

An Information Diffusion Approach for Detecting Emotional Contagion
in Online Social Networks

by

William David Cole

A Thesis Presented in Partial Fulfillment
of the Requirements for the Degree
Master of Science

Approved June 2011 by the
Graduate Supervisory Committee:

Huan Liu, Chair
Hessam Sarjoughian
K. Selcuk Candan

ARIZONA STATE UNIVERSITY

August 2011

ABSTRACT

Internet sites that support user-generated content, so-called Web 2.0, have become part of the fabric of everyday life in technologically advanced nations. Users collectively spend billions of hours consuming and creating content on social networking sites, weblogs (blogs), and various other types of sites in the United States and around the world. Given the fundamentally emotional nature of humans and the amount of emotional content that appears in Web 2.0 content, it is important to understand how such websites can affect the emotions of users. This work attempts to determine whether emotion spreads through an online social network (OSN). To this end, a method is devised that employs a model based on a general threshold diffusion model as a classifier to predict the propagation of emotion between users and their friends in an OSN by way of mood-labeled blog entries. The model generalizes existing information diffusion models in that the state machine representation of a node is generalized from being binary to having n -states in order to support n class labels necessary to model emotional contagion. In the absence of ground truth, the prediction accuracy of the model is benchmarked with a baseline method that predicts the majority label of a user's emotion label distribution. The model significantly outperforms the baseline method in terms of prediction accuracy. The experimental results make a strong case for the existence of emotional contagion in OSNs in spite of possible alternative arguments such as confounding influence and homophily, since these alternatives are likely to have negligible effect in a large dataset or simply do not apply to the domain of human emotions. A hybrid manual/automated method to map mood-labeled blog entries to a set of emotion labels is also presented, which enables the application of the model to a large set (approximately 900K) of blog entries from LiveJournal.

ACKNOWLEDGEMENTS

I would like to acknowledge and thank the faculty and staff who have enabled and inspired me to write and defend this thesis: my graduate advisor, Huan Liu; my committee members, Hessam Sarjoughian and K. Selcuk Candan; and my academic advisor, Martha Vanderberg. I would also like to thank all members of the ASU Data Mining and Machine Learning laboratory, especially Reza Zafarani, Ali Abbasi, and Pritam Gundecha, for their invaluable knowledge and guidance. Honorable mentions go to Shamanth Kumar, Xufei Wang, and Huiji Gao for their cultural insights, discussions of gastronomical matters, and general camaraderie. Finally, I would like to thank the Office of Naval Research for material support of this work.

TABLE OF CONTENTS

	Page
LIST OF FIGURES	v
LIST OF TABLES	vi
1 INTRODUCTION	1
1.1 Problem Statement	2
1.2 Challenges	2
1.3 Contributions of This Work	4
2 BACKGROUND AND RELATED WORK	6
2.1 Information Diffusion and Influence Propagation	6
2.2 Epidemic Model	7
2.3 Linear Threshold and Independent Cascade Models	7
2.4 General Threshold Model	8
2.5 Information Diffusion Literature Review	8
2.6 Sentiment Analysis	10
3 A DIFFUSION MODEL SOLUTION	13
3.1 Influence Propagation and Emotional Contagion	13
3.2 Diffusion Models as Classifiers	13
3.3 Predicting Emotion with a Diffusion Classifier	14
3.4 Selecting a Diffusion Model	16
3.4.1 Epidemic Model	16
3.4.2 General Threshold, Independent Cascade, and Linear Thresh- old Models	16
3.5 Class Labels for Emotional Prediction	18
3.6 Adapting Diffusion Model for Emotions	20
3.6.1 Nodes as finite state machines	20
3.6.2 Modeling Influence Decay	23
3.6.3 Influence Probability as Emotional Agreement	24

	Page
3.7 Baseline Prediction Method	27
3.8 Algorithms	27
3.8.1 Training State: Learning Influence Probabilities	28
3.8.2 Testing Stage: Predicting Emotions	29
3.9 Mood Labeled Corpus	29
3.10 Methods for Mapping Moods Labels to Emotion Labels	31
3.10.1 Normalized Google Distance	32
3.10.2 SentiWordNet	32
3.10.3 Manual Labeling	33
4 EXPERIMENTS	34
4.1 Data Set: LiveJournal	34
4.2 Preprocessing	34
4.3 Experimental Parameters	36
4.4 Cross Validation	37
4.5 Output	37
5 EXPERIMENTAL RESULTS	39
5.1 Overview	39
5.2 Trends	40
5.3 Analysis	41
6 CONCLUSIONS AND FUTURE WORK	44
REFERENCES	46

LIST OF FIGURES

Figure	Page
3.1 Agreement Matrix, $M_{v,w}$	26
5.1 Improvement Margin over all parameter values	42

LIST OF TABLES

Table	Page
3.1 Sequence of blog entries	25
4.1 Top Mood Labels by Emotion	36
5.1 Marginal Accuracy, Model vs. Baseline	40

Chapter 1

INTRODUCTION

The Worldwide Web—or simply “the Web”—is a real part of modern society as demonstrated by the widespread availability of the internet, the amount of time people spend online, the highly interactive nature of “Web 2.0” sites, and the effects such sites can have on our everyday lives. Currently 240 million people in the United States (77% of population) have access the internet via fixed sources (cable, DSL, dial-up) [39], spending an average 32 hours online per month. It is reasonable to suggest that these figures would be less consequential if the dominant mode of use of the internet were passive and the content were static, as in the previous era now known as “Web 1.0”. Presently, user-generated content and interactive websites such as social network sites, real-time (video) chats, and now-venerable weblogs (blogs)—so-called Web 2.0—the online experience can be as interactive as experiences in the real world. The shift from static to dynamic content on the Web has caused a shift in human communication akin to that of the adoption of the telephone in the early 20th century, and the degree of participation in this new mode of social interaction underscores this point. There are more than 750 million active users on Facebook, sharing more than one billion pieces of content every day [17]. The effects of participation in Web 2.0 permeate our lives. Social networks help us establish and maintain friendships and social ties not only by supporting person-to-person communication, but also by providing a personal publisher/subscriber model that enables users to easily make content available to interested parties en masse. Other Web 2.0-era sites have wide-ranging effects on our lives. User-submitted reviews guide our product purchases and use of services. Blogs alter our political opinions and potentially our voting behavior. The nascent use of mobile internet-enabled devices promises to further accelerate growth of the number of participants and the degree of participation in interactive Web use. For example, those who utilize Facebook’s mobile applications are twice as active as those who use strictly non-mobile platforms[17].

In spite of the pervasiveness of the Web 2.0 in our lives and the potential implications for the emotional state of users, there is little scientific knowledge about how it affects our emotions, directly or indirectly. The proliferation of personal blogs in the first years of Web 2.0 demonstrated a widespread desire for personal expression by way of this new medium, from exposition of the mundanities of one's daily life to outpourings of happiness, sadness, excitement, disappointment, love, and hate. In its many forms, emotional content clearly played a role in the ascent of the Web as an extension of society. Given that humans are fundamentally emotional beings, experiencing all things with and through emotion, it is not hard to imagine that the consumption of emotional content through online sources could have an impact on the emotions of the consumer. Yet to my knowledge, no significant research has been conducted to verify or refute such a claim.

1.1 Problem Statement

The very real significance of Web 2.0 in our lives, the probable emotional implications thereof, and the lack of empirical research in examining the resulting emotional effects on users motivates this work. Due to the popularity of online social networks such as Facebook, MySpace, and LiveJournal, and the plentiful data provided by these websites, I restrict the focus of this work to them and attempt to answer the following question. Do emotions expressed in online social networks have a direct and measurable effect on users emotions? In other words, are emotions are measurably contagious in the context of online social networks?

1.2 Challenges

To determine if emotions are contagious in online social networks, a variety of challenges must be overcome. A subset of these challenges pertain to research in context of the Web.

- **Scale.** Scale is a challenge for research in a Web context since extremely large data sets are the norm, and computational costs can be substantial even for efficient algorithms. For example, Facebook currently has over 750 million active users, MySpace has 125 million users, and Twitter has approximately 106 million users. With such enormous user bases come even larger sets of content that must be reckoned with for a representative study.
- **Noise.** Web data is rife with noise, which takes two main forms. First is noise in the form of functional code such as HTML and JavaScript. The second pertains to content, specifically in the form of spam and marketing content, or content that otherwise has no emotional bent. Such content will obscure emotional contagion.[27].
- **Data sources.** Data must be acquired from a source that not only manifests emotional contagion, but that is also reasonably convenient to obtain. Data acquisition may involve difficult tasks such web-crawling and page scraping. Furthermore, some sites such as Facebook limit the extent to which automated crawling of pages is possible, further complicating the task.

This work in particular involves some additional challenges. To my knowledge the topic of emotional contagion in online social networks is relatively novel, having been addressed in only a few studies [46]. Thus solutions must be devised to the following issues.

- **Analysis of emotional content in textual data.** Direct, consistent, and standard indicators of emotion are generally unavailable in online social networks. Features of the data that might provide this information must be identified and suitable techniques to extract it must be applied. Analysis of natural language text is inherently difficult, and simple solutions are rare. Sentiment analysis techniques

presented in the literature may be useful, but are generally ineffective for domain-independent emotional content that is of interest in this work.

- **Influence detection.** Given a way to detect emotion, how can it be determined that a user’s emotions are subject to social influence? The topic of social influence has a fairly rich history in the literature, but it is unclear if existing methods are applicable to emotional content. If the literature is used as a starting point, the difficulties posed by homophily and confounding influences must still be overcome. Homophily is the tendency for similar users to associate with one another—the “birds of a feather” effect. For instance, it is possible that generally happy users are naturally drawn to other happy users, making it difficult to discern if one has affected another’s emotions. Confounding influence is a source of influence associated with an environment shared by some users in a given network, such as local or national events, weather, etc [2].
- **Lack of ground truth.** Given solutions to the challenges listed above, it is impossible to know the true extent to which emotion is contagious, thus limiting the ability to assess the validity of any methods applied to the task.

1.3 Contributions of This Work

This work makes the following contributions.

- A model is devised to predict emotional influence among users and their friends in a large online social network using a novel variant of a general threshold model based on existing work in information diffusion.
- The model is demonstrated by experiment to outperform a baseline model on a large data set from LiveJournal, demonstrating a likelihood that emotion does propagate through online social networks.

- A large set of blog mood labels is mapped to a small set of emotion labels facilitating sentiment classification of any an arbitrary set of mood-labeled blog posts.

The next chapter introduces some background and provides a literature review on information diffusion and sentiment analysis. The following chapter presents the details and challenges of the problem to be solved and a solution thereto. Finally, experiments are described in which the performance of the solution is compared with a baseline method with respect to a set of blog posts from LiveJournal. Critical analysis and conclusions on the results of the experiments is provided, and possible avenues for future work on the problem are discussed.

