The Report Committee for Adam Lane Whipple
Certifies that this is the approved version of the following report:

Comparison of Algorithms for Twitter Sentiment Analysis

APPROVED BY
SUPERVISING COMMITTEE:

Supervisor:

Constantine Caramanis

Kathleen Suzanne Barber

Comparison of Algorithms for Twitter Sentiment Analysis

by

Adam Lane Whipple, B.A., M.B.A.

Report

Presented to the Faculty of the Graduate School of

The University of Texas at Austin

in Partial Fulfillment

of the Requirements

for the Degree of

Master of Science in Engineering

The University of Texas at Austin

May 2017

Dedication

I dedicate this work to my family whose sacrifice and support afforded me this tremendous opportunity.

Abstract

Comparison of Algorithms in Twitter Sentiment Analysis

Adam Lane Whipple, M.S.E.

The University of Texas at Austin, 2017

Supervisor: Constantine Caramanis

Sentiment Analysis has gained attention in recent years owing to the massive increase in personal statements made at the individual level, spread across vast geographic and demographic ranges. That data has become vastly more accessible as micro-blog sites such as Twitter and Facebook have released public, free interfaces. This research seeks to understand the processes behind Sentiment Analysis and to compare statistical methodologies for classifying Twitter sentiments.

Table of Contents

List of Tables

Introduction to Theoretical Framework

The growth of social media in recent years has opened the flood gates of individual sentiment on a massive scale, which has contributed to the "Big Data Revolution" (Ceron, Curini, & Iacus, 2016). Applications in sentiment analysis have been employed in areas such as "politics, law making, sociology, and psychology" not to mention applications in business intelligence (Devi, 2016; Vercellis, 2009). Users employ micro-blogs such as Twitter and Facebook, among others, to express sentiments, opinions, complaints, and general comments on any number of topics including products and services offered by companies (Agarwal, Xie, Vovsha, Rambow, & Passonneau, 2011). The challenge for researchers is to comb through the multitudes of data and sort ample noise from valuable information (Ceron et al., 2016).

Statement of Problem

Over the past few years, researchers have proposed a variety of methods to analyze and classify micro-blog data, including: Naïve-Bayes, Support Vector Machines, Fuzzy Clustering, K-Means Clustering, Neural Networks, and many others (R. Jaya, S. Kumar, 2016). Often either the research presents the performance of one method but fails to compare that method to any other, or the methods of analyzing performance are varied between research projects making it difficult to determine which methods are best under given circumstances.

This report seeks to form a baseline method of comparison and then to utilize that baseline to compare methods in sentiment analysis. For example, accuracy is tremendously important, but what about performance? Considering the number of distinct individual micro-blogs can number in the hundreds of thousands or even millions, performance could be a factor and balancing the tradeoff of accuracy and performance are important considerations.

Purpose of study

The purpose of this study is to establish a baseline for comparison of methods in Sentiment Analysis, then, to compare a large set of tweets using that baseline. The comparison will measure accuracy and performance of each method. This is important to help answer the question, "which method is best under which circumstance?" The unit of analysis will be statistic methods: Naïve Bayes and Support Vector Machines, and, time permitting, Stochastic Gradient Descent Classification, etc.

Literature Review

To better understand sentiment analysis of Twitter data, it is important to understand the source, Twitter, as well as the steps involved in text based sentiment analysis including: acquisition, preprocessing, and transformation. The following discussion expands on those topics to illuminate the importance of each step in sentiment analysis.

Numerous definitions exist to describe the act of utilizing formal methods in acquisition, processing, and reporting on data from social media such as Twitter. A few definitions stood out. Upshall defined text mining as "an umbrella term covering a wide range of software tools, including natural language analysis, use of statistical techniques, and machine learning, designed to extract entities (names of people and places), index terms, and relationships" (Upshall, 2014). Zeng, et.al, defined social media analytics as, "concerned with developing and evaluating informatics tools and frameworks to collect, monitor, analyze, summarize, and visualize social media data, usually driven by specific requirements from a target application" (Zeng, Chen, Lusch, & Li, 2010).

