Sentiment Analysis of Twitter Data with Hybrid Learning for Recommender Applications

Conference Paper · November 2018

DOI: 10.1109/UEMCON.2018.8796661

3 authors:

Ketaki Gandhe

Aparna S. Varde
Montclair State University

Xu Du
Montclair State University

Sentiment Analysis of Twitter Data with Hybrid Learning for Recommender Applications

Ketaki Gandhe
Department of Computer Science
Montclair State University
Montclair, NJ, USA
gandhe.ketaki@gmail.com

Aparna S. Varde
Department of Computer Science
Environmental Mgmt. PhD Program
Montclair State University, NJ, USA
vardea@montclair.edu

Xu Du
Earth and Environmental Studies
Environmental Mgmt. PhD Program
Montclair State University, NJ, USA
dux3@montclair.edu

***Abstract*— This paper proposes a sentiment analysis approach to extract sentiments of tweets based on their polarity and subjectivity, classify them and visualize results graphically. This helps to understand opinions of existing users that can be helpful in future recommendations. Our proposed approach entails a hybrid learning method for classification of tweets based on a Bayesian probabilistic method for sentence level models given partially labeled training data. For implementation, we use AWS to extract data from Twitter, store extracted data in MySQL databases and code Python scripts in order to implement the analyzer. The graphical models are viewed using IPython Notebook. The results of this work would be helpful in providing recommendations to users for product reviews, political campaigns, stock predictions, urban policy decisions etc. The novelty of this research lies mainly in the hybrid learning method for sentiment analysis. We present our approach along with its implementation, evaluation and applications.**

Keywords*— *Data Analytics; Hybrid Learning; Recommenders; Opinion Mining; Social Media; Twitter; Urban Policy

I. Introduction

Sentiment Analysis refers to the automated study and investigation of evaluative text and tracking of predictive judgments therein. [1]. Due to the advent of social media, people enter their feeds on various events, products, current affairs and so on. These feeds are the actual opinions of influential people who feel free to express their views on social networking sites such as Twitter, Facebook etc. If we analyze these feeds, we are often likely to get truer, clearer opinions of people than from a guided survey [2]. Results of such analysis can be useful in various areas as follows.

Product Reviews: "What do people think about a particular product?" We can find reviews of any product/event/movie etc. from sentiment analysis. These results can be useful to the buyer as well as the seller. The buyer can pick up the best product as per their requirement. The seller can keep a closer watch on the product reviews, which can be used to improve the quality of the product.

Political Elections: "What are the public sentiments about the candidate/campaign?" The mood of the public is especially important during political campaigns. The results in this case can be extremely useful for candidates to design or alter their campaigns. Candidates can get a clear picture of the issues that really concern people. Also, the results can be used for prediction of the winner.

Search Engine Optimization (SEO): "What are people talking about as trending news?" Creating SEO friendly content is the key for any Website to get a high rank. From sentiment analysis, we can figure out the hot topics that interest people and that should be displayed on search engine home pages as headlines.

Stock Market: "What stocks should we invest in to maximize our gains? Market sentiment is the attitude of investors. Nowadays, investors are known to measure market sentiment through the use of news analytics, which include sentiment analysis on textual stories about companies and sectors. Thus, sentiment analysis can be used to find the market sentiment and hence predict the price development in stock markets.

Urban Policy: "What is the reaction of the city residents to various policies implemented by their legislators? These policies could pertain to the urban legislations are passed with respect to issues such as managing the environment, making information accessible to the public, imparting education, making healthcare more affordable and providing a good quality of life to residents on the whole. They could be related to general legislation on the whole, or to specific action pertaining to certain significant events. The opinions expressed by the public on Twitter can offer insights into their extent of satisfaction with urban policy matters. The analysis of these public sentiments can be useful for providing recommendations to urban agencies for decision making.

