2016 8th International Conference on Computational Intelligence and Communication Networks

Sentiment Analysis of Online Communities using Swarm Intelligence Algorithms

Lavika Goel
Dept. of Computer Science and Information Systems
BITS Pilani, Pilani campus
lavika.goel@pilani.bits-pilani.ac.in
goel.lavika@gmail.com

Anurag Prakash
Computer Science
BITS Pilani, Pilani campus
f2013061@pilani.bits-pilani.ac.in

***Abstract*— People today spend a lot of their time on the internet and a majority of which is spent on surfing various social networks like Facebook, twitter, Instagram etc. So much time is spent online on these social hubs that our opinions, about almost everything seems to be influenced by the larger opinion formed up in these social networks. In this project, opinion mining due to social swarming is discovered using revolutionary algorithms of swarm intelligence. The build-up of sentiment in a conversation about a product or event is studied here as it provides an alternative way of analysing the presence of sentiment in online communications. In the end we hope of devising a better solution for understanding sentiments in online communication than existing methods.**

Keywords-Sentiment analysis, Swarm Intelligence, Ant Colony optimization metaheuristic, Artificial Intelligence, Genetic Algorithms

I. INTRODUCTION

Internet today is extremely widespread and is used by over 2 billion people. Weblogs, social networks etc. are what attracts people the most as it allows the people to say their mind. This new and virtual way of interaction has led to people being connected to each other even when they are a million miles away. This burst of social interaction in the information age has led to large formation of strong opinions about events, products, people etc. between members of an online social group. These members without any central source of guidance tend to collaboratively, through simple means of communication and interaction, end up forming up a single opinion. Today, rather than the spread of information through the classical medium of communication like the TV, newspaper etc. we have in social networks a faster and more personal way which is method of choice for the future generation. Understanding the above process in which we study the formation of opinions in social swarms provides us with huge potential in understanding the workings of online communities. In this project we try to understand this using the help of revolutionary algorithms of swarm intelligence as we think it will be the best and closest way to represent the people of online communications as members of a swarm. Each posting on the network has a sentiment involved with it and an outreach of the user who has posted this sentiment. This information is used to evaluate the build-up of sentiment in the ongoing thread of conversation. After this swarm intelligence algorithm are used. Swarm intelligence is a part of the discipline of Artificial Intelligence. The objective of this discipline is to study the swarming behavior of social insects. The algorithm we have used is called Ant Colony Optimization, which studies the working of ants and how they find food sources, their homes etc.

II. LITERATURE SURVEY

Twitter has been a powerful tool to gauge the sentiment of the mass public about anything. The analysis of its sentiment has been of utmost importance to a lot of previous researchers. They achieved some important milestones. Traditional Machine Learning approaches like Support Vector Machine, Maximum Entropy classification and Naïve Bayes were used by Go et. al. [14] in their paper on sentiment analysis. Their approach used information about emoticons as noisy labels so as to perform distant supervised learning. Special weights are given to these emoticons which help in skewing the emotion conveyed in a tweet towards the positive or negative side.

A Hybrid approach using both Corpus based and dictionary based methods to determine the sentiment of tweets words used by Kumar et. al. [14] in their papers on sentiment analysis. Their work is mostly based on natural language processing of the tweet and finding the semantic orientation of verbs, adverbs etc. they define a score for each tweet based on the location and kind of identifier present in the tweet. the Corpus based method provided the semantic orientation of the tweet whereas the final sentiment was evaluated using a linear equation.

Work done by Duyu Tang et. al. [15] is also relevant to our project. In this paper, the authors propose learning continuous word representations as features for sentiment classification. They find the syntactic context of words and the sentiment information of sentences to better understand the sentiment conveyed by the user through their tweet.

