EMPLOYING TOPOLOGICAL DATA ANALYSIS ON SOCIAL NETWORKS DATA TO IMPROVE INFORMATION DIFFUSION

Khaled Almgren

Under the Supervision of Dr. Jeongkyu Lee

DISSERTATION

SUBMITTED IN PARTIAL FULFILMENT OF THE REQUIREMENTS

FOR THE DEGREE OF DOCTOR OF PHILOSOPHY IN COMPUTER SCIENCE

AND ENGINEERING

THE SCHOOL OF ENGINEERING

UNIVERSITY OF BRIDGEPORT

CONNECTICUT

May, 2018

EMPLOYING TOPOLOGICAL DATA ANALYSIS ON SOCIAL NETWORKS DATA TO IMPROVE INFORMATION DIFFUSION

Khaled Almgren

Under the Supervision of Dr. Jeongkyu Lee

Approvals

Committee Members

Name	Signature	Date
Dr. Joengkyu Lee		12/10/17
Dr. Navarun Gupta		12/4/17
Dr. Miad Faezipour	Miad Faezipour	12, 4, 2017
Dr. Hassan Bajwa		12/4/17
Dr. Minkyu Kim		12/14/17

Ph.D. Program Coordinator

Dr. Khaled M. Elleithy		2/15/18

Chairman, Computer Science and Engineering Department

Dr. Ausif Mahmood	Ausif Mahmood	2-15-2018

Dean, School of Engineering

Dr. Tarek M. Sobh	T. Sobh	2/15/2018

EMPLOYING TOPOLOGICAL DATA ANALYSIS ON SOCIAL NETWORKS DATA TO IMPROVE INFORMATION DIFFUSION

EMPLOYING TOPOLOGICAL DATA ANALYSIS ON SOCIAL NETWORKS DATA TO IMPROVE INFORMATION DIFFUSION

ABSTRACT

For the past decade, the number of users on social networks has grown tremendously from thousands in 2004 to billions by the end of 2015. On social networks, users create and propagate billions of pieces of information every day. The data can be in many forms (such as text, images, or videos). Due to the massive usage of social networks and availability of data, the field of social network analysis and mining has attracted many researchers from academia and industry to analyze social network data and explore various research opportunities (including information diffusion and influence measurement).

Information diffusion is defined as the way that information is spread on social networks; this can occur due to social influence. Influence is the ability affect others without direct commands. Influence on social networks can be observed through social interactions between users (such as *retweet* on Twitter, *like* on Instagram, or *favorite* on Flickr). In order to improve information diffusion, we measure the influence of users on social networks to predict influential users. The ability to predict the popularity of posts can improve information diffusion as well; posts become popular when they diffuse on

social networks. However, measuring influence and predicting posts popularity can be challenging due to unstructured, big, noisy data. Therefore, social network mining and analysis techniques are essential for extracting meaningful information about influential users and popular posts.

For measuring the influence of users, we proposed a novel influence measurement that integrates both users’ structural locations and characteristics on social networks, which then can be used to predict influential users on social networks. centrality analysis techniques are adapted to identify the users’ structural locations. Centrality is used to identify the most important nodes within a graph; social networks can be represented as graphs (where nodes represent users and edges represent interactions between users), and centrality analysis can be adopted.

The second part of the work focuses on predicting the popularity of images on social networks over time. The effect of social context, image content and early popularity on image popularity using machine learning algorithms are analyzed. A new approach for image content is developed to represent the semantics of an image using its captions, called keyword vector. This approach is based on Word2vec (an unsupervised two-layer neural network that generates distributed numerical vectors to represent words in the vector space to detect similarity) and *k*-means (a popular clustering algorithm). However, machine learning algorithms do not address issues arising from the nature of social network data, noise and high dimensionality in data. Therefore, topological data analysis is adopted. It is a noble approach to extract meaningful information from high-dimensional data and is robust to noise. It is based on topology, which aims to study the geometric shape of data. In this thesis, we explore the feasibility of

topological data analysis for mining social network data by addressing the problem of image popularity.

The proposed techniques are employed to datasets crawled from real-world social networks to examine the performance of each approach. The results for predicting the influential users outperforms existing measurements in terms of correlation. As for predicting the popularity of images on social networks, the results indicate that the proposed features provides a promising opportunity and exceeds the related work in terms of accuracy. Further exploration of these research topics can be used for a variety of real-world applications (including improving viral marketing, public awareness, political standings and charity work).

ACKNOWLEDGEMENTS

My thanks are wholly devoted to God who has helped me all the way to complete this work successfully. I would like to express my sincere gratitude to my advisor Prof. Jeongkyu Lee for the continuous support of my Ph.D study and related research, for his patience, motivation, and immense knowledge. His guidance has helped me to conduct my research and write this thesis. I could not have imagined having a better advisor and mentor for my Ph.D. study. Besides my advisor, I would like to thank the rest of my thesis committee: Dr. Miad Faezipour, Dr. Hassan Bajwa, and Dr. Navarun Gupta, for their insightful comments and encouragement, and also for their constructive feedback, which incented me to widen my research from various perspectives. A special thank goes to Dr. Minkyu Kim for all of his constructive feedback and helpful comments.

I owe a debt of gratitude to my father and mother for helping me to achieve my goals. I would thank my lovely wife (Fahda), brothers, sisters, and the rest of my family for supporting me spiritually throughout writing this thesis and my life in general. I cannot imagine writing this thesis without their support.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1: INTRODUCTION

1.1 Overview

The explosion of social networks has attracted many people from the business, healthcare, and political sectors into using social networks to achieve multiple goals. They can reach out to millions of people through social networks in minutes. People from these sectors are able to maximize their objectives since social networks have changed the way that information is exposed and propagated. A social network can be defined as:

Definition 1: A Social interaction si,j= {ui —> uj} is an action on SN that represent an activity from users up to uj.

Social networks allow users to create profiles, share posts and interact with each other users [1]. For example, users share *images* on Instagram, while users share *tweets* where on Twitter. Posts are propagated using social interactions. Social interactions can be defined as:

Definition 2: A Social interaction $s_{i,j}$= {u_i —> u_j} is an action on SN that represent an activity from users u_i to u_j.

For example, user u_1 can interact with user u_2 on Twitter by *retweeting a post* shared by u_2, where the *retweet* is represented as $s_{1,2} = \{u_1 \longrightarrow u_2\}$.

The simplicity of social networks has made them popular platforms for information diffusion (i.e., information spread), and, as a result, attracted billions of users to generate, consume and propagate content. In 2015, Twitter had a total of 310 million active users, sharing 500 million *tweets* per day [2]. Flicker had a total of 92 million users in 2014, who shared about 1 million images per day and contributed to 2 million groups [3]. Instagram has more than 300 million daily active users, sharing more than 95 million images and videos per day which receive about 4.2 billion *likes* daily [4]. The roles that Facebook and Twitter played during the 2008 U.S presidential elections and the Arabic spring are examples of information diffusion within social networks, where both showed a dominant role in the content generation and propagation [5].

Analyzing social network data can help us understand and improve many phenomena, including information diffusion and influence measurement. Information diffusion on social networks occurs through influencing other users to spread posts. Influence on a social network can be defined as:

Definition 3: Influence on social networks is the ability to make users react to posts of u_i by performing social interaction toward the post of u_i [6].

For example, when users u_1 share a *tweet*, several users can endorse the *tweet* by *retweeting* it. Understanding what influenced the users to *retweet* the *tweet* can improve information diffusion. Influence on social networks can be observed in the form of social interaction, and measuring influence can be used to predict influential users that diffuse posts massively on social networks.

When posts diffuse on social networks, they become popular. Understanding what makes a post popular can also help to improve information diffusion since posts can be diffused based on their content. Popularity on a social network can be defined as:

Definition 4: Popularity is a degree of social interactions that a post is receiving from other users.

By this definition, an image that receives 5,000 *likes* on Instagram is more popular than an image that gets 50 *likes*.

Analyzing social network data can be very challenging due to unstructured, big, noisy data. Therefore, social network mining and analysis techniques are essential for extracting meaningful information about influential users and popular posts.

1.2 Research Problem

On social networks, there exist billions of posts that are created every day by millions of users. Among these posts, only a few diffuse and become popular, while the majority get ignored or little attention. It is our desire to investigate what makes the few posts become popular. It is true that a post has a higher chance of becoming popular if it is posted by an influential user (since influential users can influence other users to perform social interactions). Due to a large number of users, we need to identify the most influential users. Moreover, we observed that not all the posts created by the same user become popular; this motivated us to investigate the influence of the post itself. In addition, the popularity of a post can change over time. Therefore, we predict the popularity of the post over time. By employing machine learning algorithms, we predicted what makes posts diffuse on social networks. However, traditional machine learning algorithms do not address problems arising from the nature of the social networks, such as noisy and high dimensional data, and due to the exponential growth of social networks, it is necessary to develop a technique that analyzes social networks data to predict information diffusion efficiently that addresses these problems. Therefore, a different approach is employed, i.e., topological data analysis, to predict the popularity of images on social networks. Topological data analysis can handle noise and high dimensionality in data, which will be discussed further. The two sub-problems are formalized below:

Problem 1 Predicting the Influential Users: Users can influence others using their structural locations and/or attributes. Social networks can be represented by a graph, where users are denoted by nodes and social interactions are denoted by edges. Since nodes can have attributes (such as activeness) and the structure of the graph can have some meaning, we measure the influence of users to predict the most influential.

Problem 2 Predicting the Popularity of posts: Popularity of posts can be changed over time; it can be increased or decreased. There are many factors that can affect the changes in popularity over time. Therefore, we use social context, content, and early popularity to predict the future popularity of posts. However, the nature of social network data is noisy and high-dimensional, therefore, we investigate a different approach to address these challenges.

1.3 Research Scope

We focused on Instagram and Flickr (two of the most widely used multimedia social networks), Digg (a well-known news social network), and StackOverflow (a social network for programmers) for our data source. For the features that are used in the influential user prediction, we employ users' structural locations on social networks and users' characteristics. We focused on images and employ social context, content and early popularity to predict an image's future. We adopt *likes* on Instagram, *favorite* and *comment* on Flickr, and *vote* on Digg to identify social interactions.

1.4 Motivation behind the research

In this dissertation, we investigate the problem of using social network mining and analysis techniques to analyze social network data to improve information diffusion. We tackle this problem for the purpose of predicting influential users and popular images, which can have a variety of applications (including the improvement of viral marketing, public awareness, the credibility of presidential candidates, or charity campaigns).

Although many researchers have already investigated different problems found in social network analysis, the available approaches that analyze social network data for improving information diffusion are limited. Our idea for predicting influential users is motivated by the fact that using different aspects of influence can play a key role in enhancing the prediction of influential users. Our reasoning for predicting the popularity of images is motivated by the fact that time plays an important role in increasing or decreasing popularity (since popularity can be dynamic). For investigating these two problems, our goal is to improve information diffusion. However, the nature of social network data has many challenges, such as noisy and high dimensional data. Therefore, we further investigate the feasibility of topological data analysis for analyzing information diffusion on social networks.

1.5 Potential Contributions of the Proposed Research

This dissertation has several potential contributions. The potential contributions of the dissertation are summarized as *follows*:

1. A novel influence measurement (based on users' structural location and attributes) is proposed to predict influential users on social networks, which outperforms existing influence measurements in terms of correlation.
2. We investigate the adaptability of the influence measurements to different social networks by employing the influence measurements to Digg and Flicker.
3. We predict the popularity of images by considering the dynamics of popularity over time, using social context, content and early popularity.
4. A novel approach is proposed, built upon well-established techniques in the field of natural language processing (NLP) and clustering; it represents the semantics of images using their captions, which aims to represent multiple meanings from multiple keywords.
5. We investigated the feasibility of topological data analysis for social network analysis and mining since topological data analysis has not been previously investigated for social network analysis and mining to address the issues arising from the nature of social network data by addressing the problem of image popularity on social networks.

CHAPTER 2: LITERATURE SURVEY

Recently, the investigation of information diffusion on social networks has attracted many researchers from computer science because it has real-world applications. To improve information diffusion on social networks, the prediction of the influential users and popular posts needs to be addressed. In this chapter, we provide a comprehensive survey that reviews the state of art. In order to predict the influential users on social networks, we categorize the influence measurements by three folds: (1) model, (2) type, (3) algorithm (as shown in Figure 2.1). We categorize the recent research by three aspects to predict the popularity of images on social networks: (1) social network data types, such text, (2) approach (such as learning models), and (3) problem formalization (such as classification).

2.1 Predicting Influential Users Survey

First, Zafarani et al., proposed Influence measurement by models to categorize influence measurements. Their classification describes the characteristics of the influence measurements, for example, users' attributes such as activeness [7]. The influence measurement models are categorized into observation-based and predicting-based models. However, this classification does not cover the various points of view of those measurements. Therefore, we add two more classifications that consider different points of view. They are Influence measurement by types and algorithms.

For the second classification, the influence measurement by types describes the kinds of structures that are used for measuring influence. This classification includes (i) Context, (ii) Content, and (iii) Hybrid. In the context category, researchers consider only network structure, while in the content category, they consider only content from social networks, such as posted images on Flicker. In the hybrid category, both content and context are used.

The third classification, i.e, the influence measurement by algorithms, consists of the techniques that are used to build the measurements. The algorithms used in this classification include (i) Social Network Measures, such as centrality analysis, (ii) Social Network Properties, such as a number of *tweets* on Twitter, and (iii) Information Cascade Modeling, such as diffusion of content.

2.1.1 Models

Below, we discuss the influence measurement models and their subcategories. They are prediction-based and observation-based models [7].

2.1.1.1 Prediction-based Model

In order to measure influence, the prediction-based model utilizes the structural location of users in a network or users' attributes. This model is classified into location model, attribute model, and location and attribute model [7]. In the location model, the influence of a user is determined by the user's structural location on social networks. This approach uses network measures, such as centrality analysis to measure influence [7].

Several papers study users' influence on Twitter. Weng et al., propose TwitterRank to identify influential users [8]. They define influence as the ability to generate content with interesting topics. They predict influential users based on the topical similarity between users and link structure. Sun and Ng predict influential users based on the interactions of posts [9]. They define influence as the ability to share posts that generate many implicit and explicit interactions. They consider two types of interactions: explicit interactions, i.e., replays and implicit interactions, i.e. posts that talk about the same topic. Cha et al., propose several measurements that are based on different models [10]. One of their measurements defines influence as the ability to attract many users to *follow* influential users. Kwak et al., propose three influence measurements [11]. Two of their measurements are based on the structural location of users. One measurement defines influence as the ability to attract users to *follow* other users. The second measurement defines influence as the ability to attract important neighbors. Maharani et al., use two influence measurements to predict influential users [12]. Their measurements can be defined as users who attract many important users to *follow* them. They build their influence model using undirected relationships between users. Weibo is also one of the social networks that papers have used to measure influence. Li et al., proposes a new measurement based on the user-to-user influence [13]. The user-to-user influence considers four factors that represent four types of interactions on social networks. They define influence as the ability to generate content that generates high *retweeting* strength, *commenting* density, mentioning density, and *tweets* that are similar to the influential user's *tweets*. Liao et al., propose WeiboRank to rank users [14]. They define influence as the ability to attract many important *followers* based on three processes, i.e., *follow*, *repost*, and *comment-only*. They introduce dependence to trace the source of influence. Zhang et al., analyze influence using three social actions, i.e., *following*, *retweeting*, and *commenting* [15]. They define influence as the ability to attract many actions from important neighbors.

Other papers have used Digg, Flicker, and Delicious to measure influence. Ghosh and Lerman predict influential users on Digg [16]. They define influence as the average number of *votes* that each story receives. They state that non-conservative models are the best in predicting influential users. An example of a non-conservative model is information spread. Their methodology shows that users with the most important neighbors are the most influential users. Lu et al., propose LeaderRank to identify influential users on social networks [17]. They define influence as the ability to attract important neighbors to perform interactions such as *voting*. Their measurement is based on the users' structural locations on social networks.

In attribute model, users influence others using their personal attributes. This approach employs network measurements to quantify the influence of each user. For example, a user can be called active if the user has shared many posts. This can be measured by the weight of each node [8]. Leavitt et al., are one of the first research groups to study the effect of users' attributes on influencing users on Twitter [6]. The measurement utilizes the attribute model based on the users' abilities to make other users engage in conversations. Cha et al., propose another measurement that is based on the popularity of users on Twitter [10]. They define influence as the ability to make other users engage in conversations. Anger and Kittl use several influence measurements on Twitter that are based on different models [18]. One of their measurements is based on the popularity of users. They define influence as the ability to attract many users to *follow* the influential users. Another measurement is users' activeness. It is based on the users' contribution level. They define influence as the ability to generate many posts, which can show users' activeness. Erlandsson et al., predicted influential users on Facebook based on their activities; they define influence as the ability to generate a post that attracts many interactions [19]. Ishfaq et al., proposed a measurement to predict influential bloggers [20]; they define influence as the ability to publish many posts that have a positive review and is relatively popular is considered influential. In this paper, the authors proposed a model to predict measurements for identifying influential bloggers [20]; the measurements are based on the sentiments of blogs, the number of posts, and the popularity of users based on how interactive they are. Khan et al., proposed a measurement to rank users on Twitter based on how active they are; they simply ranked users using the number of *tweets* [21]; they define influence as the ability to post many *tweets*. Oro et al., proposed a measurement to detect topical influential users on Twitter and Yelp based on the content of the message and the context of the social network; they define influence as the ability to express opinions on popular topics [22].

2.1.1.2 Observation-based Model

Observation-based models use the amount of influence that each user generates, for example, the number of influenced people, users' ability to spread information, and the power of users to increase the value of products. This approach can be classified into two models: Role Model and Diffusion Model [8].

In role model, the influence of each user is based on the power of users; for example, a teacher can influence students because the teacher is in a position of power.

The teacher's influence can be measured by the number of students [8]. Lee et al., identify influential users on Twitter with the time series of information adoption [23]. They define influence as the ability to generate content that is read by many people. They assume that influence is time-sensitive where users who *tweet* first have a higher probability of becoming influential. They track *tweets* to measure the spread. The user who has many effective readers is regarded as the role model. Sun et al., define influence as the number of effective audience members that users have [24]. The effective audience can be implicit or explicit. The implicit effective audience is the users who *follow* other users and are exposed to their posts. On the other hand, the explicit effective audience is the users who perform interactions toward the influential users' posts.

In the diffusion model, the influence of users is measured by their ability to spread information. The influence is measured by how much the information has diffused on a social network. For example, *tweets* on Twitter can spread if they are transferred among many users in a short period of time. Influence can be measured using the cascade size [8]. Bakshy et al., propose an influence measurement by tracking the diffusion of URLs on Twitter [25]. Influence is defined as the ability to generate a URL that diffuses massively on Twitter. They define the cascade size as the *reposts* of URLs from the user's *followers* and their *followers*. Several papers define influence as the ability to generate content that spread on social networks. They use the diffusion of *tweets* to measure influence [6], [10]-[11], [18], [26].

2.1.2 Types

In this section, we discuss the types of structure that influence measurements are based on. They are classified into the context, content, and hybrid.

Context measurements measure influence using the structural properties of social networks by considering users on social networks and the relationships between the users. Context measurements use the graph theory to present users as nodes and relationships as edges. Two papers analyze influence using the *followers* and friendship networks [11], [18]. Lu et al., measure influence using the friendship network [17]. Cha et al., propose a measurement using the *follower*'s network [10].

Content measurement use content produced by users in measuring influence. In this type, researchers use content in building the influence model such as the diffusion of *tweets* on Twitter. Several works consider the power of generated content by users as an effective indicator of influence, such as the number of *tweets* in different topics and *tweets*' similarity [10]-[11], [18], [21], [23], [26]-[27]. Ishfaq et al., used the users' blogs [20].

Hybrid measurements integrate network structure and content. In this type, the focus is on the dynamic process that takes place on social networks. For example, favorites on Flicker or *retweets* on Twitter. Researchers build the influence model based on the users' posts as nodes and the dynamic processes as edges. Therefore, each user will be represented as a node where the node can be weighted to represent the number of shared posts and a directed edge between two users when they interact through their posts. For example, [16] build the influence model using the users who *vote* on stories on

Digg, where the edges represent *votes* on the images and users as nodes. Several studies propose influence measurements that use *followers* network and content [8], [12]-[15], [22], [24].

2.1.3 Algorithm

In this section, we present three major algorithm types used in identifying influential users. They are social network measures, information cascade modeling, and social network properties. Social network measures are based on social network theory. Information cascade modeling uses the information diffusion theory. Social network properties use existing social network measurements such as the number of *retweets* on Twitter. Network measures fundamentally show the power of users while information cascade modeling shows the power of content

Social networks can be presented as graphs that are comprised of nodes and edges. Nodes can represent actors, where edges represent relationships between actors [25], [28]. Since networks are represented as graphs, several network measurements can be utilized for social networks. There are two main network measurements that have been applied to identify influential users: centrality analysis and network algorithms.

Centrality analysis ranks users by their structural locations on social networks. Centrality represents the importance of users on networks [8], [29]. There are different centrality analysis techniques are used to reflect the importance of standing positions of the users. Centrality analysis techniques will be discussed in more details in the next chapter.

Lee et al., apply PageRank on Twitter [23]. There are many proposed algorithms that are based on PageRank. For example, WeiboRank applies PageRank on Weibo [14]. Zhang et al., use weighted PageRank that combines several interaction types such as *follow* and *retweet* [15]. Yi et al., combine interactions and connections [30]. Li et al., considers four types of interactions [13]. Twitter- Rank uses the topical similarity between users [8]. Ghosh and Lerman propose the normalized α centrality [16]. Normalized α centrality utilizes α centrality [31]. Lu et al., propose LeaderRank [17]. It is similar to PageRank. The difference between them is LeaderRank is parameter-free. Sun and Ng propose a measurement to identify starter posts [9]. They identify starter posts using a modified degree centrality. Starter posts have many *followers* and *follow* very few. Maharani et al., propose using complex degree centrality.

Few researchers used well-known networking algorithms to find the influential users. For example, Sun and Ng identify starters based on the Shortest Path Cost algorithm [9]. The basic idea behind this algorithm is to measure the influence of a node by observing how many other nodes will be affected if that node is removed. Oro et al., employ a three-layer network to model the interactions of users on topics using keywords [22].

In order to measure influence, several researchers use existing social network properties from social networks, such as the number of *retweets* or the number of *tweets* on Twitter. These measurements can reflect many characteristics such as the popularity of users or diffusion of posts. Kwak et al., rank users using the total number of *retweets* on Twitter [11]. This measurement can reflect the popularity of *tweets*. Cha et al., use the total number of mentions, which can show the ability of users to make other users engage in conversations [10]. Anger and Kitll combine several Twitter properties in two measurements [18]. The first measurement is the average number of *followings* over the total number of *followers*. Their other measurement computes the total number of mentions and *Retweets*. Leavitt et al., also propose two measurements that use Twitter properties [6]. The first measurement reflects the spread of content where the second measurement reflects the conversational activities that the *tweet* generates. The first measurement is based on the total sum of *Retweets* and attribution over *tweets*, while the second measurement is the total sum of replays and mentions over the number of *tweets*. Reilly et al., use the number of *tweets* and the number of *Retweets* in identifying influential users [26]. Their measurement considers the diffusion of *tweets*. They consider the percentage of the diffused *tweets* over the users' *tweets*. Ishfaq et al., used the properties of blogs in addition to analyzing the sentiments of blogs using machine learning [20]. Khan et al., used the number of *tweets* provided by Twitter [21].

The theory of information cascade is defined as how information is transferred to users' *followers* and so on [7]. A simple example of information cascade is *Retweet* on Twitter. However, a *retweet* is considered a social network property technique since it uses the *Retweet* property. In information cascade modeling, researchers use the information cascade theory to model and measure influence to identify influential users

Lee et al., identify influential users on Twitter using the adoption of *tweets* based on adoption times [23]. Their model basically considers the users who are first exposed to the users' *tweets,* i.e., effective readers. Since many people *follow* more than one user, each user can *tweet* the same information. They consider the first person who posted the *tweet* as influential. Therefore, early users are more influential.

Bakshy et al., propose an information diffusion model to identify influential users on Twitter [25]. Their model uses the *repost* of users' posts that contain the URLs based on time. Their model uses an influence tree that represents the influential user's post as the seed node and the users who *repost* the URLs as leaves. They do not use *retweet*; they track the actual posts that contain URLs. They measure the cascade size from the total number of users in the influence tree. Their model has two cases when it comes to assigning influence scores to users. The first case gives full credit to the first person who posts the URL in the influence tree. The second case occurs when one user *follows* two people who post the same URL. In this case, the influence score can be given to the last user who posts the URL or it can be divided equally among the users who posted the URL. Erlandsson et al., employed association rule learning to predict the influential users; they have used the number of users and number of posts [19].

Table 2.1. Classification of the state of art by their definitions, models, number of measurements, types, algorithms, and datasets.

Influence Definition	**Influence Model**	**#Measurements**	**Model**	**Type**	**Algorithms**	**Reference**
Average number of *votes*	Graph-based	1	Location Model	Hybrid	Centrality Analysis	[16]
Number of posts that generate many implicit and explicit interactions	Graph-based	3	Location Model	Hybrid	Centrality Analysis, Network Algorithms	[9]
Number of topics posted in *tweets* that further generate many other *tweets*	Graph-based	1	Location Model	Hybrid	Centrality Analysis	[8]
Number of *followers*, number of users that engage in conversations, the diffusion size of *tweets*	Graph-based	3	Location Model, Attribute Model, Diffusion Model	Context, Hybrid, Content	Centrality Analysis, Social Network Properties	[10]

Number of *followers*, number of important neighbors, number of *tweets* that are red by many people	Graph-based, Tree-based	3	Location Model, Role Model	Context, Content	Centrality Analysis, Information Cascade Modeling	[23]
Number of important neighbors that perform interaction	Graph-based	1	Location Model	Context	Centrality Analysis	[17]
Number of important *Followers* based on interaction	Graph-based	1	Location Model	Hybrid	Centrality Analysis	[14]
Number of actions from important neighbors	Graph-based	1	Location Model	Hybrid	Centrality Analysis	[15]
Number of important *followers*, number of *retweets*	Graph-based, Tree-based	2	Location Model	Hybrid	Centrality Analysis, Social Network Properties	[11]
Number of people that are engaged in conversation	Graph-based	1	Location Model	Hybrid	Centrality Analysis	[13]
Number of important *followers* based on interactions, size of *tweets* propagation	Linear-based	2	Attribute Model, Diffusion Model	Hybrid	Social Network Properties	[6]
Number of *tweets* that make users popular or make them active, diffusion size of *tweets*	Linear-based	2	Attribute Model, Diffusion Model	Context, Content	Social Network Properties	[18]
Certain personal attributes and many important neighbors	Linear- based	2	Attribute, Location Model	Hybrid	Centrality Analysis	[30]
Diffusion size of URL on Twitter	Tree-based	2	Diffusion Model	Content	Information Cascade Modeling	[25]
Diffusion size of content	Tree-based	2	Diffusion Model	Content	Information Cascade Modeling	[26]
Number of interactions between users	Graph-based	2	Location Model	Hybrid	Centrality analysis	[12]

Number of effective audience	Graph-based, Tree-based	6	Role Model	Context	Information Cascade Modeling	[24]
Number of posts that attracts many interactions	Linear-based	1	Attribute Model	Content	association rule learning	[19]
Number of positive blogs and how active a user is	Linear-based	2	Attribute Model	Content	Social Network Properties, Machine learning	[20]
Number of posts	Linear-based	1	Attribute Model	Content	Social Network Properties	[21]
Number of posts that express opinions on popular topics	Graph-based	1	Location Model	Hybrid	Network algorithms.	[22]

2.2 Predicting the popularity of social network posts

We reviewed the recent research on the topic of post popularity prediction and categorize the research in terms of the data type, approach, and problem type; these three categories can explain the research in detail.

