Predicting Information Diffusion and Content Engagement in Social Networks

Samhita Karnati

Advisor: Professor Andrea LaPaugh

Submitted in partial fulfillment

of the requirements for the degree of

Bachelor of Science in Engineering

Department of Computer Science

Princeton University

June 2018

I hereby declare that I am the sole author of this thesis.

I pledge my honor that I have not violated the Honor Code in the completion of this thesis.

_____

Samhita Karnati

Abstract

Contents

Chapter 1

Introduction

According to the Pew Research Center, $\frac{2}{3}$ of American adults reported using social media as their primary or secondary source of news in 2016 [11]. Nowadays, we rely on social media to stay up-to-date on politics, beauty, sports, and everything in between. As users of social media, we want the content we see to be as relevant to our interests as possible. And on the other side of things, content creators and publishers want us to engage with their media. Thus, our social media is becoming increasingly personalized, with content recommended to us based on our friends, our follows, and our likes. However, many of us still scroll through our feeds, not clicking or reading anything, until something catches our eye. Why did that something catch our eye? That is precisely what content creators want to know to get us to better engage.

In a paper explaining Netflix's collaborative filtering algorithms and the business purpose they serve, a senior engineer on their data engineering team noted that if a user goes to the search bar instead of watching one of the recommended shows and movies in the main section of the home page, their algorithms have failed [6]. Their goal is to present such a well-curated list that a user has no need to search for a specific title. Most popular social media sites do not have a search bar in the same way that Netflix or Google does – users cannot search for specific content, only

profiles. As users of social media, we either click on what we see or scroll past it as we do not have the ability to search for specific articles or videos that we might find interesting. Thus, it is even more important for content creators that social media recommends to each user content they will engage with.

So, what makes users of social media engage with content? Is it because the content was shared by a specific user, because there was an interesting title, or because or some other factor? This study seeks to answer these questions and determine what characteristics of a user are most predictive of whether or not they will engage with a specific piece of content. To do so, a dataset from Facebook users will be collected and used to model the diffusion of content and to predict content engagement in the network represented by these users.

By the end of this project, the goal is to have found the profile characteristics that are most highly correlated with engagement. Using this information would allow content creators to increase visibility and also allow users of social media to have more interesting and useful social feeds.

Chapter 2

Background

2.1 Networks and Graphs

There are a number of different types of networks, including migration networks, trade networks, and social networks. With the introduction of social networking platforms came online social networks. This project focuses on such online social networks and any mention of social networks refers to *online* social networks unless otherwise specified.

Social networks, and networks more generally speaking, can be modeled as graphs. Traditionally, users are represented as nodes and interactions between users are represented as an edge connecting the relevant nodes. These interactions can include the act of friending, following, liking, sharing, etc. In the friending context, edges are not directed, as friending is a mutual interaction; both nodes are equally involved in the interaction and there is no directionality regarding the friendship. On the other hand, following, liking, and sharing are all directed actions in that one user follows another, likes another users content, or shares content to another user. These interactions are represented with directed edges as there is directionality involved in the interaction. In addition, nodes and edges can have weights associated with them. For example, a

weighted undirected edge representing a friendship with weight of 2 could mean that these two nodes have a friendship that is twice as strong as an edge with weight 1. Similarly, if an edge representing likes from one node a to another node b has weight 5, it can be interpreted that user a liked user b's content 5 times.

2.1.1 Link Prediction Methods

Given that edges represent interactions between two users, predicting an edge is analogous to predicting an interaction between two users. This is useful for this project as one of the goals is to predict the spread of information in a network and one user sharing content to another user is an interaction that can be captured as an edge between the two user nodes. Methods for such link prediction often draw on the structure of the graph to determine the likelihood of a particular edge forming. These structural elements include the number of neighbors a node might have and the degree of a node. Different methods make different assumptions about what structural qualities are most important. For the following link prediction techniques, if x be a node, let $N(x)$ be the set of neighbors of x in the graph G.

Common Neighbors

The most intuitive and simplest link prediction technique is based on the common neighbors score [9]. For two nodes x and y,

$$score(x, y) = |N(x) \cap N(y)|$$

Essentially, this score asserts that the edge between x and y is more likely if both nodes have a large overlap of neighbors. In the friending context, this means that x and y are more likely to be friends if they share many common friends.

Adamic-Adar

Another way to to measure the proximity of two nodes is called the Adamic-Adar score. This score computes the similarity between two nodes x and y by looking at a common feature z that the two nodes share:

$$score(x, y) = \sum_{z \in N(x) \cap N(y)} \frac{1}{log(N(z))}$$

This score refines simply counting common features by weighting rarer features more heavily [9].

Preferential Attachment

A third way to predict an edge between two nodes is to look at the quantity of neighbors each has. For two nodes x and y, this preferential attachment score is defined as

$$score(x, y) = |N(x)||N(y)|$$

This score makes the assumption that the probability that a new edge contains a node x is proportional to the size of the set of neighbors of x, $|N(x)|$ [9]. Thus, nodes that have more neighbors are more likely to get a new neighbor.

