**SYNOPSIS**

**ON**

**“Comments Analysis and Implementation”**

Submitted in

Partial Fulfillment of requirements for the award of Degree

*Of*

Bachelor of Technology

*In*

Computer Science and Engineering

*By*

**(Project ID:21B08)**

**Kushagra Manoj Gupta (CS) - 1716410129**

**Manmohan Krishna (CS) - 1716410136**

**Kushal Omar (CS) - 1716410131**

**Kushagra Sharma (CS)- 1716410130**

**Batch (2017-2021)**

Under the supervision of

**Dr. Harsh Dev**

**(Professor)**



**Pranveer Singh Institute of Technology**

Kanpur-Agra- Delhi National Highway-2

Bhauti – Kanpur – 209305

(Dr. A.P.J. Abdul Kalam Technical University)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Signature of the Guide**

**Acknowledgment**

Every project big or small is successful largely due to the effort of a number of people who have always given their valuable advice or a helping hand to resolve problems. I sincerely appreciate the inspiration; support and guidance of all those people who have been taking efforts in making this project a success.

We have to express our appreciation to the **Prof. Dr. Harsh Dev** for sharing her pearls of wisdom and precious time with me during the course of this research. I would also like to thank all the faculty members of **Pranveer Singh Institute of Technology** for their critical advice and guidance without which this project would not have been possible.

Last but not the least I place a deep sense of gratitude to my family members and my friends who have been constant source of inspiration during the preparation of this project work.

**Synopsis**

1. **Introduction**

Rapid evolution in technology and internet brought us to the era of ecommerce. Ecommerce is nothing but selling and buying goods or services online. Many customers share their good or bad opinions about products or services online nowadays. These opinions become a part of decision making process to make impact on business model. Also understanding and considering reviews, will help to gain trust of customer which will help to expand business. Thus, we need to study and analyze customer reviews.

Usually reviews are given in text format. Single product has number of reviews. It is hardly possible to read each review in detail. Many researches shown pictorial representation is more effective and can be memorized, understood easily rather than textual representations. So if we are going to convert textual reviews into visual format, it will enhance reliability in decision making.

Existing work shows that various approaches are used for sentiment analysis like machine learning, corpus based, NLP based or even based on clustering. Also few researches consider neutral reviews for analysis. Many of them do not have visual representations for end results or complex visual representations which are not user oriented.

The scope of research work does not include:

1. Consideration of emotions
2. Work aims only to evaluate expressions about product not the quality of product.

Thus, the dissertation work will provide novel approach based on hadoop environment to provide visual representation of sentiment analysis results applied to online product review given by customer.

Nowdays people have lots of friends on their social media account so when they share their post on social media it is not possible to read each and every comment because it will take time.So this project will help in these types of situations where user need not to read all comments and will get an average of comments that is whether its post has been liked by their friends or not.This project is also very helpful for celebrities for tracing those accounts who troll them .Basically this project will tell you whether your post is liked by your friends or not.

Comment analysis is a new field of research born in Natural Language Processing (NLP), aiming at detecting subjectivity in text and/or extracting and classifying opinions and

sentiments. Sentiment analysis studies people’s sentiments,

opinions, attitudes, evaluations, appraisals and emotions towards services, products, individuals, organizations, issues, topics events and their attributes . In comment analysis text is classified according to the following different criteria:

• the polarity of the sentiment expressed (into positive, negative, and neutral);

• the polarity of the outcome (e.g. improvement

* **Agree or disagree with Topic**
* **Good or bad news**
* **Support or Opposition**
* **Pros and Cons**
  1. **Objective and Motivation Towards The Project**

Internet is rich source of reviews on ecommerce products or online services. Customer always prefers to read reviews before paying money to service provider. But it is hardly possible to read all reviews in today's fast life. Also every review may provide new information of product or feature of product. So there is probability of missing any important review given by consumer.

We need to identify polarity of review i.e. whether it is positive, negative or neutral. Comment analysis will assist us to find out polarity of reviews. Due to large number of reviews, we can use hadoop environment for the analysis of data. Visuals can be 80% efficient than textual format. So if we visualize all the reviews, it will make easier decision making process for consumer. Consumer will be able to see all reviews at a glance and he/she will take decision faster.