Another approach followed by Ghiassi et. al. [1] used neural networks accounting for the sensitivity in the sentiments of tweets. using this approach there was able to reduce the number of neutral category tweets drastically The importance of emoticons industry search was very profound as the sound that emoticons are very useful compressions of moods by the uses of Twitter. We were able to classify huge amount of tweets through their approach. They also used specialized models such as DAN2 to find messages of interest for better brand building.

Apoorv Agarwal et. al. [16] through their paper investigated two models, namely a tree kernel and a feature based one to show that both outperform a previously proposed state-of-the-art unigram based model for both 2-way and 3-way classification. Their feature analysis revealed

978-1-5090-1144-5/16 $31.00 © 2016 IEEE
DOI 10.1109/CICN.2016.71

that those features are the most useful which combine the prior polarity of words and their parts-of-speech tags.

Yet another sentiment analysis approach utilized by martinez-camara et. al. [7] compared all the traditional methods of sentiment analysis achieved in the past ranging from papers of Go et. al. [8] in 2009 to Jungherr et. al. [9] in 2012. It contrasts the pros and cons of each approach followed for sentiment analysis done in last few years and provides a good stepping stone for further work in the area.

Work done by Stylios et. al. [6] to compare the opinion mining Dan between a simple decision tree algorithm and the PSO algorithm showed the drastic improvement that can be seen when revolutionary algorithm is used instead of a traditional algorithm like decision trees. there papers were important to signify the importance of using Swarm algorithms instead of traditional ones for sentiment analysis.

One more Revolutionary approach followed by Basari et. al. [2] use the hybrid method of support vector machines and particle Swarm Optimization. it was revolutionary as both of these together had not been used before. their method used the simple support vector machine model wherein the Optimization for the kernel was done using the particle Swarm Optimization Technique. Their use of SVM instead of a nature based algorithm as used in many other papers was the ability of SVM to resolve the issue of over-fitting and the outstanding generalization capability it brings with it. Where is search has also done sentiment classification using the n-grams method and feature weighting. A better accuracy than the standard SVM model was achieved by them.

A basic nature inspired theory of learning was developed by Banerjee et. al. [4], where in they used a method called the CHI statistical measure and also ACO. They were able to accurately model and predict the future behavior of a large population. there proposed algorithm outperformed the CHI methods that were already in place. their emphasis was more on the Q-learning model. A similar approach was taken by Kaiser et. al. [5] In their work where they used a simple implementation of ant Colony Optimization to understand sentiment build up in blogs on a gaming website. On the other hand, a Bee colony optimization algorithm was used by sumathi et. al. [10] for selecting features in opinion mining. Features selected through the ABC approach of the Bee colony optimization algorithm has shown to improve the accuracy of other algorithms like Naive Bayes. Other relevant research work done in papers was found in Goel et. al. [17] and Mousavidin et. al. [18]. Information about Ant Colony Optimization metaheuristic and its practical application was gathered from books and papers by M. Dorigo and other authors [23] [24] [25].

III. METHODOLOGY

In the following section we discuss the methodology discussed which includes data discussion, algorithm used.

A. About data

Data collected from twitter is in the form of list JSONS which have various attributes amongst which we have all the information about the post like the number of retweets the up votes etc. It also contains the information of the user like user id and the number of followers they have etc. Finally, the posting they've made are also used. Equal weight is given to all three of these values.

Twitter data has the following format:

'User_id', 'id_str', 'created_at', 'favourite_count', 'retweet_count', 'followers_count', 'text' the value and definition of all these can be found at the information page of twitter's tweepy API.

Data from Reddit was collected using Praw API. Through this API text of a huge amount of threads from various genres are chosen according to the no of comments and the depth in replies to comments. The data in this is simple plain text and since there is no limit in length of post and paragraph usage, it was difficult to perform sentiment analysis on this without the use of natural language techniques. Reddit data contains only a string of text pertaining to the post posted by the user. No user information is along with this data.

B. Pre-processing

For things like no of followers and no of up votes, we use feature scaling methods mainly range normalization, in which we give the attributes a value between zero and one where the maximum gets a value of 1 and the minimum get a value of 0 and the rest get a value of real value – base value / max value – base value.