2.1.2 Weighted Link Prediction

The previous section, introduces various methods for predicting new links in an unweighted graph. However, it is often useful to predict weighted links. In their paper, Murata, et al. explore ways to modify common link prediction scores to take into account the weights on edges [10]. If x and y are nodes, $N(x)$ is the set of neighbors of node x, and $w(x, y)$ is the weight of edge between nodes x and y, then the scores from 2.1.1 are modified as follows:

1. Weighted common neighbors:

$$score(x, y) = \sum_{z \in N(x) \cap N(y)} \frac{w(x, z) + w(y, z)}{2}$$

2. Weighted Adamic-Adar:

$$score(x, y) = \sum_{z \in N(x) \cap N(y)} \frac{w(x, z) + w(y, z)}{2} \times \frac{1}{log(\sum_{z' \in N(z)} w(z', z))}$$

3. Weighted preferential attachment:

$$score(x, y) = \sum_{x' \in N(x)} w(x', x) \times \sum_{y' \in N(y)} w(y', y)$$

2.2 Diffusion in Social Networks

One of the major areas of research this project draws from is information diffusion in social networks. Analysis of information diffusion has been done on a variety of social networks, including social media sites (e.g. Facebook, Twitter), LinkedIn, and even blogspaces. Most diffusion research seeks to model the interactions between neighbor nodes in a network, which can interpreted as real-world social pressure.

Li, et al. defines two different scenarios under which this social pressure occurs [8]. The first is individual influence, which refers to a single node in a network being able to influence the surrounding nodes. An example of individual influence is the impact that social media influencers have on their followers. In a directed graph, a node that has a lot of individual influence can often be detected if it has a very large outdegree in comparison to its indegree. Li, et al. notes that nodes with great individual influence are very important as they are ultimately the ones who are speeding up the diffusion of information in a given network. Thus, one can measure social influence by

predicting a node's ability to spread information [8]. The next scenario is community influence. A community is defined as "a subset of the network in which the users are densely connected" and communities often have similar attributes, e.g., they like to play badminton, or their research area is similar [8]. Research in community influence is around detecting communities and then in understanding how communities have influence in their larger social networks.

2.2.1 Traditional Diffusion Models

Two of the most used information diffusion models are the Independent Cascade Model and the Linear Threshold Model. Both seek to encode social pressure and the scenarios in which they might occur.

Independent Cascade Model

The Independent Cascade Model (ICM) assumes that a node can only influence a node that it is connected to, and that the nodes are making a binary decision (e.g. whether or not to engage with a given piece of content). This model is referred to as a sender-centric model as senders activate receivers. For a node `v` that has been activated, the model proceeds as follows [8]:

1. Consider all the neighbors of `v`.
2. Each neighbor weighs the fact that `v` chose to become activated, and its own set of influences (e.g. personal preferences).
3. Each neighbor either activates or stays inactive.
4. Repeat for each newly activated node.

Note that inactive nodes can become activated, but in this model activated nodes cannot turn inactive. This model can take into account a variety of interesting attributes that describe the strength of influence that one node `v` has on another node

w. For example, do nodes v and w have lots of mutual friends? Does v only activate occasionally, meaning that node w should pay attention as there is a reason that v activated in this particular interest? If these influences cannot be quantified as a probability in some way (it is often very difficult to do so), probabilities are often assigned uniformly at random. In fact, predicting information diffusion probabilities in social networks is its own area of research.

Linear Threshold Model

The Linear Threshold Model (LTM) also uses the influence of neighboring nodes to determine whether or not a node becomes active. As iterations progress, an inactive node becomes active as more and more of its neighbors become active. Starting with a set of seed active nodes in the graph, a threshold is set for each of the inactive nodes in the graph. This threshold is based on external and internal influences, but, as with LTM, if these cannot be determined, the threshold is often selected uniformly at random for each node. After each iteration, a given inactive node will become active if the sum of the weights of the edges with active neighbor nodes exceeds its threshold.

2.2.2 Extensions with User and Content Features

ICM and LTM are two of the most well-researched models for information diffusion and a number of papers explore modifications and improvements. One of the most relevant for this project is a paper by Lagnier, et al., which looks at how to take into account content and user profiles [7]. Lagnier, et al. notes that the social pressure represented by ICM and LTM is very important, but that these models fail to explicitly represent the content of the information being diffused, the user's profile, and the user's willingness to diffuse information.

In the paper, they show how to integrate these important signals into simple three

feature functions [7]:

1. **Thematic interest** is defined as the interest of a given user in the piece of information. They model this as the proximity between user profiles and the content diffused.

2. **Activity** is determined via a training set and represents the willingness of a node to diffuse information. It is measured as the ratio between the number of pieces of information received and diffused by a node and the number of pieces of information received by that node.

