International Journal of Engineering & Technology, 8 (1.11) (2019) 33-39

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET

Research paper

Short Text Mining: Machine Learning and Statistical Modelling Approaches Compared

Omar H. Al-Barahamtoshy

Information System Department, Cairo, Egypt
**E-mail:* oalbarahamtoshey@gmail.com

Abstract

With the growth of technology, social media has gained popularity and now plays a key role in modern day to day communication. Given such trend, social media has gained increasing influence on our society to the extent it has become a part of our language to say I am going to "Tweet" about some thought. Like any community driven content, people find complex means to interact with each other. Twitter offers people the ability to tag their tweets with hashtags to specify the topic of tweet's content. However, like any community driven convention, there exists many tweets which do not have hashtags. In this paper, we seek to explore the methods in the literature that can categorize tweets without hashtags. We have evaluated one method, which proved to be very promising due to its flexibility and extensibility to many applications. We also discuss future enhancement and extension possibilities, and provide a critique of the current method's drawbacks.

Keywords: *Tweets, Topic Categorization, Short Text Mining, Natural Language Processing, Online Learning, Knowledge Transfer*

1. Introduction

Social media has become a key communication tool for many of our planet's population. Over a 1.86 billion Internet Monthly active users use Facebook [https://www.statista.com/statistics/264810/] and 300 million daily active tweeters [https://www.statista.com/statistics/282087/]. It has even become an integrated part of modern age slang. Just like if you want to search for something, you "Google it", if you want to contact a friend you just met, you "friend them on Facebook" and if you want to share new thoughts with the world, you "Tweet it". It has gone beyond the normal usage to being a part of everyday lives; it is now a source for news, advertisement campaigns, opinion polling and even a platform for organizing demonstrations and revolutions such as the Arab Spring in 2011.

This has spurred the need for categorization of social media posts. Specifically, extracting tweets related to an interesting topic has shown a growing demand, for many different applications, such as opinion statistics for governments, companies, personalized and targeted advertisement, crime anticipation and hate speech recognition.

The structure of a tweet is simple, it's a short text message of only 140 characters called a "tweet" that users post to their profiles. Hashtags are words preceded by the '#' symbol. Users use hashtags to "tag" their tweets in order to relate it with a specific topic. Hashtags help users find things others say about a specific topic. However, hashtags are community-driven convention and are not automatically assigned to tweets. There exists a large amount of tweets without hashtags. Also, the same topic might have multiple hashtags for example, #GameOfThrones and #GoT are both hashtags related to the TV

Series "Game of Thrones". Also, many tweets could be related to a specific topic, but the user might not put the relevant hashtag for many reasons that may include the 140 character limit, that the tweet is in a series of tweets talking about the same topic, that the user does not know a suitable hashtag, and sometimes because the user is lazy.

Finding tweets related to a topic without hashtags can be difficult using currently available search tools, since you can only search using a keyword. For instance, if a user is writing a series of tweets about a topic, you might find a hashtag in only the first one, therefore, when searching; you will only be able to obtain the rest of the tweets by getting them from the user's profile. Of course, it is very difficult to automate this for any applications such as opinion statistics.

Among the common commercial applications do some companies to see if the user's interaction with their administrators is positive use profile monitoring or not. Many companies would like to understand the sentiment of users' replies to administrators as a form of customer satisfaction monitoring. In addition, companies would want to be alerted if any discussion goes off topic and therefore they need to know if a certain tweet is an outlier to the cluster of common discussions.

Copyright © 2019 Authors. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

2. Research context and questions

With this background and motivations, we ask the following research questions
—RQ1 Is it possible to detect a tweet's topic without hashtags?
—RQ2 Is it possible to suggest hashtags to users based on their tweet content?
—RQ3 Is it possible to detect whether a tweeter's tweet is in concordance with their other tweets?
As noted in the previous section, users sometimes do not tag their tweets with hashtags because of many reasons that may include the 140-character limit, or that the tweet is in a series of tweets talking about the same topic. Detecting relevant tweets would serve greatly to better suggestions in search results for users from tweets, which do not have hashtags. Also, when writing a tweet, the user might not know a suitable hashtag, and sometimes the user is just lazy. Suggesting a relevant hashtag will help the users find suitable hashtags and might encourage them to follow the convention if they are lazy.

