Application of Text Classification and Clustering of Twitter Data for Business Analytics

Alrence Santiago Halibas
Faculty of Computing Sciences
Gulf College
Muscat, Oman
alrence@gulfcollege.edu.om

Abubucker Samsudeen Shaffi
Faculty of Computing Sciences
Gulf College
Muscat, Oman
abobacker.shaffi@gulfcollege.edu.om

Mohamed Abdul Kader Varusai Mohamed
Faculty of Computing Sciences
Gulf College
Muscat, Oman
varusai@gulfcollege.edu.om

Abstract — **In the recent years, social networks in business are gaining unprecedented popularity because of their potential for business growth. Companies can know more about consumers' sentiments towards their products and services, and use it to better understand the market and improve their brand. Thus, companies regularly reinvent their marketing strategies and campaigns to fit consumers' preferences. Social analysis harnesses and utilizes the vast volume of data in social networks to mine critical data for strategic decision making. It uses machine learning techniques and tools in determining patterns and trends to gain actionable insights.**

This paper selected a popular food brand to evaluate a given stream of customer comments on Twitter. Several metrics in classification and clustering of data were used for analysis. A Twitter API is used to collect twitter corpus and feed it to a Binary Tree classifier that will discover the polarity lexicon of English tweets, whether positive or negative. A k-means clustering technique is used to group together similar words in tweets in order to discover certain business value. This paper attempts to discuss the technical and business perspectives of text mining analysis of Twitter data and recommends appropriate future opportunities in developing this emerging field.

Keywords— Twitter, Sentiment Analysis, Decision Tree, k-means, Social Media

I. Introduction

The social media has redefined the nature of how companies strategize their business processes. The social media contains a massive volume of unstructured data (e.g. tweets, comments, blogs, forum discussions, user post, and reviews) that can be used for business intelligence such as customer profiling and content analytics. Twitter, which is a social networking online service, is mainly used as a marketing and promotion tool by most companies. Specifically, twitter data contains not only user information, but also texts that contain subjective information (such as user sentiments) towards a particular issue. From a business perspective, the wealth of tweets is enough for companies to gather sufficient feedback about their products and services from their customers without having to spend for costly customer surveys and interviews. On the other hand, analyzing and extracting information from unstructured data poses a formidable challenge to data miners. Humans can easily find patterns and trends in documents but this ability is limited when a large amount of data is involved. However, with the help of analytical tools and techniques, such challenge is achievable. Hence, the industry that surrounds sentiment analysis is gaining momentum and currently the focus of social media research [1]. In fact, [2] forecasted that by 2022, the market for text analytics will rise to $8.79 billion with a growth rate of 17.2%. The report described that the increasing growth trend is attributed to the pressing need of companies for social media analytics, predictive analytics and the customization of their business applications. In a way, companies have seen the benefits of text mining using sentiment analysis in improving their services, monitoring of brand or company reputation, gaining competitive advantage, and other analytical uses with financial gains [3]. Its business value cannot be ignored, especially by marketing and customer support units of an organization.

This paper mainly examines the use of sentiment analysis in business applications. Furthermore, this paper demonstrates the text analysis process in reviewing the public opinion of customers towards a certain brand and presents hidden knowledge (e.g. customer and business insights) that can be used for decision making after the text analysis is performed. More so, [4] stressed that there is limited academic literature surrounding text analytics of Twitter data, as a result, this paper attempts to contribute in this developing field by providing a practical guide on how to mine and analyze customers' tweets.

This paper is divided into four (4) sections. Following this section is Section II which provides Background Information and Related Work of the following topics: Sentiment Analysis, Decision Tree, and Text Clustering. Section III demonstrates the Experimental Setup combined with the Results and Discussion and Section IV presents the Conclusion and Future Work.

978-1-5386-3761-6/18/$31.00 ©2018 IEEE

II. BACKGROUND AND RELATED WORK

A. *Sentiment Analysis*

Sentiment Analysis is used by companies to analyze data to understand the users' sentiments or opinions regarding their products or services. According to [5], Sentiment Analysis is "a process that automates the mining of attitude, opinions, views, and emotions from text, speech, tweets, and database sources through Natural Language Processing (NLP)". It is also referred to as opinion mining and emotion analysis. It uses textual data to analytically collect, analyze, model and validate for various business intelligence applications. Fig. 1 illustrates different approaches and techniques to sentiment analysis.

