International Journal of IT & Knowledge Management (ISSN: 0973-4414)
Volume 11 • Issue 2 pp. 1-13 Jan 2018 - June 2018 www.csjournals.com

Forecasting Trendiness Weight of a Term in microblogs using Trendiness Distance & Poisson Distribution

Pradyumansinh Jadeja[1], Dr. Ketan Kotecha[2]
[1]Computer Engineering Department, Nirma University, Ahmedabad – 38248, Gujarat, India
[2]Parul University, Waghodia, Vadodara-391760, Guajat,
[1]pradyuman.jadeja@darshan.ac.in, [2]drketankotecha@gmail.com

Abstract: The rapid innovations in recent technology have alchemized the human lives from dawn to dusk. The 'exponential growth' in the usage of a smartphone has elevated standard of living of multitude as part of their natural habits. Social media has transformed the way of daily communication. From dot-and-dash era to recent phenomena in the field of communication, recent tunnels of communications have been uplifted in number as well as in quality through constant research and development worldwide. The latest list of tunnels of communication grades "Social Media" as an ineluctable tunnel considering its huge importance through which majority of earth population share their beliefs, feelings, experiences, liking - disliking, dissatisfaction, hatred, love, feedbacks and infinite human traits can be added here. The conspicuous excrescence in recent information environ indicate us to be conscientious to explore ongoing trends in social media, decipher its implicit betterment in the unborn future encompassing utilization patterns, archetypes of communication and engagement techniques. This paradigm shift would genuinely inspire us to predict sort of "THE NEXT BIG-BANG" for communicators, researchers, technocrats, leaders, governments, corporates and many since its inception. We have put forth more accurate and refined prediction algorithm which deals with Trendiness distance of term and instrument Poisson distribution function in addition to offering a sense of meaning to stand alone Trendy terms and concept of Trend/No-Trend border.
Keywords: Trend weight Prediction, Poisson Distribution, Trendiness Distance, Trendy Term Pair, Trend / No-Trend Border.

Introduction

Through systematic approach of analytical procedure when we observe different tweet behaviors we can notice that different individual users have few chance to have similar theme of tweeting about a specific topic, but when we collect tons of tweets from lots of users in nearer geographical regions during specified time span, we can observe that these tweet collection may give measurable weight to specific topic. With the implementation of Trendiness Distance, we are able to find out Trendy Terms and effectiveness of Terms i.e. how much trendy a Term was during given time interval? This gives thorough about increasing trend, decreasing trend or steady trend of a given term in Time Interval.

This may be more helpful if we are able to predict trendiness weight of a term in next Interval or Tweet-Bin based on the knowledge available with us, this prediction of trendiness distance of a term can be utilized as a tool which helps to plan future tasks for company/organizations. If one gets prior knowledge of Trend of a specific product or service feedback, one can get a better idea to serve more effectively in future and this leads to improving service or product quality for future requirements. Let's have a real life example where this model can be applied. The Government of India announced the DEMONITISATION on 8 November 2016. As per the guidelines of government, people required to submit old currency notes in bank accounts. As this was the sudden announcement by the government, there was a huge rush of people to deposit amount in the bank. At every bank, there were people standing in queues to deposit old currency notes & also queues of people to exchange old currency notes with the new one. In previous implementation Titled "Trend weight & Trendiness Distance", trendy words like 'bank', 'que', 'modi', 'money', 'india', '2016', 'people', , 'notes', 'cash', 'demonetisation', 'govt', 'new', 'black' and many more were identified. Among these terms 'people', 'que', 'money', 'notes', 'cash' gives the idea about people are standing in a queue for money (either for withdrawal or for deposit). What if we are able to predict trendiness weight of these words in a more systematic way? If trendiness weight gains significance with, it is evident for bankers to arrange more staff or arrange better facilities to serve people in a better efficient and effective manner. Similarly, if trendiness weight loses significance for next few time intervals, it gives surety to bankers about no more queues with the bank in next few days so they can divert their resources to where it is needed.

The Poisson distribution is a discrete probability distribution for the counts of events that occur randomly in a given time-frame. In microblogs, millions of users are writing about different topics at a different time and all tweets are purely independent of one another. So we can say that to tweet is an independent event performed by users and content generated by all users across different time intervals and across different geographical regions are also independent of one another, this nature of content generated by microblogs

leads us to apply Poisson distribution for the analysis of the content of microblogs. In this paper, we have used discrete probability distribution function "Poisson Distribution" along with Trendiness Distance count as an input to forecast trendiness distance or trendiness weight of a given term in next Tweet-Bins. The same approach can be applied for next series of intervals, which gives future behavior of terms (upward / downward trend).

The remaining part of the paper is arranged in following manner. Section 2 summarizes work done relevant with the concept of the Trend, like a stochastic model to explain the growth of topics, identifies topic distributions over time series, demonstration of how real-world outcomes are predicted using social media content. Section 3 describes proposed work, the concept of Poisson distribution, Trend / No-Trend border & concept of Trendy Term Pair. Section 4 describes Flow Chart, Proposed Algorithms, architecture, data preparation procedure with data set design and results. Section 5 includes conclusion of the paper.