Given this background, we focus on our goals. We consider Twitter to get social media feeds. Tweets constitute microblogging with maximum character limits. Thus, we find that tweets being compact are very good for efficient sentiment analysis. Our goals are thus to:

- Analyze tweets and discover useful knowledge for various areas such as product reviews
- Provide inputs that could be potentially useful for recommender applications in these areas

978-1-5386-7693-6/18/$31.00 ©2018 IEEE

II. Related Work

A broad overview of the existing work in sentiment analysis is presented in [3]. The authors describe existing techniques and approaches for an opinion-oriented information retrieval. In [4], the authors use Weblogs as datasets for sentiment analysis and use emoticons assigned to blog posts as indicators of users' moods. The authors apply SVM (support vector machines) and CRF (conditional random field) learners to classify sentiments at the sentence level and then investigate several strategies to determine the sentiment.

The work in [5] uses emoticons to form a training set for sentiment classification. They collect texts containing emoticons from Usenet newsgroups. Datasets are divided into "positive" (texts with happy emoticons) and "negative" (texts with sad or angry emoticons) classes. Emoticon-trained classifiers, SVM and Naive Bayes, are able to obtain up to 70% accuracy on the test set.

The authors [6] use Twitter to collect training data and perform a sentiment search. They construct corpora by using emoticons to obtain "positive" and "negative" samples, and then use various classifiers. The best result is obtained by the Naive Bayes classifier. The authors are able to obtain up to 81% accuracy on their test set. In the work [7], the authors use the Twitter corpus to predict political elections in Germany. Their results show that Twitter is indeed used extensively for political deliberation.

In the research by [8], the authors augment accuracy of sentiment analysis by properly identifying semantic relationships between sentiment expressions and subjects. They are able to achieve precision of 75%-95% depending on inputs. The dataset they use consists of about a half million Web pages and a quarter million news articles. They do not use Twitter datasets.

In [9], the focus is on Twitter for the task of sentiment analysis. They use a method for the automatic collection of a corpus that can be applied to train a sentiment classifier. They use TreeTagger for POS-tagging and observe the difference in distributions among positive, negative and neutral sets. The classifier they use is based on the multinomial Naive Bayes that uses N-gram and POS (part-of-speech) tags as features.

Our work is somewhat orthogonal to the existing literature. We consider sentiment analysis on tweets in particular. The novelty of our research is that we propose a hybrid learning approach that works even when pre-labeled training data is not fully available. The goals are specifically to mine the opinions of users with the intention of providing good recommenders for applications.

III. Sentiment Analysis Models And Methods

We first describe the models and methods that exist in the literature on sentiment analysis in order to propose our specific approach in this paper.

A) Document Level Model

In this model, whole document is classified as positive or negative. Documents can be opinionated. A document may be a review on a Website such as Trip-advisor or another source such as a company whitepaper.

Consider the example shown in Fig. 1. In this example, the user has posted review of a hotel. This review has multiple sentences. It includes some positives ("Beautiful hotel", "great location") and some negatives ("sad service", "just painful"). The overall opinion of the review is calculated in document level classification. Due to multiple types of sentiments, classifying a document can be challenging task.

Fig. 1. Example of document level model

B) Sentence Level Model

This model classifies a single sentence as positive, negative or neutral. Since a single sentence is generally likely to have only one sentiment (positive or negative), it is easier to classify than document. Also, it is more accurate than document level classification. An example of this model appears in Fig. 2.

Fig. 2. Example of sentence level model

In sentence level classification, two sub-tasks are performed:
1. *Subjectivity classification*: Determine whether the sentence is a subjective or an objective sentence,
2. *Sentence-level sentiment classification*: If the sentence is subjective, determine whether it expresses a positive or negative opinion.

For instance, in Fig. 2 the tweet is highly subjective and has a positive sentiment.