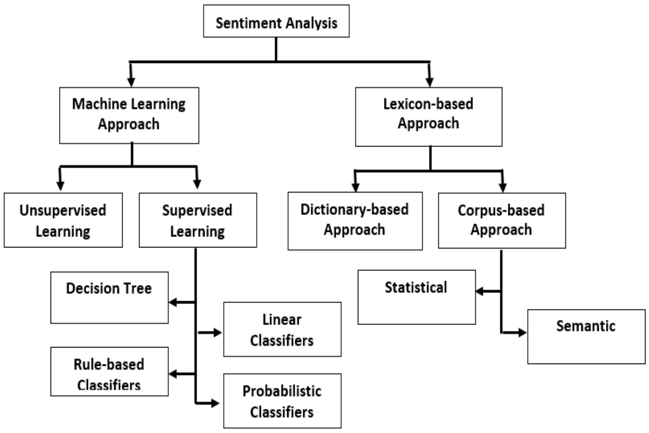


Fig. 1 Sentiment Analysis, *Source:* [4, Fig. 1]

Several studies have pointed out the importance and benefits of customer sentiment analysis for business operations. Majority of the purposes are for business improvement and decision support [7] [8]. In the study of [9] for instance, a decision support system using sentiment analysis with data mining was proven to be more efficient as opposed to the use of qualitative and quantitative methods in customer satisfaction research. The study pointed out that customer satisfaction is the key factor that influences decision making, thus, it is important to know customers' sentiments to manage the quality. Similarly, a case study conducted by [10] revealed a good use of unstructured text analytics for product decisions. The study stressed the significance of information triangulation from different unstructured text sources, including newspaper articles, to create confidence in the information for critical decisions. Moreover, text analytics is used by the Information Technology Services (ITS) of Florida State University to capture the stream of incoming service requests across business units [11]. Its success in extracting insights from the texts has allowed the company to provide an effective IT service management. It is said that when sentiments are positive, it is likely that the customers are satisfied [12]. Therefore, knowing how to positively exploit customers 'data presents infinite advantages for companies.

B. *Text Classification using a Decision Tree*

Text classification is an automated process of classifying text in natural language to predefined categories [13]. It is best used for predicting binary or nominal class labels [14]. A plethora of learning algorithms are used for text classification, including Naïve Bayes Classifier, Decision Trees, Support Vector Machines, Neural networks, and many others. The statistical approach in solving a classification problem involves two learning methods: supervised learning method (as shown in Fig. 2) which uses a training dataset to build the classification model prior to application in a test set, and unsupervised learning method which does not use any known labels in mining the data. Both of these methods produce quantitative evaluation results that can be easily reported.

A decision tree, which is of interest in this paper, is an example of a supervised learning method. Alpaydin [15] defines a decision tree as an efficient nonparametric method for classification and prediction of data. Furthermore, he states that it is a machine learning technique that uses a hierarchical data structure. According to [16], decision trees learn and respond quickly and interpretable, hence, they are normally used for classification. Moreover, decision tree algorithms are greatly utilized in data mining since they are proven to produce rational classification models and good accuracy levels [17]. The experiment of [18] revealed that the Decision Tree performed with 100% accuracy as opposed to Naïve Bayes' 86% accuracy performance. Because of a decision tree's simplicity to be understood and interpreted, and accuracy performance, it has been chosen as the machine learning technique for this experiment.

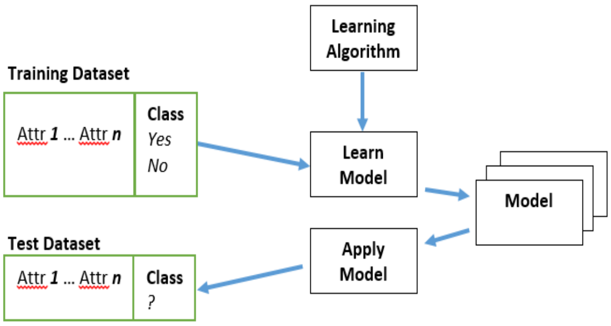


Fig. 2 Text Classification Building Model Approach, *Source:* [11, Fig. 4.3]

C. *Text Clustering*

Clustering is one of the commonly used unsupervised learning methods in analyzing the context of text data in natural language form [19]. It is a mathematical approach in collecting and segmenting similar documents into clusters. It helps trim down the volume of unstructured text and provide a simpler understanding and thematic structure of the data. It also provides the keywords in each cluster that is useful in extracting valuable insights, hence, customer sentiments can be summarized using these keywords. The clustering process is illustrated in Fig. 3.