Thus, our main objectives are:

1. Dealing with neutral reviews: Output must consider neutral reviews as they make impact on decision making.
2. Improved Efficiency: Many reviews are given for single product. Using map-reduce environment we can improve efficiency of sentiment analysis.
3. Comment Analysis: To determine attitude of mass people towards particular product or service.
4. User oriented Data Visualization: Customers are mainly nontechnical persons. So we aim to visualize results in user readable format.
5. **Methodology of the Project**
6. **Data Extraction**

We are going to use [www.amazon.in](http://www.amazon.in) as our source for data extraction. Reviews on Kindle are extracted using crawler implemented using Beautiful Soup library . It automatically extracts all reviews on a single product using one seed URL. Then crawler will navigate through webpages to extract all the reviews. Extracted reviews are classified into two datasets which are used for training and testing. 70% of reviews are used for testing, rest of 30% reviews are used for testing.

1. **Data Preprocessing**

Data preprocessing includes proper fragmentation of data and cleaning of data. Here, in research work we are going to use NLP preprocessing techniques like removal of stop words, chunking data, stemming etc. Data preprocessing will lead us to robust data which has less noise. For data preprocessing, use of Natural Language Tool Kit (NLTK) library implemented in python is considered. NLTK is a platform for natural language processing developed in python.

1. **POS tagging**

Part of Speech (POS) tagging assists us to identify actual part of sentence which has expression or feelings. It deals with Word Sense Disambiguaty (WSD) . Single word may have one or more tags.

For example:

The back door = back tagged as adjective

On my back = back tagged as noun

Thus, POS tagging is used to determine single tag for instance of word.

WordNet is a large English lexical database. WordNet consists of four POS tags Nouns, verbs, adjectives and adverbs, with few cross-POS pointers. WordNet also provides set of synonyms, antonyms. It find out all possible POS tag for instance of word and we have to choose proper tag using cross validation. For cross validation Penn Treebank Tagger is used. It consists of total 36 tags.

The output of POS tagging is expected as below:

For example:

Input: And now for something completely different

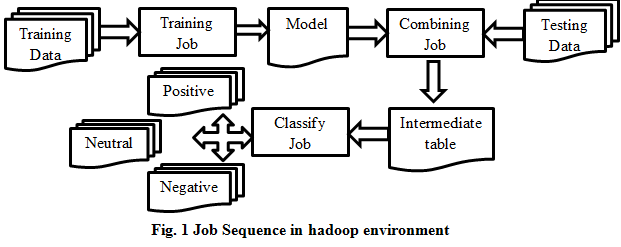
Output: [('And', 'CC'), ('now', 'RB'), ('for', 'IN'), ('something', 'NN'),

('completely', 'RB'), ('different', 'JJ')]

1. **Text Classification**

The process of text classification is divided into two stages: Training stage and testing stage. In training stage, classification model is created using testing dataset. In testing stage, accuracy of classification is evaluated using classification module.

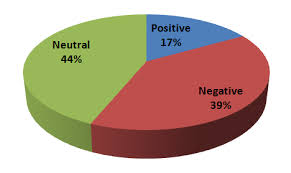
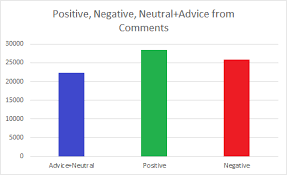
As we are going to use map-reduce environment to apply Naïve-Bayes classifier , the processing is carried out as follow:

****

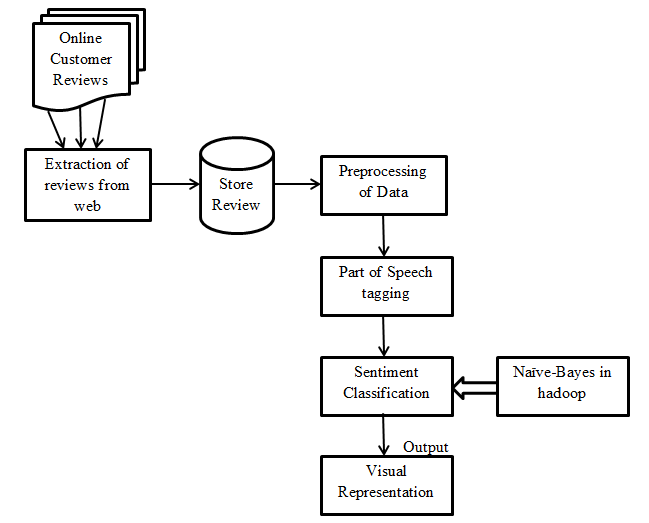
Mapper class will assign sentiment values and reducer class will evaluate polarity. Training job will create model. Combining job will combine model and test data. Classify job will perform polarity determination.