Related Work

Sitaram Asur et al. [1] have proposed a stochastic model to explain the growth of topics which are trending with time and concluded that it follows lognormal distribution. They also proposed that topics those are long-trending follows a geometric distribution. Their analysis produced conclusion that tweet-rate and number of followers of any twitter users are not contributing trends. They also found that trendy content was largely news from traditional media, which are amplified by followers doing retweets on Twitter to make it trends. George H. Chen et al. [2] have developed hypothetical avocation for the applicability of nearest-neighbor-like classification of time series. Latent source model for time series is proposed, which redirects to a "weighted majority voting" classification rule that can be computed approximately by a nearest-neighbor classifier. They have concluded by different experiments that weighted majority voting do have the same misclassification rate as nearest-neighbor classification in observation in the less time series. They have used weighted majority to predict Trendy news topics, where they are succeed to identify such "trending topics" in advance of Twitter 79% of the time. Noriaki Kawamae & Ryuichiro Higashinaka [3] presented a topic model that identifies topic distributions over time series. In the proposal Trend Detection Model (TDM), each document uses a latent trend class variable. Continuous distribution over time and a probability distribution over topics are there with the trend class. Experiments reveled that TDM is useful as a general model for the analysis of the trend evolution. Sitaram Asur & Bernardo A. Huberman [4] have demonstrated how real-world outcomes is predicted using social media content. Especially they have used the Twitter.com chatter to predict box-office revenues for movies. They further shown improved forecasting power of social media using sentiments extracted from Twitter. First, they assess how attention is generated for different movies and how that changes. After that they focus on viral marketing and pre-release hype mechanism on Twitter. Their observation says that movies that are well discussed about will be well-watched. They have employed package named LingPipe linguistic to propose sentiment analysis classifier. They also used the DynamicLMClassifier that takes categorized character sequences' training events as input. Yaniv Altshuler et al. [5] have proposed an experimental model for the social scattering dynamics of spreading network patterns. Based on the statistics and analysis of discussion of community members they have predicted future trends using their proposed information diffusion models. They have applied a lower bound for the probability of a pattern concept to grade any trend as domestic or international trend in social network for any level of spreading. They have used results that studied the local social influence among members to model interaction between users. Shuyang Lin et al. [5] have studied the problem of predicting dynamic trends in social networks. They have designed trend forecasting algorithm using proposed model a Dynamic Activeness (DA) which works on the novel concept of activeness. DA model contains elements like activeness propagation, decay of activeness and action generating process. Christoph Trattner at Al [7] have used external sources of knowledge like social networks for prediction of interactions in online social networks. As a conclusion they have produced two different types of feature sets network- and content-oriented feature sets from data sources.

Proposed Work

A. Poisson distribution

At the point when the possibility of achievement or success is next to no and the quantity of trial or dataset is expansive, an approximation to the binomial distribution names Poisson distribution is utilized which is discovered by Simeon-Denis Poisson in 1838. The Poisson distribution is also known as the law of small numbers because in spite of having the many opportunities of occurrence the Poisson events occur very rarely. In probability theory and statistics, the Poisson distribution is a discrete probability distribution which implies the probability of a number of events occurring in a bonded period of time if these events occur with a regular average rate and irrelevant of the time since the latest event. The Poisson distribution is applicable

to the phenomena having a greater number of possible results and each result is rare. A Poisson distribution becomes a good approximation of the binomial distribution for a large number n of trials and small probability $\mathbf{p \sim \mu/n}$, $\boldsymbol{\mu}$ a constant ($\mu<<1$). In the limit $\mathbf{n \to \infty}$ **and** $\mathbf{p \to 0}$ so that the mean value $\mathbf{np \to \mu}$ stays finite, the binomial distribution becomes a Poisson distribution [9].

If we discuss Poisson distribution with respect to microblog application Tweeter, in Tweeter millions of people/users are tweeting (posting a message) at the different time about different topics and all tweets are pure independent from each other. So we can say that "to tweet" is an independent event performed by users and content (Tweets) generated by users across different time intervals and across different geographical regions are also independent of each other. Also, there is very less change of discussion about the same topic by millions of users in fixed time duration, this nature of content generated by Tweeter leads us to apply Poisson distribution for the analysis of the content of microblogs.

Let X = No of Events in a given Interval and λ is mean value then probability of observing x events in a given interval is denoted by [8].

$$P(X) = e^{-\lambda} * \frac{\lambda^x}{x!} \qquad (1)$$

Where

- x = Expected value in next time cycle
- λ = Mean no of events per time cycle
- e ≈ 2.718282 (The symbol e is a mathematical constant which is the base of the natural logarithm, the unique number whose natural logarithm is equal to one)

In our case we do not have way to find λ (mean value of given interval), so we are considering previous interval value as λ as a mean of a Term.

$$\mathbf{x = S_eM + \lambda} \qquad (2)$$

$$S_eM = \sigma_M = \text{Standard Error of the Mean} = \sigma/\sqrt{N} \qquad (3)$$

Where

- σ = Standard deviation
- N = Number of Samples
- For Poisson distribution $\sigma = \sqrt{\lambda}$

The input to this approach is a collection of trendy words with some statistical information of those terms in last Tweet-Bin and output will be predicted as Trend Weight of the same term in next Tweet-Bin. As a previous Tweet-Bin statistics, we may have variation or combination of frequency of Term in Bin, TF-IDF, Trendiness Distance & number of tweets that contains given term.

Suppose that we are having numbers of Tweet-Bins of a day and each tweet-Bin do have approximate 5000 tweets or approximate one to two hours duration for collection of tweets; Now for a term "demonitisation" for Tweet-Bin no 1 (say 10:00 Am to 11:00 AM, dated 02-11-2016) the weight of the term in this bin is say 44 then what will be the probability of observing 44 ± SeM weight in next time interval or next Tweet-Bin (11 AM to 12 AM, dated 02-11-2016).

B. Effect of Selection of Type of input on Result

We have found out the probability of a given term (weight) [eq. (1)] in next Tweet-Bin with reference to the Weight of Term in current Tweet-Bin. This procedure needs to be carried out for next consecutive time spans (Tweet-Bins) by applying predicted value or actual weight as input, which gives us series of weights of the term in next cycles. For same cycles, raw tweets are available from which actual weights and trendiness distance are calculated. Predicted series values (weights) and actual weights are plotted in given time series and it comes under observation that both graphs follow same direction & shape, hence the weight forecasted follows the actual weights in series (Tweet-Bins).