2.2.1 Social network data type:

In this category, we classify the research studies based on the social network data type that they are focused on (which are text and multimedia, with a focus on images).

Yu et al., tried to predict how many times a *tweet* is *Retweet*ed using user, text, and temporal features [32]. Hong et al., predicted the popularity of *tweets* using *tweet* content, topical information of *tweets*, users, and temporal features [33]. Zaman et al., focused on predicting whether a *tweet* will be *Retweet*ed or not by analyzing the interaction patterns between users, users information, and *tweet* content [34]. They all measured popularity using the number of *Retweets* [32]-[34].

McParlane et al., predicted the popularity of images on Flickr using image content, image content, and user information [35]; they classified the images according to a set number of scenes for the image content. They measured popularity using the number of *comments* and *views*. Khosla et al., predicted how many times an image is *viewed* on Flickr using image content and social context [36]. Cappallo et al., predicted the popularity of images on Flickr and Twitter using images content [37]. The study considered content from both popular and unpopular images; they measured popularity using a normalized number of *views*. Can et al., predicted the popularity of images posted on Twitter and Flickr using hashtags, user information, along with image low and high-level features (such as color) [38]. The researchers measured the popularity of images on Twitter based on the number of favorites and *retweet*, and a number of *views* and *comments* on Flickr. Yamaguchi et al., employed social, content, and text features to predict the popularity of images on Chictopia (a fashion-based social network) [39]. The team measured popularity based on the number of *votes*. Totti et al. classified the popularity of images using aesthetical, semantic and social features on Pinterest [40]; they measured popularity using the number of *repins*. Fiolet classified the popularity of images on Instagram by ranking the popularity of images, using user information and image information [41]. He measured popularity based on the number of *likes*. Niu et al., ranked the popularity of images using network-based modeling on Flickr [42]; the number of *views* was used as a popularity measurement. Gelli used visual sentiments, image content, and context features to predict the normalized number of *views* of images on Flickr [43]. Aloufi et al., used users' information, number of groups that users belong to, number of tags, images' colors, gist, and sentiments to predict the popularity of images on Flickr; they have selected the number of *views* as a popularity measurement [44]. Hu et al., have predicted the popularity of images on Flickr using tag feature and visual features [45]; they selected the number of *views* as the popularity measurement. Mazloomet al., have predicted the popularity of images on Instagram using visual and brand features [46]; they selected the number of *likes* as the popularity measurement.

2.2.2 Approach:

The approaches represent the algorithms or models that researchers use to perform the prediction. The related work either used a learning model (such as machine learning) or a non-learning model (such as network measures).

Yu et al., employed a logistic regression model [32]. Hong et al. used a logistic regression classifier [33]. Zaman et al., employed a collaborative filtering model [34]. Several papers used support vector machine [35]. Several papers employed a regression model based on support vector machine [36]-[37], [43], [46]. Can et al., used a regression model based on linear regression, support vector machine, and random forest [38]. Yamaguchi used regression analysis [39], while Totti et al., used random forest [40]. Aloufi et al., employed support vector machine [44]. Han et al.

On the other hand, some researchers used non-learning models. Fiolet simply ranked the images based on different features, without using any prediction model [41]. Niu et al., employed a weighted bipartite graph model [42].

2.2.3 Problem Type:

Recent works have formalized the problem of popularity prediction in three ways: classification, retrieval, and regression. In regression, researchers try to quantify the popularity of posts; in classification, they classify the popularity to a set of classes (e.g. popular or not popular). In retrieval, researchers rank the images from most to least popular (as in search engine performance evaluation).

Yu et al., formalized the problem as regression [32]. Several papers formalized the problem as a classification problem [34]- [35], [40], [44], [47]. Several other papers formalized the problem as that of retrieval [32], [36]-[37], [41]-[43], [46].

Table 2.2. State of art on image popularity prediction.

Social network data type	**Problem**	**Approach**	**Popularity Measurement**	**Social Network**	**Reference**
Text	Regression	Learning model	Number of *retweets*	Twitter	[32]
Text	Classificati on	Learning model	Number of *retweets*	Twitter	[33]
Text	Classificati on	Learning model	Number of *retweets*	Twitter	[34]
Image	Classificati on	Learning model	Number of *comments* and *views*	Flickr	[35]
Image	Retrieval	Learning model	Number of *views*	Flickr	[36]
Image	Retrieval	Learning model	Number of *views*	Flickr	[37]
Image	Regression	Learning model	Number of favorites and *retweet* (Twitter), and Number of *views* and *comments* (Flickr)	Twitter and Flickr	[38]
Image	Retrieval	Learning model	Number of *votes*	Chictopia	[39]
Image	Classificati on	Learning model	Number of *repins*	Pinterest	[40]

Image	Retrieval	None-Learning model	Number of *likes*	Instagram	[41]
Image	Retrieval	None-Learning model	Number of *views*	Flickr	[42]
Image	Retrieval	Learning model	Number of *views*	Flickr	[43]
Image	Retrieval	Learning model	Number of *views*	Flickr	[44]
Image	Regression	Learning model	Number of *views*	Flickr	[45]
Image	Retrieval	Learning model	Number of *likes*	Instagram	[46]

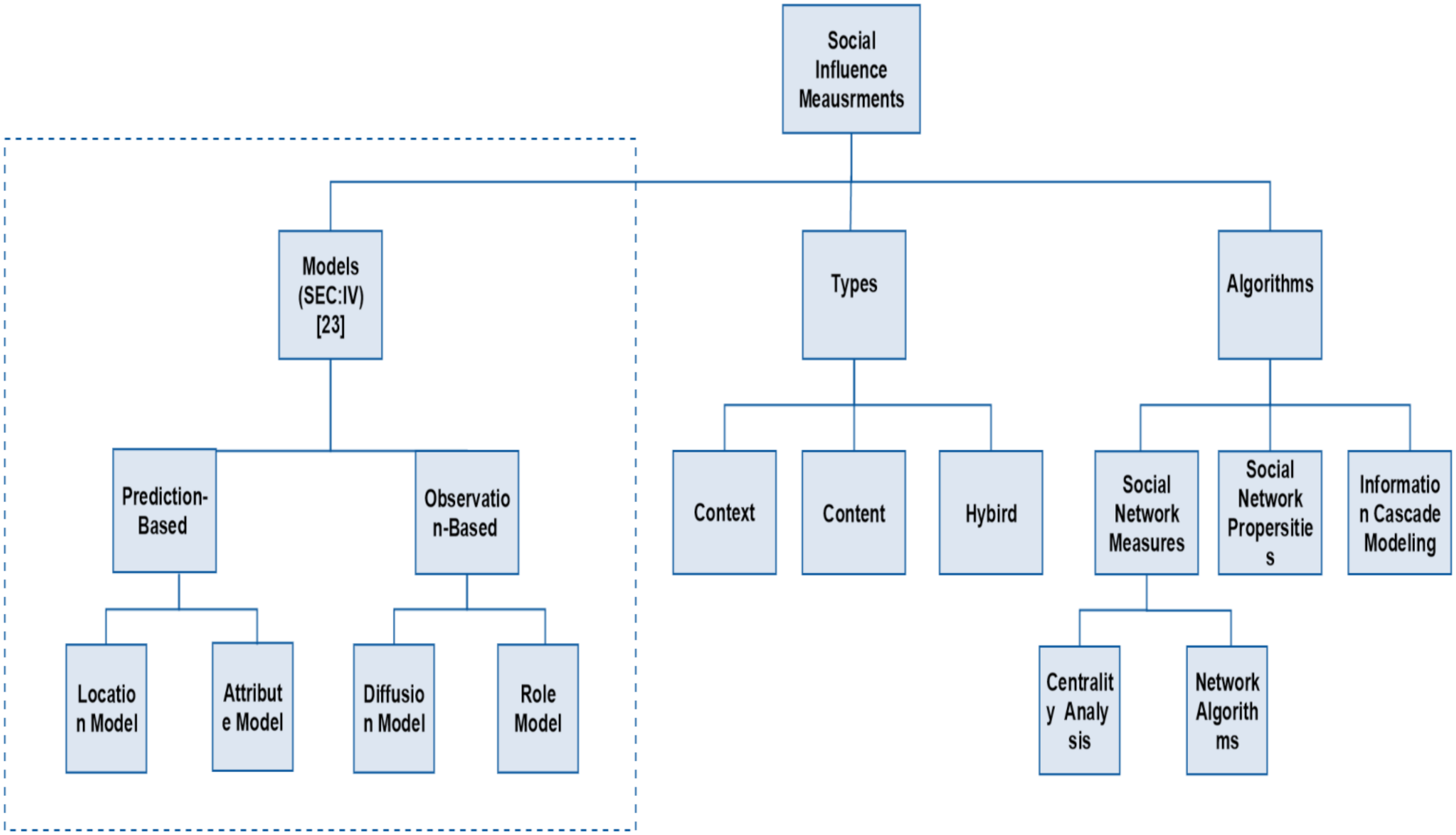


Figure 2.1. Social Influence Measurements Classification.

CHAPTER 3: BACKGROUND THEORY

We employ several approaches several approaches to predict influential users and popular posts (including centrality analysis, Gaussian naïve Bayes, word2vec, and *k*-means). The approaches are used to analyze social network data.

3.1 Centrality Analysis

One of the most important concepts in social network analysis is centrality analysis. In this chapter, we will discuss this approach and explain the different techniques used to analyze the centrality of nodes on social networks.

Graph theory has been previously used to represent social networks. Graphs are used to represent social networks in terms of nodes and edges. Nodes represent users, while edges represent interactions between users. Centrality analysis is used to find important nodes on social networks, based on their structure. Freeman (1978) and Newman (2001) state that centrality is an important attribute of social networks [41]-[42]. There are several centrality analysis algorithms (which include in-degree centrality, weighted in-degree centrality, eigenvector centrality, and page rank centrality).

3.1.1 Centrality Analysis Techniques:

Centrality analysis techniques are categorized as degree-based, distance-based, and network-based techniques. Categories are determined by how the centrality of nodes is computed.

3.1.1.1 Degree-based centrality analysis:

The degree of actors on social networks is considered as an indicator of importance in this type.

Degree Centrality:

Degree centrality (i.e., C_n) considers the number of neighbor nodes as a measure of importance [7]. The idea behind this algorithm is that people who have many direct neighbors are important. The degree can compute using the *follow*ing equation:

$$C_d(v_i) = d_i \tag{3.1}$$

,where d_i is the number of connections to node $_i$.

However, in some cases, a graph can be directed. For example, on Twitter users can *follow* other users but others can choose to not *follow* them back. Therefore, d_i is further divided into $d_i^{(in)}$ and $d_i^{(out)}$, where $d_i^{(in)}$ is the number of direct incoming connections and $d_i^{(out)}$ is the number of outgoing connections; these two measures are also considered centrality analysis techniques (which are called in-degree and out-degree respectively). In the case of weighted and direct graphs, new centrality analysis techniques can be introduced: $d_i^{(wout)}$ and $d_i^{(win)}$, where $d_i^{(wout)}$ considers the weighted outgoing edges, and $d_i^{(win)}$ considers the incoming weighted edges.

Modified degree centrality:

Modified degree centrality, i.e., C_{md}, measures centrality of nodes by computing the difference between in-degree and out-degree for all nodes. It is computed using the *follow*ing equation:

$$C_{md} = d_v^{in} - d_v^{out} \tag{3.2}$$

, where d_v^{in} represents the nodes that point to v and d_v^{out} represents the nodes that v points to.

Complex degree centrality:

Complex degree centrality (i.e. C_c) considers the Probabilistic Partnership Index to compute the centrality of node [12]. It is calculated as *follows*:

$$C_c = (d_v \times wd_v)^{0.5} \tag{3.3}$$

, where wd_v is the weighted degree of A.

3.1.1.2 Distance-based centrality analysis:

The focus is on the distance between actors on the social network of this type.

Betweenness Centrality:

Intermediate people are important in the real world; they connect other people together. C_b ranks node *i* in term of how many nodes i connect. Let ∀ nodes *j* that interact with *s* that go through *i*, where *i* connects *j* and *s* [7]. We compute the number of shortest paths between *s* and *j* that go through node *i*. For example, if Charlie has two *followers*, he would have the highest centrality since he is the only one who can connect all the users. Let $\mathrm{C_b(v_i)} = \sum_{\mathrm{j \neq s \neq i}} \frac{\sigma_{\mathrm{st}}(\mathrm{v_i})}{\sigma_{\mathrm{st}}}$, where $\sigma_{\mathrm{st}}(\mathrm{v_i})$ is the shortest paths between nodes *j* and *s* that pass through *i*, and σ_{st} is the number of shortest paths between j and s. We normalize C_b by the max C_b. To implement this algorithm, we adopted the algorithm proposed in [10].

$$C_b(v_i) = \frac{C_b(v_i)}{\max(C_v(v_i))} \tag{3.4}$$

Closeness Centrality:

Nodes that are located in the middle of the network and not far from other nodes are considered more important, based on this measure. Closeness measures the centrality of nodes by computing the length of' the average shortest path between a node and all the other nodes in a graph [7]. It is normalized by the total number of nodes -1. It is computed as *follows*:

$$C_c(v_i) = \left[\frac{\sum_{j=1}^{N} d(i,j)}{n-1} \right]^{-1} \tag{0.5}$$

, where $d(i,j)$ is the distance between nodes *i* and j, and *n* is the number of nodes in the graph.

In Figure 3.1, the centrality analysis techniques discussed earlier are applied to a few graphs to illustrate the importance of each node. In the example, we compare between the centralities of nodes *x* and *y* (where *x* has a higher centrality than *y*). In the first graph, we apply degree centrality; since degree centrality computes the number of neighbor nodes, *x* has a centrality of 4 where *y* has a centrality of only 1. In the next graph, we applied in-degree (where x has also a centrality of 4, while y has a centrality of

0) since it is based on the number of incoming edges. In the next graph, we applied out-degree centrality. As we can see, *x* has a centrality of 4, while *y* has only 0 (because *x* has four outgoing edges, while *y* has none). When using Betweenness centrality, *x* has a centrality of 1, while *Y* has centrality of 0.86. If using the closeness equation, *x* has a centrality of 0.67, while *y* has a centrality of 0.57 because *x* is more in the middle than *y*.

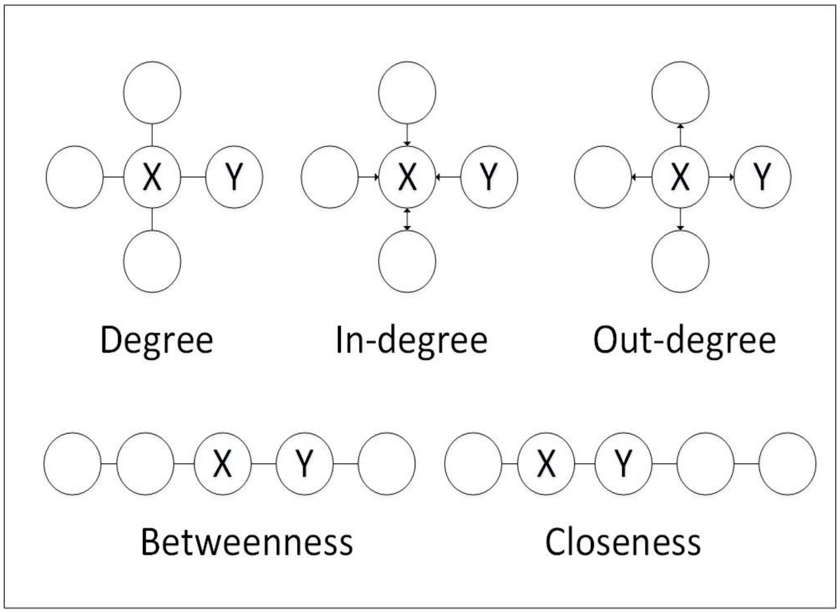


Figure 3 1. Basic centrality analysis techniques.

However, in some cases, the importance of nodes are also related to the importance of the neighbor nodes. For example: if node x has many adjacent nodes that are not important, and node y has few adjacent nodes that are important, then y can be more important than x. Therefore, more centrality analysis techniques are needed to consider the depths of the graphs.

3.1.1.3 Network-based centrality analysis:

The focus is on the structure of the social network as well as actors in this type of analysis.

Eigenvector Centrality:

Eigenvector centrality (i.e., C_e) was the first centrality analysis algorithm that considered the depth of social networks [7]. In eigenvector centrality, the importance of nodes is determined by their neighbors' importance. It is computed using the *follow*ing equation:

$$C_e(v_i) = \frac{1}{\gamma}\sum_{j=1}^{n} A_e(v_i) \qquad (3.6)$$

, where A is the adjacency matrix and γ is a constant representing the largest eigenvalue.

PageRank Centrality:

Pagerank (i.e., C_p) is a variation of eigenvector centrality [7]. Eigenvector centrality encounters some problems. For example, the centrality of nodes is passed to all neighboring nodes, which can make them have the same centralities. This is not efficient because not all the nodes linked to popular nodes are necessarily popular. In Pagerank, the importance of nodes that is passed to neighboring nodes is divided by the number of neighboring nodes. For example, a node is important if it is being pointed by nodes that are also being pointed at by many other nodes. Pagerank is computed using the *follow*ing equation [7].

$$C_p(v_i) = \alpha\sum_{j=1}^{n} A_{j,i}\frac{C_p(v_i)}{d_j^{out}} + \beta \qquad (30.7)$$

, where β is an attenuation constant, and α is a constant used to avoid zero centralities *x*, A is the adjacency matrix, and $\mathrm{d_j^{out}}$ is the out-degree of nodes. However, if the out-degree of any node is null, then $\mathrm{d_j^{out}}$ will be equal to 1.

Normalized α centrality:

Normalized α centrality (i.e., $\mathrm{C_{N\alpha}}$) considers the importance of the incoming neighbors as well as external factors. Bonacich and Lloyd state that the centrality of users do not only depends on their connections, but they also depend on some external factors [31]. Therefore, they proposed α centrality where α represents the importance of endogenous versus exogenous factors. Endogenous factors represent the importance of incoming connections where exogenous factors represent the external factors. α-centrality is calculated using the equation below:

$$C_\alpha = v\ (\textstyle\sum_{t=0}^{k\rightarrow\infty} \alpha^t\ A^t\ ) \qquad (30.8)$$

,where v represents the vector of exogenous factors and $A^t$is the adjacency matrix. Ghosh and Lerman (2010) further normalized this measure by the total C_α ∀ i neighbors [16]. Ghosh and Lerman (2010) further proved that their measurement converges [16].

$$C_{N_\alpha} = \frac{C_\alpha}{\sum_{i,j}^{n}(\alpha^t\ A^t)_{i,j}} \qquad (3.9)$$

,where $\sum_{i,j}^{n}(\alpha^t\ A^t)_{i,j}$ represents the centrality value between j and i.

LeaderRank:

LeaderRank, i.e., C$_{ld,}$ is very similar to PageRank. However, it adds a node to the graph, called ground node, which makes the graph well connected; this makes the algorithm parameter-free. LeaderRank computes the influence score s$_i$ for each node at time t. However, it neglects the score for the ground node. LeaderRank is computed as below:

$$c_{ld}(t+1) = \sum_{j=1}^{N+1} \frac{a_{i,j}}{k_j^{out}} s_j(t)$$

(30.10)

, where $\frac{a_{i,j}}{k_j^{out}}$ represents the random walk of nodes, and a$_{ji}$ represents the directed edge from j to i. Therefore, if the an edge exists, a$_{ji}$ = 1 otherwise a$_{ji}$ = 0. k_j^{out} is the out-degree of *j*, i.e. number of nodes that *j* point to.

For example, in Figure 3.2, using eigenvector centrality analysis, we computed the centrality of the nodes. As shown in the figure, node A has the most centrality with a centrality rate of 0.182, where nodes B, C, and D come next with a centrality rate of 0.091.

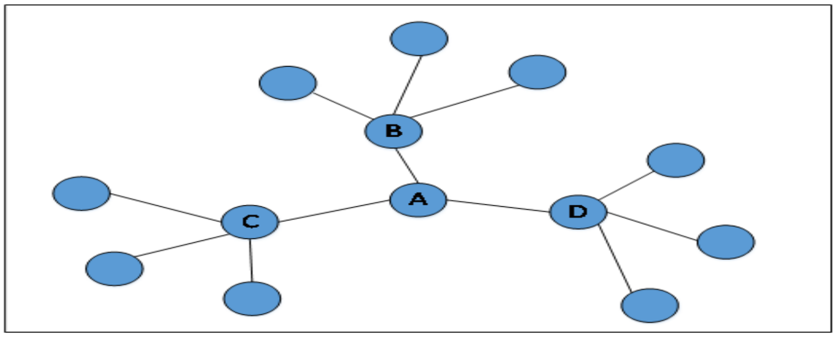


Figure 3.2. Eigenvector centrality analysis technique.

3.2 Naïve Bayes

Naïve Bayes classifiers are a supervised learning probabilistic classifiers that are based on the Bayes' theorem. Bayes theorem describes the probability that an event will happen based on a condition. Bayes' theorem works well with conditional cases. For example, the relationship between a person who has cancer and his/her age [48]. Naïve Bayes can be useful in solving many problems, including natural language processing.

3.2.1 Bayes Theorem

Bayes' theorem is mathematically computed based on the probabilities of two random variables *A* and *B*, and the conditional probability that observing one variable given the occurrence of the other variable [49]. It is mathematically represented as *follow*ed:

$$P(A|B) = \frac{P(B|A)\,P(A)}{P(B)} \tag{30.11}$$

, where P(A) and P(B) are the probabilities of observing the two random variables *A* and *B* regardless of each other, and P(A|B) is the conditional probability of observing *A* given the occurrence of the *B*.

3.2.2 Naïve Bayes Classifiers

Naïve Bayes classifiers are a group of classifiers that are based on Bayes' theorem [48]. Using naïve Bayes, we formalize the problem as a supervised learning problem, where we use a vector *X* to classify a variable *Y*, which is equivalent to the conditional probability of computing the occurrence of *Y* given *X*, i.e., P(*Y*|*X*).

X can be represented as {x_1; x_2 : : : ;x_n}, where x_i is a Boolean random variable in vector *X* and $x_i \in \{-1, 1\}$. For example, let's assume that we want to classify the price of a house as high or low. *Y* can be the classes representing the status of the prices, where *X* can be a vector containing group variables, such as size, and location. Therefore, we can mathematically represent the Bayes rule as:

$$\mathrm{P(Y\ =\ y_i\ |X\ = x_k)} = \frac{\mathrm{P(X = x_k\ |Y = y_i)P\ (Y = y_i)}}{\sum_j \mathrm{P(X = x_k\ |Y = y_i)P\ (Y = y_i)}} \tag{30.12}$$

, where y_m represents the *mth* possible value for *Y*, x_k represents the *kth* possible vector value for *X*, and where the summation is over all values of the random variable *Y*. Now, we can define the conditional probability as *followed*:

$$P(Y = y_i |\ X = x_k) = \frac{P(Y=y_k) \prod_i P(X_k \mid Y=y_k)}{\sum_j P(Y=y_k)\ \prod_i P(X_k \mid Y=y_k)} \tag{30.13}$$

Now to classify a new variable *Y*, we derive the Naïve Bayes classification rule below:

$$Y\ \leftarrow \arg max_{yk}\ \frac{P(Y=y_k) \prod_i P(X_k \mid Y=y_k)}{\sum_j P(Y=y_k)\ \prod_i P(X_k \mid Y=y_k)} \tag{30.14}$$

There are three types of Naïve Bayes classifier: Gaussian Naïve Bayes, Multinomial Naïve Bayes, and Bernoulli Naïve Bayes. Multinomial and Bernoulli Naïve Bayes classifiers work with discrete variables, while Gaussian Naïve Bayes classifier works with continuous variables.

3.2.2.1 Gaussian Naïve Bayes

Sometimes, *X* variables are not Boolean, they are continuous, and the classic naïve Bayes classifier works with Boolean variables. In order to tackle this problem, the

Gaussian Naïve Bayes classifier is proposed [50]. Therefore, using a Gaussian Naïve Bayes, we assume that for each variable Y_i . X_i is represented as a Gaussian distribution, which is defined by a mean and standard deviation, and are needed for training the classifier [50].

$$\mu_{ik} = E[X_i \mid Y = y_k] \tag{30.15}$$

$$\sigma_{ik}^2 = E[(X_i - \mu_{ik})^2 \mid Y = y_k] \tag{30.16}$$

,where μ and σ are the mean and standard deviation. We compute the probability of X as *follows*:

$$P(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{1}{2}(\frac{x-m}{\sigma})^2} \tag{0.17}$$

,where π represent the probability of Y.

3.3 Word2Vec

NLP is a field of computer science that deals with representing human language. It combines artificial intelligent and computational linguistics to process human languages [51]. One of the most complicated problems in NLP is to make a computer program understand the meaning of words. In order to tackle this problem, many researchers use neural networks [52]. Researchers use neural networks to find the most meaningful representation of words by generating numerical vectors. Recently, Word2vec was proposed by researchers from Google [53].

Word2vec is an unsupervised algorithm based on a neural network that aims to learn distributed representation of words. As known before, neural networks take a long time to process text, but Word2vec learns much faster than other neural-network based algorithms [53]. Simply, Word2vec takes the text as an input and generates numerical vectors for words that represent each word in the vector space. When training Word2vec, words against other words that neighbor those words in the input corpus are considered by either considering the context to predict a word or using a word to predict a context. That is accomplished by implementing continuous bag of words and skim-gram architectures. Continuous bag of words is used when using the context to predict the word, while skim-gram is used when a word is used to predict a context. Skim-gram architecture has shown to be more accurate on large datasets. Figure 3.3 shows the two architectures.

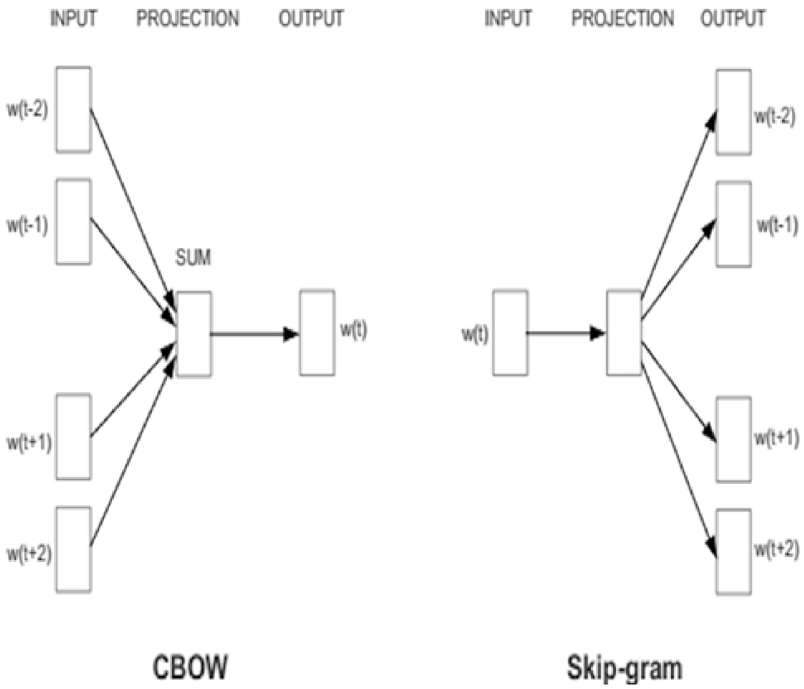


Figure 3.3. Continuous Bag of words and skim-gram architectures [53].

