Analyzing and Modeling Information Diffusion using Topic Inference Relationship Prediction

CS-511 Advanced Datamanagement Project

Nikolaus Nova Parulian
nnp2

Chaoyun Chen
chaoyun2

Wenzhuang Chi
wc4

Mei Mei
meim2

Mentor: Carl Yang

ABSTRACT

Information in social media flows from one user to others. The process when a piece of information is spread and reaches individuals through interactions is called Information Diffusion. Information Diffusion is an important field in social media mining to give a better understanding in how or why information can spread within the network. However, people often neglect the relation between information when learning about information diffusion. In this project, we will focus on modeling information diffusion and uncover hidden topic within information propagation to produce the relationship between individuals using the topic.

Keywords

Information Diffusion, Machine Learning, Topic Modeling, PCoA

1. INTRODUCTION

The type of information diffusion covered in this paper is information cascades. In the social network, there are nodes and edges with nodes stand for users and edges stand for relationships among them. As a fundamental element in social networks, understanding relationship functions is crucial to facilitate many important tasks or relationship-specific community detection. Thus, we need to figure out the semantics of those relationships discovering the pattern of information diffusion; we can make a better prediction of how information flows among (specific topics) and relationships [7].

Learning information diffusion is important to understand how a piece of information flows. For example, tracking customers that are influenced by a particular internet marketing tweet, measuring how fast information about earthquake spreading in social media. Whose author give the most impact for other blog authors given an information cascade.

NETRATE, a probabilistic model of diffusion to infer transmission rate within network given cascades that occur over time introduced by Gomez et al. [5]. However, the model just considers the cascades and time given only without covering any underlying parameter that might occur within each cascade. Accordingly, when we implement the model in the real data with numerous cascades, NETRATE algorithm often failed to predict the relation between nodes.

Barbieri, Nicola *et al.* [3] presented a model to provide a topic-aware influence-driven propagation in the Independent Cascade. This work is closely related to what we want to achieve in this paper. However, Barbieri et al. do not take into account the transmission rate in their modeling framework. For this project, we start with the observations that a node/user in social media may have the same or different interests *(i)*, information that is propagating within a network can be clustered into several topics of interests *(ii)*, user that have same interests may share similar topic of interest *(iii)*, the faster some users share same topics within the network, the closer they relation for that particular topics *(iv)*.

To produce a better prediction for the algorithm, we try to uncover underlying parameter that lies within the cascades and use that to improve the NETRATE model. We use EM framework to model a Topic Distribution using sample cascades then produce a words probability within the particular topic. Next, we use the probability to cluster the cascade and use the probability weight as a variance that affecting the transmission rate. Using this method, we prove that we get not only a better prediction but also the inference of what kind of topics has faster transfer rate within the edges. Since the infection probability is heavily depending on the transmission rate, a better prediction can be achieved if the EM framework provides the more accurate transmission rate. Furthermore, the transmission rate prediction with topic inference can be used to provide neighbor node (friends of a friend) recommendation for the observation node based on specific topics.

2. PROBLEM FORMULATION

NETRATE Algorithm. Gomez-Rodriguez et al. [5] proposed NETRATE algorithm to infer transmission rate. Suppose we have a number of Nodes N, and given a set of C contains cacsades $t^1, ..., t^c$ while t^c is a time series of the

cascades propagating to the nodes at some observation time. We can use convex optimization approach to maximize the log likelihood of the network inference problem and get this formula to be optimized:

$$L(\mathbf{t^1}..\mathbf{t^c};\mathbf{A}) = \sum \Psi_1(\mathbf{t^c};\mathbf{A}) + \Psi_2(\mathbf{t^c};\mathbf{A}) + \Psi_3(\mathbf{t^c}) \quad (1)$$

where for each cascade $\mathbf{t^c} \in \{t^1, ..., t^c\}$, each function can be derived into

$$\Psi_1(\mathbf{t^c};\mathbf{A}) = \sum_{i:t_i \leq T} \sum_{t_m > T} \log S(T|t_i;\alpha_{i,m}) \quad (2)$$

$$\Psi_2(\mathbf{t^c};\mathbf{A}) = \sum_{i:t_i \leq T} \sum_{t_j < t_i} \log S(t_i|t_j;\alpha_{i,m}) \quad (3)$$

$$\Psi_3(\mathbf{t^c};\mathbf{A}) = \sum_{i:t_i \leq T} \log \sum_{j:t_j < t_i} H(t_i|t_j;a_{j,i}) \quad (4)$$