BACKGROUND AND RELATED WORK

This work builds on existing work in two areas of research: information diffusion and sentiment analysis. Information diffusion literature provides some basic models which can be adapted for the purpose of analyzing how emotion spreads through a network. Work in sentiment analysis provides methods to classify textual data in terms of emotion. In this chapter I discuss some background and significant publications in these two areas.

2.1 Information Diffusion and Influence Propagation

As the title suggests, a major portion of this thesis is founded on work in information diffusion. Information diffusion is a variant of an area of study in the social sciences called diffusion of innovations, which “seeks to explain how, why, and at what rate new ideas and technology spread through cultures” [44] and has been applied in the study of such diverse topics as the practice of boiling to disinfect water by villagers in Peru and the dominance of QWERTY keyboard layout over the Dvorak layout [36]. Research in diffusion of innovations has spawned various theoretical models, providing the basis for work in information diffusion, which examines the spread of information in networks. A sub-area of information diffusion, influence propagation, has come to prominence more recently and is commonly applied to the marketing domain to study the process by which new products are adopted in a social network. Marketers use information diffusion models to maximize the implementation of marketing to increase the likelihood that commercial products will be widely adopted. Influence propagation supplies the intuition with which a solution to the problem of emotional contagion can be developed. The spread of emotions can be thought of as a special case of influence propagation, thus influence propagation models can be adapted to determine whether or not emotions propagate in social networks. The literature offers four main approaches to modeling influence propagation: epidemic model, threshold model, cascade model,

and game theoretic approach. I discuss the first three models in the following pages. I forego discussion of the game theoretic approach because a fundamental property of game theory is motivation of actors to optimize outcomes [18], in contrast to the subject of this work. An assumption of this work is that the phenomenon of emotional contagion is not the result of *intentional* behavior of actors; it is the result of a natural process of influence.

2.2 Epidemic Model

The approach to influence propagation I discuss first is the epidemic model, which is based on the cycle of infectious disease in a host. Nodes in a network follow a cycle of being susceptible (S), infected (I), or recovered (R) (in that order) depending on contact with infected nodes and the temporal properties of the respective disease. Disease propagates from from infected to susceptible nodes with a given probability; recovery (or removal from the network, modeling death) follows infection after a given period of time elapses[20]. Recovery may be followed by another susceptibility state, depending on the disease. Use of the term *contagious* with respect to emotion suggests a natural similarity between the subject of the hypothesis and general epidemiology. However, it is questionable whether the S-I-R(-S) pattern applies to emotions to the same extent that it does to infectious disease, since human emotion does not necessarily follow a rigid cycle like that of an infectious disease.

2.3 Linear Threshold and Independent Cascade Models

The linear threshold model (LTM) employs a network in the form of a weighted graph where each node v chooses a threshold θ_v . Starting with an initial set of active nodes, the activation function f_a (see Definition 2.3.1) is evaluated at discrete time steps. If f_a exceeds θ_v , v becomes active. With strictly positive real edge weights, f_a increases monotonically, modeling the intuition that as more of v 's friends become active, the greater the probability that v will become active.

DEFINITION 2.3.1 *Activation Function for LTM.* Given node v , v 's active neighbors, A_v , and an edge weight $\omega_{w,v}$ mapped to each edge (w, v) , the activation function, f_a , is the sum of edge weights between node v and A_v .

$$f_a() = \sum_{w \in A_v} \omega_{w,v}$$

Like LTM, the independent cascade model (ICM), models propagation in discrete time steps using a weighted graph. At a given timestamp an active node v attempts to influence his inactive neighbor w to become active. w becomes active with a probability equal to the weight of the link from v to w (without regard to the state of any other neighbors). If w does not activate, v can make no more attempts to influence w .

2.4 General Threshold Model

A general threshold model (GTM) that generalizes both LTM and ICM is presented by Kempe et al[21]. To generalize LTM, the activation function is permitted to be an arbitrary monotone function of the edge weights between v and his neighbors, as opposed to being strictly linear. To generalize ICM, the authors describe it as a special case of GTM where the activation function of node v is a constant probability specified as part of the system.

2.5 Information Diffusion Literature Review

The advent of online social networks, along with the need to understand various processes they manifest, has presented scientists with many problems suitable for information diffusion techniques. One such problem that has received a lot of attention in recent years is influence maximization (IM) of word-of-mouth (or viral) marketing. Consider a network of customers to whom a firm intends to market some product. It is desirable to maximize the effectiveness of marketing expenditures by targeting the most influential nodes in the network. Domingos and Richardson introduced the idea of enhancing marketing strategies with a word-of-mouth approach using probabilistic models of interaction [15, 35]. Kempe et al applied diffusion of innovations theory to the marketing

IM problem. They devised a greedy hill-climbing algorithm to a generalization of the linear threshold model to achieve the first provable performance guarantees of influence maximization [21]. The literature has been enriched by various attempts to improve on early work in IM and to address subproblems thereof. Kimura et al devise a method to rank influential nodes using diffusion data learned from an instance of the independent cascade model [22]. Chen et al present methods to enhance the speed and efficacy of the method presented by Kempe et al [21] using a random process and node-degree heuristics [9]. Other work exploits network modularity, i.e., the presence of communities within a network, to improve the performance of the greedy hill-climbing approach [8, 42]. Some research in IM has focused on models of influence not based on ICM or LTM. Lim et al propose a method to find influential nodes in blog networks based on the influence of the documents they create as shown by activity elicited from other nodes [26]. Agarwal et al also tackle the problem of identifying influential nodes by calculating an influence score based on attributes of a node's blog posts such as novelty, quantity of in-links, and post length [1].

Some existing work focuses on information diffusion in certain special cases. Sadikov et al use a k-tree model to estimate properties of a cascade of information, such as size and depth, where the complete cascade is not observable [37]. Another study presents a model for diffusion in a dynamic network based on the classic epidemic model of diffusion [3]. Some networks may contain signed links to depict negative and positive relationships between nodes. Cai et al use bipartite graphs to model the spread of influence in a signed network and demonstrate that link sign has a measurable effect on the nature of influence in a network [7]. Lastly, Kwon et al propose a model of information diffusion based on implicit as well as explicit links in an online social network [24]. They claim that over 85% of information diffusion in online social networks actually occurs without explicit links, necessitating models that address this fact.

Information diffusion models have been used to study network epidemics such

as viruses and rumors. Tripathy et al use the information cascade model as a basis to examine some methods to combat rumors in social networks [40]. Another study proposes methods to vaccinate networks against epidemics in the context of linear threshold model [25].

As pointed out by Goyal et al [19], most information diffusion models assume that influence actually *exists* in social networks; node-to-node influence probabilities are often modeled as uniform distributions or taken as input parameters rather than being based on properties of the network under examination. The authors present methods to calculate influence probabilities based on a log of actions performed by users in the network. Other work has addressed the problem of determining influence probabilities by using expectation maximization in the context of the IC model[38].

2.6 Sentiment Analysis

Sentiment analysis—also known as opinion mining—describes the extraction of subjective content from text. User-generated content on the web has fueled much research in this area as businesses seek to find out what users think of their (or their competitors’) products, political organizations look for ideological trends in the population, and investors try to gauge trends in the stock market. A prominent thread in opinion mining research addresses the sentiment classification problem, where machine learning techniques are used to classify text sentiment as either positive or negative, usually with respect to a particular domain [27]. This general approach points us in the right direction for classifying emotional content in OSNs, but does not give us outright the necessary solution. A significant hurdle to overcome is that the corpus to be classified is domain-independent. Existing techniques have been used to classify sentiment with respect to movies and products [33, 28], for example, but have been unsuccessful at general-domain sentiment classification.

With the rise of applying machine learning techniques to text analysis, as well as with the availability of large corpi of [labeled] subjective texts from online review

sites such as Amazon, Epinions, and IMDB, sentiment analysis and the related areas, opinion mining and subjectivity analysis, have become very active areas of research [32]. Pang et al present an early attempt to apply machine learning techniques to the task of extracting opinions from text. They apply Naive Bayes, entropy maximization, and support vector machine to the now-infamous IMDB movie review data set and assess the relative performance of each method in classifying reviews of movies as positive or negative. Their work highlights the increased difficulty involved in identifying sentiment over identifying topics, most likely due to the subtlety with which sentiment may be expressed in text [33]. Other work in the early 2000's attempts to extract sentiment from stock message boards [12], classify product reviews from websites such as Amazon and Epinions [13, 41], glean product reputations from the web at-large [29], and identify and classify opinions of news items [43, 45]. A primary challenge in the literature concerns which features of a text are best suited for classification purposes. Representing documents as merely a collection of its component words, i.e., the bag-of-words approach, is common due to conceptual and computational simplicity; it requires minimal preliminary analysis beyond parsing documents into words. Some research attempts to improve on this method by utilizing parts of speech and phrase-, sentence-, and document-level level features with varying degrees of success.

An important lesson demonstrated in the existing research in sentiment analysis is the importance of corpus domain, more specifically, that a classifier trained on one domain is largely useless for classification of documents in a different domain [32]. Some studies have investigated this problem in detail and have expanded knowledge and enhanced techniques to overcome it, but a reliable solution remains elusive [4, 34, 23, 6].

Current research in sentiment analysis continues to improve on the basic techniques put forth by early studies, as well as find new applications for them. Paltoglou and Thelwall report improvements in classification accuracy using variants of TF-IDF

term weighting along with machine learning techniques [31]. A method to improve sentiment classification via detection of negation in sentences using conditional random field is proposed by Councill et al with favorable performance results [11]. The popularity of the microblogging site Twitter (www.twitter.com) has elicited numerous attempts to classify sentiments of “tweets” [30, 14, 5].

Sentiment analysis as a service is a nascent business paradigm. Numerous firms including as Sysomos (www.sysomos.com), Autonomy (www.autonomy.com), SAS (www.sas.com), and Corpora (www.corpora.com) now market sentiment analysis features in software and services.

A DIFFUSION MODEL SOLUTION

The objective of this thesis is to determine whether emotion propagates in online social networks, but devising a solution to this problem requires overcoming various challenges. These challenges include the large scale and noise of data from the Web, extraction of emotional content from text, as well as detection of influence propagation in the presence of other sources of influence such as homophily and confounding influence. I address these challenges in this chapter. First I discuss how to use diffusion models to form general framework in which I model emotional propagation contagion. I then describe...

3.1 Influence Propagation and Emotional Contagion

Existing work in influence propagation provides a natural starting point to tackle the problem of emotional contagion in online social networks. Specifically, diffusion models used in the literature provide a framework with which the propagation process in a large social network can be modeled. I conjecture that the phenomenon of emotional contagion follows similar rules of social influence as those governing the propagation of actions such as joining a group or buying a product, albeit with different psychological underpinnings. Whereas the latter may be the result of herd mentality, peer pressure, or pragmatism, the former is likely to be the result of the empathy or sympathy. This assumption permits leveraging diffusion models for the purpose of modeling emotional contagion.

