WWW.IJRPR.COM

International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

Predicting Information Diffusion by Learning Social Network Embedding

Mr. Abhishek Kesharwani[1], Dr. Samarendra Mohan Ghosh[2]

[1]Research Scholar, Department of CSE Dr. C.V. Raman University, Bilaspur (C.G.)
[2]Professor, Department of CSE, Dr. C.V. Raman University, Bilaspur (C.G.)

ABSTRACT:

The process on proximity structures or known graphs has been the main focus of investigation and modelling of the temporal distortion of information on social media. However, these models are unable to adequately capture the complexity of the underlying reality, because it is impossible to do so with such constrictive assumptions. It results from the interaction of various actors and media. Here, we propose a novel approach to find a mapping of the observed temporal dynamic onto a continuous space.

Information infiltration can be effectively modelled using a thermal diffusion process by projecting the nodes involved in intrusion cascades into a latent representation space. This basically entails learning a distortion kernel for which the proximity of nodes in the projection space matches the proximity of their infection time in cascades.

There are a number of distinctive features that set the suggested method apart from others. It does not rely on any pre-existing diffusion structure because its parameters are immediately learned from cascade samples without the need for any other information.

A heat diffusion approach is used to successfully model information intrusion by projecting nodes implicated in intrusion cascades onto a latent representation space. Essentially, this involves learning a distortion kernel for which the nodes' proximity in the projection space matches the nodes' infection time.

When compared to current approaches, the suggested one has a number of special qualities. It doesn't depend on any pre-existing diffusion structure because its parameters are immediately learned from cascade samples without the need for any other information. It does not depend on any prior knowledge because its parameters are immediately learned from cascade samples without the need for any other information. structure of the dissent. Compared to discrete models, the inference time for predicting the irruption of a new piece of information is significantly decreased since the solution to the irruption equation may be written in a closed form in the projection space. Both real datasets and synthetic networks have been used for experiments and comparisons with baselines and alternative models. They demonstrate the effectiveness of the suggested approach in terms of inference speed and prediction quality.

Categories and Subject Descriptors

I.2: Learning through Artificial Intelligence;

E.1: Data Structures Graphs and networks

Keywords - Machine learning; Information diffusion; Social networks

Introduction:

A lot of recent study has been spurred by the rise of social networks and social media websites. Diverse General tasks have been investigated, including Community Detection, Link Prediction, Social Network Analysis, and Social Network Annotation. The study of the temporal dissemination of information via this kind of media is a crucial area of research. It seeks to investigate how user behaviours, like posting a link on Facebook or retweeting something on Twitter, impact the dissemination of content like images, online gossip or videos. Although research on this word-of-mouth phenomenon predates the invention of computer science, the volume of information provided by the rise of online social networks has made new advancements possible and presented an unparalleled opportunity (for a survey of related studies, see Section 4). The majority of the early research in this field came from social science or epidemiology literature. Several propagation models have been created or modified for use with internet data, including the linear threshold models (LT) [13] and the independent cascade models (IC) [7, 22]. These models make an effort to anticipate and comprehend the observed propagation's dynamics. In recent years, research has also concentrated on prediction tasks like buzz detection, which determines if a specific piece of material will have a significant influence on the network [20], identifying opinion leaders, determining whether a network node will be crucial to the dissemination of content [13,17], or predicting which users (or network nodes) will be contacted in the future by a specific piece of information [19].

The probabilistic modelling of information diffusion based on explicit interactions between network nodes is the foundation of the majority of current propagation models. It has been demonstrated that while these models perform well when the underlying connection structure is typical of the filtration channels, they may have various issues when used to social networks that are taken from the Internet [19]. Due to their dependence on the network's structure, these models often seek to estimate the propagation probabilities along user-to-user linkages, assuming that information only spreads over these links. This makes it harder for them to explain future diffusion because the underlying mechanism is far more intricate:

Diverse interactions among diverse users on several interleaving networks lead to infringement. It is challenging to identify and measure user interactions themselves [25, 29]. Furthermore, in order to prevent over-hitting, these methods—which essentially involve learning propagation pathways via graph links—need a lot of observations.

Several recent studies have proposed that before modelling a diffusion with respect to the extracted network, the implicit structure of the diffusion should be discovered from user behaviours [29, 8, 25].

These studies employ the extracted graph to make predictions after first identifying the graph structure that best fits the observed distortion under a few hypotheses regarding the distortion mechanics. These methods are frequently based on a distinct cascading mechanism, meaning that data iteratively leaps from one user to another based on a few link transfer probabilities. When drawing deductions, such an iterative approach necessitates making successive decisions, which could result in low When errors are committed early in the diffusion process, performance suffers. Furthermore, inference by these models necessitates the application of costly Monte-Carlo simulation methods in order to forecast the dissemination of data.

In this study, we concentrate on modelling the diffusion of information in order to forecast which users, given the user at the time, will be contaminated by a specific piece of material. the origin of the misunderstanding. We propose working in continuous areas where we learn the temporal dynamics of diffusion from observations, as opposed to using a graph-based approach that would require dealing with discrete structures.

Our method is based on the heat diffusion theory and involves learning heat diffusion kernels that determine, given the original source of diffusion, the likelihood that each node in the network would be reached by the proposed information.

One benefit of this approach is that it is independent of a previous graph structure, and the model is directly constructed using diffusion cascades that were observed. Additionally, while working with novel cascades, the utilization of a continuous space representation enables extremely quick inference. This study offers three contributions:

- By embedding users in a continuous latent space, we offer a novel method for learning insertion processes.
- We suggest a model extension that would enable us to consider the type of information being presented, leading to differentiated discovery procedures that rely on the characteristics of the data being used.
- We use three Web-extracted corpora and generated datasets to compare this strategy with baseline and alternative approaches.
- This is how the paper is structured: Our general approach and the models we suggest are presented in Section 2, which also denotes the notations used throughout the work. In section three, we evaluate our models against a number of baselines using both artificial and actual datasets. The relevant research on the subject of diffusion in social networks is reviewed in Section 4. Section 5 brings our work to a close and offers some suggestions for potential future projects.

