A SOCIAL NETWORK ANALYTICS BASED RECOMMENDATION SYSTEM

SEDAT PAYAL

BOĞAZİÇİ UNIVERSITY

2019

A SOCIAL NETWORK ANALYTICS BASED RECOMMENDATION SYSTEM

Thesis submitted to the

Institute for Graduate Studies in Social Sciences

in partial fulfillment of the requirements for the degree of

Master of Arts

in

Management Information Systems

by

Sedat Payal

Boğaziçi University

2019

DECLARATION OF ORIGINALITY

I, Sedat Payal, certify that

- I am the sole author of this thesis and that I have fully acknowledged and documented in my thesis all sources of ideas and words, including digital resources, which have been produced or published by another person or institution;
- this thesis contains no material that has been submitted or accepted for a degree or diploma in any other educational institution;
- this is a true copy of the thesis approved by my advisor and thesis committee at Boğaziçi University, including final revisions required by them.

Signature...

Date ......12.08.2019..............................

ABSTRACT

A Social Network Analytics Based Recommendation System

In this thesis, different frameworks for a recommendation system based on social network analytics is investigated. In these frameworks, three different potential customer identification approaches are examined and corresponding successes are analyzed. In order to exploit the underlying network structure, three networks, restaurant-user, user-user and restaurant-restaurant, are generated. In the first approach, potential users are ranked and selected according to a combination of pagerank values and community scores of both restaurants and users. In the second approach, users are ranked according to the sentiments scores of their comments in conjunction with pagerank of restaurants. In the third approach, node embeddings for the restaurant-user network are computed and used to find the similarities between users and restaurants. Then, based on these similarities, potential users are ranked for a given focal restaurant. With the aim of comparing the successes of these three frameworks, dataset is splitted into three and success rates are calculated based on the percentage of the actual customers recommended by the generated models. Experiments in this research shows that Ranks framework utilizing the community structure together with the network ranking of both users and brands reached up to 50% and on average achieved 9.61% accuracy when the number of potential customers to be recommended is taken as 100. So, frameworks utilizing the underlying network structure can be exploited to improve the prediction capability of recommendation systems that find potential customers for a given company or brand.

ÖZET

Sosyal Ağ Analitiği Temelli Bir Öneri Sistemi

Bu tezde, sosyal ağ analitiklerine dayalı bir öneri sistemi için farklı çerçeveler incelenmiştir. Bu çerçevelerde, üç farklı potansiyel müşteri tanımlama yaklaşımı incelenmiş ve bunlara karşılık gelen başarılar analiz edilmiştir. Temel ağ yapısından yararlanmak için, restoran kullanıcısı, kullanıcı kullanıcısı ve restoran restoranı olmak üzere üç ağ üretilir. İlk yaklaşımda, potansiyel kullanıcılar hem restoranların hem de kullanıcıların topluluk sayfası puanlarının ve sayfa sıralaması değerlerinin bir birleşimine göre sıralanır ve seçilir. İkinci yaklaşımda kullanıcılar, yorumlarının duygu puanlarına ve restoranların sayfa sıralamasına göre sıralanır. Üçüncü yaklaşımda, restoran-kullanıcı ağı için düğüm yerleştirmeleri hesaplanır ve kullanıcılar ile restoranlar arasındaki benzerlikleri bulmak için kullanılır. Daha sonra, bu benzerliklere dayanarak, potansiyel kullanıcılar belirli bir odak restoranında sıralanır. Bu üç çerçevenin başarısını karşılaştırmak amacıyla veri kümesi üçe bölünmüş ve üretilen modeller tarafından önerilen gerçek müşteri yüzdesine göre başarı oranları hesaplanmıştır. Bu araştırmadaki denemeler, topluluk yapısını hem kullanıcıların hem de markaların ağ sıralamasıyla birlikte kullanan Sıralamalar çerçevesinin %50'ye kadar ulaştığını ve önerilen potansiyel müşteri sayısı 100 olarak alındığında ortalama % 9.61 doğruluk elde ettiğini göstermektedir. Altta yatan ağ yapısını kullanan çerçeveler, belirli bir şirket veya marka için potansiyel müşteriler bulan öneri sistemlerinin öngörme kabiliyetini geliştirmek için kullanılabilir.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