3.2 Diffusion Models as Classifiers

In the preceding section, I suggested that diffusion models are generally applicable to human emotions. Recall that the problem under consideration is whether emotional contagion exists in online social networks. However, diffusion models are generally used to examine the nature of propagation in networks *given* the existence of propaga-

tion. To overcome this issue, it can be useful to employ a diffusion model as a classifier. This is done as follows. First, train the model on a network of bloggers and emotion-labeled blog entries, i.e., learn the influence probabilities of the network. Second, use the model to run a diffusion simulation that predicts the emotions of a test set of blog entries. It should be possible to draw conclusions about the existence of propagation in the network based on the accuracy of the model’s predictions given that they are based on a propagation model. That is, if the model has a high rate of accuracy, it follows that it accurately describes an underlying process of propagation in the network, confirming the hypothesis. However, this is where the absence of ground truth becomes a problem. It is unreasonable to expect that the emotions of all bloggers are subject to external influence, or that all of any single blogger’s posts are subject to external influence. Therefore maximizing absolute predictive accuracy is not a valid objective for the model. In fact, it is unknown what level of accuracy would prove (or refute) the hypothesis. Therefore I use a baseline prediction method for comparison that permits drawing stronger conclusions from the results of the diffusion model’s predictions. I describe the baseline method and the conclusions it supports in detail later in this work.

3.3 Predicting Emotion with a Diffusion Classifier

Having established that there may be a way to employ a diffusion model to test the hypothesis, I now present some details on the proposed method. Given an online social network defined as a graph $G = (V, E)$, where V is the set of users and E is the set of links between the users, there is a set of actions C performed by users in the network. For sake of example, let C be a set of blog entries. Let S be the set of emotions that may be expressed in C . Each entry in C maps to exactly one emotion S . T is the chronologically ordered set of times at which users performed actions. Entries in C are referenced by a tuple (v, t, s) where v is the user, $t \in T$ is the time stamp of the entry, and $s \in S$ is its emotion. A user who has posted a blog entry with emotion s is referred to as “active” or that she has “activated” with respect to s . Given a blog entry created by user

v , a diffusion model trained on a set of past blog entries made by v 's friends is be used to predict the emotion of the current entry. The precise details of the prediction method is discussed later in this chapter. The key aspect of this method that differentiates it from most other work in influence propagation is that the objective is to predict a class label, i.e., an emotion, *given* an action, not to predict the *occurrence* of the action itself. For instance, diffusion models have been used to predict whether some individual will adopt use of a cellular phone and when that might occur. It is not an assumption of this work that emotional influence is great enough to spur someone to, say, post a blog entry in the absence of an existing impetus to do so, although clearly that is a possibility. It is the opinion of the author that the human desire to express emotion is too complex for any network-oriented model to predict precisely when such expression might occur. However, for the purposes of this work I believe it is reasonable to attempt to predict emotion given an instance of expression and inputs from the network.

To formalize the notion of emotional contagion, the following definition is provided for *Propagation of Emotion*.

DEFINITION 3.3.1 *Propagation of Emotion.* We say that emotion $s_i \in S$ is propagated from user u to user v iff (i) $(v, u) \in E$; (ii) $\exists (u, t, s_i), (v, t', s_i) \in C$ where $t < t'$; and (iii) $\neg \exists (u, t'', s_j) \in C$ where $t < t'' < t'$.

For emotion to propagate from user u to user v , three conditions must be met. Condition (i) requires that there must be a link from v to u . I acknowledge the possibility of emotion propagating between users where no explicit link exists, however such cases are ignored in this work for simplicity. Condition (ii) says that user u must perform an action prior to another action with the same emotion performed by user v . It might also be useful to require that links exist prior to t from condition (ii), if link timestamp information is available. For many online networks, link timestamps are not available, so I ignore the possibility that actions in C occur prior to the creation of

links in E . Condition (iii) states that propagation can only occur between user u 's most recent action prior to user v 's action. It is plausible that a action can be influential as long as it is available for consumption by others. Condition (iii) limits the set of actions that must be considered to those which are most likely to have influence, i.e., those that are temporally closest to a future action.

3.4 Selecting a Diffusion Model

Having established why and how diffusion models can be used to examine my hypothesis, I need to discuss which model(s) is (are) most suitable for the task. Four diffusion models were covered in Chapter 2: epidemic, linear threshold (LTM), independent cascade (ICM), and General Threshold (GTM). In this section I review the attributes and behavior of these models and analyze whether they are applicable and useful to assess emotional contagion.

3.4.1 Epidemic Model

A fundamental aspect of the epidemic model is the disease infection cycle that it is intended to simulate. An individual is initially susceptible to the disease, and becomes infected with a certain probability if she comes into contact with an infected individual. After the infection period elapses, the individual becomes recovered and perhaps immune, depending on the disease. If emotional contagion is modeled in this way, it follows that an emotion is only contagious if an individual is susceptible to it. For instance, when a person is happy, they are not susceptible to the influence of another's sadness. While this might be true in some cases, I feel it is too restrictive for an initial study on the topic of emotional propagation, and leave exploration of the epidemic model for future work.

3.4.2 General Threshold, Independent Cascade, and Linear Threshold Models

As discussed in Chapter 2, general threshold model (GTM) generalizes both linear threshold and independent cascade models (LTM and ICM, respectively), so I discuss

GTM first since its attributes also define the other two models. Part of the intuition behind GTM is that in a network, a node's actions are subject to the influence of neighbors given two parameters. The first parameter is edge weight, which models the degree of influence a neighbor has on a node. One might think of neighbors with high edge weights as being close friends or relatives, whose influence is stronger than neighbors with low weights, representing less significant roles in an individual's social network. Edge weight is a suitable parameter for this work to the extent that it models a basic property of human social networks and is likely to be applicable to the emotion domain. The second parameter is the activation threshold θ that is associated with each node to model the minimum intensity of influence (from any or all neighbors) required to cause the node to become active. One might think of this as the amount of peer pressure, either in terms of the number of friends or the amount of direct pressure, an individual can sustain before participating in some activity, such as joining a group that her friends have become members of. Use of edge weights as described above implies use of an appropriate threshold as a basis for gauging the significance of the influence exerted on a node by its neighbors.

The other significant element of GTM is the activation function f_a which defines key behavior of the the model and differentiates its major subclasses of GTM, namely LTM and ICM. I examine the properties of variants of the activation function to determine a suitable implementation for this work.

In all subclasses of GTM, f_a is evaluated at discrete time steps and decides if active neighbors influence a node to become active at a given time stamp. For the purposes of this work, a node is said to become active when it has expressed some emotion, e.g., by way of a blog entry. Given a node v , f_a decides whether v will express the emotion of active neighbors when she posts her next blog entry. Evaluation at discrete time steps is suitable for this work since it is only necessary to calculate influence at the instant a node posts an entry, that is, when emotion needs to be predicted; the value

of the function at continuous intervals between posts is of no concern.

Another property of f_a that must be examined is how it changes with respect to the set of active neighbors. ICM offers a distinct approach with respect to these properties. As described in Chapter 2, when a node v becomes active at time t in ICM, it has one chance to influence its inactive neighbors, that is, to cause them to become active. If any neighbor w does not become active at time t , v cannot influence w at a later time. As the name of the model indicates, ICM permits influence to cascade from a node v to a neighbor w independent of influence from other neighbors of w . This is useful to model strong ties, where a node can be influenced by a single strongly-tied neighbor. In contrast to ICM, LTM says that f_a decides if v becomes active by summing the weights of edges to v 's active neighbors. If f_a exceeds the threshold, θ_v , v becomes active. Defining f_a in such a way means that influence is assumed to be collective over all neighbors of v without specific regard to strong ties. To cover the broadest range of possible real-world situations without overgeneralizing, I employ a method that not only considers the collective influence of a node's neighbors but also emulates strong ties. If edge weights are learned based on the historical influence of neighbors, it can be assumed that edges between neighbors with very strong ties, i.e., are highly influential, will have high weights, while those with weak ties will have low weights. With a suitable activation function, strong ties will exert appropriate influence regardless of the activation status of weak ties. However, weak ties may still collectively exer

3.5 Class Labels for Emotional Prediction

If a diffusion model is to be used as a classifier to predict emotion, it is necessary to have a suitable set of class labels. The domain that is the subject of this work may manifest the spectrum of human emotion: happiness, sadness, anger, anticipation, disappointment, etc. However, as demonstrated by work in sentiment analysis, basic subjectivity is difficult to detect in natural language text, let alone the specific nuances of the myriad emotional states being expressed. Given this fundamental difficulty it is reasonable to

use a limited set of labels to represent the emotions expressed in the texts we seek to analyze. Drawing from the sentiment analysis literature, I reduce the set of emotions to two categories: *negative* and *positive*. Sentiment analysis generally deals with classification of text regarding a subject or object, such as movies, products, or political figures, so using *negative* and *positive* as class labels is natural and intuitive. I believe the class labels used in sentiment analysis can be applied to emotion in general based on a theory from psychology that classifies emotions based on two dimensions: pleasant/unpleasant and calm/excited []. I equate *positive* with “pleasant” and *negative* with “unpleasant” and disregard the the calm/excited dimension for simplicity. This classification scheme is adequate given the novelty of the problem. If it is demonstrable that emotion propagates when taken as values on a one-dimensional scale, it is easy to conduct further study using more emotional dimensions to make a more fine-grained assessment about emotional contagion.

In addition to the class labels mentioned above, I utilize a third class label, *neutral* (to be construed as the midpoint between *positive* and *negative*), to classify text that expresses no discernible emotion or ambiguous or ambivalent emotion. Such texts constitute noise in the attempt to detect the emotional contagion, but it can be considered a valid class label as a control variable for my method. If results show that expressions labeled *neutral* are as predictable as non-*neutral*-labeled expressions, it might indicate that my method is not viable for its intended purpose in this work. A sufficient influence to cause activation when f_a exceeds θ_v , in a sense modeling the effect of peer pressure. Due to the nature of the domain of this work, I ignore the property of ICM that states neighbors have only one chance to influence a node. Instead I consider the actions of neighbors to have influence within some window of time. This is reasonable since the subject of this study entails emotional actions such as blog posts that persist over time and may exert influence for an arbitrary length of time after the time stamp of the action. For instance, a user may encounter a friend’s blog entry one

or more days after it was actually written, at which time the influence of the post is still viable.

3.6 Adapting Diffusion Model for Emotions

Thus far I have discussed various aspects of how a diffusion model can be used to examine the phenomenon of emotional contagion. However, diffusion models such as GTM are not useful for this work without some modification. In the following pages I describe some issues with GTM and solutions thereto.