The Poisson distribution function is used to calculate the probability of term weight (Actual Term Weight in Current Tweet-Bin ± SeM [eq. (3)]) in next Tweet-Bin. Three kinds of inputs are to be taken for experiments 1) Number of Tweets that contains given word/term, 2) Frequency of Term in Tweet-Bin and 3)

Trendiness distance of Term in Tweet-Bin. Among these, the number of tweets containing term and Trendiness distance produces good results in comparison of considering only TF. So, type of input also plays a major role to produce meaningful and good results as input data must be logically fit for algorithm or formula.

While analyzing tweet results and forecasting probability in microblogs, two things are most important & challenging. First is the quantum of data to be handled and the other is an amount of time required to predict as one needs to deal with tons of data and must produce results with no time. In experiments after getting set of Trendy Terms, there is no need to dig deep for next Tweet-Bin result, one just needs to have few statistics of a given term in current Tweet-Bin which reduces an overall size of the dataset to be processed for next prediction and hence requires less processing time which results in speed. & same results can further be used for next Tweet-Bin processing..

C. Adding sense of meaning to Trendy Terms

If individual trendy terms are considered like 'black', 'money', 'que', '2016' & 'India', no one is able to grab meaning from individual terms because 'black' may be considered as colour to represent any living or non-living objects, there no evidence to say that term 'black' represents black-money. Same way 'que' may be referred as a queue of people, queue of students or queue at bus-stop; there is no direction specified by this terms which says that 'Queue of people at Bank'. Now instead of individual trendy terms if anyhow we are able to identify 'Trendy Term Pair' it will be more meaningful to recognize the meaning of Trend. Following trendy pairs add essence to the meaning of trend then individual trendy terms. The cluster of trendy term pair 'Man – Que', 'Bank – Que', 'India – Que', 'People – Que' and 'Day – Que' represents that there may be a queue of man or people in India at Bank. The cluster of trendy term pair 'Black - Money', 'Modi - Money', 'India - Money', 'Bank - Money', 'New - Money' represents that there matter of money may be black money or new money at bank going with India. The cluster of trendy term pair 'New - Notes', 'Bank - Modi', '#demonetisation - Modi', '#demonetisation - India' represents talk of demonetization, new notes in India.

We can say trendy term pair adds more meaning to understand the trend, so as an experiment we have paired top 56 trendy terms, which result in total 1540 combinations.

Permutations: Arrangement of different objects (in our case Trendy Terms) in a definite order is called permutation. One should use permutations if there is a need for the number of arrangements of objects and orders of arrangement are also important for a problem. We are not interested in which order terms appeared in the discussion, we just want to measure the presence of term pair in any order [10].

Combinations: On many occasions, the arrangement of different objects is not of importance, but selection of r objects from given n objects is of importance. A combination is a process of selecting few or all different objects where the sequence of objects is not considered [10]. One should use combinations if they need of the number of ways of selecting objects and the order of selection is not to be considered. The number of selections of r objects from the given n objects is denoted by.

$$^{n}C_{r} = C(n,r) = \frac{n!}{r! \times (n-r)!} \quad \textbf{(4)}$$

Here n = number of objects and r = number of combinations, in our case n = 56 (top 56 trendy terms are considered as pair) and r = 2 (we want pair of terms)
So,

$$^{56}C_{2} = C(56,2) = \frac{56!}{2! \times (56-2)!} = 1540 \text{ [eq. (4)]}$$

After combinations, now the goal is to find out trendy term pairs from 1540 pairs, pairs occur more frequently in the discussion. One needs to dig deep again in the statistics calculated previously, so now algorithm calculates a frequency of a pair of the term (in any combination) in Tweet in each Tweet-bin. Trend weight of each pair is calculated in all Tweet-Bins and found top trendy term pair which is listed in Table – 2.

D. Trend / No-Trend Border

While working with the concept of Trend weight and Trendiness distance [11] to find out trendy terms, it is observed that all terms are not contributing to generate the trend. There are tons of terms processed in a

derivation of trendy terms, among these, lots of terms do not have any contribution toward the trend in the collected bunch of Tweets. By observing top trendy terms from each Tweet-Bin, one can say 70 to 80% terms actually does not give direction toward trend, either those terms are not discussed heavily in Tweet-Bin (Time-Span) / region or these terms are so independent of another term that it cannot contribute because of loneliness. Trendiness distance [11] gives evidence that major terms in Tweet-Bin are so far from trendy terms and we can neglect them for any calculation without affecting results. Only 10 to 20% terms are nearer to trendy term and can be considered as real tread makers. If we remove terms having too far distance form trendy terms in Tweet-Bin, whose stake in total quantity is 70 to 80%, all over data set sized is reduced. The only issue with this approach is how to conclude that which terms are considered to neglect, how to detect the boundary of trendiness distance (Trend/No-Trend Border) such that we can say with confidence that remove all terms from Tweet-Bin having trendiness distance more than the detected boundary. Detection of Trend / No-Trend border using Trendiness distance is again area of research because this boundary is different for the different bunch of data. Table – 3 shows different statistics derived from dataset which provides evidence of Trend / No – Trend border

