Sentiment Analysis for Amazon Reviews Dataset

Ashwin Nimhan
Indiana University, USA
animhan@indiana.edu

Vipul Munot
Indiana University, USA
vipmunot@indiana.edu

Manashree Rao
Indiana University, USA
manarao@indiana.edu

Abstract

In this paper we analyze the sentiment of review texts from amazon and identify/extract the entites mentioned in the review to provide context to the derived sentiment. For the task of sentiment analysis we experiment with various classifiers and compare their performance to identify the most effective model. We also evaluate various encoding schemes like one-hot encoding and word embeddings.

1 Introduction

Before buying a product or service, consumers often search the Web for expert reviews, but increasingly also for opinions of other consumers, expressed in blogs, social networks etc. Many useful opinions are expressed in text-only form (e.g., in tweets).User-generated reviews are thus valuable resources for decision making.

In this paper, we consider free text customer reviews of products and services. Given a set of texts discussing products, entity sentiment analysis identifies prominent entities that the reviews are about and determining the semantic orientations (positive, negative or neutral) of opinions expressed on product features in reviews. Aspect specific sentiment analysis for reviews is an important task of sentiment analysis and opinion mining.(Liu, 2012).

2 Problem Statement

In order to extract sentiments and associated entities, Amazon reviews should be analyzed in the following way:

1. The first part involves the extraction of the target entity from a sentence. It involves the following NLP tasks:
   (a) Part-of-speech tagging
   (b) Named entity recognition (NER)
2. In the second part, we assign an overall score for each review. This will allow us to structure the information from thousands of reviews and add a value to the end customer by identifying the sentiment of the review alongwith the entities/topics that the review is about.

The specific problem formulation: Given a review as form of a sentence l_i , identify the sentiments/scores for the entire sentiment and identify entities $e_{1...n}$ from the review.

Models are evaluated using F1 scores.

3 Background

3.1 Definition of main concepts

An opinion could be simply defined as a positive or negative sentiment,view, attitude, emotion, or appraisal about an entity (product, person, event,organization or topic) or an aspect of that entity from a user or group of users.

Following that defined, an opinion can be mathematically defined as a 5 tuple $(e_j, a_{jk}, so_{ijkl}, h_i, t_l)$ where e_j represents a target entity and a_jk is the k-th aspect/feature of the entity e_j. so_{ijkl} is the sentiment value of the opinion from the opinion holder h_i on aspect a_{jk} of entity e_j at time t_l. That value can be positive, negative, or neutral, or even a more granular ratingcan be used. h_i is the opinion holder and t_l is the time when the opinion was expressed (Liu, 2010).

Sentiment analysis, also called opinion mining, is the field of study that analyzes peoples opinions, sentiments, evaluations, appraisals, attitudes, and emotions towards entities such as products, services, organizations, individuals, issues, events, topics, and their attributes. The term opinion can be used to denote opinion, sentiment, evaluation,

appraisal, attitude, and emotion. Sentiment analysis and opinion mining mainly focuses on opinions which express or imply positive or negative sentiments. Sentiment Analysis has a wide arrange of applications, almost in every domain. The industry surrounding sentiment analysis has also flourished due to the proliferation of commercial applications as we now have a huge volume of opinionated data in the social media on the Web.

3.2 Applications

With the explosive growth of social media (e.g., reviews, forum discussions, blogs, micro-blogs, Twitter, comments, and postings in social network sites) on the Web, individuals and organizations are increasingly using the content in these media for decision making. Nowadays, before buying a product, one is no longer limited to asking ones friends and family for opinions because there are many user reviews and discussions in public forums on the Web about the product. For an organization, it may no longer be necessary to conduct surveys, opinion polls, and focus groups in order to gather public opinions because there is an abundance of such information publicly available. Each site typically contains a huge volume of opinion text that is not always easily deciphered in long blogs and forum postings. The average human reader will have difficulty identifying relevant sites and extracting and summarizing the opinions in them. Automated sentiment analysis systems are thus needed.