3.6.1 Nodes as finite state machines

The basic definition of GTM describes node state as binary with respect to a given action: a node is either active or inactive. Also, in most applications of threshold models, node activation is progressive, i.e., once a node has become active, it may not become inactive again. Node state in GTM can be defined as a finite state machine as shown in Definition 3.6.1 below.

DEFINITION 3.6.1 *GTM Node State Machine (with respect to node v)*

$$\Sigma = \{\sigma(\cdot)\} = \left\{ 1 \text{ if } f_a > \theta_v \right\}$$

$$Q = \{inactive, active\}$$

$$q_0 = inactive$$

$$\delta(inactive, 1) \rightarrow active$$

$$F = \{active\}$$

The machine has two states as prescribed: *inactive*, the initial state, and *active*, the final state. Node v transitions from *inactive* to *active* when f_a exceeds the activation threshold, θ_v .

The main problem with GTM as can be seen in the model in Definition 3.6.1 is that node state is binary. Recall that in this work, the activation state of a node, i.e., a user's emotional state, is expressed with respect to some emotion in the set of

emotion labels S . For purposes of modeling diffusion of different emotions, binary activation is clearly insufficient unless $|S| = 1$. Thus where $|S| = n$, a node must have n possible active states to model each possible user emotional state, plus an inactive state. Relatedly, there must be a transition between each pair of states to model the fact that emotional contagion is a perpetual process in which emotions are temporary, and users may experience an arbitrary sequence of emotions over time. This contrasts cases of social influence where the action under consideration represents a (more or less) terminal state, such as joining a group. To model emotional contagion with these requirements, we define a state machine with n active states to generalize the machine in Definition 3.6.1.

DEFINITION 3.6.2 *n-active-state Node State Machine*

$$\Sigma = \{ \sigma(\cdot), \tau \} \text{ where } \sigma(\cdot) = \left\{ \sigma_s \text{ if } f_a(A_v^s) > \theta_v \text{ and } \arg \max_s (f_a(A_v^s)) = s \in S \right\}$$

$$Q = \{ inactive, q^s \}, \forall s \in S$$

$$q_0 = inactive$$

$$\delta(q, \sigma_s) \rightarrow q^s, \delta(q, \tau) \rightarrow inactive, \forall q \in Q$$

$$F = Q$$

As in Definition 3.6.1, the n -active state machine defines the alphabet Σ in terms of a function $\sigma(\cdot)$ which is in turn defined in terms of the activation function f_a . To model the influence of neighbors with respect to each possible emotion, the activation function now takes a parameter A_v^s , the set of a node v 's neighbors that are active with respect to emotion s . The new function is called the *emotion activation function*, and uses the influence probability $\pi_{v,w}$ mapped to each edge (v, w) . The calculation of $\pi_{v,w}$ is discussed in detail later in this chapter. The emotion activation function is monotonic over non-negative values of $\pi_{v,w}$, returns a value between 0 and 1, and can be updated incrementally. Goyal et al provides proofs of these properties [19].

DEFINITION 3.6.3 *Emotion Activation Function.*

$$f_a(A_v^s) = 1 - \prod_{w \in A_v^s} 1 - \pi_{v,w}$$

The machine for node v reads a symbol σ_s whenever emotion s is the *dominant emotion*.

DEFINITION 3.6.4 *Dominant Emotion.* When emotion $s \in S$ is dominant for node v , it means that the activation function $f_a(A_v^s)$ exceeds v 's activation threshold θ_v and is greater than $f_a(A_v^{s'})$ for all $s' \in \{S - s\}$.

In other words, the intensity of influence of a dominant emotion expressed by a subset of a v 's active neighbors $A' \subseteq A_v$ is greater than the intensity of any emotion expressed by any other subset of v 's active neighbors, $A'' \subseteq \{A_v - A'\}$. The alphabet also includes τ , which represents a time window that is the amount of time an active neighbor is influential.

Each active state $q_s \in Q$ in Definition 3.6.2 has a corresponding element in Σ . q_s represents a condition whereby the influence of emotion s was dominant in a previous time step. A transition is made to q^s when σ_s is read, regardless of current state. This transition function models the fact that whenever the influence of emotion s is dominant, the model predicts the user will post a blog entry expressing s , regardless of any previous entries or absence thereof. When the n -state machine reads τ , a transition occurs to *inactive* from any other state. The implications of a transition to *inactive* is discussed in more depth later in this chapter.

For direct application in this work, an *emotion-oriented state machine* is presented. The emotion-oriented state machine is an instance of the n -active-state machine with three active states—one for each emotion label we observe: *negative* (-), *positive* (+), and *neutral* (0).

DEFINITION 3.6.5 *Emotion-Oriented Node State Machine*

$$\Sigma = \{\sigma(\cdot), \tau\} \text{ where } \sigma(\cdot) = \left\{ \begin{array}{l} - \text{ if } f_a(A_v^-) > \theta_v \text{ and } \arg \max_s(f_a(A_v^s)) = - \\ + \text{ if } f_a(A_v^+) > \theta_v \text{ and } \arg \max_s(f_a(A_v^s)) = + \\ 0 \text{ if } f_a(A_v^0) > \theta_v \text{ and } \arg \max_s(f_a(A_v^s)) = 0 \end{array} \right\}$$

$$Q = \{inactive, q^-, q^+, q^0\}$$

$$q_0 = inactive$$

$$\delta(q, \sigma^s) \rightarrow q^s, \delta(q, \tau) \rightarrow inactive, \forall q \in Q \text{ and } \forall \sigma^s \in \{\sigma(\cdot)\}$$

$$F = Q$$

3.6.2 Modeling Influence Decay

To more accurately reflect real-world user behavior, it is desirable to model influence decay. Influence decay supports the notion that the influence of an action such as a blog entry should diminish with time. In online social networks, influence decay may occur as a result of displacement with more recent content (as in news “feeds”), loss of relevance, or by time-oriented displays. For instance, consider a blog entry c made by user w that appears in user v 's blog feed. After some amount of time, it is likely that c will be displaced in the feed by other blog entries made by w or other friends of v . In this way c will eventually lose the ability to influence v .

Two possible approaches for modeling influence decay are continuous time and discrete time. The continuous time approach calculates influence decay as a continuous function of time, e.g., as exponential decay. That is, the influence probability of an action decreases continuously with time. The discrete time approach says that influence probability of an event remains constant for a window of time, τ . Once the time window has expired, the influence probability becomes zero. Compared to continuous time model, discrete time approach has some limitations in terms of predictive power for some variables, but has the advantage of being significantly cheaper to calculate [19]. The disadvantages are irrelevant to our work, so we opt to use discrete time approach.

Discrete-time influence decay is modeled in the node state machine by includ-

ing τ in the alphabet and a transition to *inactive* from any other state when τ is read by the machine. The machine reads τ when the specified time window elapses after node activation. Recall the example given in the previous paragraph where blog entry c is made by user w at time t expressing emotion s . Per discrete-time influence decay, at time $t' = t + \tau$, w is no longer active and is removed from A_v^s . Thus the degree of influence of s on v decreases, lessening the chance that v will post an entry expressing s .

3.6.3 Influence Probability as Emotional Agreement

GTM assigns weights edges to model the probability that a node will be influenced to perform an action by a neighbor. In this work, edge weight $\pi_{v,w}$ indicates the probability that a node v will be influenced by a neighbor w in terms of emotion by way of blog entries. Most existing work on influence propagation assumes edge weights are given and for experimental purposes, assigns them randomly [19]. Since the objective of this work is to study a real-world phenomenon, $\pi_{v,w}$ must reflect the underlying relationship between a pair of users in the network. It is reasonable to assume that a pattern of historical influence between a pair of users can be a predictor of future influence. Therefore $\pi_{v,w}$ is computed in the following way. Consider a chronological sequence of blog entries made by user v and neighbor w . For consecutive entries where the first is made by w and the second is made by v , record a success if and only if $emotion(v, t') = emotion(w, t)$ where $t' \leq t + \tau$ for a nonzero value of τ ; record a failure otherwise. That is, if a blog entry by user v expresses the same emotion as a previous entry by v 's neighbor w and the two entries are within a given nonzero time window, it indicates that v has emotional agreement with w . After all entries have been read, $\pi(v, w)$ is calculated as the ratio of successful trials to the total number of trials with respect to each pair of neighbors. Thus $\pi(v, w)$ expresses the degree to which v is influenced by w . To avoid overstating influence in cases where multiple entries by v fall within the influence window of a single act by w , each act by w is permitted to be

included in exactly one trial.

It should be noted that the method used to compute $\pi(v, w)$ above is a coarse heuristic that is not guaranteed to provide a perfect metric of the degree of influence one user has on another. Any attempt to find such a metric must at minimum conquer with the age-old scientific problem of distinguishing causation (influence) from correlation. One way in which the given method may fail is as follows. Consider the sequence of blog entries in Table 3.1.

t_0	v	<i>positive</i>
t_1	w	<i>positive</i>
t_2	v	<i>positive</i>

Table 3.1: Sequence of blog entries

Assume all entries are posted within the time window τ . The method given for computing $\pi(v, w)$ records a success at both time t_1 and t_2 . However, the fact that user v expressed a positive emotion at t_0 suggests that v was already feeling positive before or at time t_1 , and w 's emotion had no influence on v at time t_2 . A stronger case for the existence of influence might be made if only cases where a user's emotion is altered by a neighbor's emotion are considered. However, it is also feasible that w 's emotion at time t_1 reinforced or enhanced v 's emotion at time t_2 , influencing her to express positive emotion. There are innumerable possible cases we can consider that may either support or contradict any relatively simple computation devised to assess influence between users. Since the objective of this method is to establish influence probability but not to directly predict influence, it is reasonable to assume that given a large enough sample, the various cases average out to a reasonable approximation of the desired metric.

Assuming that a node may influence another's emotion, it is interesting to consider whether the strength of influence depends on the actual emotion under consid-

eration, that is, if node w has more influence on node v , e.g., with respect to negative emotion than to positive emotion (or vice versa), and specifically what the respective influence probabilities are. To attempt to answer this question, influence probability is computed by recording not only whether v agrees with w for a given pair of entries, but also the emotion of the entries. This is done by counting instances of each possible sequential combination ($emotion(v,t'), emotion(w,t)$) in a three-by-three confusion-style matrix called the *agreement matrix* (see Figure 3.1), where the axes are labeled with the set of emotion values, the x- and y-axes correspond to $emotion(v,t')$ and $emotion(w,t)$, respectively, and a cell is incremented when the corresponding sequence of sentiments is encountered.

	neg	neut	pos
neg	0	0	0
neut	0	0	0
pos	0	0	0

Figure 3.1: Agreement Matrix, $M_{v,w}$

For example, consider a sequence of entries made by user v and a friend w at times t' and t , respectively. If $emotion(w,t) = neutral$ and $emotion(v,t') = positive$, the trial is recorded by incrementing $M_{v,w}[2, 1]$. Successful trials are tallied in the main diagonal of $M_{v,w}$; unsuccessful trials are tallied in all other cells. Influence probability with respect to emotion of the *influencing* action $s = emotion(w,t)$, is defined as follows.