Experiments and Output

E. Flow Chart

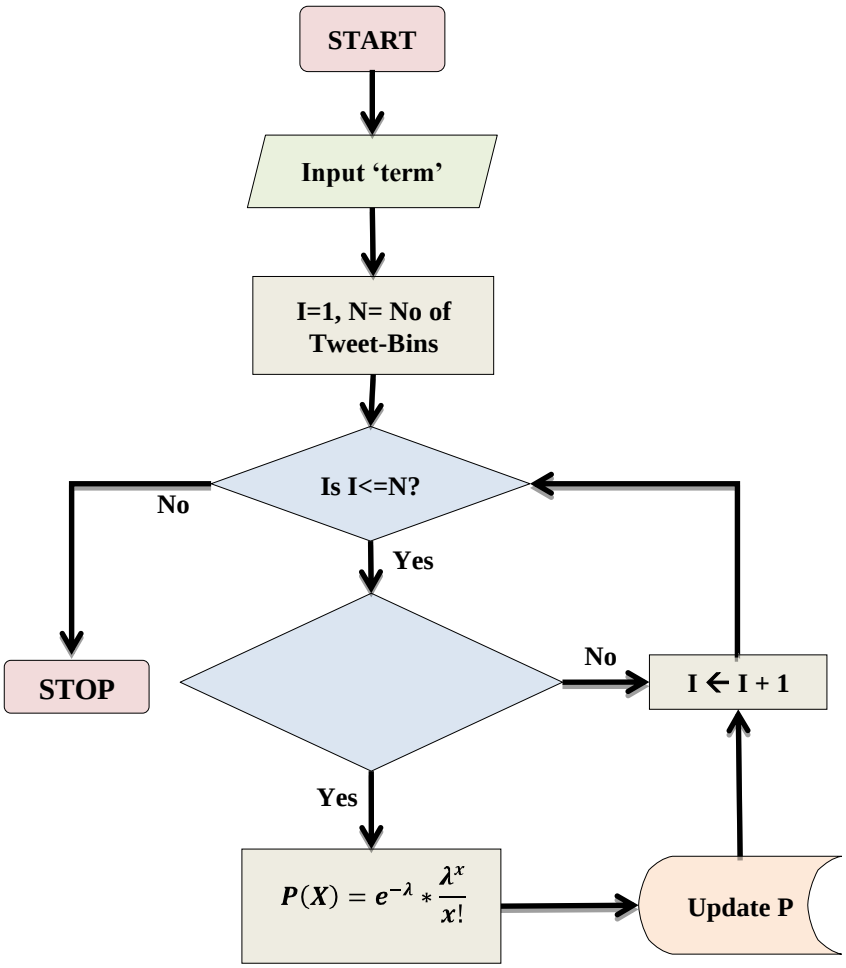


Figure 1. Flow Chart

F. Architecture

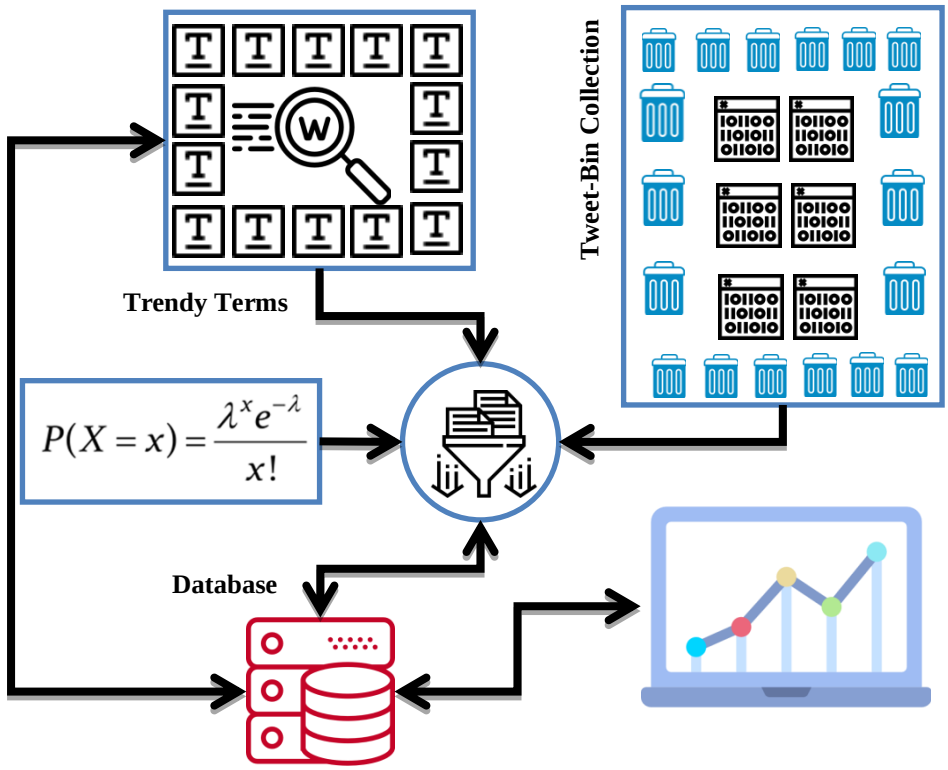


Figure 2. Architecture

G. Algorithm

CALCULATE_POISSON_VALUE (λ, *e*, *N*)

1 x = $\lambda + \left(\sqrt{\lambda} / \sqrt{N} \right)$ **[eq. (2)]**

2 $p = e^{-\lambda} * \frac{\lambda^x}{x!}$ **[eq. (1)]**

3**Return (*p*)**

Figure 3. Algorithm to Calculate Poisson Value

Fig.3 shows algorithm which takes λ (Mean no of events per Interval), e (e ≈ 2.718282, the base of the natural logarithm), and N (Number of Samples) as an argument and calculates Poisson Value