K-means is a clustering technique in cluster analysis that finds a user-specified number of clusters that are represented by their centroids.

The basic k-means algorithm is described as:

```
1 Select k points as initial centroids
2 Repeat
3 Form k clusters by assigning each point to the closest centroid
4 Recompute the centroid of each cluster
5 Until centroids do not change
```

The algorithm iteratively assigns each point to one of the k groups based on the specified features and the points are clustered based on feature similarity [20]. Firstly, the points are assigned to initial centroids. Each point is assigned to the nearest centroid and the centroid of that cluster is updated by computing the mean of all points that are assigned to the centroid' cluster. This process is repeated until a stop criterion is satisfied.

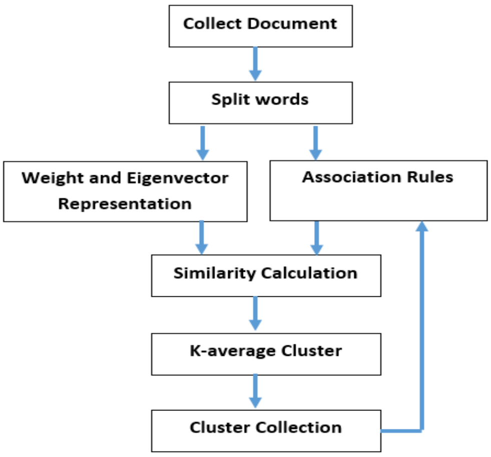


Fig. 3 Text Clustering, Source: [3, Fig.4]

III. EXPERIMENTAL SETUP

A. *Software Specification*

Rapidminer is a data-driven data mining tool that discovers patterns in large collections of text [21]. It is an analytic platform that integrates machine learning and predictive model deployment used by data scientists. It contains rich libraries of data science and machine learning algorithms [22]. The software used in this experiment is the latest version - RapidMiner Studio Educational version 8.0.001.

B. *Hardware Specification*

The experiment was developed using the following hardware specifications:

Processor: Intel Celeron CPU N2830 @2.16GHz 2.16GHz
RAM: 4 GB (3.98 GB usable)
System Type: 64-bit Operating System

C. *Text Classification Process Model*

The text classification process involves many tasks such as data extraction, data cleaning, pre-processing and feature extraction, and classification [23]. The detailed process diagram in Rapidminer is illustrated in Fig. 4. The processes applied to this experiment are each described in the succeeding sections.

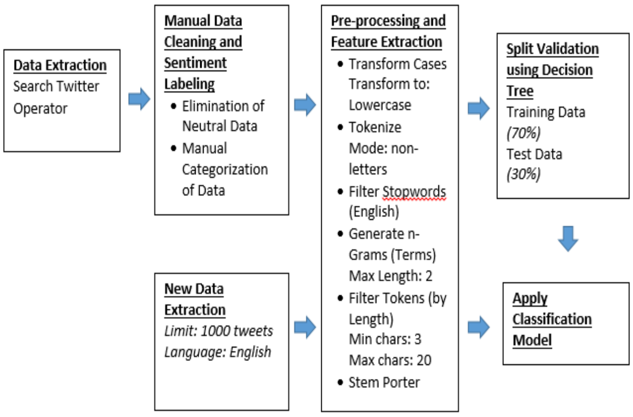

Data Extraction
Search Twitter Operator
Manual Data Cleaning and Sentiment Labeling
• Elimination of Neutral Data
• Manual Categorization of Data
Pre-processing and Feature Extraction
• Transform Cases Transform to: Lowercase
• Tokenize Mode: non-letters
• Filter Stopwords (English)
• Generate n-Grams (Terms) Max Length: 2
• Filter Tokens (by Length) Min chars: 3 Max chars: 20
• Stem Porter
Split Validation using Decision Tree
Training Data (70%)
Test Data (30%)
New Data Extraction
Limit: 1000 tweets
Language: English
Apply Classification Model

Fig. 4 Text Classification Process in Rapidminer

1) *Data Extraction*

An access token, which is used to get authentication access to extract data from the Twitter database, is given upon logging into a Twitter account. Authorization details that include the API key are required to establish the connection and allow a search query. An example set of Twitter is generated when setting the parameters for the Search Twitter operator. In this experiment, a query parameter is configured using the brand name of a popular food brand. Other search parameters include returned tweet and language limit of 500 English tweets, and the result type of recent or popular tweets. This contains attribute and label types including tweet ID, username, number of retweets, original text, date and time it was created, language, and many others. This example set is saved in an MS Excel file.