C) Supervised Learning Method for Analysis

In sentiment analysis, supervised learning involves techniques of learning from human-annotated labeled training data (as in supervised learning elsewhere). The training datasets have examples with a *text and label* pair. Consider the example shown in Fig. 3.

```
"after years with that carrier's expensive plans and horrible customer service ,
portability seemed heaven-sent","pos"

"here's the brief synopsis : the phone is tiny , cute , feels kind of plastic-like ( as if
it might break ) , but seems pretty sturdy", "pos"

"it has lots of little cute features , my favorite being the games and the pim ( personal
information manager -- i.e. organizer ) , and the radio !","       pos"

"i spent hours setting up the stations ( accepts about 13-14 , i believe ) , though the
reception is unpredictable","neg"

"the headset that comes with the phone has good sound volume but it hurts the ears like
you cannot imagine!","neg"
```

Fig. 3. Example of training set in supervised learning for sentiment analysis

In this example, the reviews and the polarity of reviews are used as training data. These training data reviews are classified manually. When these datasets are used for training, the learning technique uses the concerned functions to map these to new unseen examples (input). The respective algorithm would then classify the unseen input correctly. We can collect the correct datasets, determine the input features, select the algorithm to be used and run the algorithm on training data. Once this is done, the accuracy of the algorithm can be evaluated using test data.

D) Unsupervised Learning Method for Analysis

This type of learning for sentiment analysis does not need human-annotated data. It uses lexical methods to classify the unlabeled data. Opinion words and phrases are the dominating indicators for sentiment classification.

Unsupervised learning in sentiment analysis is generally based on such opinion words and phrases. It performs classification based on some fixed syntactic phrases likely to be used to express opinions. (e.g., noun phrases) Unsupervised learning overcomes the limitation of supervised methods (where pre-labeled training datasets are essential).

IV. Proposed Approach: Hybrid Learning

We propose a hybrid approach combining supervised and unsupervised learning in sentiment analysis. This is because we intend to take advantage of labeled training data whenever available but also need to classify tweets that lack specific labels. The approach is described next.

A) Overview of Approach

We strongly prefer sentence level models in this proposed approach of hybrid learning for sentiment analysis. This is because are useful for microblogging sites such as Twitter, since the maximum limit of a tweet is 280 characters. Thus, a sentence level model would fit better as a sentence would typically have less than 140 characters (very few documents are that small). Sentence level models would also be more precise for sentiment classification due to likely having only one sentiment per tweet.

We propose to build a classifier for sentiment analysis deploying the classical Naive Bayes concept [10]. The Naive Bayes algorithm uses some probability theory aspects explained as follows.

$$P(C_j|D) = P(D|C_j)\, P(C_j) / P(D)$$

where, $P(Cj|D)$ = probability of instance D being in class C_j
$P(D|C_j)$ = probability of generating instance d given class C_j
$P(C_j)$ = probability of occurrence of class C_j
$P(D)$ = probability of instance D occurring

Thus, in the case of unlabeled samples in the training data, Naive Bayes can find the probability of them being either positive or negative based on similar pre-classified data. In many practical applications, parameter estimation for Naive Bayes models uses the method of maximum likelihood. Thus, it calculates all the probabilities of a feature being positive or negative using the training dataset. The probability of a sentence in test data to be positive or negative is calculated based on the formula herewith. For multiple feature data sets, Naive Bayes assumes that each feature is independent of other features in the dataset.

Thus, in our context, Naive Bayes would be interpreted with an example as follows. *Given that a person expresses an opinion in a university Website tweet, we need to know whether the person is male or female, furthermore whether he/she is a professor or a student.* This classification is performed based on learning from pre-classified datasets of tweets with the gender and occupation included, by applying the probability concepts herewith.

Based on these concepts, we build a classifier to conduct sentiment analysis, focusing on specific words in the tweets that correspond to features in the item of interest with respect to the given domain. The steps of our hybrid approach for sentiment analysis are explained next.

B) Steps of Sentiment Analyzer

We build the sentiment analyzer with the following steps:

1. Create a Twitter Developer Account: Twitter requires authentication by OAuth (Open Authorization) to use the Twitter API for any application. To collect tweets using this, the user needs to authenticate requests. We thus create a developer account to get the authentication.

2. Collect the Tweets: To collect tweets, we use Twitter API and Amazon Web Services [AWS]. For this, we create the S3 bucket and then code Python scripts to collect tweets with keywords, e.g., iPhone, Samsung Galaxy, Amazon Fire etc.

3. Store the Tweets in a Database: Once the desired data is in the S3 bucket, we download the files, convert to CSV files and import them to a MySQL Database for further computation.