Well-known tasks for Natural Language Processing (NLP) involve general sentiment analysis for various types of texts or speech. Such models are capable to output the general sentiment of a sentence or piece of text. However, tasks such as mining product reviews require more than high level sentiment analysis, rather it requires entity level sentiment analysis (or aspect specific sentiment analysis) on the level of review-specific features (for instance the sentiment/score of display in a product review of a smart phone). Other applications are public opinion predictions, opinion mining or emotion detection. In many applications, the goal is to perform this sentiment analysis over time. This analysis can naturally be used for other tasks such as recommender systems or summarizing tasks. [War+11]In this paper, Ward proposed Empath, a new framework for evaluating entity-level sentiment analysis which leverages objective measurements of entities in various domains such as people, companies, countries, movies, and sports, to facilitate entity-level sentiment analysis and tracking and demonstrated the utility of Empath for the evaluation of a large-scale sentiment system by applying it to various lexicons using Lydia, their in-house large scale text-analytics tool, over a corpus consisting of more than a terabyte of newspaper data. [Ser+15]

3.3 Related Work

In recent years, opinion mining or sentiment analysis (Liu, 2010; Pang and Lee, 2008) has been an active research area in NLP. One task is to extract `people's` opinions expressed on features of entities (Hu and Liu, 2004). For example, the sentence, 'The picture of this camera is amazing', expresses a positive opinion on the picture of the camera. picture is the feature, how to extract features from a corpus is an important problem. There are several studies on entity extraction (e.g., Hu and Liu, 2004, Popescu and Etzioni, 2005). Previous researches have proposed several models to address this task.

There are ATE (Aspect Term Extraction) datasets publicly available but according to (Pavlopoulos and Androutsopoulos et al.,)these datasets are not entirely satisfactory, mostly because they contain reviews from a particular domain only (e.g., consumer electronics), or they contain reviews for very few target entities, or they do not contain annotations for aspect terms.

The method of Hu and Liu (2004), dubbed H & L , first extracts all the distinct nouns and noun phrases from the reviews of each dataset (Algorithm 1) and considers them candidate distinct aspect terms. It then forms longer candidate distinct aspect terms by concatenating pairs and triples of candidate aspect terms occurring in the same sentence, in the order they appear in the sentence (lines 711). For example, if battery life and screen occur in the same sentence (in this order), then battery life screen will also become a candidate distinct aspect term. We can also have other patterns like NP + Prep + CP or CP + with + NP where Noun/noun phrase (NP) contains the part word and the class concept phrase (CP) contains the whole word.

This method was further extended by Pavlopoulos and Androutsopoulos by including an additional pruning step that uses continuous vector

space representations of words (Mikolov et al., 2013a; Mikolov et al., 2013b; Mikolov et al., 2013c). The vector representations of the words are produced by using a neural network language model, whose inputs are the vectors of the words occurring in each sentence, treated as latent variables to be learned and candidate distinct aspect terms whose vector was closer to the common language centroid than the domain centroid were discarded.

Double Propagation (Qiu et al., 2009) is a state-of-the-art unsupervised technique for solving the problem. It mainly extracts noun features, and works well for medium-size corpora. The extraction rules are designed based on relations described in dependency trees. But for large corpora, this method can introduce a great deal of noise (low precision), and for small corpora, it can miss important features.

Zhang, Liu, Lim and OBrien-Strain propose a new feature mining method, which enhances that in (Qiu et al., 2009). Firstly, two improvements based on part-whole patterns and no patterns are introduced to increase recall. Part-whole or meronymy is an important semantic relation in NLP, which indicates that one or more objects are parts of another object. no pattern is another extraction pattern. Its basic form is the word no followed by a noun/noun phrase, for instance, no noise as people often express their short comments or opinions on features using this pattern. Both types of patterns can help find features missed by double propagation.