3. **Social pressure** on each user is quantified given the number of neighbors that have already diffused the information.

Lagnier, et al. represents each user by a vector of these three features to be used in their probabilistic modeling. The basic model is to start with a seed set of initial diffusers and then iteratively compute, for each time step, the three feature functions. The node's probability to diffuse is based on a linear combination of these features, with learned parameters.

They ran their models on two blog datasets and compared results with ICM, LTM, and other approaches. They found that the content of the information plays the biggest role (of the three additional pieces of information they experimented with) in the diffusion process [7].

2.3 Content Engagement

Most of the research related to content engagement in social networks comes out of business schools and psychology departments. A study from the College of Business Administration at East Carolina University explored different strategies that are used in social media marketing to drive consumer engagement. Ashley and Tuten sampled

social media content from select brands on the top 100 brands based on Interbrand's Best Global Brands valuation study. The media content included one week of Facebook and MySpace posts, one week of tweets, one week of content from blogs and forums, and all video and photo content between June 2010 and August 2010. A number of message strategies were evaluated including interactivity, functional appeal, emotional appeal, and exclusivity. They found that exclusivity and emotional appeal were the two message strategies most highly-correlated with consumer engagement [3].

Another study by Chu and Kim looked at what the determinants are of consumer engagement in social networks. They examined an interesting facet of the social pressure that other studies have noted: trust. As their work deals primarily with consumer engagement, the goal was to determine when a user would buy a product that they heard about through social media. However, the same thought process can be extended to the more general case of content engagement. If a user trusts the source where the content is coming from, they will be more likely to engage with it [4].

The spread and engagement of user-generated content is another interesting area of study within content engagement. One of the underlying ideas is that an individual with a lot of social influence will be able to generate content that is diffused and engaged with more than users with less social influence. Susarla, et al. created a dataset by scraping YouTube and found that diffusion and engagement can be viewed as two-step process. In the first stage, a piece of information's search characteristics matter most. This stage triggers initial diffusion that creates the space for subscriber networks. After these subscriber networks are established, engagement with content depends more on experience characteristics (i.e. how the content is presented to users is more important than the content itself) [12]. This study is very interesting because it distinguishes the roles and influences between a piece of information's content, and

the way it is presented.

Finally, Weeks and Holbert from the School of Communication at the University of Columbus looked at how specifically news content diffuses and is engaged with in social medias. They used the Pew Research Center for the People and Press's telephone survey on social networks data. Weeks and Holbert looked at various user characteristics and how they were correlated to willingness to diffuse information and engagement with information. Gender was apparently the most correlated factor, with those that identify as female more likely to share news content than males, but those that identify as male more likely to read and engage with news content than females [13]. Their study shows that profile information is correlated with engagement and can therefore be used to predict engagement.

2.4 Marketing Tools

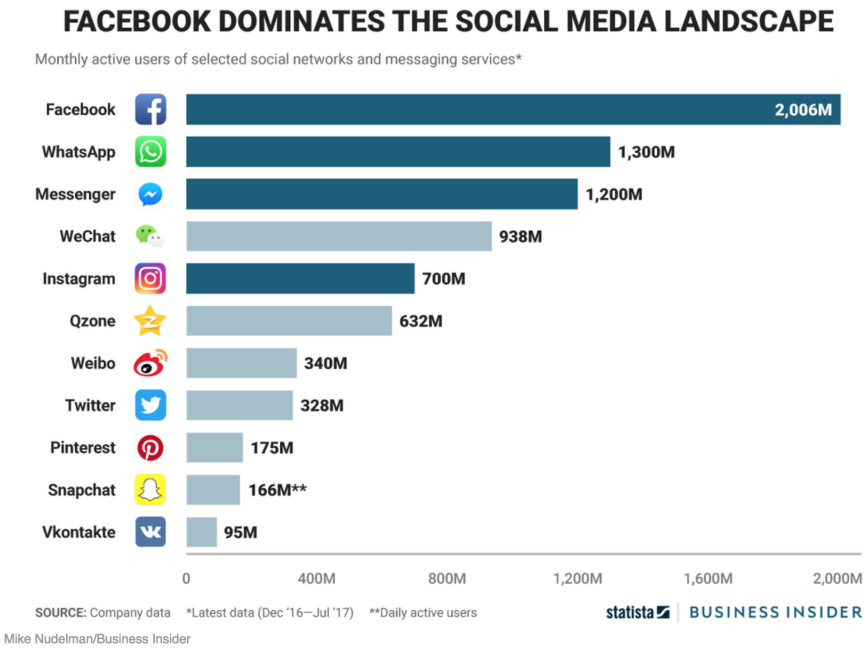
Recently, there have been a number of "smart" marketing tools to help content creators and publishers drive more engagement. one such tool is called Pi, which is described as "an AI-powered social marketing tool [that] can predict engagement" by using "a neural network to build a custom model for each user's patterns." While the code for Pi and other similar marketing tools is not available, their existence shows that there is interest in and a market for better marketing tools. Content creators and publishers want to better present their work to potential content consumers and are turning to personalization and recommendation techniques, which are based on user profiles and content profiles, to better do so.