1. **Data visualization**

The end results are displayed using graphs and charts. We can use tag cloud which is built using frequency of occurrence of words. For data visualization we are going to use Python GUI Tinker open source library developed specially for data visualization. We will focus on visual representation which can be easily understood by non-technical people. Visual representations may look like:



Thus, the proposed system comprises of different techniques is as shown below:



**Fig.1 Proposed System Architecture**

|  |  |  |
| --- | --- | --- |
| **1.** | **Data Source** | [www.amazon.in](http://www.amazon.in) |
| **2.** | **Data Extraction** | Beautiful Soup python library |
| **3.** | **Preprocessing Data** | Natural Language Tool Kit(NLTK) python library |
| **4.** | **POS tagging** | Penn Treebank POS tagger/Wordnet |
| **5.** | **Text Classification** | Naïve –Bayes classifier(Map-reduce variant) |
| **6.** | **Data Visualization** | Python GUI Tinker for visualization |

**Table 1: Overview of tools and techniques to be used**

1. **Modules in the Project**

The user who has to deal with numerous reaction of people on his/her post will find it easier to know the summary of the whole so that he/she can act accordingly inorder to do gain peoples favor in their next post.This will ultimately save the user’s time by not tallying again and again rather this will do all the work and provide the result.

This will create a fast and error free way for these users .There are many studies that provide methods and tools used for sentiments analysis. The most used tools for detecting the

feelings polarity (negative and positive affect) of a message is based on the emoticons. Emoticons are face-based and symbolize sad or happy feelings, although there are a wide

range of non-facial variations. To extract the feelings polarity from emoticons, different set of common emoticons can be used (http://messenger.msn.com/Resource/Emoticons.aspx; http://www.cool-smileys.com/text-emoticons;http://messenger.yahoo.com/features/emoticons). Another method is the Linguistic Inquiry and Word Count that allows analysing not only positive and negative but also emotional, cognitive, and structural components of a text based on the use of a dictionary containing words and their classified categories.

For example, the word “agree” belongs to the word

This software is available at http://www.liwc.net/. Happiness Index is a sentiment

scale that uses the popular Affective Norms for English Words (ANEW) . It gives scores for a given text between 1 and 9, indicating the amount of happiness.It is based on an English lexical dictionary called WordNet that ghaters adjectives, nouns, verbs etc. into synonym sets called synsets. Each synset is associated to three numerical scores Pos(s), Neg(s), and Obj(s) which indicate how positive, negative, and

“objective” (neutral) the terms contained in the synset are.

The scores, which are in the values of [0, 1] and add up to 1, are obtained using a semi-supervised machine learning method. The tool, used in opinion mining, is based on

WordNet an English lexical dictionary that collect nouns, verbs, adjectives and other grammatical classes into synonym sets (synsets). Another tool is the PANAS-t . The tool consists of an adapted version of the Positive Affect Negative Affect Scale (PANAS) , method used in psychology. The PANAS-t tracks increases or decreases in sentiments over time; it is based on a large set of words associated with eleven moods: joviality, assurance, serenity, surprise, fear, sadness, guilt, hostility, shyness, fatigue, and attentiveness. This method computes the score for each

sentiment for a given time period as values between [−1.0,

to indicate the change. The open source tool SailAil Sentiment Analyzer (SASA) was evaluated with 17,000 labeled tweets on the 2012 U.S. Elections. It was evaluated

also by the Amazon Mechanical Turk (AMT), where

“turkers” were invited to label tweets as positive, negative, neutral, or undefined. The SASA python package version

0.1.3 is available at https://pypi.python.org/pypi/sasa/0.1.3The tool explores artificial intelligence and semantic Web techniques. It uses Natural Language Processing (NLP) techniques to infer the polarity of common sense concepts from natural language text at a semantic level, rather than at the syntactic level. SenticNet was tested to measure the level of polarity in opinions of patients about the National Health Service in England . SenticNet version 2.0 is available at http://sentic.net/. In EWGA and FRN tools are used. The EWGA tool uses an entropy-weighted genetic algorithm for an efficiently selection of features for sentiment classification using a wrapper-model. While the FRN uses a feature relation network considering two syntactic n-gram relations: parallel relations and subsumption . Sentiment140 formerly known as Twitter Sentiment discovers the positive and negative opinions and sentiment of a brand, product, or topic on Twitter. This tool uses classifiers built from machine learning algorithms. Unlike other tools that show aggregated numbers which makes it difficult to assess how accurate their classifiers are, this tool is able to classify individual tweets. The NRC Hashtag Sentiment Lexicon (version 0.1) is a list of words with associations to positive and negative sentiments. The lexicon is distributed in three files: unigrams-pmilexicon.txt, bigrams-pmilexicon.txt, and pairs-pmilexicon.txt. The NRC Emotion Lexicon is comprised of frequent English nouns, verbs, adjectives, and adverbs annotated for eight emotions (joy, sadness, anger, fear, disgust, surprise, trust, and anticipation) as well as for positive and negative sentiment.