3. Literature review

Due to popularity in common social media applications, such as twitter, short text mining has attracted growing interest in recent years. Three main trends were found to be gaining popularity. The first trend is the use of traditional text mining algorithms such as bag of words, term expansion, directly on tweet datasets, and attempting to add meta-information in order to enhance performance [1; 2; 3]. The second trend is knowledge transformation, by training learners such as Latent Dirichlet Allocation[4, 5] on large annotated text datasets, then attempting to use the learner on short text data[6, 7]. Third trend is using non-machine learning algorithms, such as data compression and measuring the compressibility of test tweets when concatenated to training tweets [8].

3.1 Traditional Text Mining Approach

As an example for the first trend, [9] describe their work as an intuitive approach, that extracts a set of tweet features, focusing on the twitter user intentions, to assign a class label for each tweet. Their focus on features set used contains unigrams, POS tags, emoticons, sentiments of words and POS tags of sentiments.
Classification is based on features extracted from tweets and it takes into consideration the author's information as well. Because the classes being used are generic classes, feature extraction proved to be a problem for the authors. Therefore, a greedy strategy was used to extract a set of relevant features. Features included: authorship, time/date, emphasis words and characters, the "@" sign for the Private message class (referred to on twitter as a mention), etc.
The datasets used consists of 5407 tweets from 684 twitter users. The dataset is all in English. Tweets have been manually labelled and there then stop words were removed putting the dataset at 6747 unique words.
The publicly available WEKA implementation of Nave Bayes classifier is used with a 5-fold cross validation model.

3.2. Knowledge Transformation

Li et al [10] argue that including a tweet author's information with the traditional Bag Of Words approach, makes the BOW with authorship (BOW-A) far superior to BOW because twitter users or authors usually follow a tweeting pattern that puts most of their tweets into a small number of categories or classes and continued to support that with the experiment results. The comparison between the traditional BOW, BOW -A and BOW-8F showed that the proposed 8F improved results or accuracy by 32.1% over the traditional BOW while BOW-A improved results only by 18.3% over the traditional BOW. As an example of the second trend, we discuss [11] whose goal is to monitor the world's events through a real-time analysis of the Twitter stream. In particular, authors aim at detecting the topic or class of each tweet in the stream upon arrival basis or upon hitting the streamline in a real life situation.
What sets this work distinct from previous work is that the set of topics are not known a priori, and hence it is a problem of unsupervised classification. Classical techniques depending on analyzing the tweet's text directly is difficult to apply due to the small size of a tweet (usually limited to 140 characters by definition). So, finding semantic distances between tweets (which is the first step for tweet categorization) is not possible because most of the context is missing. Previous work tries to infer the context using sentiment analysis and emoticons within the text but they required the set of topics to be predefined.
Instead, the authors tap on Wikipedia to recover the context. Other works used Wikipedia as a training set for supervised classification; however, this paper uses it in a rather unique way. In particular, each tweet is associated with the Wikipedia page, which has the largest possible amount of words from the tweet. Afterwards, the distance between tweets is set to be the distance between their associated Wikipedia pages, measured by the length of the shorted path between the two pages' categories in the Wikipedia category tree (which is a taxonomy).
Afterwards, Multidimensional Scaling (MDS) is used to compute a 2D vector for each tweet in the dataset such that the Euclidean distance between any two vectors is as close as possible to their Wikipedia distance. This set of vectors can be used to visualize how tweets are clustered. Moreover, algorithms such as discriminant function analysis can be used for unsupervised clustering on the set of vectors. The authors then evaluate the prediction accuracy of this method using leave-one-out cross validation and show that it is superior to two other techniques: 1) the string edit distance and 2) latent semantic analysis (LSA) on the term-tweet co-occurrence matrix. Moreover, they show that their technique is far more robust to random noise than LSA.
Finally, they remark as future work the possibility of using different distance measures between Wikipedia pages, such as the LSA on the work-page co-occurrence matrix. *on-Machine Learning Algorithms*