TWITTER

Twitter is one of many social media platforms useful in social media analytics. He, et al., call attention to some of the unique challenges in analyzing social media as compared to other transaction data:

> First, social media cover general users' opinions about almost every aspect of our life. Second, there are always fresh content on social media and the content are updated consistently and timely by numerous online users. Third, social media contents are associated with metadata in various attributes such as user, location, likes, time, dislikes, etc. Fourth, social media data have quality issues and contain a lot of noise and spams, which need to be sifted through to figure out what data can be trusted. Real benefit can be obtained by analyzing massive social media

> data in real time and gaining trustworthy insights while social media data are continuously coming in high speed. (He et al., 2015)

As He, et al.(He et al., 2015), point out, analytics in social media present some unique challenges. Devi, et al., expand on other challenges:

- Context: a word can be interpreted positively or negatively based on context,
- Semantic ambiguity: presences of positive or negative words do not necessarily decide the polarity of the entire text,
- Sarcasm: use of irony or mockery to convey contempt,
- Comparatives: comparisons of two subjects can be interpreted differently depending on the viewpoint of the analysis. (Devi, 2016)

Sentiment analysis of twitter data poses some major challenges with the abundance of noise, grammatical errors, use of acronyms and emoticons combined with the challenges noted above. One other notable fact about Twitter is volume. According to one source, an average of 6,000 tweets per second are posted to Twitter, which amounts to about 200 billion tweets per year ("Twitter Usage Statistics - Internet Live Stats," 2016).

PHASES OF SENTIMENT ANALYSIS

Phases common in twitter sentiment analysis include data acquisition, pre-processing, and transformation (Wahyudi & Putri, 2016; Wikarsa & Thahir, 2016).

Data acquisition

Data acquisition involves defining the domain datasets (Wahyudi & Putri, 2016). While most studies define the dataset per some set of criteria, a movie, a public figure, a

company, etc., others utilize the entire corpus of tweets (Biever, 2010; Garikar, Marakarkandy, & Dasgupta, 2015; Sato, Huang, & Yen, 2015; Wahyudi & Putri, 2016).

Twitter provides an application program interface (API) useful in accessing tweets based on a search or stream. In addition, twitter publishes references to numerous libraries in various programming languages for the twitter API ("Twitter Developer Documentation," n.d.).

Pre-processing

The raw data from the twitter API arrives in JSON format and contains the tweet text along with metadata about the tweet such as author, location, language, and much more. A complete list of the fields contained in the JSON object can be located on the developer documentation site ("Twitter Developer Documentation," n.d.). While some applications may utilize some or all metadata, others concentrate strictly on the contents of the text itself. The text of a tweet may contain a variety of information that needs to be removed or altered to be processed through data transformation. The process of altering the text data is called pre-processing (Agarwal et al., 2011; Arian, Hosniyeh S. Speily, 2016; Wahyudi & Putri, 2016; Wikarsa & Thahir, 2016).

The steps involved in pre-processing vary among studies, however common patterns emerge (Agarwal et al., 2011; Arian, Hosniyeh S. Speily, 2016; Oussalah, Escallier, & Daher, 2015; Wahyudi & Putri, 2016; Wikarsa & Thahir, 2016). Typically, pre-processing involves removal of stop words, removal of punctuation, removal or translation of acronyms, removal of URL, removal or translation of emoticons / emoji, conversion to lower-case letters, language filter (if not filtered during data acquisition), and in some cases, tokenization and or stemming.

Transformation

Data transformation is the replacement of variables by a function of that variable (Nicholas J. Cox, 2005). The exact function employed in data transformation varies by study. The following discussion describes some of algorithms utilized in the literature.