DEFINITION 3.6.6 *Emotion Influence Probability*

$$\pi_{v,w}^s = \frac{M_{v,w}[s, s]}{\sum_{i \in S} M_{v,w}[i, s]}, \text{ where } s \in S = \{negative, neutral, positive\}$$

Definition 3.6.6 divides the number successful trials for a given emotion (on main diagonal) by the total number of trials where $emotion(w,t) = s$ (columnar sum).

We can calculate the overall influence probability, i.e, without respect to a particular emotion, by dividing the sum of all values in the main diagonal by the sum of all values in the matrix. Many other calculations are possible and may be useful for analyzing node attributes such as influentiality and influenceability. Investigation of such calculations is left for future work.

3.7 Baseline Prediction Method

It was previously mentioned that the absence of ground truth regarding the existence and extent of emotional propagation in online social networks requires a suitable baseline method to benchmark the predictive accuracy of the proposed method. Since the problem this work attempts to solve is relatively novel, there are few reliable baselines. Zafarani et al use a method of computing the “average” sentiment over a sliding time window to predict propagation with reasonable results [46]. Machine learning techniques such as Naive Bayes Classifiers, Support Vector Machines, and K^* are other possibilities. For ease of computation, a fairly simple baseline method, majority class prediction, is used in this work as follows. First, compute the distribution of emotion labels mapped to blog entries by user v . Then, where a prediction is to be made for user v , predict the majority label from v 's emotion label distribution. This method has the benefit of controlling for the emotional bias that any particular user may have. For instance, if user v 's blog entries are all labeled *negative*, the baseline method predicts *negative* and will achieve 100% accuracy. Thus the accuracy of the diffusion classification method proposed in this work cannot be overstated due to user bias when compared with the baseline. In the opposite case where user v 's distribution is evenly divided among the possible emotion labels, the baseline method performs the same as random prediction, a reasonable lower accuracy bound.

3.8 Algorithms

As with any machine classification method, the solution presented in this work is executed in two stages: a training stage and a testing stage. In the training stage influence

Algorithm 1: Learn Influence Probabilities

input : Blog entry corpus C , user network $G = (V, E)$

output: a set of *agreement matrices* M

```
1 for  $c \in C$  do
2    $u \leftarrow \text{user}(c)$ ;
3   for  $v \in \text{neighbors}(u)$  do
4      $l \leftarrow \text{lastEntry}(v)$ ;
5     if  $\text{timeStamp}(l) < \text{timeStamp}(c) + \tau$  and  $\text{checked}(l) = \text{FALSE}$  then
6        $\text{updateAgreementMatrix}(u, v)$ ;
7        $\text{checked}(l) \leftarrow \text{TRUE}$ ;
8 return  $M$ 
```

probabilities in the form of agreement matrices are computed for all active pairs of neighbors. In the testing stage the emotion-oriented diffusion model is applied using the probabilities from the training stage to make predictions on the emotion labels of blog entries for users whom the model dictates should be influenced by neighbors. The algorithms presented address the challenge presented by web scale data by requiring only pass over the corpus of blog entries each.

3.8.1 Training State: Learning Influence Probabilities

The algorithm used to perform the training stage is shown in Algorithm 1. Using a training subset of the corpus, the outer loop iterates through blog entries sorted by ascending time stamp. For each blog entry c made by user v , the algorithm checks all neighbors of v . If the last blog entry l (if any exists) made by neighbor w has a time stamp within the time window τ of c , the agreement matrix $M_{v,w}$ is incremented as described in Section 3.6.3. The algorithm flags l as it is checked to ensure that it is not used more than once to in the computation of the agreement matrix. Output is a set of agreement matrices for each pair of neighbors in the network. The training algorithm is quite efficient for web scale data sets as it requires only a single pass over the corpus of blog entries. Given a corpus of size n , the time complexity of the training stage is bounded by the size of the corpus and the maximum node degree of the network d , i.e., $O(nd)$.

3.8.2 Testing Stage: Predicting Emotions

As in the training phase, the outermost loop of testing algorithm iterates over a chronologically sorted set of blog entries from a test subset of the corpus. For each blog entry c made by user v , the algorithm checks all neighbors of v . If the last blog entry l (if any exists) made by neighbor w has a time stamp within the time window τ of c , w is added to A_v^s , where s is the emotion label of l . The algorithm flags l as it is checked to ensure that it is not used more than once to in the computation of propagation. After all neighbors are checked a prediction of the emotion label of c is made in the following way. First calculate $f_a(A_v^s)$ for all $s \in S$ per the definition in Section 3.6.1. Predict $emotion(c) = s$ if and only if the following are true.

1. $f_a(A_v^s) > \theta_v$
2. $f_a(A_v^s) > f_a(A_v^{s'}), \forall s' \in \{S - s\}$

Otherwise, no prediction is made and the trial is discarded. Predictions are compared with the emotion labels assigned as described in Section 3.9 to assess the performance of the model. The time complexity of the testing stage is the same as the training stage, $O(nd)$.

3.9 Mood Labeled Corpus

The method presented thus far assumes there is a corpus of documents such as blog entries where each document is correctly labeled with an emotion label. To the knowledge of the author, no such corpus exists with the labels proposed for this thesis: *positive*, *negative*, and *neutral*. Documents in some online social networks consist of completely unlabeled free-form text (Facebook, Twitter) while those in other networks may be labeled with “moods” (LiveJournal, MySpace). Mood labels vary from site to site, but usually consist of mood-oriented words or short phrases a user selects from a list, ostensibly to express her mood, i.e., emotional state, when the document was created. In

Algorithm 2: Predict Emotion Labels

input : Blog entry corpus C , user network $G = (V, E)$, a set of *agreement matrices* M

output: a set of emotion predictions, P

```
1  $P \leftarrow \emptyset$ ;  
2 for  $c \in C$  do  
3    $u \leftarrow \text{user}(c)$ ;  
4   for  $v \in \text{neighbors}(u)$  do  
5      $l \leftarrow \text{lastEntry}(v)$ ;  
6     if  $\text{timeStamp}(l) < \text{timeStamp}(c) + \tau$  and  $\text{checked}(l) = \text{FALSE}$  then  
7       switch  $\text{emotion}(l)$  do  
8         case negative  
9            $w \leftarrow \text{add } w \text{ to } A_v^-$ ;  
10        case positive  
11           $w \leftarrow \text{add } w \text{ to } A_v^+$ ;  
12        case neutral  
13           $w \leftarrow \text{add } w \text{ to } A_v^0$ ;  
14         $\text{checked}(l) \leftarrow \text{TRUE}$ ;  
15    $\text{add predict}(c)$  to  $P$ ;  
16 return  $P$ 
```

some cases mood labels consist of arbitrary text input by the user. Whether or not mood labels are available, natural language processing and machine learning techniques may be used to analyze complete documents and assign emotion labels. However, as mentioned in Chapter 2, current research suggests that existing methods are inadequate for classifying the emotion of domain-independent text. Unless this study is limited to documents relevant to a specific domain, for example movies, music, or particular consumer products, it is unclear that existing methods will classify the document emotion with reasonable accuracy. On the other hand, if it is assumed that a set of mood labels are mapped to the documents in the corpus, the emotion labeling task becomes simpler and potentially more accurate. With the additional assumption that a mood label reflects the sentiment of its respective document, mood labels can be used as a stepping stone between document content and sentiment labels. Since most mood labels belong to a relatively small set of words and short phrases, our sentiment labeling task is reduced in scale from the cardinality of the set of documents in the corpus to the cardinality

of the set of mood labels mapped to the set of documents, which could be a reduction of a few orders of magnitude. Even more significant is the reduction in computational complexity. Instead of analyzing the text of an entire document of arbitrary length to extract sentiment-oriented content, at most a single sentence and at least a single word or emoticon must be analyzed. Given these benefits, I assume that documents in the corpus are mapped to mood labels. Note that this assumption does not limit the ability to map sentiment labels to a non-mood-labeled corpus. It would simply be necessary to perform an additional step to extract implicit mood labels for each document, then proceed with the labeling methods we present in this work.

Assuming that documents in the corpus are labeled with moods is very convenient for the emotion labeling task, but there is another reason to restrict this work to such a corpus. A corpus that is limited to documents with mood labels should, to some extent, be filtered for noise, where noise consists of documents that have little or no emotional content. Clearly having a mood label does not guarantee that a document contains emotional content, nor does the absence of a mood label guarantee that it does not. However, it is reasonable to assume that the presence of a mood label increases the likelihood that a document will have an emotional bent, and constitutes a viable heuristic for corpus noise reduction.

3.10 Methods for Mapping Moods Labels to Emotion Labels

There are various methods that can be employed to map mood labels to the set of emotion labels used in this work, including applications of Normalized Google Distance or SentiWordNet, as well as manual assignment. In the following pages I discuss how these methods could be applied and qualitatively assess which is the most suitable for this work.

3.10.1 *Normalized Google Distance*

Normalized Google Distance (NGD) is a method to quantify the similarity between two terms by using results from queries to the Google search engine[10]. Some research suggests that a sentiment polarity label can be assigned to any term in the following way. First, select a pair of terms that reflect polar opposite sentiments to define the negative and positive poles. These terms are called the “anchors”. For instance, we might pick “good” and “bad” for the positive and negative poles, respectively. Next, calculate NGD between a term w and each of the anchors. Third, map w to the polarity of the anchor term with which w has the smaller NGD value. For instance, if NGD is smaller between w and “good” than w and “bad”, we say that e is “good”. However, in preliminary experiments we found NGD metrics used in this way to be unpredictable and inconsistent for sentiment polarity classification. For instance, using “happy” and “sad” as anchors, this method yielded counterintuitive results by mapping “mellow”, “amused”, “hopeful”, “cheerful”, and “optimistic” to the negative pole and “intimidated”, “stressed”, “sore”, “anxious”, and “disappointed” to the positive pole.