Chapter 3

Approach

From the previous chapter, it is clear that there has been quite a bit of research done in the information diffusion and content engagement areas. However, there are many ways in which this past work can be extended.

First, it is possible to use a more relevant dataset for diffusion analysis and prediction. Past work on predicting diffusion has been done on datasets from blogspaces, MySpace, and Twitter. However, as seen from the following chart, no study has been done on today's most popular social networking site [5]



In order to improve on past work, this project uses data collected from Facebook users. The data includes what articles appeared on a user's Facebook news feed and which ones they actually click on, including article features (who shared the article, the title, how many likes/reactions it has, etc.) and user public profile features (gender, location, work, etc.). This dataset is used to predict diffusion using both weighted link prediction methods and the use of feature functions as introduced in 2.2.2.

The second main contribution of this study is to use this granular dataset to use training to study engagement. While studies have ben done on predicting content engagement in social networks, these studies do not work in the same way that "smart" tools for marketers work. More specifically, the models from these studies are not trained on a dataset to determine how to present content to users. This project will develop an engagement prediction model using feature functions informed by previous work done in content engagement and will also use a more traditional classifier (a random forest classifier) to predict content engagement.

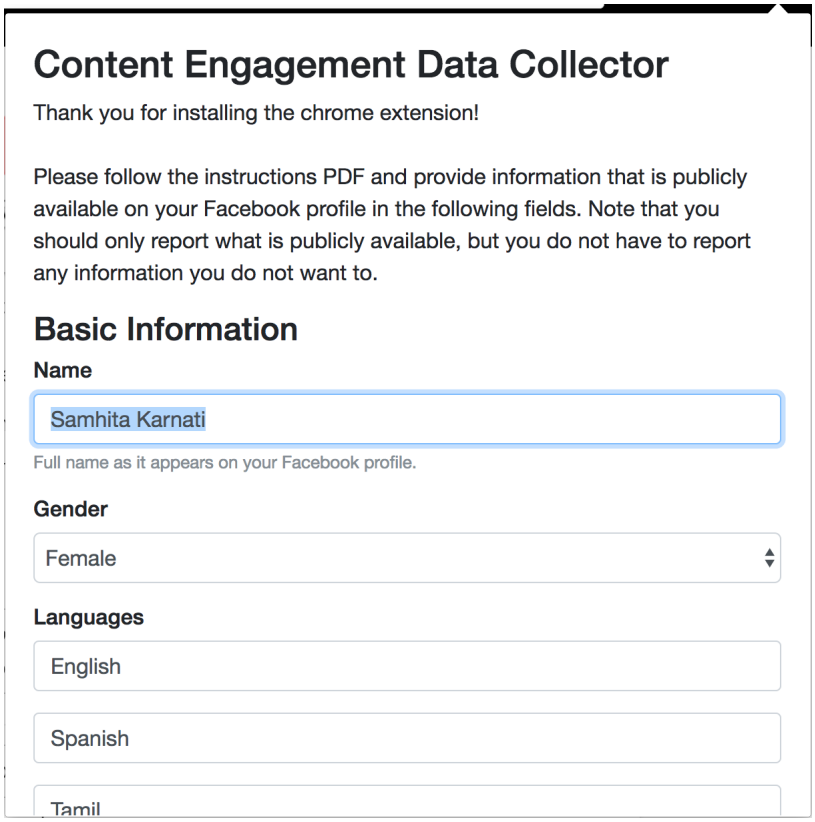
Chapter 4

Data Collection

4.1 Creating the Data Collector

The necessary data for this project is what articles appear on a person's Facebook News Feed, which ones they click on, what information was presented to them when they viewed the articles, and user's public profile information. In order to collect this data from users, a Chrome extension was used. This choice means that only those who use Facebook on their Chrome browser could be used in the data collection. However, given that Chrome is the most popular web browser, at 58.4% market share (the next most popular is Firefox at 13.54%) [2], this is not be too limiting.