For the posting we do many things. Firstly, we only extract the characters which are part of the ASCII table as non ASCII characters do not matter for our problem. Even emoticons are removed. Then we tokenize the posting into different words. Links, citations etc. are also removed as they do not convey sentiment. Words are also stemmed to their root words so as to make it easier to classify them as conveying positive or negative sentiment. Also stop words, i.e. words which are very common and do not convey much sentiment on their own are also removed from list of words. Term frequency methods could also have been used but it will be only used in future iterations.

C. Algorithm

The dataset is divided into 10 different subsets. Nine of these are used for training and the last one is used for testing. The method used for this division is a simple one in which every tenth record is added to a 'subset'. Namely in the first iteration the first, eleventh, twenty-first and so on records are used to update, then in the second iteration the second, twelfth, twenty-second and so on records are used. For testing the tenth, twentieth, thirtieth etc. records are chosen.

During training two pairs of pheromone and heuristic arrays are constantly updated according to their previous value and the predictions. These arrays are namely positive pheromones, negative pheromones, positive heuristics and negative heuristics. They are mainly used to keep track of the development of the ant colony system for this experiment and also used to help the ant make the right decision. These arrays are of the same length and have been initialized with a single value which corresponds to the initial value. During testing these arrays are not updated but just used to prediction the path taken and accordingly checked. Error are

checked accordingly when the prediction by the ant based on the calculated values and the sentiment of the posting evaluated by the simple natural language techniques do not match.

The algorithm uses a simple double bridge – 1 denoting positive sentiment and the other denoting negative sentiment. This is brought into code in the form of two arrays to denote them. Arrays are used just to keep track of previous values and this could have been easily recognized through a simple variable. A constant amount of pheromone is dropped on the path taken by the next user (ant) in the conversation. For a post containing positive sentiment the pheromone dropped is on the positive trail and similarly for a post containing negative pheromone the pheromone is dropped on a negative trail.

The path predicted by our algorithm uses the current amount of pheromone on the paths and a heuristic function. The heuristic function can also be called a weight function, as it used as one in this experiment. This analogy is valid as the heuristic simply gives a weight to all the paths (namely two in this case) to account for the previous bias in choosing this path. This is done to prevent a few posts of the opposite sentiment post to lead to bias which does not follow a general sentiment conveyed in the whole conversation. The heuristic function uses the sentiment of the posting by the user, the number of likes and comments it has got and the outreach of the user (for example the number of followers she has got on twitter etc.) Heuristic values are updated according the similarity between the prediction and the actual values. For example, the heuristic (weight) for positive pheromone is increased if the ant's prediction and the sentiment conveyed by our simple natural language techniques are both positive. The heuristic (weight) is decreased on the other hand for the case when a positive path is chosen by the ant but the sentiment of the post evaluated by our simple natural language techniques is negative. The heuristic values are not updated when there is a neutral sentiment in the post so as to prevent erosion of weights if a few neutral posts come together. It also does not seem logical to penalize a path when the other path is not chosen because in that case both paths will have to be penalized.

Also evaporation (of opinion) takes place to further emphasize the path preferred (positive/negative) by the users in our conversation. This is done in accordance to human behavior. People tend to forget the old point of views and sentiments and tend to follow the recent trend. This evaporation of pheromones leads to lesser emphasis on the sentiments conveyed in posts which happened a long time back. Values are reset for the paths if pheromone levels on both fall below a threshold. This is done to prevent working with low values. This also corresponds to the fact that ants take random paths when the pheromone amounts on the different trails drops below a value than they can detect. It also mimics human behavior wherein the users tend to make up their own opinion when the crowd does not strongly take a stand towards one side.

