A Survey on Trust and Distrust Propagation for Web Pages

Zenith Azim[1], Gypsy Nandi[2]
[1,2]Don Bosco College of Engineering and Technology, Assam Don Bosco University
Airport Road, Azara, Guwahati - 781017, Assam. INDIA.
zenithazim@gmail.com, gypsy.nandi@dbuniversity.ac.in

Abstract:*Search engines are the hub for information retrieval from the web. But due to the web spam, we may not get the desired information from the search engines. The phrase web spam is used for the web pages that are designed to spam the web search results by using some unacceptable tactics. Web spam pages use different techniques to achieve undeserved ranking in the web. Over the last decades researchers are trying to design different techniques to identify the web spam pages so that it does not deteriorate the quality of the search results. In this paper we present a survey on different web spam techniques with underlying principles and algorithms. We have surveyed all the major spam detection techniques and provided a brief discussion on the pros and cons of all the existing techniques. Finally, we summarized the various observations and underlying principles that are applied for spam detection techniques.*

Keywords: TrustRank, Anti-TrustRank, Good-Bad Rank, Spam Detection, Demotion.

1. Introduction

Billions of people are connected through internet every day. Billions and billions of information are being shared or retrieved from the web. So in this situation where there are billions of source of information or we can say billions of sources of answer for a single question, a question can be raised as "what to trust and what not to trust". Trusted or correct information can help a user to take a decision, receive recommendations, etc. In this situation knowing what to trust is very important, but what not to trust is equally important as well.

In websites, web spam can also be referred to as the hyperlinked pages on web that are indented to mislead the search engine results. The spam has been identified as one of the most important challenges faced by the search engines [1]. As for example, a pornography site can spam the web by adding many different keywords to its page which are not adult in nature and which can lead the users to surf those spam pages that were actually meant for searching for some other topics. Spammers make those keywords invisible to human eyes by using different color schemes. Another regular technique used by the spammers is creation of many sites and pointing to a single spam or target site; this will result into high ranking of the spam site in the search engine result, as many of search engines rank the site based on the incoming links to the site. Some spammers also try to spam the web by creating URLs having numerous dots(.), dashes (/) and some other symbols and also by using some words repeatedly

used in the URL which may be search by the users as queries. As can be seen, there are many ways of spamming the web which we can be split into different categories like content spam, link spam, and so on. We will go through the details of each category in the later section of this paper.

Propagation of trust is easy to achieve because propagation of trust is transitive. Suppose Alia is a good friend of Ben and Ben is a good friend of Ricky [2]. Then we can say that Alia trust Ben and Ben trust Ricky. From this we can make a decision that Ricky can trust Alia, which is shown in Figure1.

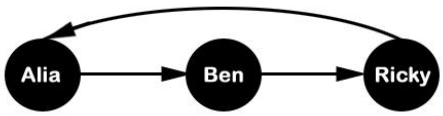

Figure1: Trust Propagation in Networked Environment

But distrust is somewhat trickier to compute as distrust is not at all transitive. Suppose Alia distrust Ben and Ben distrust Ricky. Ricky may be closer to Alia than Ben or Ricky may be further away. So in the propagation of distrust a transitivity problem gets raised and also it needs to be taken care of as to how to overcome with the problem of conflict of information.

2. Different Forms of Web Spam

There are different forms of web spam which can be mainly classified into different categories as mentioned below [3].

a) Content Spam: Content Spam is the most widespread form of web spam. The search engines use the page content to rank web pages in information retrieval model. The bb spammers can use the hidden drawback of this information retrieval model to spam the web pages. There are different ways of content spamming.

- **Title Spamming:** The title of a webpage has a very important role in information retrieval. The spammers spam the title of the spam page by over stuffing it so that the spam page gets a higher ranking.

- **Body Spamming:** Spamming the body of web page is the easiest and one of the most widespread techniques. If the spammers want to cover a defined set of query then the spammer can repeatedly use those words in the spam page that appear in the query. The spammers make those repeatedly used words invisible to the human eyes by using some color scheme.
- **Meta-Tag Spamming:** Meta-tags play an important role in the development of web pages. Spammers add their spam content to these meta-tags which lead the search engines to ignore the meta-tags in case of ranking a web page.
- **URL spamming:** The spammers spam the URL with the words which are in the targeted query. Thus the spam page gets the high ranking in the search engine results for those particular queries.