As an example of the third trend, we discuss [11, 12] whose goal is to filter interesting tweets, that is, tweets related to a user-defined query, from other tweets using data compression. It also adds the ability to handle multiple languages, internet slang and misspelled words which are

hard to deal with using approaches like BOW unless these cases are covered individually. The Compression-based Tweet Classification (CTC) method classifies tweets by determining their compressibility against both the positive and negative examples. Dataset consists of user defined tweets (tweets defined by a single hashtag).
The approach uses an approximation of the conditional Kolmogorov complexity of the tweet under question to two different sets of tweets: N most recent tweets, which match the query and N most recent tweets, which do not. If the approximate conditional Kolmogorov complexity were low, it would indicate a strong relationship between the tweet under question and the corresponding set of tweets. To decide to which set of tweets (interesting tweets or uninteresting tweets) the tweet under question should be classified, the authors use the ratio between the approximate conditional Kolmogorov complexities to both sets and decide in favor of the numerator if the ratio was below a given threshold. The authors then verify the effectiveness of their approach compared to two traditional machine learning algorithms and highlight the crucial advantage that their approach is language agnostic. Their approach has been applied in different contexts before including spam filtering, authorship attribution, language recognition and other fields as well such as biology and music. The authors note that although Twitter mining was studied in the literature, they were the first to apply the approximate conditional Kolmogorov complexity to achieve it. Since, we were interested in the aspects in [13, 14]. Because of its simplicity, ability to process multilingual data without code or algorithm changes beside the fact that it does not need any semantic preparations for the tweets of the dataset, as discussed in details in section 4, we decided to further investigate this model or approach and use it as our approach to answer our research questions.

While exploring this model, there were some notable errors in the evaluation method or their interpretation. For instance, category #f1 detection reported AUC (see Fig 1) is below 0.2 for all of the methods investigated in [15] including that of the one being proposed in the paper. Moreover, despite the authors considering this to be a rather low AUC, inverting the output of the classifier will result in a result of 0.8. The reasoning for this is that in the ROC space (precision vs recall), inverting the output of the classifier, inverts the corresponding AUC. The reason for this is simple, in the ROC space; a perfect classifier is the one that predicts zero false positives and all true positives. However, if a classifier predicts 0 true positives and all false positives, this means it's also a perfect classifier, because simply you could state that if the classifier outputs true, you will consider it to be false, and false to be true and therefore inverting the ROC curve and the AUC (for further illustration, a bad classifier would be one with AUC 0.5, it would be equivalent to random guessing). Furthermore, this interpretation made in the paper results in another error in the results as, under the new light, two of the methods the authors are comparing against their own will outperform the author's CTC by a significant margin.

However, since we are interested in the ability to process data from different languages without change and the fact that it does not need semantic preparation for the tweets, as we are going to discuss in detail in section 4, and to be able to correctly judge the method, we decided to choose this method for investigation for our study.

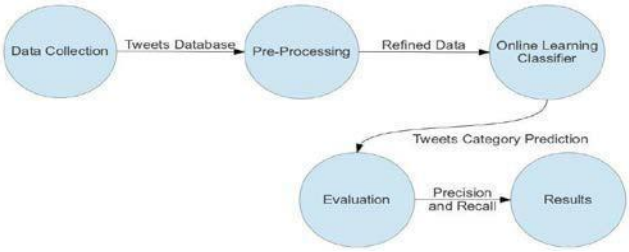


Fig. 2: Flowchart of the research method

3.3. Data Collection

We have crawled tweets from 4 hashtags and other random tweets using a query of stop words, using twitter streaming API[2]. Categories and tweet counts are displayed in Table I. The process of collecting the tweets started at the same time on 02:00 GMT 2013-12-24, and ended on 23:39 GMT 2013-12-28. The difference in the number of tweets is due to the limitations imposed by the twitter API and the rate of tweets per topic. This variance in tweet count however should have no effect on the performance of the algorithm because of two steps we took. The first is we build a separate model for every individual category. Moreover, the second is that we use a 10-fold cross validation (for more details see subsection 4.4.2).