4. Implement the Analyzer: We implement the sentiment analyzer using TextBlob, a Python library for processing textual data. The details of the implementation are explained in the next section.

5. Analyze the Results: We get the information about the polarity of each feature in a tweet, which is stored in json file. The Python script calculates and classifies the features with polarities from this file. Thus, as an output, we get the number of positive and negative tweets about features of a product, e.g., for a given model of the iPhone, it gives the average polarity of tweets for its battery, camera etc.

6. *Visualize the Output:* Using polarity information, we visualize the data and present it in a user-friendly manner. Graph plotting can be done using IPython Notebook, MatLab etc. GUI development can be done as needed. This extends the console-based approach to *interactive* computing in a qualitatively new direction, providing Web-based applications suitable for capturing the computation process: developing, documenting, executing code and communicating the results.

After building the sentiment analyzer using these steps, the results plotted in graphical form allow the end users to easily detect which features are good or bad in an item. This can be helpful for making decisions.

V. Implementation of Sentiment Analysis Approach

We implement the sentiment analyzer using TextBlob. This is a Python library for text data processing that provides a consistent application programming interface (API) for diving into common NLP (natural language processing) tasks such as POS (part-of-speech) tagging, noun phrase extraction and further analysis. TextBlob stands on the giant shoulders of NLTK and Pattern. NLTK is the Natural Language Tool Kit for Python that helps to build Python programs to work with human language data. Pattern is Python's Web mining module with tools for machine learning, data mining, network analysis and more.

In this implementation, we use the TextBlob classifier module to classify the tweets as positive or negative. Tweets are stored in MySQL database. MySQL DB module of Python is used to communicate with database. Using this, MySQL connection is established, tweets are fetched from table and each tweet is processed as follows.

First, the tweet is cleaned. For example, consider a tweet from 2014: "Yes it's true, the revolutionary iPhone6 is up for launch, finally! \\Have you Pre-registered? \#iPhone6india @MehekMahtani". We need to remove hashtags, usernames etc. If there is any URL, we should remove that as well. Also, extra spaces, multiple characters should be removed. We use following two functions as shown in Fig. 4 for tweet cleaning.

```
def replaceDuplicates(s):
    #replace repitions
    pattern = re.compile(r"(.)\1{1,}", re.DOTALL)
    return pattern.sub(r"\1\1", s)

def processTweet(tweet):
    # clean the tweets
    #Convert to lower case
    tweet = tweet.lower()
    #Remove www.* or https?://*
    tweet = re.sub('((www\.[\s]+)|(https?://[^\s]+))','',tweet)
    #Remove @username
    tweet = re.sub('@[^\s]+','',tweet)
    #Remove additional white spaces
    tweet = re.sub('[\s]+', ' ', tweet)
    #Replace hashtags with word
    tweet = re.sub(r'#([^\s]+)', r'\1', tweet)
    #trim
    tweet = tweet.strip('\'"')
    return tweet
#end
```

Fig. 4. Functions to clean the tweets

First, we convert each tweet into lowercase. Then we check if it contains any URL by searching for *www* and *https* in the tweet. If found, we just remove the URL. Similarly if we find @username, we remove it as we do not need the username to classify polarity. We also replace hashtag with a normal word that describes the hashtag. By processing each tweet in this manner, we minimize clutter and provide clean tweets to Textblob for better accuracy. Pattern, the Web mining module of Python is used to find repetitions of particular characters in a tweet. We replace the repeated characters for better accuracy. After cleaning the tweet, we pass the tweet to Textblob for classification.