In the past, people used to shop in physical stores, in which the items availability are limited. But nowadays the Internet allows people to access vast amount of resources online. These resources make it harder to find an item or service that is needed. To help people solve this problem, recommendation systems are suggested to guide the people and provide them with more specialized options (Mahata, A., Saini, N., Saharawat, S., & Tiwari, R., 2016). A recommendation system is a computer program that recommends the most related items to users they would probably need by using user preferences, interactions and past behaviors. There are mainly 3 types of recommendation systems, content-based, collaborative and hybrid recommendations systems. With the ever-growing sea of data, people are in need of a means to filter out the unnecessary information according to their preferences to narrow down the available choices. Recommendation systems come into use for this issue and are applied to a majority of areas from restaurants to movie and location recommendations (Taneja, A., Gupta, P., Garg, A., Bansal, A., Grewal, K. P., & Arora, A., 2016).

Customer reviews come in handy when trying to filter out unwanted choices as they provide important information for consumers by relating to past experiences of customers who have made preferences and reflected on them. It is understood that consumers are factoring in the reviews of a product or service when making a purchase decision (Salehan & Kim, 2016). Customer reviews are also valuable to businesses as they are the thoughts of their current customers which could affect potential customers. Therefore, businesses are evaluating and inspecting customer reviews to improve the quality of their products and/or services. Recommendation systems are also incorporating the reviews and preferences of consumers into the recommendation process (Luca, 2016).

Targeted advertising is a type of filtered advertising, where the focus is on finding the most related audience or an audience with a specific characteristic. These characteristics may be demographic or psychographic such as income, age, opinions, interests, etc. Behavioural characteristics in the form of historical data can also be utilised such as browser and payment history. The aim of targeted advertising is to find the people that will most likely use or purchase their products and/or services, thus maximising their profits by minimising time and resource waste. Because the advertisements are targeted at the audience most likely to purchase or use the product or service, the investment on advertisement will be more likely to return the highest revenue rather than general distribution of the advertisement. Utilising information and communication technology (ICT) help minimise advertising costs as it is faster and less costly to reach more people online than other platforms such as printed advertisements or radio and televised advertisements. The most important element in online targeted advertisement is to uncover the potential audience to distribute the advertisement.

Therefore, in this thesis, network and textual data analysis approaches were combined to develop a framework to discover and recommend potential customers to restaurants for targeted advertising. Unlike traditional methods that seek target audiences through explicit networks of restaurants, implicit and explicit restaurant and user networks that were extracted from large amounts of historical user-restaurant interactions were utilized. Subsequently, three different frameworks that utilizes solely the network analytics approaches to identify the target audience were developed. The network types, ranking algorithms, community detection and embeddings are the basis for selecting target audiences of a given focal restaurant in the proposed frameworks. A potential user for a focal restaurant is the one who has not previously interacted with the focal restaurant but who were interested in other restaurants that are similar to the focal restaurant or are similar to the users who have been to the focal restaurant or a member of the same user community.

In the Ranks Framework, FinalRank approach was developed to identify the target audience. First, for a given restaurant for which potential customers are to be recommended, similar/related restaurants, which are assumed to belong to the same community, were identified through community detection algorithm applied to an undirected restaurant-restaurant network. Second, using a global directed restaurant-restaurant network, influence scores are calculated for restaurants and the first N restaurants are selected which have the highest N influence scores and belong to the same community of the focal restaurant, where N is a model parameter. Third, the initial target pool of users (customers) are selected by considering related restaurants of the focal restaurant. And then, similar users are identified by applying community detection algorithm to the undirected user-user network. In the fifth step, user CommunityRank scores are calculated for each user in initial target user pool users by considering the number of focal restaurant's users that are present in that community. In the sixth step, the final influence scores are calculated by combining user PageRank and CommunityRank scores with a hyperparameters for the all users in the initial target user pool. Finally target user list selected by considering their FinalRank scores.