POISSION_MODEL_TO_TWEER-BINS(TERM)

```
1CONSTANT e← 2.718 281 828
2   for eachTweet-Binwhich containsTerm
3       TermWeight←Tweet-Bin[Term].Weight
4       TrendinessDistance←Tweet-Bin[Term].TrendinessDistance
5       TweetsWithTerm← TWEET_COUNT_OF_TERM(Term, Tweet-Bin)
6       TermWeight_PoissionValue← CALCULATE_POISSON(TermWeight, e, Tweet-Bin.TermCount)
7       TrendinessDistance_PoissionValue← CALCULATE_POISSON
                                    (TrendinessDistance, e, Tweet-Bin.TermCount)
8       TweetsWithTerm_PoissionValue← CALCULATE_POISSON
                                    (TweetsWithTerm, e, Tweet-Bin.TweetCount)
9       UPDATE_ POISSON_VALUE_TO_DB(Tweet-Bin, Term, TermWeight_PoissionValue,
                                    TrendinessDistance_PoissionValue,
                                    TweetsWithTerm_PoissionValue)
10end
```

Figure 4. Algorithm to Update Poisson Values of given Term for Next Tweet-Bin

Fig. shows algorithm which takes Term as an argument, for each Tweet-Bin it calculates the Poisson weight of a Term for next Tweet-Bin with three kinds of inputs (Term Weight, Trendiness Distance of Term &count of number of Tweets wreathing given Term in current Tweet-Bin)

In this algorithm, Poisson weight of a given Term for next Tweet-Bin (Time-span) is reckoned from statistics available with current Tweet-bin. Identical procedure compassed for each Tweet-Bin collection. One requires computing mean number of events per intervals (Tweet-Bin) and expected value before advancing towards Poisson value. As having unavailability of the way to find λ (mean value of given interval), we are accounting previous interval value as λ (mean value of a Term). A justifiable number of experimentations have been brought off with distinct types of inputs, say for the mean number of events 1) Term Weight in current Tweet-Bin or 2) Trendiness Distance of a Term in current Tweet-Bin or 3) Count of the number of Tweets wreathing given term in current Tweet-Bin are taken into consideration. There is a needfulness to have a number of samples to compute the desired value, so 1) number of Terms or 2) number of Tweets in Tweet-Bin can be considered as number of sample dependent on the selection of λ. If λ is supposed as Trend Weight in current Tweet-Bin or Trendiness Distance of a Term in current Tweet-Bin, number of Terms in Tweet-Bin is supposed as Sample count but in case if λ is supposed as number of Tweets wreathing given term in current Tweet-Bin, number of samples is supposed as the number of Tweets in Tweet-Bin. Fig. 4 represents algorithm CALCULATE_POISSON_VALUE (λ, e, N) to calculate expected value and Poisson value, x is expected value and p is Poisson value, this algorithm returns p to calling algorithm. Fig. 3 represents algorithm POISSION_MODEL_TO_TWEER-BINS(TERM) to update Poisson Values of given Term for Next Tweet-Bin. *Tweet-Bin[Term].Weight* represents weight of given Term in current Tweet-Bin, *Tweet-Bin[Term].TrendinessDistance* represents Trendiness Distance of given Term in current Tweet-Bin. Algorithm TWEET_COUNT_OF_TERM (*Term, Tweet-Bin*) takes Term and Twee-Bin an argument and returns count of number of Tweet contains given Term in current Tweet-Bin. Variables TermWeight_PoissionValue, TrendinessDistance_PoissionValue &TweetsWithTerm_PoissionValue represents Poisson weight of a given term for next Tweet-Bin for input Term Weight, Trendiness Distance & number of Tweet contains given Term respectively. Function UPDATE_ POISSON_VALUE_TO_DB(Tweet-Bin, *Term, TermWeight_PoissionValue,TrendinessDistance_PoissionValue,TweetsWithTerm_PoissionValue*)updates different Poisson values to database for a given.

Now in the database for all trendy terms in each Tweet-Bin, we have three actual weights and three predicted weights which are further represented in form of the graph which gives idea and comparison of actual value and predicted value in the graph.

H. Data Set

While dealing with microblogs, it takes greater time and human efforts to prepare dataset on which research analysis needs to be performed and thus it is a herculean task for any researcher. One wishing flavour of pragmatic approach, he would be benefited if he gets real life Tweets as the dataset. Tweeter offers API to pull haphazard Tweets. One of the unique features of Tweeter API is to produce geographical tweets from provided latitude, longitude & radius, which eventually proves to be highly fruitful for the identification of local Trends. In experiments, different Indian cities have been considered as data collection source within stipulated time period. One can conclude that produced dataset has been shifted to real life title.

- Tweets in Dataset: 4, 44,456
- Considered Indian Territories: Gujarat, Maharashtra, Rajasthan, Delhi
- Stipulated Time: 1st November, 2016 to 5th November, 2016

I. Experiemntal Setup

All raw tweets & processed tweets are stored in Database for implementation of the algorithm and to save intermediate results. The first step is cleaning of tweets before applying algorithm/calculation to intermediate data. Trendy terms are found using "Trendiness Distance" implementation which further used as input to the algorithm. Output to be expected is "Predicted Term Weight of given Term in next Tweet-Bin".

Following steps carried out on row tweets

1. CleanTweets: This includes removing stop words, very short words, user detail and non-text details
2. Segmentation of tweets
3. Dig collection of Trendy Terms from processed tweets using "Term Weight – Trendiness Distance" implementation
4. For each Trendy Terms calculate next Tweet-Bin weight using algorithm

J. Results

The input to experiments is set of Trendy terms and output is predicted weights of each trendy term for further few Tweet-Bins. Table – 1 shows actual and predicted the weight of term "demonetisation" for all Tweet-Bins, one can observe from the table that Poisson value follows the actual value in major Tweet-Bins. Along with predicting weight work experiment was carried out to found out Trendy Term Pairs to add the sense of meaning to Trend, Table – 2 Shows top Trendy pairs identified. Also, results of observation "Trend / No-Trend Border" are available with Table – 3.