Once the tweet is classified, we extract features from the tweet (e.g., Battery Life, Size, Looks etc.). For this, we make use of Sentiword [11] which is a huge lexical source of words and their respective sentiments. We extract Nouns and Verbs from the tweet and check the feature list to find out whether any of the features are present in the tweet. An example of a feature list appears in Fig. 5.

```
pFeatures=["iphone6","iphone
6","phone","battery","size","bluetooth","software","price","usage","use","keypad","look","
looking","camera","display","color","handsfree","upgrade","screen","feel","feels","shape",
"audio","construction","apps","memory","video","games","security","service","rebate","
internet","quality","OS"]
```

Fig. 5. Feature List Example

If a token matches, we store the features with polarity, subjectivity and sentiwords (if any) of tweet in json file. Thus, we get the class (positive / negative) and the feature of the tweet. For this we download the positive and negative words and import them in the database. We find that adjectives and adverbs in the tweet (ADJ and ADV) show the actual sentiment / opinion about the feature (Noun / Verb) in the tweet. To store sentiwords, we classify them in two categories. The sentiment is 1 if the word is positive and it is 0 if the word is negative. Fig. 6 shows a partial snapshot of a sentiword table in a MySQL database.

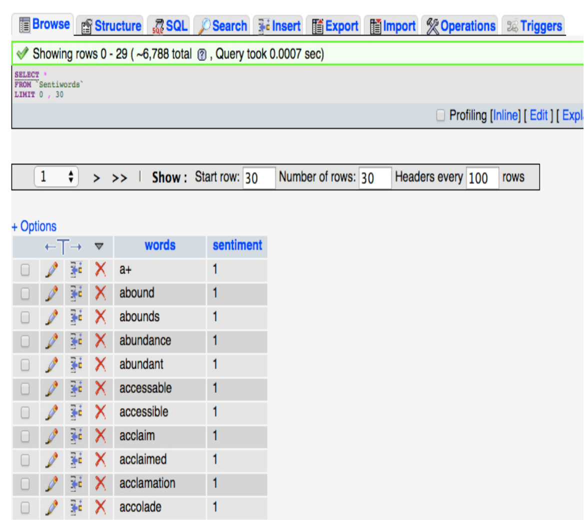

Fig. 6. Partial snapshot of a sentiword table

After conducting this implementation, we get an output file which stores the polarities of the tweets. Fig. 7 shows an example of a *json* file used to store the output.


```
{
    "polarity": 0.175,
    "sentiwords": [
        "revolutionary"
    ],
    "features": [
        "iphone6"
    ],
    "subjectivity": 0.825
},
{
    "polarity": 0.5,
    "sentiwords": [
        "unimaginable"
    ],
    "features": [
        "iphone6"
    ],
    "subjectivity": 0.5
},
```


Fig. 7. Example of output file

This output is with reference to the iPhone 6 product example from 2014. A Polarity of 1.0 means that the statement is 100% positive while negative polarity is denoted by -1.0. Also, the statement is either subjective or objective. If the tweet is objective, it is denoted by 0.0 whereas the extent of subjectivity is expressed by the term *"subjectivity"* in output file. Thus, higher the number, higher is the subjectivity of the tweet. The output can be used for visualizing the results. Visualization can be done by graph plotting and helps to make the output more appealing to the end users at-a-glance. An example of graph plotting is shown in the experimental results as described next.

VI. Experimental Evaluation

We conduct the performance evaluation of our sentiment analyzer with real data from Twitter. A summary of our evaluation is presented herewith.

An important category of our experiments involves collecting tweets in the smartphone domain. This is with the goal of providing recommendations to buyers and sellers of various smartphones and helping product launches. We collect real data from tweets and process it as follows.

A) Data on iPhone 6

The datasets we download here consist of around 7000 tweets (2000-iPhone6, 2000-iPhone6 plus, 1000-Samsung Galaxy, 1000-Amazon Fire, 1000-HTC) from the year 2014. These tweets are analyzed and used to find the sentiments about the products. We plot the results of sentiment analysis in graphical form, so as to enable the end users to easily analyze which features are good or bad in the products.

An example of such a graphical plot appears in Fig. 8. This figure shows that according to tweets, sentiments are positive for iPhone6 model in general (0.2) but are negative specifically for its Camera (-0.8) and its Battery (-0.4). Thus, we can infer that influential users entering these tweets are more interested in the overall outlook of the iPhone 6 per se than in some of its individual features. Such information can be used in product recommendations.

