Published by :
http://www.ijert.org

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 8 Issue 11, November-2019

Detection Impostures of Cloud Can be Calculated by Heuristic Function

Imposture Calculation by Heuristic Function

Dr. Cherukuri Ravindranath
ENTC
Christ (Deemed to be University)

Prof. Sunil Suresh Ghadge
It
Trinity College of Engineering and Research, Pune

***Abstract*— Online reviews play a crucial and important role in today's electronic commerce. The opinion information can guide the purchasing behavior of any people or customer that can write any opinion text, this can let the people give undeserving positive opinions or review to some target objects in order to promote the objects, or to give unjust or malicious negative opinions to some other objects in order to damage their reputations. Products with the large percentage of reviews mostly tend to attract new customer rather than other related software. Due to reason of fame, imposture, fraud, or to make profit some people tend to write deceptive or spam reviews to deliberately attract or mislead the consumer. This gives unfair positive or negative result, and such kind of imposture is known as spammers or opinion spammers. Today sentiment analysis and opinion mining and becomes a popular important and preferred task. So here we need to identify and filter out the review spam to provide the real and trustful review services.**

Keywords*— *(1)Imposture (2)Heuristic function (3) Improved A* algorithm (4) Square grid (5) Heuristic value (6Ssecurity (7) Fuzzy logic

1. INTRODUCTION

Recent years we have witnessed the rapid growth of smart mobile devices with numerous mobile software on cloud store and the number goes above 1.6 million to make our life simpler. With the development of the Internet, people are more likely to express their views and opinions on the Web.
The developer of the mobile software on the play store tends to explore the ranking of app by writing good reviews and performance as much high as possible. Due to growing trade of software, most of the people rely on online apps or product and purchase them on the basis of rank, reviews and rating given to the software.

Hence there is necessity to detect spam reviews of the spammers we use Author Spam city Model (ASM). The opinion spam identification task has great impacts on today growing industrial and academia communities. The opinion information also benefits the business organizations.
This ASM model is an unsupervised model based on Bayesian theorem. It helps to detect spam reviews and genuine reviews. The next step is analysis on genuine reviews by NLP algorithm that will give final result that reviews for particular app is positive (good) or negative (bad).

Text Mining is an immensely popular software of NLP that aims at extracting patterns and structured information from textual content. Due to its importance, many frameworks have been developed to facilitate the development of text mining software. The rating results will be shown graphically on basis of rates or stars given to software by the users. And also, the ranking position will be calculated to see the difference between all three parameters. Finally, we see the observations and underlying principles applied for spam detection. And at the end the result will be showed in graphical or statistical format by aggregating results of all the parameters with the help of fuzzy key word and Heuristic function to form cluster, also utilized to find nearest path to reach different cluster.

To do analysis on the parameters described above we first connect to google cloud and then fetch the data of the software for further processing of data or information. The additional part in this software is accuracy will be provided. If the user is demanding for some specific feature in software, then the exact result will be shown depending on the demand of user and other related software will be secondary part of display.
In addition to this software, we also provide security that is very important issue for the customer.

Security leaks and confidential data disclosure from web and mobile apps are quite common today. Security flaws originate at the development stage and also when software is cloned with extra functions that run in the background and perform malicious actions. Third-party libraries and APIs used in software development may contain malicious code that could steal data stored by the software. Developers need to ensure that only trusted libraries are used in the software.

While executing procedures such as logging, sensitive data should not appear in the logs. With the increasing number of technologically rich system software hitting the market, system have become the new target for hackers.