Nowadays, content-based spamming is being overcome with the use of link-based ranking algorithms. However, in such a case, the spammers start to spam the web by using the links of the web pages which is referred as Link Spam.

b) Link Spam: Link Spam can be mainly divided into two main categories - Incoming links and Outgoing links.

- **Incoming Links:** Incoming link spamming can be done in two different ways. First, the spammer can create a *link farm* (a bunch of interconnected linked pages) and then connect that *link farm* to its targeted spam page to gain a high Page Rank score as the Page Rank score is calculated based on the number of incoming links to a particular page.
- **Outgoing Links:** A spammer has full access to the outgoing links of its page and can add anything to it to get a higher page ranking.

3. Discussion On Standard Existing Anti_Spam Techniques

Many anti-spam algorithms have been proposed to fight against web spam. Among these techniques, link based semi-automatic algorithms that uses human expertise for propagation are considered to be the most efficient and effective techniques. These algorithms fall under two categories - Trust propagation and Distrust propagation.

In the Trust propagation category, the first algorithm proposed is Trust Rank Algorithm (2004) [4], where in the link structure of the entire web, few manually selected seed set of trusted pages is used to discover the other pages that are likely to be good or considered to be trusted pages. The second Trust propagation algorithm is Topical Trust Rank algorithm (2006) [5] which overcomes community biasness problem of Trust Rank algorithm by using the topical information to partition the seed set and calculate the trust score of each topic separately. The combination of these trust score for a page is used to determine its ranking. The third trust propagation algorithm is CredibleRank algorithm (2007) [6] where the credibility information is incorporated to check the quality of each page on the web.

Distrust propagation algorithms propagate distrust from the seed set of bad pages in the reverse direction of the incoming links to the entire web. Anti-Trust Rank algorithm (2006) [7] is the first distrust propagation algorithm. In this algorithm the seed set of spam pages are used to propagate anti-trust in the reverse direction to the entire web to detect the spam pages.

Trust propagation algorithms have the philosophy that good pages connect to good pages. These algorithms have limitations in some situations as not much analysis is done to show the connection from good-to-bad links. On the other side, Anti-Trust propagation algorithms have limitations in its penalizing factor, where it penalizes some good pages which unknowingly get connected to bad pages. In thiscase the good page should not be penalized. This issue is overcome by *Trust Rank and Distrust Rank Algorithm (TDR)* (2006) [8] which linearly combine the Trust Rank and Distrust Rank of a page. This TDR algorithm shows some improvement over the previous trust propagation and distrust propagation algorithms but has got some limitations too. It only shows improvement when one is dominant over the other, i.e, if *Trust* score is dominant over *Distrust* score or vice versa. Good-*Bad Rank Algorithm (2008)* [9] is another technique where both Good Rank and Bad Rank has impact on each other propagation. Here the Good Rank score and the Bad rank score of a page denote the good side and bad side of a page.There are some other anti-spam algorithms which discuss about the content and users' feedback information; however this is beyond the scope of this paper as we focus mainly on link based spam detection techniques.

4. Frame Work of the Standard Existing Anti Spam Techniques

Trust Rank Algorithm was the first trust propagation algorithm which was proposed in the year 2004. As spam detection is a very difficult task, Trust rank algorithm uses human assistance. The algorithm uses pages search and indexed by Atla Vista search engine. The algorithm first selects a small *seed* set of pages whose "spam status" needs to be determined. A human expert then examines the seed pages, and tells the algorithm if they are spam (*bad* pages) or not (*good* pages). Finally, the algorithm identifies other pages that are likely to be good based on their connectivity with the good seed pages [3]. Each web page has some incoming links which are also called as *inlinks* and outgoing links which are also called as *outlinks*. The total number of *inlinks* of a page is called *indegree* and it is defined as $I(p)$ and total *outlinks* of a page is called *outdegree* which is defined as $\omega(p)$.