2) *Manual Data Cleaning and Sentiment Labelling*

The returned example set is reviewed by human labeler who was tasked to filter irrelevant data and retaining those which are significant to the experiment. The labeler was instructed to label the tweets as positive, negative, or neutral with an accompanying guideline. Out of the 500 tweets, 352 neutral tweets were discarded since they serve no purpose in the analysis. The remaining example set of 148 tweets (approximately 42%) which comprise the training corpora. The sentiment labeling is manually conducted by two (2) experts who classified each tweet as either positive or negative. Additionally, the experiment ensured that there is a balanced dataset, hence, it included 71 positively-labelled tweets and 76 negatively-labelled tweets.

3) *Feature Extraction*

A preprocessing stage is necessary to clean (or remove the "noise" in) the data in preparation to build the model. Five operations were applied at the preprocessing stage, namely Transform Cases, Tokenize, Filter Operator, Filter Token, Stem Porter to extract useful features of the data. The following describes each operation in order of execution:

a) The Transform cases operator converts the characters in a document to lowercase so that the words, “Hello” and “hello” are the same.

b) The Tokenize operator splits the texts of a document into a sequence of words (or tokens) by setting the non-letters mode as splitting points.

c) The Filter Stopwords (English) operator removes the unwanted words from a document such as is, are, the, of, etc. These words are commonly used in the text but are deemed useless when used because it provides no content information.

d) The Generate n-Grams (Terms) operator creates n-length of tokens in a document. The operator will check words that frequently follow one another. In text analysis, it is essential that tokens are grouped together in order to extract more meaning. Single tokens such as “health”, “living”, “summer”, and “breeze” may provide little information as opposed to paired (or 2 gram) tokens, such as “healthy_living” and “summer_breeze”. Thus, it will be easier to understand the context of the associated terms.

e) The Filter Tokens (by Length) operator removes tokens based on their length. For instance, if the minimum and maximum characters are set to 3 and 20, respectively, then it will remove texts having less than 3 characters and more than 20 characters in length.

f) The Stem Porter operator reduces the length of the words until a minimum length is reached or when the words are in its base form. For example, the words “electricity” is shortened to electric”, “revival” is shortened to “reviv”, “communism” is shortened to “commun”, and so on. This operation makes the comparison between words easier.

The result of the Process document operator is a list of words with total and document occurrences. In this case, the operation produces 1819 word list.

In between the Process Document and Split Validator operators are two operators, namely Set Role and Select Attributes operators. The Set Role is used to identify the target attribute role (label). The Select Attributes selects attributes with no missing values. The example set contains 1819 regular attributes. Each attribute is listed separately in columns. A Term Frequency-Inverse Document Frequency (TF-IDF) score, which is the weighting factor, is assigned to each attribute in order to identify its significance to a document in a corpus. A TF-IDF score of 0 indicates that the term did not appear in a document, whereas a TF-IDF score of 0.617 means that the term is significant as opposed to a TF-IDF score of 0.283.

4) *Split Validation using Decision Tree*

The Split Validation operator contains three (3) other operators, namely Decision Tree, Apply Model and Performance.

This experiment used supervised learning, which has a training data set and a test data set, in training the classifier to identify the polarity. This method is essential in training the classifier on how to predict the polarity of unknown Twitter data. The classification model is first tested and built using the training dataset and the classification accuracy is run using the test model.

Table 1 shows the performance vector for a decision tree with varying split ratios. A split ratio that is set to 50%-50% means that 1/2 of the data is used in the training while the remaining 1/2 of the data is used in the testing. The table further shows that the accuracy is better for a 70%-30% split ratio as compared to the other two split ratios, hence, this split ratio (with Accuracy of 70.45% and Classification Error of 29.55%) is used in this experiment. This experiment ensured that appropriate parameters that can provide the best possible model to maximize the prediction are selected.