Transmission Likelihood. For this experiment we use Exponential Model transmission likelihood f given as

$$f(t_i|t_j;\alpha_{j,i}) \begin{cases} \alpha_{j,i}.\exp^{-\alpha_{j,i}(t_i-t_j)} & \text{if } t_j < t_i \\ 0 & \text{otherwise} \end{cases} \quad (5)$$

Survival function. S is a probability of a node survives uninfected until time T. Given the transmission likelihood, we can derive the survival function of our equation as

$$\log S(t_i|t_j;\alpha_{j,i}) = -\alpha_{j,i}(t_i - t_j) \quad (6)$$

Hazard function. H is a Hazard function or instantaneous infection rate of edge $j \rightarrow i$. Given the exponential transmission likelihood model, we get our hazard function as

$$H(t_i|t_j;\alpha_{j,i}) = \alpha_{j,i} \quad (7)$$

With this formulation, we can use Convex Optimization to get A that maximize the log-likelihood of the equation (1)

However, this algorithm just considers one parameter cascade time only to infer transmission rate. As a result, when Gomez *et al.* tested the algorithm in the Meme Tracker dataset [1] with numerous cascades just produced about 40% accuracy [5]. Therefore, we will try to apply our proposed solution on the same dataset to verify whether or not our proposed solution combining the topic modeling is better than the NETRATE only as a baseline.

3. PROPOSED SOLUTION: NETRATE + TOPIC INFERENCE

To solve the problem, we proposed a solution to combine NETRATE algorithm and Topic Model inference for the cascades that happened within the network. The idea is simple, supposed we can cluster the cascade into several topics we can use the information to precisely define the convex formula (1) for each topic and get less variability to be solved by the convex optimization. By clustering observed cascades into several topics and integrating with the NETRATE algorithm, we can better predict the transmission rate and activation probability for specific topics.

Expectation Maximization (EM). To infer the topics we proposed Expectation Maximization (EM) multinomial topic modeling [4] to get expectation probability of words p_j, k by using calculation

$$\varrho(\theta;\theta^{(n)}) = \sum_{ij}(\sum_{k} x_{i,k} \log p_{j,k}) + \log \pi_j \quad (8)$$

And then we minimize the variational free energy in the M-step using

$$p_j^{(n+1)} = \frac{\sum_i x_i w_{ij}}{\sum_i x_i^T 1 w_{ij}} \quad (9)$$

and

$$\pi_j^{(n+1)} = \frac{\sum_i w_{ij}}{N} \quad (10)$$

where i is number of documents, j is number of words, k is number of topics that want to be inferred, $p_{j,k}$ is the probability of the words j within topic k.

From the EM Topic modeling inference we get the probability of words within each topic and we can use it to infer the probability of a cascade belong to each topic. However, for this topic modeling inference, we can only get the probability of topic assignment only. Therefore, instead of separating each observation cascade into several formula (1), we ingested the probability weight into the transmission likelihood [5] and derived it to the hazard function:

$$f(t_i|t_j;\alpha_{j,i};\theta_{ck}) \begin{cases} \theta_{ck}.\alpha_{j,i}.\exp^{-\alpha_{j,i}(t_i-t_j)} & \text{if } t_j < t_i \\ 0 & \text{otherwise} \end{cases} \quad (11)$$

and the hazard function will be

$$H(t_i|t_j;\alpha_{j,i};\theta_{ck}) = \alpha_{j,i}.\theta_{ck} \quad (12)$$

where θ_{ck} is a probability of the cascade c belong to the topic k.

$$\theta_{ck} = \frac{\sum_j x_j p_{jk}}{\sum_K \sum_j x_j p_{jK}} \quad (13)$$

With this formula we can produce k numbers of $\alpha_{j,i}$ that will infer how fast the transmission rate between node $j \rightarrow i$ for that particular topics from the expectation .