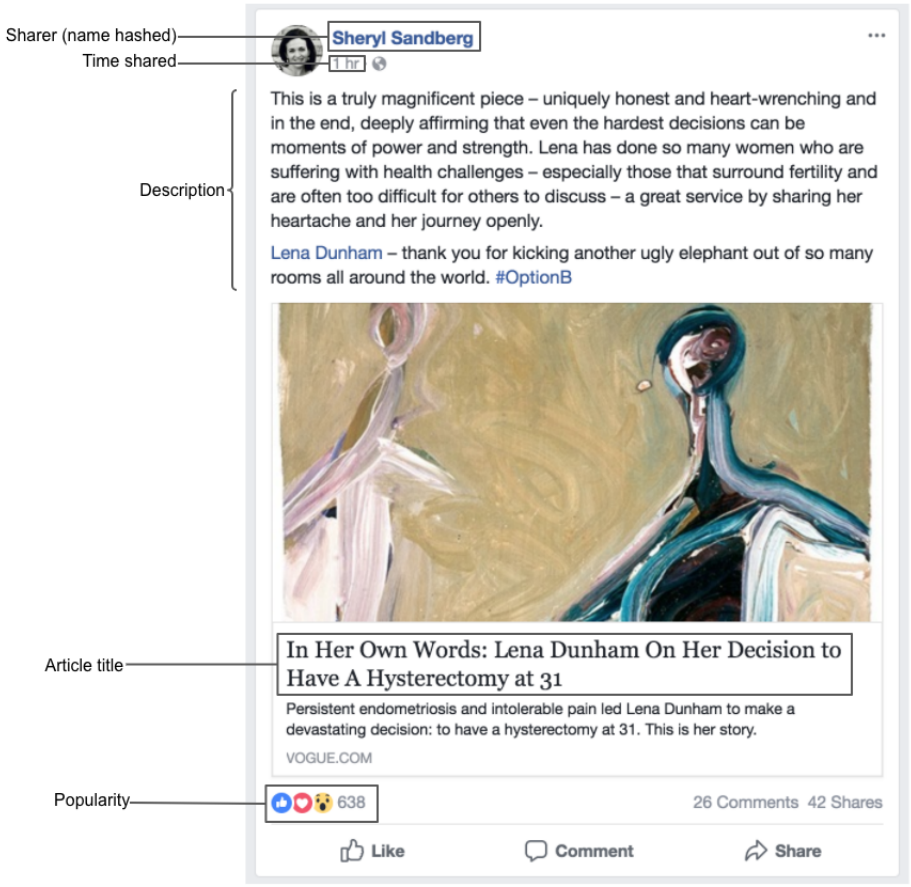
The extension contains two components: the popup where users can provide public profile information and a background script that records articles and associated information. The popup has a simple UI shown in the following screenshot:



It asks users for basic profile information, places they have lived, education, and where they have worked. While users are requested to only self-report information that is publicly available, they do not need to report information that they do not want to. The name field is hashed to ensure anonymity of participants.

The background script has more functionality as it parses the html to find articles and the accompanying information. In order to only track articles on the Facebook News Feed, the extension determines that the current tab's URL is `https://facebook.com`, the only URL where the News Feed is shown. The articles are found by searching the html on the page for occurrences of `a href=`, which indicates the start of a URL. Since this project is only interested in articles, links that are relative and contain `facebook.com` are not considered. The extension also parses out the description the article was shared with (if it exists), the sharer (which is hashed), the time that it was shard, whether or not there was an image presented with the link, the title, and how many likes and reactions the article has. Also stored is the

current time that the URL was seen and whether or not the URL was clicked on, which serves as the metric for engagement. The following shows a screenshot of an example article shared on Facebook, with the recorded information labeled:



Each URL and its associated information is stored in a local array. If the extension records that an article is seen again on the News Feed, it is only recorded as a distinct instance if the current time it was viewed is more than five minutes after the last time it was viewed. This threshold of five minutes is set because the average Facebook session lasts 5.02 minutes [1]. This local array is flushed to the database every time an article is clicked and an hour after the last flush occurred. A similar logic is used to sync the data.

4.2 Database Storage

In order to store the data collected across multiple Chrome extension, AWS tools are used. DynamoDB databases, one to store user information and the other to store articles, are used as DynamoDB is an easy-to-use NoSQL database service that has a flexible data model. For the users table, each row represents an individual participant in the study. Thus, it makes sense that the primary key is the hashed name of the participant. Each row in the articles table represents an interaction between a user and an article, with an interaction meaning that the user saw the article or that they clicked on the article. In order to represent this, the primary key for this table is the hashed name of the user and the sort key is the URL that they interacted with.

In order to interact with the databases, Lambda functions are used. AWS Lambda is very convenient it allows code to be run without provisioning or managing servers. There are three necessary interactions: saving an article, saving a user, and getting an article (used when the locally stored articles are being flushed to the database to check whether or not the user-article pair already exists). The functions that allow for these interactions are written in Python.

Finally, API Gateway is used to link the databases to the Lambda functions and create the API endpoints that can be called from the extension code.

4.3 Getting Participants

The process of recruiting participants consisted of two phases. First, I reached out to my friends and family individually. In this way, I reached out to 190 people, of which 126 said that they would participate. Of this 126, only 98 actually completed all the necessary instructions.

Needing more participants, I sent emails to both my eating club's (Tower) listserv, the Entrepreneurship Club members listserv, and various residential college listservs.

This closed the gap in the necessary participants. In the end, I had NUMBER using the extension for the minimum three-week period. The extension collected NUMBER user-article interactions. These methods for getting participants has certain drawbacks and limitations that are addressed in 6.4.1.