Wrod2vec can be used to solve many problems in NLP, such as semantic similarity detection, next word prediction, sentiment analysis, and word recommendations. Word2vec results were surprising and interesting to many researchers in the NLP field; for example, it detects similarity between two words from different languages that have similar meanings, such as *thanks* and *gracias*, which means thanks in Spanish. Also, it was found to be useful in analogies, such as “man” is to “boy” what “woman” is to “girl” [53]. Since Word2vec represent words by numerical values, we can simply project them in the vectors. For example, as seen as in Figure 3.4, numbers are projected near each other, where animals are projected near each other as well.

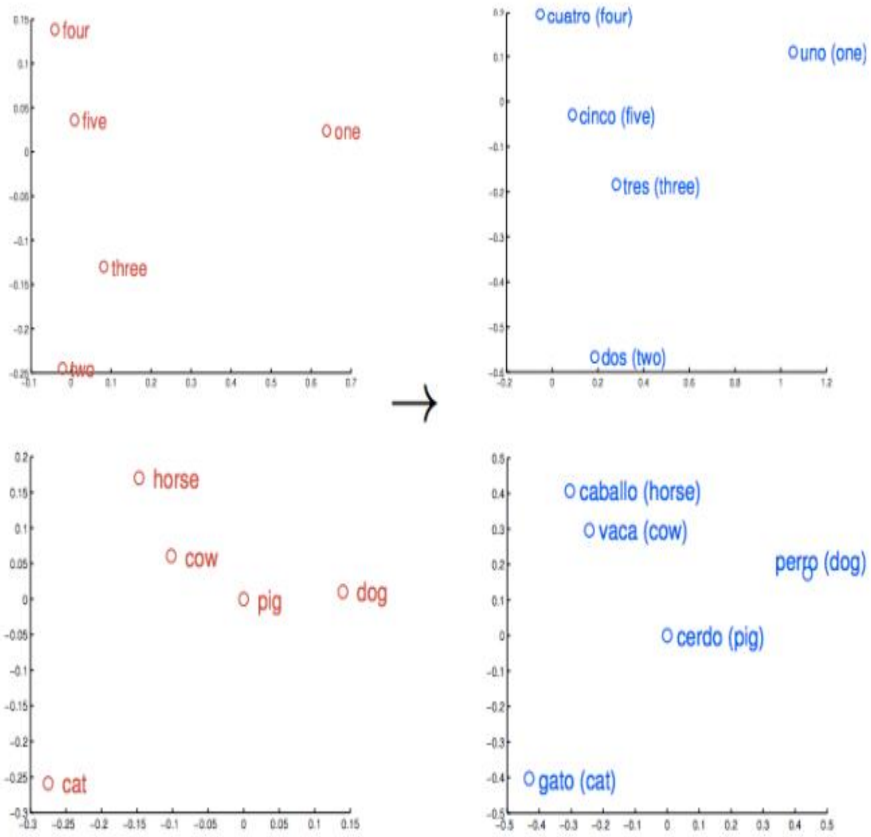


Figure 3.4. Words represented by word vectors generated using Word2vec and projected in the vector space.

3.3 *k*-means

Clustering problems are one of the most classic learning problems. In clustering problems, researchers try to group objects that are similar to each other [54]. One of the most popular clustering algorithms is *k*-means, which is an unsupervised algorithm [55]. The basic idea is to choose a fixed number of clusters and define a centroid for each cluster. At first, centroids are selected randomly. Then, the object is placed within the nearest centroid. After the first round of classifying the objects, the centroids are re-computed based on all the objects in the cluster. Then the objects are assigned to the nearest centroid again. We repeat these steps until no changes exist. *k*-means aims to maximize an objective function, which is computed as *follows*:

$$j = \sum_{j=1}^{k} \sum_{i=1}^{n} \left\| x_i^{(j)} - c_j \right\|^2 \tag{3.18}$$

, where k is the number of clusters, n is the number of objects, x is each object, c is the centroid, $\left\| x_i^{(j)} - c_j \right\|^2$ is the square distance between each object and the centroid. *k*-means can help in analyzing social network data to solve many problems, such as word clustering as shown in Figure 3.5.

To summarize, *k* -means is performed in four steps:

1. Choose *k* clusters and place a centroid in each cluster.
2. Assign each object to a cluster based on the nearest centroid.
3. Recalculate each centroid after assigning all objects.
4. Repeat step 2 and 2, until no changes exist to the centroids.

We employ the discussed approaches to solve the two problems stated in Chapter 3, which are predicting the influential users and popularity of images on social networks.

In order to predict influential users based on the importance of users in terms of network structure, we employ centrality analysis. For predicting the future popularity of images and stability of image popularity on social networks, we employ several features including the content of images, and in order to extract and represent the content of images, we employ Word2vec and *k*-means. For predicting the popularity of images, we use Gaussian naïve Bayes. The two problems are addressed in the *follow*ing two chapters.

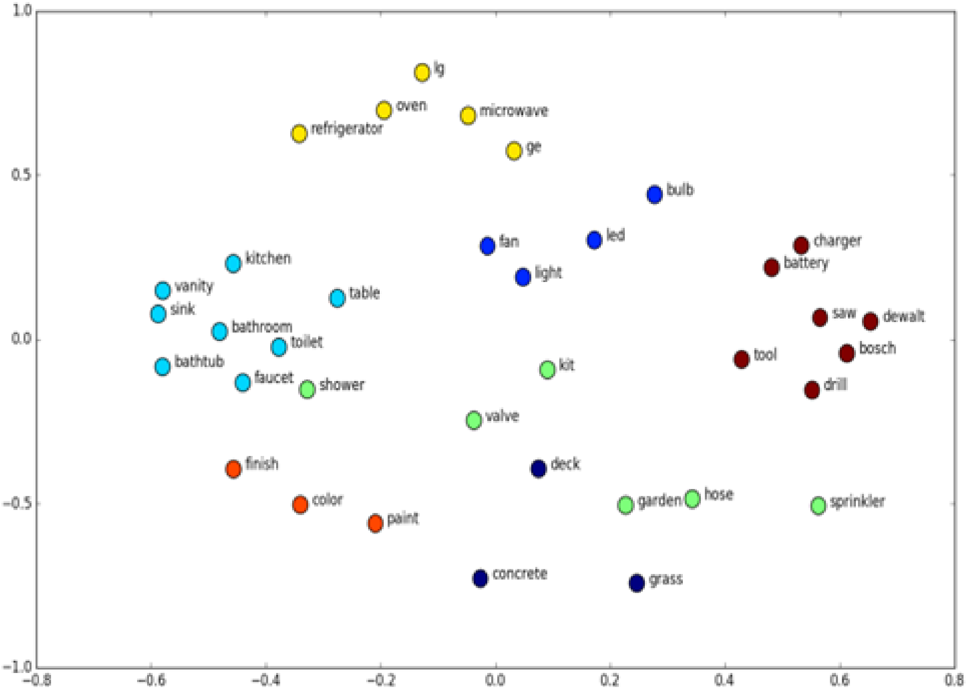


Figure 3.5. Words clustering using k-means.

3.4 Topology

The history of topology goes back to 1736 when Leonhard Euler applied graph theory to the problem of the Seven Bridges of Koenigsberg [56]. The problem of the Seven Bridges of Koenigsberg resulted from the fact that the Pregel River crosses the city of Koenigsberg, which results in four islands. These four islands were connected using seven bridges. Euler wondered if it was possible to walk through Koenigsberg by crossing each bridge only once. Euler collected information about the city and bridges, and then converted the problem to a graph G (V, E) of V (|V| = n) nodes and E (|E| = m) edges as shown in the right diagram in Figure 3.6. By representing the problem as a graph, he observed that it is actually not possible to cross the city by passing the bridges only once. In addition, he discovered that by stretching or squeezing the graph without tearing it apart, the solution is not affected. This was the basis that coined topology.

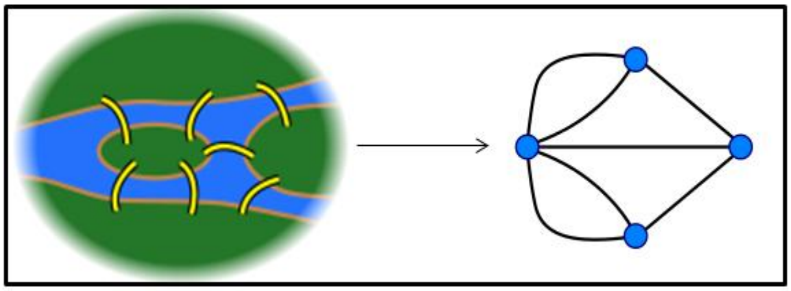


Figure 3.6. The problem of the Seven Bridges of Konigsberg. The left figure represents a map of the city, and the island [57].

Topology is a branch of mathematics that is concerned with qualitative geometric information, e.g., the study of identifying the connected components of a space, more generally connectivity and homology [57]**.** Topology studies the properties of space that are algebraically invariant (i.e., spaces that stay unchanged under any kind of algebraic transformation without tearing or gluing) [58]. Topology has two main tasks: shape measurement and representation. Topology can be defined as below:

Definition 1.

Assume a set X that contains a collection τ of subsets of X. τ is defined as a topology of X if it has the *follow*ing properties)[58] **:**

1. Both ∅ and X are in τ,
2. the union of the elements of any subcollection of τ is in τ, and
3. the intersection of the elements of any finite subcollection of τ is in τ.

If τ is a topology of X, then the ordered pair (X,τ) is called a topological space. Moreover, a subset u of X is called an open set, if u ∈τ. The *follow*ing example illustrates the concept of topology and topological space.

Example 1.

Set X contains three elements, X= {a,b,c}. Many possible topologies τ of X can be found. For example, one topology contains X, and another topology contains X, {{a,b},c} as shown in Figure 3.7.

Points, as well as a set of neighbor points for each point, construct a topological space [59]. Any two topological spaces (E,τE) and (N,τN) have homeomorphism between them if there is a function f that is continuous, one to one, and a bijection between the two spaces. Then, the two topological spaces would have the same topological type and are basically the same in terms of topology. A widely-known example of homeomorphism is the homeomorphism between a donut and a mug. Homology measures connectivity by counting the number of wholes, connected components, faces, and triangles [60] .It can relate a serial of algebraic objects to topological space. A simplex is a topological space made of points, lines, segments, triangles, or their n-dimensional counterparts. A simplicial complex consists of multiple simplexes and complexes as shown in Figure 3.8.

3.4 Topological Data Analysis

Topological data analysis is a set of techniques invented to extract insight from data by studying its shape, which is driven by the fact that data has a shape, and a shape has some meaning [61]. Topological data analysis is based on algebraic topology, a subfield of topology that aims to quantify shapes using persistent homology. Persistent homology is used to compute the topological features, such as holes, components and graph structure, of data at different resolutions by considering different radii from the data points [62] as shown in Figure 3.9. It increases the radius to connect more data points. First, persistence homology must represent the space as a simplicial complex, and then we apply homology to discover the holes in the simplicial complex [62]. The persistent homology concept provides stability and robustness against noise due to the fact that noise cannot be persistent [63].

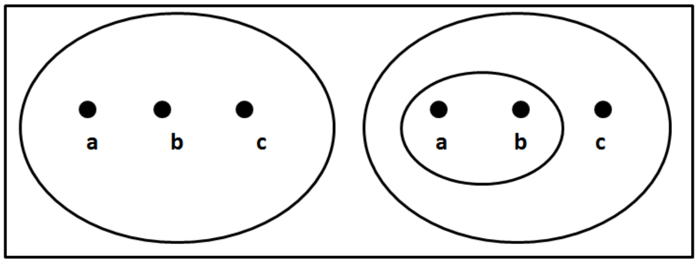


Figure 3.7. Possible topologies of set X. In the left diagram, t = X, while t = X, {{a,b},{c}} is in the right diagram.

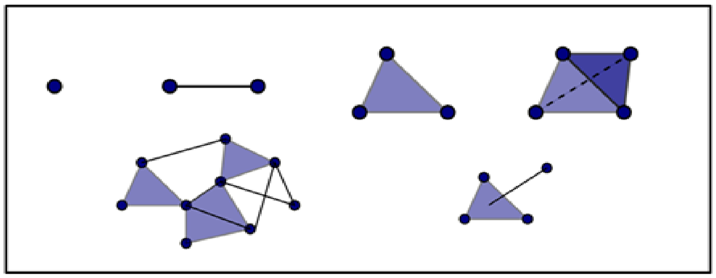


Figure 3.8. The upper diagrams are examples of simplexes: a point, a line, a triangle, and a tetrahedron. The lower diagrams are examples of a simplicial complex: many triangles with many lines and one triangle with one line.

Topological data analysis studies shapes that have three main properties [64]:

1. The shapes are not dependent on specific coordinates,
2. the shapes are not changed under any transformation without tearing the shape apart, and
3. the shapes are produced in a compressed representation that contains infinite distances.

In topological data analysis, high dimensional data in a point cloud is represented by distances, which are one-dimensional information. Therefore, it is independent of the dimensions of the data as shown in the *follow*ing example. This makes topological data analysis a powerful technique to address high dimensional data.

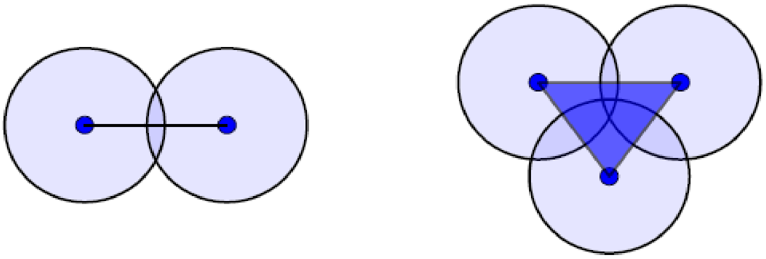


Figure 3.9. Example of how persistent homology increase the radii.

Example 2.

On Twitter, let us have two users called U1 and U2. Each user uses a profile image to represent his/her visual identity. For each user, one vector is used to store the pixels for the user’s profile image, which has 1000 dimensions. For users U1 and U2, we store their images in vectors A and B, respectively. Cosine distance is one metric to evaluate the distance or closeness. The cosine distance between A and B based on their profile images provides the distance or closeness of the two users, which is one-dimensional information.

In order to perform topological data analysis, mapper algorithm is adapted [57], [65]-[66]. Mapper is a method for topological data analysis. The aim of this algorithm is to extract, simplify and visualize high dimensional data. The mapper algorithm takes an inter-point distance matrix ($D \in RN\times N$, where N= the number of data points) as the input. As for the parameters, users specify f, called a filter function in mapper, (which is computed for each data point and used to partition the data, such as density estimation), clustering algorithm (such as hierarchical clustering), and a cover method that is responsible for dividing the filter function output ranges of data points into intervals by specifying the number of intervals S, and overlap ratio p. Here, the overlap is needed to determine connectivity between two intervals in topological data analysis. All data points in one cluster are in the same interval. All data points in one interval, however, are not necessarily in the same cluster.

Mapper generates a simplicial complex that represents clusters of data points and the relationship between them. The simplicial complex consists of nodes and edges. Each cluster is represented by a node, while edges represent the connectivity between the clusters (if $p\neq0$). Clustering algorithms are used to move from a topological version to a statistical one, where mapper is not dependent on a specific clustering algorithm. A summary of the mapper is presented below; for an in-depth description, refer to [66].

Let $U = \{U\alpha\}\ \alpha\in A$ be a finite covering of the space X, so that set A is finite. We define the simplicial complex N {U} whose vertex set is the indexing set A, and where a family $\{\alpha0,\ \alpha1,..,\ \alpha k\}$ spans a k-simplex in N(U) if and only if corresponding clusters have a point in common. It is necessary to generate reference maps $f: X\rightarrow Z$, where X is a given point cloud and Z is the reference metric space. With the reference maps, subsets $X\alpha=f^{1}U\alpha$ are constructed. Different filters can be used: density estimation, eccentricity, and graph laplacians [66].

A simple example of a circle using mapper is shown in Figure 3.10. The left figure is a point cloud of a circle with random variation, X, and the right figure is the simplicial complex of the point cloud, N (U). We arbitrarily selected four levels for this example. The colors represent how filtered the data are. In this example, density estimator is used for filtering the data (red being the densest and blue being the least dense). Edges show the connectivity of clusters of the point cloud. If this is an example of the image popularity analysis, then the left figure is a point cloud of the social media image dataset, and the right is the clustering output of the image dataset from mapper. In addition, the output of mapper can be interpreted in such a way that the shape of the point cloud is a circle, and closer clusters may have higher similarity.

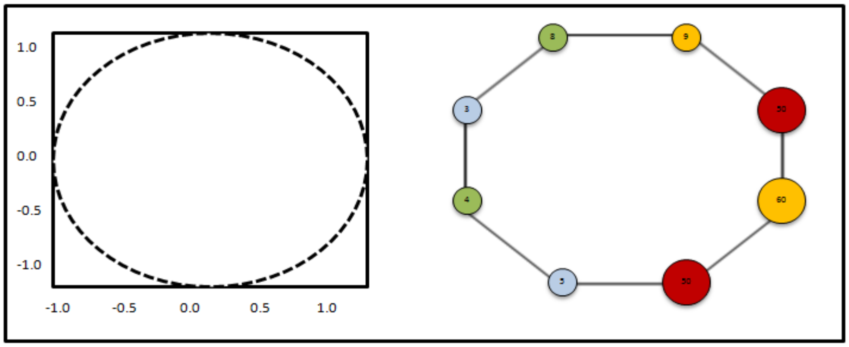


Figure 3.10. A unit circle and the result of topological analysis using a mapper. The size of the nodes on the right indicate the size of the cluster, and the numbers written inside the nodes indicate the number of data points.

CHAPTER 4: PREDICTING THE INFLUENTIAL USERS

In this chapter, we discuss the approaches used to predict influential users on social networks. We further explain the experiment and display the results.

4.1 Overview

In social studies, social influence is defined as "change in an individual's thoughts, feelings, attitudes or behaviors that result from interaction with another individual or a group [67]." Therefore, sociologists have been studying social influence for a long time because it is very important in decision making and information spread [68]. Katz states that influence is related to three main values: finding the personification of certain values, competence, and the position in a strategic social location [69]. The first value is based on the people's attributes, the second value is about people's knowledge and experiences, while the third value is represented by people's social locations in a group [69]. Because of the availability of social network data, measuring influence on social networks can be used for predicting influential people. In addition, most of the social networks provide their own APIs that can provide users with simple access methods, such as Flickr API[1] and Twitter API[2].

[1] https://www.ftickr.com/services/api/

[2] https:lldev.twitter.com

Influential people are users on social networks who attract many people to their posts on social networks using social network interactions including *tweets* on Twitter, and *photos* on Flickr. Users can interact with each other by performing social interactions such as *retweet* and *follow* on Twitter, and *comment* and *favorite* on Flickr. Influential users can be employed in many useful applications including viral marketing, recommendations systems, and expert search engines [9]-[11], [23], [56], [70]-[74]. As mentioned in Chapter 3, we are going to predict the influential users using their structural location and attributes.

4.2 Approach

Predicting influential users on social networks encounter several challenges. First, current measurements do not consider all the characteristics of social networks. Second, influence measurements are not feasible for all social networks. To address the first challenge, we consider two aspects of social networks i.e., user's structural location in a network and attributes. We do not build our measurements on specific social network properties.

The third challenge is the absence of ground truth data [73]. To address this issue, Gosh and Lerman proposed an empirical estimate of influence that is used as ground truth data [16]. They state that the average number of *votes* can estimate the users' influence effectively as shown below:

$$Estimate\,of\;\;Influence = \frac{Number\,of\;\;social\,interactions}{Number\,of\;\;neigbour\,nodes} \tag{4.1}$$

They show the statistical significance of their measure using the URN model [16]. We adopt their approach as a ground truth. To measure influence, we propose a hybrid measurement to predict influential users. This measurement integrates:

(1) Users' structural location in a network, and

(2) Users' attributes.

The structural location in a network can be computed using centrality analysis. On the other hand, users' attributes on social networks such as users activeness are counted for measuring influence because users' attribute is one of the three influence types of the Katz communication model [75]. In order to improve the performance of influence measurement, we employ different centrality analysis algorithms to our measurement and integrate it with users' attributes to predict the influential users. In order to model influence, we use the graph theory since social networks can represent as a graph.

4.3 Influence Model

Influence on social networks is modeled using the graph theory. In a graph G(V, E), we can represent users in social networks as nodes V, and interactions as edges E. Edges can be either directed or undirected and either weighted or unweighted depending on the characteristics of social networks. For example, *retweet* on Twitter is directed while *friendship* on Facebook is undirected. In a graph, influence flows between users through edges. A graph can be denoted using adjacency matrix or adjacency list [76]. As shown in [77], social networks can be sparse networks. Therefore, adjacency lists are more efficient to represent social networks in terms of time and space because of the *follow*ing explanation. In adjacency lists, only the node that points to other nodes will be stored in row cells. For example, there are three nodes *a*, band *c* where *a* points to band *c* and *b* points to *c*. In the first row, there are *a*, *b* and *c*. In the second row, we have band *c*, while in the third row, we have the only *c*. This example shows how space is utilized effectively as shown in Figure 4.1. In this work, we sort the adjacency list using MergeSort [76].

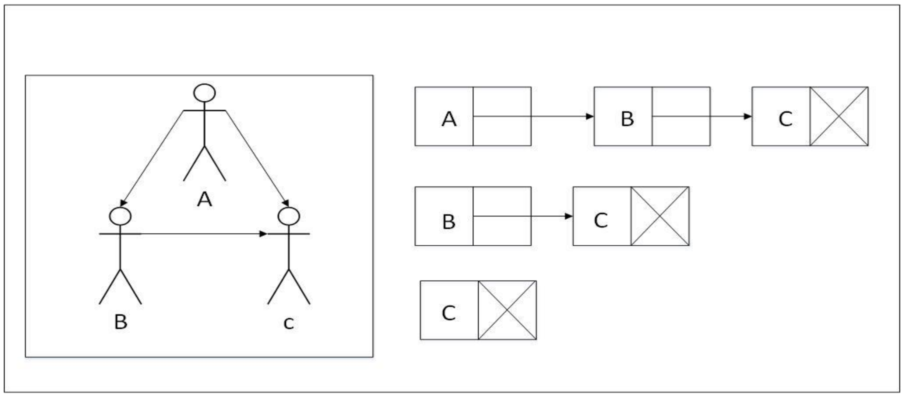


Figure 4.1. A representation of social network using adjacency list.

4.4 Influence Measurement

As mentioned before, influence can be measured using structural location and users' attributes. In this section, we explore the different measurements used in predicting influential users.

4.4.1 Structural location of users on social networks

Different centrality analysis techniques are used to reflect the importance of the standing positions of the users. Overall, all these centrality analysis techniques can be used to show the importance of nodes in a graph. Therefore, to represent the structural location of users on social networks, we propose a user location module that is used to measure users' structural locations using centrality analysis. The user location module applies one of the centrality analysis techniques to the influence model to measure the influence of users. Note that the influence model represents the actual interactions between users as a graph. Figure 4.2 shows a representation of the module.

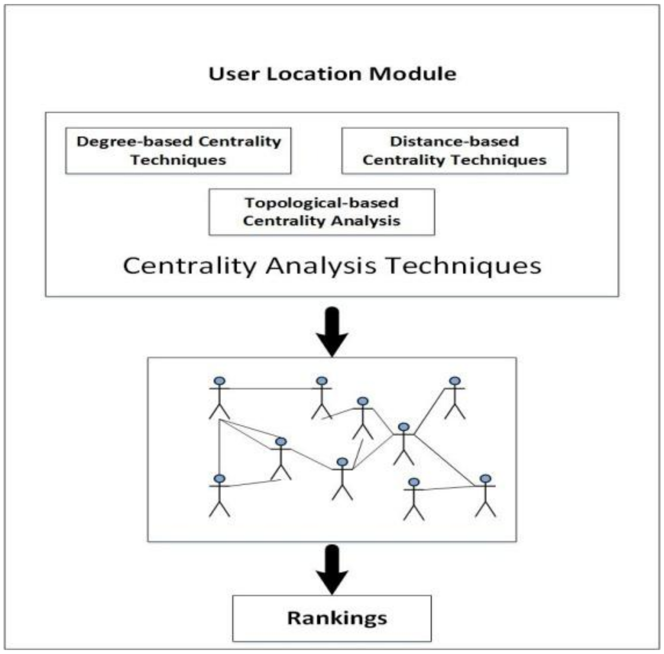

User Location Module
Degree-based Centrality Techniques
Distance-based Centrality Techniques
Topological-based Centrality Analysis
Centrality Analysis Techniques
Rankings

Figure 4.2. User Structural Module.

4.4.2 Users Attributes

In this module, attributes are used to measure the nodes influence such as activeness as shown in Figure 4.3. We use weights to consider the importance of each attribute, i.e., User Attribute Module [78]. The equation below is used to compute the influence of nodes:

$$UI(v_i) = \sum_{j=0}^{n} att_i \times w_i \tag{4.2}$$

, where $\sum_{i=0}^{n} w_i = 1$. Each w_i will be assigned to each attribute denoted as att_i.

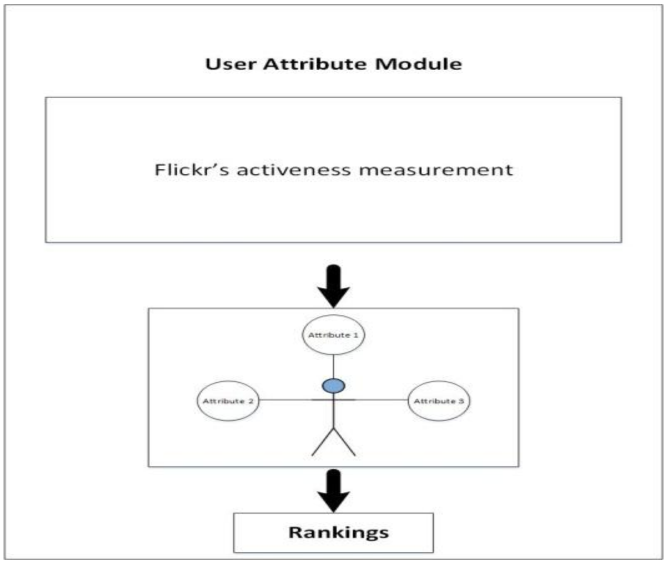


Figure 4.3. User Attribute Module.

4.4.3 Users Structural location and attributes

In this module, we integrate the users' structural locations and attributes to compute a node's influence on a graph as shown in Figure 4.4.