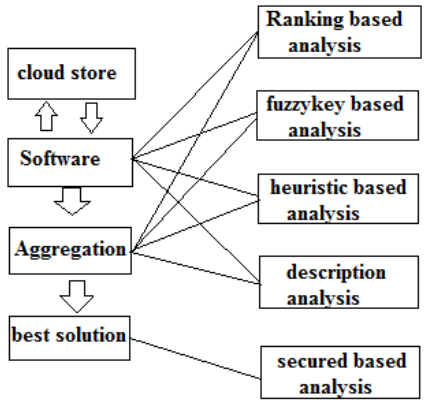
There is tremendous need for security that will protect personal data or private information from potential threats, or from leveraging the privacy of customer that can develop as attack vector. For providing security we will provide installation of the software. The access will be given if the user is ready to give access or to share information or else data would be blocked. Further if app & software is not getting installed due to inaccessibility reason then user will be given the facility it gives bogus or fake information to the system.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 8 Issue 11, November-2019

2. LITERATURE REVIEW

We are providing an android software for users so as to give security and recommend alternatives for best software. The first step we choose is to do analysis on parameters as review, rank, rating.

II System Architecture



At the first step the user will pass the name of software which he wants to check whether it is good, recommended, featured software. Then it will connect to cloud and with the help of key it will fetch information as version, size, category, ranking, review, rating and previous history related to software.

Now this information or data of software must be stored, so that the further analysis would be carried out, hence we use cloud. And now the data analysis will be performed.

At first the analysis will be done on reviews of software, here the first step would be to detect genuine reviewers and spam reviewers. This work is carried out by Author Spam city Model (ASM) model which works in the Bayesian setting. It is observed by some characteristics of abnormal behaviors which are likely to be linked with spamming.

After we differentiate the genuine reviewers and spam reviewers, we now need to do analysis on genuine algorithm with help of Natural Language Processing algorithm. The next analysis is done on rating as if software rating is below three then it will be considered as negative result and if software rating is more than three then it will be considered as positive result. Ranking of software is calculated on result of review and rating. If the result of reviews and rating is best then the software rank must be in first top list and if not i.e if the reviews and rating are not good then they must list down the line of top listed or popular software. In description analysis we are sorting our keyword of features of software rather than a user reading the description in paragraph. After all this analysis all positive and negative result are grouped and make group by Heuristic function in this positive result and negative result can be created. We also provide security to the users for the protection of private information, here we give access to the user to set if user wants to block or allow access to its personal information as contacts, gallery, location, device ID (IMEI/MEID/ESN), subscriber ID (IMSI), SIM serial (ICCID), phone and mailbox number, incoming and outgoing call number, GPS location, network location, list of accounts (including your Google e-mail address). And if the software is not installing due to blockage then we can provide bogus or fake information to the system.

3. ***Fuzzy key based analysis:*** As we know that rating manipulation is an important perspective for detecting rating imposture. It's obvious that when software is published it is rated by user. Indeed, user ratings. The higher ratings with numerous downloads on leaderboard it attack number of users. Hence in rating analysis, the manipulation of rates can be showed in result with help of graph. As we observe that normal always receives similar type of average rating and that with other fraudulent apps receives higher ratings in some time period. This can be easily seen on graph generated with R language. Within analysis we also perform other way of calculating result of rates. As if app's rating is below 3 then it will be considered as negative result and if software rating is more than three then it will be considered as positive depending on previous history.

we address the problem of supporting efficient yet privacy-preserving fuzzy keyword search services over encrypted data. Specifically, we have the following goals: i) to explore new mechanism for constructing storage efficient fuzzy keyword sets; ii) to design efficient and effective fuzzy search scheme based on the constructed fuzzy keyword sets; iii) to validate the security of the proposed scheme.