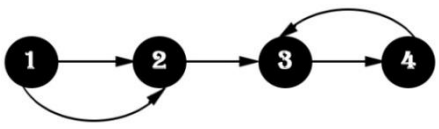

Figure 2.A Simple Web Graph

In figure 2 the *indegree* of node 2 is two and the

*outdegree*is one. The algorithm uses two types of matrix representation of web graph - one is transition matrix and another one is inverse transition matrix. The Transition matrix T is defined as:

$$T(p,q) = \begin{cases} 0, & if(q,p) \notin \varepsilon \\ 1/\omega(q), & if(q,p) \in \varepsilon \end{cases}$$

The Transition matrix of figure 2 is shown below:

$$T = \begin{bmatrix} 0 & 1 & 1/3 & 0 \\ 0 & 0 & 1/3 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1/3 & 0 \end{bmatrix}$$

The Inverse Transition matrix is defined as

$$U(p,q) = \begin{cases} 0, & if(p,q) \notin \varepsilon \\ 1/\omega(q), & if\ (q,p) \in \varepsilon \end{cases}$$

The Inverse Transition matrix of figure 2 is shown below:

$$U = \begin{bmatrix} 0 & 1 & 1/3 & 0 \\ 0 & 0 & 1/3 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1/3 & 0 \end{bmatrix}$$

Trust Rank algorithm somewhat rely on PageRank algorithm. PageRank algorithm is a well known algorithm for computing ranking of web pages based on the graph of the web or link information. It mainly uses the inlinks of a page for ranking [3]. The PageRank r(p) of a page is defined as:

$$r(p) = \sum_{q:(q,p)\in\varepsilon} \frac{r(q)}{\omega(q)} + (1-\alpha) \cdot \frac{1}{N}$$

where α is decay factor and N is the total number of pages.

Trust Rank algorithm uses transition matrix T and total number of web pages N as input. First the algorithm selects the seed set that returns a vector s(p). As the philosophy of Trust Rank algorithm is that "Trust flows out of good pages", the preference is given to the outlinks of a page so that many other pages can be reached. This technique is called as inverse PageRank. The difference between PageRank algorithm and inverse PageRank is that PageRank algorithm use inlinks of a page for ranking and InversePageRank algorithm uses outlinks of a page for ranking.

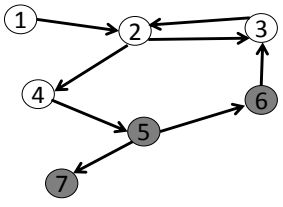

Figure 3 A Simple Web Graph [3]

Figure 3 is a simple web graph consisting of seven web pages of which pages 1, 2, 3, and 4 are good pages and page 5, 6, and 7 are bad pages. Let us now analyze how the TrustRank algorithm works. To choose an initial appropriate seed set *S*, considering L to be two, it is chosen as *S* = {2, 5}, as these web pages have the maximum number of outlinks. Here L is the size of the seedset. For calculation, a *SelectSeed* function (for selecting seed set) is used that uses Inverse Transition matrix as shown below:

$$s = \alpha \,.U\,.\, s + (1-\alpha)\,.\frac{1}{N}.\, 1_N$$

where α is the decay factor which is taken as 0.85, N is the number of pages and accordingly s will be changing for each iteration. After 20 iterations, the *SelectSeed* function for figure 3 will be:

s = [0.08 , 0.13 , 0.08 , 0.10 , 0.09 , 0.06 , 0.02]

As from the above result, the most desirable pages are page 2 followed by page 4, and so on. In the next step the elements are ordered in the decreasing order of their s score which is denoted by the Rank function σ. Again, for all web the pages in figure 3, the Rank function (σ) will be:

σ = [2, 4, 5, 1, 3, 6, 7]

The third step of the algorithm invokes the oracle function on L desirable pages and the static score distribution d which respond to the good seed pages are set to 1. In the fourth step of the algorithm the static score distribution d is normalized in a way that the most desirable good seed sum up to 1. The most desirable seed set is {2, 4} where both pages 2 and 4 are good seeds. The static score distribution v for figure 3 will be:

$$v = [0,\ \tfrac{1}{2},\ 0,\ \tfrac{1}{2},\ 0,\ 0,\ 0]$$

After 20 numbers of iteration, the trust score t^* for figure 3 will be:

t^* = [0, 0.18, 0.12, 0.15, 0.13, 0.05, 0.05]

Where t^* is calculated as:

$$t^* = \alpha_\beta \,.\, T.\, t^* + (1 - \alpha_\beta).v$$

Because of iterative propagation, the trust score of the good seed pages no longer have score of 1. But still the good seed pages have the highest score. In figure 4, the good seed pages 2, 3, and 4 got the higher scores. Page 5 which is a bad or distrusted page received a high score as it is directed by good page 4. This is the main drawback of TrustRank algorithm which does not look for good-bad links.