We use a parameter τ to controls the relative importance of the two measurements. It is computed as below:

$$IM_i = \tau \times UI_i + (1 - \tau) \times US_i \tag{4.3}$$

, where τ value will be evaluated in experiments.

4.5 Experimental Set up

In this section, we discuss the evaluation criteria and equation used in evaluating the influence measurements. We further represent the datasets used in the experiments. Note that the influence measurements are implemented using Java.

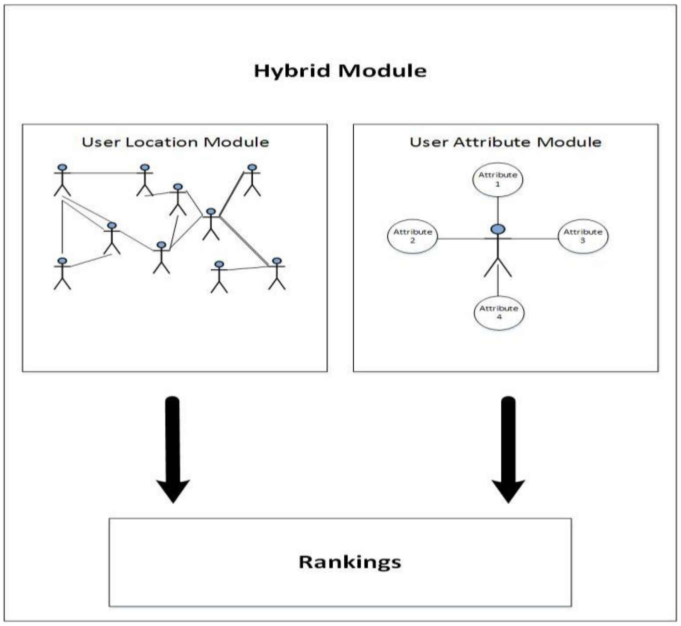


Figure 4.4. Users Structural location and attributes.

4.5.1 Evaluation

In order to measure the statistical significance of between rankings produced by the influence measurements and the ground truth, we use Pearson's Correlation Coefficient. Pearson's Correlation Coefficient measures how much two variables are related to each other by measuring the linear dependence between them. The possible outcomes from Pearson's Correlation is a value, i.e., r, between [−1, 1] representing the negative and positive relationship strength respectively. A strong correlation exists if $r \geq \pm 0.5$ where medium correlations exist when $\pm 0.5 \leq r$ and $r \geq \pm 0.3$; otherwise, there is a weak relationship or no relationship when $r = 0$ [79]. It is calculated as *follows*:

$$\mathrm{r}_{\mathrm{EM,PRD}} = \frac{\sum_{i=0}^{n}(\mathrm{EM}_i \mathrm{PRD}_i) - \mathrm{n}\,\overline{\mathrm{EM}}\,\overline{\mathrm{PRD}}}{\mathrm{n}\,\sigma_{\mathrm{EM}}\,\sigma_{\mathrm{PRD}}} \tag{0.4}$$

, where EM refers to the rankings of users ranked using the ground truth, and PRD is the ranking of users ranked using the influence measurements.

4.5.2 Dataset

We have used three datasets to assess the measurements discussed in this thesis: Flickr and Digg datasets to investigate the adaptability of influence measurements to different social networks. Flicker is a social network that is based on images. The dataset is retrieved from one of the groups in Flicker that includes users, images, interactions, and other metadata such as photo tags using the Flickr API. A total of 30.759 users have participated in the group. Some of them can be popular by posting images, while others only interact with other users. For example, 1.559 users have uploaded 4.991 images. There are 46.059 interactions between users representing *comments* and favorites. Digg is a social network that allows users to share news stories. Users can interact with each other by voting on stories. The dataset contains 139.409 users and 1.534.314 edges representing voting. Among these users, 474 users have shared 3.553 stories. This dataset is provided by [33]. StackOverflow is a social network for programmers; programmers can ask and answer questions related to programming. In this social network, users can rate the questions and answers. The interactions between users can be in the form of answers or ratings. The dataset contains 40.395 nodes and 246.492 edges, which represent the interaction between users in the form of answers. There are 26836 users who posted 263264 threads. This dataset is provided by [79] and can be downloaded from [3]. These datasets are different in nature. Digg is focused on news, Flicker is focused on images and photographers and StackOverflow is focused on programmers.

By conducting our analysis on these datasets, we are considering different behaviors on social networks. Table 4.1 shows the characteristics of datasets including

[3] https://www.ics.uci.edu/~duboisc/stackoverflow/

the number of nodes, which represents the number of users, the number of edges that represent interactions, and a number of posts

Table 4.1. Datasets characteristic.

Dataset	Number of nodes	Number of edges	Number of posts	Number of contributors
Flicker	30759	46213	4838 Images	1420
Digg	139409	1534314	3553 Stories	474
StackOverFlow	40395	246492	263264 threads	26836

4.6 Preliminary results

In the experiments, we employed the measurements discussed earlier to the Dataset to retrieves from Flickr and Digg. The results shown are grouped into three groups based on the correlation results for each dataset, i.e., Gr1, Gr2, and Gr3. These groups contain weak, medium and strong correlations respectively according to [79].

4.6.1 Structural-Based Influence Measurements

First, we employed the measurements based on the structural locations to both datasets and compared the rankings produced by these measurements with the ground truth. First, we employed the measurement to Flickr, which is discussed in the next paragraph.

As shown in Table 4.2 and Figure 4.5, on Flickr, the weighted in-degree centrality is the most correlated measurement with EM. The weighted incoming interactions represent the tie strength between users. This shows that tie strength is a good indicator of influence. Eigenvector and PageRank centrality come in the second and third places respectively, which shows that considering the importance of nodes is also an important indicator of influence. In-degree centrality is the fourth most correlated measurement with EM, *follow*ed by degree centrality. This shows that the number of connections a user has is a good indicator of influence. Note that the correlation results from these measurements are assigned to GR2. The closeness and Betweenness centrality techniques have weak correlations with EM and therefore assigned to Gr1.

When we employed the measurements to Digg, most the correlations rates have significantly increased. However, the order of the most correlated measurements has changed slightly. The weighted in-degree centrality is still the most correlated measurement with EM, with a correlation rate of 0.74. PageRank centrality jumped into the second place. Eigenvector centrality dropped one position, and in-degree jumped one position, making them come in the third place. Degree centrality has jumped to the third place as well. Note that all correlation rates produced by these measurements are now assigned to Gr3. The closeness and Betweenness centrality techniques are still the least correlated measurements; however, their correlation rates have increased to become medium correlations, and therefore are assigned to Gr2.

Then, we employed the measurements to StackOverflow, most the correlations rates have increased from the results produced on Flickr. However, the order of the most correlated measurements has changed slightly. The weighted in-degree centrality is still the most correlated measurement with EM, with a correlation rate of 0.61. Degree centrality comes in the second place with a correlation rate of 0.37, while in-degree comes in the third place with a correlation rate of 0.34. PageRank and Eigenvector centrality techniques come next with a correlation rata of 0.16. The closeness and

Betweenness centrality techniques are still the least correlated measurements with no correlation. The weighted in-degree is assigned to Gr3, degree and in-degree centrality techniques are assigned to Gr2, and the rest of the measurements are assigned to Gr1.

Table 4.2. Correlation between structural and attributes based influence measurements and an estimate of influence.

Structural-Based Influence Measurements	**Flickr Dataset**	**Digg Dataset**	**StackOverflow Dataset**
	Correlation Rate		
Closeness Centrality	0.28	0.18	-0.01
Betweenness Centrality	0.29	0.48	-0.09
Degree Centrality	0.30	0.66	0.37
In-degree Centrality	0.312	0.66	0.34
Weighted In-degree Centrality	0.42	0.74	0.61
Eigenvector Centrality	0.323	0.66	0.32
PageRank Centrality	0.316	0.70	0.32
Attribute Measurement	0.49	0.89	0.61

4.6.2 Attribute-Based Influence Measurement

We employed the attribute-based measurement, which represents how active users are, to the datasets. As shown in Table 4.2 and Figure 4.5, the attribute measurement achieves higher correlation than the structural-based measurements, which show that when a user is more active, the user can have more influence. However, on StackOverflow, the attribute measurement archives the same correlation rate as the weighted in-degree. The correlation rate in Flickr is medium, while it becomes strong on Digg and StackOverflow.

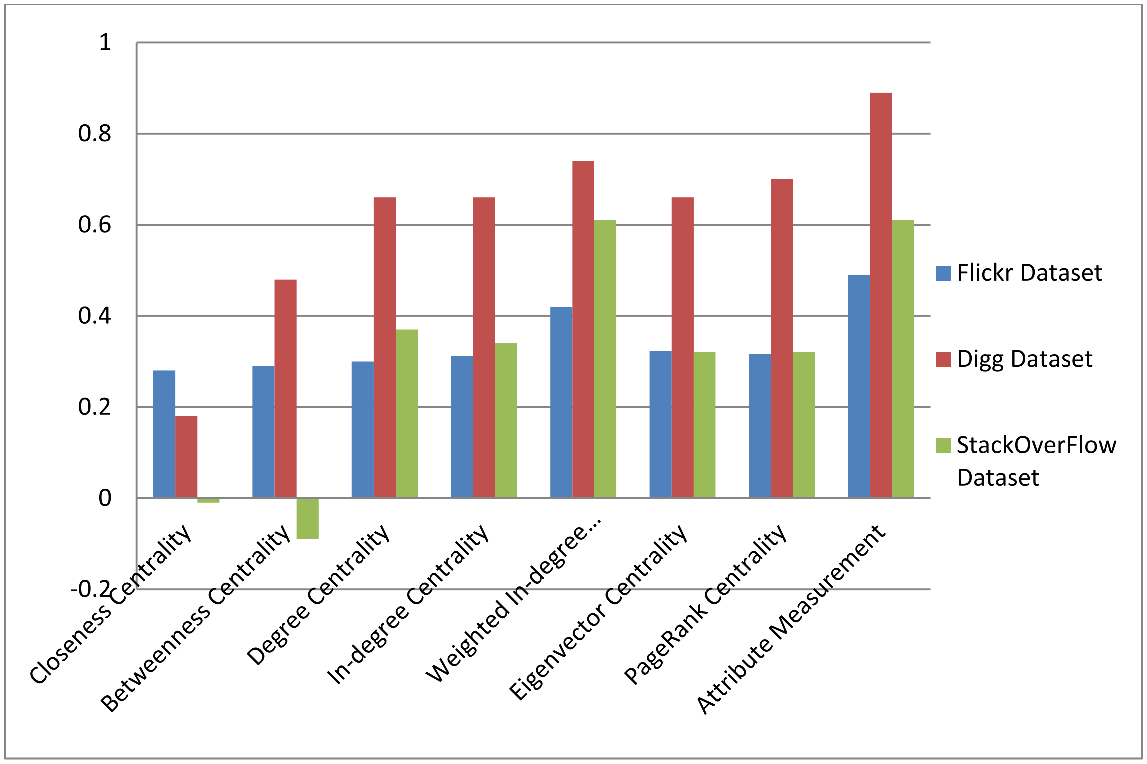


Figure 4.5. Correlation between the estimate of influence and structural and attributes influence measurements. The X-axis represents the correlation rates, where the Y-axis represents the influence measurements.

4.6.3 Hybrid Influence Measurements

We selected the top correlated structural-based measurements with EM, and integrate them with attribute-based measurement according to the equation presented before. We varied the τ value to observe the importance of the measurements over each other, and to find the optimal correlation rate.

First, as shown in Table 4.3 and Figure 4.6, we employed the four hybrid measurements to Flickr. For the correlation between the hybrid measurements and EM, a hybrid eigenvector is correlated with the ground with a correlation rate = 0.5. Hybrid PageRank has a correlation with EM with a rate equal to 0.503. Hybrid weighted in-degree has a correlation with EM with a rate of 0.502, while the correlation for hybrid in-degree and EM decreases to 0.496. Among these measurements, the hybrid PageRank is the most correlated measurement with EM. The measurements still have medium correlation rates, however, the correlation rates have increased over the previous measurements. Since the hybrid measurements act differently using different τ values, we apply different τ values ranging from 0.0005 to 0.9 to the hybrid measurements. We found that hybrid PageRank starts with 0.49 and then slightly increase to 0.5 and then slightly decrease to 0.49 when τ values are between 0.0005 and 0.45. Moreover, hybrid PageRank starts to decrease after that. This shows that hybrid PageRank performs better when the attribute-based influence measurements are more important than the structural-based influence measurement. For hybrid eigenvector, we found that the correlation stays within the same range for all the τ values which shows that attribute-based influence measurements are as important as structural-based influence measurements. Hybrid weighted in-degree starts with a high correlation with EM and then starts to decrease. This shows that the attribute-based influence measurement is more important than the structural-based influence measurements. Hybrid in-degree also starts with high correlation with EM and then dramatically decrease, which shows that attribute-based influence measurements are much more important than the structural-based influence measurements.

We further employed the measurements to Digg dataset. As shown in Table 4.3 and Figure 4.6, the correlation rates between the hybrid measurements and EM. The hybrid eigenvector is correlated with EM, with r = 0.903. Hybrid PageRank has a correlation with EM with a rate of 0.904. Hybrid weighted in-degree has a correlation with EM of 0.901 while the correlation for hybrid in-degree and EM decreases to 0.89.

Among these measurements, the hybrid PageRank is the most correlated measurement with the ground truth.

As for the importance of the attribute-based influence measurements over the structural-based influence measurements, we found that hybrid PageRank starts with 0.9 and then slightly increase to 0.904 and then slightly decrease to 0.8 when τ values are between 0.0005 and 0.05. However, hybrid PageRank starts to decrease after that. This shows that hybrid PageRank performs better when the attribute-based influence measurement is more important than the structural-based influence measurement. For hybrid Eigenvector, we found that the correlation stays within the same range for all the values which show that attribute-based influence measurement is as important as structural-based influence measurements. Hybrid weighted in-degree starts with a high correlation with EM and then starts to decrease. This shows that the attribute-based influence measurement is more important than the structural-based influence measurements. Hybrid in-degree also starts with high correlation with EM and then dramatically decrease, which shows that attribute-based influence measurements are much more important than the structural-based influence measurements.

We further employed the measurements to StackOverflow dataset. As shown in Table 4.3 and Figure 4.6, the correlation rates between the hybrid measurements and EM. The hybrid eigenvector is correlated with EM, with r of 0.61. Hybrid PageRank has a correlation with EM with a rate of 0.62. Hybrid weighted in-degree has a correlation with EM of 0.62 while the correlation for hybrid in-degree and EM decreases to 0.61. Among these measurements, the hybrid PageRank is the most correlated measurement with the ground truth.

As for the importance of the attribute-based influence measurements over the structural-based influence measurements, we found that hybrid PageRank starts with a correlation rate of 0.61 and then decrease to 0.49. This shows that hybrid PageRank performs better when the attribute-based influence measurement is more important than the structural-based influence measurement. For hybrid Eigenvector and weighted in-degree, we found that the correlation stays within the same range for all the values which show that attribute-based influence measurement is as important as structural-based influence measurements. Hybrid in-degree also starts with high correlation with EM and then decrease, which shows that attribute-based influence measurements are much more important than the structural-based influence measurements (Figures 4.7, 4.8 and 4.9 illiterates the behavior of the hybrid measurements using different τ values).

Table 4.3. Hybrid influence measurements comparison in terms of correlation.

Hybrid Influence Measurements	**Flickr Dataset**	**Digg Dataset**	**StackOverflow Dataset**
	Correlation Rate		
Hybrid Eigenvector	0.50	0.903	0.62
Hybrid PageRank	0.503	0.904	0.62
Hybrid In degree	0.496	0.89	0.62
Hybrid Weighted In degree	0.502	0.901	0.62

4.6.4 Comparison of related works measurements

We selected our proposed hybrid eigenvector to compare with the related works because it is the most stable hybrid measurements in terms of different τ values. This measurement is empirically compared with seven of the state of art influence measurements in Flickr and Digg datasets in terms of correlation. The results are shown in Table 4.4 and Figure 4.9.

For Flicker's social network, degree centrality is assigned to Gr1, where hybrid Eigenvector is assigned to Gr3. The rest of the measurements are assigned to Gr2. This can show that users' attributes and the number of *followers* are important factors for determining influential users since hybrid eigenvector is the only measurement that considers these characteristics. The only common thing between the measurements is that they all consider the degree of nodes, which shows that the number of *followers* each user has is an indicator of influence.

On Digg's dataset, all measurements are classified to Gr3. However, one measurement shows a very strong correlation where the other measurements have similar correlations. Therefore, we further divide Gr3 into Gr3.1 and Gr3.2 for very strong and strong correlations, respectively. The hybrid eigenvector is classified to Gr3.1 while other measurements are classified to Gr.3.2. Results for Digg's dataset have a similar trend to the results on Flicker. However, all correlation rates have significantly increased. Also, measurements that were classified to Gr.1 and Gr.2 have jumped to Gr.3.

On StackOverflow's dataset, only the hybrid measurement is assigned to Gr3. Complex and in-degree centrality techniques are assigned to Gr2, while the rest of the measurements are assigned to Gr1. This can show that users' attributes and the number of *followers* are important factors for determining influential users since hybrid eigenvector is the only measurement that considers these characteristics.

The hybrid eigenvector is the most correlated measurement with the EM in all the datasets. First as discussed earlier, we hypothesized that users' attributes are an important indicator of influence, which is one of the bases of the hybrid eigenvector. Complex centrality is the second most correlated measurement in both datasets. It is due to the fact that it considers the tie strength. LeaderRank is the third top correlated measurement with the empirical measurement in both datasets. Eigenvector centrality is the fourth most correlated measurements with EM in Flicker's dataset, while it is the fifth most correlated measurement in the Digg's dataset. Normalized Alpha Centrality is the fifth most correlated measurement in Flicker's dataset, but its correlation increases to the top fourth correlated measurement in Digg's dataset. PageRank is the sixth most correlated measurement in Flicker's dataset, while it is the second most correlated measurement in Digg's dataset. The previous four measurements have a similar correlation rate because they all consider the depth of the social network. However, they perform differently in both datasets. In-degree centrality is the seventh most correlated measurement in Flickr's dataset, while it is the second most correlated measurement in Digg's dataset. Degree centrality is the least correlated measurement in the Flicker's dataset, while it is the fourth top correlated measurement in Digg's dataset.

4.7 Result Discussion

The correlation rates of the measurements have significantly increased in Digg's dataset compared to the other datasets. We hypothesize that this is because of the social network characteristics and nature; for example, Flicker is a social network that supports images, while Digg acts like a new medium. This means that feasibility of influence measurements depends on their adaptability to social networks. We can conclude that hybrid-based influence measurements are better than single-based influence measurements, and users' attributes are more important than their structural locations in a network in term of correlation.

4.8 Performance Analysis

The measurements are grouped into iterative and noniterative. In-degree, Complex, and modified degree are non-iterative techniques, where the rest of the measurements are iterative algorithms. As shown in Table 4.5, non-iterative measurements have a linear runtime complexity of O(m) since they only need to compute the degree of each node once it is given m edges. They have a space complexity of O(m+n) because they store the ranking of each node n based on its adjacent edges m. Iterative measurements have the same exponential runtime and space complexity of O(m) and O(m+n) per iteration, respectively, since they need to compute and store the ranking list for each iteration. All of the measurements have I/O cost of O(m+ n). To obtain a sense of their complexities in the implementation, we have computed the runtime of each measurement in the two datasets. The results confirmed the performance analysis in terms of O notation. In Flicker Dataset, modified degree and in-degree take 10 ms to complete using wall clock time, which makes them the fastest measurements. Complex centrality took slightly longer runtime because it computes the exponent for each node. These influence measurements are well suited for large-scale social networks. For eigenvector, PageRank, and LederRank, we limited the number of iteration to 55 since they are already proven to converge. PageRank takes 100 ms, where eigenvector is executed in 170 ms. Hybrid measurement takes slightly more time than eigenvector since they are both based on computing the eigenvector. LeaderRank completes in 345 ms. Normalized Alpha centrality takes 26,876 ms since it iterates much more than the previous measurements. There was not a lot of variation in the performance of each measurement in the Digg dataset. Eigenvector centrality was the fastest iterative measurement where LeaderRank was the second fastest iterative measurement. Normalized Alpha Centrality again took the longest running time of 40,056 ms. These results reflect the complexities of influence measurements. Table 4.5 shows the summary of the performance evaluation for the selected influence measurements in term of complexity. Table 4.5 shows the runtime complexity that represents the number of steps to run the algorithm, and the space complexity shows the computational resources needed by the algorithms. I/O is the cost of input and output management. The runtime in both datasets represents the actual time spent by each algorithm.

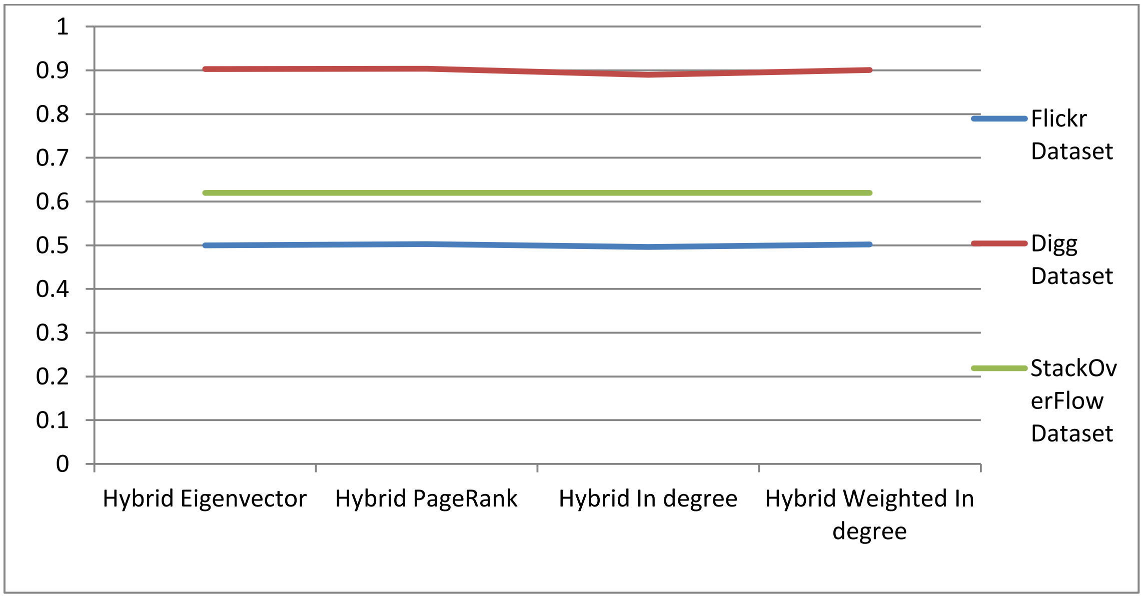


Figure 4.6. Correlation between the estimate of influence and hybrid influence measurements. The X-axis represents the correlation rates, where the Y-axis represents the influence measurements.

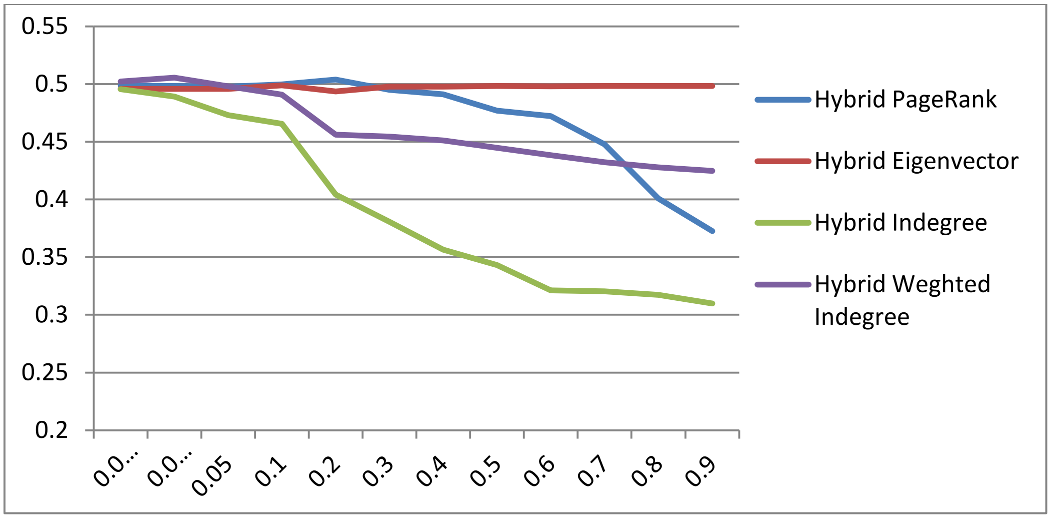


Figure 4.7. Hybrid influence measurements performance over different t values in terms of correlation on Flickr dataset. The X-axis represents the correlation rate where the Y-axis represents the t values.

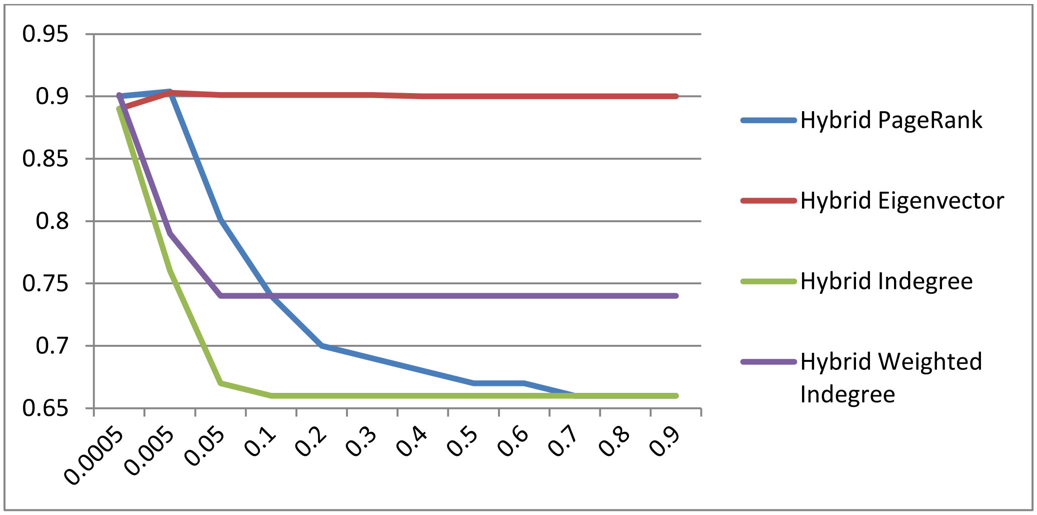


Figure 4.8. Hybrid influence measurements performance over different t values in terms of correlation on Digg dataset. The X-axis represents the correlation rate where the Y-axis represents the t values.