Naïve Bayes

Naïve Bayes (NB) is considered one of the fastest and simplest methods useful in sentiment analysis (Wikarsa & Thahir, 2016). NB assumes all features are mutually independent (Patil, Rupali; Bhavsar, R.P; Pawar, 2016).

$$P(c|x) = \frac{P(x|C)P(c)}{P(x)}$$

Support Vector Machine

"Support Vector Machine is an algorithm that works using a nonlinear mapping to transform the original training data to a higher dimension. In this new dimension, will seek to separate hyperplane linearly" (Wahyudi & Putri, 2016). Essentially, SVM finds the optimal separation between support vectors, data points, nearest to the frontier. If a hyperplane cannot be discerned from given data, the data is transformed to a higher dimension using a kernel method.

$$\left[\frac{1}{n}\sum_{i=1}^{n}\max(0,1-y_i(w*\ x_i+b))\right]+\lambda||w||^2$$

Stochastic Gradient Descent Classification

Stochastic Gradient Descent (SGD) Classification is an iterative learning method considered advantageous when the training set size is large (Tripathy, Agrawal, & Rath, 2016). Essentially, SGD finds coefficients of a function while minimizing cost.

$$w_{t+1} \coloneqq w_t - \eta \sum_{i=1}^{n} \nabla Q_i(w)/n$$

Feature Selection

Utilizing Support Vector Machines and Naïve Bayes algorithms are not uncommon in sentiment analysis but methods in feature selection vary. In general, feature selection can be broken into two categories: filters and wrappers. Filters are independent of the learning algorithm where the user employs some method of determining which features to use and which to discard. Wrappers are "modifications … which choose important features as well as conduct testing / training" (Chen & Lin, 2006). Another definition of wrapper adds clarity, "the learning algorithm is wrapped into the attribute selection procedure, so that based on different subset of attributes multiple classifiers can be generated and select the subset which gives the best performance. Because of the complete size of space of attribute subsets the wrapper approach become[s] cost prohibitive so text classification are often forced to settle for the filter approach (Patil, Rupali; Bhavsar, R.P; Pawar, 2016).

Determining Validity

The literature review yielded a variety of determinations for the validation of any given algorithm. Validity or accuracy can be measured using 10-fold cross validation (Wahyudi & Putri, 2016; Wikarsa & Thahir, 2016). Confusion matrix and ROC curves measure AUC (Wahyudi & Putri, 2016) , F1-score (Yang, Geng, & Liao, 2016), are also present in the literature. Ceron, et.al, use Mean Absolute Error (MAE) and Chi Squared (Ceron et al., 2016).

The above determinants may not work for clustering algorithms. Halkidi, et.al (Halkidi, Batistakis, & Vazirgiannis, 2001), describe three approaches of validation

techniques for clustering: external criteria, internal criteria, and relative criteria. External criteria "reflects our intuition about the clustering structure of the data set" based on pre-defined criteria. Internal criteria refer to "quantities that involve the vectors of the data set themselves," in other words, evaluating how the clusters are grouped internally and relative to other clusters in the data set. Relative criteria evaluates clustering by comparing it to other clustering schemes.

Determining Performance

Performance considers computational efficiency in terms of time. Given that algorithms are platform agnostic, measuring performance in one implementation could be considered arbitrary. Still, to compare algorithms in sentiment analysis, there is some value in determining the performance efficiency of one algorithm over another so that a user might select based on the needs of the endeavor. For example, there may be a situation where accuracy could be sacrificed in lieu of faster performance or vice versa. In the literature review if performance is considered at all, run time is the only metric measured (Ceron et al., 2016).

Methodology - Design / Methods and Procedures

This research will endeavor to compare Naïve Bayes, Support Vector Machines and Stochastic Gradient Descent for accuracy and performance in Sentiment Analysis of Twitter data. If possible, additional algorithms will be included: Random Forest, etc.