To sum up, TrustRank algorithm propagate Trust from the seed set of good pages to the outgoing links. But sometime spam page creator manages to put link on good pages to the spam pages. This type of drawback of TrustRank algorithm is outperformed by Anti-TrustRank algorithm [4]. Anti-TrustRank algorithm propagates anti- trust starting from the seed set of spam pages in the reverse direction along the incoming links. The algorithm selects the pages having high *PageRank* as the seed set so that higher number of pages can be reached in small number of hops while going in reverse direction along the incoming links. Here the Anti-TrustRank algorithm uses the same concept of Transition matrix and Inverse Transition matrix as TrustRank algorithm. Anti-

TrustRank is computed by using the Invesrse PageRank algorithm on the seed set. Anti-TrustRank score α^* is calculated as:

$$\alpha^* = \alpha'_\beta . T. \alpha^* + (1 - \alpha'_\beta).v'$$

The pages are ranked in the descending order of their PageRank score. This ordering could represent the estimated spam content of the pages. Finally, by declaring a threshold we can estimate which pages are having a greater score than the threshold value for spam detection. The only way to evaluate this is to check manually. The problem has also been discussed to be solved by using a heuristic practice which selects spam pages with 100% precision. Heuristic practices compile the list of a substring whose appearance in the URLs' is most certain to indicate that the page is a spam. It also compares the results of Anti-TrustRank algorithm with TrustRank algorithm and it can be seen that the Anti-TrustRank algorithm does much better than the TrustRank algorithm in detecting spam pages with high precision in different levels of recall and also detects the spam pages with relatively high PageRank, which is the main objective of the Anti-TrustRankalgorithm[10]. Hence, it can be concluded that the TrustRank algorithm can be used for the task of spam demotion while the Anti-TrustRank algorithm is used for the task of spam detection. But these techniques either use a good seed set or a bad seed set, which causes the loss of many useful information of the other side.

Later, it was analyzed that combining the use of both good seed set and bad seed set can lead to better results [11]. In [12], a linear combination of TrustRank score and Anti-TrustRank score of each page was carried out. But the issue with this technique is that it shows only a little improvement than the TrustRank and Anti-TrustRank algorithm.

The drawback of TrustRank and Anti-TrustRank algorithm is solved by the integrated framework of Trust-Distrust Rank (TDR) algorithm which was proposed in the year 2011. TDR uses both trustworthy side and untrustworthy side of each page which in turn makes full use of both good seeds and bad seeds. T-Rank score represents the trustworthiness and D-Rank score represents the untrustworthiness of a page [8]. The TDR algorithm uses a penalizing factor where the T-Rank/D-Rank in each iteration is penalized by the T-Rank/D-Rank of the target in the previous iteration. The T-Rank/D-Rank of a page is split equally by the number of outlinks/inlinks of the page. Then, it is propagated to page's outlink-neighbors/inlink-neighbors. The T-Rank of a page p is represented by t(p) while the D-Rank of a page p is represented by d(p). T-Rank t(p) is formalized as:

$$t(p) = \alpha \sum_{q:q\to p} \frac{\beta t(p)}{\beta t(p)+(1-\beta)d(p)} . \frac{t(p)}{\omega(q)} + (1-\alpha).v(p),$$

where, g is the static distribution vector of good seeds same as TrustRank algorithm and (1-β).d(p) is used to penalize the propagation of trust. D-Rank d(p) is formalized as:

$$d(p) = \alpha' \sum_{q:p\to q} \frac{(1-\beta).d(p)}{(1-\beta).d(p)+\beta t(p)} . \frac{d(p)}{l(q)} + (1-\alpha').v'(p),$$

where v' is the static distribution vector of bad seeds same as Anti-TrustRank algorithm and βt(p) is used to penalize the propagation of distrust. $\beta(0 \le \beta \le 1)$ is the penalty factor which represents the impact of T-Rank and D-Rank on each other's propagation.

TDR algorithm takes Web graph, trust vector g of good seed, distrust vector v' of bad seeds, penalty factor β and decay factor α as input. It iteratively computes t and d and finally, the algorithm return T-Rank score't' and D-Rank score'd' as output.