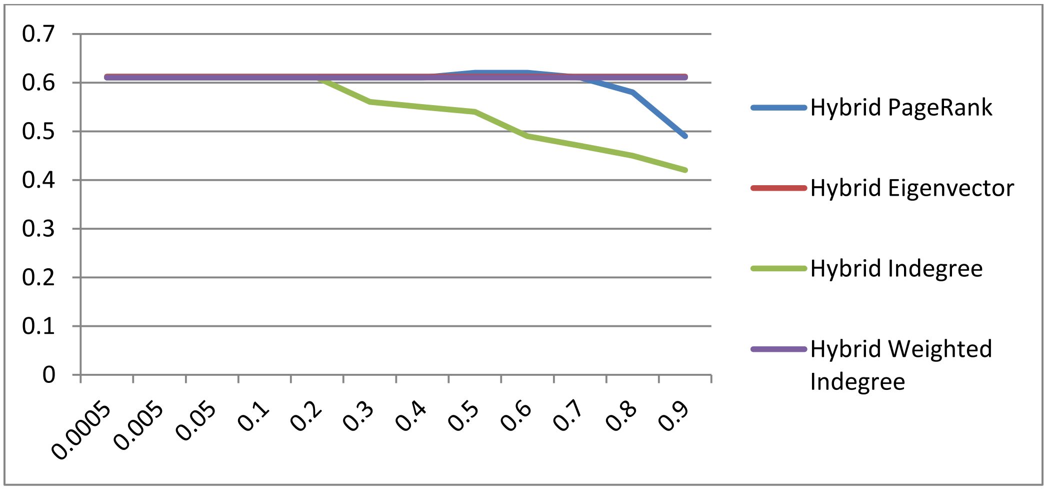


Figure 4.9. Hybrid influence measurements performance over different t values in terms of correlation on StackOverflow dataset. The X-axis represents the correlation rate where the Y-axis represents the t values.

Table 40.4. Comparison between the proposed measurement and the existing measurements in terms of correlation.

Influence Measurements	**Flickr Dataset**	**Digg Dataset**	**StackOverflow Dataset**
	Correlation Rate		
Hybrid Measurement	0.50	0.90	0.62
Complex Centrality [9]	0.38	0.70	0.36
LeaderRank [17]	0.327	0.69	0.14
Eigenvector Centrality [12]	0.323	0.66	0.32
PageRank Centrality [11]	0.316	0.70	0.32
Normalized Alpha Centrality [16]	0.42	0.74	0.144
In-degree Centrality [10]	0.312	0.66	0.34
Modified Degree Centrality [9]	0.27	0.67	0.37

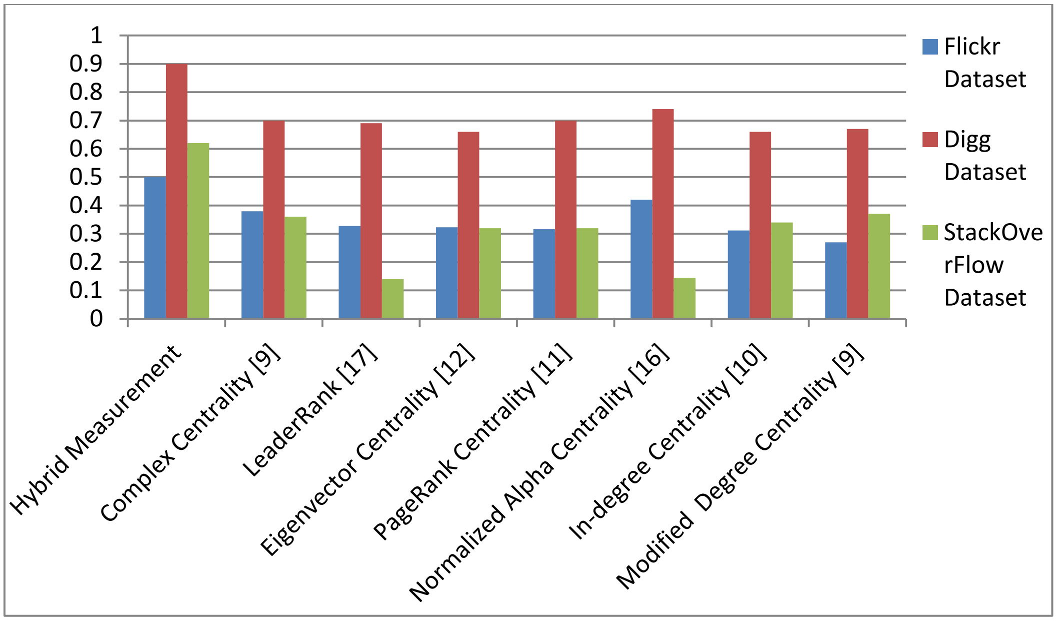


Figure 4.10. Comparison between the proposed measurement and the existing measurements in terms of correlation. The X-axis represents the correlation rates, where the Y-axis represents the influence. measurements.

Table 4.5. Performance of measurements, n = nodes, m = edge.

Influence Measurements	**Runtime complexity**	**Space complexity**	**I/O cost**	**Runtime in milliseconds (Flicker dataset)**	**Runtime in milliseconds (Digg dataset)**
Hybrid Measurement	O(m) per iteration	O(m+ n) per iteration	O(m +n)	178 (iterations = 55)	3396 (iterations = 55)
Complex Centrality [9]	O(m)	O(m+n)	O(m+n)	30	225
LeaderRank [17]	O(m) per iteration	O(m+ n) per iteration	O(m +n)	345 (iterations = 55	5435 (iterations = 55)
Eigenvector Centrality [12]	O(m) per iteration	O(m+ n) per iteration	O(m +n)	170 (iterations = 55)	3349 (iterations = 55)
PageRank Centrality [11]	O(m) per iteration	O(m+ n) per iteration	O(m +n)	100 (iterations = 55)	6045 (iterations =
Normalized Alpha Centrality [16]	O(m) per iteration	O(m+ n) per iteration	O(m +n)	26,876 (Alpha <0:1)	40,056 (Alpha< 0.05)
In-degree Centrality [10]	O(m)	O(m+n)	O(m+n)	10	40
Modified Degree Centrality [9]	O(m)	O(m+n)	O(m+n)	10	45

CHAPTER 5: PREDICTING THE POPULARITY OF IMAGES USING MACHINE LEARNING ALGORITHMS

In this chapter, we discuss the approaches used to predict popular images on social networks. We further explain the experiments and show the results.

5.1 Overview

As mentioned in Chapter 3, we will predict the popularity images along the timeline. Images have become important media for communication between users on social networks. As a result, a significant number of papers have investigated several topics related to images, including predicting image popularity [35]-[40], [43]. The previous works have not considered the prediction of image popularity along the timeline. However, image popularity can be decreased or increased over time. We investigate the information about an image within an hour of upload to predict its future popularity (after a day, after a week, or after a month) and stability of stability of popularity. We employ social, content and early popularity features to predict an image's popularity over time.

5.2 Popularity Measurement

Webster's dictionary defines popularity as "the state of being liked, enjoyed, accepted, or done by a large number of people [80]." The reality of this definition can be found on social networks, through users' interactions. On Instagram, users can show their admiration for the image by liking it. Intuitively, an image that receives many *likes* can be considered popular. Therefore, we adopt the number of *likes* as our popularity measurement. We classify the number of *likes* to as either low or high, where low means unpopular and high means popular.

However, popularity is subjective. In order to determine popular and unpopular images, we adopt the Pareto principle (80%−20%) to compute the threshold using the number of *likes* as used in [35]. The Pareto Principle is the observation that 80% of effects are caused by 20% of the causes. An image that receives a number of *likes* that is greater or equal to this threshold is considered popular. In our dataset, we observe that 20% of the images receive 99% of the total number of *likes*. The thresholds of popularity criteria are: *likes* greater or equal than 49 for the first hour, 69 for the next day, 75 for the next week, and 76 for the next month. For example, if image *i* receives 49 *likes* during the first hour it is shared, it will be considered popular. If the number of *likes* increases to 70 within the first day, it will stay popular for the first day. However, if the number of *likes* during the week is less than 75, *i* will become unpopular after the first week. If the number of *likes* stays less than the threshold after one month, it will stay unpopular. Image *i* started out popular in the first hour, then it kept its popularity for the first day. However, it became unpopular after one week.

Figure 5.1 shows the distribution of the number of images with respect to the number of *likes* (in the first hour, next day, first week, and first month). Both axes are log scaled. The x-axis is the normalized number of *likes* by the popularity threshold for each time period; therefore, the overlapped line represents the normalized popularity thresholds. Y-axis is the normalized number of images by the maximum number of images.

In order to represent image content, we introduce keyword vector. It is a novel approach, built upon well-established techniques in the field of NLP and clustering. It represents the semantics of images using their captions. Keyword vector represents multiple meanings from multiple keywords

As mentioned earlier, we observed that not all images posted by the same user become popular. This observation motivated us to investigate the relationship between the content and popularity

5.3 Approach

We investigate the relationship between social context, an image's semantics and image's early popularity, and their popularity. These features represent the early information that is retrieved in the first hour of image upload to Instagram.

5.3.1 Social Context

We select the social context (which represents the information of the users who upload images) because several works showed that the popularity of the user who uploads an image is correlated with the image's popularity [35], [37]-[40], [43]. To represent user popularity, we chose the number of *followers* because it is an indicator of a user's popularity. We normalize the number of *followers* by the maximum number of *followers* because (from Figure 5.1), the normalized distributions are similar to each other and the change ratio is more important. This is computed using the *follow*ing equation:

$$\mathrm{Si} = \frac{\mathrm{Log10(foli+1)}}{\mathrm{Log10(Max(\#Fol))}} \qquad (5.1)$$

, where $s_f \in \{0,1\}$ and fol is the number of *followers*.

5.3.2 Images semantics

Oglesbee states that ”Looking at a picture without a caption is *like* watching television with the sound turned off” [81]. Understanding the meaning of images can be challenging; photographers use captions to describe images to help viewers understand the photographers’ point of view. Captions are a small description of images that are usually placed under the images (See Figure 5.2). As shown in Figure 5.2, the first image on the left is that of a house in the countryside. The caption reads as ”*rustic residence*”, which explains the meaning from the image. Thus, multimedia social networks (such as Flicker and Instagram) support captions. We are trying to find images that have a similar meaning. And since semantics are subjective, the text gives better results than computer vision for extracting semantics.

In this paper, we analyze the effect of an image’s semantics on its popularity. Using captions, we extract the semantics of an image using word2ve and clustering techniques by introducing the keyword vector. The keyword vector represents the semantics of an image in a numerical form. Word2vec aims to map words that have a similar meaning to nearby points using a continuous vector space. For example, dog and cat would be mapped to nearby points because they are both pets. Word2vec uses a neural network to learn distributed representations of words. We understand that not all photographers provide captions, but still many do to explain the images. The process of extracting the image’s semantics has four steps:

1. **Step 1**: **Captions Preprocessing** In this step, we remove stop words and special characters from captions and tokenized the remaining words [78]. The remaining words are referred to as keywords. For example, the two images shown in Figure 5.2 are shared on Instagram and annotated with two captions: "*rustic residence*" and "*the girl and the goat*". If the captions have any special character (such as *&*) or stop words (such as *and*), they will be removed. We then tokenized the remaining keywords. Therefore, for the two captions: we will end up with the *follow*ing keywords: ”*rustic*” and ”*residence*” as well as ”*girl*” and ”*goat*”. The result from this step is a caption words vector W_j , for each image containing the keywords extracted from image *i* caption. $W_j = \{w_0, ..w_n\}$, where w_j represents a single keyword.
2. **Step 2: Vector representations of keywords** To understand the meaning behind keywords, we employ Word2Vec [52]. Word2Vec is an unsupervised two-layer neural network that generates distributed numerical vectors to represent words [52]. It groups similar words based on their vectors in the vector space to detect similarity. For example, with a pre-trained model using a Skip-gram model on 100 billion words, the vector of *goat* is [0.05, 0.04...]. When computing the most similar words to *goat*, we get *goats, sheep, pig, llama*, and *cow* with very high cosine similarities. The output from this step is a 300-dimensional vector, WV_l for each keyword, W_j, $WV_i = \{x_0, ..x_{300}\}$.
3. **Step 3: Semantic word clustering:** Different words can be similar based on their semantics. For example, an *oven* and *refrigerator* can both indicate *kitchen, food,* or *electronics*. Therefore, words that have similar meaning should be clustered. To do so, *k*-means is employed. We cluster unique WV of all images into k number of

clusters using *k*-means. The result from this step is a dictionary that includes each keyword with its corresponding cluster, DW={[W_0, WC_k],….,[$W_{n,}$$WC_k$]}.

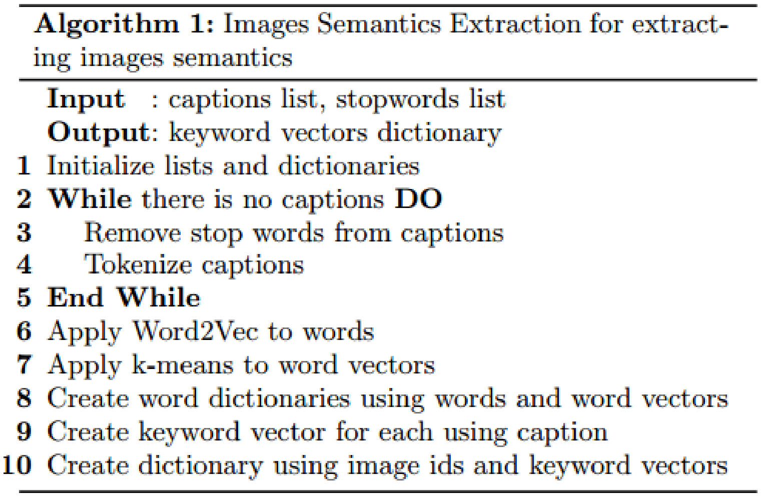
4. **Step4: Keyword Vector Generation** Using captions, photographers can use different keywords to describe multiple objects in an image. Therefore, by a caption, one image may have multiple meanings from multiple keywords. Keyword vector is introduced to represent the multiple semantics of an image. The keyword vector is a binary vector representing the semantics of keywords with a length of k. We check WV for each image to see which clusters the W fall into. The clusters corresponding to the WVs are represented as 1, otherwise as 0. With this bit info, we compose KWV for an image. Each image has one KWV. For example, using our model, we cluster the keywords from Figure 5.2. We end up with the *follow*ing keyword clusters {*girl : 1, goat : 2, resident : 1, rustic : 3*}, where k=3. We see that *girl* and *resident* are clustered together while *goat* and *rustic* are placed in clusters 2 and 3 respectively. The two keyword vectors for images 1 and 2 are KWV_1 = [0, 1, 1] and KWV_2 = [1, 1, 0] which are computed using the *follow*ing equations.

$$KWV_v = \{c_0, ..., c_m\} \tag{5.2}$$

$$c_y = f(x) = \begin{cases} 1, & if\ cw_j \in cc_x \\ 0, & if\ cw_j \notin cc_x \end{cases}, \tag{5.3}$$

where $c_j \in [0,1] \forall images$. c_y is a binary value representing whether the keywords belong to cluster cc_x or not.

The keyword vector can explain the semantics of images by considering multiple keywords of different meanings due to the different semantic clusters of keywords. It also combines similar keywords since keywords that are semantically similar will be considered as one meaning because they belong to the same semantic cluster. As shown in Algorithm 1, there are 10 simple steps to compute the keyword vectors for images. This algorithm accepts two lists and generates a dictionary containing the keyword vector for each image. We initialize lists and dictionaries used in the algorithm. We then remove the stop words from captions, tokenized each caption, and generate CWV for each image, which is implemented using NLTK [78]. Next, we apply Word2Vec to the words to compute the vectors for each word (VW) [82]. After that, we apply *k*-means to cluster the words based on the word vectors (KWV) implemented using Sklearn [83]. Then we create a dictionary consisting of words and clusters. We create the keyword vector for each image, then finally create the dictionary for images and keyword vectors.



Algorithm 1: Images Semantics Extraction for extracting images semantics

```
   Input  : captions list, stopwords list
   Output: keyword vectors dictionary
 1 Initialize lists and dictionaries
 2 While there is no captions DO
 3     Remove stop words from captions
 4     Tokenize captions
 5 End While
 6 Apply Word2Vec to words
 7 Apply k-means to word vectors
 8 Create word dictionaries using words and word vectors
 9 Create keyword vector for each using caption
10 Create dictionary using image ids and keyword vectors
```

5.3.3 Early Popularity

There has not been any research that considers the early popularity in predicting images future popularity. As shown in Figure 5.1, the distribution of images and *likes* over different time frames show similar trends relatively; this can mean that there is a possibility that early information about popularity can be used to predict future popularity. However, the popularity of an image is not necessarily constant over time; it can increase or decrease along the timeline. Based on our empirical analysis on our dataset, about 10% of the images have changing popularity over time.

We investigate the early popularity of an image to predict its future popularity as well as the stability of images popularity. On social networks, we collect popularity data, i.e., number of *likes* during the first hour the image is uploaded. We then employ the popularity threshold to classify the early popularity. This feature is a binary variable that represents popular images as 1 and 0 for unpopular. The popularity variable (*EP*) is computed based on the Pareto principle threshold as *follows*:

$$EP_i = \begin{cases} 1\ if\ likes \geq popularity\ threshold \\ \qquad\qquad else\ 0 \end{cases} \tag{5.4}$$

5.3.3 Prediction

For predicting the popularity of images, we employed several classifiers including SVM, Naïve Bays, Decision tree, and Random Forest. Gaussian Naïve Bays outperformed the other classifiers in terms of accuracy. Therefore, we only include the results generated by Gaussian Naïve Bays. Gaussian Naïve Bayes computes the probability of each class, instead of computing the distance as explained in Chapter 5.

To employ Word2Vec, we use Gensim, a python library that implements Word2vec [82]. The pre-trained model that is used for the experiments is provided by

[82]. This model is trained on 100 billion words from Google News and achieved an accuracy rate of 73%. It can be downloaded from [4].

For *k*-means, we varied *k* between 4 and 1000 and observed that when *k* > 250, we get better results. This is due to small clusters of words give better results than larger words clusters. For *k*-means, we varied *k* between 4 and 1000 and observed that when *k* > 250, we get better results. This is due to small clusters of words give better results than larger words clusters. To implement the algorithms, we used the implementation from Scikit-learn [83].

5.4 Experimental Set up

In this section, we discuss the evaluation criteria used in evaluating the predictions models. We further represent the datasets used in the experiments.

5.4.1 Evaluation

For evaluating the accuracy of our models, we adopt sensitivity (i.e., true positive rate). In the medical field, sensitivity is used to report the proportion of people with the disease who are correctly identified as sick [84]. It is widely used in the medical field because the percentage of people with the disease is much smaller than the percentage of healthy people [84]. We are interested in predicting the popular images which is much less than unpopular images. Therefore, sensitivity reflects the ability of our model to predict popular images. The sensitivity is computed as *follow*:

[4] https://code.google.com/p/word2vec/.

$$Sensitivity = \frac{Number\ of\ True\ Positives}{\text{Number of P opular Images}}, \qquad (05.5)$$

where true positives are popular images that are correctly identified as popular.

5.4.1 Datasets

5.4.1.1 Instagram Dataset

We crawled the data using the Instagram API[5]. There are two methods to retrieve images from Instagram. The first method is to retrieve the recent images using given users IDs; the second method is to retrieve images based on a given geographical location. We choose the first method because we wanted our data to be completely random. This approach requires users' IDs. Based on our experiment with the Instagram API, we noticed that Instagram users' IDs are simply numbered from one to millions. Therefore, we randomly selected more than 1, 000, 000 IDs. Using these IDs, we triggered the Instagram API to check whether these users are private or public (since users have the choice to make their profiles publicly available to users or not). We found 149, 520 users who are public. However, among these users, there are 89, 093 who shared at least ten images.

We use these users when we were collecting the data because they are active. We were able to retrieve two random datasets containing their images that were uploaded during the first hour. For our experiments, we retrieved 69, 000 images. However, after preprocessing, we have 51, 647 images. Set 1 contains 39, 302 images, while Set 2 has 12, 345 images. Set 1 was retrieved between January 2016 and February 2016, while set

[5] https://api.instagram.com/

2 was retrieved between March 2016 and April 2016. Set 1 is used for training while testing is performed on Set 2. These images have received 16, 331, 397 *likes*. Instagram does not provide the timestamps of the *likes*; therefore, we tracked the number of *likes* for each image using its ID for the three time periods stated earlier.

5.4.1.2 Flickr Dataset

We used Flickr data to analyze the effects of our proposed semantic feature on an image's popularity. McParlane et.al. used this data to predict the popularity of images [35]. For the popularity measurements, they adopted the number of *views* and number of *comments*. They provided the dataset, popularity measurements, and popularity thresholds. The dataset contains a total of 867, 312 images. The distribution of the dataset with regard to the popularity of measurements is shown in Figure 5.3.

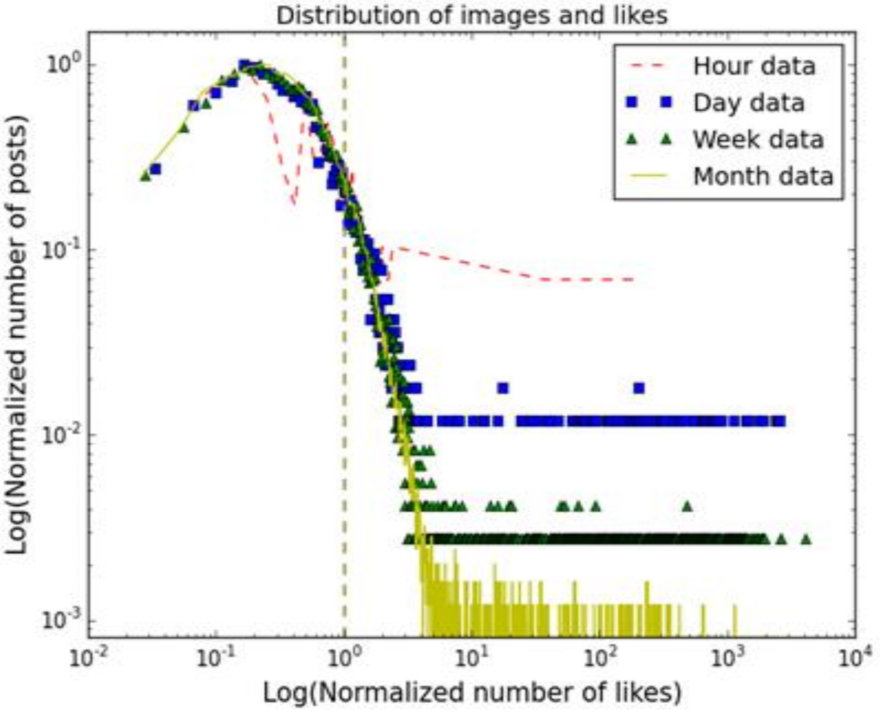
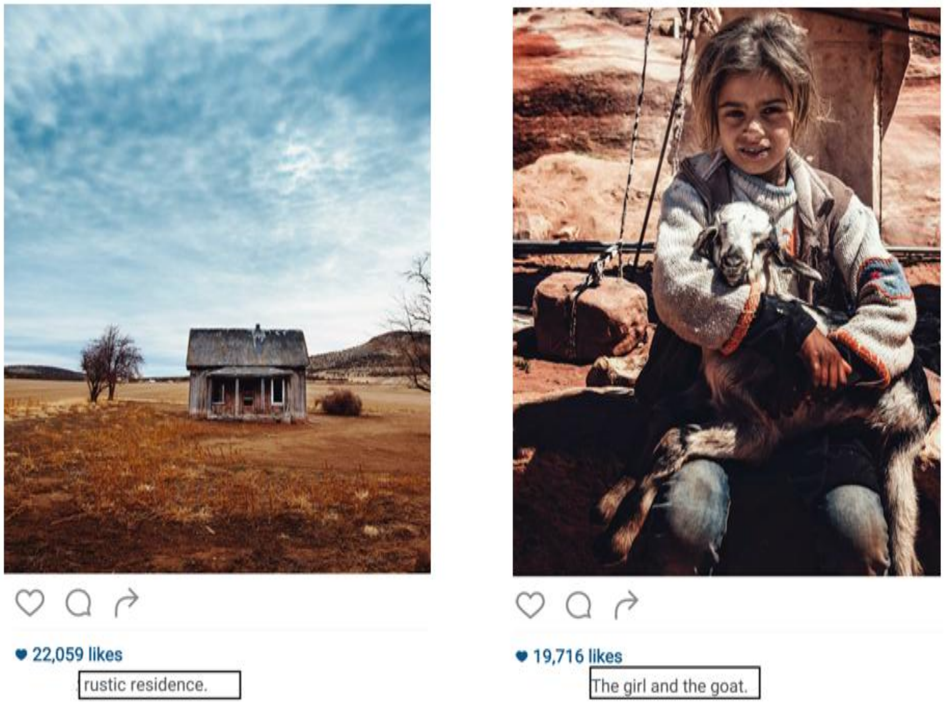


Figure 5.1. The graph represents the distribution of images and likes for the first hour, next-day, first week, and first month for the Instagram dataset. The line represents the overlapped normalized popularity thresholds. The x-axis is the normalized number of likes while the y-axis is the normalized number of images.



22,059 likes
rustic residence.

19,716 likes
The girl and the goat.

Figure 50.2. Two images retrieved from Instagram with their captions: the image on the left is described as ”rustic house”, while the image on the right is described as ”the girl and the goat”.

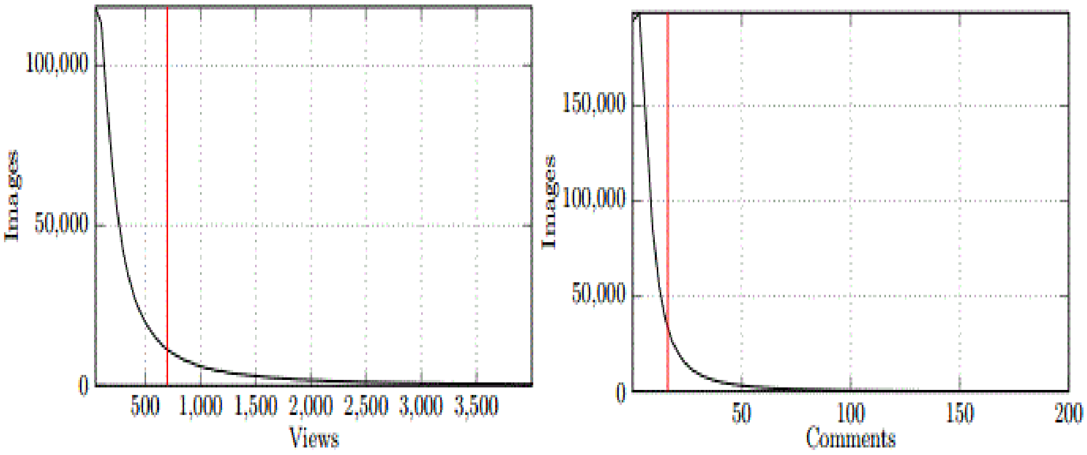


Figure 50.3. The graph shows the number of Views (left) and a number of comments (right), and the number of images distribution for the Flickr dataset used by McParlane et al., [31]. The red line is the popularity threshold based on the Pareto principle.

Figure 5.4 shows another example of how we create the keyword vector for an image extracted from Instagram.

