Topic: **Sentiment Analysis on Movie Reviews**
Team Members: Holly Cao, Angel Fu, Laurence Ininda

1. Introduction

Sentiment analysis is the task of classifying a document, sentence, or phrase according to sentiment, often in the form of ordinal regression for example, on a five-point scale where 1 is negative, 2 is slightly negative, 3 is neutral, 4 is slightly positive and 5 is positive. Sentiment analysis can be performed on various datasets to predict or assign sentiment to a group of text such as movie reviews or email messages. For this project, the dataset will be The Rotten Tomatoes Movie Reviews originally collected by Pang and Lee. With several techniques used to perform sentiment analysis, what this dataset presents is the challenges of analyzing sentiment by overcoming obstacles such as negation, sarcasm, terseness and language ambiguity among others. Our project will focus on implementing sentiment analysis techniques from a baseline algorithm to one that tackles the challenges that our dataset presents. The classification will classify the phrases on a five-point scale with the following values: negative, somewhat negative, neutral, somewhat positive and positive.

2. Related Research Articles
    a. Discussion.

Phrases, sentences and documents contain lexical information as well as sentiment information that give the collection of words altogether meaning. Yet, unsupervised vector-based approaches to semantics largely fail to capture sentiment information that is at times crucial (Maas et al, 2013). Capturing semantic as well as sentiment information can be accomplished in various ways such as learning word vectors via an unsupervised probabilistic model of documents and extending this model with a supervised sentiment component that embraces many social and attitudinal aspects of meaning (Maas et al, 2013). However, as Socher et al. relay, semantic word spaces have been very useful but cannot express the meaning of longer phrases in a principled way. To understand compositionality in tasks, there is need for richer supervised training and evaluation resources and more powerful models of composition and thus, a model such as the Recursive Neural Tensor Network (RNTN) trained on the Stanford Sentiment Treebank – the first corpus with fully labelled parse trees attempts to capture meaning in a collection of words (Socher et al, 2013). RNTN, when compared to several supervised compositional models such as Recursive Neural Networks (RNNs), matrix-vector RNNs, and baselines such as Naïve Bayes (NB), bigram-NB and Support Vector Machines (SVM) shows to have an even higher performance score and unlike bag of words models, RNTN accurately captures complexities in sentiment such as negation that spans several words (Socher et al., 2013).

A simple sentiment analysis algorithm, such as Pointwise Mutual Information and Information Retrieval (PMI-IR) can be used to classify a review on a binary scale of either recommended or not recommended (Turner, 2001). Such an algorithm can be considered a baseline for most semantic evaluation tasks. In a lot of real-world cases, there are other instances where a document or review could be placed on finer-grained scales. By presenting a multi-class text categorization, there are exciting factors to consider classifying such text. Pang and Lee first demonstrate that humans can distinguish slight differences in rating reviews on an *n*-ary scale and then they present three types of algorithms to classify text on such an *n*-ary scale: one-vs-all, regression and metric labelling – all of which can be distinguished by how explicitly they attempt to leverage similarity between labels (Pang and Lee, 2005). Yessanalina and Cardie (2011), in assigning sentiment to documents or phrases across a polarity spectrum (a five level ordinal sentiment scale in their case), present a general learning-based approach for phrase-level sentiment analysis that adopts an ordinal scale and is explicitly compositional in nature by modelling each word as a matrix and combining words using iterated matrix multiplication (Yessenalina and Cardie, 2011). Other factors such as the length of the phrase that is analysed play a crucial role in the performance of each of these algorithms (Socher et al, 2013). It is clear that there are various techniques that can be applied towards sentiment analysis even specifically for our task. Starting from a baseline would provide a good foundation from which to build such an algorithm and analyse how it performs.

b. List of papers and summary:
    i. RNTN: Recursive Deep Models for Semantic Compositionality Over a Sentiment Treebank:

This experiment analyzes several larger quantitative evaluations of phrase sentiments, as well as two linguistic phenomena that are important in sentiment. At the same time, it introduces the Stanford Sentiment Treebank as an overall better tool compared to its full sentence labelling alternative. The experiment uses the bag of features as the baseline method and experiments on the variation of recursive model.

    ii. Learning Word Vectors for Sentiment Analysis:

This research combines unsupervised learning (of semantic which can only produce a relative scale of sentiments) and supervised learning (in sentiment which captures more social and attitudinal aspects of meaning). It offers an alternative method to the more common matrix factorization-based technique, which is the word vector induction that is proven more accurate in this research.

    iii. Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales (Pang and Lee, 2005):

The paper first demonstrates that humans can distinguish slight differences in rating reviews on an *n*-ary scale and then they present three types of algorithms – one-vs-all, regression and metric labelling – that can be distinguished by how explicitly they attempt to leverage similarity between labels. Next, Pang and Lee apply a similarity measure based on the positive-sentence percentage to the metric labelling framework showing that they show provides similar improvements over other algorithms.

    iv. Compositional Matrix-Space Models for Sentiment Analysis (Yessenalina and Cardie, 2011):

The paper describes a general approach for phrase-level sentiment analysis that takes phrase-level intensity on a spectrum by adopting a five-level ordinal sentiment scale and presenting *a learning-based method that assigns ordinal sentiment scores to phrases.*

    v. Thumbs Up or Thumbs Down? Semantic Orientation Applied to Unsupervised Classification of Reviews:

This paper presents a simple unsupervised learning algorithm for classifying reviews as recommended (thumbs up) or not recommended (thumbs down) that could also be a baseline algorithm for our model. The algorithm starts by running a POS tagger to identify phrases (JJ NN, for example) and estimate the semantic orientation of phrases in a review.

3. Strategy of Problem Solving (System Project:

Your algorithm and how you plan to implement and test it. Include both a simple version that you are confident you can complete before the deadline and a more elaborate version that you want to implement if you have time. You can implement existing algorithms or modifications of existing algorithms.)

a. Since this task is originally a challenge that has now closed, we set our baseline algorithm as the top-voted contestant's process of processing data which are:
    i. Load Dataset
    ii. Clean reviews, tokenize and lemmatize
    iii. Collect dependent values and convert to one-hot encoded output
    iv. split into train and validation sets

v. get # of unique words and max length of review available in cleaned reviews
vi. actual tokenizer of keras (a deep learning API framework he used) and convert to sequences
vii. early stopping prevent overfitting
viii. fit model/produce graph

b. Some things that could possibly be implemented depending on feasibility and result score:
   i. Combining supervised for sentiment and unsupervised for semantic learning (paper 2)
   ii. Combining/comparing data with IMDb database (paper 2, and another kaggle challenge)
   iii. Similarity score between fragments (paper 3)

4. Method of Evaluation:
   The following are the two ways in which we could evaluate our system:
   1. By submitting our system on the Kaggle submission website and obtaining a score of our system. This score will be enable us to see how our system performs against other systems that people and other teams have designed.
   2. By creating a test set from the data provided.
      For this, we would set aside some of the data from the training set for testing, remove the sentiment values and use this test set for evaluation. Since we have the answer key already provided in the data, this would enable us to perform our own evaluation.
   3. Manually annotating the test set using Amazon Turk and testing against this annotated data.
      This would involve creating a task where humans can assign sentiment to the phrases and then test our system against this manually annotated data. The advantage of this is that we would know where our system fails and have access to the answer key and also not have to split the training data as suggested above.
   4. Testing our system against other similar movie review datasets available online.

5. Collaboration:

In our group of 3, each person is going to attempt to implement the baseline algorithm following notebook instructions. Because the writer made some interesting choices, it is important to understand how the data are formed. Then each group member will pick certain subtopics and attempt to implement them. If there are two features that both improve the scores of the data, we will attempt to combine them and test the effects.