1. **Technologies in the Project**

|  |  |  |
| --- | --- | --- |
| **1.** | **Data Source** | [www.amazon.in](http://www.amazon.in) |
| **2.** | **Data Extraction** | Beautiful Soup python library |
| **3.** | **Preprocessing Data** | Natural Language Tool Kit(NLTK) python library |
| **4.** | **POS tagging** | Penn Treebank POS tagger/Wordnet |
| **5.** | **Text Classification** | Naïve –Bayes classifier(Map-reduce variant) |
| **6.** | **Data Visualization** | Python GUI Tinker for visualization |

1. **Research material**

* **Beautiful Soup**

The Beautiful Soup is a python library. Beautiful Soup is a python package and as the name suggests, parses the unwanted data and helps to organize and format the messy web data by fixing bad HTML and present to us in an easily-traversible XML structures.

## **What is web-scraping?**

Scraping is simply a process of extracting (from various means), copying and screening of data.

When we do scraping or extracting data or feeds from the web (like from web-pages or websites), it is termed as web-scraping.

Web-scraping is used in an enterprise in a variety of ways –

### 1. **Data for Research**

### 2. **Products prices & popularity comparison**

### 3. **SEO Monitoring**

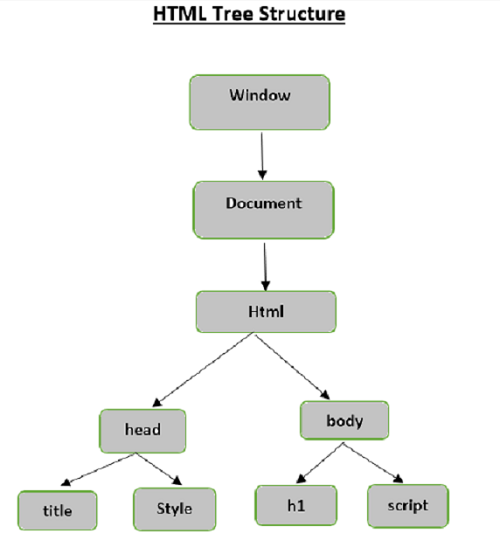
### 4. **Search engines**

### 5. **Sales and Marketing**

# **Souping the Page**

## **HTML tree Structure**

Before we look into different components of a HTML page, let us first understand the HTML tree structure.



The root element in the document tree is the html, which can have parents, children and siblings and this determines by its position in the tree structure. To move among HTML elements, attributes and text, you have to move among nodes in your tree structure.

# **Beautiful Soup - Kinds of objects**

When we passed a html document or string to a beautifulsoup constructor, beautifulsoup basically converts a complex html page into different python objects. Below we are going to discuss four major kinds of objects:

* Tag
* NavigableString
* BeautifulSoup
* Comments
* **Neural Network**

This paper describes our deep learning system for sentiment analysis of tweets. The main contribution of this work is a new model for initializing the parameter weights of the convolutional neural network, which is crucial to train an accurate model while avoiding the need to inject any additional features. Briefly, we use an unsupervised neural language model to train initial word embeddings that are further tuned by our deep learning model on a distant supervised corpus. At a final stage, the pre-trained parameters of the network are used to initialize the model. We train the latter on the supervised training data recently made available by the official system evaluation campaign on Twitter Sentiment Analysis organized by Semeval-2015. A comparison between the results of our approach and the systems participating in the challenge on the official test sets, suggests that our model could be ranked in the first two positions in both the phrase-level subtask A (among 11 teams) and on the message-level subtask B (among 40 teams). This is an important evidence on the practical value of our solution.