Sand Model:

The idea of cascade has historically been used to illustrate infringement on networks. A cascade is a series of users in influenced by certain data (for example, the list of people who enjoyed a particular YouTube video). While it's easy to determine when a user has been infected by a piece of content, it's typically impossible to determine who infected them. A cascade explains to whom and when an item spreads over the network, but it doesn't explain how infringement occurs.

Given a social network composed of a set of N users[1] μ = (u1…… un), cascades correspond to sets of users infected by the propagated information. Depending on the kind of network and the task in concern, the propagated information can for instance correspond to a given topic, a particular u rl, a special c tag, etc. In the following, we consider C to be a set of cascades over a given network, divided in two subsets of distinct cascades: C □ c t as: C the set of training cascades and Ct □ C the set of testing cascades. A cascade c € C is defined

- A source s^c € u which is the user at the source of the cascade- i.e, the first user that published the item concerned by the diffusion.
- A set of contaminated users S^c U such that ui S^c means that u^i has participated to the cascade c- i.e., the user has performed some action (retweet, like, comment...) that is considered as an infection by c (liking a video, publishing a tweet with a special c hashtag...); S^c is the set of users who have not participated in c.

- A contamination timestamp function defined over S^c such that $t^c(u_i)$ corresponds to the timestamp at which $u_i \in S^c$ has first participated in the cascade. We consider that the contamination timestamp of the source is equal to 0.
- A feature vector $q_c \in \mathbb{R}^Q$ that characterizes the content of the cascade c, with Q the size of the *content features space*[2]. This features vector is usually defined as the content of the publication.

Recommended Method:

It is the goal of the suggested models to forecast information division. The main concept of these models is to map the actual process of information fusion into a process of heat fusion in a continuous (euclidean) space. In order to do this, we acquire diffusion kernels that demonstrate the dynamics of diffusion from a set of training cascades. Let us denote Z = R^n an educlidean space of n dimensions- also called latent space[3].

Learning such a diffusion kernel comes down in our case to learning a mapping of each node of the network to a particular location in Z such that, for a given metric, the latent space explains the contamination timestamps observed in the training cascades. Figure 1 depicts a diffusion process where users have been projected in a latent space w.r.t. their timestamps of contamination in training cascades. This approach has several advantages:

- A continuous optimization problem that is easily resolved using traditional optimization techniques is translated to the learning problem.
- Without the use of a graph structure or presumptions about the propagation of information, the propagation model is learned directly from the observations.

1We talk about users throughout the paper, but the results remain valid for any other kinds of nodes.

2For example, when dealing with textual information, the feature vector qc may be a t f –id f vector.

3See Section 3 for a discussion concerning the choice of the dimension n.

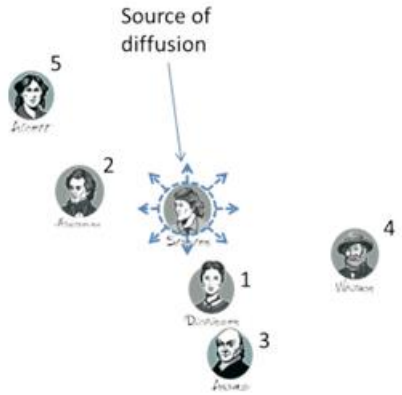

Figure 1: Diffusion in a latent space: every user has a position within this area. At the center of the image lies the source of the diffusion, from which information spreads uniformly in all directions. The figures represent the order of contamination among the various users based on the modelled diffusion process: the closer a user is to the source, the more quickly he becomes infected by information from the source.

- Because the inference of the idiom may be carried out in continuous space, the prediction can be computed very quickly without the use of an explicit discrete simulation. In addition to having a high processing cost, simulation can produce incorrect findings, with results from multiple simulations of the same diffusion showing significant variation.
- Lastly, by making the geometry of the latent space reliant on the information that spreads, the method makes it simple to integrate the content information.

Diffusion Kernel. Let us consider a geometric manifold X.

We de ne heat diffusion as a function f(xt) : X * R^+ R where f(xt) is the heat at location x at time t.

Such a process can be described by the following heat equations:

$$\begin{cases} \frac{\partial f}{\partial t} - \Delta f = 0 \\ f(x,0) = f_0(x) \end{cases} \quad (1)$$

where the process's beginning condition is denoted by f0(x); (1) is the operator for Laplace. In a particular domain with suitable boundary conditions, the fundamental solution to these heat diffusion equations is known as the heat diffusion kernel [11].

In order for K: R+ X X R to calculate the heat at location x and time t, knowing that the heat source is y, we define a diffusion kernel K(t yx).

It simulates the heat diffusion at time t = 0 when an initial unit of heat is placed at location y.

This beginning state is equivalent to:

$$K(0, y, x) = \delta(y - x) \quad (2)$$

where δ is the *dirac function*. In an Euclidean space of n dimensions, the diffusion kernel can be written as:

$$K(t, y, x) = (4\pi t)^{-\frac{n}{2}} e^{-\frac{||y-x||^2}{4t}} \quad (3)$$

The foundation of our methodology is the diffusion kernel, which will be utilized to simulate the flow of information between network nodes.

Latent Space Diffusion Kernel Learning. Although certain studies examine particular instances of heat intrusion according to the network's structure [14,17], we wish our model must be separate from any explicit network that has been predefined. To do this, we suggest employing kernels as defined in equation 3 to represent the information propagation in a Euclidean space and directly learning these kernels from observed cascades. Therefore, learning a representation of nodes in this latent space that enables the fusion kernel to account for the cascades seen in the training set is the aim. This may be considered an issue in figuring out the best diffusion.

rewrite the diffusion kernel as a function K(t s^c u_i) which returns a value corresponding to the contamination score of node u_i at time t knowing that the source of the contamination- the initial contaminated node- is s^c. We de ne Z = (z_{u1}…… z_{uN}) such that z_{ui} R^n denotes the location of user u_i in the latent space R^n.