Table – 1. Poisson value of term "demonetisation" in different Tweet-Bins

Bin No	Frequency Of Term in Bin	Total Term Frequency inBin	Poisson % of Term in Next Bin	No of Tweets with Given Termin Bin	Total Tweets in Bin	Poisson % of Term in Next Bin
1	41	39663	37	44	5000	37
2	42	39536	37	45	5000	37
3	42	39206	37	46	5000	37
4	28	38703	37	30	5000	39
5	42	38432	37	48	5000	37
6	17	38200	43	18	5000	47
7	28	37712	37	29	5000	40
8	39	38813	37	39	5000	37
9	50	38885	39	54	5000	37
10	63	40262	45	62	5000	39
11	61	39798	44	65	5000	40

12	40	39707	37	43	5000	37
13	38	38994	37	40	5000	37
14	58	38617	44	60	5000	38
15	17	37477	43	17	5000	48
16	17	36796	43	18	5000	47
17	43	39005	37	48	5000	37
18	47	40216	38	52	5000	37
19	44	40441	37	48	5000	37
20	49	40168	39	52	5000	37
21	52	39155	40	57	5000	38
22	39	38612	37	40	5000	37
23	48	38797	39	51	5000	37
24	17	13519	39	6	1717	47
25	29	37991	37	31	5000	39
26	22	36721	39	24	5000	42
27	67	39301	49	72	5000	42
28	77	40021	29	80	5000	46
29	84	40742	33	87	5000	51
30	52	40278	40	56	5000	38
31	48	39469	39	51	5000	37
32	37	39211	37	39	5000	37
33	44	39538	37	47	5000	37
34	22	37979	40	22	5000	43
35	17	36997	43	20	5000	45
36	44	38343	38	45	5000	37
37	50	40120	39	53	5000	37
38	66	40359	47	73	5000	43
39	61	40316	44	67	5000	40
40	56	39114	42	61	5000	39

Table 2. Top 100 Trendy Pairs

Sr.	Pair	No of Tweet-Bin Contains	No Of Tweets Contains	Sr.	Pair	No of Tweet-Bin Contains	No Of Tweets Contains
1	day - today	95	6305	26	black – india	93	1009
2	black - money	94	4489	27	like – look	94	982
3	day - days	94	3734	28	day – love	93	958
4	india - modi	95	3491	29	new – time	91	951
5	year - years	95	3185	30	modi – people	93	947
6	2016 - vote	94	2557	31	day – just	92	938
7	india - new	93	2317	32	day – time	92	920
8	day - india	94	2200	33	day – modi	93	905
9	day - man	94	2084	34	bank – que	92	901

10	india - man	93	2079	35	new – trump	93	896
11	india - time	93	1812	36	love – man	92	860
12	modi - money	94	1604	37	just – man	93	853
13	man - new	92	1504	38	india – que	93	848
14	day - new	92	1484	39	hai – modi	91	846
15	man - modi	95	1358	40	□□□ – □□□□	92	843
16	black - modi	93	1351	41	□□□ – □□□□	91	826
17	new - year	91	1339	42	man – people	92	824
18	man - que	93	1320	43	bank – modi	93	819
19	india - money	93	1273	44	like – video	93	816
20	hai - man	92	1198	45	#demonetisation - modi	92	813
21	india - today	94	1094	46	modi – sir	91	813
22	bank - india	92	1083	47	like – man	93	811
23	modi - new	94	1060	48	day – thank	93	784
24	india - people	93	1051	49	new – notes	91	777
25	man - time	92	1025	50	day – way	92	770
51	man - way	92	747	76	day – world	92	640
52	bank - people	90	730	77	day – people	91	639
53	cash - india	91	728	78	bank – cash	92	637
54	india - way	93	722	79	day – like	92	635
55	bank - man	90	719	80	vote – year	87	634
56	#demonetisation - india	92	719	81	man – money	90	630
57	new - que	90	718	82	new – way	93	628
58	come - india	88	716	83	just – like	93	625
59	india - make	92	714	84	bank – new	87	623
60	day - life	92	711	85	day – make	90	619
61	man - want	93	708	86	modi – notes	89	618
62	day - live	92	704	87	day – great	92	617
63	bank - money	89	702	88	2016 – man	90	611
64	day - year	91	701	89	day – que	90	610
65	india - live	90	701	90	bank – day	88	607
66	money - people	86	691	91	□□□ – □□□	91	604
67	people - que	93	691	92	black – man	92	597
68	modi - time	91	686	93	just – time	91	590
69	just - new	93	685	94	come – man	93	587
70	just - love	92	683	95	new – people	88	582
71	hai - india	93	677	96	love – way	92	577
72	man - year	92	673	97	money – new	89	575
73	day - work	93	663	98	know – man	93	574
74	india - just	89	653	99	india – work	92	574
75	2016 - india	89	647	100	come – day	92	572

Table 3. Trend / No Trend Border Statistics for 50 Tweet-Bins

International Journal of IT & Knowledge Management (ISSN: 0973-4414)
Volume 11 • Issue 2 pp. 1-13 Jan 2018 - June 2018 www.csjournals.com