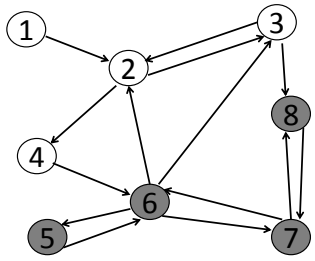

Figure 4. Web graph of good and bad pages [8]

If we consider Figure 4 and take {1, 2} as good seeds and {5, 7} as bad seeds, α and α' = 0.85, β = 0.5, then the TDR algorithm can be applied to compute 't' and 'd' and the result is as follows:

t = [0.166, 0.366, 0.152, 0.148, 0.004, 0.088, 0.021, 0.055]

d = [0.000, 0.019, 0.028, 0.065, 0.231, 0.297, 0.259, 0.102]

The result of TDR algorithm overcomes the drawback of TrustRank and Anti-TrustRank algorithms. All the good pages scores higher T-Rank than bad pages and all the bad pages scores higher D-Rank than good pages.

In 2013 another algorithm Good-Bad Rank (GBR) algorithm was proposed [9]. *GoodRank* score denotes the good side of a page and *BadRank* score denotes the bad side of a page. *GoodRank* represents trustworthiness of the page while *BadRank* represents possibility of the page of being spam. It has been experimentally proved that the GBR algorithm performs much efficiently than the TDR algorithm. The only difference between TDR algorithm and GBR algorithm is that GBR penalizes good-to-bad trust propagation and bad-to-good distrust propagation at the source end while TDR penalizes at the target end [9]. GoodRank score of a page p is denoted by g(p) and BadRank score of a page p is denoted by b(p). To find the possibility of page p being reputable, it is calculated as follows:

$$\frac{g(p)}{g(p)+b(p)}$$

To find the possibility of a page p being spam, it is calculated as follows:

$$\frac{b(p)}{g(p)+b(p)}$$

Same as TDR algorithm, the GoodRank/BadRank score of a page is split equally by the number of outlinks/inlinks of the

page and then is propagated to its neighbors. The GoodRankg(p) of page p is formulated as:

$$g(p) = \alpha.\sum_{q\in l(p)} \frac{g(q)}{|\omega(q)|}.\frac{g(q)}{g(q)+b(q)} + (1-\alpha).v(p)$$

The BadRankb(p) of a page p is formulated as:-

$$b(p) = \alpha' . \sum_{q\in\omega(p)} \frac{b(q)}{|l(q)|} . \frac{b(q)}{g(q)+b(q)} + (1-\alpha') . v'(p)$$

Here GBR penalizes trust/distrust at the source end.

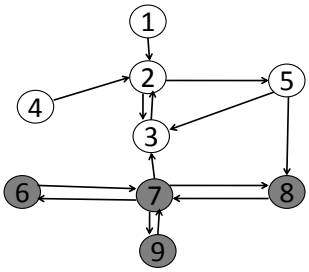

Figure 5. Web Graph of Good and Bad Pages [9]

Appling GBR algorithm in Figure 5 by setting decay factor $\alpha = \alpha' = 0.85$ and taking {1, 2} as good seeds and {7,9} as bad seeds, *GoodRank* and *BadRank* scores are as follows:

g = [0.132, 0.341, 0.202, 0.078, 0.144, 0.001, 0.044, 0.057, 0.001]

b = [0.000, 0.002, 0.030, 0.084, 0.019, 0.084, 0.453, 0.084, 0.243]

Finally, experimental results of GBR algorithm show that *BadRank* clearly outperforms both Inverse PageRank and Anti-Trust Rank. GBR not only has overtaken the drawbacks of TDR but GBR also has better time efficiency than TDR.

Table 1 below summarizes the benefits and limitations of the standard anti-spam techniques that have been discussed above.

TABLE 1. BENEFITS AND LIMITATIONS OF ANTI-SPAM TECHNIQUES

SI No	Anti-Spam technique	Benefits	Limitations
1	TrustRank	Filter index technique used based on trust propagation	TrustRank uses manually elected seed set of only trusted pages.
2	Anti-TrustRank	Filter index technique used based on distrust propagation	Anti-rustRank uses manually selected seed set of only disrusted pages
3	Trust-Distrust Rank	Uses the benefits of both TrustRank and Anti-TrustRank algorithms	Propagation of Trust/Distrust is penalized by target's current value of Distrust/Trust
4	GoodBad Rank	Overcomes the drawback of TDR algorithm	Works only on the Link Structure

Table 2 below summarizes the standard Anti-Spam techniques based on the measure of trust and distrust.