The obtained diffusion kernel is:

$$K_Z(t, s^c, u_i) = (4\pi t)^{-\frac{n}{2}} e^{-\frac{||z_{s^c} - z_{u_i}||^2}{4t}} \quad (4)$$

The problem of modelling how information propagates corresponds to finding the optimal value of Z according to every cascade c € C. The empirical risk of the model is then defined as:

$$\mathcal{L}(Z) = \sum_{c \in C_l} \Delta(K_Z(., s^c, .), c) \quad (5)$$

where $\Delta(K_Z(., s^c, .), c)$ is a measure of how much, given a source s^c, predictions performed by the diffusion kernel K_Z differs from the observed diffusion in c. Different Δ functions can be defined and we focus on a particular case based on a ranking measure. The final learning problem is an optimization problem which consists in finding Z^* such that:

$$Z^* = argmin_Z \mathcal{L}(Z) \quad (6)$$

The problem of learning diffusion as a ranking problem

The contamination tendency is modelled by the distortion kernel provided a certain information source, of any node at time t. However, there is no complete support for learning the kernel function, which would translate into a continuous time function that shows the evolution of heat at any given place. Only the contamination times of the various nodes in a cascade are provided by the observations. The kernel will be restricted from contaminating the various nodes in their true temporal order of infection by means of this partial supervision.

The following limitations will be applied in practice:

- Given two nodes u_i and u_j such that u_i and u_j are con terminated during cascade c- I .e u_i S^c and u_j S^c- and respecting $t^c(_{ui}) < t^c(u_j)$, KZ must defined such that ∀t,KZ(t s^c u_i) > KZ(t, s^c ,u_j)
- Given two nodes u_i and u_j and a cascade c such that u_i is in S^c and u_j not in S^c, KZ must be defined such that∀t KZ(t, $s^{c,}$, u_i) > KZ(t, $s^{c,}$,u_j)

The constraints basically aim at finding embedding's such that users who are contaminated first are closer to the source of the contamination than users contaminated later (or not contaminated at all). With the expression of KZ given in equation 4,we can easily rewrite these two constraints as:

$$\forall (u_i, u_j) \in S^c \times S^c,$$
$$t^c(u_i) < t^c(u_j) \Rightarrow ||z_{s^c} - z_{u_i}||^2 < ||z_{s^c} - z_{u_j}||^2$$
$$\forall (u_i, u_j) \in S^c \times \bar{S^c},$$
$$||z_{s^c} - z_{u_i}||^2 < ||z_{s^c} - z_{u_j}||^2 \quad (7)$$

By the use of classical hinge loss functions, these constraints can be handled by defining a ranking objective Δ_{rank} such as:

$$\Delta_{rank}(K_Z(., s^c, .), c) = \sum_{\substack{u_i, u_j \in S^c \times S^c \\ t^c(u_i) < t^c(u_j)}} max(0, 1 - (||z_{s^c} - z_{u_j}||^2 - ||z_{s^c} - z_{u_i}||^2)) + \sum_{u_i, u_j \in S^c \times \bar{S^c}} max(0, 1 - (||z_{s^c} - z_{u_j}||^2 - ||z_{s^c} - z_{u_i}||^2)) \quad (8)$$

Algorithm for Learning

The primary training goal is

$$\mathcal{L}_{rank}(Z) = \sum_{c \in \mathcal{C}_\ell} \Delta_{rank}(K_Z(., s^c, .), c) \quad (9)$$

This model is known as Content Diffusion Kernel (CDK). Various techniques can be employed to maximize the objective function. Using a classical stochastic gradient, we offer Algorithm 1 illustrates the smell technique, which iterates until a stop criterion is met (usually a number of iterations without a significant improvement in the global loss). Once all user embedding's in U have been randomly initialized (line 3), the algorithms sample a cascade from the training set C and two users, u_i and u_j, with u_j being a user that is either non-infected or contaminated after u_i in the diffusion process described by cascade (lines 6to8). When the restrictions of defined in equation 7 are not adhered to with a sufficient margin [5] for this cascade and the pair of users u_i and u_j, embedding's u_i, z_{uj}, and z s^c are modified toward their respective steepest gradient directions with a learning rate (lines13 to15) that decreases as the number of iterations increases.

Complexity of inference and learning:

Let T be the number of iterations. The learning complexity is O (T* n), where n is the size of the latent space. Once Z has been learned, the inference process is simple. Fora cascade c € Ct, we just compute the distance between the user s^c and very other user in U. The inference complexity fore very cascade is then O (N* n), where Nis the number of users. Considering that n<< N, this turn out to be much smaller than the complex it y of most alternative discrete methods. For instance, the inference step of the very famous Independent Cascade model 6, which is a probabilistic model where diffusion probabilities are defind on edges of the networks graph, requires to consider at each time.[4] Various methods of initialization can be used. In our tests, we employed a uniform initialization between -1 and 1.

Section 3 presents the details and findings of this model. [5] As determined by the hinge loss function, see equation 8.

Algorithm 1 Stochastic gradient descent algorithm

1: **procedure** SGD RANK DIFFUSION KERNEL LEARNING($\mathcal{U}, \mathcal{C}_\ell, T$)
2: $\tau \leftarrow 0$
3: $\forall u \in \mathcal{U}, z_u^{(\tau)} \leftarrow random$
4: **while** $\tau < T$ **do**
5: $Z^{(\tau+1)} \leftarrow Z^{(\tau)}$
6: Sample $c \in \mathcal{C}_\ell$
7: Sample $u_i \in S^c$
8: Sample $u_j \in \mathcal{U}$ with $t^c(u_i) < t^c(u_j)$ or $u_j \in \bar{S^c}$
9: $d_i \leftarrow ||z_{s^c}^{(\tau)} - z_{u_i}^{(\tau)}||^2$
10: $d_j \leftarrow ||z_{s^c}^{(\tau)} - z_{u_j}^{(\tau)}||^2$
11: $\delta_d \leftarrow d_j - d_i$
12: **if** $\delta_d < 1$ **then**