TABLE 1 Performance Vector for Decision Tree

	Split Ratio		
	50%-50%	**70% - 30%**	**80% -20%**
Accuracy/ Classification Error	53.42% / 46.58%	70.45% / 29.55%	68.97% / 31.03%

5) *Apply Model*

The Apply Model operator applies a model on an example set. In this experiment, the Decision Tree mode, which was first trained, is applied as illustrated in Fig. 1. A new example set of 1000 tweets was extracted, pre-processed, and feed into the model to predict the polarity of each tweet. This example set is processed to be compatible with the model with the same attributes that were used to generate the model.

6) *Results and Discussion for Text Classification*

There are several metrics that measure the performance of the prediction. Some of the most widely used metrics are accuracy, precision, and recall. These metrics use four (4) prediction outcomes that are defined in Table 2.

TABLE 2 Prediction Outcomes

Abbr	Outcome	Description
tp	true positive	No. of correct positive prediction
tn	true negative	No. of correct negative prediction
fp	false positive	No. of incorrect positive prediction
fn	false negative	No. of Incorrect negative prediction

Precision is computed as the number of true positives divided by the total number of predicted positive while Recall (also known as Sensitivity) is computed as the number of true positive divided by the total number of positives. Likewise, Accuracy is computed as the total number of correct predictions divided by the total number of the dataset. The formulas for accuracy, precision, and recall are as follows:

$$Precision = \frac{tp}{tp + fp}$$

$$Recall = \frac{tp}{tp + fn}$$

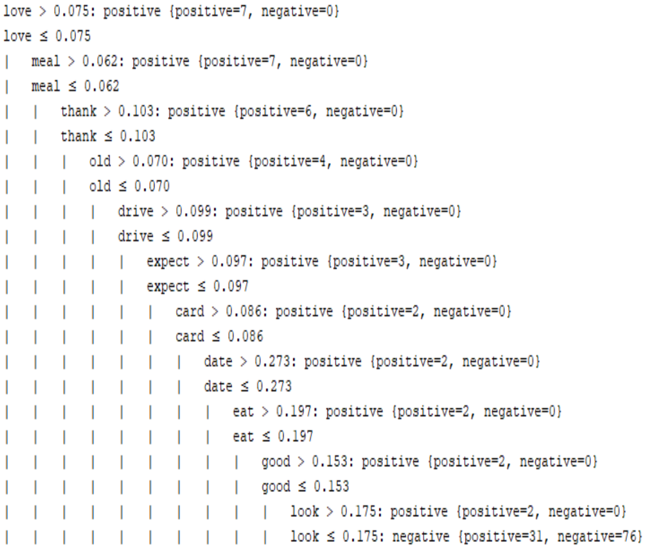
$$Accuracy = \frac{tp + tn}{tp + tn + fp + fn}$$

Table 3 presents the confusion table that summarizes the prediction performance of the binary classification model.

TABLE 3 Confusion Matrix

	True Negative	True Positive	Class Precision
Predicted Negative	23 (tn)	13 (fn)	63.89%
Predicted Positive	0 (fp)	8 (tp)	100%
Class Recall	100%	38.10%	

From the test data, the model correctly predicted 8 positive terms with no predicted error on false positives, thus, garnering a precision of 100%. On the other hand, the model correctly predicted 8 positive terms but wrongly predicted 13 positive terms, thus garnering a recall of 38%. The accuracy of the model is 70.45%.



```
love > 0.075: positive {positive=7, negative=0}
love ≤ 0.075
|   meal > 0.062: positive {positive=7, negative=0}
|   meal ≤ 0.062
|   |   thank > 0.103: positive {positive=6, negative=0}
|   |   thank ≤ 0.103
|   |   |   old > 0.070: positive {positive=4, negative=0}
|   |   |   old ≤ 0.070
|   |   |   |   drive > 0.099: positive {positive=3, negative=0}
|   |   |   |   drive ≤ 0.099
|   |   |   |   |   expect > 0.097: positive {positive=3, negative=0}
|   |   |   |   |   expect ≤ 0.097
|   |   |   |   |   |   card > 0.086: positive {positive=2, negative=0}
|   |   |   |   |   |   card ≤ 0.086
|   |   |   |   |   |   |   date > 0.273: positive {positive=2, negative=0}
|   |   |   |   |   |   |   date ≤ 0.273
|   |   |   |   |   |   |   |   eat > 0.197: positive {positive=2, negative=0}
|   |   |   |   |   |   |   |   eat ≤ 0.197
|   |   |   |   |   |   |   |   |   good > 0.153: positive {positive=2, negative=0}
|   |   |   |   |   |   |   |   |   good ≤ 0.153
|   |   |   |   |   |   |   |   |   |   look > 0.175: positive {positive=2, negative=0}
|   |   |   |   |   |   |   |   |   |   look ≤ 0.175: negative {positive=31, negative=76}
```