3.10.2 *SentiWordNet*

A second method for emotion labeling is to use a term sentiment database like SentiWordnet [16]. The database must simply be queried using a mood label as the search term, and the database returns a pair of numeric values indicating the degree of negative and positive sentiment the queried term represents. SentiWordnet is easy to use and has the potential be a useful tool for providing sentiment labels, it is not without drawbacks. The SentiWordnet lexicon is based on the WordNet lexicon, which is a subset of entries from a dictionary. As such, it should be possible to acquire sentiment labels for a significant subset of mood labels in the corpus. But clearly SentiWordnet is not suitable for labeling terms that are not normally found in a dictionary, such as multi-word expressions, misspelled words, nonsensical text, and emoticons, or terms that are simply

omitted from the WordNet lexicon. Any term not listed in the database would have to be labeled by some other method. Also, SentiWordnet is not a deterministic classification tool, and terms for which SentiWordnet yields ambiguous polarity values require additional processing. Furthermore, like a dictionary, SentiWordNet contains multiple entries for many words to support different word senses. In order to acquire polarity labels for words with more than one entry, either word-sense disambiguation must be used—a nontrivial task—or somehow the classifications of multiple entries must be aggregated, e.g., by taking the arithmetic mean. The former strategy is undesirable due to complexity, the latter due to the arbitrariness of applying mathematical calculations to the imprecise metric that SentiWordnet provides.

3.10.3 Manual Labeling

The last method considered for emotion labeling is a manual process, which has both advantages and disadvantages. The disadvantages are that manual labeling can be too subjective if not performed by a sufficient number of judges, is limited by the expertise of the judges, and can be time-consuming. A significant advantage is that almost any mood label can be interpreted in terms of emotions reasonably well (preferably by a native speaker of the respective language) if it has any clear meaning at all. Furthermore, it is possible to map a large portion of a typical corpus to a relatively small number of mood labels. For example, from a sample of 942K LiveJournal blog entries, there are 43K unique mood labels after some preprocessing. The top 200 most common mood labels map to over 90% of the blog posts. It is reasonable in terms of time and effort to label 200 mood labels by hand in order to yield around 850K entries for experimentation, so this is the option I choose. Manual labeling is complemented by comparing the manually assigned labels with labels from SentiWordNet, where possible, and resolving contradictions through manual analysis.

Chapter 4

EXPERIMENTS

To test the hypothesis of emotional contagion using the solution described in the previous chapter, experiments are carried out using a data set from LiveJournal. In this chapter, the data set, preprocessing method, and experimental set up are discussed.

4.1 Data Set: LiveJournal

LiveJournal ¹ provides a set of data that is well-suited for this work. It is a web site for personal blogging that incorporates features of social networking by supporting “friendship” between users. A large sample of blog entries serves as the corpus for this work and the friendship links between posting users are used to create the network graph. LiveJournal supports labeling blog entries with a “mood”, which are used to assign emotion labels to entries, enabling execution of the solution described in the previous chapter to ascertain whether emotional contagion occurs.

4.2 Preprocessing

The raw data set used for this work comprises 5 million LiveJournal blog entries from the Spinner3r weblog data set used for ICWSM 2009 ². This data set contains URLs and the textual body of blog entries among other data, but lacks two key data points for this study: mood labels (henceforth, simply “moods”) and posting date. The HTML of each entry was “scraped” to acquire the missing data points. Entries for which moods or posting date was either absent or unobtainable were removed from the corpus. This step eliminated about 75% of the original data set, primarily due to the absence of moods. Entries made by users with no active friends, i.e., friends whose entries were unavailable in the Spinn3r data set, were also removed. Lastly all entries dated prior to 1/1/2008 and after 9/30/2008 were removed.

The initial set of 50K unique moods in the data set was filtered and processed

¹www.livejournal.com

²<http://www.icwsm.org/2009>

to homogenize the corpus and maximize labeling coverage. First extraneous punctuation was removed, such as quotation marks and parentheses, as well as leading and trailing spaces, taking care not to destroy emoticon strings composed partially or entirely of punctuation characters. Since users employ nearly infinite variation when creating moods, this step required several iterations to identify the numerous ways in which meaningful words were obscured. Next all examples were removed that contained moods consisting primarily of characters in non-Latin character sets (Cyrillic was quite common) by inspecting character value ranges. 43K unique moods remained after these processing steps.

Emotion labels were manually assigned to approximately 1200 moods that appeared ten or more times in the filtered set, which covered 94% of entries in the corpus. From the remaining unlabeled moods, about 60 common words were identified that appeared as initial words of multi-word moods that did not alter the overall emotion of the mood. Examples include emphasis words such as “absolutely” and “much”, stop-words such as “the” and “from”, and other modifiers such as “generally”, “kind of”, and “still”. These words were removed, enabling assignment of emotion labels to many more instances where the remaining text matched moods that were mapped to emotion labels in the manual labeling step. Negation with “not” was examined as a possible means of expanding the set of labeled moods, but surprisingly such constructions were not prevalent in the corpus.

Last, *polarity conjunction* was used to process unlabeled moods consisting of conjunctions, that is, those that consisted of two sub-expressions joined with “and”, “&”, or “but”. enabling assignment of existing labels to previously unlabeled moods.

DEFINITION 4.2.1 *Polarity Conjunction.* The polarity of an expression in the form $m_i + w + m_j$, where m_i and m_j are emotion-labeled sub-expressions with respective polarities $s_i, s_j \in \{\text{negative}, \text{neutral}, \text{positive}\}$, and w is a conjunction word in $\{\text{and}, \&, \text{but}\}$,

is evaluated according to the following rules:

- | | |
|---|----------------------------|
| (i) $s_i + s_j = s_j + s_i$ | <i>Commutativity</i> |
| (ii) $s_i + s_i = s_i$ | <i>Identity</i> |
| (iii) $s_i + s_j = s_i$, where $p_j = neutral$ | <i>Neutral Conjunction</i> |
| (iv) $s_i + s_j = NEUTRAL$, where $s_i \neq s_j$ and $s_i, s_j \neq NEUTRAL$ | <i>Cancellation</i> |

Polarity conjunction permitted labeling of a reasonable amount of additional moods without overly subtle or complicated linguistic analysis. After applying polarity conjunction analysis to unlabeled moods, 97% coverage of the corpus of blog entries was achieved. The top 25 mood labels for each emotion label by frequency in the corpus are shown in Table 4.1. The distribution of mood labels in the corpus is the following: *negative*, 39%; *positive*, 48%; *neutral*, 13%.

<i>negative</i>	aggravated, angry, annoyed, anxious, blah, blank, bored, confused, cranky, crappy, crushed, cry, depressed, disappointed, drained, exhausted, frustrated, nervous, nostalgic, pissed off, sad, sleepy, stressed, tired, worried
<i>positive</i>	accomplished, amused, artistic, awake, bouncy, calm, cheerful, chipper, contemplative, content, crazy, creative, curious, determined, ecstatic, excited, good, happy, hopeful, okay, optimistic, pleased, relaxed, thoughtful, Very good
<i>neutral</i>	busy, cold, devious, ditzzy, dorky, drunk, flirty, full, geeky, high, horny, hot, hungry, indescribable, mischievous, More emotions, naughty, nauseated, nerdy, Normal, sick, sore, thirsty, weird, working

Table 4.1: Top Mood Labels by Emotion

4.3 Experimental Parameters

The model takes two parameters: τ , the time window for influence propagation (also for node deactivation), and θ , the node activation threshold. To obtain a suitably broad set of results from my experiments, a reasonably wide range of values for each parameter

was selected, and iterated trials over each combination of values. The minimum interval between blog entry dates in the corpus is one day. I use a two-day increment for τ over a range of 1 to 27 days. Since θ is a probability threshold, its valid range is [0-1]. A range of [0-0.9] in increments of 0.1 is used since no activation probability can be greater than 1.

4.4 Cross Validation

To provide reliable results, it is useful to use k -fold cross-validation when conducting experiments that employ training and testing a model. 10-fold cross validation was run on the model in this work. Typically, the initial step of cross validation involves dividing the set of experimental data randomly into k sets of equal size. Since the data is a chronologically ordered set and must remain ordered for the model to work properly, a slightly different approach must be taken to divide the data into sets. Instead of randomly partitioning the data, the ordered set is simply divided into k subsets of chronologically ordered instances where no subset overlaps another in terms of time stamp. The model is trained on each possible set of nine subsets (in order) and tested on the respective remaining subset. This process is applied over the 140 parameter combinations for a total of 1400 iterations of the model.

4.5 Output

The output of the experiments consists of a record of a trial for each blog entry in the “test” subset of the corpus. The actual emotion label for the entry and the values of f_a^s calculated as of the time stamp associated with the entry are stored. This data is preliminary as it respects the time window parameter but not the threshold parameter. The emotion label for the entry is predicted by applying the threshold parameter in a second step. The performance metric of the model is prediction accuracy, acc_v^M , represented as the ratio of correct predictions to the total number of predictions with respect to user v . The accuracy of the model’s predictions is benchmarked with a baseline prediction method. The baseline method computes the label assigned to a majority of entries made

by user v , and always predicts that label for v 's entries. If all labels occur with the same frequency in v 's entries, baseline accuracy for v , $acc_v^{H_0}$, is set at chance, i.e., $1/3$.

EXPERIMENTAL RESULTS

In this chapter, experimental results are discussed. First an overview of the results is presented, followed by a discussion of some trends in the data, and finally a critical analysis of the results.

5.1 Overview

To analyze the performance of the model with respect to baseline, *marginal accuracy* Δ_v is a primary metric.

DEFINITION 5.1.1 *Marginal Accuracy, Δ_v .*

$$\Delta_v = acc_v^M - acc_v^{H_0}, v \in V$$

The marginal accuracy is simply the difference between the predictive accuracy of the model and the baseline with respect to given user. Table 5.1 displays a summary of selected results. For each set of parameters, the metrics are split into two sets: one for users where Δ_v is positive, i.e., the model outperforms the baseline, and another where Δ_v is negative, i.e., the baseline does better. Each set contains the percentage of users and average Δ_v for the respective category. A count of the users where $\Delta_v = 0$ is also included. The results reveal that the model's predictions are significantly better than baseline, as shown by the *improvement margin*.