Principal Coordinate Analysis (PCoA). To visualize our experiment results in an aesthetic way, we use Principal Coordinate Analysis (PCoA) to predict the location of each node according to their transmission rate in the two-dimensional space.

Principal Coordinate Analysis is a means of visualizing the level of similarity of individual cases of the dataset. The algorithm aims to place each object in N-dimensional space such that the between-object distances are preserved as well as possible [2].

As we need a distance matrix for this coordinate prediction, we can convert the α_{ji} transmission rate from the experiment results into the distance matrix assuming that higher transmission rate provides higher similarity of two nodes. Therefore, we can formulate the distance following this equation

$$d_{ij} = \log \frac{max(\alpha_{ij})}{\alpha_{ij}} \quad (14)$$

With this equation, we can produce the distance matrix relative to the maximum transmission rate in the network and generate the visualization of the network using the PCoA coordinate prediction.

4. DATASET

To test our proposed solution, we used Meme Tracker Dataset; a big dataset contains phrases/memes in blogs or news websites. This dataset also contains hyperlinks when the blogs are quoting other sites on their page. We use hyperlinks between blog posts to trace the flow of information. A site publishes a piece of information and uses hyperlinks to refer to the same or closely related pieces of information published by other web sites. These other sites link to still others and so on. A cascade is thus a collection of time-stamped hyperlinks that are cited on the sites.

Using the hyperlink connection within the page domain and the parent domain (the link that they quote with), we construct the edges within each domain name and build a Meme Tracker Network as a ground truth for our edges prediction (fig 1). In addition, we also used the hyperlink and the page

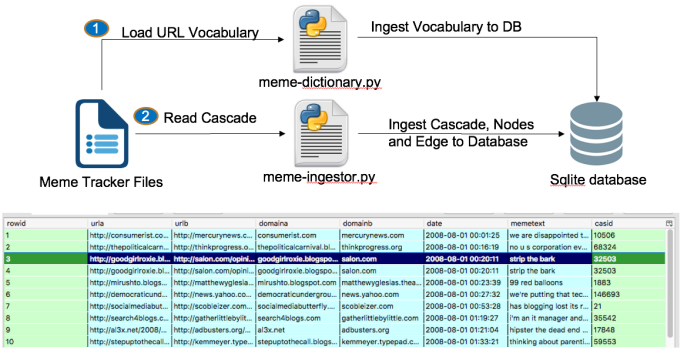

Figure 1: Meme Tracker Dataset Reconstruction

created time on the dataset to build a cascade of hyperlink within the network. As a result we produced 45,000 nodes, 186,000 edges and 540,000 observation cascades ready to be analyzed (fig 2).

5. EXPERIMENTS AND RESULTS

Firstly, we implement the NETRATE algorithm to the dataset to a direct network with *parents* $\rightarrow$ *child* relation. As a result, we get that the NETRATE produced 65% accuracy overall (figure 3). This proves that given a real data cascades, the NETRATE algorithm may not working properly due to unknown parameter either in each node, edges, or the cascades itself that might be affecting the prediction.

EM Topic Modeling. Our EM implementation is straight forward. Firstly, From all the cascades data we draw sample

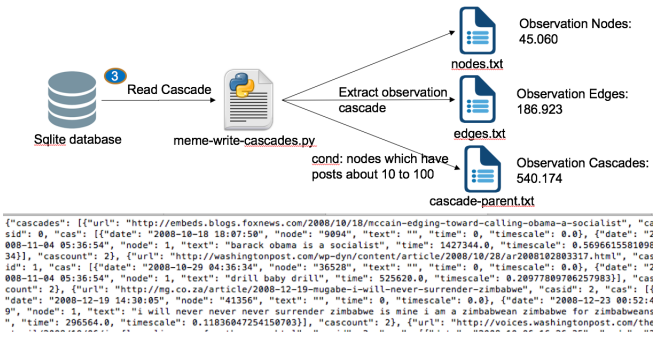

Figure 2: Meme Tracker Cascade Extraction

about 20,000 cascades and extract the term frequency from the parents' memes and get a distribution of words offer the memes. From this job, we produce 31,494 different words and 1,797,584 word frequency set to be analyzed by the EM algorithm.