$z_{u_i}^{(\tau+1)} \leftarrow z_{u_i}^{(\tau)} + \alpha(\tau) \times 2(z_{s^c}^{(\tau)} - z_{u_i}^{(\tau)})$
$z_{u_j}^{(\tau+1)} \leftarrow z_{u_j}^{(\tau)} + \alpha(\tau) \times 2(z_{u_j}^{(\tau)} - z_{s^c}^{(\tau)})$
$z_{s^c}^{(\tau+1)} \leftarrow z_{s^c}^{(\tau)} + \alpha(\tau) \times 2(z_{u_i}^{(\tau)} - z_{u_j}^{(\tau)})$
end if
$\tau \leftarrow \tau + 1$
end while
$Z \leftarrow Z^{(\tau)}$
end procedure

All potential infection situations at time 1, which rapidly becomes unmanageable. In actuality, graphical model inference is carried out by using. A Monte-Carlo approximation that involves simulating the diffusion process extensively, starting at the cascade's source and tracking the diffusion probabilities on the graph's links. The inference complexity of this approximation of IC is $O(r * \text{Succs} * \frac{}{Sc}/)$, where /$S^C$ / Succs is their average out degree, and is the average number of infected nodes in the simulations that were run. And r is the quantity of simulations that are employed in the MCMC approximation.

A more accurate approximation of the distribution of nodes infection probability must be obtained if the probabilities of nedon links are weaker. In Section 3, more details regarding computation times are provided.

Diffusion Kernel Based on Content

The content of each cascade can now be taken into consideration by considering an extension of the preceding model.

It will spread differently throughout the network. Therefore, we aim to describe various dissemination techniques based on the information's substance and source. We base our expansion on the following concepts:

(i) To begin, the device kernel will continue to modify the propagation in the alatent space, with each user co-corresponding to a specific place.

(ii) Secondly, the content will affect the latent space's metric and prevent it from being isotropic around the source.

The content will also determine how it propagates in the latent space. Each potential piece of material will therefore correlate to a certain latent space metric, leading to distinct propagation strategies. Users' geolocation and the metrics will be simultaneously learned through training cascades. This work has established the content measure in a way that allows the content to function as an ectingthe l.

The source's position within the latent space7 The following describes this model, which is called the Content-based Source Diffusion Kernel (CSDK).

We consider a parameterized function called content embedding function and denoted

$f\theta : R^Q \rightarrow R^n$. It will map any content information into a particular location in the la tent space θ, being the set of parameters of this function8. The function will map two different contents q_c and q_c to two different locations f θ (q_c) and f θ (q_c) in the latent space as illustrated in

Figure 2. Let us de ne the new diffusion kernel as a function $K^{CSDK}{}_{Z\theta}$ (q^c ,t, s^c, u_i)which measures the contamination of user u_i at time t knowing that the source of the diffusion is, s^c and the content of the cascade is q^c £ R^Q. In order to consider both the source of the contamination and the content that di uses, based on the content embed ding function f θ , we propose to model K^{CSDK} such that:

$$K_{Z,\theta}^{CSDK}(q^c, t, s^c, u_i) = (4\pi t)^{-\frac{n}{2}} e^{-\frac{||z_{s^c} + f_\theta(q^c) - z_{u_i}||^2}{4t}} \qquad (10)$$

The location of the source z_{s^c} + fθ (q_c) now depends on both the latent representation of the source user s^c and on the embedded content f θ(q_c).

Thus, two different contents will match two different starting points, producing two different insertion kernels (see Figure 2).

The learning task will be to minimize on both θ and Z the following objective function since the content embedding function and the user locations will be learned concurrently:

$$\mathcal{L}_{CSDK}(Z) = \sum_{c \in \mathcal{C}_\ell} \Delta_{rank}(K_{Z,\theta}^{CSDK}(q^c, ., s^c, .), c) \qquad (11)$$

The final learning problem can thus be written as:

$$\Delta_{rank}(K_{Z,\theta}^{CSDK}(q^c, ., s^c, .), c) = \sum_{\substack{u_i \in S^c \\ u_j \in S^c \\ t^c(u_i) < t^c(u_j)}} max(0, 1 - (||z_{s^c} + f_\theta(q^c) - z_{u_j}||^2 - ||z_{s^c} + f_\theta(q^c) - z_{u_i}||^2))$$

$$+ \sum_{\substack{u_i \in S^c \\ u_j \in \bar{S}^c}} max(0, 1 - (||z_{s^c} + f_\theta(q^c) - z_{u_j}||^2 - ||z_{s^c} + f_\theta(q^c) - z_{u_i}||^2)) \quad (12)$$

It is optimized by a stochastic gradient descent technique akin to the one previously described.

Experiments

Data Sources

Our models were evaluated using multiple datasets from different sources, both fake and online.

Actual datasets:

Three Web-extracted datasets have been used:

The first dataset is from the AAAI International .The 2009 Conference on Weblogs and Social Media (ICWSM) released a corpus of 44 million blog entries gathered over a one-year span [4]. We think about

7 options were examined, and this one provided a decent middle ground.

8 Here, f θ is regarded as a linear function.

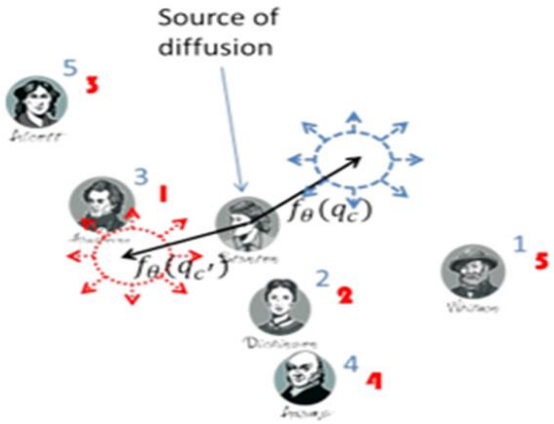

Figure 2: Diffusion model of CDSK: Diffusion originates from a translation of the source user by a vector f θ (q_c), which is contingent upon the case's content. Cade. Here, the source user has two distinct contents, q_c and q_c spread, which correlate to two distinct embedded locations, f θ (q_c) and f θ(q_c).

These two cascades consequently correspond to two different source locations in the latent space, even if they share a common source.

The order of contamination by the two contents is indicated by the two numbers next to each user.