D. Preliminaries

There are several methods to quantitatively measure the string similarity. In this paper, we resort to the well-studied edit distance for our purpose. The edit providing an overview of how fuzzy search scheme works over encrypted data. Assume Π=(Setup(1λ), Enc(sk,), Dec(sk,)) is a symmetric encryption scheme, where sk is a secret key, Setup(1λ) is the setup algorithm with security parameter λ, Enc (sk, λ1) and Dec (sk, λ2) are the encryption and decryption algorithms, respectively. Let Tw_i denote a trapdoor of keyword wi. Trapdoors of the keywords can be realized by applying a one-way function f, which is similar as Given a keyword wi and a secret key sk, we can compute the trapdoor of wi as Tw_i= f (sk, wi). We begin by constructing the fuzzy keyword set Swi, d for each keyword wi∈W (1≤ i≤ p) with edit distance d. The intuitive way to construct the fuzzy keyword set of wi is to enumerate all possible words w_i' that satisfy the similarity criteria ed (w_i, w_i')≤ d, that is, all the words with edit distance d from wi are listed. For example, the following is the listing variants after a substitution operation on the first character of keyword: -

CASTLE: {AASTLE, BASTLE, DASTLE, ZASTLE}.

Based on the resulted fuzzy keyword sets, the fuzzy search over encrypted data is conducted as follows:

1. To build an index for wi, the data owner computes trapdoors Tw'=f (sk, w'_i) for each w'_i∈ Swi, d with a secret distance ed (w_1, w_2) between two words $w_1$and $w_2$is the number of operations required to transform one of them into the other. The three primitive operations are 1) Substitution: changing one character to another in a word; 2) Deletion: deleting one character from a word; 3) Insertion: inserting a single character into a word. Given a keyword w, we let Sw, d denote the set of words w′ satisfying ed (w, w′) ≤ d for a certain integer d. Using edit distance, the definition of fuzzy keyword search can be formulated as follows: Given a

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 8 Issue 11, November-2019

collection of encrypted data files C = (F_1, F_2, . . . , FN) stored in the server, a set of distinct keywords W= {w_1, w_2, ..., wp}with predefined edit distance d, and a searching input (w, k) with edit distance k(k≤d), the execution of fuzzy keyword search returns a set of file IDs whose corresponding data files possibly contain the word w, denoted as F IDw: if w = wi ∈ W , then return F IDw_i; otherwise, if w 6∈ W, then return {F IDw_i}, where ed(w, wi)≤ k. Note that the above definition is based on the assumption that k ≤ d. In fact, d can be different for distinct i key sk shared between data owner and authorized users. The data owner also encrypts FID_wi^{as}Enc (sk, FID_wi^kwi).

The index table {({Tw_i'} w′i∈Swi, d' Enc (sk, FID_wi^kwi))} wi∈W and encrypted data files are outsourced to the server for storage;

2. To search with w, the authorized user computes the trapdoor Tw of w and sends it to the server;

3. Upon receiving the search request Tw, the server compares it with the index table and returns all the possible encrypted file identifiers {Enc (sk, FID_wi^kwi)} according to the fuzzy keyword definition in section III-D. The user decrypts the returned results and retrieves relevant files of interest. This straightforward approach apparently provides fuzzy keyword search over the encrypted files while achieving search privacy using the technique of secure trapdoors. However, this approach has serious efficiency disadvantages. The enumeration method in constructing fuzzy keyword sets would introduce large storage complexities, which greatly affect the usability. Recall that in the definition of edit distance, substitution, deletion and insertion are three kinds of operations in computation of edit distance. The numbers of all similar words of wi satisfying ed (wi, w'_i) ≤ d for d = 1, 2 and keywords and the system will return {F IDwi} satisfying approximately$2k \times 26, 2k^2 \times 26^2$, and$^4 3^k \times 26^3$, ed (w, wi) ≤ min {k, d} if exact match fails.

4. ***Ranking analysis*****:** In this module we calculate the new ranking of mobile software on the genuine reviews. If current rank of mobile software is good and calculated rank is also good then it will consider as positive result. If in both rank the difference is there then it will be considered as negative result.

5. ***Aggregation:*** In aggregation module we use k-means clustering algorithm. This algorithm helps for making clusters of positive result and negative result. Here we combine all the parameters to get final result.

6. ***Description analysis*****:** Here we provide user only the keywords of features rather than descriptions in big paragraphs. In this module we compare features of multiple apps of same category and after comparison we show as result best app in between those apps.