Wong and Lam (2008) employ hidden Markov models and conditional random fields for extracting product features from auction websites. Here they have formulated the tasks of product feature extraction and hot item feature summarization as a single graph labeling problem using conditional random fields and a characteristic of this graphical model is that it can model the inter-dependence between neighbouring tokens in a Web page, tokens in different Web pages, as well as various information such as hot item features across different auction sites.

To extract entities and chunking, we will be using the chunking module from nltk which builds upon this double propogation method and Tagging and Partial Parsing method (Church, Young, and Bloothooft., 1996)

For Rule Based Classifier systems, Chikersal,Poria,Cambria have created a system based on rules that are dependent on the occurrences of emoticons and opinion words in tweets. Each tweet is split into n-grams, processed using TF-IDF and POS tagged and then has to be labeled as 'positive', 'negative' or 'neutral' and a Support Vector Machine (SVM) (Cortes and Vapnik,1995) is used for training on the tweets. Whereas, Chin-Sheng Yang and Hsiao-Ping Shih are using search keywords' occurence to extract the candidate opinion sentences (COS_j) of product feature f_j from the dataset. The set of candidate opinion sentences are assumed as review sentences which express opinions on product feature f_j and applied for the subsequent rule learning task. To learn the desired set of product feature extraction rules from the candidate opinion sentences COS j automatically, they have adopted the class association rule mining algorithm (Yang et al. 2010) as the underlying technique. For sentiment analysis their proposed approach integrates the lexicon-based approach (Hu I& Liu 2004) and the set of product feature extraction rules $PFE - R_j$.

Deep learning models have achieved remarkable results in computer vision (Krizhevsky et al., 2012) and speech recognition (Graves et al., 2013) in recent years. Within natural language processing, much of the work with deep learning methods has involved learning word vector representations through neural language models (Bengio et al., 2003; Yih et al., 2011; Mikolov et al., 2013) and performing composition over the learned word vectors for classification (Collobert et al., 2011). Recursive Neural Networks have recently obtained state of the art performance on several natural language processing tasks. (Paulus I& Socher et al., 2014) have used global belief recursive neural networks (GB-RNNs) which are based on the idea of extending purely feedforward neural networks to include one feedbackward step during inference which allows phrase level predictions and representations to give feedback to words. Coulbert et.al have designed a system for multiple benchmarks while avoiding task-specific engineering and have used a single learning system able to discover adequate internal representations and have implemented NLP tasks like Part-Of-Speech Tagging, Chunking, Named Entity Recognition, Semantic Role Labeling from scratch using neural networks.

4 Data Collection and Labeling

4.1 Data Collection

We have selected product review datasets provided by Julian McAuley, UCSD. This dataset contains product reviews and metadata from Amazon, including 142.8 million reviews spanning May 1996 - July 2014. This dataset includes reviews (ratings, text, helpfulness votes), product metadata (descriptions, category information, price, brand, and image features), and links (also viewed/also bought graphs). A subset of 100,000 reviews from electronics category is taken for evaluation.

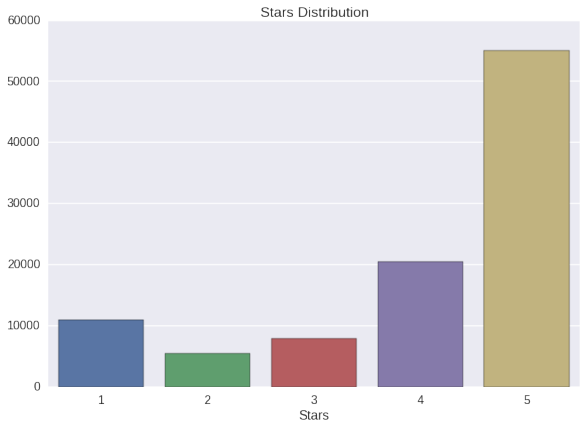
4.2 Pre-processing

For this project our focus is on only two columns namely, 'reviewText' and 'overall'. However the reviews set has rating stars not sentiment associated with it. Hence, we have processed the data and associated 'positive' for reviews with rating 4 or 5, 'negative' for reviews with rating 1 or 2 and 'neutral' for reviews with rating 3. We have processed the data and finally it has only 2 columns, 'labels' and 'text' where 'labels' have the sentiment and 'text' has the review text. Entire process is automated, data is stored, extracted using MongoDb, pre-processed using Pandas(python).