4.4 Cleaning Up the Data

TODO

1. user hashes in the articles table but not in the users table (problematic for using profile features to predict engagement, but data can still be used for analyzing diffusion)

2. User hashes in the users table but not in the articles table (again, problematic for predicting engagement component of thesis, and also problematic for analyzing diffusion)

3. Removing entries with improperly parsed html

Chapter 5

Prediction Methods

5.1 Building an Interaction Network

One of the most interesting parts of this dataset is that it captures not only what each participant saw, but also who shared it with them. Since each name is hashed to a unique ID, it is possible to thus reconstruct a network of interaction. Note that this is different from recreating the actual social network as friendship information is not available. This network is interesting to analyze in and of itself and can also be used for predicting diffusion.

The network is represented as a directed graph where nodes represent users and an edge represents a user sharing an article with another user. Weights on edges correspond to the number of articles shared. Given that time information is associated with each user seeing a particular article, this network graph can be constructed at multiple time steps. Doing so, allows the use of past interaction history in predicting future interactions.

5.2 Predicting Diffusion

This project explores two ways to predict diffusion: edge prediction from the network graph and model-based predictions using the feature functions described in 2.2.2.

5.2.1 Edge Prediction

Given the network representation outlined above, predicting diffusion is analogous to predicting the existence of an edge in the interaction network. To do so, a random user-article pair is selected. If the user did see that article in the dataset and there are multiple instances of it being seen, a random instance is chosen and the network graph at the time step immediately prior to the instance is constructed. If the user did not see the article in the dataset, the expected time to edge appearance is used to determine at what time step to construct the network graph. In order to find the expected time to edge appearance, the article view time data is used to fit a probability distribution. If the standard deviation is not too large, this method of determining the appropriate time to predict that a particular edge did not exist will be successful. However, if the standard deviation is very large, then this will not yield the best results.

With the appropriate graph constructed, edges are predicted using various similarity scores for weighted graphs. The three that are used in this study are the weighted common neighbors, weighted Adamic-Adar, and weighted preferential attachments scores, which are introduced in 2.1.2. For the three scores, a connection weight $score(x, y)$ is assigned to each pair of nodes x and y in the network, and then a ranked list in decreasing order of $score(x, y)$ is produced. If $N(x)$ is the set of neighbors of x in a network and $w(x, y)$ is the weight of the edge between nodes x and y, then the scores are defined as follows:

1. Weighted common neighbors:

$$score(x, y) = \sum_{z \in N(x) \cap N(y)} \frac{w(x, z) + w(y, z)}{2}$$

2. Weighted Adamic-Adar:

$$score(x, y) = \sum_{z \in N(x) \cap N(y)} \frac{w(x, z) + w(y, z)}{2} \times \frac{1}{log(\sum_{z' \in N(z)} w(z', z))}$$

3. Weighted preferential attachment:

$$score(x, y) = \sum_{x' \in N(x)} w(x', x) \times \sum_{y' \in N(y)} w(y', y)$$

These scores are evaluated at various acceptance thresholds using Receiver Operating Characteristic (ROC) curves. ROC curves show the true positive rate plotted against the false positive rate at various decision thresholds and are useful in evaluating binary classifiers. Given the different assumptions that these scores are based on, determining the most successful score for link prediction is indicative of qualities of the network and the way information spreads.

5.2.2 Incorporating Content and User Profiles

While the edge prediction methods do take into account social pressure and past interaction, two important sets of features they do not incorporate that are available with this dataset are content profiles and user profiles. In order to use this information, the diffusion prediction strategy developed by Lagnier, et al. is used.

As introduced in 2.2.2, this model is based on explicitly quantifying thematic interest (the interest of a given user in the current piece of content), activity (the willingness of a user to diffuse information), and social pressure. For a given node

x (a user profile) and a given piece of content c, these three feature functions are defined as:

1. Thematic interest:

$$S(x, c) = sim(\texttt{tf-idf}(x), \texttt{tf-idf}(c))$$

where sim is the cosine similarity (also used by Lagnier, et al.) and $\texttt{tf-idf}(n)$ is the `tf-idf` matrix of the user or content profile n. `Tf-idf` is a way to represent the content of a document or piece of text. It stands for term frequency (the proportion of occurrences of a specific term to the total number of terms in the document) times inverse document frequency (inverse of the proportion of documents that contain that word/phrase). In order to create a `tf-idf` matrix for a piece of text, first the stop words (like *the* and *and*) are removed. Then all words are stemmed (such that *runner*, *running*, and *run* all match to the same word, *run*) For a user x, $\texttt{tf-idf}(x)$ is built from the words that make up their public profile. For a piece of content c, $\texttt{tf-idf}(c)$ is built from the words that make up the title and description associated with it. Stemming and `tf-idf` matrix generation are performed using `scikit-learn` methods.