Fig. 5 Tree Description

Fig. 5 provides the Decision Tree description. The minimal leaf size is set at 2 (positive and negative). The minimal gain ratio is set at 10% since the tree produced a good representative of words given this value, otherwise, this value will be decreased to let the tree grow. The tree depth is set to a maximum depth of 15 in order to make it readable and comprehensible

The attributes that contribute to the improvement of the tree are shown in Fig. 5. For instance, the attributes: love, meal, thank, old, drive, expect, card, date, eat, good, look are highly significant to the label. Moreover, if the TF-IDF score for the term “love” is greater than 0.075, then the attribute is categorized as positive, otherwise, it is negative. The TF-IDF score of each term are listed in Fig.5.

TABLE 4 Prediction Classification Statistics

Nominal Value	Absolute Count	Fraction	Confidence			
			Min	Max	Average	Dev.
Negative	852	0.852	0	0.710	0.605	0.252
Positive	148	0.148	0.290	1	0.395	0.252

Table 4 shows that the model predicted approximately 85% negative comments with a confidence value of 0.605 as opposed to approximately 15% positive comments with a confidence value of 0.395. This result poses a highly significant issue to a company or brand because it suggests a negative branding to the business. With this kind of information, a company can exercise a more informed critical decision. More so, there are anecdotal evidences that support the correlation of sentiment and brand reputation. In fact, [24] suggests that it is essential to know the influence of Twitter data to a brand reputation. In this case, a company can act upon this serious issue immediately.

D. *Text Clustering Process Model*

The clustering process, which is implemented in Rapidminer, is shown in Fig. 6. It includes data extraction, pre-processing and feature extraction, and clustering. The subsequent sections described the processes that are applied to this experiment.

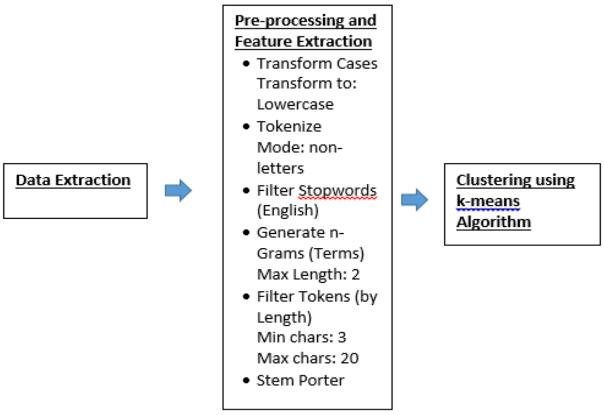

Data Extraction
Pre-processing and Feature Extraction
• Transform Cases Transform to: Lowercase
• Tokenize Mode: non-letters
• Filter Stopwords (English)
• Generate n-Grams (Terms) Max Length: 2
• Filter Tokens (by Length) Min chars: 3 Max chars: 20
• Stem Porter
Clustering using k-means Algorithm

Fig. 6 Text Clustering

1) *Data Extraction*

The Read Excel operator was used to read the example set from an Excel file.

2) *Pre-processing and Feature Extraction*

The aforementioned pre-processing and feature extraction process is likewise applied to this experiment. Similarly, the output of this process was feed to a Data-to-Similarity operator which computes the similarity of each example with every example in the given example set.

3) *Clustering using k-means Algorithm*

The clustering process uses the k-means operator. Since clustering is categorized as an unsupervised learning, it does not need a label attribute. K-means is efficient even with multiple runs. This experiment performed 10 repeated run times. It simply determines the set of k clusters and assigns each example to a specific cluster as shown in Fig. 7.