**OUR DEEP LEARNING MODEL FOR SENTIMENT CLASSIFICATION**

The architecture of our convolutional neural network for sentiment classification .It is mainly inspired by the architectures used in for performing various sentence classification tasks. Given that our training process requires to run the network on a rather large corpus, our design choices are mainly driven by the computational efficiency of our network. Hence, different from, which presents an architecture with several layers of convolutional feature maps, we adopt a single level architecture. Nevertheless, single-layer architectures have

been shown in to perform equally well. Our network is composed of a single convolutional layer followed by a non-linearity, max pooling and a soft-max classification layer. In the following, we give a brief explanation of the main components of our network: sentence matrix, activations, convolutional, pooling and softmax layers. We also describe how to adapt the network for predicting sentiment of phrases inside the tweets.

* **Tensor Flow**

## **Tensors**

"A tensor is a generalization of vectors and matrices to potentially higher dimensions. Internally, TensorFlow represents tensors as n-dimensional arrays of base datatypes." (<https://www.tensorflow.org/guide/tensor>)

It should't surprise you that tensors are a fundemental apsect of TensorFlow. They are the main objects that are passed around and manipluated throughout the program. Each tensor represents a partialy defined computation that will eventually produce a value. TensorFlow programs work by building a graph of Tensor objects that details how tensors are related. Running different parts of the graph allow results to be generated.

Each tensor has a data type and a shape.

**Data Types Include**: float32, int32, string and others.

**Shape**: Represents the dimension of data.

Just like vectors and matrices tensors can have operations applied to them like addition, subtraction, dot product, cross product etc.

In the next sections we will discuss some different properties of tensors. This is to make you more familiar with how tensorflow represnts data and how you can manipulate this data.

### **Creating Tensors**

Below is an example of how to create some different tensors.

You simply define the value of the tensor and the datatype and you are good to go! It's worth mentioning that usually we deal with tensors of numeric data, it is quite rare to see string tensors.

For a full list of datatypes please refer to the following guide.

<https://www.tensorflow.org/api_docs/python/tf/dtypes/DType?version=stable>

string = tf.Variable("this is a string", tf.string)

number = tf.Variable(324, tf.int16)

floating = tf.Variable(3.567, tf.float64)

### **Types of Tensors**

Before we go to far, I will mention that there are diffent types of tensors. These are the most used and we will talk more in depth about each as they are used.

* Variable
* Constant
* Placeholder
* SparseTensor

With the execption of Variable all these tensors are immuttable, meaning their value may not change during execution.

For now, it is enough to understand that we use the Variable tensor when we want to potentially change the value of our tensor.

Most of the information is taken direclty from the TensorFlow website which can be found below.

<https://www.tensorflow.org/guide/tensor>

# **TensorFlow Core Learning Algorithms**

In this notebook we will walk through 4 fundemental machine learning algorithms. We will apply each of these algorithms to unique problems and datasets before highlighting the use cases of each.

The algorithms we will focus on include:

* Linear Regression
* Classification
* Clustering
* Hidden Markov Models

It is worth noting that there are many tools within TensorFlow that could be used to solve the problems we will see below. I have chosen the tools that I belive give the most variety and are easiest to use.

## **Linear Regression**

Linear regression is one of the most basic forms of machine learning and is used to predict numeric values.

In this tutorial we will use a linear model to predict the survival rate of passangers from the titanic dataset.

This section is based on the following documentation: [*https://www.tensorflow.org/tutorials/estimator/linear*](https://www.tensorflow.org/tutorials/estimator/linear)

### **How it Works**

Before we dive in, I will provide a very surface level explination of the linear regression algorithm.

Linear regression follows a very simple concept. If data points are related linearly, we can generate a line of best fit for these points and use it to predict future values.

Let's take an example of a data set with one feature and one label.

## **Clustering**

Now that we've covered regression and classification it's time to talk about clustering data!

Clustering is a Machine Learning technique that involves the grouping of data points. In theory, data points that are in the same group should have similar properties and/or features, while data points in different groups should have highly dissimilar properties and/or features. (<https://towardsdatascience.com/the-5-clustering-algorithms-data-scientists-need-to-know-a36d136ef68>)

Unfortunalty there are issues with the current version of TensorFlow and the implementation for KMeans. This means we cannot use KMeans without writing the algorithm from scratch. We aren't quite at that level yet, so we'll just explain the basics of clustering for now.

#### **Basic Algorithm for K-Means.**

* Step 1: Randomly pick K points to place K centroids
* Step 2: Assign all the data points to the centroids by distance. The closest centroid to a point is the one it is assigned to.
* Step 3: Average all the points belonging to each centroid to find the middle of those clusters (center of mass). Place the corresponding centroids into that position.
* Step 4: Reassign every point once again to the closest centroid.
* Step 5: Repeat steps 3-4 until no point changes which centroid it belongs to.