2. Activity:

$$A(x) = \frac{out(x)}{in(x)}$$

where $out(n)$ and $in(n)$ are the outdegree and indegree respectively of node n.

3. Social pressure:

$$P(x) = \frac{\sum_{r \in R(x)} I(c \in T(r))}{|R(x)|}$$

where $R(x)$ is the set of all users that user x has previously received content from, $T(r)$ is the set of pieces of content node r has seen, and I is the indicator function.

With these feature functions, it is possible to represent each user as a vector v of these three functions that evolve over time for each new piece of content c:

$$v = \begin{pmatrix} S(x,c) \\ A(x) \\ P(x) \end{pmatrix}$$

These features are then combined through simple linear combinations to generate a function for each user-content pair at each time step:

$$f_\lambda(x,c,t) = \lambda_0 + \lambda_1 v_1 + \lambda_2 v_2 + \lambda_3 v_3$$

where λ_0, λ_1, λ_2, and λ_3 are positive parameters that are learned through ridge regression to approximate a polynomial function. If a user x saw content c and time t, $f_\lambda(x,c,t) = 1$. If not, then $f_\lambda(x,c,t) = 0$. The training set consists of various (x,c,t) tuples, calculated $S(x,c)$, $A(x)$, and $P(x)$ scores, and the ground truth $f_\lambda(x,c,t)$ function outputs.

With the parameters trained, the function $f_\lambda(x,c,t)$, random user-content pairs are again selected. If the interaction exists in the dataset, $f_\lambda(x,c,t)$ is calculated for the t immediately prior to the interaction. If the interaction doesn't exist, the estimated time to edge appearance is again used to determine at what t to calculate $f_\lambda(x,c,t)$. The predicted outputs of the $f_\lambda(x,c,t)$ function are once again stored and then evaluated using ROC curves.

Based on the hypothesis that there is a correlation between user profiles and content profiles and given Langier, et. al's success using this methodology, it is expected that this technique for predicting diffusion will be more successful than simply using edge prediction.

5.3 Content Engagement

As with predicting diffusion, this project explores two ways to predict content engagement: calculating feature functions based on the content and profile features that are known to be important, and training engagement prediction classifiers that determine what the most important features are.

5.3.1 Developing an Engagement Model

As Langier, et al identified in their information diffusion study, it is possible to determine important feature functions based on content and profile features that have been shown to be important by other studies. More specifically, it is possible to quantify thematic interest (same as thematic interest in the diffusion prediction model), sharer trust, and past engagement history. For a given node x (a user profile) and a given piece of content c shared by user y, these three feature functions are defined as:

1. Thematic interest:

$$S(x, c) = sim(\texttt{tf-idf}(x), \texttt{tf-idf}(c))$$

   where sim is the cosine similarity, as in the diffusion prediction thematic interest feature function.

2. Trust:

$$T(x, c, y, t) = \frac{\sum_{c \in R(x,y)} I(c \in E(x))}{|R(x, y)|}$$

   where $R(x, y)$ is the set of all articles that user x has seen that were shared by user y, $E(x, t)$ is the set of articles that x has engaged with at time t, and I is the indicator function.

3. Past engagement history:

$$H(x, c) = sim(\texttt{tf-idf}(c), \texttt{tf-idf}(d))$$

where sim is the cosine similarity and $\texttt{tf-idf}(d)$ is a `tf-idf` matrix for all documents that user x has engaged with at time t.

As with the diffusion prediction model, these feature functions are used to represent each user as a vector that evolves over time for each piece of content c

$$v = \begin{pmatrix} S(x, c) \\ T(x, c, y, t) \\ H(x, c) \end{pmatrix}$$

with these feature functions being combined through simple linear combinations to create a function for each user-content pair at each time step:

$$f_\lambda(x, c, t) = \lambda_0 + \lambda_1 v_1 + \lambda_2 v_2 + \lambda_3 v_3$$

These parameters are learned through ridge regression to approximate a polynomial function. The testing and evaluation follows same procedure described in 5.2 for the diffusion prediction model.

5.3.2 Training an Engagement Prediction Classifier

While prior work has provided a set of features that are likely important for predicting content engagement, using other traditional classification algorithms on the data is interesting and can highlight features other than thematic interest, trust, and past engagement history that might be important for this problem. This project compares the content engagement model in 5.3.1 with a random forest classifier.

Random Forest Classification

A Random Forest (RF) is a classification technique that uses multiple decision trees to determine the label to give to a particular data point. The RF is made up of multiple decision trees, each of which solves the classification problem. A decision tree predicts a target value based on various input variables. In the tree, each node corresponds to one of the input variables and there are edges to children for each of the possible values of that input variable. Thus, each leaf represents a value of the target variable given the values of the input variables represented by the path from the root to the leaf. One of the most useful parts of decision trees is that they are easily used in cases where the data does not have purely numerical features.