In the second framework, a "SentiRank" framework was developed to identify the target audience. This framework follows the same steps as the Ranks

Framework until finding the initial target user pool. Then, sentiment analysis is used to select the users in that user pool. No user ranking and user community detection algorithms were used. User comments were analyzed to extract user positiveness similar to the study conducted by Zhang, Bhattacharyya and Ram (2016). Target users are selected for the focal restaurant according to their SentiRank values.

In the last framework, the "Embeddings" framework, a completely different approach named "Embedding" approach was considered. In this approach, an explicit restaurant-user network was utilized and node embeddings generated by using the node2vec algorithm (Grover & Leskovec, 2016). In order to find the embeddings, both user and restaurants are considered as nodes within the network. After running the node2vec algorithm, similarity detection was applied to all nodes. Then, the best hyperparameter for the target user calculation was selected. Finally, the target user list was selected according to these similarities.

To evaluate the performance of the target customer recommendation frameworks, experiments were performed on a large dataset collected from Zomato API. First, the collected data was partitioned into training, validation and testing datasets considering the periods of the interactions occurring between restaurants and users. Here, data in the training period was used for building a model for audience selection, whereas the data in the validation period was used for the hyperparameter optimization to select the best model parameters. The last dataset in the test period was used for testing the effectiveness of the frameworks. In order to obtain comparative results, the performances of the proposed approaches were analyzed with success rates of all frameworks by using precision-recall metrics.

The key contribution of this study is that it has three different novel frameworks for targeted advertising using a dataset related to online restaurant brands which was downloaded from Zomato and contains comparative success results of these frameworks.

In the remainder of the thesis, existing related studies are mentioned in the literature review chapter. Thereafter, in the methodology chapter, the proposed recommendation frameworks are described in detail. The results and findings chapter presents the simulation results of the proposed frameworks and compares them with each other. Finally, the conclusions of this thesis are given and future works are presented.

CHAPTER 2

LITERATURE REVIEW

2.1 Recommender system

Recommender systems have become widespread in the literature and can be found within many different fields and topics.

Recommender systems can be grouped as content-based, collaborative and hybrid systems. Content-based systems utilize information regarding the item in question as well as the user's preference profile. They hypothesize that if a user was interested in an item in the past, they will once again be interested in it in the future. Similar items are usually grouped based on their features. User profiles are constructed using historical interactions or by explicitly asking users about their interests.

2.1.1 Content based recommender system

Many studies based on content-based recommender system have been conducted such as, Lops P., de Gemmis M., Semeraro G. (2011). They first explain the basic concepts and the terminology and then provide an analysis of the up-to-date systems used in several systems, by explaining with advanced and classical methods to represent user profiles and items. The study of Rohani, Kasirun and Ratnavelu (2014) provides an enhanced content-based system incorporating information of the user's friends as well as the person's own preferences. In the paper by Di Noia,, Mirizzi, Ostuni, Romito and Zanker (2012), they developed a recommender system that depends on the information encoded in the web of data, they produced a content-based recommender system that use the data accessible inside Linked Open dataset (dbpedia, freebase and linkedMDB) to recommend movies to the users and they evaluate the effectiveness of their algorithms by measuring the accuracy with precision and recall metrics. The study of Di Noia, Mirizzi, Ostuni, Romito and Zanker (2015) shows an extensive survey of semantic depiction of user profiles and items to answer the basic problems of the content-based approaches and proposes a classification of semantic approaches into top-down and bottom-up methods that first one depends on the integration of external knowledge sources and second method depend on slight semantic representation using the idea that usage of a word in large textual documents determines the meaning of the word. Pazzani and Billsus (2007) discuss content-based recommendation systems thoroughly by explaining how many different domains it is used such as web pages, news articles, restaurants, television programs and items for sale. In the article by Van Meteren and Van Someren (2000) content-based filtering techniques are used to recommend small articles about home enhancements that use only positive feedbacks and needs to be very dynamic. Some studies propose a community-based scholar recommendation model and compare it with content based recommender system using Louvain's community detection algorithm for communities on academic social network dataset called SCHOLAT and they claim that they outperform the content-based user recommender system according to results (Chen et al., 2013).