The paths which have been trained are used to evaluate the learning of the algorithm. The ants make the prediction according to the weights (heuristic) and then the sentiment is evaluated for the post. Whenever a prediction does not match the sentiment the error value is incremented. This is only done in the testing phase where the tenth 'subset' of the data is used.

Figure 1 given below provides the above written algorithm in the form of a Pseudo Code.

After that Figure 2 shows the Pseudo Code is written in a step by step manner in a bulleted format.

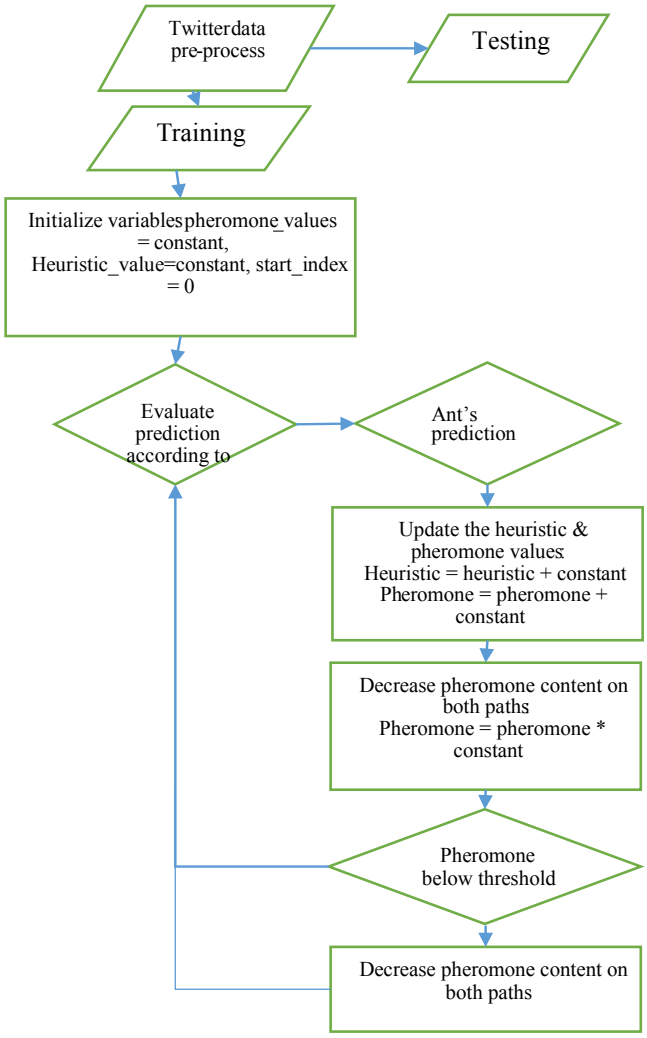

Figure 1: Pseudo Code as depicted in the form of a simple flow chart

Pre-processing: tokenization, word stemming (removal of prefixes/suffixes etc.), term-frequency-inverse document frequency, duplicate word removal, acronyms, emoticons, root words etc. 1. Initialization:

Create arrays for pheromone values, heuristics and ant's decision

Values for all the parameters required like pheromone update etc. $p^p = p^n$ = constant

$h^p = h^n$ = constant

i = 0

2. Looping over the training part of the conversation list. /* TRAINING */

Ant's decision = sign of difference of pheromone * heuristic.

d_i = sign $((h_{ip} * p_{ip}) - (h_{in} * p_{in}))$

Sentiment of posting evaluated using NLP techniques.

If post is carrying sentiment

Update the heuristic values and the pheromone amounts appropriately. Add if prediction matches decision and subtract if vice versa.

$h_{pred} = h_{pred}$ + constant

$p^{path} = p^{path}$ + constant

Decrease pheromone content on both paths to account for evaporation.

$p_i = p_{i-1}$ * evaporation

If the values of pheromones decrease too much then reset all values.

3. Looping over the testing part of the conversation list. /* TESTING */

Ant's decision = sign of difference of pheromone * heuristic.