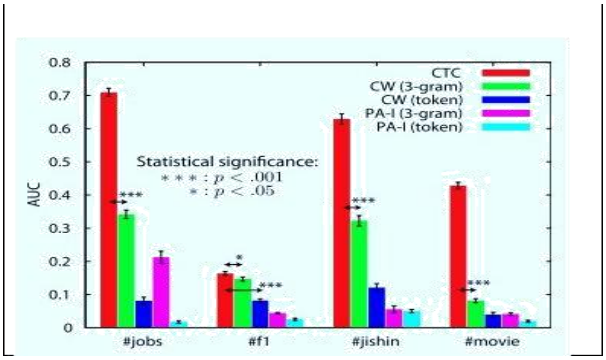


Fig. 1: Excerpt from: Mean AUC values under PR curve for each hashing. Errors bars indicate standard errors of 10 -fold CV. Paired t-test with the Holm correction for multiple comparisons were conducted

4. Method

In this section we describe in detail the procedure and steps we took to get our results. An overview of the procedure is shown in Fig. 2. We first start by collecting data; the data we have collected contain 376,808 tweets from five categories using Twitter's streaming API[1] over the course of 4 days. The next step was pre-processing. We have not done a lot of pre-

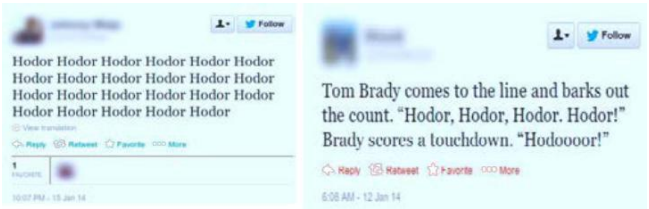


Fig. 3: Two tweets showing unusual usage of the character name "Hodor"

Table 1: Collected tweets dataset

Hashtags	Tweets Count
#movie	7,515
#Obama	6,828
#iPad	110,294
#gameofthrones	10,146
other	242,025

4.1. Data Preprocessing

Misspellings and Internet slang is very common in tweets, also made-up abbreviations and usages of words in uncommon contexts are often present due to both the 140 character rule and the uncommon wording that is influenced by the internet culture, i.e. memes, and can't be mitigated with BOW approaches. Methods relying on knowledge transformation require extensive clean-up and pre -processing to deal with such problems because of the nature of long texts, from which they learn, as they are usually well written documents with proper grammar and

1 https://dev.twitter.com/streaming/overview

2 http://socialmedia-class.org/twittertutorial.html

spelling. However, the data compression measure will be mildly affected by such problems because all of the information in use come from the same dominion of tweets. So even if some user when speaking about a certain topic was using new slang or abbreviations, most probably many other users will use them as well. Actually, the way of writing slang and abbreviations, might be considered as a feature of the topic itself. For instance, using a word like "Hodor" in uncommon contexts such as just repeating the word, is in fact a deterministic feature of a tweet related to the TV series (and hashtag) "#GameOfThrones", because even though the word is the name of one of the characters on the show, it's the only word the character ever says, and as usual, twitter community has adopted the word into entertaining and new contexts which would require custom exception rules if it was to be treated by normal semantics clean-up methods. An example for such usage is the two tweets shown in Fig. 3.