2.1.2 Collaborative based recommender system

The other type of recommender system is collaborative systems that make recommendations according to preferences of similar people and use user's interactions with items and other users. Collaborative systems are based on the assumption that if two users like some common items then that users will probably like some common items in the future.

Many different types of collaborative based recommender system studies have been realized. In the study by Sarwar, Karypis, Konstan and Riedl (2001), it is claimed that they overcome some of the drawbacks of traditional collaborative filtering systems by proposing a new type of recommender system called item-based collaborative recommender system. In their paper they analyze different item-based recommendations and try to find different techniques for computing item-item similarities such as item-item correlation vs cosine similarities between items. They claim they have much better performance and better quality than even the best of traditional collaborative recommender systems. The study by Ekstrand, Riedl and Konstan (2011) argues that there is no recommender system that fits all and that recommender systems must be developed for specific tasks. In their paper they disscuss a wide variety of the available choices of collaborative filtering systems for different subjects and provide researchers an introduction to the important issues underlying recommenders. According to the study conducted by Schafer, Frankowski, Herlocker and Sen (2007), collaborative filtering is the process of evaluating items through the opinions of other people and collaborative filtering technology brings together the opinions of large communities. In their study, they first introduce the core concepts of the algorithm and design decisions and then they discuss how to evaluate collaborative filtering systems.

Another study about collaborate filtering was put forward by Herlocker, Konstan, Terveen and Riedl (2004). They analyze the important decisions in evaluating the collaborative filtering recommender systems such as the user tasks being evaluated, the types of analysis and datasets being used, the ways in which prediction quality is measured, the evaluation of prediction attributes other than quality, and the user-based evaluation of the system as a whole. They also show test results from their analysis of various metrics and all tested metrics drop into three classes which have strong within-class correlation. The study done by Bobadilla, Serradilla and Hernando (2009) is about memory-based collaborative filtering. They propose the idea that the user with greater knowledge has greater weight in the recommendation calculations. To implement this goal, they extended the core equations of the memory-based collaborative filtering and process the scores resulted by each user in a different number of level tests.

Another collaborative filtering recommender system is developed by Zheng, Ma, Lyu and King (2009). They propose a collaborative based web service recommender system (WSRec) to resolve difficulties in the web service selections. WSRec consists of user contribution mechanism and novel collaborative filtering algorithm for web service prediction. WSRec is implemented by Java language and they test it with thousands of public web services around the world and they collect millions of test results which they claim the WSRec achieves better prediction accuracy than other methods.

2.1.3 Hybrid recommender system

Hybrid systems combine both systems and replace the disadvantages of one system with the advantages of another system and thus build a more robust system. For example, by combining collaborative filtering methods, where the model fails when new items don't have ratings, with content-based systems, where feature information about the items is available, new items can be recommended more accurately and efficiently.

The studies of Bozanta and Kutlu (2018; 2019) are precises example of using a hybrid approach on recommender system for location-based services. They use a hybrid approach to eliminate drawbacks encountered in individual approaches by combining user-based and item-based collaborative filtering with content-based filtering and contextual information. They collected user-visit histories, venue related information (distance, category, popularity and price) and contextual information (weather, season, date and time of visits) related to individual user visits from Twitter, Foursquare and Weather Underground. According to their study they have better results than baseline approaches. Wei et al. (2014) also propose a hybrid system for recommending movies by using tags and ratings. They also further developed their model by incorporating additional information collected from historical user ratings.