DEFINITION 5.1.2 *Improvement Margin.* *The improvement margin is the difference between the fraction of total users where $\Delta_v > 0$ and the fraction of users where $\Delta_v < 0$ for a given set of experimental parameters.*

The improvement margin is 9% in the worst case (where $\theta = 0.0$ and $\tau = 1$) and 40% (where $\theta = 0.9$ and $\tau = 3$) in the best. Furthermore, in all cases the magnitude

Parameters		Users		$\Delta_v > 0$		$\Delta_v < 0$	
θ	τ	All	$\Delta_v \neq 0$	%	$\bar{\Delta}_v$	%	$\bar{\Delta}_v$
0.0	1	10827	5696	31	0.52402	22	-0.42687
0.0	3	14181	7902	37	0.38236	19	-0.28899
0.0	5	14730	8276	37	0.35799	19	-0.25062
0.0	7	14918	8416	37	0.34282	19	-0.23217
0.0	15	15108	8361	35	0.33012	20	-0.20449
0.0	21	15142	8286	34	0.33166	21	-0.20015
0.0	27	15160	8231	33	0.33287	21	-0.19923
0.3	1	10447	5017	33	0.51980	15	-0.45265
0.3	3	13952	7275	38	0.38504	14	-0.29641
0.3	5	14535	7688	39	0.36071	14	-0.25039
0.3	7	14735	7842	38	0.34504	15	-0.22850
0.3	15	14938	7826	36	0.33138	16	-0.19671
0.3	21	14974	7742	35	0.33372	17	-0.19221
0.3	27	14992	7701	34	0.33472	17	-0.19147
0.6	1	7592	2794	33	0.59850	4	-0.48867
0.6	3	11269	5028	41	0.45528	4	-0.25856
0.6	5	12060	5517	41	0.42633	5	-0.21327
0.6	7	12362	5685	40	0.41320	6	-0.18580
0.6	15	12743	5806	38	0.39349	8	-0.16674
0.6	21	12812	5756	37	0.39158	8	-0.16476
0.6	27	12849	5726	36	0.39092	9	-0.16470
0.9	1	5260	1856	35	0.74070	0	-0.43862
0.9	3	9155	3826	41	0.58332	1	-0.21223
0.9	5	10109	4335	41	0.54692	2	-0.18339
0.9	7	10486	4522	40	0.52574	3	-0.17459
0.9	15	10995	4684	38	0.49525	4	-0.18006
0.9	21	11093	4641	37	0.49585	5	-0.18468
0.9	27	11135	4637	36	0.49466	5	-0.18352

Table 5.1: Marginal Accuracy, Model vs. Baseline

of $\bar{\Delta}_v$ is greater where $\Delta_v > 0$ than where $\Delta_v < 0$ by a margin ranging from 0.06715 (where $\theta = 0.3$ and $\tau = 1$) to 0.37109 (where $\theta = 0.9$ and $\tau = 3$).

5.2 Trends

Some trends are observable in the results with regard to the effect of the experimental parameters on model performance. As Figure 5.1 shows, the improvement margin generally increases as τ decreases. This trend supports one of the premises of the model, that of influence decay. Recall that τ is the time window in which user v 's

entry must follow an entry by her friend u in order to be considered to have been influenced by u . If influence decays with time, it is expected that u 's newer entries, i.e., those within a smaller time window of a subsequent entry by v , will yield better predictive accuracy and u 's older entries, i.e., those within a longer time window, will yield worse accuracy. This intuition is supported by the results. However, where $\theta < 0.7$, improvement margin decreases from $\tau = 3$ to $\tau = 1$. It is reasonable to attribute this to the need for a minimum amount of time to pass (perhaps one to two days) before influence is propagated from one u to v , if for no other reason than that it takes v some time to actually browse her friends' entries, for instance.

Another trend is the effect of propagation threshold, θ , on the improvement margin. Specifically, the model's accuracy improves when θ increases, particularly where θ ranges from 0.3 to 0.5. This relationship is a desirable result of thresholding in influence propagation models, and demonstrates the suitability of such models for the purpose of this work. If the model's performance did not respond (or responded unpredictably) to changes in θ , it would indicate that either the function that describes the behavior of user v is not monotonic with respect to link weights as determined by past agreement between v and her friends, or that v is generally unpredictable. Neither of these undesirable cases seem to apply in this work.

5.3 Analysis

While the model clearly outperforms the baseline prediction method, whether or not this outcome supports my hypothesis that emotional contagion exists in online social networks must be considered. First, consider what baseline performance means. As described in Chapter 3, baseline performs with chance-level accuracy at worst and 100% accuracy at best. Since overall, the model always outperforms baseline, clearly it is at least more accurate than random prediction. With optimal parameter values, the model gives a 40% improvement margin over baseline predictions. Considering that the baseline uses a static probability distribution to predict v 's emotion, specifically the

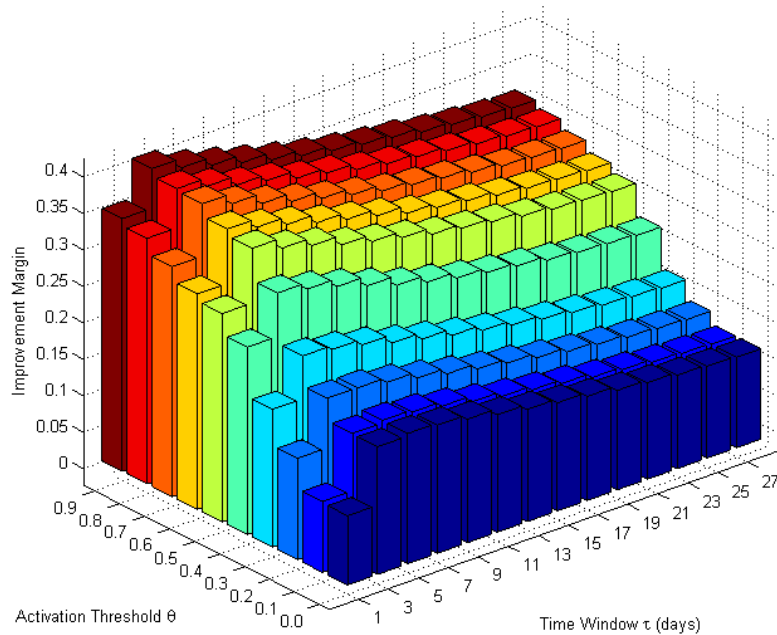


Figure 5.1: Improvement Margin over all parameter values

distribution of emotions actually expressed by v , the strength of the model becomes clearer. The model uses static weights that respect each emotion label and are based on historical agreement between v and each of her friends. These weights are only brought to bear for prediction if the respective friend has expressed some emotion within a sliding time window of a new entry made by v . The fact a model which predicts a user's emotion based essentially on prior activity of her friends outperforms a model which makes predictions based on that user's own activity makes a fairly persuasive case in support of the hypothesis.

Even though the model performs significantly better than baseline, there exist other arguments that may be used to contradict the claim that emotions propagate in online social networks. One such argument is that of confounding influences. Confounding influence refers to an external context that may influence parties who share that context. For instance, people in a particular geographic region may simultaneously experience depression during a long, dreary winter. In theory, such an influence could

affect the results of a study like this one by causing temporal colocation of emotions among friends. Intuitively, it seems likely that confounding factors occasionally exert influence in this way. However, on a large scale, such as among the set of thousands of users and blog entries this work considers, it is highly improbable that confounding influence has a significant effect on the model's ability to predict emotion.

Another argument against the existence influence propagation is homophily. That is, the predictability of user v 's emotions using the emotions of her friends can be explained as a result of existing similarities between v and her friends, instead of as a result of actual influence of v 's friend's emotions on her. A fair amount of research addresses the topic of homophily and influence, providing various methods to differentiate the two phenomena and making differing claims about the extent to which influence actually exists in online social networks with respect to various behaviors. Instead of relying on complex methods to support or refute the existence of emotional contagion, an intuitive argument is offered here. Excluding the minority of clinically depressed and chronically cheerful individuals, the nature of human emotional states is fluid and changeable. It is reasonable to assert that the average person experiences a range of emotions and moods over a sufficiently long period of time. By virtue of the general variability of human emotions, I suggest that homophily is, on the whole, irrelevant to the hypothesis of this work.

CONCLUSIONS AND FUTURE WORK

Motivated by the pervasiveness and, moreover, the importance of social networking sites in our lives in the Web 2.0 era, this work attempted to determine whether emotion spreads through an online social network in order to ascertain whether use of Web 2.0 sites has a real effect on the emotional lives of users. A method was presented which employed as a classifier a novel variant of a diffusion model to predict the emotions of a large set of blog entries based on the emotions expressed in a user's friends' blog entries. The model presented generalizes existing information diffusion models in that the state machine representation of a node is generalized from being binary to having n activation states in order to support an arbitrary set of labels to model emotion diffusion. The model's predictions were compared with a baseline method that always predicts the majority label in a given user's emotion label distribution. The model significantly outperformed the baseline method in terms of prediction accuracy. The experimental results presented make a strong case for the existence of emotional contagion in online social networks in spite of possible alternative arguments such as confounding influence and homophily, since these alternatives are likely to have negligible effect in a large dataset or simply do not apply to the domain of human emotion. Also presented was a hybrid manual/automated method to classify mood-labeled blog entries as positive, negative, neutral which enables the application of the model to a large set (approximately 900K) of blog entries from LiveJournal.

Devising a solution to the problem of detecting emotional contagion required overcoming certain challenges. The domain of Web data introduced issues of large scale, noise, and data set selection and availability. The challenge of Web-scale data was overcome by efficient algorithms which required only two passes over the corpus of blog entries. By using an available set of mood-labeled blog entries from LiveJournal, the issues of noise and data set selection were both minimized. By comparing the

predictions of a diffusion model using a good baseline method, it was possible to make reasonable claims about the existence of emotional contagion in the face of other possible explanations such as homophily and confounding influence. The mood-labeled blog corpus also greatly simplified the emotion classification problem by reducing the set and size of texts to be analyzed.

This thesis makes several contributions to the the field. First, the use of an information diffusion model in the emotion domain is novel and may open up new avenues of research in social network analysis. Second, a novel variant of a threshold diffusion model generalizes node state to support n -class node activation labels (as opposed to existing models with binary nodes). Third, emotion labels (*negative*, *positive*, or *neutral*) were mapped to a large set of mood-oriented words and phrases that could be used in future research in emotion analysis. Last, a strong likelihood that emotional contagion actually occurs in online social networks was empirically demonstrated.

The work presented in this thesis paves the way for further work in several directions. First, it would be interesting to apply the presented method to other sets of data, for instance from other social networks, to ascertain if the results of this work were anomalous or reflect a general trend of emotion diffusion in social networks. Second, it is useful to study the attributes of users in the LiveJournal network or other networks to examine the roles that dominance of influential users and subordination of easily-influenced users play in the diffusion of emotions. Another possible line of research concerns the trends in the improvement margin, as mentioned in the previous chapter. In particular, it would be interesting to find out why the model's improvement over baseline inflects at certain parameter values.