4.2 Algorithm

We have applied the data compression measure. In this section we describe the algorithm steps and in subsection 4.4.2 we describe in detail our setup and parameters for the algorithm. This is an online learning algorithm i.e. an algorithm which constantly evolves and modifies itself according to the data it witnesses. The algorithm starts by building a data model category. Each model is made of two sets of training tweets, one set contains tweets belonging to the category *positive tweets* and the other set contains tweets which don't *negative tweets*. The number of tweets in the model is kept constant at a certain count *N*. If we want to test whether a tweet is positive or negative (belong to the class in question or not), we first compress each of the sets, and make note of the size of the output, then add the testing tweet to both sets and compress them again. Subtracting both compression sizes (before adding tweets and after), results in a compressibility measure, which can be seen as a measure of the degree of "belongingness" or content similarity of the tweet to each of the sets or the classes, positive *Cp* or negative *Cn*. This measure can be more formally described as a Kolmogorov complexity (the Kolmogorov complexity of a string is the length of the shortest program to specify the string on a universal computer). We compare the compressibility $f(x)$ for both positive and negative sets by dividing them along with a normalisation variable γ. Formally, the definition of compressibility is as in equation (1).

$$f(x) = \frac{C_p(x) + \gamma}{C_n(x) + \gamma}. \tag{1}$$

Where $Cp(x)$ and $Cn(x)$ are defined by

$$Cp(x) = Z(Mp.x) - Z(Mp) \tag{2}$$

$$Cn(x) = Z(Mn.x) - Z(Mn) \tag{3}$$

Where Z is any compression function, and Mp and Mn are the set of N positive and negative tweets respectively. The compression function can be any of the well-known algorithms for compression such as Deflate[3] (used in gzip), block-sorting[4] ,

prediction by partial matching[5], and/or dynamic Markov compression[6]. We have resorted to using the Deflate algorithm for its efficiency and speed. We have tried several implementations in different languages that include GPU computing using OpenCL, however we noticed little gain in the overall time because for our needs, the normal implementation cited and was sufficient. If ratio of the compressibility measures ratio is above or below a certain threshold, the tweet is classified as positive or negative. If the tweet is a training tweet, we add the tweet to the positive set, or to the negative set of tweets and if the number of tweets in a set is more than *N*, the oldest tweet is removed from the set. The γ normalization variable allows the model to control its recall and precision trade-off by giving more bias towards one of the two decisions. A simple flowchart of the procedure is shown in Fig 4.

4.3 Experimental Setup

In this section, we describe the experimental setup and specific parameter values used for the algorithm and detail our dataset refinement and evaluation strategy.

4.3.1 Data Setup.

As we mentioned in subsection 4.2, no special semantic clean up, no spelling correction nor slang transformation is needed to refine the dataset. However, we removed all retweets since they are redundant and therefore correlated information and could cause the data to lose its independent identical distribution and therefore falsely affect performance in a positive or negative manner. We also removed user names since, even though they might contribute to the categorisation, but are sometimes irrelevant and could act as noise. Lastly, in order to test our research question regarding tweet categorisation and suggestion without hashtags, we created two copies of the dataset and removed all hashtags from one of them, in order to evaluate the effect of hashtag presence on the decision of categorisation. We also considered tweets from one language, English. However, since the algorithm does not make use of any language semantics, the algorithm can be applied in the same fashion to tweets from any language. The resulting tweet statistics after refinement is shown in Table II. We have then concatenated all the tweets from all the categories into one big chunk, and sorted them in a chronologically ascending manner.

4.3.2 Algorithm Setup

As we have described in section 4.3, the algorithm builds a model for every category, containing two sets of positive and negative tweets. The tweet is then added to both sets as illustrated in Table II. A compressibility measure of the tweet is calculated for both sets by subtracting the compression size before and after adding the tweet. Both measures are then compared (divided with a normalization variable) and by comparison to a threshold, the tweet is categorised. We now describe in detail what we did in our experiment. We have repeated the following procedure twice, once for the dataset with hashtags and once for the dataset without hashtags.