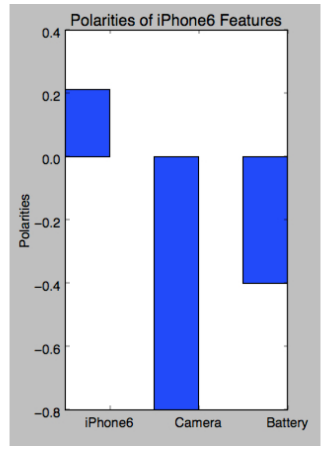

Fig. 8. Graph plotted from sentiment analysis for iPhone 6 product review

In retrospect, we find that these recommendations have actually been useful. The iPhone6 in 2014 did indeed get very well-received by customers overall, however its battery seemed to pose some problems and its camera was often rated relatively low as compared to that of other products (such as Samsung Galaxy). Thus, the experimental evaluation presented herewith based on sentiment analysis conducted corroborates the real reception of the product. This confirms the validity of the sentiment analysis.

B) Data on Peatland Fires

Peatlands have much organic matter caused by decomposition of plant residue. Indonesia has the maximum peatlands in South East Asia. Pollutants from these fires affect neighboring areas, e.g., Singapore due to which Indonesian Peatland Fires (IPFs) are considered international hazards in Environmental Management. Airborne particulates pollution is a major concern. Research shows that rhinitis, asthma, and respiratory infections increase if particulate concentration is of hazardous level [12]. Thus, regulatory policies have been passed by Singaporean urban agencies to counterbalance the hazardous impact of IPFs. Singapore has an air quality system PSI (Pollutant Standards Index) with 6 pollutants: sulphur dioxide (SO_2), particulate matter (PM10), fine particulate matter (PM2.5), nitrogen dioxide (NO_2), carbon monoxide (CO) and ozone (O_3). Their environmental agency publishes PSI levels hourly through websites such as haze.gov.sg. People get this PSI information through and tweet their reaction on daily PSI level and air quality.

We use this Twitter data in the experiments shown here and analyze it based on the approach described in this paper. This gauges the sentiments of the public expressing their opinion on the policies taken by their agencies to deal with this event, namely Peatland fires. The results of the sentiment analysis are summarized in Fig. 9. The data shown in this chart is based on two different user groups in the same region but different time periods, separated by six months.

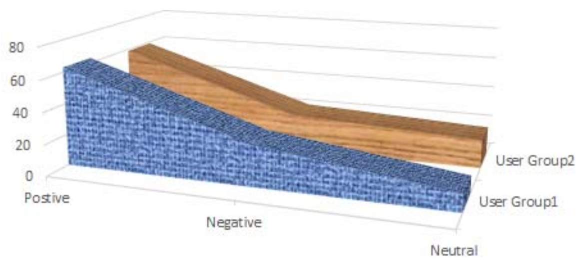

Fig 9. Area Chart for Sentiment Analysis of IPF Impact

The chart shows that policies to counterbalance the effect of IPF-based pollution appear to be fairly good since around 60% of users express positive sentiments. Yet, there is potential for improvement as around 25% of users are neutral and 15% are negative in their sentiments. This opinion mining thus provides useful inputs to government bodies in the respective region and also to its prospective residents.

C) Data on NYC Ordinances

We investigate data on ordinances or local laws in the NYC metropolitan area [13]. In the experiments shown here, we collect around 5000 tweets posted by the public in NYC pertaining to ordinances on various general policies pertaining to the economy, transportation etc. These tweets are subject to sentiment analysis using the approach explained herewith. The results are summarized in the pie chart in Fig, 10.

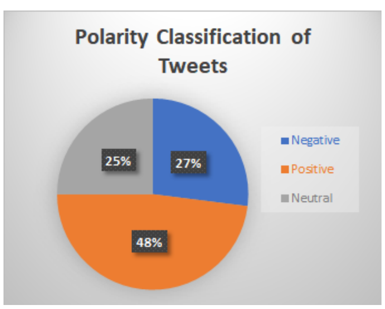

Fig. 10. Pie Chart on Public Reactions to NYC Ordinances

It is clear from this pie chart that the percentage of positive tweets is the highest of all, thus we can infer that residents of NYC seem to approve of their urban policies on the whole. However, note that positive sentiments are conveyed by less than half the total number of residents here, which means that there is scope for further enhancement with respect to urban policy. Such charts can serve as recommendations to the urban management agencies. Further details can be provided on the specific aspects in which there is need for improvement, e.g., similar charts expressing sentiments on policies in the economic sector alone or transportation sector alone etc. Thus, if the greatest percentage of negative tweets come from a particular sector, it can be inferred that there is need for better policies in that sector. Conversely, if the public seems highly satisfied with policies on a given aspect, e.g., environment, that serves as positive feedback in recommendations.