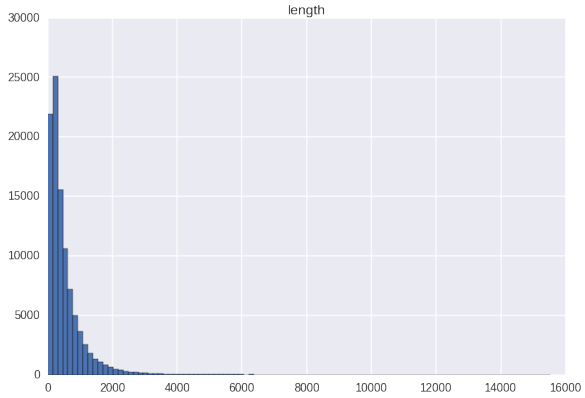
4.3 Exploratory Data Analysis of data

We have analyzed the 2 attributes of data that we will be working with, i.e., distribution of ratings and distribution of review text.

Distribution of labels



Distribution of messages



5 Method and Data Splitting

5.1 Data Splitting

We have used top 100000 reviews in our system. Data has been split into 80% Train,10% Dev, 10% Test There are 80,000 reviews in training set, 10000 for development set and 10000 for final test set.

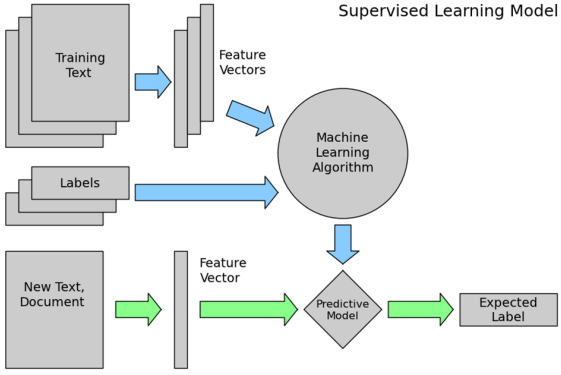
5.2 Textual Data to vectors

We are converting each review, represented as a list of tokens into a vector that machine learning models can understand which essentially requires three steps, in the bag-of-words model:

1. counting how many times does a word occur in each message (term frequency)
2. weighting the counts, so that frequent tokens get lower weight (inverse document frequency)
3. After the counting, the term weighting and normalization can be done with TF-IDF, using scikit-learn's TfidfTransformer.

5.3 Machine Learning Techniques

Once the data is converted to vectors as above; ML techiniques are employed.



We are using the following different classifiers to analyze document level sentiment:

1. BernoulliNB:
   Naive Bayes classifiers are a family of simple probabilistic classifiers based on applying Bayes' theorem with strong (naive) independence assumptions between the features. Naive Bayes classifier for multivariate Bernoulli models are suitable for discrete data.

BernoulliNB				
	Precision	Recall	F1-Score	Support
negative	0.64	0.44	0.52	3291
neutral	0.13	0.15	0.14	1584
positive	0.83	0.88	0.85	15125
avg / total	0.74	0.75	0.74	20000
accuracy score	74.69%			

2. Random Forest:
   Random forests is a notion of the general technique of random decision forests that are an ensemble learning method for classification, regression and other tasks, that operate by constructing a multitude of decision trees at training time and outputting the class that is the mode of the classes (classification) or mean prediction (regression) of the individual trees. Random decision forests correct for decision trees' habit of overfitting to their training set. In sklearn, a random forest is a meta estimator that fits a number of decision tree classifiers on various sub-samples of the dataset and use averaging to improve the predictive accuracy and control over-fitting. The sub-sample size is always the same as the original input sample size but the samples are drawn with replacement.