REFERENCES

- [1] Nitin Agarwal, Huan Liu, Lei Tang, and Philip S. Yu. Identifying the influential bloggers in a community. In *WSDM '08: Proceedings of the international conference on Web search and web data mining*, pages 207–218, New York, NY, USA, 2008. ACM.
- [2] Aris Anagnostopoulos, Ravi Kumar, and Mohammad Mahdian. Influence and correlation in social networks. In *KDD '08: Proceeding of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 7–15, New York, NY, USA, 2008. ACM.
- [3] Sitaram Asur, Srinivasan Parthasarathy, and Duygu Ucar. An event-based framework for characterizing the evolutionary behavior of interaction graphs. *ACM Trans. Knowl. Discov. Data*, 3:16:1–16:36, December 2009.
- [4] Anthony Aue and Michael Gamon. Customizing sentiment classifiers to new domains: A case study. In *Proceedings of Recent Advances in Natural Language Processing (RANLP)*, 2005.
- [5] Albert Bifet and Eibe Frank. Sentiment knowledge discovery in twitter streaming data. In Bernhard Pfahringer, Geoff Holmes, and Achim Hoffmann, editors, *Discovery Science*, volume 6332 of *Lecture Notes in Computer Science*, pages 1–15. Springer Berlin / Heidelberg, 2010. 10.1007/978-3-642-16184-1_1.
- [6] John Blitzer, Mark Dredze, and Fernando Pereira. Biographies, Bollywood, boom-boxes and blenders: Domain adaptation for sentiment classification. In *Proceedings of the Association for Computational Linguistics (ACL)*, 2007.
- [7] Keke Cai, Shenghua Bao, Zi Yang, Jie Tang, Rui Ma, Li Zhang, and Zhong Su. Oolam: an opinion oriented link analysis model for influence persona discovery. In *Proceedings of the fourth ACM international conference on Web search and data mining*, WSDM '11, pages 645–654, New York, NY, USA, 2011. ACM.
- [8] Tianyu Cao, Xindong Wu, Song Wang, and Xiaohua Hu. Oasnet: an optimal allocation approach to influence maximization in modular social networks. In *Proceedings of the 2010 ACM Symposium on Applied Computing*, SAC '10, pages 1088–1094, New York, NY, USA, 2010. ACM.
- [9] Wei Chen, Yajun Wang, and Siyu Yang. Efficient influence maximization in social networks. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '09, pages 199–208, New York, NY, USA, 2009. ACM.

- [10] Rudi Cilibrasi and Paul M. B. Vitányi. The google similarity distance. *CoRR*, abs/cs/0412098, 2004.
- [11] Isaac G. Councill, Ryan McDonald, and Leonid Velikovich. What’s great and what’s not: learning to classify the scope of negation for improved sentiment analysis. In *Proceedings of the Workshop on Negation and Speculation in Natural Language Processing, NeSp-NLP ’10*, pages 51–59, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics.
- [12] Sanjiv Das and Mike Chen. Yahoo! for amazon: Extracting market sentiment from stock message boards. In *In Asia Pacific Finance Association Annual Conf. (APFA)*, 2001.
- [13] Kushal Dave, Steve Lawrence, and David M. Pennock. Mining the peanut gallery: opinion extraction and semantic classification of product reviews. In *Proceedings of the 12th international conference on World Wide Web, WWW ’03*, pages 519–528, New York, NY, USA, 2003. ACM.
- [14] Nicholas A. Diakopoulos and David A. Shamma. Characterizing debate performance via aggregated twitter sentiment. In *Proceedings of the 28th international conference on Human factors in computing systems, CHI ’10*, pages 1195–1198, New York, NY, USA, 2010. ACM.
- [15] Pedro Domingos and Matt Richardson. Mining the network value of customers. In *Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining, KDD ’01*, pages 57–66, New York, NY, USA, 2001. ACM.
- [16] Andrea Esuli and Fabrizio Sebastiani. Sentiwordnet: A publicly available lexical resource for opinion mining. In *In Proceedings of the 5th Conference on Language Resources and Evaluation LREC06*, pages 417–422, 2006.
- [17] Facebook. Statistics. [urlhttp://http://www.facebook.com/press/info.php?statistics](http://www.facebook.com/press/info.php?statistics), 2011. Accessed July 18, 2011.
- [18] Robert Gibbons. *A Primer in Game Theory*. Financial Times Prentice Hall, 1992.
- [19] Amit Goyal, Francesco Bonchi, and Laks V.S. Lakshmanan. Learning influence probabilities in social networks. In *WSDM ’10: Proceedings of the third ACM international conference on Web search and data mining*, pages 241–250, New York, NY, USA, 2010. ACM.

- [20] Daniel Gruhl, R. Guha, David Liben-Nowell, and Andrew Tomkins. Information diffusion through blogspace. In *Proceedings of the 13th international conference on World Wide Web, WWW '04*, pages 491–501, New York, NY, USA, 2004. ACM.
- [21] David Kempe, Jon Kleinberg, and Éva Tardos. Maximizing the spread of influence through a social network. In *KDD '03: Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 137–146, New York, NY, USA, 2003. ACM.
- [22] Masahiro Kimura, Kazumi Saito, Ryohei Nakano, and Hiroshi Motoda. Finding influential nodes in a social network from information diffusion data. In *Social Computing and Behavioral Modeling*, pages 1–8. Springer US, 2009. 10.1007/978-1-4419-0056-2_18.
- [23] L.W. Ku, Y.T. Liang, and H.H. Chen. Opinion extraction, summarization and tracking in news and blog corpora. In *Proceedings of AAAI-2006 Spring Symposium on Computational Approaches to Analyzing Weblogs*, 2006.
- [24] Yong-Suk Kwon, Sang-Wook Kim, Sunju Park, Seung-Hwan Lim, and Jae Bum Lee. The information diffusion model in the blog world. In *Proceedings of the 3rd Workshop on Social Network Mining and Analysis, SNA-KDD '09*, pages 4:1–4:9, New York, NY, USA, 2009. ACM.
- [25] Marc Lelarge. Efficient control of epidemics over random networks. In *Proceedings of the eleventh international joint conference on Measurement and modeling of computer systems, SIGMETRICS '09*, pages 1–12, New York, NY, USA, 2009. ACM.
- [26] Seung-Hwan Lim, Sang-Wook Kim, Sunju Park, and Joon Ho Lee. Determining content power users in a blog network. In *Proceedings of the 3rd Workshop on Social Network Mining and Analysis, SNA-KDD '09*, pages 5:1–5:8, New York, NY, USA, 2009. ACM.
- [27] Bing Liu. *Web Data Mining*. Springer Berlin Heidelberg, 2007.
- [28] Jingjing Liu, Yunbo Cao, Chin-Yew Lin, Yalou Huang, and Ming Zhou. Low-quality product review detection in opinion summarization. In *Proceedings of the Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, pages 334–342, 2007. Poster paper.

- [29] Satoshi Morinaga, Kenji Yamanishi, Kenji Tateishi, and Toshikazu Fukushima. Mining product reputations on the web. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '02, pages 341–349, New York, NY, USA, 2002. ACM.
- [30] Alexander Pak and Patrick Paroubek. Twitter as a corpus for sentiment analysis and opinion mining. In Nicoletta Calzolari, Khalid Choukri, Bente Maegaard, Joseph Mariani, Jan Odijk, Stelios Piperidis, Mike Rosner, and Daniel Tapias, editors, *LREC*. European Language Resources Association, 2010.
- [31] Georgios Paltoglou and Mike Thelwall. A study of information retrieval weighting schemes for sentiment analysis. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, ACL '10, pages 1386–1395, Stroudsburg, PA, USA, 2010. Association for Computational Linguistics.
- [32] Bo Pang and Lillian Lee. Opinion mining and sentiment analysis. *Foundations and Trends in Information Retrieval*, 2(1-2):1–135, January 2008.
- [33] Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan. Thumbs up?: sentiment classification using machine learning techniques. In *Proceedings of the ACL-02 conference on Empirical methods in natural language processing - Volume 10*, EMNLP '02, pages 79–86, Stroudsburg, PA, USA, 2002. Association for Computational Linguistics.
- [34] Jonathon Read. Using emoticons to reduce dependency in machine learning techniques for sentiment classification. In *Proceedings of the ACL Student Research Workshop*, 2005.
- [35] Matthew Richardson and Pedro Domingos. Mining knowledge-sharing sites for viral marketing. In *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '02, pages 61–70, New York, NY, USA, 2002. ACM.
- [36] Everett Rogers. *Diffusion of Innovations*. Free Press, 1995.
- [37] Eldar Sadikov, Montserrat Medina, Jure Leskovec, and Hector Garcia-Molina. Correcting for missing data in information cascades. In *Proceedings of the fourth ACM international conference on Web search and data mining*, WSDM '11, pages 55–64, New York, NY, USA, 2011. ACM.
- [38] Kazumi Saito, Masahiro Kimura, Kouzou Ohara, and Hiroshi Motoda. Behavioral analyses of information diffusion models by observed data of social network. In Sun-Ki Chai, John Salerno, and Patricia Mabry, editors, *Advances in*

Social Computing, volume 6007 of *Lecture Notes in Computer Science*, pages 149–158. Springer Berlin / Heidelberg, 2010. 10.1007/978-3-642-12079-4_20.

- [39] Internet World Stats. United states of america, internet usage and broadband usage report. [urlhttp://http://www.internetworldstats.com/am/us.htm](http://www.internetworldstats.com/am/us.htm), 2011. Accessed July 18, 2011.
- [40] Rudra M. Tripathy, Amitabha Bagchi, and Sameep Mehta. A study of rumor control strategies on social networks. In *Proceedings of the 19th ACM international conference on Information and knowledge management*, CIKM '10, pages 1817–1820, New York, NY, USA, 2010. ACM.
- [41] Peter D. Turney. Thumbs up or thumbs down?: semantic orientation applied to unsupervised classification of reviews. In *ACL '02: Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, pages 417–424, Morristown, NJ, USA, 2002. Association for Computational Linguistics.
- [42] Yu Wang, Gao Cong, Guojie Song, and Kunqing Xie. Community-based greedy algorithm for mining top-k influential nodes in mobile social networks. In *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '10, pages 1039–1048, New York, NY, USA, 2010. ACM.
- [43] Janyce Wiebe, Eric Breck, Christopher Buckley, Claire Cardie, Paul Davis, Bruce Fraser, Diane Litman, David Pierce, Ellen Riloff, Theresa Wilson, David Day, and Mark Maybury. Recognizing and organizing opinions expressed in the world press. In *Proceedings of the AAI Spring Symposium on New Directions in Question Answering*, 2003.
- [44] Wikipedia. Diffusion of innovations. [url-http://en.wikipedia.org/wiki/Diffusion_of_innovations](http://en.wikipedia.org/wiki/Diffusion_of_innovations), 2011. Accessed March 23, 2011.
- [45] Hong Yu and Vasileios Hatzivassiloglou. Towards answering opinion questions: separating facts from opinions and identifying the polarity of opinion sentences. In *Proceedings of the 2003 conference on Empirical methods in natural language processing*, EMNLP '03, pages 129–136, Stroudsburg, PA, USA, 2003. Association for Computational Linguistics.
- [46] Reza Zafarani, William Cole, and Huan Liu. Sentiment propagation in social networks: A case study in livejournal. In Sun-Ki Chai, John Salerno, and Patricia Mabry, editors, *Advances in Social Computing*, volume 6007 of *Lecture*

Notes in Computer Science, pages 413–420. Springer Berlin / Heidelberg, 2010.
10.1007/978-3-642-12079-4 52.