Textual Entailment and Semantic Sentence Similarity via Recursive Neural Networks

Presenter: Tianxiao Zhang, CS, UIUC
In collaboration with: Daniel Khashabi, John Wieting, CS, UIUC

Introduction/Overview

Purpose:
Investigating Recognizing Textual Entailment and semantic sentence relatedness, using Recursive Neural Networks (RNN).

Importance:
Textual Entailment is one of the killer problems in natural language processing, and one of the most important steps in understanding text.

Why RNN:
They are fast, and capture compositionality in sentences. Also, they are easy to train via standard algorithm for Neural Networks.

Textual Entailment

Consider two sentences (A) and (B). Suppose everything in the text(A) is true and using (A) we want to evaluate the truth of the claim in sentence (B).

Text: "The young boys are playing outdoors and the man is smiling nearby"
Hypothesis: "The kids are playing outdoors near a man with a smile"
Result: Entailment

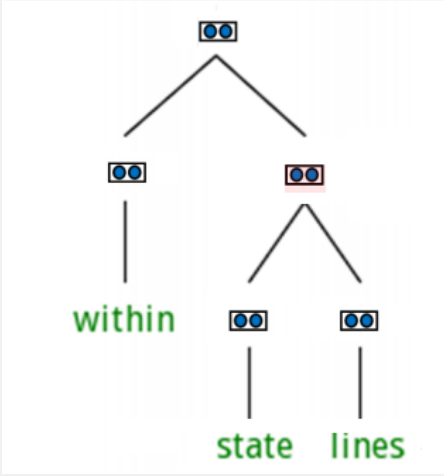
Text: "Two children and an adult are standing next to a tree limb."
Hypothesis: "Two children and an adult are not standing next to a tree limb."
Result: Contradiction

Text: "A group of kids is playing in a yard and an old man is standing in the background"
Hypothesis: "A group of boys in a yard is playing and a man is standing in the background"
Result: Neutral

Recursive Neural Networks

Suppose we have a semantic representation for each word with some vectors. For experiments we use the representation given in [2]. Given the vector representation we can parse the sentence and recursively combine the vectors so that we get to the root of the tree. The recursive combination of the vectors is done using some vector operations, which is also called a Recursive Neural Network (RNN).

Consider each sentence and its parse tree. Then recursively combine the word-vectors until getting the root vectors. This is parsing via RNN.



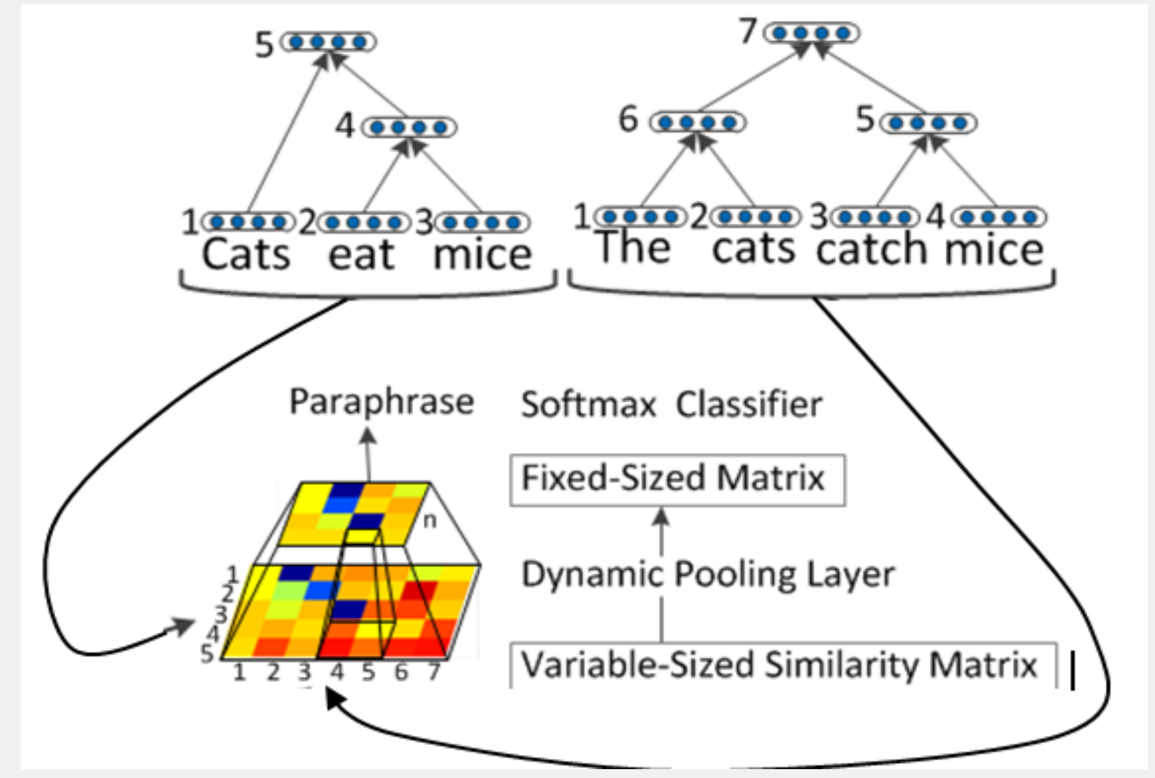
Training such a network is done via error Back-Propagation.

Sentence similarity measure

Suppose you are give two sentences and their parsed trees and vectors via RNN. How do you compare them? We need to establish a similarity measure between two sentences. This is done via creating similarity vector between word-vectors.

Since classifiers could consider only a fixed size of features, we need to pool the features [1] by random similarity feature selection.

In addition to the similarity measure we add other features that account for entailment of words, given their context. This is done using Recurrent Neural Network (under review).



Similarity matrix and dynamic pooling. .

Prediction using similarity table

Given a table of similarity values we can predict the label for the problem. For that we use logistic regression which is an algorithm for binary classification.

Since we are dealing with a multi-class classification problem (entailment, neutral, contradiction), we use 3 binary classifiers using One-Vs-All strategy.

Results

Entailment
In order to assess the performance of the system, we tested it on a recent dataset for entailment (SemEval2014 (Task1)).

System	F1	Accuracy
RNN	0.804	0.695
LCLR	0.7396	0.871
Baseline	NA	56.2

Performance of three different systems. LCLR [3] is our recent discriminative learning model . RNN is the current work which outperforms in terms of F1. Baseline is a simple algorithm based on edit distance.

Future work

Better feature pooling
Our experiments shows that random similarity feature pooling is not very accurate, although being fast. Therefore we need to develop a better strategy for pooling features into a fixed-sized feature space. In order to do that, we want to cluster features based on a shallow level of features (for example POS tags) and weight them based on this..

Better word-vector representation and language model
Another issue is that, the vector representation of the words, and the language model (RNN, which combines the word-vectors) are not fully compatible. We should look for ways to jointly train the language model and the word-vectors. We have started some works in this direction.

References

[1] Socher, Richard, et al. "Dynamic Pooling and Unfolding Recursive Autoencoders for Paraphrase Detection." *NIPS*. Vol. 24. 2011.
[2] Turian, Joseph, Lev Ratinov, and Yoshua Bengio. "Word representations: a simple and general method for semi-supervised learning." *Proceedings of* ACL, 2010.
[3] Chang, Ming-Wei, et al. "Discriminative learning over constrained latent representations." HLT, 2010.

CS @ ILLINOIS

P.U.R.E. Promoting Undergraduate Research in Engineering

CAT

Rockwell Collins