Tweet-Bin No	No Of Tweets	No of Terms	Terms with Trendiness Distance				Tweet-Bin No	No Of Tweets	No of Terms	Terms with Trendiness Distance			
			<= 50	50 to 80	80 to 99	>=99				<= 50	50 to 80	80 to 99	>=99
1	5000	19855	9	58	4937	14842	26	5000	19909	10	65	5046	14778
2	5000	20051	8	62	4872	15101	27	5000	19938	17	94	5012	14798
3	5000	20340	9	59	4835	15428	28	5000	20909	10	78	5230	15581
4	5000	20286	8	55	4791	15424	29	5000	21064	12	79	5015	15946
5	5000	20184	5	39	4768	15367	30	5000	20590	13	64	4834	15666
6	5000	20934	11	66	4580	16266	31	5000	20251	16	87	4879	15253
7	5000	20235	9	36	4682	15499	32	5000	20367	18	109	4926	15296
8	5000	18934	3	35	2561	16332	33	5000	20695	7	54	4695	15932
9	5000	18635	4	40	2575	16012	34	5000	19569	4	21	2495	17045
10	5000	19163	5	49	2771	16333	35	5000	19861	11	56	4773	15010
11	5000	19495	9	51	4833	14593	36	5000	19700	22	99	4857	14700
12	5000	19956	8	58	4965	14917	37	5000	20924	19	106	4968	15812
13	5000	19674	10	62	4902	14690	38	5000	21179	11	68	5101	15988
14	5000	19386	12	60	4901	14401	39	5000	20433	15	95	4819	15489
15	5000	19892	12	63	4591	15214	40	5000	20669	20	83	4850	15696
16	5000	19444	6	26	4645	14761	41	4609	18901	11	54	4317	14508
17	5000	19312	18	63	4918	14295	42	5000	21046	6	41	4765	16228
18	5000	20052	8	46	5036	14954	43	5000	20426	4	20	2430	17968
19	5000	20437	11	59	4980	15376	44	5000	20600	18	80	4774	15710
20	5000	20422	15	66	4936	15390	45	5000	21212	12	84	4924	16180
21	5000	20337	14	52	4845	15412	46	5000	21160	9	77	5124	15941
22	5000	19729	20	77	4816	14796	47	5000	21453	13	91	4952	16384
23	5000	19481	19	86	4843	14514	48	5000	20818	15	78	4876	15834
24	5000	20686	9	58	4719	15891	49	5000	20014	4	48	4948	15010
25	5000	19949	5	22	4744	15173	50	3962	16769	17	81	16654	0

Term: demonetisation

X = Tweet-Bin No, Y=Trendiness Weight

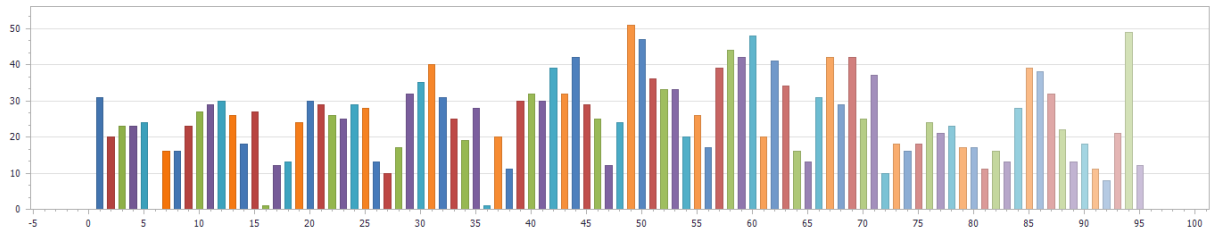
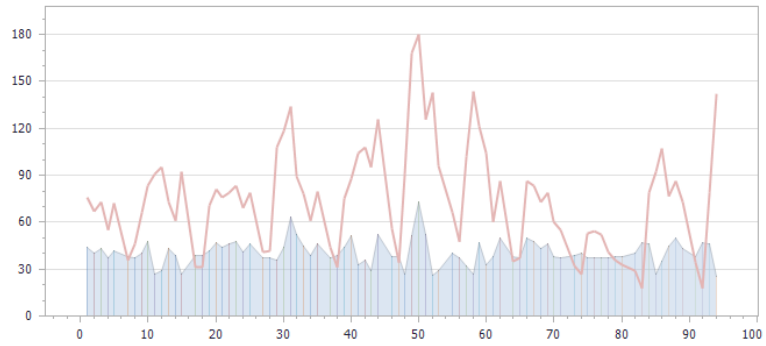


Figure 5. Tweet Count Chart of a Term "demonetisation" over TimeSpan



X = Tweet-Bin No,
Y= Red Line
Actual Trendiness Weight
Y = Blue lines
Poisson distribution Value

Figure 6. Trendiness Weight & Poisson distribution Value Chartof a Term "demonetisation"

Term: que (which represents Queue)

X = Tweet-Bin No, Y=Trendiness Weight

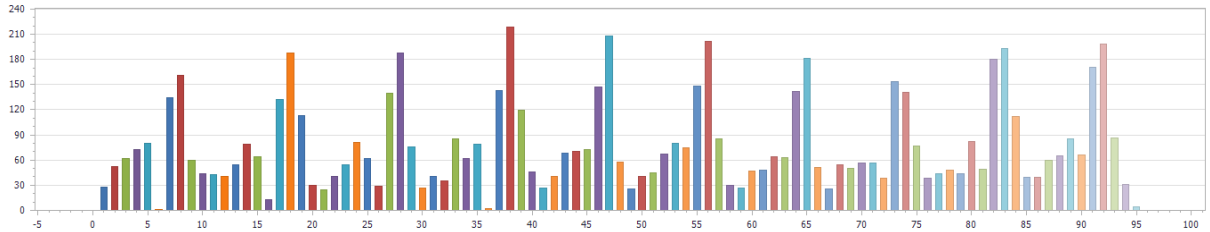
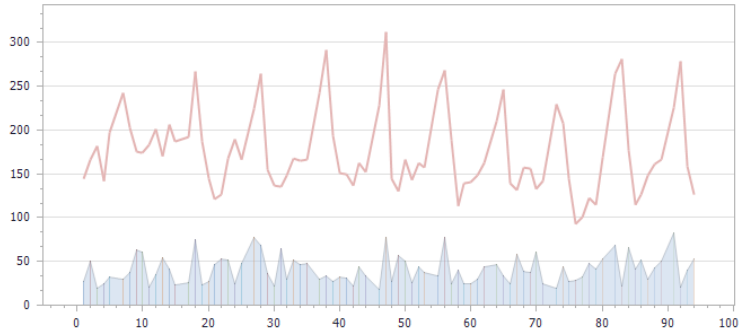