- Every blog should be a user on the social network, and cascades are made up of collections of related posts: each linked post component .The graph is a cascade. The writers of the posts that make up a cascade are represented by a collection of users, and the timestamps of the posts that they were infected with serve as a representation of the cascade. Moreover, we used hyperlinks to extract an oriented graph: we generate a link b an if there is at least one link in the training set between a post by user a and a post by user b. Our models do not require this graph, but some of the baseline models that were employed in the tests do.
- The Meme tracker corpus mentioned in [16] is where the second dataset was taken from. The articles in this corpus were gathered from blogs and news websites during the

  U.S. presidential campaign, 2008. Building the corpus involves tracking the flow of brief phrases or memes across the internet. The users, cascades, and social graph have been defined similarly in this dataset as in ICWSM. Since there is a significant difference in the content of the posts, we did not use the CSDK model on this dataset.
- Digg, a collaborative news platform where users can add links to stories (articles, blog posts, videos, etc.), is where the third dataset was taken from. Other users can then, if they're interested, dig these stories. Depending on how many diggs they have, stories appear on Digg's front page. Every user-provided digg is regarded as a user examination, and we use tales as cascades. We gathered the whole Digg history—every story published, every digg, and every comment—over the course of a month using the Digg stream API. We constructed a graph in the manner described below, which will be utilized by the IC and Graph Diffusion baselines:

For each user a who has digged a post created by user b, we create a link b → a in the graph. We filtered the users of each dataset to keep about 5000 users with the most posts. Table 1 gives some statistics about the datasets sizes.

Synthetic Information.:

To have a better understanding of how the content information is handled by the CSDK model, we have also produced synthetic datasets for which we have control over the relationship between the ambiguity behaviour and the content information.

We assume that any cascade content is made up of one word (w) out of a set of Q potential words in order to create such datasets. The diffusion follows a specific IC model for each word, represented by the symbol E_w, where E_w is the transition matrix between nodes, representing the likelihood of the diffusion. 99 percent of the values in the Q transitions matrices are equal to 0, and all remaining values are created at random.

To create sparse matrices, values are sampled from 0 to 001. In order to create a new cascade, we first select a source user (u) and a content word (w) at random, then use the cascade is generated using the matching transition matrix E_w. Increasing the value of Q results in more intricate propagation systems. The generating mechanism we use is equivalent to a classical single IC model if Q = 1. With a set of 1000 users, we employed 10,000 cascades for training and 10,000 cascades for testing in these datasets.

Measures of assessment

For training and testing, the cascade set C is split into two subsets, C and Ct, for every dataset. We want to determine which user or users will eventually become infected for each cascade in the testing set. With users representing documents and cascades representing questions, this can be viewed as an information retrieval problem. Mean Average Precision (MAP) and Precision-Recall curves are used to assess the performance. Due to Every model forecasts a contamination score for every user in the testing set, indicating the likelihood that the user will become infected by each cascade. Our then utilize Mean Average Precision to assess our success, as done in [6], after sorting users in descending order. Assume that c_k is the user u_k rank for cascade c. Percentage of infected users among the top k users in the ranking order, or $P_{c,k}$, is the precision at rank k for cascade c.

Average accuracy is defined as follows:

$$MAP = \frac{1}{|\mathcal{C}_t|} \sum_{c \in \mathcal{C}_t} \frac{\sum_{u_k \in S^c} P_{c,\sigma_{c,k}}}{|S^c|}$$

Additionally, we visualize performances using Precision-Recall curves.

A limited amount of recall points are produced by cascades that often only reach one or two people. After that, we display the average accuracy rather than precision at a recall value, but rather at each recall point. Thus, there is a greater variance for high recall points and only a small number of cascades have many users. Keep in mind that every experiment was conducted ten times, and the findings represent the average of the ten separate runs.

Initial Conditions:

We evaluate our models against a number of benchmarks and cutting-edge models. First, two naive baselines were used:

Nb-App: We calculate each user's percentage of cascades in the training set that he encounters. We're In each cascade in the testing set, utilize that value as the user's infection score. It is equivalent to a tendency to contract any cascade infection.

Mean- rank: According to this model, the contamination score of a user interface (UI) is the inverse of the user's average rank in the training cascades. This means that the higher the user's infection score for cascades from the testing set, the sooner the user is likely to be infected by cascades in the training set.

We also contrast our methods with the most advanced models in addition to these baselines:

- **IC model**: We applied the traditional independent cascade model (IC), which is frequently employed as a comparative technique in literature. IC makes use of the community graph, and functions in a discrete fashion: each neighbour u_j of a user u_i who contracts the infection at time step t has a chance $p_{i,j}$ of contracting the infection at time step t+1. Every $p_{I\,j}$ is learned in an EM-fashion throughout the learning phase [22]. It is challenging to calculate the precise likelihood that a user may contract the infection at some stage because this is a stochastic model. Thus, we employ a Monte-Carlo approximation, in which numerous simulations are performed given an initially infected person.

The number of simulations in which a user became infected is equivalent to his final contamination score. Take note. Temporal extensions of the IC model have been suggested, but our investigations did not use them because they did not improve the results of this experimental technique [15].

- **Netrate:** We present findings for the Net Rate model's exponential variant, which is detailed in [8]. Here, Net Rate is utilized as a cutting-edge technique that does not require an understanding of the network architecture to forecast the dissemination of information, which is also true of our methods. Be aware that there are several variations of this model (power law and rayleigh-[8]) that provide comparable outcomes. Section 4 provides a detailed description of Net Rate.

- **Graph Diffusion**: Finally, we contrast our approaches with the model put forth in [17,11], which is predicated on a graph diffusion similar to our own. The authors of the paper design a specific kernel over the network structure in this model rather than discovering the optimal kernel like we do. The used equally on the various outbound links, as this kernel models. In contrast to the illusion model in our method is obviously dependent on the knowledge of the network structure and is not learned over the set of training cascades. Because they require a user profile, which is not present in our datasets, we do not compare to models like [15] or [23], which also use the content.

Findings

Contentless models:

The mean average precision (MAP) for each is displayed in Table 2. models using the three authentic datasets. First, it is evident that the IC model outperforms all other baseline models.