We tried to cluster the memes into 15 topics using the EM algorithm. Besides, we initialize the topic center using k-means to get the possible centers/centroids from our memes - words distribution. Therefore, our topic probability result will be more precise because it is supported by the k-means and sharpened using the EM algorithm. The result of the Topic Clustering algorithm can be seen in the table 1. This process produced p_{jk} which contains probabilities of j words in the k topics.

Topic 1 (war)	Topic 2 (motivation)	Topic 3 (peace)	Topic 4 (politic)	Topic 5 (opression)
mars	know	commitment	bein	january
jihad	efficient	applaud	fails	jiving
lose	divisi	king	political	signs
bureaucratic	looking	fulfill	diaz	silencing
para	karni	peace	kala	water
world	scourge	laptop	families	peace
Topic 6 (housing)	**Topic 7 (journal)**	**Topic 8 (history)**	**Topic 9 (game)**	**Topic 10 (army)**
using	journal	history	game	mars
standards	involves	values	bureaucratic	para
british	stairway	para	heartiest	conveying
tv	drilling	mars	fulfill	history
electricity	establishment	applaud	tap	world
history	precipitate	peace	prophet	applaud
Topic 11 (teamwork)	**Topic 12 (election)**	**Topic 13 (finance)**	**Topic 14 (danger)**	**Topic 15 (innovation)**
tarnishes	great	pale	danger	phones
applaud	hours	unsought	wait	rates
team	world	middle	mars	great
great	king	charge	country	imagine
good	tarnishes	neo	duties	january
federal	task	baby	federal	defense

Table 1: EM Topic Modeling Results

Finally, we use the word probabilities and apply them into the other memes in the dataset to cluster the memes belong to the topics (eq. 13). In this case, we will get the probability of the cascades belong to each topic and produce θ_{ck} for each cascade.

Experiment on direct/known network. For the direct and known network, the algorithm will get all cascades within the close network. To test our solution combining NETRATE and topic weight we try to run the inference within the direct network and compare the result with the baseline which is 65% accuracy. To compare the accuracy, we produce the graphic that compares the accuracy within the number of cascades and number of nodes (fig 3)

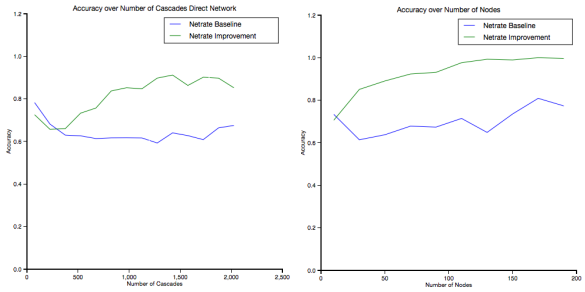

Figure 3: accuracy over a number of cascades and nodes

From the figure, we can see that our NETRATE Improvement solution using topic inference can beat the baseline by providing about 80% accuracy on average. We can also see that our topic inference can improve NETRATE algorithm by providing consistent accuracy given an increasing number of cascades that infecting our observation nodes that prove the method robustness comparing to the baseline that slowly degraded when the number of cascades is increasing. In addition, the method is more robust when the observation nodes we want to infer are increasing.

Visualize the Prediction In addition to the comparison analysis, we also provide visualization tools in Jupyter-Notebook to produce an insight visualization of our transmission rate prediction using Principal Coordinate Analysis (PCoA) [6] to compute the coordinate prediction in the 2-Dimensional space. The example can be seen in Figure 4.