Sentiment of posting evaluated using NLP techniques.

If post is carrying sentiment

If the prediction matches the sentiment, then increase the value for correct else increase the value for the incorrect.

Decrease pheromone content on both paths to account for evaporation.

Figure 2: Pseudo code in a step by step format.

IV. RESULTS

This section discusses the results of the experiment pertaining to the two datasets that we have discussed above. It talks about how the data was collected. What are the results it gives and the possible reasons for these results.

A. Data collection

Twitter data used in this experiment consisted of data manually harvested directly from twitter using the python wrapper of their twitter API. The data was collected on 14th February 2016 and contains posts on Valentine's Day (all of them contained the word '#ValentinesDay' in them. It contains about 20000 records of which 9/10th were used for testing and 1/10th was used for training.

Reddit data used in this experiment consisted of data manually harvested from Reddit using the Praw API available on Github. It was collected over a period of 2 days in which various threads pertaining to different genres were collected. Data from each thread was restricted to at most 50 comments with each comment having 20 replies. Here only few threads were used as there was no correlation between them. Also 9/10th of the data was used for training and 1/10th was used for testing.

B. Discussion and parameters

There were many parameters in this algorithm which we had to choose to get the best values. Some testing was done to arrive at a good set of values for these parameters to suit the dataset. They are

- Pheromone initial = 0.01
- Pheromone update = 0.1
- Heuristic initial = 0.01
- Heuristic update = 0.1
- Evaporation rate = 0.5

Using these values, the algorithm was run for training and testing parts.

For the twitter dataset as mentioned above in data collection section 1998 records were selected for testing after the remaining records were used in the training part to get a list of the required values to be checked for the build-up of opinion. Of these 1998, 1799 records were correctly predicted by the algorithm and 199 were incorrectly predicted. This gives us an accuracy of 90.04%.

For the Reddit dataset as mentioned above in data collection section 289 records were selected for testing after the remaining records were used in the training part. 210 records resulted in correct sentiment prediction while 79resulted incorrect prediction leading to an accuracy of 72.66%. The accuracy is lower for the Reddit dataset because it is a smaller dataset and also there are vast differences in lengths of the various posts. It is possible to improve upon this by use of robust natural language techniques.

Below is a table provided showcasing the results of this experiment.

Table 1: Shows the results of the experiment.

Dataset	Correct predictions	Incorrect predictions	Accuracy
Twitter data	1799	199	90.04%
Reddit data	210	79	72.66%

Below graphs show buildup of positive heuristic and negative heuristic in the given experiment.

As we can see from the graphs below that the positive heuristic values take a slight dip initially but jump quickly afterwards and stay high. On the other hand, the graph of negative heuristic buildup stays constant after a few bumps in the start. This showcase that initially a negative opinion

was being formed but as soon as a lot of positive posts came the path for the positive heuristic took a bump and so most of the later predictions by the ant were for the positive path. Since the ant did not predict a negative sentiment after the huge influx of positive posts the negative heuristic value does not change much. This showcases the high bias towards the sentiment in the initial buildup. A long penalty causing number of posts are required to invert the graph if the initial sentiment is strong for one particular sentiment.