7. ***Secured based analysis*****:** At the installation time of mobile software that asked for access permission of our personal information and when we give permission, they access our personal data like location, mobile id, contacts, SMS, videos, images etc. Many software does not have need of this information they may be misused of our personal data. For that in this module we provide blocking and accessing permission for that particular app of our data. If we block data access permission then that particular mobile app cannot fetch data from mobile.

3. HEURISTIC COST AND HEURISTIC FUNCTION IN A*

The Heuristic cost can be calculated by using Heuristic function. This Heuristic function uses the distance metric used to calculate the movement cost. The choice of distance metric is done according to the requirements. The two most popular distance metrics are: a. Euclidean distance b. Manhattan distance The Euclidean distance between two n-dimensional vectors x and y can be defined as:

$$d(x, y) = \sqrt{\sum n\ i=1\ (xi - yi)}$$

And the Manhattan distance between two n-dimensional vectors x and y can be defined as:

$$d(x, y) = \sum n\ i=1\ |xi - yi|\ |$$

The heuristic cost is estimated by considering that there is no obstacle between the current node and the destination node. The Accuracy and Speed of A* algorithm depend on this heuristic cost. If the heuristic cost is zero then the A* algorithm will work similar to Dijkstra's algorithm, which will give very accurate output but it will be slow in performance. If the heuristic cost is greater than the cost of moving from current node to destination node, then A* will not guarantee to find the shortest path but it will run faster, i.e. the accuracy will be reduced and the speed will be increased. If the Heuristic cost is equal to the cost of moving from current node to destination node then the A* will only follow the correct path/nodes and avoid exploring irrelevant nodes, making it very fast. For the accurate (or shortest path) the Euclidean distance can be used. And for faster path-finding, the Manhattan distance can be used.

4. OBJECTIVE OF THE RESEARCH WORK

Due to this Speed-Accuracy trade-off, the A* have ability to change its behavior according to the requirements. i.e. if path needs to be found in the situations (like video games), where just finding the path in less time is important than optimal path, then the Heuristic cost can be kept greater or equal to the cost of moving from current node to destination node. But, if the path needs to be found in the situations (like rescue operations), where the path should be accurate as well as should be found in less time, then the Heuristic cost should be equal to the cost of moving from current node to destination node. The Speed-Accuracy trade-off of the A* algorithm with respect to Heuristic cost can be given as follow: Notation: d(n) is the cost of moving from current node to destination. As shown in Fig. 1, as the value of heuristic function h(n) increases from 0 to infinity, the speed of the algorithm increases i.e. time required by algorithm decreases and the accuracy of algorithm decreases i.e. Path length computed by algorithm increases. Since, at h(n) = 0, the A* algorithm will act like Dijkstra's algorithm and will provide accurate result but with less speed. And at h(n) > d(n), the A* algorithm will act like Greedy Best-First-Search algorithm and will provide high speed but with less accurate result.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 8 Issue 11, November-2019

5. PROBLEM FORMULATION

Data parser: In this module we will pass the name of android software, then it will connect to the Google play store and fetch information like app version, app size, category, ranking, review, rating and previous history related to android software. Then whole information is stored in Cloud.

Review analysis: In Review analysis module first, we have to find spam review and genuine review from all fetched reviews for that purpose we use Author spam city model (ASM). Under this following feature are considered for separating spam review and genuine review.

a. ***Content Similarity:*** New reviews giving every time is time consuming, spammers are likely to do copy reviews across similar products. Therefore, it captures the content similarity of reviews (using cosine similarity) of the same author. Here we chose the maximum similarity to capture the spamming behavior.

b. ***Maximum Number of Reviews:*** Posting many reviews in a single day also shows an abnormal behavior of spammers. In this we compute the maximum number of reviews in a day for single author.