4) *Results and Discussion for Text Clustering*

In Fig. 7, the end process generated four (4) clusters with corresponding items: Cluster 0 with 28 items, Cluster 1 with 920 items, Cluster 2 with 33 items, and Cluster 3 with 19 items. The words in each cluster provide insights on what the cluster is all about. For simplicity, Cluster 3 is chosen for evaluation due to its reasonable number of items.

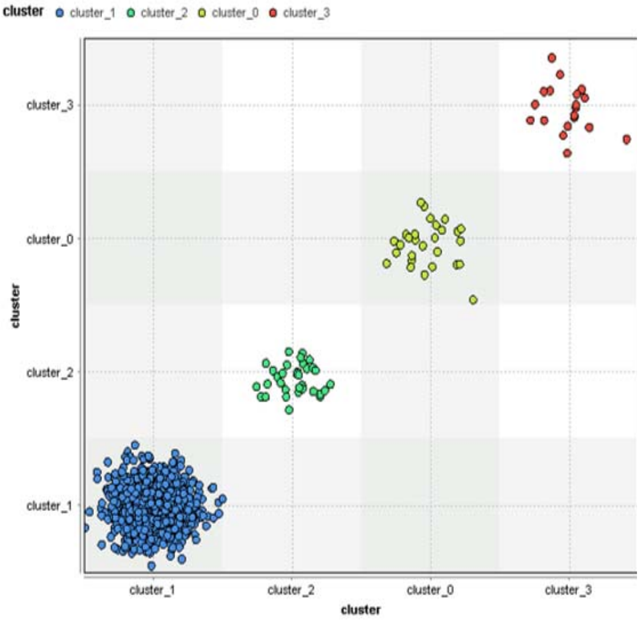


Fig. 7 Cluster Results

Table 5 lists the significant attributes (or words) in Cluster 3. Note that some words were omitted (*) as it reveal significant clues on the brand name that is used for this experiment. This may flag ethical issues when ignored.

TABLE 5 Cluster 3 attributes

boycott_*	boycott_*	boycott_sponsor	cat
cat_meat	dog	dog_cat	evil
evil_boycott	*	*_boycott	meat
meat_trad	sponsor	trade	trade_evil
winter	boycott	olympi	olympi_dog

As seen in Table 5, only two negative words, such as *boycott* and *evil*, are included in the list. Although less significant as single words, when paired up with the other words the insights ultimately change. Moreover, when these insights are combined with the result in the text classification having 85% negative classification prediction, the theme of the insights will be strengthened. Combining the words in Table 5 will create meaningful information that can be used for critical decision making.

IV. CONCLUSION AND FUTURE WORK

The outcome of evidence-based decision making contributes to the improvement of a brand. Having a text analysis of customer feedback and reviews allows effective quality management. With sentiment analysis, companies can now strategically reposition their businesses according to customers' sentiments.

This paper provided an introduction and rationale behind the value of text analytics of Twitter data to businesses in gaining customer views on products and services, and brand. This paper also discussed several related work in sentiment analysis for business applications. Importantly, it demonstrated a practical application of text classification and clustering of Twitter data, and revealed ways on how to analyze these to gain business insights.

Although the classification accuracy rate for this experiment is already acceptable in this application domain. It is suggested that future work needs to increase the accuracy of the classification model by improving data preparation and experimenting with other classification algorithms.

Future work in this field can also be focused on real-time analytics of Twitter data stream. Since there is a massive amount of tweets collected daily, handling real-time analytics is difficult. Therefore an automated sentiment analysis, which runs in high processing and large memory computing resources, is required.

REFERENCES

[1] B. Liu, "Sentiment Analysis and Opinion Mining," *Synth. Lect. Hum. Lang. Technol.*, vol. 5, no. 1, pp. 1–167, 2012.

[2] marketsandmarkets.com, "Text Analytics Market by Component (Software, Services), Application (Customer Experience Management, Marketing Management, Governance, Risk and Compliance Management), Deployment Model, Organization Size, Industry Vertical, Region - Global Forecast to 20," 2017.

[3] F. N. Ribeiro, M. Araújo, P. Gonçalves, M. André Gonçalves, and F. Benevenuto, "SentiBench - a benchmark comparison of state-of-the-practice sentiment analysis methods," *EPJ Data Sci.*, vol. 5, no. 1, 2016.

