# Final Project Report for CSC 544

Zheng Tang
zhengtang@email.arizona.edu

## Introduction

Recurrent neural networks (RNNs) have been successfully proved to be one of the best tools for natural language processing (NLP) tasks and achieved perfect results than other methods. However, the lack of interpretability limits further improvements on their architectures. I found a paper written by Ming et.al. [1], which presents a visual analytics method for understanding and comparing RNN models for NLP tasks. I decide to re-implement the visualization tool described in this paper using the techniques they mentioned.

## Background Techniques and Tools

- Recurrent Neural Network
- Co-Clustering
- Comparative Visualization
- D3
- Scikit-learn
- Pytorch
- Highchart

## Achieved Goals

- Implemented the LSTM Model for POS Tagging task
- Created dataset for visualization
- Implemented the Word Clouds Layout
- Implemented the memory chips layout for hidden state clusters
- Implemented the Co-Clustering connections Layout
- Implemented the chart for visualizing word's responses according to hidden dimensions

## Approach

I first trained a LSTM model and during training I saved the expected response for each word, hidden unit pair.
I converted the expected responses to matrix and ran co-clustering on that matrix. So I got all words clusters and hidden states clusters.
I set the color scale based on the responses. I set both luminance scale and color scale, so that the visualization will appear difference between positive and negative responses, as well as map zero to a neutral color.
Based on materials above I built the visualization for RNN.

# Data and Codes

## Data:

- data.js: Prepared data for visualizing.
  - words: contains all input words. It is an array of json objects. Each includes three fields.
    - name: the word itself
    - weight: 20-Euclidean distance between the word and centroid of the cluster, where 20 is the upper bound of the distance
    - responses: the responses score to each hidden dimensions
  - hidden_units: hidden units id in clusters
  - edges: aggregate responses between a word cluster Wi and a hidden state cluster Hj
- lstm.model: pre-trained LSTM model

## Codes:

- lstm.py: lstm model training and testing. (You may need data for this, but I didn't put those huge dataset in GitHub repo)
- vectorizer.py: served for lstm.py as a vectorizing tool.
- utils.py: turn lstm.model to data we need, but I need to manually change this json format to data.js.
- data_viz.js: the visualization code to present all these data
- index.html: the front end of visualization, you just need to open it in any browser.

# Result

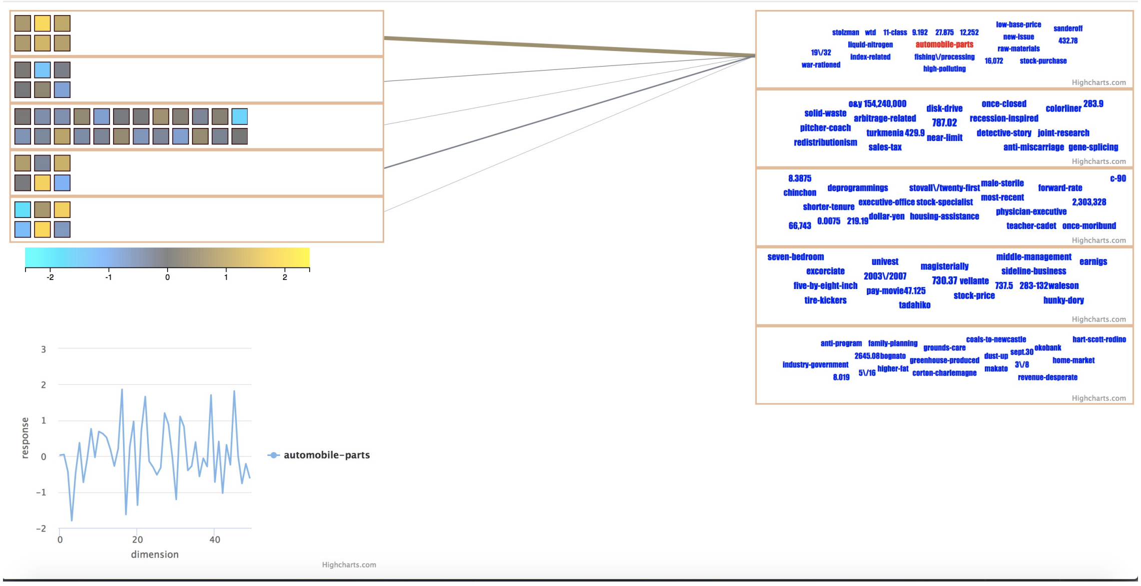
-2
-1
0
1
2
3
2
1
0
−1
−2
response
0
20
40
dimension
automobile−parts
Highcharts.com
stolzman wtd 11-class 9.192 27.875 12,252 low-base-price sanderoff new-issue 432.78 liquid-nitrogen automobile-parts raw-materials 19\/32 index-related fishing\/processing 16,072 stock-purchase war-rationed high-polluting
o&y 154,240,000 disk-drive once-closed colorliner 283.9 solid-waste arbitrage-related 787.02 recession-inspired pitcher-coach turkmenia 429.9 near-limit detective-story joint-research redistributionism sales-tax anti-miscarriage gene-splicing
8.3875 c-90 chinchon deprogrammings stovall\/twenty-first male-sterile forward-rate most-recent shorter-tenure executive-office stock-specialist 2,303,328 physician-executive 66,743 0.0075 219.19 dollar-yen housing-assistance teacher-cadet once-moribund
seven-bedroom univest middle-management earnigs excorciate magisterially sideline-business 2003\/2007 five-by-eight-inch 730.37 vellante 737.5 283-132 waleson pay-movie 47.125 stock-price tire-kickers tadahiko hunky-dory
anti-program family-planning coals-to-newcastle hart-scott-rodino grounds-care okobank 2645.08 bognato dust-up sept.30 greenhouse-produced home-market industry-government 3\/8 5\/16 higher-fat makato 8.019 corton-charlemagne revenue-desperate
Highcharts.com

Figure 1: The screenshot of my project. The top left part is the hidden states with a color legend. In its bottom, is the response chart for selected word. The right part is a interactive word clouds, you can click on any word in it. Between word clouds and hidden states is the edges represent the aggregate responses between selected word cluster and all hidden state clusters.

## Future Works

First, I want to improve the interactive functions for this visualization tool. So we can change model type and hyper parameters to train different models.
The word cloud provided by highchart is not satisfactory, so I plan to find a better tool or write my own customized word cloud generator.
Also, the Glyph for test sequence is very interesting. I'd like to add this part in the future.

## References

[1] Yao Ming, Shaozu Cao, Ruixiang Zhang, Zhen Li, and Yuanzhe Chen. Understanding hidden memories of recurrent neural networks. *dim*, 1(1.0):0–5.