c. ***Reviewing Burstiness:*** As per study of opinion spammers are usually not longtime members of a site. Genuine reviewers are using their accounts time to time to post reviews. Thus, it is useful to exploit the activity freshness of an account in detecting spamming. By using the activity window, we define reviewing burstiness (difference of first and last review posting dates). If reviews are posted over a long time, it probably indicates normal behavior. However, when all reviews are posted within a very short time period it is likely to show be a spam infliction.

d. ***Ratio of First Reviews:*** Spamming early can impact the initial sales as people rely on the early reviews. Hence, spammers would try to be among the first reviewers for products as this enables them to control the sentiment. For each author we compute the ratio of first reviews to total reviews. First reviews refer to reviews where the author is the first reviewer for the products.

e. ***Duplicate/Near Duplicate Reviews*:** Spammers often post multiple reviews which are duplicate/near-duplicate versions of previous reviews on the same product to boost ratings.

f. ***Extreme Rating:*** The reason is to promote/demote products spammers by giving extreme ratings (1 or 5) on a 5-star rating scale, it shows that the developers intuition is to inflict spam and to place product in top list of play store.

g. ***Rating Deviation*:** Due to review spamming the most impact results on ranking and rating parameters that usually involves wrong projection either in the positive or negative way so as to alter the true sentiment on products. This gives hints that ratings of spammers often deviate or mislead from the average ratings given by other reviewers.

h. ***Early Time Frame:*** Spammers often review their product early to inflict spam as the reviews given in early stage can greatly impact user's sentiment on a product and may buy it. The definition says that if the reviews of the product are given before more than seven months they are no more considered as early or fresh reviews. In other case, if reviews are posted just after launch of the product or software this feature attains a value of 1.

i. ***Rating Abuse:*** This feature captures the improper usage caused by multiple ratings on the same product to make product or app popular. Multiple ratings or reviews on the same mobile software are unusual. This tends to identify the spam city. Indirectly it mostly focuses on the rating dimension rather than content. Rating abuse is defined as similarity of ratings of an author, towards its reviews on app.

After separating genuine review, we perform Natural Language Processing algorithm. Here NLP is a field of artificial intelligence, computer science, and computational linguistics that concentrates on interactions between computers and human natural languages. As such, NLP is related to the area of human-computer interaction. In this algorithm four types of filtration can be done. First filtration is tokenization in this token can be separated out form the whole review statement. Eg: -suppose "This platform is very good" is the review given by the user. from this review each token is separated out like as 'This', 'platform', 'is',' very ','good'. After separating tokens second stop word filtration can be done. In this stop words are found and then it will be removed from the collection of the tokens. Ex. 'is ','are' like words are found from the collection of the tokens and then it will be removed. After that we used stem filtering on the collection of tokens. In this suffix are found. Ex. 'ing', 'tion' and after that those suffixes are removed. After performing all of this filtration fourth filtration is sentiment analysis can be done in this by using special keywords, we found Positive and negative result of that particular review.

6. MOTIVATION / PROBLEM STATEMENT DEFINITION

The Visual Cryptography & Heuristic A* algorithm is a method of encrypting secret image into n number of random shares and distribute to the n number of entities to get the nearest path. The encrypted secret image can be decrypted by stacking the distributed random shares from the n number of entities. Visual Cryptography can also be applied to encrypt the secret information into an image. However, the Visual Cryptography Scheme (VCS) suffers from security because of its software implementation. Since the software model of VCS could run-on general-purpose processor requires more power energy and also leave space for security issues. So, this research is motivated to develop high speed and energy efficient hardware system for secure multimedia software.

7. PROPOSED RESEARCH METHODOLOGY

The data flow of the proposing fuzzy key word and grid computing architecture for secure multimedia transmission is described in figure 1. The architecture uses a well-known Heuristic function and distance metric that is nothing but grid computing scheme to compress the size of the secret shares. Since the embedding process increases the execution time of the entire system and also degrades the quality of the regenerated image, FPGA devices are selected as computing platform. The proposed system provides an integrated

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 8 Issue 11, November-2019

environment to process images and it supports only single image format either .gif or .png. The proposed system consists of three basic operations these are compressed share generation, encoding and decoding. Second is done by path finding and distance metrics. Decoding is done by the stacking of the secret shares at the receiver's end.