Table 3: Collected tweets dataset

[3] https://en.wikipedia.org/wiki/DEFLATE	5https://en.wikipedia.org/wiki/Prediction_by_partial_matching
[4] https://en.wikipedia.org/wiki/Burrows%E2%80%93Wheeler_transform	6https://en.wikipedia.org/wiki/Dynamic_Markov_compression

Hashtags	**Tweets Count**
#movie	5,497
#Obama	2,834
#iPad	79,539
#gameofthrones	5,530
other	106,723

We started off with 4 models, one for each category, note that even though there are 5 categories (#movie, #Obama, #gameofthrones, #iPad and other), our classification is a binary decision of whether a tweet belongs to a given category or not, and therefore for every model, the positive set would include training tweets belonging to that category, and the negative set would include tweets not belonging to it (i.e. from any of the other 4 categories). We fixed our model set size N at N=200. Also we have set our γ normalization parameter to 0 so the decision would not be biased towards more recall or precision. Also, we did not set a threshold value, we saved only the scores in order to be used in the evaluation step where we calculate the separability of the scores using standard Mann Whitney. The data as mentioned in the previous section 4.4.1 is sorted chronologically. 10 fold cross validation is used (the data is divided into 10 parts (folds), we train on 9 parts and test on the 10th, repeating 10 times while changing the testing part every time). For every fold, the training and testing tweets were sequentially given to the model while reserving their chronological order. This simulates a real life usage situation where tweets always come in one after the other chronologically. If the tweet is in the training set, the model is modified by either adding it to the positive set if the tweet's category matches that of the set or the negative set if it does not match. If the tweet is in the testing set, the compressibility score is calculated and the score saved. If the positive or negative sets' size exceeds N, the oldest tweet is removed from the set.

4.4.3 Evaluation Setup

After obtaining the tweet scores as described in the previous section 4.4.2, we calculate standard Mann Whiteny statistics for every class in every fold. Mann Whitney works by attempting to find a threshold that best divides two sets of scores such that one of them is greater than some threshold (see Fig 5). This measure is equivalent to the area under the curve (AUC) where the curve here would be the Receiver Operating Characteristic (ROC) curve of precision vs recall. We also measured the standard deviation of the model across different folds to assess the stability of the model.

5. Results and discussion

The resulting AUCs for different categories is shown in Fig
6. As you can see the results are very good showing that the model is very stable with little variance among folds of 10 fold cross validation. These results suggest that the model can in fact be used for real life applications.
We have applied a data compression technique to attempt at categorising tweets both with hashtags and without. As we mentioned in the literature review, we noticed an error in the interpretation of results for the method and its comparison to other methods. We therefore wanted to investigate in order to correctly evaluate the method. Upon comparing our results to the original authors, after doing what we think is a just interpretation of the results, we see that our results are comparable for one
category (#f1), but generally outperform that of the original authors. We have scored an average of more than 0.9 on all 4 classes in both datasets. While the original authors reported variant results of 0.7, 0.2 (correct interpretation 0.8), 0.6 and 0.4. However our parameters were different from those of the original authors. Unfortunately, it is very difficult to reproduce their results because of the dataset difference. However, according to our findings, the method has shown great results and large potential for further investigation and research regarding the tuning effects of different parameters and further extensions. Even with our preliminary results, we believe that our study has been able to answer two of our three questions and provided basis for further research, which could lead to answering the third question through a simple extension to the model.
The first question was answered positively since our method has been able to categorise the data without hashtags with both great precision and recall. The second question is also answered by our results since a straightforward application would be to categorise the tweet and therefore suggest hashtags relevant to the output category. The third question can be investigated in the future using an extension to the current method, where instead of having the model for a category, it would be for a person, where the positive tweets are those, which belong to the person, and negative tweets are those, which belong to anyone else. The method shows great potential because not only does it allow for a consistent solution that can work on any twitter account that writes in any given language, we believe it could easily be able to also perform with good results on accounts which uses more than one language, not only in different tweets, but also more than one language in the same tweet. Compared to the other methods in the field, this method has proved to be simple, extensible, and effective. It does not suffer from a lot of the problems which other methods do, such as the need to pre-process datasets to clean-up misspellings and uncommon abbreviations, instead it even leveraged such abbreviations as unique features of the category, and can work on any language seamlessly without any modification or need for any semantic or grammatical definitions from any professionals.