1. **Conclusion**

Proposed method in research work aims how to improve quality of commnet analysis on textual product reviews and simple visual representation of obtained results which will be useful for nontechnical users.

Also methodology is based on training and testing will improve accuracy of results of analysis.

Work to be done in thesis has tremendous practical applications for both individual customer and service provider. Individual customer takes its benefit for decision making and service provider can take advantage to improve quality of service as well as for new product design.

Thus, we conclude:

1. Proposed new approach using open source technologies to represent textual reviews in the form of visual representation.
2. NLP based text classification is to be used to improve effectiveness of analysis.
3. Application of map-reduce environment will help us to improve speed and reliability of analysis.

Future work includes:

1. Aggregating reviews from two or more sites.
2. Feature Extraction from textual review data.
3. Consideration of Emoticons.
4. Application of data preprocessing in hadoop environment.

## **Sources**

1. Chen, James. “Line Of Best Fit.” Investopedia, Investopedia, 29 Jan. 2020, [www.investopedia.com/terms/l/line-of-best-fit.asp](http://www.investopedia.com/terms/l/line-of-best-fit.asp).
2. “Tf.feature\_column.categorical\_column\_with\_vocabulary\_list.” TensorFlow, [www.tensorflow.org/api\_docs/python/tf/feature\_column/categorical\_column\_with\_vocabulary\_list?version=stable](http://www.tensorflow.org/api_docs/python/tf/feature_column/categorical_column_with_vocabulary_list?version=stable).
3. “Build a Linear Model with Estimators  :   TensorFlow Core.” TensorFlow, [www.tensorflow.org/tutorials/estimator/linear](http://www.tensorflow.org/tutorials/estimator/linear).
4. Staff, EasyBib. “The Free Automatic Bibliography Composer.” EasyBib, Chegg, 1 Jan. 2020, [www.easybib.com/project/style/mla8?id=1582473656\_5e52a1b8c84d52.80301186](http://www.easybib.com/project/style/mla8?id=1582473656_5e52a1b8c84d52.80301186).
5. Seif, George. “The 5 Clustering Algorithms Data Scientists Need to Know.” Medium, Towards Data Science, 14 Sept. 2019, <https://towardsdatascience.com/the-5-clustering-algorithms-data-scientists-need-to-know-a36d136ef68>.
6. Definition of Hidden Markov Model, <http://jedlik.phy.bme.hu/~gerjanos/HMM/node4.html>.
7. “Tfp.distributions.HiddenMarkovModel  :   TensorFlow Probability.” TensorFlow, [www.tensorflow.org/probability/api\_docs/python/tfp/distributions/HiddenMarkovModel](http://www.tensorflow.org/probability/api_docs/python/tfp/distributions/HiddenMarkovModel).
8. [1] A. Go, R. Bhayani, and L. Huang. Twitter sentiment classification using distant supervision. In CS224N Project - Report, Stanford, 2009.
9. [2] N. Kalchbrenner, E. Grefenstette, and P. Blunsom. A convolutional neural network for modelling sentences. In ACL, 2014.
10. [3] Y. Kim. Convolutional neural networks for sentence classification. In EMNLP, 2014.
11. [4] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean. Distributed representations of words and phrases and their compositionality. In NIPS, 2013.
12. [5] S. M. Mohammad, S. Kiritchenko, and X. Zhu. Nrc-canada: Building the state-of-the-art in sentiment analysis of tweets. In Semeval, 2013.
13. [6] V. Nair and G. E. Hinton. Rectified linear units improve restricted boltzmann machines. In ICML, 2010.
14. [7] J. W. Ronan Collobert. A unified architecture for natural language processing: deep neural networks with multitask learning. In ICML, 2008.
15. [8] Y. Shen, X. He, J. Gao, L. Deng, and G. Mesnil. A latent semantic model with convolutional-pooling structure for information retrieval. CIKM, 2014.
16. [9] Y. Shen, X. He, J. Gao, L. Deng, and G. Mesnil. Learning semantic representations using convolutional neural networks for web search. In WWW, 2014.
17. [10] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. Journal of Machine Learning Research, 15:1929–1958, 2014.
18. [11] S. M. M. Xiaodan Zhu, Svetlana Kiritchenko. Nrc-canada-2014: Recent improvements in sentiment analysis of tweets, and the Voted Perceptron. In SemEval, 2014.
19. [12] M. D. Zeiler. Adadelta: An adaptive learning rate method. CoRR, 2012.