The data flow of the cryptographic process is demonstrated and the sequence of steps

1. Select the image as input
2. Create encrypted shares using an appropriate encoding algorithm for intended secret image.
3. Prepare the dictionary for encrypted shares.
4. In dictionary find the word by using key.
5. Get the string and apply the encryption algorithm to compress the data.
6. Then generation halftone shares using description Method.
7. Filter process is applied to the distance metric to share your encrypted key.
8. Filters are used to improve the quality of the reconstructed image to minimize the noises for sharpening the input secret image.

The proposed architecture is then used for processing multimedia data like images, video and audio information.

8. POSSIBLE OUTCOME / RESULT

An energy efficient architecture for system architecture to store and transmit secure multimedia data over wired channel.
Estimation of hardware resources required for the proposed architecture and comparison statement with contemporary architectures in literature.
Estimation of the performance attributes like speed, memory and data handling capacity of the proposed architecture.

9. CONCLUSIONS

As we know that there is rapid growth in mobile devices and the users are using software from cloud store for their convenience or entertainment. As numerically known that there are more than millions of software on cloud store. User relies on the reviews, rating, number of downloads of software and fuzzy key word of algorithm which may be fake so as to make profit. So here we are introducing a software to detect whether the app and software composes imposture or not. We identify evidence as fuzzy keyword, rating and description for detecting imposture. A unique perspective of this approach is that all the evidences of fake will be shown in the result. In this software we also ensure users privacy by adding security component in this software so that there will be no leaking of private or personal data or information of users. Here we give software utility or users to allow or block access of their personal data while installing the software. The features as accuracy of the software and privacy of user's data component are composed in this software.