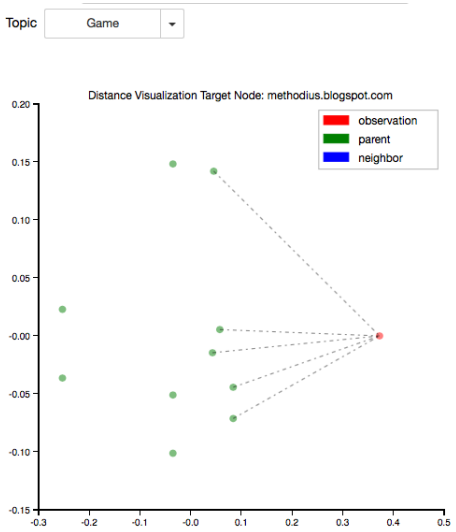

Figure 4: Network Topic Visualization

This visualization will plot all the parent node in the predicted space, in which we can infer that any nodes that do not have edges are having low / no transmission rate α. From the example, five nodes share same interests with the observation node related to the topic about Game, while the others are not.

Experiment on Neighbor Recommendation Network. After successfully experimenting in the close direct network, this brings us to our next question, how good is our proposed solution can work to predict relationship among unknown network, which will restrict the cascades to the specific node.

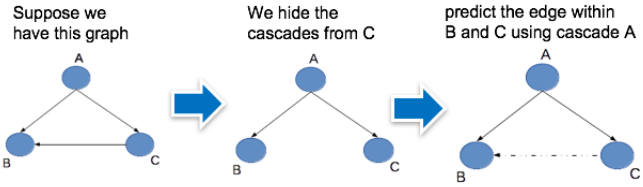

Figure 5: Neighbor prediction illustration

For this experiment, we still use the Meme Tracker dataset and sampling the nodes that have a neighbor relation as our observation nodes describe in fig. 5. However, to simulate this unknown network condition, we used cascades that started from the parent node only and used that information to predict if there is a relation between the observation and neighbor nodes. Also, to stress out our testing environment, we also add random (false) neighbors draw from the true neighbors.

Using the NETRATE baseline and Topic Inference solution brings us the result in figure 6.

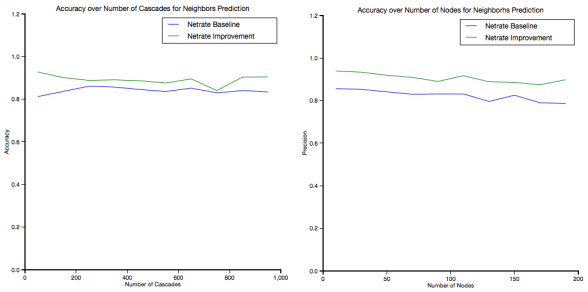

Figure 6: accuracy over a number of cascades and nodes

From the graph, we can see our Netrate + Topic Inference model again give use better prediction than the baseline. Using Netrate baseline, predicting this unknown network produces 80% accuracy while the Netrate + Topic Inference model produces 90% accuracy on average.

We also build visualization tools to visualize this neighborhood network prediction which can be seen in figure 7.

From the visualization, we can see that our neighborhood prediction model works well defining some nodes that do not have a direct relation (edges) depicted by blue nodes but have a strong relation to the particular topic given the node location in the 2-D space closer to the observation node described by the edges connecting them. This feature can be used for building a topic recommendation system, or friends of friend recommendation based on particular subject interest assuming higher transmission rate might support the argument that they are closely related.

Implementation on Full Network. The solution can

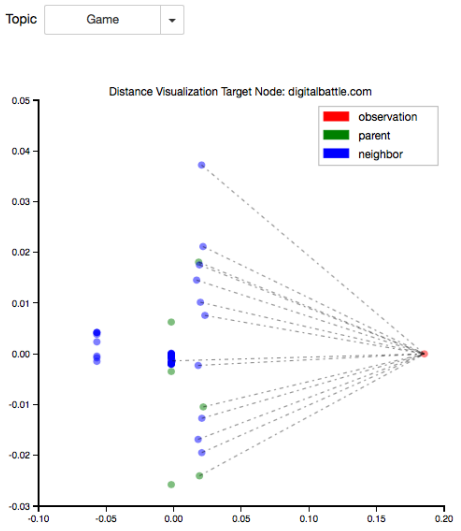

Figure 7: Neighbor prediction Visualization

be implemented into the full network by calculating all the transmission rates within the nodes in the observation network. Applying this method will produce transmission rate α matrix, and further, we can visualize the transmission rate and project it into 2-D / 3-D visualization as we can see in the figure 8 . From the visualization we can see that some