The RF combines the results from multiple decision trees. Each decision tree's result can thought of as a vote, which the random forest outputting the category that the majority of decision trees voted on.

Using RF for Engagement Classification

This project uses `scikit-learn`'s RF method for classification. As mentioned, categorical features (like gender being male or female) is perfectly fine for RF, so there is little data preprocessing to do. The training set consists of input variables across all features from both the user and article that the row pertains to, and the output value is a `true` or `false` value, indicating whether or not the user engaged with the article.

TODO: dealing with NULL variable values

Chapter 6

Results

6.1 Dataset Exploration

6.1.1 Mapping the Network

6.1.2 Trending Articles

6.2 Predicting Diffusion

6.3 Predicting Engagement

6.4 Limitations and Additional Considerations

6.4.1 Data Limitations

While this dataset will be sufficient for the goals of this project, there are certainly aspects in which it falls short. For example, there are other forms of social media where people share articles on, like Twitter. Furthermore, this extension only tracks what users read on their laptops; there are many people who prefer to read articles on their phones or tablets.

In terms of what the extension stores with relation to each user-article interaction, one possibly important characteristic is what picture is shown to the user. While I track whether or not a picture was presented with an article, I do not store that picture for further analysis. One of my friends mentioned that he does not even read the title of articles sometimes, and only looks at the picture to determine if an article will be interesting. Incorporating such information into this thesis would require storing the images and doing some image analysis, which is out of the scope of this text-based project.

Given that the Institutional Review Board (IRB) requires a consent form and the guarantee that participants are at least 18 years of age, I was limited in my participant acquisition. For example, I could not simply get participants through Mechanical Turk or some other anonymous but mass system. This also means that I primarily got participants who are my friends at Princeton. They have similar friend lists on Facebook to me and are therefore seeing similar articles. While this might be helpful in that there will be many more connections in the network that I create with my participants based on sharer hashes, there are certain features that will likely be blown out of proportion. An example is articles published by "The Daily Princetonian." Given that most participants are from Princeton, they will be highly likely to see and click on any articles that have the word "Princeton" in the title or comment. It will be interesting to see if these types of features are more highly correlated with engagement than others.

Finally, as the extension parses out html looking for particular tags and strings, it is dependent on the structure of Facebook not dramatically changing during the course of this study. Also, when I sent the extension to my first few participants, Zuckerberg had just announced that Facebook was going to make a conscious effort to show more personal content and less public content from pages, in order to facilitate more "meaningful interactions" [14]. I saw no difference over the course of data collection

with regard to how many articles I was seeing on my News Feed, but it would be interesting to know when Facebook made that change to see if it was reflected in the data in any way.

Chapter 7

Conclusion

Bibliography

[1] Most popular social networking apps in the united states as of november 2017, by average session duration, Nov 2017.

[2] Browser & platform market share, Jan 2018.

[3] Ashley, C., and Tuten, T. Creative strategies in social media marketing: An exploratory study of branded social content and consumer engagement. *Psychology & Marketing 32*, 1 (2015), 15–27.

[4] Chu, S.-C., and Kim, Y. Determinants of consumer engagement in electronic word-of-mouth (ewom) in social networking sites. *International journal of Advertising 30*, 1 (2011), 47–75.

[5] Dunn, J. Facebook totally dominates the list of most popular social media apps, July 2017.

[6] Gomez-Uribe, C. A., and Hunt, N. The netflix recommender system: Algorithms, business value, and innovation. *ACM Transactions on Management Information Systems (TMIS) 6*, 4 (2016), 13.

[7] Lagnier, C., Denoyer, L., Gaussier, E., and Gallinari, P. Predicting information diffusion in social networks using content and user?s profiles. In *European conference on information retrieval* (2013), Springer, pp. 74–85.

[8] Li, M., Wang, X., Gao, K., and Zhang, S. A survey on information diffusion in online social networks: Models and methods. *Information 8*, 4 (2017), 118.

[9] Liben-Nowell, D., and Kleinberg, J. The link-prediction problem for social networks. *journal of the Association for Information Science and Technology 58*, 7 (2007), 1019–1031.

[10] Murata, T., and Moriyasu, S. Link prediction of social networks based on weighted proximity measures. In *Proceedings of the IEEE/WIC/ACM international conference on web intelligence* (2007), IEEE Computer Society, pp. 85–88.

[11] Shearer, E., and Gottfried, J. News use across social media platforms 2017, Sept 2017.

[12] Susarla, A., Oh, J.-H., and Tan, Y. Social networks and the diffusion of user-generated content: Evidence from youtube. *Information Systems Research 23*, 1 (2012), 23–41.

[13] Weeks, B. E., and Holbert, R. L. Predicting dissemination of news content in social media: A focus on reception, friending, and partisanship. *Journalism & Mass Communication Quarterly 90*, 2 (2013), 212–232.

[14] Zuckerberg, M., Jan 2018.