Likewise, several experiments have been conducted. The results of these experiments have actually been found useful in providing recommendations to prospective users.

VII. Recommender Applications

Based on sentiment analysis conducted with real data, we now describe its targeted applications in recommenders.

A) Product Reviews

We consider reviews from the angles of sellers, buyers and product launches.

Seller: Our sentiment analysis approach can predict how the product is doing in the market. For example, by studying the graphical plot shown in Fig. 8, a seller can determine that users are interested in the iPhone6 overall. Likewise, with other such plots, Apple can get an idea of how well a given iPhone model is received by public, what features should be incorporated in next model etc.

Buyer: From a buying angle, our approach can help in comparing features from various sellers to decide which product or service best suits their needs, e.g., if features pertaining to hotel reviews from Websites like tripadvisor.com are visually depicted (analogous to Fig. 8), users can select a hotel based on previous reviews.

Product Launch: Our approach can make existing users affect new product launches. For instance, controversies related to movies can lead to a good start. Referring to Fig. 8, if there were negative tweets about a movie (as for cameras and batteries here), its launch would likely succeed (as the iPhone 6 launch did). This is because existing viewers would convince new ones to watch the movie to find controversies.

B) Political Elections

We focus on outcome prediction and campaigning processes in political elections where recommendations matter.

Outcome Prediction: The sentiments expressed by influential users on social media can be used to predict victory in elections. For example, if a candidate is as well-received as an iPhone 6 (in Fig. 8), he or she is quite likely to win. This is because people freely express their views about candidates / parties on social media, so if their sentiments are positive, that reflects well about candidates.

Campaigning Processes: Influential users create awareness among people about positive and negative changes. For instance, candidates not leading based on tweets, can outline strategies referring to positive sentiments expressed about their opponents, and build campaigns accordingly.

C) Search Engine Optimization

In applications for SEO, i.e., Search Engine Optimization, we consider market trends and blogging.

Market Trends: Trends in the market pertaining to hot topics can be captured using our sentiment analysis approach. For example, with reference to our experiments, if we find that *iPhone* occurs more frequently than *HTC,* we can conclude that people are more excited about the iPhone than the HTC smartphone. Adding these keywords in Website contents will lead to more hits which in turn will help in increasing the page rank, through SEO.

Blogs: Several blogs become popular if the content is interesting. Our sentiment analysis approach can help find topics in which people are interested. Thus, bloggers can get information about people's choices. This information can be used to create new blogs and improve existing ones.

D) Stock Market

In the stock market area, we focus on two aspects, namely, bulls & bears, and price changes. We see how recommenders impact these applications.

Bulls & Bears: When a high proportion of investors express a bearish (negative) sentiment, some analysts consider it to be a strong signal that a market bottom may be near. Likewise, if sentiment is bullish (positive), analysts consider that market will go up. Our sentiment analysis experiments taking into account such polarities with graphical plots (see Fig. 8) can thus be helpful in estimating bulls & bears in the stock market.

Price Changes: Investors and stockholders measure sentiments by analyzing and mining textual stories about companies and sectors. Positive sentiments could lead to increase in stock prices whereas negative sentiments about the company could lead to decrease in prices. Our sentiment analysis approach could thus cause influential users to have an impact on stock prices.

E) Urban Policy

In this work, we consider urban policy issues related to specific events and general legislation.

Specific Events: Policy makers often take certain measures to act upon significant events that have occurred, e.g., flood, famine, fire etc. The sentiments expressed by users on social media sites such as Twitter enable us to gauge the reaction of the public on the satisfaction with these policies, with respect to how much they cater to addressing the respective issues pertaining to the corresponding events. Sentiment analysis on this can provide recommendations to policy makers on these specific aspects, so they can enhance current policies as needed and plan for future occurrences accordingly.