Model	n	Memetracker	ICWSM	Digg
CDK	5	0,176	0,660	0.170
	10	0.257	0.721	0.212
	30	0.344	0.769	0.273
	50	0.355	0.774	0.285
	100	0.347	0.771	0.282
	200	0.357	**0.776**	0.302
	500	0.363	0.773	0.280
CSDK	5	-	0.605	0.255
	10	-	0.663	0.304
	30	-	0.714	0.348
	50	-	0.731	**0.352**
	100	-	0.744	0.352
	200	-	0.732	0.350
	500	-	0.748	0.351
IC		0.372	0.712	0.197
Netrate		0.287	0.187	0.162
Graph Diff.		**0.374**	0.483	0.082
Nb_App		0.180	0.112	0.077
Mean_Rank		0.187	0.121	0.206

Table 2: Results on 3 real datasets: *Memetracker*,

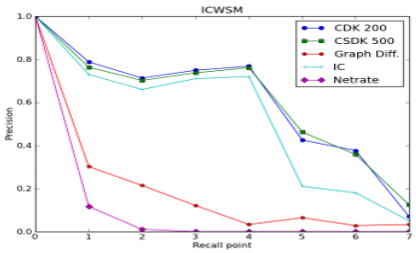

Figure 3: Precision at recall points for the main models on the ICWSM dataset. Results of CDK and CSDK are given for specific latent space dimensionality.

	Nb. of Users	Nb. of Links	Nb. of train Cascades	Nb. of test Cascades	Avg cascade size
meme	5000	4372	2377	600	2.17
icwsm	5000	17746	19027	4711	2.22
digg	4751	71263	150000	66744	2.43

Table 1: Some statistics about our real datasets.

Digg and ICWSM. The CDK and CSDK results are provided for a range of values of n, the latent space Z's dimension.

Given that Mean Rank and Nb App are based on naive heuristics, this is hardly shocking. The Net Rate model's poor performance results from the learning process. When used to forecast information diffusion, it depends on too much training data to prevent over-training. Ultimately, the Graph Diffusion technique performs better than IC on the Meme Tracker dataset but worse on the other two corpora, indicating that the underlying assumptions are less applicable to the three datasets than the IC assumption on the dissemination of information.

On the Meme tracker and ICWSM datasets, our method (CDK) and IC produce comparable results; on the Digg dataset, however, CDK performs noticeably better than IC. database. Please take note that, similar to Net Rate, our approach is not predicated on any understanding of the network's structure, which means that it can perform on par with or better than IC while utilizing less data. The size of the latent space also affects the CDK model's performance; a larger space may tend to over it, while a smaller area produces lower-quality predictions. The CDK model requires around 15 minutes to infer all scores for every cascade on the Digg dataset. Comparatively, the IC model requires more than one day, and the Net Rate model, which is a few days longer, is one level more sophisticated than IC. A typical desktop computer has been used for all of these tests.

Integration of content:

The Content-based approach (CSDK) performs worse than the other models when the content is integrated (Table 2). Compared to the Digg dataset, the ICWSM dataset clearly produces better predictions. The quality of this model is really determined by the information provided by the content of each cascade; in the case of ICWSM, the content is noisy due to the way it was captured (using RSS feeds that only provide a portion of the blog posts' content), whereas the Digg content is clearly more informative because it contains the entire content of news articles.

We have conducted studies on synthetic datasets with various features to further investigate and comprehend how the CSDK model depends on the caliber of the material.

Amounts of material (see Table 3). It is evident that when the variation of the content (the number of words taken into consideration) rises, the performance of all the algorithms deteriorates. The more content information that is taken into consideration, the more complicated the work becomes. Less CDSK deteriorates than all alternative methods and still achieves good performance. It consistently outperforms cutting-edge techniques. As demonstrated by these tests, CSDK is more resilient to a more intricate propagation system that is depending on content than traditional methods. The precision of each model at various recall points is finally shown in Figures 3 and 4, where we have only included the best CDK and CSDK versions. CSDK performs noticeably better than CDK on the Digg dataset, and it is evident from these curves that our methods achieve more precision than classical methods.

Model	n	5 words	10 words	20 words	30 words	40 words	50 words
CDK	10	0.323	0.205	0.147	0.111	0.102	0.098
	30	0.422	0.301	0.207	0.146	0.128	0.121
	50	0.414	0.304	0.207	0.158	0.136	0.128
	100	0.430	0.304	0.210	0.155	0.140	0.126
CSDK	10	0.394	0.243	0.184	0.139	0.135	0.124
	30	0.605	0.442	0.301	0.218	0.200	0.179
	50	0.615	0.466	**0.346**	0.259	0.234	0.219
	100	**0.631**	**0.469**	0.343	**0.271**	**0.248**	**0.228**
IC		0.482	0.317	0.211	0.163	0.125	0.111
Netrate		0.289	0.150	0.175	0.137	0.017	0.017
Graph Diff.		0.308	0.091	0.081	0.084	0.073	0.076
Nb_App		0.118	0.101	0.088	0.085	0.079	0.081
Mean_Rank		0.209	0.196	0.165	0.160	0.151	0.143

Table 3: Results (MAP values) of our models and baselines n artificial datasets generated with different number of words. Results of CDK and CSDK are given for several values of n, the dimension of the latent space $\mathcal{Z}$.

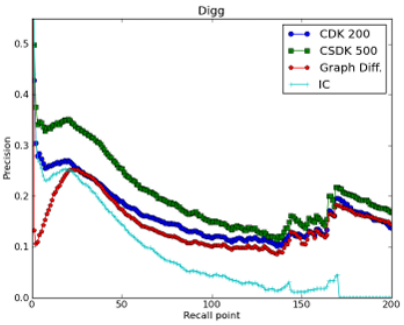

Figure 4: Precision at recall points for the main models on the Digg dataset. Results of CDK and CSDK are given for specific latent space dimensionality.