TRAINING DATA

All algorithms in this comparison involve supervised learning, which requires a training set complete with pre-defined polarity. To achieve this, the research will utilize a large dataset consisting of nearly 1.6M tweets, acquired from Sentiment140 (Alec Go, n.d.). This research acknowledges that this data set may present certain problems. First, the data may or may not have been pre-processed and what pre-processing steps employed is a mystery. Second, the polarity assignments could be incorrect leading to poor training. Finally, this set only captures positive and negative sentiments but fails to gauge neutrality.

APPLICATION

Part of this research is to create an application capable of loading tweet data, pre-processing it, and utilizing statistical methods to transform the data. Once trained using each method, the application will compare accuracy and performance

Data Acquisition

The data can be sourced either from Twitter directly or through other sources where the data has already been pre-processed and polarity assigned for use as training data. Once the application has trained on a dataset, the application shall set-up simple queries to pull data: “apple”, “Pizza Hut”, “Trump”, etc. and gauge their sentiment.

Pre-Processing

As noted above, pre-processing can involve a variety of steps. This research will accomplish the following preprocessing steps:

- Emoticon / Emoji - Emoticons and emoji can either be removed altogether or converted using some defined lexicon. Initially, the application will simply remove them. Later, it may prove advantageous to utilize a lexicon of pre-defined polarity assignments for each of the 741 Unicode emoji acquired from (Novak, Petra K.; Smailovic, Jasmina; Sluban, Borut; Mozetic, 2015)
- Remove stop words
- Remove any URL
- Remove any direct address
- Convert to lower case
- Tokenization
- Stemming / Lemmatization
- Remove duplicates

Transform the processed data using selected methods

The application shall transform the same data through multiple algorithms for comparison. Scikit Learn libraries will be employed for all transformations. Performance will be determined after pre-processing using metrics built into the Scikit Learn libraries (Pedregosa et al., 2012).

Results

Beyond preprocessing, two feature extractors were employed: TFidfVectorizer and HashingVectorizer. Once the vectorizer was initiated, SelectKBest and chi-squared were used for feature selection. To transform the data, Multinomial Naive Bayes was used. For Support Vector Machine, LinearSVC was used. For basis of comparison, Stochastic Gradient Descent Linear Classifier was also used. Individual results are as follows:

Table 1: Training time of TFidfVectorizer

Set	Set Size	Training Time (seconds)
Train	947.176	46.917
Test	633,274	32.079

Table 2: Training time of HashingVectorizer

Set	Set Size	Training Time (seconds)
Train	947.176	40.393
Test	633,274	27.014

Table 3: Results of Statistical Methodologies

Method	**Vectorizer**	**Train Time (seconds)**	**Test Time (seconds)**	**Accuracy**
Multinomial Naïve Bayes	TFidfVectorizer	0.225	0.054	76.5%
Multinomial Naïve Bayes	HashingVectorizer	0.195	0.047	76.6%
LinearSVC	TFidfVectorizer	14.828	0.029	77.5%
LinearSVC	HashingVectorizer	7.303	0.019	77.9%
SGDClassifier	TFidfVectorizer	1.490	0.028	77.6%
SGDClassifier	HashingVectorizer	12.072	0.018	77.3%

Table 4: Classification Report: Multinomial Naïve Bayes with TFidfVectorizer

	precision	recall	f1-score	support
0	0.77	0.76	0.76	315665
1	0.76	0.77	0.77	315780
Avg. / total	0.77	0.77	0.77	631445

Table 5: Classification Report: Multinomial Naïve Bayes with HashingVectorizer

	precision	recall	f1-score	support
0	0.77	0.76	0.76	315665
1	0.76	0.78	0.77	315780
Avg. / total	0.77	0.77	0.77	631445

Table 6: Classification Report: SGDClassifier with TFidfVectorizer

	precision	recall	f1-score	support
0	0.79	0.75	0.77	315665
1	0.76	0.79	0.78	315780
Avg. / total	0.78	0.77	0.77	631445

Table 7: Classification Report: SGDClassifier with HashingVectorizer

	precision	recall	f1-score	support
0	0.80	0.75	0.77	315665
1	0.76	0.81	0.79	315780
Avg. / total	0.78	0.78	0.78	631445