General Legislation: Urban regions often have local laws that affect the general lifestyle of the public on a daily basis. These could pertain to transportation systems, healthcare issues, use of mobile devices, education facilities and so on. Residents often express their opinion on such policy matters through social media to make their voice heard. Sentiment analysis of such data can enable legislators to get a good idea of the impact their legislation makes on the common public. This can be useful for recommendations to enable better decision making through public involvement.

VIII. Conclusions

Microblogging has emerged as a major type of communication today. Large amounts of data in microblogging sites provide useful inputs for sentiment analysis. In our work on sentiment analysis in this paper, we make the following contributions:

- Present a method for opinion mining that would be useful in various recommender applications
- Propose a hybrid learning approach with Naive Bayes for sentiment analysis, using probabilistic estimates where exact labels are not available
- Conduct evaluation on real data relevant to product reviews and urban policy
- Visualize the results of sentiment analysis for easy depiction to end users to facilitate recommendations

Ongoing work includes conducting more experiments with other domains. Some of our ongoing research also entails incorporating commonsense knowledge in sentiment analysis to simulate human judgment in opinion mining [14, 15]. As future work, we could consider enhancing our approach further to include a combination of classifiers in an ensemble.

References

[1] Das, S. and Chen, M. 2001. Yahoo! for Amazon: Extracting market sentiment from stock message boards. *Asia Pacific Finance Association Annual Conference (APFA)*, pp. 79-86.

[2] Whitelaw, C., Garg, N., Argamon, S. 2005. Using appraisal groups for sentiment analysis. *ACM CIKM*, pp. 625-631.

[3] Pang, B. and Lee, L. 2004. A sentimental education: sentiment analysis using subjectivity summarization based on minimum cuts. *ACL)*, Article 271.

[4] Yang, C., Lin, K. H., and Chen, H. 2007. Emotion classification using web blog corpora. *IEEE/WIC/ACM Intl. Conf. Web Intelligence (WI)*, pp. 275-278.

[5] Read, J. 2005. Using emoticons to reduce dependency in machine learning techniques for sentiment classification. *ACL Student Workshop*, pp. 43-48.

[6] Go, A., Bhayani, R., and Huang, L. 2009. Twitter sentiment classification using distant supervision. Technical report, Stanford Digital Library Technologies Project.

[7] Tumasjan, A., Sprenger, T.O., Sandner, P.G., Welpe, I.M. 2010. Predicting Elections with Twitter: What 140 Characters Reveal about Political Sentiment. *AAAI Conference on Weblogs and Social Media,* pp.*178-185*.

[8] Nasukawa, T., and Yi, J. 2003. Sentiment Analysis: Capturing Favorability Using Natural Language Processing. *K-CAP*, pp. 70-77.

[9] Pak, A., and Paroubek, P. 2010. Twitter as a Corpus for Sentiment Analysis and Opinion Mining. *LREC*, pp. 19-21.

[10] Russell, S. and Norvig, P. 2003. *Artificial Intelligence: A Modern Approach*. 2nd Edition. Prentice Hall.

[11] Baccianella, S., Esuli, A. and Sebastiani, F. 2010. Sentiwordnet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining. *LREC.*

[12] Zhou, J., Chen, A., Cao, Q., Yang, B., Victor, W., Chang, C. and Nazaroff, W. 2015. Particle Exposure during the 2013 Haze in Singapore: importance of the built environment. *J. of Building & Environment*, pp. 14-23.

[13] NYC Council.Legislature http://legistar.council.nyc.gov/, 2018.

[14] Tandon, N., Varde, A. and de Melo, G. 2017. Commonsense Knowledge in Machine Intelligence. ACM SIGMOD Record, 46(4): 49-52.

[15] Puri, M., Du, X., Varde, A. and de Melo, G. 2018. Mapping Ordinances and Tweets using Smart City Characteristics to Aid Opinion Mining. *The Web Conference WWW Companion Volume,* pp. 1721-1728.