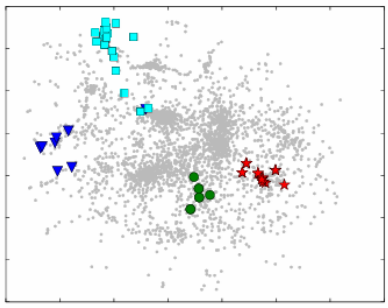

Figure 5: The Digg dataset users in 2D. Light gray dots represent users, and groups of identical symbols represent four cascades from the testing set.

The Conversation:

Despite the fact that knowledge dissemination in social networks is a thoroughly studied topic (see the next section), the concept has not been adequately defined. These phenomena include the following: a comparatively small percentage of the daily content created online will become well-known and become viral. Table 1 shows that the majority of cascades in our datasets only reach a very small percentage of users. This sparsity makes it rather challenging to learn the dynamics of user sessions. In order to show the diffusion with distance, we project users onto a Euclidean space in this study.

This give sour CDK model an important property: forany triplet of users (u_i, u_j, u_k),we have the triangle inequality $||z_i z_j||^2 \leq ||z_i, z_k||^2 ||+ z_k, z_j||^2$.This indicates that if users U_I and U_{IJ} never communicate with one another during training but instead engage with a third user, our model tends to set u_i and u_j to be quite close. It is impossible for a model like IC to learn such a characteristic. We have trained a CDK model with a latent space of size to better visualize this capacity to regroup people with similar behaviours.

n=2 on the Digg corpus and suggest displaying the way users have been projected onto this area (Figure 5). We have highlighted the users who were selected at random. Sen cascades are taken from the test set. This figure indicates that our model has a natural tendency to form clusters, each of which represents a group of users who are typically involved in the same cascades. We hypothesize that the ability to group users who exhibit similar behaviour opens the door to additional uses, especially with regard to the employment of CDK and CSDK for visualization applications that could enable people to comprehend the various dissemination strategies for a specific dataset. Using a distance also meant that we consider to be asymmetric in the CDK model, Diffusion from u_i to u_j is the same as diffusion from u_j to u_i, for example. This is a compelling theory that has been the subject of recent discussion [3]. We are now testing a version of this work in which users—whether senders or recipients of the content being propagated—are projected at different points in space.

Connectedly:

The fusion process has previously been examined in relation to product acceptance in [1]. Throughout this piece, the a_u uses two elements to model consumer adoption of a product over time: the impact of word-of-mouth and the magnitude of a marketing effort. The availability of vast amounts of internet data in the early 2000s allowed researchers to propose social graph-based techniques such as the Independent Cascade model (IC) [7, 22] and the Linear Threshold models (LT) [13], which both represent a process of user-to-user contamination. Since then, numerous additional developments have been brought about by the explosive rise of social media platforms like Facebook and Twitter.

A number of IC and LT extensions have been suggested. An asynchronous extension of the IC model (ASIC), for example, was presented by [21] and allows IC to incorporate the temporal size. In [17,11], user interactions are modelled using a heat diffusion process that takes place on the social graph. Many cascades cannot be described by user interactions alone, as this paper has shown. To determine the likelihood of a diffusion, the writers in [23] consider user proles. [12] or [15] infer the contamination by using user proles and information content.

- The same concept has been applied in [26], which predicts the likelihood of user dissent by integrating tweet content. It is important to remember that all of these models assume that the graph on which the propagation takes place is known.

  This turns out to be a compelling hypothesis: the social graph that an online social network defines (friends, followers, subscriptions, etc.) is frequently unidentified, irrelevant, or incomplete [25].

  Two primary families of approaches have been researched in order to get around this restriction.

Link prediction techniques comprise the first family. Given a population of online social networks and a collection of observations (such as hash tags, movie reviews, and purchasing patterns, usage...), the objective is to identify a group of connections (friends, followers, influencers, etc.) that most accurately describe the actions of the targeted person.

- All of these models model the propagation process to infer the most likely linkages, but none of them have been specifically designed for diction prediction. Using a greedy algorithm, Net Inf [9] and Connie [18] find a fixed number of ties between users that maximize the possibility of a collection of information observed diffusions on the hypothesis of an IC-like diffusion. In [8], a broader framework was put forward, and the Net rate model—which we employed in our experiment—ments as a starting point, is employed to forecast the contamination between users. Similar to IC, Net rate is a cascade model that seeks to determine the transmission probabilities between user pairs. The first advancement in IC is that they directly estimate probabilities from observable deviations rather than relying on the social network. Second, they infer the period after which the occurrence of the diffusion takes place using an exponential delay. Later, these pieces were expanded upon in [10]. Recently, [25] calculated the inter-user influence and deduced a graph with the most predicted relationships using transfer entropy.
- The second family use statistical learning rather than graph-based methods. Studying the relationship between is one straightforward but effective technique. the quantity of users who have contracted the infection both quickly and over an extended length of time [24]. [29] estimates the

volume of intrusion according to a chosen group of users' infection duration.

The models that we have presented in this article don't require are founded on a novel method that models the propagation as a heat diffusion process in a continuous latent space, regardless of the social network. Recent research has examined heat disruption methods, and in particular, fusion kernels, for a variety of purposes, including classification [14], dimensionality reduction [27], and node ranking [28]. The study in [17], where the authors choose marketing prospects using a diction kernel, is the one that most closely resembles ours.

Conclusion:

On the basis of the heat diffusion kernel, we have introduced a new family of information diffusion models. Their distinctiveness aims to use an embedding of nodes discovered from observed cascades to formulate illusion as a process in a continuous space. Some intriguing features of these models

1) They do not need a predefined derived network structure, which is frequently unavailable for social Applications.
2) As the observations provide them with first thand knowledge. Their operation is one or two orders of magnitude faster than that of classical discrete models due to the continuous environment.
3) Their modification of the geometry of the latent space facilitates the integration of the material. Results from both synthetic and real-world datasets show how well these methods can replicate information diffusion and take into consideration the diffusion process, including content information. They sometimes outperform the most advanced reference models, but they also compete with them.

At the moment, two research avenues are being contemplated. The first involves creating alternative models to make greater use of the content data. In particular, we are looking at metric learning paradigms that should open up new ways to integrate this data into the latent space's geometry. Using these techniques for diffusion challenges that are not limited to social networks is the second direction.