Table 8: Classification Report: LinearSVC with TFidfVectorizer

	precision	recall	f1-score	support
0	0.79	0.75	0.77	315665
1	0.76	0.80	0.78	315780
Avg. / total	0.78	0.78	0.78	631445

Table 9: Classification Report: LinearSVC with HashingVectorizer

	precision	recall	f1-score	support
0	0.79	0.75	0.77	315665
1	0.76	0.80	0.80	315780
Avg. / total	0.77	0.77	0.77	631445

Prediction accuracy with Support Vector Machine Classification using HashVectorizer for feature vectorization yielded the highest accuracy of all individual methods attempted, though at a cost of higher training time. However, other methods could accurately predict items that LinearSVC classifier was not, which suggests that a weighted combination of predictive models might yield a higher accuracy than any one method alone. If time is a limiting factor, Naïve Bayes methods yielded accuracy near that of Support Vector Machines in a fraction of the time.

Conclusion

Each step of the Sentiment Analysis problem presented certain challenges. Data acquisition from Twitter was challenging due to limitations on use of the Twitter API. Preprocessing Unicode strings using Python 2.7 presented several challenges this version uses string default of Ascii while Twitter delivers test strings un Unicode. Often, processing required decoding or encoding text strings to handle certain circumstances. Additionally, preprocessing nearly 1.6M lines of text presented numerous challenges in efficiency. Training the data was challenging due to the sheer volume of features, feature selection became tremendously important as did additional steps in preprocessing to limit the number of erroneous features.

Other errors in prediction could easily be attributed to misclassification of training data as classification of any sentiment is a subjective endeavor. This suggests that classification of training data should be conducted by the individual or enterprise rather than relying on pre-classified data produced by individuals who may not share the same interpretation of sentiment.

Future Research

Ideas for future research include exploring comparisons of methods of wrapping Support Vector Machine rather than filtering for feature selection. Additionally, it would be interesting to compare methodologies for combining predictive models to yield greater accuracy. Furthermore, other methodologies designed to yield greater accuracy include boosting, which may yield important clues in achieving better results.

References

Agarwal, A., Xie, B., Vovsha, I., Rambow, O., & Passonneau, R. (2011). Sentiment analysis of Twitter data. *Association for Computational Linguistics*, 30–38. Retrieved from http://dl.acm.org/citation.cfm?id=2021109.2021114%5Cnpapers3://publication/uuid/83CA53FE-43D1-4BD5-BCF2-D55B82CF0F99

Alec Go. (n.d.). Sentiment140 - A Twitter Sentiment Analysis Tool. Retrieved April 23, 2017, from http://help.sentiment140.com/for-students

Arian, Hosniyeh S. Speily, O. R. B. (2016). A Method for Mining Social Media to Discovering Influential Users. *International Journal of Computer Science and Information Security*, *14*(4), 353–366.

Biever, C. (2010). Twitter mood maps reveal emotional states of America. *New Scientist*, *207*(2771), 14. https://doi.org/10.1016/S0262-4079(10)61833-7

Ceron, A., Curini, L., & Iacus, S. M. (2016). ISA: A fast, scalable and accurate algorithm for sentiment analysis of social media content. *Information Sciences*, *367–368*, 105–124. https://doi.org/10.1016/j.ins.2016.05.052

Chen, Y., & Lin, C. (2006). Combining SVMs with Various Feature Selection Strategies. *Feature Extraction*, *324*(1), 315–324. https://doi.org/10.1007/978-3-540-35488-8_13

Devi, D. V. N. (2016). Sentiment Analysis Using Harn Algorithm.