6. Evaluation

There are challenges facing the use of this method in real world applications. Among these, we would like to highlight the need for having a separate training model per category. While this might not be a real limitation, it can however prove to be less scalable than some other methods which rely on semantics because in order to categorise a tweet, it would need to be compared to many models, and therefore usage at the current status would be time consuming, the algorithm needs further performance enhancement by making an extension to the logic of the compression function in order to use it in an online (constantly changing) manner with minimal costs.
Even though our results for classifying tweets without hashtags showed great performance, our dataset was made of tweets with hashtags and then the hashtags were removed from them. It is possible that when users do not intend to use hashtags in their tweets, they write in a different manner, we were unable to investigate this because in order to investigate the difference between both cases, we would need to gather a dataset of tweets about a topic without hashtags, and since the twitter streaming API is based on search queries, obtaining data using keywords would not necessarily be sufficient because it could imply writing patterns specific to context in which those keywords are used, and hashtag are considered keywords. Therefore, something more automated would be required, however, if there is a way to collect tweets about a topic that do not have hashtags, well, that is one of our research questions in the first place (it's a paradox just like the old perception of how came first, the egg or the chicken). Therefore, the only way to collect the tweet dataset would be manual, and that would require and cost a lot of time and planning and is out of the scope of this study.

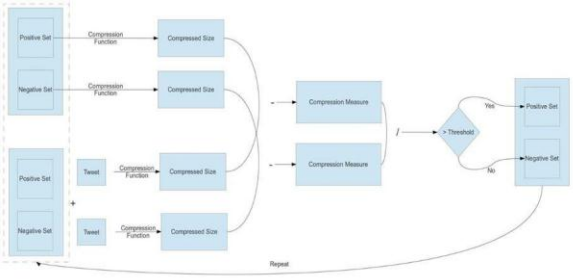


Fig. 4: Flowchart of the algorithm procedure

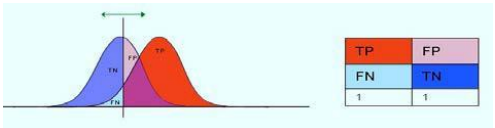


Fig. 5: Mann Whitney attempts to find a threshold that separates two sets of scores by iteratively moving the threshold and calculating true and false positive fractions



Fig. 6: Area Under the Curve for scores of each category, error bars shown in red resemble the standard deviation of the scores of the folds. First column is the AUC for the dataset with hashtags. Second column is for the dataset without hashtags.

We have claimed that the algorithm could detect categories for bilingual tweets. However, we have not tested this claim even though it does not require any implementation modification. But, a rather quick exploration showed that dataset gathering will be troublesome because twitter does not keep track of bilingual tweets and therefore there is no direct or automated method of obtaining them and would possibly require a customised solution that involves a lot of manual steps to find such tweets, keeping in mind that they are relatively scare which put a bigger load on the cost overhead for solving this problem.
We have not explored the effect of different parameters on the model. Our results were very satisfactory that we decided our research questions were answered and did not need further investigation. However, with dataset difference, we are unaware if the model might have some need for tuning.

7. Conclusion

In this paper, we presented a method for categorizing tweets with and without hashtags. Our results show that the method is very effective and easily extensible at categorizing tweets. The method is also flexible to the extent that it can categorise tweets from any language without any need for language specific semantic rules or professional annotation. We answered two research questions that suggested possible applications to better twitter search and suggesting content to users. For future research, we would like to investigate with more categories, and answer our third research question by extending the model to detect persons instead of categories. We believe this model has shown very promising performance and is worth further research.