Figure 7. Tweet Count Chart of a Term "que" over TimeSpan



X = Tweet-Bin No,
Y= Red Line
Actual Trendiness Weight
Y = Blue lines
Poisson distribution Value

Figure 8. Trendiness Weight & Poisson distribution Value Chartof a Term "que"

Term: people

X = Tweet-Bin No, Y=Trendiness Weight

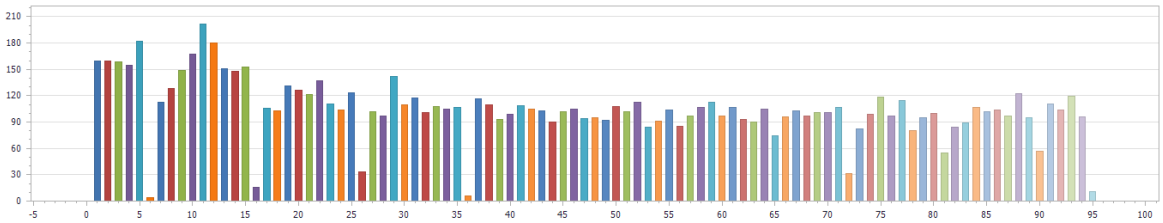
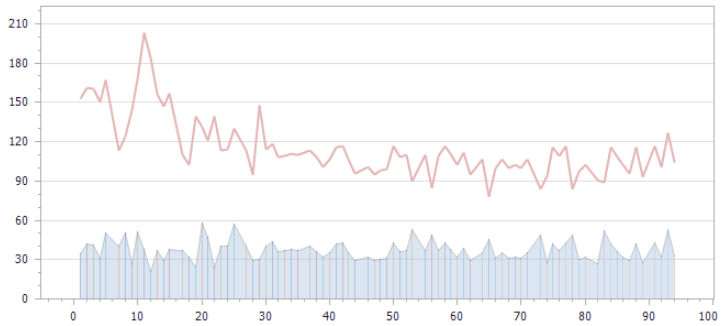


Figure 9. Tweet Count Chart of a Term "people" over TimeSpan



X = Tweet-Bin No,

Y= Red Line

Actual Trendiness Weight

Y = Blue lines

Poisson distribution Value

Figure 10. Trendiness Weight & Poisson distribution Value Chartof a Term "people"

Acknowledgment

All innovations in the world carry innumerable efforts and supports of humans, institutes & previous research. This paper would not have been written without sincere support of Nirma University. We are thankful to Nirma University for giving us platform & resources to carry out research work.

Conclusion

The nucleus of this study emphasizes that Social media has carved out an indispensable niche in this age of sapience. It has transformed the act of using words, thoughts, signs, and behaviors to express vital information in daily communication leading into creation of bunch of data in seconds. Social media generated data motivate us to prognosticate sort of "THE NEXT BIG-BANG" for amelioration of society. "Term weight – Trendiness distance" approach is employed to pinpoint trendy terms while Poisson distribution is used to add predicted weight to all Trendy terms in subsequent time cycles. Trendy term pairs are singled out by applying concept of combinations to trendy terms and using term-statistics available. To improve quality of research with a view to forestall extraneous terms from bunch, "Trend / No-Trend Border" concept is discussed.Further research on it will reduce dataset size without affecting actual results which eventuallyleads to less consumption of time and energy.

References

[1]. Sitaram Asur, Bernardo A. Huberman, Gabor Szabo (Social Computing Lab, HP Labs) & Chunyan WangSocial (Dept. of Applied Physics, Stanford University), "Trends in Social Media - Persistence and Decay", Proceedings of the Fifth International AAAI Conference on Weblogs and Social Media, 2011

[2]. George H. Chen (MIT), Stanislav Nikolov(Twitter), Devavrat Shah (MIT), "A latent source model for Nonparametric time series classification", NIPS'13 Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 1, Pages 1088-1096, December 05 - 10, 2013

[3]. Noriaki Kawamae & Ryuichiro Higashinaka, "Trend Detection Model", ACM, WWW 2010, April 26–30, 2010

[4]. Sitaram Asur & Bernardo A. Huberman, "Predicting the Future With Social Media", Web Intelligence and Intelligent Agent Technology (WI-IAT), International Conference IEEE/WIC/ACM, 2010

[5]. Yaniv Altshuler, Wei Pan, and Alex (Sandy) Pentland - MIT Media Lab, "Trends Prediction Using Social Diffusion Models", Intl. Conf. on Social Computing, Behavioral-Cultural Modeling, and Prediction", 2012

[6]. Shuyang Lin, Xiangnan Kong, Philip S. Yu, "Predicting Trends in Social Networks via Dynamic Activeness Model", CIKM'13, Oct. 27–Nov. 1, 2013

[7]. Christoph Trattner, Denis Parra, Lukas Eberhard and Xidao Wen, "Who will Trade with whom? Predicting Buyer-Seller Interactions in Online Trading Platforms through Social Networks", WWW'14 Companion, April 7–11, 2014

[8]. Scott Hendrickson, Josh Montague, Jeff Kolb, Brian Lehman, "Trend Detection in Social Data", GNP, June 2015

[9]. Hao Hu, "Poisson distribution and application", Department of Physics and Astronomy, University of Tennessee at Knoxville,Knoxville, Tennessee, USA, October 20, 2008

[10]. C.L. Liu, Introduction to Combinatorial Mathematics, McGraw-Hill, Inc. 1968.

[11]. Pradyumansinh Jadeja & Dr. Ketan Kotecha, "Trend Weight & Trendiness Distance - A Novel Approach to identify Trends in Microblogs ", International Journal of Advanced Research in Engineering and Technology (IJARET), Volume 8, Issue 6, 2017