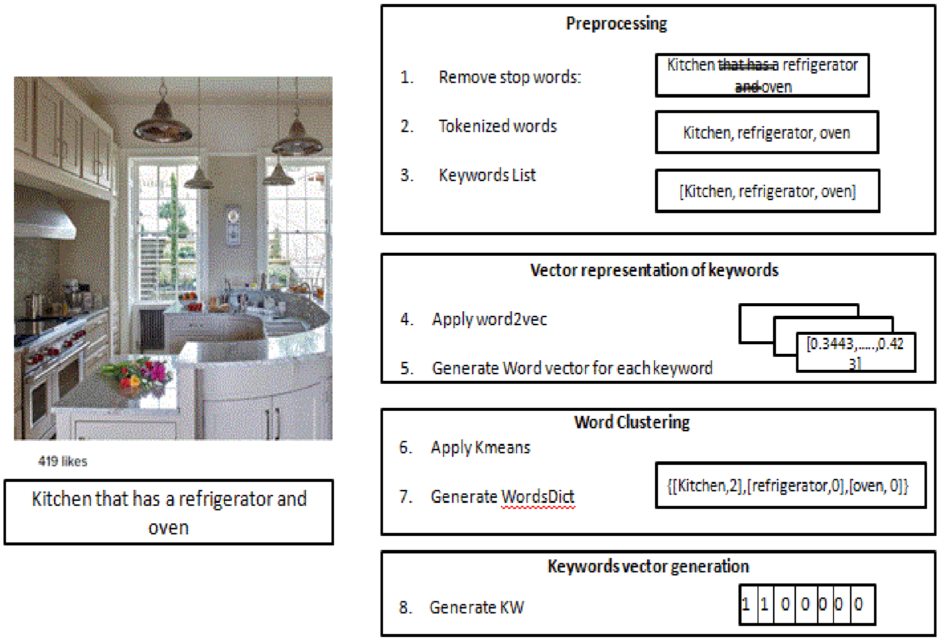

419 likes
Kitchen that has a refrigerator and oven
Preprocessing
1. Remove stop words:
Kitchen ~~that has a~~ refrigerator ~~and~~ oven
2. Tokenized words
Kitchen, refrigerator, oven
3. Keywords List
[Kitchen, refrigerator, oven]
Vector representation of keywords
4. Apply word2vec
5. Generate Word vector for each keyword
[0.3443,.....,0.42 3]
Word Clustering
6. Apply Kmeans
7. Generate WordsDict
{[Kitchen,2],[refrigerator,0],[oven, 0]}
Keywords vector generation
8. Generate KW
1 1 0 0 0 0 0

Figure 50.4. Another example of representing the semantics of images using keyword vector.

5.6 Preliminary Results

In this section, we present our preliminary results for predicting the future popularity of images as well as the stability of images popularity.

5.6.1 Predicting the future popularity of images along the timeline on Instagram:

User information can predict popular images with a sensitivity rate of 0.55, as shown in Table 5.1. This indicates that images posted by powerful users can become popular based on the users' popularity. However, not all images posted by the same users become popular. For predicting the popularity after one week and one month, the sensitivity rates decreased slightly. This shows that as time passes, the correlation between the popularity of users and popularity of their images decrease, which means that an image posted by a popular user can become popular in a short period of time, however, as time passes, it may not be able to keep its popularity. Table 7 shows that image semantics is the least important feature for popularity prediction, with a sensitivity rate around 0.38. This shows that there is a weak correlation between images semantics and their popularity. For popularity prediction over time, there is a slight improvement in the popularity prediction accuracy from 0.38 to 0.40. This improvement means that images semantics have more effect on popularity as time passes, which is the opposite from the social context. This shows that in long-term popularity, an image's semantics can be more important than the user's information. Images early popularity is the most crucial feature in popularity prediction with a sensitivity rate of 0.90 as shown in Table 5.1.

We observe that early popularity is linked closely to future popularity. This is because popularity may become saturated after the first hour. However, there are no improvements on prediction during different periods. This suggests that more features are needed to determine the changes in popularity over time. The results are shown in Figure 5.5.

5.6.2 Predicting the stability of images popularity on Instagram:

Based on the analysis of images popularity on Instagram, we see that images who usually start popular stay popular, and images that start unpopular stay unpopular. However, this is not the case all the time. We observed that the popularity of several images had changed over time. In this experiment, we investigate what may drive such a behavior. We employ the three features discussed earlier to predict the stability of popularity. Our results show that early popularity and user information cannot predict the changes of popularity, even so, they are observed to be important in predicting image future popularity, where image semantics can predict the popularity changes with a sensitivity rate of 0.34, which shows that the semantics of an image has an effect on it popularity as time passes.

On the other hand, we investigated what makes the popularity of an image stay stable over time. We found that social context and early popularity are perfectly linked with stable popularity with a perfect sensitivity rate of 1.0. For the early popularity result, we hypothesize that this can be because popularity may be saturated after the first hour. Moreover, the social context result indicates that a user's popularity can make the popularity of his/her images stable over time. An image's semantics is also highly correlated with stable popularity with a sensitivity rate of 0.76, which also shows that the content of an image can make the image keep its popularity for a long time. The results are shown in Figure 5.6.

5.6.3 Comparison between our semantic feature and the related work:

We use the MIR-Flickr 1M Collection to compare our work with [35]. McParlane et al., provided the experimental settings, including the testing and training data [35]. They also used the Pareto Principle to compute the popularity threshold adopted in this work. For fairness of comparison, we used the same accuracy matrix (the proportion of the total number of predictions that were correct), testing data, and popularity measurements (*views* and *comments*). They reported that there are only 1,000 test samples provided in their dataset but we found 1, 657 samples. Therefore, we choose 1,000 samples randomly for testing. They employed a combination of social, context and visual features. They achieved accuracy rates of 0.76 and 0.59 for *comments* and *views*, respectively. As shown in Table 5.2, our semantic feature outperforms the related work and increased the accuracy rates to 0.78 and 0.69 for *comments* and *views* respectively.

Table 50.1. The accuracy of future popularity over different time periods.

Future Popularity	Accuracy	Image's semantics	Early Popularity
Day	0.57	0.38	0.90
Week	0.56	0.39	0.90
Month	0.55	0.40	0.90

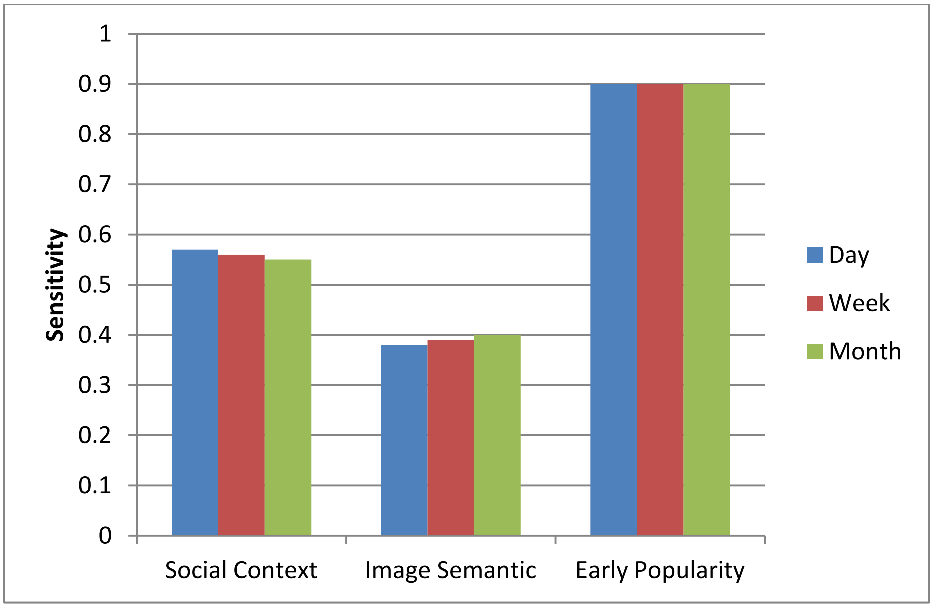


Figure 05.5. The accuracy of future popularity over different time periods.

Table 5.2. The accuracy of stability of popularity prediction using social context, content and early popularity.

Stability of popularity	**Accuracy**		
	Social Context	**Content**	**Early popularity**
Constant	1	0.76	1
Changing	0	0.34	0

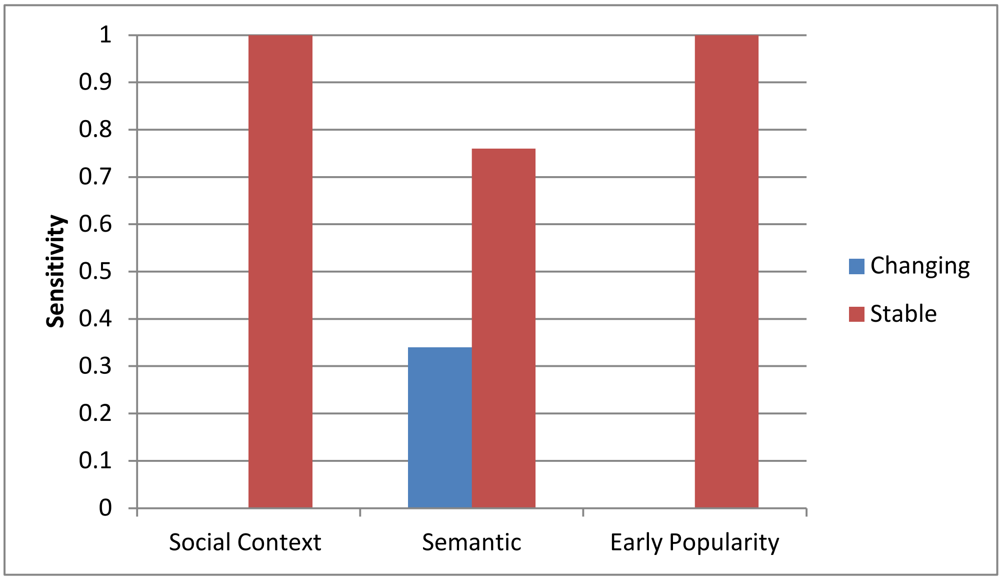


Figure 05.6. The accuracy of stability of popularity prediction using social context, content and early popularity.

Table 50.3. Comparison of our proposed semantic feature to the features used by McParlane et al., [35].

Approach	**Accuracy**	
	Views	**Comments**
Our approach	0.69	0.78
McParlane et al. [8]	0.59	0.76

CHAPTER 6: PREDICTING THE POPULARITY OF IMAGES USING TOPOLOGICAL DATA ANALYSIS

In this chapter, a new approach for analyzing social network data is presented. This approach is based on topological data analysis.

6.1 Overview

These days, finding meaningful data from social networks can be challenging because social network data can be high dimensional and noisy [85]-[87]. Therefore, extracting meaningful information from such data has become more critical. Topological data analysis as an alternative approach for mining social network data. Topological data analysis is an approach based on applied mathematics that analyzes data using a set of techniques from topology [57], [88]. It analyzes high dimensional data by analyzing the geometric shape of the data and has been shown to be robust to noise [57], [89]-[91], which will be further discussed in the upcoming sections. Topological data analysis has been adopted in many areas of study, such as biology [59], [89], [91], image processing [66], and financial analysis [92], [93].

A topological data analysis approach is used to address the problem of image popularity on social networks, specifically on Instagram, to investigate the feasibility of topological data analysis for social network analysis and mining since topological data analysis has not been previously investigated for social network analysis and mining to address the issues arising from the nature of social network data. The same popularity measurement proposed in Chapter 5 is used.

6.2 Features

Social context, as well as image semantics, are employed to analyze the effect of these features on an image's popularity.

6.2.1 Image Content

In order to extract semantics from images, we use images' captions as used in Chapter 5, however, the approach is different. To extract semantics from captions, Word2vec is used. As mentioned earlier, Word2vec [53] aims to map words that have a similar meaning to nearby points using a continuous vector space. When enough data, usage, and contexts are provided, Word2vec can guess a word's meaning based on past appearances using the neural network, which is used to learn distributed representations of words; it represents each word in the vector-space using a 300-dimensional vector [53]. These vectors can be used to establish a word's association with other words in terms of the similarity between the words' meanings. For example, apple is to fruit is like orange is to fruit.

In our approach, we first tokenized the image's caption. Then, we remove stopwords and special characters, such as with. Since one caption from each image can have a number of words and each word has its own contribution to the image, all words-vectors from a caption are averaged to make one representative caption vector considering all the contributions of the words for one image. After this, each image has one caption vector with 300 dimensions, CC, which is computed as *follows*:

$$CC = \frac{1}{N}\sum_{i=1}^{N}(v_0, \ldots\ldots, v_{300}), \qquad (6.1)$$

where n represents the number of words, and V represents the 300-dimensional vector for each word.

For example, let us have an image with a caption of “kitchen with refrigerator and oven”. First, we tokenized the words from the caption, we will have five words: [kitchen, with, refrigerator, and, oven]. Then, the stop words are removed. Therefore, [with, and] are removed. The three remaining words will be converted to numerical forms using Word2vec. Each word is represented by a 300-dimensional vector, called v. Finally, we compute the average of the three vectors to represent the image content, CC=13($v_{kitchen}$+$v_{refrigerator}$+v_{oven}). This example is illustrated in Figure 6.1.

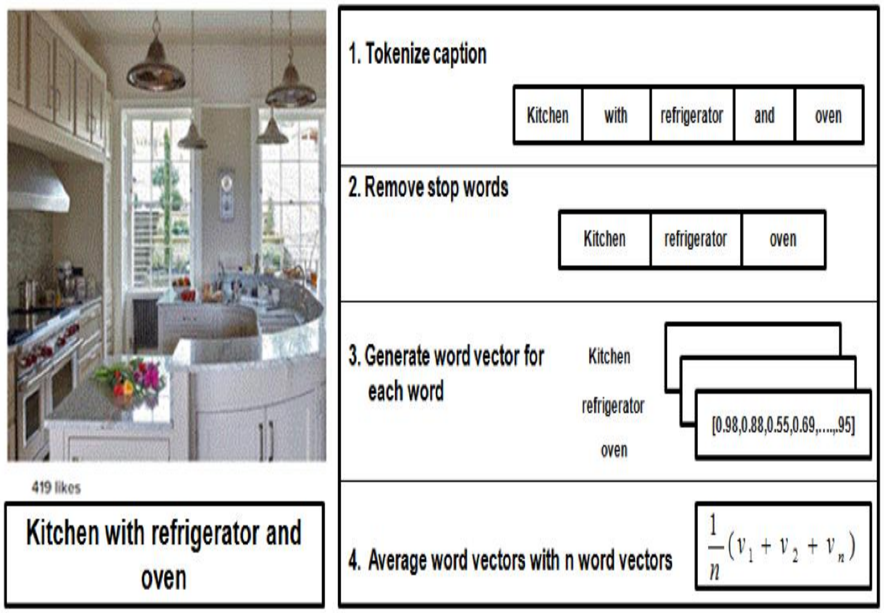


Figure 6.1. An example of how the image content is represented.

6.2.2 Social Context

As proposed in Chapter 5, the normalized number of *followers* is selected. The number of *followers* is normalized because we want to focus more on the order of magnitude of the *followers*, which shows that the ratios of the number of *followers* are more important than the exact number of *followers*.

6.2 Approach

Topological data analysis can be generalized to solve various problems. As mentioned earlier, the input to mapper is a distance matrix, while the output is a set of clusters.

A distance matrix is a square matrix that represents the distances between the elements in a set [94]. Since there are many problems that can be solved using clustering algorithms, topological data analysis can be adapted to solve many research problems on social network analysis and mining. Moreover, any distance metric can be used, such as Euclidean or cosine distances.

For the content feature, we compute the distances between any two images *i* and *j* using the cosine distance [95], called CD, of their 300-dimensional caption vectors, i.e., CC, which is calculated as *follows*:

$$CD\left(\overrightarrow{\mathrm{cc_i}}, \overrightarrow{\mathrm{cc_j}}\right) = 1 - \ \|\overrightarrow{\mathrm{cc_i}}\| \left\|\overrightarrow{\mathrm{cc_j}}\right\| \cos\theta, \tag{6.2}$$

$$\cos\theta = \frac{\overrightarrow{cc_i} \cdot \overrightarrow{cc_j}}{\|\overrightarrow{\mathrm{cc_i}}\| \left\|\overrightarrow{\mathrm{cc_j}}\right\|} \tag{6.3}$$

Cosine distance is used because the similarity between cc$_i$ and cc$_j$ is shown using the directions of the two vectors. For the social context feature, we compute the distances between any two images *i* and *j* using the Euclidean distance [96] of their one-dimensional feature, called *D*, which is calculated as *follows*:

$$D(S_i, S_j) = \sqrt{(S_i - S_j)^2} \qquad (6.4)$$

Euclidean distance is used because the distance between any two users based on their number of *followers* is shown by computing the difference between the number of *followers* each user has.

With these distances, a distance matrix *M* is created for each feature. Then, each distance matrix is employed separately to mapper to cluster the data to analyze the relationship between the popularity of images and each feature.

Because the 80/20% rule was used to determine popularity, the ratio of popular images in each cluster is normalized by 0.2. Therefore, if the normalized ratio of popular images in a cluster is 1.0, then the effects of the feature on the popularity of the cluster were neutral. However, if the ratio of popularity is greater than 1.3, the popularity ratio is considered high, while if the ratio is less than 0.70, it is considered as a low ratio of popularity.

Regarding the images' popularity, the clusters can be classified into three groups: the low possibility of popularity, Gr1; neutral, Gr2; and the high possibility of popularity, Gr3, based on the criteria discussed above. If an image falls into Gr3, then it can be said that the image has a higher possibility of becoming popular, and if an image falls into Gr1, it has a lower possibility of becoming popular. Note that the ratio of popularity in each cluster is computed for three intervals: during the first hour, after the first day, and after the first week. Therefore, an image can belong to Gr1 in the first hour, then belongs to Gr3 after the first day, if the ratio of the popularity in that cluster increases after one day.

6.3 Prediction

Our mechanism predicts image popularity based on the cluster with the nearest centroid, which is determined by computing the distance between each image and the cluster's centroid. The centroid of a cluster d, i.e., C_d is computed as *follows*:

$$C_d = \frac{1}{N} \sum_{i \in im} x_i, \tag{6.5}$$

where *N* represents the number of images in the cluster *d*, while *x* contains the images in the cluster, which are represented using either of the two features discussed earlier.

For the prediction of images using the image content's feature, the cosine distance is used to compute the distance. Therefore, in order to predict the popularity of images, the nearest cluster's centroid is determined by finding the cluster with the centroid that has the lowest cosine distance with the image's content. The objective function is computed as *follows*:

$$y = \arg\ \min \left\| \mathrm{CD}(\mathrm{C_i}, \mathrm{CC_j}) \right\| , \tag{6.6}$$

where y represents the cluster with the highest cosine distance to the image's content.

On the other hand, for predicting the popularity of images using the social context's feature, Euclidean distance is used. In this case, the images will belong to the cluster that has the shortest distance to the image’s social context. In this case, the objective function will change slightly, which is computed as *follows*:

$$y = \text{arg min} \left\| \mathrm{d}(\mathrm{C_i}, \mathrm{S_j}) \right\| , \tag{6.7}$$

where y represents the cluster with the shortest Euclidean distance to the image’s social context.

Moreover, the images in our dataset are already labeled into popular and unpopular images using the Pareto principle as discussed Chapter 5; therefore, we use these labels to determine whether the images are assigned to the correct clusters, i.e., Gr1 or Gr3 or not. For example, if an image is popular, and clustered to one of the Gr3 clusters, it means that it is correctly identified as popular. If an image is clustered in one of the Gr1 clusters, and the image is unpopular, it means that it is correctly identified as unpopular. However, if a popular image is clustered to Gr1, this means that it is not correctly identified as a popular image. In our experiments, we predict both the popular and unpopular images.

6.4 Experimental Setup

For this experiment, the Instagram dataset presented in Chapter 5 is used. However, only three tie intervals are included: 1 hour, 1 day, and 1 week.

6.4.1 Implantation

As mentioned earlier, a mapper [65] is implemented to perform topological data analysis. It is available in a Python package. Density estimation as the filter function is used.

In order to convert captions to numerical form, Gensim, a Python library that implements Word2vec is implemented as discussed in Chapter 5. In order to compare topological data analysis and clustering algorithms, *k*-means and hierarchical clustering are implemented. Hierarchical clustering is implemented using Scikit-learn [83]; we have used an average linkage, and for connectivity, we have employed kneighbors graph algorithm. In order to determine the cut-off, we have used the parameter n–cluster in [83].

In addition, *k*-means is implemented using the Natural Language ToolKit [97]. For selecting the initial means, *k*-means++ is used [98]. Both packages are implemented in Python. The number of clusters varies between 5 and 15 to observe their effects on popularity; however, only experiments with five clusters are presented since the results are almost identical.

6.4.2 Evaluation

In order to evaluate the accuracy of the three approaches, the F-score is computed. F-score computes both precisions and recalls to compute the accuracy of the test, which represents the harmonic mean of precision and recall. It is computed as below:

$$\mathrm{F}1 = 2 \times \frac{\mathrm{percision} \times \mathrm{recall}}{\mathrm{precision} + \mathrm{recal}} \tag{6.8}$$

F-score is computed for both prediction classes: popular and unpopular images.

6.5 Empirical Results Using Topological Data Analysis

In this section, we discuss the experiments and results for topological data analysis. Topological data analysis is employed using the two features discussed earlier to cluster the images in the training dataset and then compute the ratio of popularity in each cluster to identify clusters with high or low ratios of popularity. The number of intervals used in the experiment is five as mentioned earlier. Then, we predict the popularity of images using the proposed approach.

6.5.1 Clustering

First, we employed the mapper using the image content feature. The results show that from cluster 1 to cluster 5, the ratios of popular images increases. Cluster 1 has the lowest ratio, 30% lower than neutral, and cluster 5 has the highest, 55% higher than neutral, while clusters 2–4 have neutral ratios of popularity. Therefore, we assigned cluster 1 to Gr1, cluster 2–4 to Gr2 and cluster 5 to Gr3.

Next, we employed the social context feature to the mapper, and the results show that the ratios of popularity have increased significantly. In this experiment, the ratios of popularity decreased from clusters one to five, which produces a monotonic decrease relationship between the clusters. Cluster 1 has the highest ratio of popularity, 305% higher than neutral, while cluster 5 has the lowest ratio of popularity, 95% lower than neutral. No cluster with a neutral ratio of popularity is observed in this experiment. Clusters 1 and 2 are assigned to Gr3, while the remaining clusters are assigned to Gr1. Both features generate a monotonic trend since the ratios of popularity increases or decreases along the clusters. The trends are shown using a sample from the dataset in Figures 6.2 and 6.3.

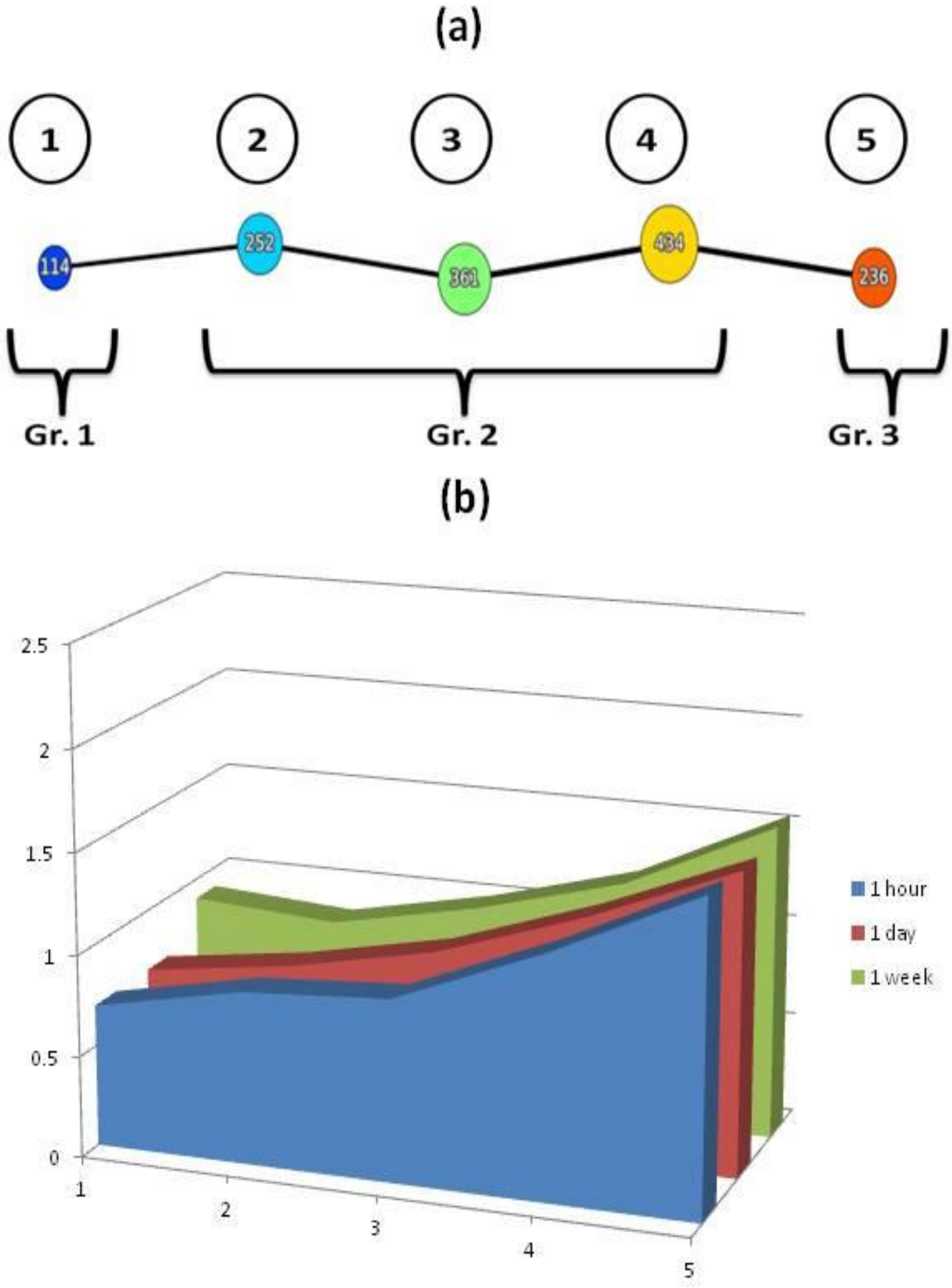


Figure 6.2. (Figure a) represents the clusters generated by mapper using the content feature, while (Figure b) represents the ratio of popularity in the cluster among the clusters (x-axis represent the clusters, while the y-axis represents the ratio of popularity.