Random Forest				
	Precision	Recall	F1-Score	Support
negative	0.9	0.17	0.28	3250
neutral	1.00	0.00	0.00	1582
positive	0.78	1.00	0.88	15168
avg / total	0.82	0.78	0.71	20000
accuracy score	78.43%			

3. SVM linear kernel
   Linear models have linear decision boundaries (intersecting hyperplanes). SVM linear kernel is used as number of features is larger than number of observations. Here features are the word vectors generated.

   Choice of Linear Kernel

   (a) As text is often linearly separable, Linear kernel works well with linearly separable data

   (b) Text has a lot of features. The linear kernel is good when there is a lot of features. That's because mapping the data to a higher dimensional space does not really improve the performance. In text classification, both the numbers of instances (document) and features (words) are large.

   (c) Training a SVM with a linear kernel is faster than with another kernel.

Support Vector Machine Linear				
	Precision	Recall	F1-Score	Support
negative	0.9	0.17	0.28	3191
neutral	0.22	0.15	0.14	1552
positive	0.78	1.00	0.88	15257
avg / total	0.82	0.78	0.71	20000
accuracy score	65.23%			

4. Naive Bayes
   GaussianNB implements the Gaussian Naive Bayes algorithm for classification. The likelihood of the features is assumed to be Gaussian distributed.

Gaussian Naive Bayes				
	Precision	Recall	F1-Score	Support
negative	0.33	0.48	0.28	3271
neutral	0.13	0.15	0.14	1224
positive	0.54	0.47	0.5	15125
avg / total	0.37	0.21	0.25	20000
accuracy score	34.69%			

5. BernoulliRBM (neural network)
   A restricted Boltzmann machine (RBM) is a generative stochastic artificial neural network that can learn a probability distribution over its set of inputs. The BernoulliRBM implementation consists of binary visible units and binary hidden nodes and the elements in the visible and in the hidden layers are assumed to be Bernoulli distributed. Parameters are estimated using Stochastic Maximum Likelihood (SML), also known as Persistent Contrastive Divergence (PCD).

BernoulliRBM				
	Precision	Recall	F1-Score	Support
negative	0.9	0.17	0.28	3500
neutral	0.22	0.15	0.14	1650
positive	0.78	1.00	0.88	14843
avg / total	0.75	0.80	0.71	20000
accuracy score	75.23%			

5.4 Rule Based Classifier

Rule based classifier:

For the rule-based classifier we break down our problem as multi-class classification problem, where each review is to have sentiment score of the review. This is also supervised classifier because we have the number of stars for each review. The sentiment score is calculated by following rules: Emotion-related Rules: Each word in review is

compared with the positive and negative dictionaries from Bing Liu Lexicon and then labeled as either "positive" or "negative". Now if the word is labeled positive then the sentiment score of the review is increased by 1 and if the word is labeled negative then the sentiment score of the review is decreased by 1.

Intensity-related Rules: If the sentences contains words such as "barely good" or "too good". Then the sentiment of the sentences is more positive rather than negative or vice versa, meaning the intensity of the sentence changes due to use of such words. Hence, we created rules to check the intensity of the sentiment in the review. If the review contains words like "too", "very" and "sorely" then the sentiment score is multiplied by 2 and if the review contains words like "barely", "little" and so on then the sentiment score is divided by 2.

Negation-related Rules: If we are using lexicon based sentiment score then, sentences like "The Ipod is not good." will give positive sentiment score. In fact, the sentiment is negative. Hence, we also included lexicon for negation, inverters and polarity flips. If the review contains words such as not,not so,etc we are multipling the sentiment score with -1.