Reference:

[1] J. Leskovec and S. A. Myers. Regarding latent social network inference's convexity. 2010; CoRR, abs/1010.5504.

[2] L. Denoyer, P. Gallinari, and A. Najar. utilizing incomplete knowledge to forecast the spread of information on social networks. WWW 12 Companion, Proceedings of the 21st International Conference Companion on the World Wide Web, New York, NY, USA, 2012, pp. 11971204. ACM.

[3] J. Kleinberg, B. Meeder, and D. M. Romero. Idioms, political hashtags, and intricate Twitter contagion are examples of how information spreads differently depending on the topic. Pages 695704 in Proceedings of the 20th International Conference on the World Wide Web. ACM (2011)

[4] H. Motoda, K. Saito, M. Kimura, and K. Ohara. generative models of asynchronous time-delayed information dispersion. Proceedings Track, Journal of Machine Learning Research, 13:193208, 2010.

[5] K. Saito, R. Nakano, and M. Kimura. Prediction of information diffusion probabilities for independent cascade model. In Proceedings of the 12th international conference on Knowledge-Based Intelligent Information and Engineering Systems, Part III, KES 08, pages 6775. Springer-Verlag, 2008.

[6] H. Motoda, M. Kimura, Y. Yamagishi, K. Saito, and K. Ohara. Diffusion probability in social networks is learned from node properties. A. Skowron, M. Kryszkiewicz, H. Rybinski, and Z. W. Ras, editors, ISMIS, Lecture Notes in Computer Science, number 6804, pages 153162. 2011 Springer

[7] Huberman, B. A. and Szabo, G. estimating the level of popularity of content found online. 2010; ACM Communications, 53(8):8088.

[8] A. Galstyan and G. Ver Steeg. content-dynamics-based information-theoretic metrics of influence. New York, NY, USA, 2013, Proceedings of the sixth ACM international conference on Web search and data mining, WSDM 13, pages 312. ACM.

[9] J. E. Hopcroft, L. Wang, and S. Ermon. Probabilistic models for inferring diffusion networks with features augmented. Volume Part II, Proceedings of the 2012 European Conference on Machine Learning and Knowledge Discovery in Databases, ECML PKDD12, pages 499514. Verlag, Springer, 2012.

[10] Weinberger, K. Q., Saul, L. K., and Sha, F. For nonlinear dimensionality reduction, a kernel matrix is learned. The twenty-first international conference on machine learning proceedings, page 106. 2004's ACM.

[11] M. R. Lyu, I. King, and H. Yang. Diffusion rank: a potential remedy for online spam. Pages 431438 in Proceedings of the 30th annual international ACM SIGIR conference on Information Retrieval Research and Development. ACM, 2007.

[12] J. Leskovec and J. Yang. modeling implicit networks' information diffusion. 599608, Washington, DC, USA, 2010, Proceedings of the 2010 IEEE International Conference on Data Mining, ICDM 10. IEEE Computer Society.

[13] A. Sharma, K. Munagala, A. Goel, and R. Bosagh Zadeh. Regarding the accuracy of information and social networks. Pages 6374 of the Proceedings of the First ACM Conference on Online Social Networks. ACM (2013).

[14] A. Java, I. Soboro, and K. Burton. The spinn3r dataset from ICWSM 2009. May 2009, Proceedings of the Third Annual Conference on Social Media and Weblogs.

[15] X. Tang, M. Chen, and Q. Yang. directed embedding of graphs. Pages 27072712, IJCAI, 2007.

[16] Feng W. and Wang J. Personalized tweet re-ranking: should I retweet? Proceedings of WSDM 13, ACM, 2013, the sixth ACM international conference on Web search and data mining.

[17] E. Muller, B. Libai, and J. Goldenberg. Discuss the network: A complicated system examines the fundamental mechanism of word-of-mouth. Letters on Marketing, 12(3):211223, 2001.

[18] B. Scholkopf, D. Balduzzi, and M. Gomez-Rodriguez. revealing the diffusion networks' temporal dynamics. ICML 11, pages 561568, Proceedings of the 28th International Conference on Machine Learning (ICML-11). ACM, 2011.

[19] A. Krause, J. Leskovec, and M. Gomez Rodriguez. determining impact and influence networks. KDD 10, New York, NY, USA, 2010; Proceedings of the 16th ACM SIGKDD international conference on knowledge discovery and data mining. ACM.

[20] B. Scholkopf, J. Leskovec, and M. Gomez-Rodriguez. utilizing survival theory to model the spread of knowledge. In ICML (2013)

[21] Heat Kernel and Analysis on Manifolds, A. Grigoryan. AMS/IP Advanced Mathematics Studies. Mathematical Society of America, 2009.

[22] H. Hacid and A. Guille. The temporal dynamics of information diffusion in online social networks can be predicted using this model. In WWW 12 Companion, Proceedings of the 21st International Conference Companion on the World Wide Web, ACM, 2012.

[23] E. Tardos, J. Kleinberg, and D. Kempe. making the most of a social network's ability to distribute influence. KDD 03, pages 137146, Proceedings of the 9th ACM SIGKDD international conference on knowledge discovery and data mining. ACM (2003).

[24] J. Laerty and R. I. Kondor. Diffusion kernels for discrete input spaces, such as graphs. Volume 2, pages 315322, ICML, 2002.

[25] E. Gaussier, P. Gallinari, L. Denoyer, and C. Lagnier. utilizing user profiles and content to forecast the spread of information in social networks. ECIR 13, 2013, European Conference on Information Retrieval.

[26] L. Backstrom, J. Kleinberg, and J. Leskovec. The dynamics of the news cycle and meme monitoring. KDD 09, pages 497506, New York, NY, USA, 2009; Proceedings of the 15th ACM SIGKDD international conference on knowledge discovery and data mining. ACM

[27] I. King, M. R. Lyu, H. Ma, and H. Yang. Heat diffusion methods are used in social network mining to pick marketing candidates. New York, NY, USA, 2008; Proceedings of the 17th ACM Conference on Information and Knowledge Management, CIKM 08, pages 233242. ACM