References

[1] P. Koniusz, F. Yan, P. Gosselin, and K. Mikolajczyk, (2017). Higher-Order Occurrence Pooling for Bags-of-Words: Visual Concept Detection, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 39, No. 2, Feb. 2017.
[2] N. M. Ali, S. W. Jun, M. Karis, M. Ghazaly, M Aras, (2016). Object Classification and Recognition using Bag-of-Words (BoW) Model, 2016 IEEE 12th International Colloquium on Signal Processing & its Applications (CSPA2016), 4 - 6 March 2016, Melaka, Malaysia.
[3] J. Albadarneh, B. Talafha, M. Al-Ayyoub, B. Zaqaibeh, M. Al-Smadi, Y. Jararweh and E. Benkhelifa, (2015). Using Big Data Analytics For Authorship Authentication of Arabic Tweets, 2015 IEEE/ACM 8th International Conference on Utility and Cloud Computing.
[4] S. Katsumata, E. Motohashi, A. Nishimoto, E. Toyosawa, (2016). Website Classification Using Latent Dirichlet Allocation and its Application for Internet Advertising, 2016 IEEE 16th International Conference on Data Mining Workshops.
[5] Y. Chen, and S. Li, (2016). Using Latent Dirichlet Allocation to Improve Text Classification Performance of Support Vector Machine, 2016 IEEE Congress on Evolutionary Computation (CEC).
[6] Ramos-Soto, M. Lama, B. Vazquez-Barreiros, A. Bugarin, M. Mucientes, S. Barro , (2015). Towards Textual Reporting in Learning Analytics Dashboards, 2015 IEEE 15th International Conference on Advanced Learning Technologies.
[7] R. Kilany, R. Ammar, S. Rajasekaran, (2016). A Correlation-Based Algorithm for Classifying Technical Articles, 2011 IEEE International Symposium on Signal Processing and Information Technology (ISSPIT), pp. 50-53.
[8] S. Yuan, X. Wu and Y. Xiang , (2016). Incorporating Pre-Training in Long Short-Term Memory Networks for Tweets Classification, 2016 IEEE 16th International Conference on Data Mining, pp. 1329- 1334.
[9] Önal , A. Ertugrul, (2014). Effect of Using Regression in Sentiment Analysis, 2014 IEEE 22nd Signal Processing and Communications Applications Conference (SIU 2014), pp. 1822-1825.
[10] Y. Li, Y. Zhang, C. Wang, H. Xie, G. Chen, and X. Gao, (2011). Bag-of Features Based Medical Image Retrieval via Multiple Assignment and Visual Words Weighting, IEEE Transactions on Medical Imaging ,Vol. 30, No. 11, Nov. 2011, pp. 1996 – 2011.
[11] H. Bosch, D. Thom, F. Heimerl, E. Puttmann, S. Koch, R. Kruger, M. Worner, and T. Ertl, (2013). ScatterBlogs2: Real-Time Monitoring of Microblog Messages through User-Guided Filtering, IEEE Transactions on Visualization and Computer Graphics, Vol. 19, No. 12, pp. 2022- 2031.
[12] Schmitt, D. Zellhofer, (2012). Condition Learning from User Preferences, 2012 Sixth International Conference on Research Challenges in Information Science (RCIS), pp. 1 – 11.
[13] Rafea, N. A. Mostafa, (2013). Topic Extraction in Social Media, 2013 International Conference on Collaboration Technologies and Systems (CTS), pp. 94 – 98.
[14] Shoukry, A. Rafea , (2012). Sentence-Level Arabic Sentiment Analysis, 2012 International Conference on Collaboration Technologies and Systems (CTS), pp. 546 – 550.
[15] M. Alshawabkeh, J. A. Aslam, J.r Dy and D. Kaeli, (2011). Feature Selection Metric Using AUC Margin for Small Samples and Imbalanced Data Classification Problems, 2011 10th International Conference on Machine Learning and Applications, pp. 145-150.