Rule Based Classifier	
Label	F1-Score
Negative	0.482
Neutral	0.161
Positive	0.643
Accuracy score	60.05%

5.5 Deep Learning(CNN) using word-embeddings method for Sentiment Classification

Statical models with BoW word representations face problems like word ordering information loss, data sparsity and difficulty in representation of higher level features. Word-embeddings aleviate this problems and also successfully capture relationships and compositionality in some sense. Convolutional Neural Networks for NLP involve encoding the sentences as a matrix using the word-embeddings to generate a "sentence matrix". Multiple region sizes(3) are defined and used as convolution windows over the sentence matrix with different fitlers(2) giving 6 filters in total. Filters are used to generate variable-length feature maps using an activation function. The feature maps are then processed using 1-max pooling to identify max feature value. The generated univariate feature vector is then applied a softmax to achieve a multi-class sentence classification. The model used is forked from github and modified to suit the model.

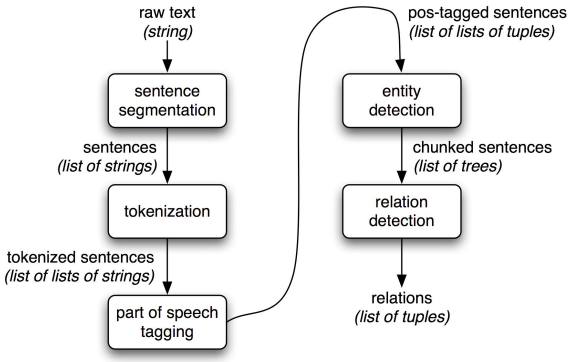


Figure 1: Information extraction pipeline

Deep Learning Model	
Label	F1-Score
Negative	0.8071
Neutral	0.19
Positive	0.9423
Accuracy score	88.45%

5.6 Identifying entities/aspects in reviews

This system takes the raw text of a document as its input, and generates a list of (entity, relation, entity) tuples as its output. To perform the first three tasks, a simple function is used that connects together NLTK's default sentence segmenter [1], word tokenizer [2], and part-of-speech tagger [3]. In named entity detection(NER), we segment and label the entities that might participate in interesting relations with one another. Typically, these will be definite noun phrases, or proper names or nouns or noun chunks.

Finally, in relation extraction, we search for specific patterns between pairs of entities that occur near one another in the text, and use those patterns to build tuples recording the relationships between the entities.

These entities are added to the output of the reviews where final output consists of overall document level sentiment analysis, and associated entities.

6 Evaluation

The models are trained on 'train.csv' and tested on development dataset 'development.csv' We are using Precision, Recall, F1-Score, Support and Ac-

curacy to evalaute our models. The baseline that we are using is the evaluation metric given by Naive Bayes classifier.

Results	
Precision	Recall
Naive Bayes	34.67
BernoulliNB	75.23
Random Forest	78.43
Linear SVM	65.23
BernoulliRBM	75.23
Deep Learning using word-embedding	88.45

As Deep learning method out-performs all other machine learning algorithms, we proceed with this method and evaluate the final performance on the test set called 'test.csv'

Deep Learning Model on unseen test data	
Label	F1-Score
Negative	0.8183
Neutral	0.2116
Positive	0.9423
Accuracy score	88.905%

The sample output from identifying named entities is as follows:

```
I've been in the technical end of
the computing business for over
20 years done a lot of networking
thought it would be nice to add
wireless capability
to my home network, so purchased
the WAP11
& WPC11(the pcmcia wireless card
goes with it. Software installed
ok & got it to work. The problem
is that the WAP11 has no useful
range. As long as you are
located in the same room and
have "line of sight" with WAP11,
it's ok. If you go out the room,
or any way of the line of sight,
even the next room, the signal
quality
goes to poor and the link drops.
I have
regular, normal wood and sheet
rock walls.
Packing mine back up for returning.
doesn't work as advertised.
Very dissapointing.
 entities:  pcmcia wireless card
 entities:  WAP11
 entities:  WPC11
 entities:  Software
```

7 Discussion and Conclusion

We have observed that the Deep Learning model outperforms all other classifiers even rule-based. The limitations of our work are that we haven't associated sentiments with specific aspects which is what we want to do in the future. We also want to do a fine-grained classifier using deep learning methods. We also want to associate other metadata with the reviews data like 'person who purchased this also bought' would help identify aspects of the same product and give more information about the user who is interested in the product and other information.