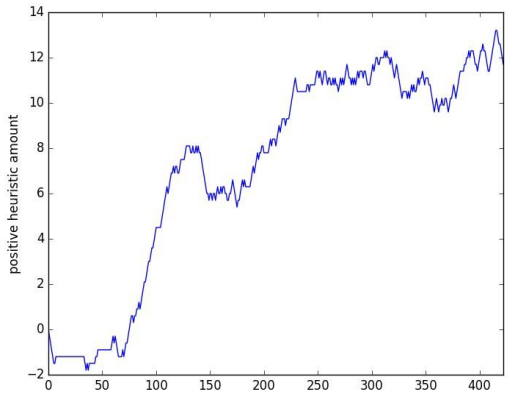

Figure 3: Positive heuristic for first few posts

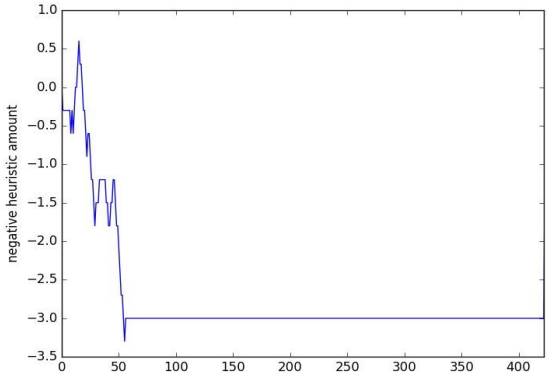

Figure 4: Negative heuristic for first few posts

V. CONCLUSION AND FUTURE SCOPE

Use of swarm intelligence, in particular Ant Colony Optimization, is a new and revolutionary method for analysis of the sentiment of online communities. The correlation between the ants of the ant colony and the users of the online social medium is quite good and thus this results in a good result when the two of them are combined. The non-centralized buildup of sentiment in online communities and the forgetfulness of the content and thus sentiment carried in posts a long time ago brings parallels between the autonomous behavior of the ants and eventual evaporation of pheromones dropped by the ants over time.

This algorithm performs better and faster than traditional algorithms used for sentiment analysis partly because it mimics human behavior. Our sentiment on various topics a lot of times is based on the opinion of others before us. Just like ants follow a path taken by previous ants believing that this path is the best path, we tend to go along with the stronger sentiment. The weakness of our algorithm lies in the fact that the stronger the sentiment towards one side in the beginning, the tougher it is to change the opinion about the topic. This algorithm does not perform well when sentiment changes quickly and drastically like in group chats. But since the evolution of sentiment in social sites like Facebook or twitter, where entities are not in direct contact with one another, takes place slowly, it is a good algorithm for deciphering sentiment for them. This contrast is brought about in the relatively low accuracy for our Reddit data. Since Reddit is a site in which threads are made pertaining to a question, inference or some posting, the users responding to it are near enough to direct contact with each other. Here sentiment in the thread can pivot quickly and therefore lead to poorer results.

This project provides a peak into new and evolutionary methods which perform better than traditional classification methods used for sentiment analysis. Though it uses a basic model of an ant colony and only a few parameters to tweak, it performs better than most traditional methods for sentiment analysis used in papers like Pak and Paroubek [11], Bifet and Frank [12], Zhang et. al. [13], Wang et. al. [19], Wilson et. al. [21] etc.

In the future we hope to incorporate other techniques like natural language processing, sentences-word correlation, use of term frequency inverse document etc. to improve our results. We hope that with the use of this we would be able to drastically improve on our current results for the Reddit dataset.

REFERENCES

[1] Ghiassi, M., J. Skinner, and D. Zimbra. "Twitter brand sentiment analysis: A hybrid system using n-gram analysis and dynamic artificial neural network." Expert Systems with applications 40.16 (2013): 6266-6282.

[2] Basari, Abd Samad Hasan, et al. "Opinion mining of movie review using hybrid method of support vector machine and particle swarm optimization, "Procedia Engineering 53 (2013): 453-462.

[3] Kumar, Akshi, and Teeja Mary Sebastian. "Sentiment analysis on twitter, "IJCSI International Journal of Computer Science Issues 9.4 (2012): 372-373.

[4] Banerjee, Soumya, and Nitin Agarwal. "Analyzing collective behavior from blogs using swarm intelligence." Knowledge and Information Systems 33.3 (2012): 523-547.

[5] Kaiser, Carolin, Johannes Kröckel, and Freimut Bodendorf. "Swarm intelligence for analyzing opinions in online communities." System Sciences (HICSS), 2010 43rd Hawaii International Conference on. IEEE, 2010.