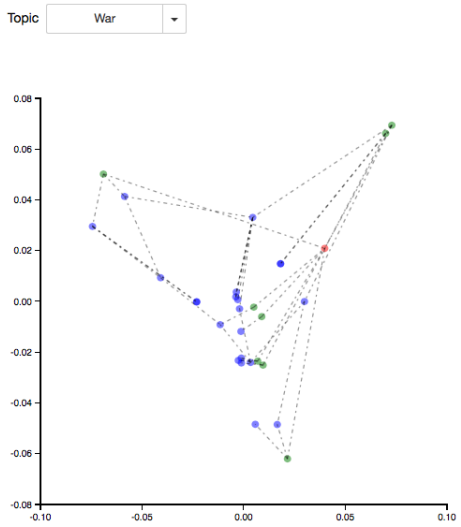

Figure 8: Full Network Transmission Rate

point are somehow clustered in some area, sharing same topic similarity. This information further can be used for community detection.

6. CONCLUSION & FUTURE WORK

In this project, we tested and validated the information diffusion prediction capability of NETRATE and proposed a new solution which combines NETRATE and Topic Modeling using EM algorithm. Two experiments were conducted for comparison of these two solutions. The results illustrated the advantage of the combination of NETRATE and Topic Inference. Based on the idea that different topics have different transmission rate, the EM algorithm helps infer topics probability for each cascade and further increasing the prediction accuracy for transmission rate and infection probability around 10% from the baseline.

Furthermore, the cascades which clustered as topics by EM algorithm can be analyzed for cascade similarity and neighbor nodes closeness. According to that information, a recommendation network could be generated to bring nodes which sharing similar topics together.

The EM Topic Modeling may not be the only topic modeling algorithm that can be used for the topic inference purpose. We can use other topic analysis such as LDA. Also, we can use supervised classification approach for learning the topic prediction if we want to further experiment in providing a model for topic classifier and get a more accurate prediction for the topic inference.

There are several possible research that can be done in the future related to this project. Firstly, we believe the proposed solution could be improved by using additional parameters such as user profile. By integrating profile data into the model, it will be practical to mining the relationship between nodes, and thus, making a more accurate prediction of transmission rate and infection probability. Secondly, combining the NETRATE + Topic Inference algorithm in a graph database is another feasible solution for better network structure visualization. Using the transmission rate, we can produce a visualization just like what we did using the PCoA into an informative graph or furthermore can be used for community detection based on the topic analysis. Finally, for the recommendation network, we think the solution could be implemented into a real-time network by using streaming process. As cascades and topics vary among the network over time, we can generate a dynamic recommendation network, which will be available for more real-time network analysis.

7. REFERENCES

[1] *MemeTracker*. http://www.memetracker.org/data.html.

[2] *Multidimensional scaling*. https://en.wikipedia.org, Apr 2017.

[3] N. Barbieri, F. Bonchi, and G. Manco. Topic-aware social influence propagation models. *Knowl. Inf. Syst.*, 37(3):555–584, 2013.

[4] C. M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006.

[5] M. Gomez-Rodriguez, D. Balduzzi, and B. Schölkopf. Uncovering the temporal dynamics of diffusion networks. *CoRR*, abs/1105.0697, 2011.

[6] J. C. Gower. *Principal Coordinates Analysis*. John Wiley Sons, Ltd, 2005.

[7] R. Zafarani, M. A. Abbasi, and H. Liu. *Social Media Mining: An Introduction*. Cambridge University Press, New York, NY, USA, 2014.