Acknowledgments

We thank Dr.Abdul Mageed for technical assistance. We also thank Proffesor Julian McAuley for providing this dataset.

References

Hu, Mingqin and Bing Liu. 2004. *Mining and Summarizing Customer Reviews*
In Proceedings of KDD 2004

Pang, Bo., Lillian Lee. 2008. *Opinion Mining and Sentiment Analysis*
Foundations and Trends in Information Retrieval pp. 1-135 2008

Liu, Bing 2010. *Sentiment analysis and subjectivity*
Handbook of Natural Language Processing, second edition, 2010

Popescu, Ana-Maria and Oren, Etzioni 2005. *Extracting product features and opinions from reviews*
In Proceedings of EMNLP, 2005.

In K. Church, S. Young, and G. Bloothooft *Tagging and Partial Parsing*

Empath: A framework for evaluating entity-level sentiment analysis
In: Emerging Technologies for a Smarter World

Jesus Serrano-Guerrero et al. 2015 *Sentiment analysis: A review and comparative analysis of web services*
In: Information Sciences 311

John Pavlopoulos and Ion Androutsopoulos *Aspect Term Extraction for Sentiment Analysis: New Datasets, New Evaluation Measures and an Improved Unsupervised Method*
ACL 2014

Lei Zhang, Bing Liu Bing, Lim Suk Hwan, Eamonn OBrien-Strain *Extracting and Ranking Product Features in Opinion Documents*
COLING '10 Proceedings of the 23rd International Conference on Computational Linguistics

Nitish Gupta, Shashwat Chandra *Product Feature Discovery and Ranking for Sentiment Analysis from Online Reviews*
IIT Kharagpur

Nitish Gupta, Shashwat Chandra *Learning to extract and summarize hot item features from multiple auction Web sites*
Knowledge and Information Systems

Jesus Serrano-Guerreroa, Jose A. Olivasa, Francisco P. Romeroa, Enrique Herrera-Viedmab *Sentiment analysis: A review and comparative analysis of web services*
Information Sciences Volume 311

Chin-Sheng Yang, Hsiao-Ping Shih *A Rule-Based Approach For Effective Sentiment Analysis*
PACIS Proceedings

Collobert, Weston, Bottou, Karlen, Kavukcuoglu, Kuksa *Natural Language Processing (Almost) from Scratch*
The Journal of Machine Learning Research

Ronan Collobert *Deep Learning for Efficient Discriminative Parsing*
Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics

Romain Paulus, Richard Socher, Christopher Manning 2014. *Global Belief Recursive Neural Networks*
Advances in Neural Information Processing Systems

http://www.nltk.org/book/ch07.html

http://colah.github.io/posts/2014-07-NLP-RNNs-Representations

8 Group Members

Group Members: Provide a blurb about each member, their interests and photo, etc.

8.1 Vipul Munot

Vipul Munot is Strategic, multidisciplinary enthusiast and vivid explorer with an eye for innovation and pixel perfection. My skill set is vast, my greatest expertise revolve in the worlds of interactive design, social media, business analysis and business intelligence. My wish is to combine my knowledge and experience in these areas, to deliver the best creative to my employers clients and their audiences. I love exploring new things, travelling, trekking, music and coffee.

8.2 Ashwin Nimhan

Data Engineer, NLP and Machine Learing Enthusiast, Web Engineer with a breadth of expertise. Enjoys to design, develop and deploy efficient full-stack data-driven solutions.

8.3 Manashree Rao

A first year Graduate student pursuing Masters in Data Science at Indiana University. My fields of interest are Data Mining, Data Engineering, Web Development and Information Visualization.