[6] Stylios, George, Christos D. Katsis, and Dimitris Christodoulakis. "Using Bio-inspired intelligence for Web Opinion Mining." International Journal of Computer Applications 87.5 (2014).

[7] Martínez-Cámara, Eugenio, et al. "Sentiment analysis in Twitter." Natural Language Engineering 20.01 (2014): 1-28.

[8] Go, Alec, Richa Bhayani, and Lei Huang. "Twitter sentiment classification using distant supervision." CS224N Project Report, Stanford 1 (2009): 12.

[9] Jungherr, Andreas, Pascal Jürgens, and Harald Schoen. "Why the pirate party won the German election of 2009 or the trouble with predictions: A response to tumasjan, a., sprenger, to, sander, pg, & welpe, im "predicting elections with twitter: What 140 characters reveal about political sentiment", "Social science computer review 30.2 (2012): 229-234.

[10] Sumathi, T., S. Karthik, and M. Marikannan. "Performance Analysis of Classification Methods for Opinion Mining." International Journal of Innovations in Engineering and Technology (IJIET) Vol 2 (2013): 171-177.

[11] Pak, Alexander, and Patrick Paroubek. "Twitter as a Corpus for Sentiment Analysis and Opinion Mining." LREc. Vol. 10. 2010.

[12] Bifet, Albert, and Eibe Frank. "Sentiment knowledge discovery in twitter streaming data." Discovery Science. Springer Berlin Heidelberg, 2010.

[13] Zhang, L., Ghosh, R., Dekhil, M., Hsu, M., and Liu, B. 2011. Combining lexicon-based and learning-based methods for Twitter sentiment analysis. Technical Report HPL-2011-89.

[14] Go, Alec, Richa Bhayani, and Lei Huang. "Twitter sentiment classification using distant supervision." CS224N Project Report, Stanford 1 (2009): 12.

[15] Tang, Duyu, et al. "Learning sentiment-specific word embedding for twitter sentiment classification." Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics. Vol. 1. 2014.

[16] Agarwal, Apoorv, et al. "Sentiment analysis of twitter data." Proceedings of the Workshop on Languages in Social Media. Association for Computational Linguistics, 2011.

[17] Goel, Lavika, Daya Gupta, and V. K. Panchal. "Hybrid bioinspired techniques for land cover feature extraction: A remote sensing perspective. "Applied Soft Computing 12.2 (2012): 832849.

[18] Mousavidin, Elham, and Lavika Goel. "A life cycle model of virtual communities." System Sciences, 2009. HICSS'09. 42nd Hawaii International Conference on. IEEE, 2009.

[19] Wang, Xiaolong, et al. "Topic sentiment analysis in twitter: a graph-based hashtag sentiment classification approach." Proceedings of the 20th ACM international conference on Information and knowledge management. ACM, 2011.

[20] Pang, Bo, and Lillian Lee. "Opinion mining and sentiment analysis. "Foundations and trends in information retrieval 2.1-2 (2008): 1-135.

[21] Wilson, Theresa, Janyce Wiebe, and Paul Hoffmann. "Recognizing contextual polarity in phrase-level sentiment analysis." Proceedings of the conference on human language technology and empirical methods in natural language processing. Association for Computational Linguistics, 2005.

[22] Liu, Bing. "Sentiment analysis and opinion mining." Synthesis lectures on human language technologies 5.1 (2012): 1-167.

[23] Dorigo, Marco, Mauro Birattari, and Thomas Stützle. "Ant colony optimization." Computational Intelligence Magazine, IEEE 1.4 (2006): 28-39.

[24] Dorigo, Marco. "Ant colony optimization." Scholarpedia 2.3 (2007): 1461.

[25] Dorigo, Marco, et al., eds. Ant Colony Optimization and Swarm Intelligence: 6th International Conference, ANTS 2008, Brussels, Belgium, September 22-24, 2008, Proceedings. Vol. 5217. Springer, 2008.