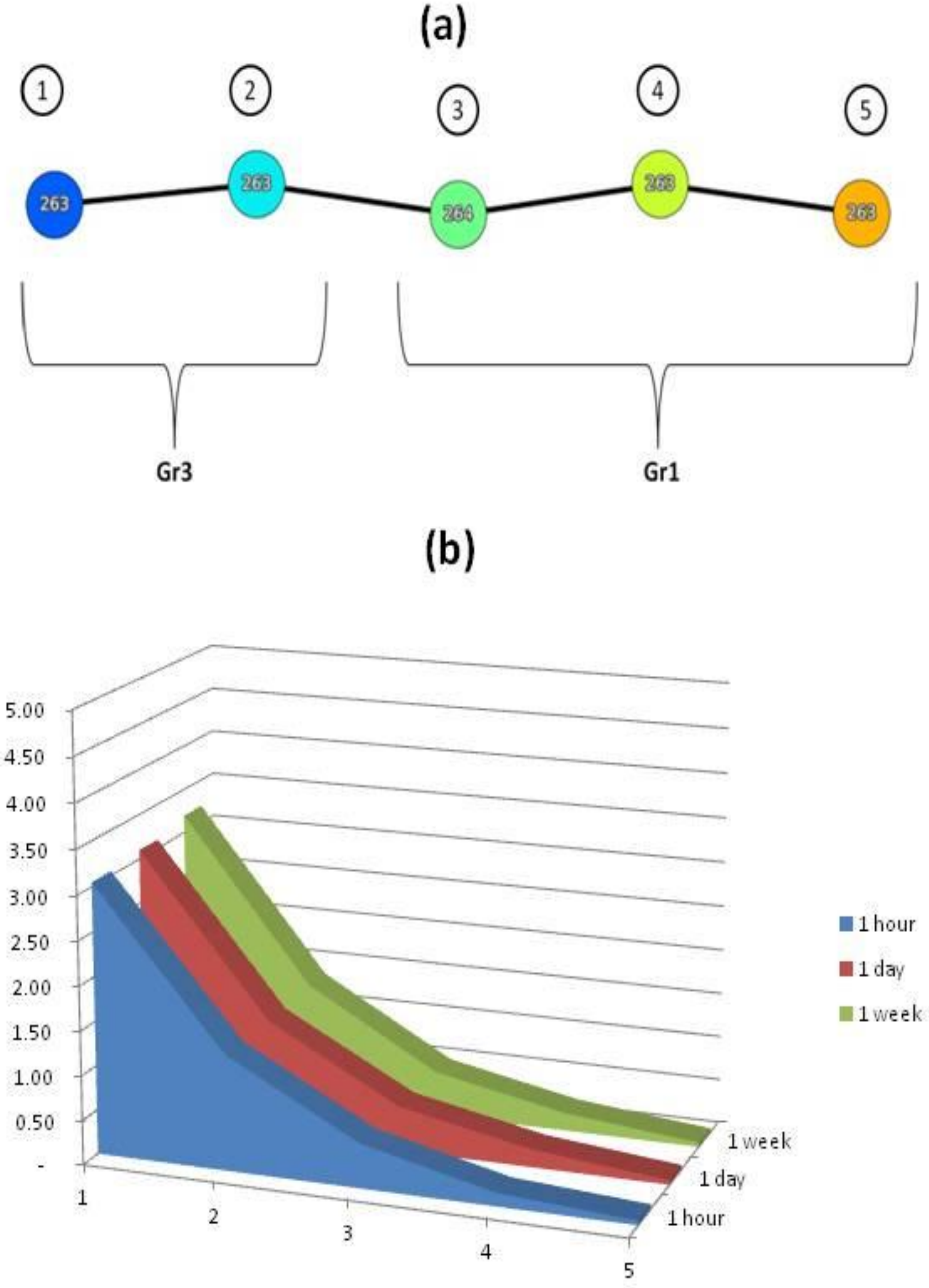


Figure 6.3. (Figure a) represents the clusters generated by mapper using the social context feature, while (Figure b) represents the ratio of popularity in the cluster among the clusters (x-axis represent the clusters, while the y-axis represents the ratio of popularity**.**

6.5.2 Prediction

In this experiment, we have predicted the popular and unpopular images using the two features. Using the image content, topological data analysis achieved an accuracy of 23% for predicting the popular images during the first hour. The accuracy of prediction has stayed the same for first day and first week periods. As for the prediction of unpopular images, topological data analysis achieved an accuracy of 68% for the first-hour prediction. Then, the accuracy has decreased to 31% for the first day and first week periods.

On the other hand, the results have increased significantly when the social context is used. During the first hour, topological data analysis achieves an accuracy of 67% for predicting the popular images. For predicting the unpopular images, the accuracy has increased to 82%. For both predictions of popular and unpopular images, the accuracy stayed the same over the first day and week periods. The results are summarized in the *follow*ing Table. For both features, the accuracy rates for the prediction of unpopular images are higher than the accuracy rates for the prediction of popular images because 80% of the images in our dataset are unpopular based on the Pareto principle.

6.6 Empirical Results Using Clustering Algorithms

In order to compare the topological data analysis approach with the clustering algorithms, we employed *k*-means and hierarchical clustering. In addition, the same distance metrics that are used for topological data analysis are used for *k*-means and hierarchical clustering.

6.6.1 *k*-means

k-means [99] is one of the most popular clustering algorithms. It clusters data into a set of clusters, i.e., k, based on the nearest mean. In *k*-means, connectivity has no meaning. Therefore, there are no relationships between clusters. *k*-means is employed using both features.

6.6.1.1 Clustering

First, we employed the image content feature, and the results show that clusters 2–5 have neutral ratios of popularity and are assigned to Gr2. However, cluster 1 has a low ratio of popularity, 6% lower than neutral, and therefore is assigned to Gr1.

Second, the social context feature is used. The ratios of popularity have increased significantly as observed using topological data analysis. The result shows that clusters 2 and 3 have low ratios of popularity, 28% lower than neutral and 66% lower than neutral, respectively. They are assigned to Gr1. Other clusters have high ratios of popularity. Cluster 4 has a perfect ratio of popularity, at 100%. Cluster 5 has a popularity ratio that is 58% higher than neutral, while cluster 1 has a ratio that is 232% higher than neutral. Clusters 1 and 4–5 are assigned to Gr3.

Table 6.1. Accuracy of topological data analysis for predicting the popular and unpopular images using the

6.6.1.2 Prediction

As discussed in the previous subsection, *k*-means failed to find any cluster with a high ratio of popularity when the image content is employed. Therefore, the prediction accuracy rate for predicting the popular images is 0.0%. However, for predicting the unpopular images, *k*-means achieved an accuracy rate of 0.39% for the first-hour prediction, and then the accuracy rate has decreased to 0.31% for the first day and week.

On the other hand, the accuracy rate of popular images using the social context has increased significantly to 0.63%. Moreover, the accuracy rates for predicting the unpopular images have increased to 0.85%. For the two predictions, the accuracy rates have stayed the same over the three-time frames. The results are summarized in the *follow*ing table.

Table 6.1. The accuracy of topological data analysis for predicting the popular and unpopular images using the image content and social context.

Accuracy Rates				
Periods	**Image Content**		**Social Context**	
	Popular Images	**Unpopular Images**	**Popular Images**	**Unpopular Images**
Hour	0.23	0.68	0.67	0.82
Day	0.23	0.31	0.67	0.82
Week	0.23	0.23	0.67	0.82

Table 6.2. Accuracy of k-means for predicting the popular and unpopular images using the image content and social context.

Accuracy Rates				
Periods	**Image Content**		**Social Context**	
	Popular Images	**Unpopular Images**	**Popular Images**	**Unpopular Images**
Hour	0	0.39	0.63	0.85
Day	0	0.17	0.63	0.85
Week	0	0.17	0.63	0.85

6.6.2 Hierarchical Clustering

In hierarchical clustering algorithm [100], clustering is performed differently. It builds a hierarchy of clusters. In hierarchical clustering, connectivity exists. Therefore, relationships exist between clusters.

6.6.2.1 Clustering

Using the image content feature, the result shows a new case, which occurred in cluster 4. Cluster 4 has a popularity ratio of 0, which means that in this cluster, the possibility for an image to become popular is 0%. This cluster is assigned to Gr1. The remaining clusters have neutral ratios of popularity and are assigned to Gr2. However, the ratio of popularity in cluster 1 has become higher than neutral after the first hour. Therefore, cluster 1 is assigned to Gr3 for the first day and week periods. For the connectivity part, no meaningful trend is detected.

Next, we employed the social context feature, and as observed in the other experiments that are based on the social context feature, the ratios of popularity have increased significantly. Cluster 4 has a popularity ratio of 0. Cluster 1 has a ratio that is 32% lower than neutral. Both clusters are assigned to Gr1. Clusters 3, 4 and 5 have high ratios of popularity: 140%, 180%, and 295% higher than neutral, respectively. They are assigned to Gr3. Moreover, the connectivity between these clusters is represented as a monotonic increase in the ratios of popularity along the connected clusters.

6.6.2.2 Prediction

As discussed in the previous subsection, hierarchical clustering failed to find any clusters with a high ratio of popularity during the first hour using the image content feature. Therefore, the accuracy rate for predicting the popular images is 0.0% during the first hour. However, as mentioned earlier, the ratio of popularity in cluster 1 has become higher than neutral; therefore, hierarchical clustering predicted popular images with an accuracy rate of 0.19 for the first day and week time frames. For predicting the unpopular images, hierarchical clustering achieved an accuracy of 49% during the first hour, and 18% for the first day and week.

As for the social context feature, the accuracy rate for predicting the popular images has increased significantly to 0.66%. Moreover, the accuracy rates for predicting the unpopular images have increased to 0.81%. For the two predictions, the accuracy rates have stayed the same over the three time periods. The results are summarized the *follow*ing table.

Table 6.3. Accuracy of hierarchical clustering for predicting the popular and unpopular images using the image content and social context.

Accuracy Rates				
Periods	**Image Content**		**Social Context**	
	Popular Images	**Unpopular Images**	**Popular Images**	**Unpopular Images**
Hour	0	0.49	0.66	0.81
Day	0.19	0.18	0.66	0.81
Week	0.19	0.18	0.66	0.81

6.7 Comparison

In this section, we will compare the performances of the three approaches in terms of accuracy using the two features.

6.7.1 Image Content

In Figures 6.5 and 6.5, we plot the accuracy rates for predicting the popular and unpopular images using the three approaches. The results show that topological data analysis outperforms the other approaches for predicting the popular and unpopular images. This shows that topological data analysis performs better than traditional data mining techniques when a high dimensional feature is employed, i.e., image content. In terms of the changes in the prediction accuracy rates over time, three approaches achieved high accuracy rates for predicting the unpopular images during the first hour.

However, the accuracy rates decreased after that. However, for predicting the popular images, the three approaches have the same accuracy rated over different time frames, except for hierarchical clustering, because, as discussed before, during the first hour, hierarchical clustering could not find any cluster with a high ratio of popular images. The results show that the popularity of images is saturated during the first week.

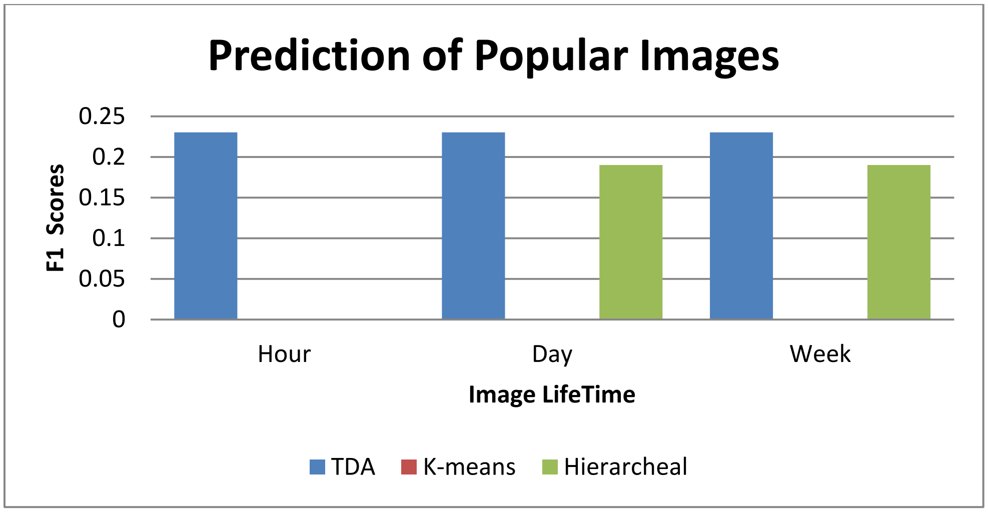


Figure 6.4. Accuracy rates for the three approaches using the image content feature for the prediction of popular images.

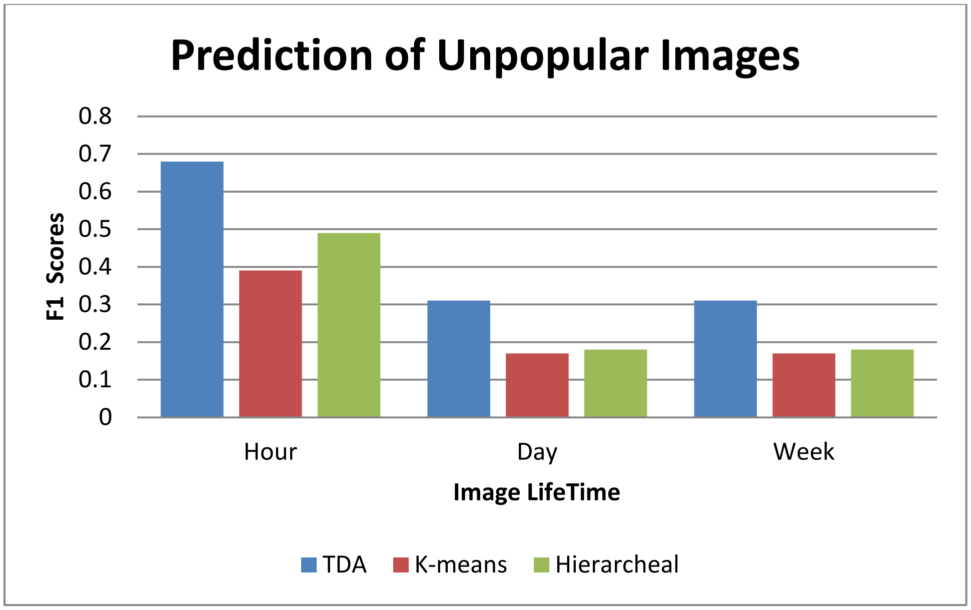


Figure 6.5. Accuracy rates for the three approaches using the image content feature for the prediction of unpopular images.

6.7.2 Social Context

The three approaches have very similar accuracy rates for predicting the popular and unpopular images. For predicting the popular images, topological data analysis slightly improves the accuracy rate with 1% more than hierarchical clustering and 4% more than *k*-means. As for predicting the unpopular images, *k*-means slightly improves the accuracy with 3% higher than topological data analysis, and 4% higher than hierarchical clustering. In terms of changes in accuracy rates over time, no changes are observed. The results show that when using a low dimensional feature, i.e., social context, traditional data mining techniques perform as well as topological data analysis.

The results show that social context achieves higher accuracy than image content, which supports the results produced by other studies that indicate that user's information has a large impact on images' popularity [35], [37]-[40], [43]. The results are plotted in Figures 6.6 and 6.7.

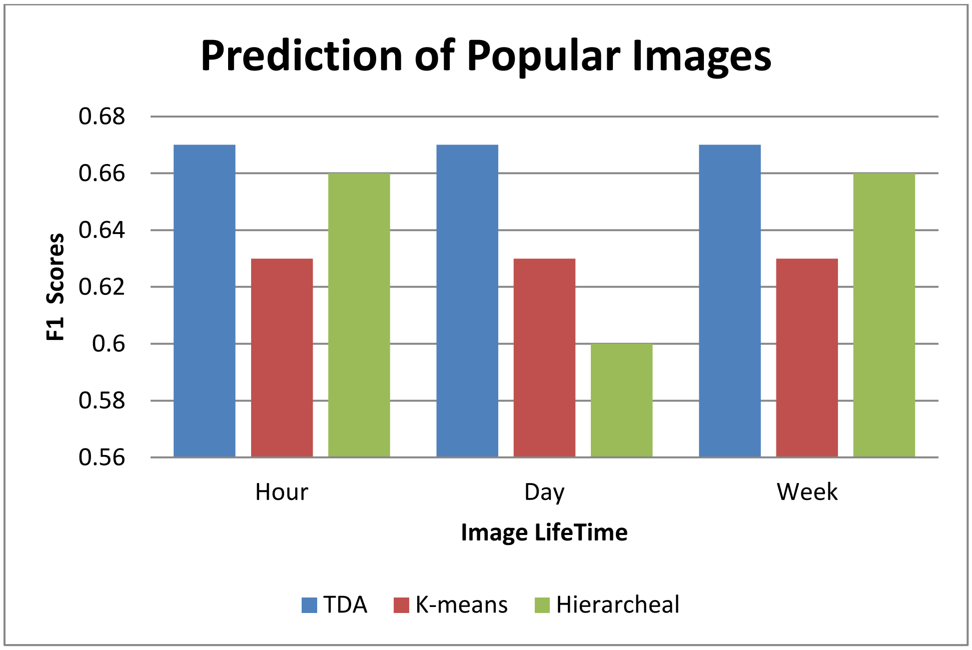


Figure 6.6. Accuracy rates for the three approaches using the social context feature for the prediction of popular images.

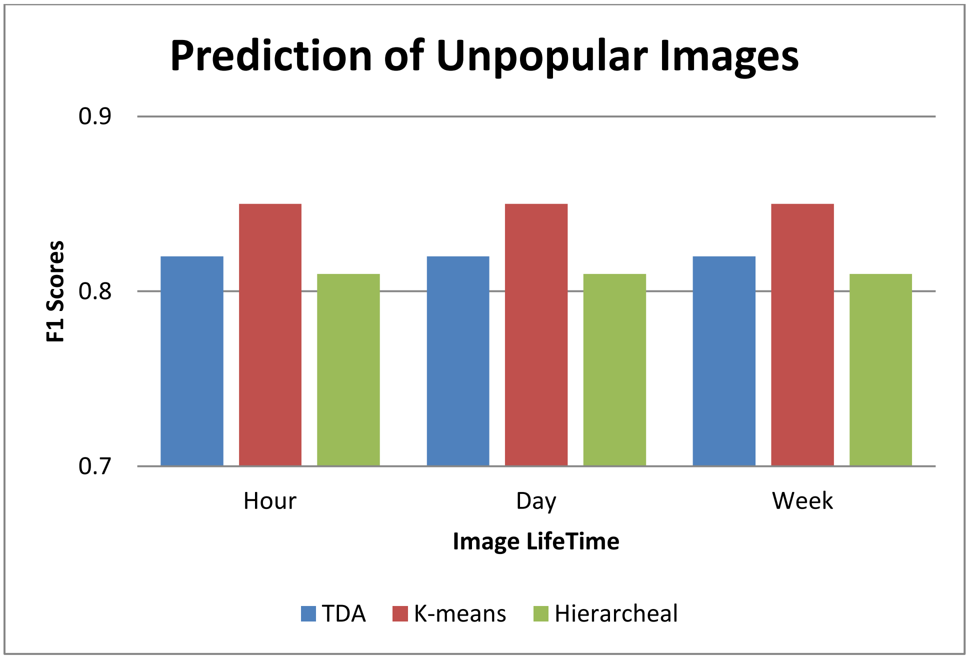


Figure 6.7. Accuracy rates for the three approaches using the social context feature for the prediction of unpopular images

6.8 Dealing with Noise

As mentioned before, topological data analysis connects data points by increasing the radii to find the shape of the data. The shapes are called homology. Topological data analysis aims to find the real shape of the data, not a noisy shape, which is not meaningful. The real shape of the data is the shape with longer persistent holes. The persistence of a hole is represented as the lifetime of the hole. Persistence homology analyzes the holes between the connected data samples to measure the persistent of the holes where the holes that stay longer is more persistent than holes that stay shorter, i.e., the longevity of the hole. As mentioned before, persistence homology computes the topological features of data at a different resolution. Therefore, if the hole doesn't last at a different resolution, it has low persistence and therefore is noise. The persistence of the hole is represented using a barcode. The long barcode represents significant feature while short barcode represents topological noise. The *follow*ing example will explain this concept.

The example is shown in Figure 6.8. In this figure, we begin with a set of data points in the first upper lower subgraph. From the first figure, we can see that the shape of the data is a circle. In order to find the real shape of the data, we employ persistent homology with different level of radii. After increasing the radii, the second subgraph shows that there are many holes among the data points, which shows that the real shape is still not found because there are many holes with short persistent. The third subgraph has two holes with higher persistence; it also shows that many holes with short persistence have died. Then we increase the radii, and all the *follow*ing subgraphs show one hole with high persistence because this hole has lasted at a different resolution. Therefore, we can stop because we have a shape with a high persistent hole and all low persistent holes have died. Persistence homology is robust to noise because the shape of the data is not noisy due to the fact that the shape has a hole with high persistence. Note that in the first two subgraphs, the shape of the data that is noisy because there are many holes with short persistence. After the shape is found, we can take look at the data to observe the meaning of the shape. For example, in the experiment discussed in Sec. 6.5, the shape of the data was a line and the meaning of the shape is the monotonic relationship of the popularity ratio between the clusters.

In order to assess this process, we have conducted another experiment that performs high resolution, where we select 11 clusters instead of 5 to compare it with the topology presented in Sec. 6.5.

First, we employed the mapper using the image content feature. Cluster 1 has the lowest ratio of popular images, 65% lower than neutral. The ratio increases as it goes through the other clusters towards cluster 11. Cluster 11 is the highest, 85% higher than neutral. Cluster 1 and 3 are more than 30% lower than neutral, so they are assigned to Gr1. Clusters 9-11 show higher than 30% at one-hour data, so they are assigned to Gr3. All the other clusters are considered neutral, so they are assigned to Gr2. Like the low-resolution results, it shows a monotonic increase trend except in cluster 2.

Then, we employed the mapper using the social context feature. The highest ratio of popularity is found in cluster 1, 365% more than neutral, while the lowest ratio of popularity is found in clusters 9-11, 95% lower than neutral. As observed in the low-resolution experiment, the ratios of popularity decrease along the clusters. On the other hand, this experiment has a cluster with a neutral ratio of popularity, i.e., cluster 4. Clusters 1-3 are assigned to Gr3, cluster 4 is assigned to Gr2, and clusters 5-11 are assigned to Gr1. Using both features, the topologies and trends of popularity are similar to the low-resolution experiment, which shows that the shapes have holes with high persistent. The shapes of the data points for the two experiment is still a line as found in Sec. 6.5. However, the meanings of the two shapes produce almost monotonic relationships between the clusters, which show that these two shapes can have a little bit of noise. The results of both experiments are shown in Figures 6.9 and 6.10.

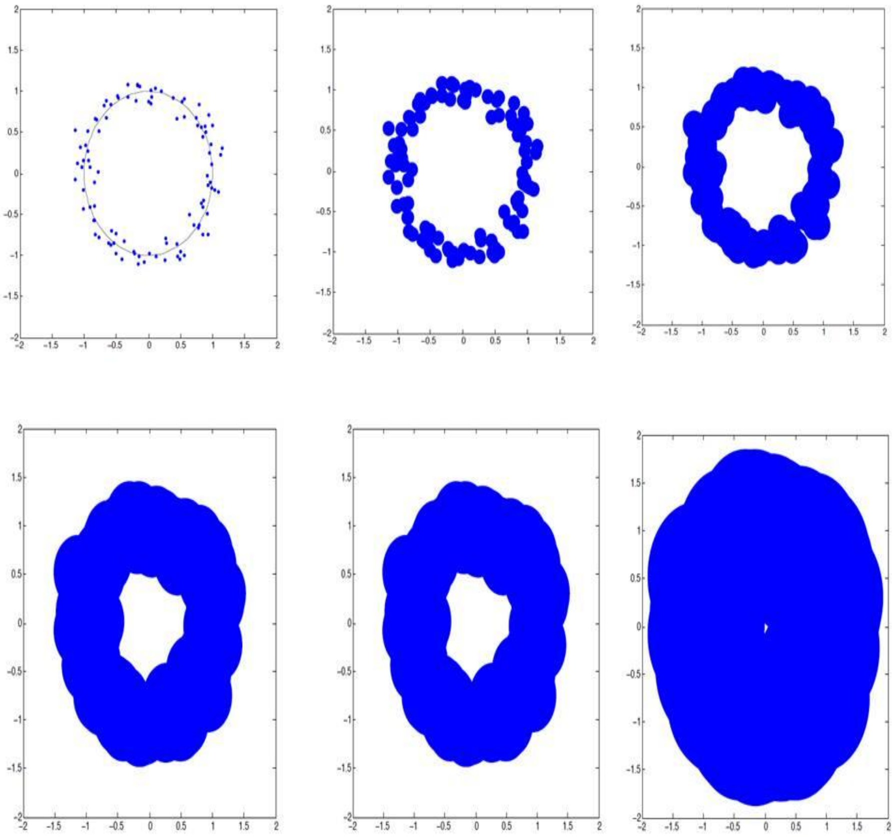


Figure 6.8. Example of how persistent homology compute persistence. The first upper left subgraph represents the data samples before we employ the persistent homology. All the figures follow is after applying the persistent homology with different radii.

