Improved Semantic Representations From Tree-Structured Long Short-Term Memory Networks

Authors: Kai Sheng Tai, Richard Socher and Christopher D. Manning
Year: 2015
Journal: Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics
Report:

1. **Abstract:**

"Because of their superior ability to preserve sequence information over time, Long Short-Term Memory (LSTM) networks, a type of recurrent neural network with a more complex computational unit, have obtained strong results on a variety of sequence modeling tasks. The only underlying LSTM structure that has been explored so far is a linear chain. However, natural language exhibits syntactic properties that would naturally combine words to phrases. We introduce the Tree-LSTM, a generalization of LSTMs to tree-structured network topologies. Tree-LSTMs outperform all existing systems and strong LSTM baselines on two tasks: predicting the semantic relatedness of two sentences (SemEval 2014, Task 1) and sentiment classification (Stanford Sentiment Treebank)."

2. **Introduction:**

LSTM being an approach to store the sequential data and producing it had served the major contributions of the semantic representations. But the obstacle of storing longer sentences and dealing with more complex data had been one of the drawbacks of the chain structured LSTM. To overcome the deficiency, they have created tree-structured LSTM which had outperformed in both the sentiment classification and semantic relation. So, now the linear sequential LSTM can be defined as the derivation of the tree structured LSTM with each node containing only one child node.

3. **Content and Approach:**

- Tree structure had two basic tasks to be considered.
  - Semantic relationship
  - Sentiment Classification
- The standard sequential LSTMs generally have one input i_t, forget node f_t, output node o_t, and hidden node h_t. They are related with some sigmoid functions as follows:
  - $i_t = \sigma\left(W^{(i)} x_t + U^{(i)} h_{t-1} + b^{(i)}\right)$,
  - $f_t = \sigma\left(W^{(f)} x_t + U^{(f)} h_{t-1} + b^{(f)}\right)$,
  - $o_t = \sigma\left(W^{(o)} x_t + U^{(o)} h_{t-1} + b^{(o)}\right)$,
  - $u_t = \tanh\left(W^{(u)} x_t + U^{(u)} h_{t-1} + b^{(u)}\right)$,
  - $c_t = i_t \odot u_t + f_t \odot c_{t-1}$

- $h_t = o_t \odot \tanh(c_t)$

  Where $\odot$represents the elementwise multiplication.

- The forget gate f_t controls the extent to which the previous cell had to be forgotten.
- The two commonly used variants of sequential LSTM are the bidirectional LSTM and Multilayer LSTM.
- **Bidirectional LSTM** is a process of taking the inputs into it and calculates the both left right nodes to result in the past and future information of the input.
- **Multi-layered LSTM** is used to capture the higher layers of the long term dependencies of the input order. The both variants can be combined and Bidirectional Multi-layered LSTM is also applicable
- The authors derived the tree structure into two main branches as the Child-sum tree LSTM and N-ary tree LSTM.
- The approach of the input is dependant on the structure of the tree used for the network. In the tree structure the LSTM contains each forget gate for each corresponding node.
- The leaf node accepts the word vectors which are correspondingly relative, as the input.
- It is given that the input gate i_j for each node j had the values nearer to '1' if they are semantically similar, else they have values nearer to zero.
- Generally, the child-sum tree LSTM is used for the very high branch factor and the unordered children.
- If the dependant of the head node are highly sensible, then we take the dependency tree LSTM.
- **N-ary tree LSTM**: If the children are ordered and the branching factor is limited to some certain value 'N', then this tree is adapted.
- Forget gate parameterization is done for the flexible chain of information from the child to parent.
- **Constituency Tree LSTMs** are a typical application of binary LSTMs when left and child nodes are distinguished.
- For every node labels are given, which suggest it to be a supervised learning, which is a subset of discrete class.
- For every node the softmax function, which converts the unnormalized vector into a normalized distributed probability function, is used to predict the label while developing.
- Then a cost function and the real value function which is greater than 1 are defined to check the relatedness of the input.
- For every node the left and right hidden nodes are defined and calculated to find the label of the current node.

- Two experiments are done to check the consistency and the efficiency of the adapted approach which are
    - Sentiment classification of the sampled from the movie review:
        - It is furthered branched into two subtasks such as the binary classification of the sentences and the fine grained classification. The binary classification results whether the sentence is positive or negative, whereas the fine grained classifies the sentence in the scale of [1-5] such as very negative, negative, neutral, positive and very positive. Each node is given the label, if the sentence matches the labelled sentence in the training set.
    - Prediction the semantic relationships of the sentences:
        - To predict the relationship similarity, it is further derived into five annotators based on the many future works done.

4. Results:

- Sentiment Classification:
    - They stated that the constituency tree LSTm had outperformed the existing algorithms on both the fine grained and the accuracy gained of the binary subtask.
    - Updating the word representation while learning, had resulted in the great boosting of the accuracy.
- Semantic Relation:
    - Even the longer sentences with around 40 -45 words had resulted in significant accuracy which is not very much possible by the sequential LSTM.
    - In the experiment, dependency tree LSTM outperformed the constituency tree algorithm.

5. Discussion and Thoughts:

- The LSTMs had the ability to store and emphasising the information from the relatively distant nodes.
- The results are very significant that the sentences without overlap also resulted in the very good binary classification and semantic relationship.
- The tree structure had helped in creating and dealing with the longer sentences,
- The introduction of the new classifications in the tree search also resulted in the very good consequences.

6. Conclusion:

- The paper had been successful in introducing the continuation of the limited chain type LSTM to the tree structured LSTM. They had even introduced new branches of the tree structure based on the usage.
- It had created the convenience of handling the longer sentences for every word getting the score based on the left and the right words by the usage of hidden layers.
- The basic requirements of the sentence classification, semantic orientation and the sentiment classification, are declared with the better results than the former algorithms.