[4] A. Moreno and T. Redondo, "Text Analytics: the convergence of Big Data and Artificial Intelligence," *Int. J. Interact. Multimed. Artif. Intell.*, vol. 3, no. 6, p. 57, 2016.

[5] V. A. Kharde and S. S. Sonawane, "Sentiment Analysis of Twitter Data: A Survey of Techniques," *Int. J. Comput. Appl.*, vol. 139, no. 11, pp. 975–8887, 2016.

[6] W. Medhat, A. Hassan, and H. Korashy, "Sentiment analysis algorithms and applications: A survey," *Ain Shams Eng. J.*, vol. 5, no. 4, pp. 1093–1113, 2014.

[7] L. Ziora, "The sentiment analysis as a tool of business analytics in contemporary organizations," *Stud. Ekon.*, pp. 234–241, 2016.

[8] S. Yaram, "Machine learning algorithms for document clustering and fraud detection," in *Proceedings of the 2016 International Conference on Data Science and Engineering, ICDSE 2016*, 2017.

[9] N. Yussupova, M. Boyko, and D. Bogdanova, "A Decision Support Approach based on Sentiment Analysis Combined with Data Mining for Customer Satisfaction Research," *Int. J. Adv. Intell. Syst.*, vol. 1&2, 2015.

[10] S. K. Markham, M. Kowolenko, and T. L. Michaelis, "Unstructured Text Analytics to Support New Product Development Decisions," *Res. Technol. Manag.*, vol. 58, no. 2, pp. 30–39, 2015.

[11] O. Muller, I. Junglas, S. Debortoli, and J. Von Brocke, "Using Text Analytics to Derive Customer Service Management Benefits from Unstructured Data," *MIS Q. Exec.*, vol. 15, no. 4, pp. 64–73, 2016.

[12] P. Khobragade and V. Jethani, "Sentiment Analysis of Movie Review," *Int. J. Adv. Res. Comput. Sci.*, vol. 8, no. 5, p. 19411948, 2017.

[13] S. M. Kamruzzaman, F. Haider, and A. R. Hasan, "Text Classification using Data Mining," *Science (80-.).*, p. 19, 2010.

[14] T. Pang-Ning, M. Steinbach, and V. Kumar, *Introduction to data mining*. 2006.

[15] E. Alpaydin, *Introduction to Machine Learning*. 2004.

[16] K. M. Sreerama, "Automatic Construction of Decision Trees from Data: A Multi-Disciplinary Survey," *Data Min. Knowl. Discov.*, vol. 2, no. 4, pp. 345–389, 1998.

[17] R. C. Barros, A. C. P. L. F. de Carvalho, and A. A. Freitas, *Automatic Design of Decision-Tree Induction Algorithms*. Springer International Publishing, 2015.

[18] A. Jain and P. Dandannavar, "text analytics framework using apache spark and combination of lexical and maching learning techniques," *Int. J. Bus. Anal. Intell.*, vol. 5, no. 1, pp. 36–42, 2017.

[19] M. Allahyari *et al.*, "A Brief Survey of Text Mining: Classification, Clustering and Extraction Techniques," *arXiv Prepr. arXiv*, vol. 1707, no. 2919, pp. 1–13, 2017.

[20] A. Trevino, "Introduction to K-means Clustering," *Datascience.com*. 2016.

[21] M. Hofmann; and R. Klinkenberg;, "RapidMiner: Data Mining Use Cases and Business Analytics Applications," *Zhurnal Eksp. i Teor. Fiz.*, 2013.

[22] K. S. Rawat, "Comparative Analysis of Data Mining Techniques, Tools and Maching Learning Algorithms for Efficient Data Analytics," *JOSR J. Comput. Eng.*, vol. 19, no. 4, pp. 56–60, 2017.

[23] H. Kaur and V. Mangat, "Dictionary based Sentiment Analysis of Hinglish text," *Int. J. Adv. Res. Comput. Sci.*, vol. 8, no. 5, pp. 816–822, 2017.

[24] M. H. Peetz, M. De Rijke, and R. Kaptein, "Estimating Reputation Polarity on Microblog Posts," *Inf. Process. Manag.*, vol. 52, no. 2, pp. 193–216, 2016.