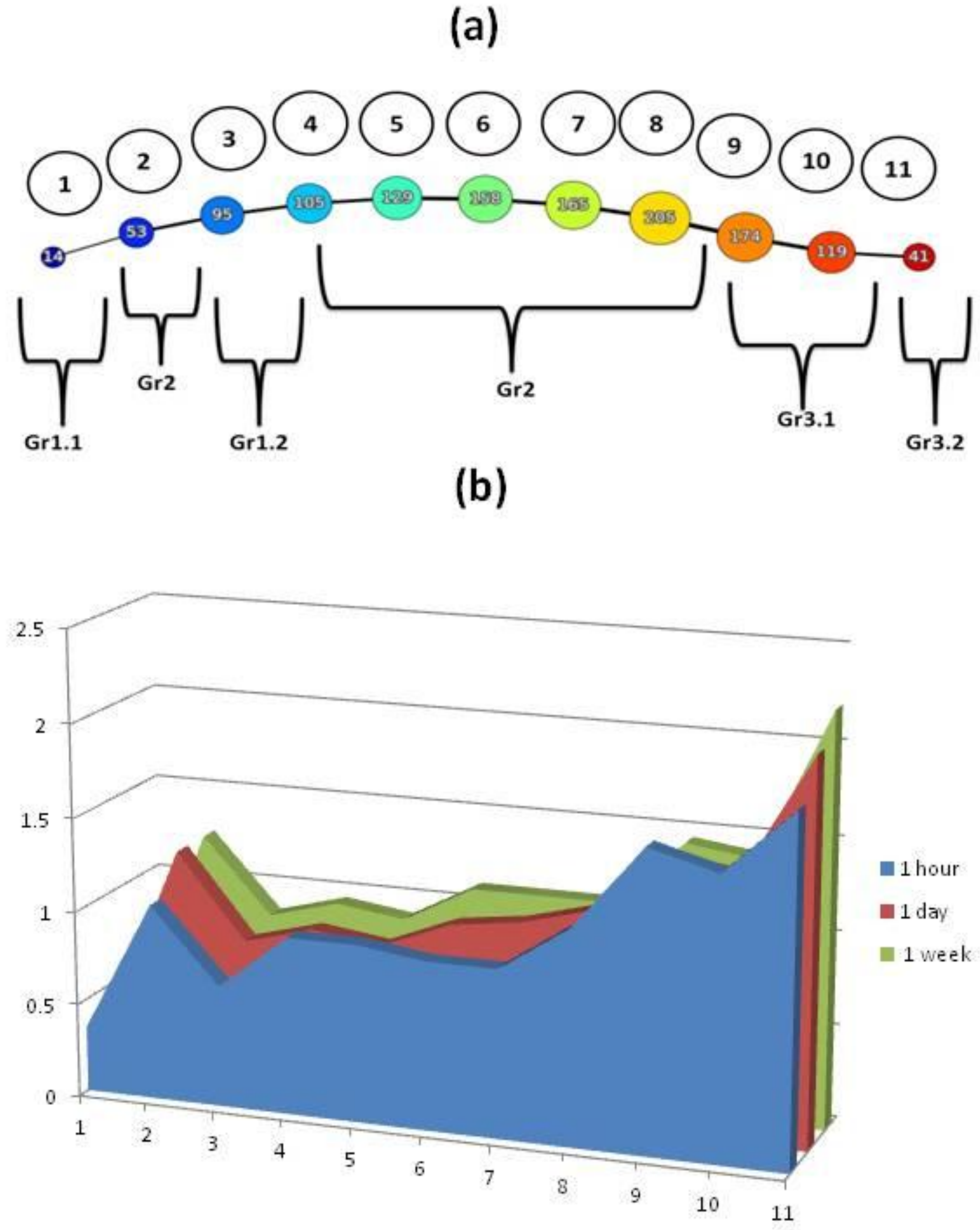


Figure 6.9. (Figure a) represents the clusters generated by mapper using the social content feature, while (Figure b) represents the ratio of popularity in the cluster among the clusters (x-axis represent the clusters, while the y-axis represents the ratio of popularity (High-resolution experiment)**.**

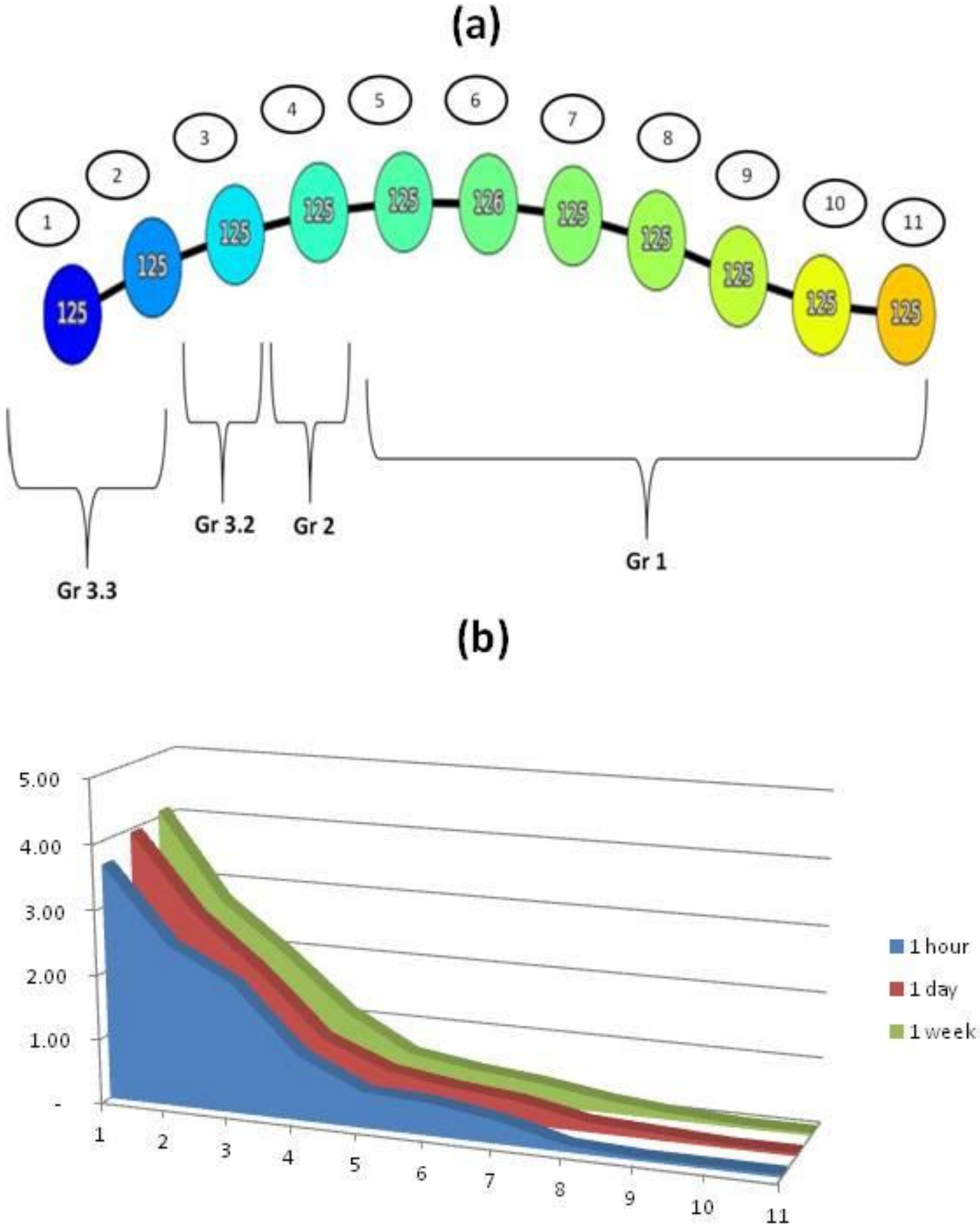


Figure 6.10. (Figure a) represents the clusters generated by mapper using the social context feature, while (Figure b) represents the ratio of popularity in the cluster among the clusters (x-axis represent the clusters, while the y-axis represents the ratio of popularity (High-resolution experiment)**.**

6.9 Discussion

In this chapter, the feasibility of topological data analysis for mining social network data is explored. The problem of image popularity is investigated by analyzing the effects of image content and social context on image popularity. In order to address this problem, random images are crawled along with their metadata from Instagram. Then, the images' captions are converted to numerical vectors using Word2vec. In addition, the normalized number of *followers* is used to represent the social context. Then, the distances of each feature are calculated and applied it to the mapper. These features are then employed to *k*-means and hierarchical clustering for comparing topological data analysis and clustering algorithms. Then, both the popular and unpopular images are predicted based on how close the images are to the centroid of the clusters. The results exhibited several outcomes:

1. Topological data analysis is feasible for social network analysis and mining;
2. Image content and social context have correlations to image popularity;
3. Topological data analysis significantly outperformed traditional clustering algorithms using the high dimensional feature, i.e., image content. It achieved higher accuracy rates than *k*-means and hierarchical clustering algorithms. It also generated meaningful connection between the clusters, i.e., a monotonic increase in the popularity ratio along the connected clusters;
4. For predicting the popularity of images using the low dimensional feature, i.e., social context, traditional data mining techniques perform as well as topological data analysis;

5. The results show that using the context feature improves the accuracy rates significantly, which confirms that the popularity of images is highly related to users' popularity;
6. For the changes of popularity over time, a trend is only observed for the prediction of popular images using the image content;
7. Lastly, the results show that popularity of images is saturated in a short period of time.

In conclusion, in order to address high dimensional and noisy data, topological data analysis proved to outperform traditional clustering algorithms. It also showed that the geometric shape of data matters and can be adapted to produce meaningful information. With regard to future work, it would be interesting to investigate feature integration using topological data analysis since topological data analysis relies on distances.

CHAPTER 7: CONCLUSIONS

In this thesis, social networks data is analyzed to improve information diffusion by predicting influential users and popular images. The proposed approaches are employed to several social networks, including Flickr, Instagram, and Digg.

For measuring the influence of users, a hybrid influence measurement is proposed, which is based on users' structural network locations and their attributes. For measuring users' influence based on their users' structural locations, several centrality analysis techniques, including PageRank and Eigenvector are employed. As for computing the user's influence based on their attributes, we adopt users’ attributes from social networks, more specifically users’ activeness. In order to represent users’ activeness, the number of uploaded posts is used. Both types of influence are integrated to predict the influential users. The influence measurements are then compared with the ground truth in term of correlation. The results show that users attributes outperform users’ structural location in predicting influential users. Moreover, the correlation increases when we integrate both features. The approach outperforms existing measurements with a correlation rate of 0.50 on Flickr dataset, and 0.90 on Digg dataset. Integrating both user’s structural location and characteristics shows stable performance in different social networks.

In order to predict the future popularity of images, social context, content and early popularity are employed. In order to represent the social context, the number of *followers* of users who uploaded the images is used to analyze the effect of users' popularity on images popularity. For analyzing the effect of image content on its popularity, the keyword vectors is introduced to represent the semantics of images using their captions. The keyword vector is built on NLP and clustering techniques. Early popularity is further employed because our analysis shows that images' popularity can be saturated within a short time of images uploads. The results show that the social context can predict the future popularity of images with a sensitivity of 0.57, however, the sensitivity decreases over time to 0.55. The semantic feature is able to predict images popularity with a sensitivity of 0.38. Moreover, the accuracy increases over time to 0.40. The early popularity outperforms the other features in predicting the future popularity of images with a sensitivity rate of 0.90.

The stability of popularity of images is further investigated and the results show that the semantic of images is only feature that can predict the changes of popularity over time with a sensitivity rate of 0.34, where social context and early popularity improve the accuracy in predicting the stable popularity of images with a perfect accuracy rate of 1. The proposed semantic feature outperforms the related work in term of accuracy, increasing the accuracy rates to 0.69 and 0.78 on the benchmarked data.

However, traditional techniques used to predict the popularity of posts do not address the challenges arising from the nature of social network data, including noise and high dimensionality in data. Therefore, topological data analysis is adopted. Topological data analysis proved to be feasible for analyzing social network data and it also outperforms traditional machine learning algorithms. It also showed that the geometric shape of data matters and can be adapted to produce meaningful information. With regard to future work, it would be interesting to investigate applying topological data analysis to other problems in social network research. It also interesting to investigate feature integration using topological data analysis since topological data analysis relies on distances.

REFERENCES

[1] N. B. Ellison, "Social network sites: Definition, history, and scholarship," *Journal of Computer-Mediated Communication,* vol. 13, pp. 210-230, 2007.

[2] Twitter, "Twitter-Company," 2015. [Online]. Available: https://about.twitter.com/en_us/company.html. [Accessed: 29- Jan- 2015].

[3] D. Etherington, "Flickr at 10: 1M photos shared per day, 170% increase since making 1TB free," *February,* vol. 10, pp. 25-45, 2014.

[4] Instagram, ""Our Story," 2016. [Online]. Available: https://instagram-press.com/our-story/. [Accessed: 15- Jan- 2016].

[5] A. Guille, H. Hacid, C. Favre, and D. A. Zighed, "Information diffusion in online social networks: A survey," *ACM SIGMOD Record,* vol. 42, pp. 17-28, 2013.

[6] A. Leavitt, E. Burchard, D. Fisher, and S. Gilbert, "The influentials: New approaches for analyzing influence on Twitter," *Web Ecology Project,* vol. 4, pp. 1-18, 2009.

[7] R. Zafarani, M. A. Abbasi, and H. Liu, *Social media mining: an introduction*: Cambridge University Press, pp. 73–251, 2014.

[8] J. Weng, E.-P. Lim, J. Jiang, and Q. He, "Twitterrank: finding topic-sensitive influential twitterers," in *Proceedings of the third ACM international conference on Web search and data mining*, pp. 261-270, 2010.

[9] B. Sun and V. T. Ng, "Identifying influential users by their postings in social networks," in *Ubiquitous Social Media Analysis*, ed: Springer, pp. 128-151, 2013.

[10] M. Cha, H. Haddadi, F. Benevenuto, and P. K. Gummadi, "Measuring user influence in Twitter: the million *follower* fallacy," *ICWSM*, vol. 10, pp. 30, 2010.

[11] H. Kwak, C. Lee, H. Park, and S. Moon, "What is Twitter, a social network or a news media?," in *Proceedings of the 19th international conference on World wide web*, pp. 591-600, 2010.

[12] W. Maharani and A. A. Gozali, "Degree centrality and eigenvector centrality in Twitter," in *Telecommunication Systems Services and Applications (TSSA), 2014 8th International Conference on*, pp. 1-5, 2014.

[13] X. Li, S. Cheng, W. Chen, and F. Jiang, "Novel user influence measurement based on user interaction in microblog," in *Proceedings of the 2013 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining*, , pp. 615-619, 2013.

[14] Q. Liao, W. Wang, Y. Han, and Q. Zhang, "Analyzing the influential people in Sina Weibo dataset," in *2013 IEEE Global Communications Conference (GLOBECOM)*, , pp. 3066-3071, 2013.

[15] Y. Zhang, J. Mo, and T. He, "User influence analysis on micro blog," in *2012 IEEE 2nd International Conference on Cloud Computing and Intelligence Systems*, pp. 1474-1478, 2012.

[16] R. Ghosh and K. Lerman, "Predicting influential users in online social networks," *arXiv preprint arXiv:1005.4882*, 2010.

[17] L. Lü, Y.-C. Zhang, C. H. Yeung, and T. Zhou, "Leaders in social networks, the delicious case," *PloS one,* vol. 6, pp. e21202, 2011.

[18] I. Anger and C. Kittl, "Measuring influence on Twitter," in *Proceedings of the 11th International Conference on Knowledge Management and Knowledge Technologies,* pp. 31, 2011.

[19] F. Erlandsson, P. Bródka, A. Borg, and H. Johnson, "Finding influential users in social media using association rule learning," *Entropy,* vol. 18, pp. 164, 2016.

[20] U. Ishfaq, H. U. Khan, and K. Iqbal, "Modeling to find the top bloggers using sentiment features," in *Computing, Electronic and Electrical Engineering (ICE Cube), 2016 International Conference on*, , pp. 227-233, 2016.

[21] R. Khan, H. U. Khan, M. S. Faisal, K. Iqbal, and M. S. I. Malik, "An analysis of Twitter users of Pakistan," *International Journal of Computer Science and Information Security,* vol. 14, pp. 855, 2016.

[22] E. Oro, C. Pizzuti, N. Procopio, and M. Ruffolo, "Detecting topic authoritative social media users: a Multilayer network approach," *IEEE Transactions on Multimedia,* 2017, to be published.

[23] C. Lee, H. Kwak, H. Park, and S. Moon, "Finding influentials based on the temporal order of information adoption in Twitter," in *Proceedings of the 19th international conference on World wide web*, , pp. 1137-1138, 2010.

[24] C. Sun, L. Zhang, and Q. Li, "Who are influentials on micro-blogging services: evidence from social network analysis," in *PACIS*, , p. 25, 2013.

[25] E. Bakshy, J. M. Hofman, W. A. Mason, and D. J. Watts, "Everyone's an influencer: quantifying influence on Twitter," in *Proceedings of the fourth ACM international conference on Web search and data mining*, , pp. 65-74, 2011.

[26] C. F. Reilly, D. Salinas, and D. De Leon, "Ranking users based on influence in a directional social network," in *Computational Science and Computational Intelligence (CSCI), 2014 International Conference on*, , pp. 237-240, 2014.

[27] U. Brandes, "A faster algorithm for betweenness centrality*," *Journal of mathematical sociology*, vol. 25, pp. 163-177, 2001.

[28] J. Scoot, "social network analysis," ed: Newberry Park CA: Sage, pp. 110–180, 1992.

[29] L. Tang and H. Liu, "Community detection and mining in social media," *Synthesis Lectures on Data Mining and Knowledge Discovery*, vol. 2, pp. 1-137, 2010.

[30] X. Yi, Y. Han, and X. Wang, "The evaluation of online social network's nodes influence based on user's attribute and behavior," in *Frontiers in Internet Technologies*, ed: Springer, pp. 9-20, 2013.

[31] P. Bonacich and P. Lloyd, "Eigenvector-like measures of centrality for asymmetric relations," *Social networks*, vol. 23, pp. 191-201, 2001.

[32] H. Yu, X. F. Bai, C. Huang, and H. Qi, "Prediction algorithm for users *Retweet* times," vol. 83, pp.9-13, 2015.

[33] L. Hong, O. Dan, and B. D. Davison, "Predicting popular messages in Twitter," in *Proceedings of the 20th international conference companion on World wide web*, , pp. 57-58, 2011.

[34] T. R. Zaman, R. Herbrich, J. Van Gael, and D. Stern, "Predicting information spreading in Twitter," in *Workshop on computational social science and the wisdom of crowds, nips*, pp. 17599-601, 2010.

[35] P. J. McParlane, Y. Moshfeghi, and J. M. Jose, "Nobody comes here anymore, it's too crowded; Predicting image popularity on Flickr," in *Proceedings of International Conference on Multimedia Retrieval*, p. 385, 2014.

[36] A. Khosla, A. Das Sarma, and R. Hamid, "What makes an image popular?," in *Proceedings of the 23rd international conference on World wide web*, pp. 867-876, 2014.

[37] S. Cappallo, T. Mensink, and C. G. Snoek, "Latent factors of visual popularity prediction," in *Proceedings of the 5th ACM on International Conference on Multimedia Retrieval*, , pp. 195-202, 2015.

[38] E. F. Can, H. Oktay, and R. Manmatha, "Predicting *Retweet* count using visual cues," in *Proceedings of the 22nd ACM international conference on Conference on information & knowledge management*, , pp. 1481-1484, 2013.

[39] K. Yamaguchi, T. L. Berg, and L. E. Ortiz, "Chic or social: Visual popularity analysis in online fashion networks," in *Proceedings of the 22nd ACM international conference on Multimedia*, pp. 773-776, 2014.

[40] L. C. Totti, F. A. Costa, S. Avila, E. Valle, W. Meira Jr, and V. Almeida, "The impact of visual attributes on online image diffusion," in *Proceedings of the 2014 ACM conference on Web science*, pp. 42-51, 2014.

[41] Fiolet, "Analyzing image popularity on a social media platform," *M.S. Thesis* Col. of Sci., University of Amsterdam, 2014. *Accessed on January, 4*, 2015, Unpublished

[42] X. Niu, L. Li, T. Mei, J. Shen, and K. Xu, "Predicting image popularity in an incomplete social media community by a weighted bi-partite graph," in *2012 IEEE International Conference on Multimedia and Expo*, pp. 735-740, 2012.

[43] F. Gelli, T. Uricchio, M. Bertini, A. Del Bimbo, and S.-F. Chang, "Image popularity prediction in social media using sentiment and context features," in *Proceedings of the 23rd ACM international conference on Multimedia*, , pp. 907-910, 2015.

[44] S. Aloufi, S. Zhu, and A. El Saddik, "On the prediction of Flickr image popularity by analyzing heterogeneous social sensory data," *Sensors,* vol. 17, p. 631, 2017.

[45] J. Hu, T. Yamasaki, and K. Aizawa, "Multimodal learning for image popularity prediction on social media," in *Consumer Electronics-Taiwan (ICCE-TW), 2016 IEEE International Conference on*, pp. 1-2, 2016.

[46] M. Mazloom, R. Rietveld, S. Rudinac, M. Worring, and W. van Dolen, "Multimodal popularity prediction of brand-related social media posts," in *Proceedings of the 2016 ACM on Multimedia Conference*, pp. 197-201, 2016.

[47] T. Hogg and K. Lerman, "Social dynamics of Digg," *EPJ Data Science,* vol. 1, pp. 1-26, 2012.

[48] K. P. Murphy, "Naive Bayes classifiers," *University of British Columbia,* pp. 1–8, 2006.

[49] V. N. Vapnik and V. Vapnik, *Statistical learning theory* vol. 1: Wiley, New York, pp. 1–12, 1998.

[50] A. Jordan, "On discriminative vs. generative classifiers: A comparison of logistic regression and naive Bayes," *Advances in neural information processing systems,* vol. 14, p. 841, 2002.

[51] G. G. Chowdhury, "Natural language processing," *Annual review of information science and technology,* vol. 37, pp. 51-89, 2003.

[52] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Advances in neural information processing systems*, pp. 3111-3119, 2013.

[53] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "word2vec," ed: accessed 2014-04--15. T. Mikolov, K. Chen, G. Corrado, and J. Dean, "word2vec," ed: accessed 2014-04--15. https://code.google.com/archive/p/word2vec/, 2014.

[54] A. K. Jain and R. C. Dubes, *Algorithms for clustering data*: Prentice-Hall, Inc., pp. 55–57, 1988.

[55] J. MacQueen, "Some methods for classification and analysis of multivariate observations," in *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, pp. 281-297, 1967.

[56] C. C. Aggarwal, "An introduction to social network data analytics," in *Social network data analytics*, ed: Springer, pp. 1-15, 2011.

[57] G. Carlsson, "Topology and data," *Bulletin of the American Mathematical Society,* vol. 46, pp. 255-308, 2009.

[58] J. R. Munkres, *Elements of algebraic topology* vol. 2: Addison-Wesley Menlo Park, pp. 54-59, 1984.

[59] S. Bansal and D. Choudhary, "Topological Data Analysis," pp. 3-17, 2014

[60] H. Cartan and S. Eilenberg, *Homological Algebra (PMS-19)* vol. 19: Princeton University Press, pp. 53–70, 2016.

[61] N. Murphy, "Topological Data Analysis," M.S. Thesis, Dept. of Math. and Stat., Colby College, 2016. Accessed on: Jun., 15, 2017. Available: https://www.colby.edu/math/program/honorsprojects/2016-Murphy-HonorsThesis.pdf

[62] A. Zomorodian and G. Carlsson, "Computing persistent homology," *Discrete & Computational Geometry,* vol. 33, pp. 249-274, 2005.

[63] D. Cohen-Steiner, H. Edelsbrunner, and J. Harer, "Stability of persistence diagrams," *Discrete & Computational Geometry*, vol. 37, pp. 103-120, 2007.

[64] B. MICHEL, "Statistics and Topological Data Analysis."

[65] D. Müllner and A. Babu, "Python Mapper: An open-source toolchain for data exploration, analysis, and visualization," http://math.stanford.edu/muellner/mapper, 2013.

[66] G. Singh, F. Mémoli, and G. E. Carlsson, "Topological methods for the analysis of high dimensional data sets and 3D object recognition," in *SPBG*, pp. 91-100, 2007.

[67] L. Rashotte, *Social influence.* The Blackwell encyclopedia of social psychology, pp.562-563, 2004.

[68] D. J. Watts and P. S. Dodds, "Influentials, networks, and public opinion formation," *Journal of consumer research*, vol. 34, pp. 441-458, 2007.

[69] E. Katz, "The two-step flow of communication: An up-to-date report on an hypothesis," *Public opinion quarterly,* vol. 21, pp. 61-78, 1957.

[70] M. S. Granovetter, "The strength of weak ties," *American journal of sociology,* pp. 1360-1380, 1973.

[71] H. Li, J.-T. Cui, and J.-F. Ma, "Social influence study in online networks: A three-level review," *Journal of Computer Science and Technology,* vol. 30, pp. 184-199, 2015.

[72] J. Li, W. Peng, T. Li, and T. Sun, "Social network user influence dynamics prediction," in *Asia-Pacific Web Conference,* pp. 310-322, 2013.

[73] F. Probst, D.-K. L. Grosswiele, and D.-K. R. Pfleger, "Who will lead and who will *follow*: Identifying influential users in online social networks," *Business & Information Systems Engineering,* vol. 5, pp. 179-193, 2013.

[74] J. Sang and C. Xu, "Social influence analysis and application on multimedia sharing websites," *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM),* vol. 9, pp. 53, 2013.

[75] E. Katz and P. F. Lazarsfeld, *Personal Influence, The part played by people in the flow of mass communications*: Transaction Publishers, pp. 271-280, 1966.

[76] A. V. Aho and J. E. Hopcroft, *The design and analysis of computer algorithms*: Pearson Education India, pp. 44–90, 1974.

[77] B. A. Huberman, D. M. Romero, and F. Wu, "Social networks that matter: Twitter under the microscope," *Available at SSRN 1313405,* 2008.

[78] S. Bird, E. Klein, and E. Loper, "*Natural language processing with Python: Analyzing text with the natural language toolkit*," O'Reilly Media, Inc., pp. 109–128, 2009.

[79] J. Benesty, J. Chen, Y. Huang, and I. Cohen, "Pearson correlation coefficient," in *Noise reduction in speech processing*, ed: Springer, pp. 1-4, 2009.

[80] Merriam-Webster, "Popularity," https://www.merriam webster.com/dictionary/popularity, 2011.

[81] L. Oglesbee, "Writing captions," *Communication: Journalism Education Today*, vol. 32, pp. 2-6, 1998.

[82] R. Rehurek and P. Sojka, "Software framework for topic modelling with large corpora," in *In Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*, pp. 46-50, 2010.

[83] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, *et al.*, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research,* vol. 12, pp. 2825-2830, 2011.

[84] H. Honest and K. S. Khan, "Reporting of measures of accuracy in systematic reviews of diagnostic literature," *BMC health services research,* vol. 2, p. 1, 2002.

[85] X. Wu, X. Zhu, G.-Q. Wu, and W. Ding, "Data mining with big data," *IEEE transactions on knowledge and data engineering*, vol. 26, pp. 97-107, 2014.

[86] J. Fan, F. Han, and H. Liu, "Challenges of big data analysis," *National science review,* vol. 1, pp. 293-314, 2014.

[87] H. Becker, M. Naaman, and L. Gravano, "Learning similarity metrics for event identification in social media," in *Proceedings of the third ACM international conference on Web search and data mining*, pp. 291-300, 2010.

[88] H. Edelsbrunner, D. Letscher, and A. Zomorodian, "Topological persistence and simplification," *Discrete and Computational Geometry,* vol. 28, pp. 511-533, 2002.

[89] M. Nicolau, A. J. Levine, and G. Carlsson, "Topology-based data analysis identifies a subgroup of breast cancers with a unique mutational profile and excellent survival," *Proceedings of the National Academy of Sciences,* vol. 108, pp. 7265-7270, 2011.

[90] H. Edelsbrunner, D. Letscher, and A. Zomorodian, "Topological persistence and simplification," in *Foundations of Computer Science, 2000. Proceedings. 41st Annual Symposium on*, , pp. 454-463, 2000.

[91] M. Nicolau, R. Tibshirani, A.-L. Børresen-Dale, and S. S. Jeffrey, "Disease-specific genomic analysis: identifying the signature of pathologic biology," *Bioinformatics*, vol. 23, pp. 957-965, 2007.

[92] M. Gidea and Y. A. Katz, "Topological Data Analysis of financial time series: Landscapes of crashes," 2017.

[93] K. B. Schebesch and R. W. Stecking, "Topological Data Analysis for extracting hidden features of client data," in *Operations Research Proceedings 2015*, ed: Springer, pp. 483-489, 2017.

[94] D. Bonchev and N. Trinajstić, "Information theory, distance matrix, and molecular branching," *The Journal of Chemical Physics,* vol. 67, pp. 4517-4533, 1977.

[95] B. Larsen and C. Aone, "Fast and effective text mining using linear-time document clustering," in *Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 16-22, 1999.

[96] M. M. Deza and E. Deza, "Encyclopedia of distances," in *Encyclopedia of Distances*, ed: Springer, pp. 1-583, 2009.

[97] S. Bird, "NLTK: the natural language toolkit," in *Proceedings of the COLING/ACL on Interactive presentation sessions*, pp. 69-72, 2006.

[98] D. Arthur and S. Vassilvitskii, "k-means++: The advantages of careful seeding," in *Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms,* pp. 1027-1035, 2007.

[99] J. A. Hartigan and M. A. Wong, "Algorithm AS 136: A k-means clustering algorithm," *Journal of the Royal Statistical Society. Series C (Applied Statistics),* vol. 28, pp. 100-108, 1979.

[100] S. C. Johnson, "Hierarchical clustering schemes," *Psychometrika,* vol. 32, pp. 241-254, 1967.