Garikar, D. D., Marakarkandy, B., & Dasgupta, C. (2015). Using Twitter data to predict the performance of Bollywood movies. *Industrial Management & Data Systems*,

115(9), 1604–1621. https://doi.org/10.1108/imds-04-2015-0145

Halkidi, M., Batistakis, Y., & Vazirgiannis, M. (2001). On clustering validation techniques. *Journal of Intelligent Information Systems*, *17*(2–3), 107–145. https://doi.org/10.1023/A:1012801612483

He, W., Shen, J., Tian, X., Li, Y., Akula, V., Yan, G., & Tao, R. (2015). Gaining competitive intelligence from social media data: Evidence from two largest retail chains in the world. *Industrial Management & Data Systems*, *115*(9), 1622–1636. https://doi.org/10.1108/02635570710734262

Nicholas J. Cox. (2005). Transformations: an introduction. Retrieved February 9, 2017, from http://fmwww.bc.edu/repec/bocode/t/transint.html

Novak, Petra K.; Smailovic, Jasmina; Sluban, Borut; Mozetic, I. (2015). Sentiment of emojis. *PLoS ONE*, *10*(12), e0144296. https://doi.org/10.1371/journal.pone.0144296

Oussalah, M., Escallier, B., & Daher, D. (2015). An automated system for grammatical analysis of Twitter messages. A learning task application. *Knowledge-Based Systems*, *101*, 31–47. https://doi.org/10.1016/j.knosys.2016.02.015

Patil, Rupali; Bhavsar, R.P; Pawar, B. V. (2016). Holy Grail of Hybrid Text Classification. *International Journal of Computer Science Issues (IJCSI), 05/2016, Volume 13, Issue 3*, *13*(3). https://doi.org/http://dx.doi.org/10.1108/17506200710779521

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., … Duchesnay, É. (2012). Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, *12*, 2825–2830. https://doi.org/10.1007/s13398-014-

0173-7.2

R. Jaya, S. Kumar, M. (2016). A STUDY ON DATA MINING TECHNIQUES , METHODS , TOOLS AND APPLICATIONS IN VARIOUS INDUSTRIES. *International Journal of Current Research and Review*, *8*(4), 34–39.

Sato, A., Huang, R., & Yen, N. Y. (2015). Design of fusion technique-based mining engine for smart business. *Human-Centric Computing and Information Sciences*, *5*(1), 23. https://doi.org/10.1186/s13673-015-0036-z

Tripathy, A., Agrawal, A., & Rath, S. K. (2016). Classification of sentiment reviews using n-gram machine learning approach. *Expert Systems with Applications*, *57*, 117–126. https://doi.org/10.1016/j.eswa.2016.03.028

Twitter Developer Documentation. (n.d.). Retrieved February 9, 2017, from https://dev.twitter.com/docs

Twitter Usage Statistics - Internet Live Stats. (2016). Retrieved February 9, 2017, from http://www.internetlivestats.com/twitter-statistics/

Upshall, M. (2014). Text mining. *Business Information Review*, *31*(2), 91–99. https://doi.org/10.1177/0266382114541180

Vercellis, C. (2009). 6. Chapter 5: Data mining. In *Business Inteligence: Data mining and optimization for decision making* (p. 18).

Wahyudi, M., & Putri, D. A. (2016). Algorithm application support vector machine with genetic algorithm optimization technique for selection features for the analysis of sentiment on twitter. *Journal of Theoretical and Applied Information Technology*, *84*(3), 321–331.

Wikarsa, L., & Thahir, S. N. (2016). A text mining application of emotion classifications of Twitter's users using Na??ve Bayes method. *Proceeding of 2015 1st International Conference on Wireless and Telematics, ICWT 2015*. https://doi.org/10.1109/ICWT.2015.7449218

Yang, L., Geng, X., & Liao, H. (2016). A web sentiment analysis method on fuzzy clustering for mobile social media users. *EURASIP Journal on Wireless Communications and Networking*, *2016*(1), 128. https://doi.org/10.1186/s13638-016-0626-0

Zeng, D., Chen, H., Lusch, R., & Li, S. H. (2010). Social media analytics and intelligence. *IEEE Intelligent Systems*, *25*(6), 13–16. https://doi.org/10.1109/MIS.2010.151