designed to overcome the vanishing gradient problem that occurs when training traditional RNNs on long sequences of data. They are particularly effective in capturing long-term dependencies in sequential data, making them well-suited for tasks such as natural language processing.

Traditional RNNs have a simple structure where each neuron processes input data and passes its output to the next time step in the sequence. However, they struggle with learning long-term dependencies due to the vanishing gradient problem, where gradients diminish as they are backpropagated through time. LSTM networks introduce the memory cell, which retains information over long sequences and selectively updates or forgets information based on the input data.

LSTMs incorporate three types of gates taking the previous hidden state h_{t-1} and the current input x_t to control the flow of information into and out of the memory cell: Forget Gate f_t determines which information to discard from the cell state, outputting a value between 0 and 1. Input Gate i_t decides which new information to store in the cell state, outputting a value between 0 and 1. Output Gate o_t controls which information from the cell state should be exposed as the output.

The cell state is updated using the forget gate, input gate, and output gate. These gates regulate how information flows into and out of the memory cell, allowing LSTMs to retain important information over long sequences while discarding irrelevant information. LSTMs also produce a hidden state h_t at each time step, which is a function of the current input, previous hidden state, and current memory cell state. The hidden state contains information that the LSTM network has deemed relevant for predicting the output at the current time step.

Bidirectional LSTMs (BiLSTM) are an extension of standard LSTMs that process input sequences in both forward and backward directions. In a BiLSTM architecture, there are typically two separate LSTM layers: one layer processes the input sequence in the forward direction and the other layer processes the input sequence in the backward direction. The outputs of these two layers are then typically concatenated or combined in some way before being passed to subsequent layers or output units. BiLSTMs are particularly useful in tasks where understanding the entire context of a sequence is important, such as natural language processing tasks like machine translation.

1.2 ESIM

The Enhanced Sequential Inference Model (ESIM) is an enhanced LSTM model introduced by Chen et al. in Enhanced LSTM for Natural Language Inference Chen (2017). The ESIM model operates by sequentially processing pairs of input sentences, typically referred to as the premise and hypothesis. It aims to determine the logical relationship between these two sentences, classifying them into one of three categories: entailment (the hypothesis logically follows from the premise), contradiction (the hypothesis contradicts the premise), or neutral (there is no logical relationship between the two sentences).

Initially, the input sentences are tokenized and embedded into dense vector representations, typically using techniques like word embeddings or contextual embeddings. The embedded representations of the premise and hypothesis sentences are then fed into bidirectional LSTM (BiLSTM) layers so the model can learn to represent a word and its context. Then, a soft alignment layer is used to compute the attention weights to obtain the local relevance between a premise and hypothesis. For the hidden state of a word in a premise, the relevant semantics in the hypothesis is identified and composed using the attention weight. Then, to further sharpen local inference information between elements and capture inference relationships, the difference and the element-wise product for the tuples of BiLSTM result and weighted summation for relevant semantics are computed and concatenated with the original vectors. At last a composition layer is applied to determine the overall inference relationship between a premise and hypothesis. In this layer, BiLSTM is used to capture local inference information and context for local inference results. Then, both average and max polling are used and concatenated to form the final fixed length vector. The vector is then put into a final multuplayer perceptron classifer, which has a hidden layer with tanh activation and softmax output layer. The model is trained using multi-class cross entropy loss.

ESIM effectively model the complex interactions and dependencies within the input sentences, leading to improved performance on tasks like natural language inference. It demonstrates strong performance on benchmark datasets like the Stanford Natural Language Inference (SNLI) dataset and the MultiNLI dataset. Its modular architecture and reliance on standard neural network components make it widely applicable and adaptable to various NLP tasks.

2 SciNLI dataset

SciNLI dataset is a large dataset for NLI that captures the formality in scientific text and contains 107, 412 sentence pairs extracted from scholarly papers on NLP and computational linguistics introduced by Sadat und Caragea (2022). It contains four classes – constrasting, reasoning, entailment, and neutral, each has 25353 pairs in training data, 500 pairs in dev data, and 1000 pairs in testing data. The training data are automatically annotated according to the logic linking words while the dev and testing data are manually annotated.

The dataset is recently introduced and not much training accuracy result is available. I could not find any paper using ESIM model to train this data set. In the paper of Sadat und Caragea (2022), they evaluate SciNLI by experimenting with traditional machine learning models using lexical and syntactic features, neural network models — BiLSTM, CBOW, CNN, and pre-trained language models — BERT, SciBERT, RoBERTa, and XLNet. They achieved an accuracy of 61.32 ± 0.08 using BiLSTM model, which is most similar with ESIM model I am using. Their best performing model based on XLNet shows 78.23% accuracy.

3 Experimenting

3.1 Data

The data was directly downloaded from https://github.com/msadat3/SciNLI, the Github repository of Sadat und Caragea (2022).

3.2 Model

The ESIM model is adapted from Github repository by Coet (2019). I followed the instructin and training the model with SNLI dataset from Bowman (2015) to verify the model works well. The detail of this part can be checked through midterm report.

3.3 Testing

I programmed prepossessing, training and testing files to work on the SciNLI dataset. Using a labeldict of "reasoning": 0, "entailment": 1, "neutral": 2, "contrasting": 3 to preprocess and training with hidden size 300, dropout 0.5, batch size: 32, learning rate: 0.0004, and max gradient norm of 10.0, I get a testing accuracy of around 66.0 ± 0.5.

3.4 Analyzing

Compared to the accuracy of 61.32 ± 0.08 using BiLSTM model in Sadat und Caragea (2022), we can clearly see that ESIM model is stronger in this kind of work. However, it still have a very large gap between the accuracy of XLNet model. From Sadat und Caragea (2022), we can see that the neural network models generally have much lower accuracy compared to pre-trained language models on SciNLI dataset. The ESIM model works better than the CNN and BiLSTM model accroding to Sadat und Caragea (2022), but it is probably more efficient and work on the pre-trained language models to improve the performance on SciNLI dataset.

References

[Bowman 2015] BOWMAN, Angeli G. Potts C.- Manning C.: A large annotated corpus for learning natural language inference. In: *Conference on Empirical Methods in Natural Language Processing* (2015). – URL https://api.semanticscholar.org/CorpusID:14604520

[Chen 2017] CHEN, Zhu X. Ling Z.-H. Wei S. Jiang H. Inkpen D.: Enhanced LSTM for natural language inference. In: *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)* (2017). – URL arXiv:1609.06038

[Coet 2019] COET, A.: *ESIM*. https://github.com/coetaur0/ESIM. 2019

[Sadat und Caragea 2022] SADAT, Mobashir ; CARAGEA, Cornelia: SciNLI: A Corpus for Natural Language Inference on Scientific Text. In: *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Association for Computational Linguistics, Mai 2022, S. 7399–7409. – URL https://aclanthology.org/2022.acl-long.511