10. REFERENCES

[1] Zhi Zhou, G. R. Arce and G. Di Crescenzo, "Halftone visual cryptography," in IEEE Transactions on Image Processing, vol. 15, no. 8, pp. 2441-2453, Aug. 2006.
[2] D. Shrestha and S. P. Panday, "Visual cryptography using image pixel transparency with cover image," 2015 9th International Conference on Software, Knowledge, Information Management and Software (SKIMA), Kathmandu, 2015, pp. 1-4.
[3] A. Ross and A. Othman, "Visual Cryptography for Biometric Privacy," in IEEE Transactions on Information Forensics and Security, vol. 6, no. 1, pp. 70-81, March 2011.
[4] Z. Wang and G. R. Arce, "Halftone Visual Cryptography Through Error Diffusion," 2006 International Conference on Image Processing, Atlanta, GA, 2006, pp. 109-112.
[5] Wei-Qi Yan, Duo Jin and M. S. Kankanhalli, "Visual cryptography for print and scan software," 2004 IEEE International Symposium on Circuits and Systems (IEEE Cat. No.04CH37512), 2004, pp. V-572-V-575
[6] Ming Sun Fu and O. C. Au, "Joint visual cryptography and watermarking," 2004 IEEE International Conference on Multimedia and Expo (ICME) (IEEE Cat. No.04TH8763), Taipei, 2004, pp. 975-978 Vol.2.
[7] T. Monoth and A. P. Babu, "Recursive Visual Cryptography Using Random Basis Column Pixel Expansion," 10th International Conference on Information Technology (ICIT 2007), Orissa, 2007, pp. 41-43.
[8] Der-Chyuan Lou, Hao-Kuan Tso, Jiang-Lung Liu, A copyright protection scheme for digital images using visual cryptography technique
Computer Standards & Interfaces, Volume 29, Issue 1, Pages 125-131.
[9] Young-Chang Hou and Pei-Min Chen, "An asymmetric watermarking scheme based on visual cryptography," WCC 2000 - ICSP 2000. 2000 5th International Conference on Signal Processing Proceedings. 16th World Computer Congress 2000, Beijing, 2000, pp. 992-995 vol.2.
[10] F. Liu and C. Wu, "Embedded Extended Visual Cryptography Schemes," in IEEE Transactions on Information Forensics and Security, vol. 6, no. 2, pp. 307-322, June 2011.
[11] R. Youmaran, A. Adler and A. Miri, "An Improved Visual Cryptography Scheme for Secret Hiding," 23rd Biennial Symposium on Communications, 2006, Kigston, Ont., 2006, pp. 340-343.
[12] J. Weir and W. Yan, "Sharing multiple secrets using visual cryptography," 2009 IEEE International Symposium on Circuits and Systems, Taipei, 2009, pp. 509-512.
[13] H. Zhu, H. Cao, E. Chen, H. Xiong, J. Tian, "Exploiting enriched contextual information for mobile app classification", in Proc. 21st ACMInt. Conf. Inform. Knowl. Manage, 2012, pp. 1617-1621.
[14] H. Zhu, E. Che, K. Yu, H. Cao, H. Xiong, J. Tian, "Mining personal context-aware preferences for mobile users", in Proc. IEEE 12th Int. Conf. Data Mining, 2012, pp.1212-1217.
[15] Clifton Phua1, Vincent Lee, Kate Smith, Ross Gayler, "A Comprehensive Survey of Data Mining-based Fraud Detection Research", Australian Research Council, Baycorp Advantage, and Monash University LP0454077, 2008.
[16] Gogu Sandeep, Sachin Malviya, Dheeraj Sapkale, "Data Mining: An Improved Approach for Fraud Detection", WSDM'08, Palo Alto, California, USA. ACM 978-159593-927-9/08/0002,2008.
[17] Geli Fei, Arjun Mukherjee, Bing Liu, Meichun Hsu, Malu Castellanos, Riddhiman Ghosh, "Exploiting Burstiness in Reviews for Review Spammer Detecction", Association for the Advancement of Artificial Intelligence,2013.
[18] Fangtao Li, Minlie Huang, Yi Yang, Xiaoyan Zhu, "Learning to Identify Review Spam", International Joint conference on Artificial Intelligence, 2011.
[19] Sihong Xiw, Guan Wang, Shuyang Lin, Philip S. Yu, "Review Spam Detection via Temporal Pattern Discovery", 2014.
[20] N. Jindal and B. Liu, "Opinion spam and analysis," in Proc. Int. Conf. Web Search Data Mining, 2008, pp.219-230.
[21] Sable Nilesh Popat*, Y. P. Singh," Efficient Research on the Relationship Standard Mining Calculations in Data Mining" in *Journal of Advances in Science and Technology* | *Science & Technology*, Vol. 14, Issue No. 2, September-2017, ISSN 2230-9659.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 8 Issue 11, November-2019

[22] Sable Nilesh Popat*, Y. P. Singh," Analysis and Study on the Classifier Based Data Mining Methods" in *Journal of Advances in Science and Technology* | *Science & Technology*, Vol. 14, Issue No. 2, September-2017, ISSN 2230-9659.

[23] Jinli Xu, Fei Hu, Xiaolei Han, "**Realization of Bidirectional A* Algorithm Based on the Hierarchical Thinking During the Process of Path Planning**" Wuhan Univ. of Technol 2008 International Conference on Computer Science and Software Engineering Year: 2008 | Volume: 1 | Conference Paper | Publisher: IEEE

[24] Ji-Xian Xiao, Fang-Ling Lu, "**An improvement of the shortest path algorithm based on Dijkstra algorithm**" 2010 The 2nd International Conference on Computer and Automation Engineering (ICCAE), Year: 2010 | Volume: 2 | Conference Paper | Publisher: IEEE