TABLE 2. ANTI-SPAM TECHNIQUES BASED ON TRUST-DISTRUST PARAMETERS

Serial No	Anti-Spam Technique	Measure of Trust	Measure of Distrust
1	TrustRank	Yes	No
2	Anti-TrustRank	No	Yes
3	Trust-Distrust Rank	Yes	Yes
4	GoodBad Rank	Yes	Yes

5. Conclusion

In this paper we surveyed most of the existing spam detection/demotion techniques and algorithms and also presented an overview of various forms of spam. We mainly focused on link-based semi-automatic spam detection and covered the most efficient four different types of spam detection techniques along with their pros and cons. In the future work, content-based spam detection/demotion techniques will be studied to come up with hybrid approaches of web spam detection that uses both link structure as well as content of web pages.

References

[1] Henzinger, M. R., Motwani, R., & Silverstein, C. (2002, September). Challenges in web search engines. In *ACM SIGIR Forum* (Vol. 36, No. 2, pp. 11-22). ACM.

[2] DuBois, T., Golbeck, J., & Srinivasan, A. (2011, October). Predicting trust and distrust in social networks. In Privacy, Security, Risk and Trust (PASSAT) and 2011 IEEE Third Inernational Conference on Social Computing (SocialCom), 2011

IEEE Third International Conference on (pp. 418-424). IEEE.
[3] Spirin, N., & Han, J. (2012). Survey on web spam detection: principles and algorithms. ACM SIGKDD Explorations Newsletter, 13(2), 50-64.
[4] Gyöngyi, Z., Garcia-Molina, H., & Pedersen, J. (2004, August). Combating web spam with trustrank. In Proceedings of the Thirtieth international conference on Very large data bases-Volume 30 (pp. 576-587). VLDB Endowment.
[5] Wu, B., Goel, V., & Davison, B. D. (2006, May). Topical trustrank: Using topicality to combat web spam. In Proceedings of the 15th international conference on World Wide Web (pp. 63-72). ACM.
[6] Caverlee, J., & Liu, L. (2007, August). Countering web spam with credibility-based link analysis. In Proceedings of the twenty-sixth annual ACM symposium on Principles of distributed computing (pp. 157-166). ACM.
[7] Krishnan, V., & Raj, R. (2006, August). Web Spam Detection with Anti-Trust Rank. In AIRWeb (Vol. 6, pp. 37-40).
[8] Zhang, X., Wang, Y., Mou, N., & Liang, W. (2011, April). Propagating Both Trust and Distrust with Target Differentiation for Combating Web Spam. In AAAI.
[9] Liu, X., Wang, Y., Zhu, S., & Lin, H. (2013). Combating Web spam through trust–distrust propagation with confidence. Pattern Recognition Letters, 34(13), 1462-1469.
[10] Pruthi, M. J., Kumar, E., & Noida, G. (2011). Anti-Trust Rank: Fighting Web Spam. International Journal of Computer Science Issues (IJCSI), 8, 315.
[11] Zhao, L., Jiang, Q., & Zhang, Y. (2008, July). From good to bad ones: Making spam detection easier. In Computer and Information Technology Workshops, 2008. CIT Workshops 2008. IEEE 8th International Conference on (pp. 129-134). IEEE.
[12] Wu, B., Goel, V., & Davison, B. D. (2006, May). Topical trustrank: Using topicality to combat web spam. In Proceedings of the 15th international conference on World Wide Web (pp. 63-72), ACM.

Authors Profile

Zenith Azim,is a post graduate student of Don Bosco School of Engineering and Technology, Assam Don Bosco University. She completed her Bachelors in Engineering from DBCET in 2012 and currently pursuing her Masters in Technology in Computer Science and Engineering. Her specialization is Data Mining.

Mrs. Gypsy Nandi,is currently working as Assistant Professor at Assam Don Bosco University. She completed her M.Sc and M.Phil in Computer Science. She is currently pursuing her Ph.D. in Computer Science. Her areas of interest are Data Mining and Machine Learning.