The study by Burke (2007) surveys the two-part hybrid recommender system, compares four different recommendation techniques and seven hybridization strategies. They implement many hybrid novel combinations and compared each other, and they claim cascade and augmented hybrids work well according to test results.

In the study Tran, T., & Cohen, R. (2000, July). It is presented a novel hybrid recommender system that combines collaborative filtering and knowledge-based approaches and they also discuss the switching mechanism between these two approaches by providing good recommendations to users.

Another study about hybrid recommendations system is conducted by Christakou, Vrettos and Stafylopatis (2007) which recommends movies to users by using a hybrid approach. They combined the content based filtering and collaborative based filtering techniques to construct a system which calculates accurate movie recommendations. In their study, content filtering is based on trained neural networks which depends on user preferences and they use Boolean and fuzzy aggregation operators to combine filtered results. They claim their system makes high accuracy predictions on MovieLens data. The study of Chen and Pu (2007) proposes a novel critiquing-based recommender system which integrates the user self-motivated critiquing facility to resolve the drawbacks of the system-proposed critiques. They claim the results of the study shows their proposed system enables users to achieve higher level of accuracy while consuming less cognitive effort.

The study by Burke (2002) is about restaurant recommendation by using a novel hybrid recommender system (EntreeC) which combines collaborative filtering and knowledge-based recommendation and shows that the knowledge-based part of the system enhances the effectiveness of collaborative filtering.

Also, the study conducted by De Campos, Fernández-Luna, Huete and Rueda-Morales (2010) is an example of hybrid recommendation system utilising Bayesian networks which is used widely and applied to problems with a high level of uncertainty. The study proposes a new Bayesian network model to deal with the problem of hybrid recommendation by content-based and collaborative filtering. They used Movielens and IMDB data sets to show the effectiveness of their model. The study by Taghipour and Kardan (2008) proposes a hybrid web recommender system that combines the conceptual relationship among web resources and semantic knowledge about the user behaviour. They tested their method with different settings and show how they improve the quality of web recommendations with their framework.

Huang, Chung, Ong and Chen (2002) used a graph-based hybrid recommender system that combines the content-based and collaborative filtering. They used hopfield net algorithm to reveal high degree book-book, book-user and user-user associations. They claim their evaluation results show that the system gains improvements according to results of precision and recall metrics.

Selecting a movie often requires users to perform numerous operations when faced with vast resources from online movie platforms. Wei, Xiao, Zheng, and Chen (2014) proposed a hybrid movie recommendation approach using tags and ratings. They built their model through the following processes. First, they constructed social movie networks and a preference-topic model. Then, they extracted, normalized, and reconditioned the social tags according to user preference based on social content annotation. Finally, they enhanced the recommendation model by using supplementary information based on user historical ratings.

2.2 Sentiment analysis

Sentiment analysis otherwise also known as opinion mining is the processing of textual information to quantitatively examine and determine the context of subjective information. Sentiment analysis is mostly used to understand the opinions and emotions of customers towards items, brands and services.

There are many researches about recommender systems that use sentiment analysis approach to filter items such as the study of Gurini, Gasparetti, Micarelli and Sansonetti (2013). They propose a user recommendation technique based on novel weighting function called sentiment volume-objectivity (SVO) function which takes both user sentiment and interests into consideration. They claim this allows them to build more user profiles to work within the recommendation process than content-based approaches and test results are better than some of the state-of-the-art user recommender systems. Another study about sentiment analysis is conducted by Singh and Mukherjee and Mehta (2011) which proposes an alternate approach to a hybrid recommender system that combines collaborative filtering with sentiment classifier in the recommendation process. They used this idea on movie review domain by using collaborative filtering for first level filtering and sentiment classifier for second level filtering and they claim result sets are more accurate and focused set. Another study is that of Li, Cui, Shen and Ma (2016) which proposes a novel model using social networks and mines user preference information expressed in microblogs using sentiment analysis by evaluating the similarity between online movies and TV episodes. They claim that it is the first approach that can solve the "cold-start" problem in movie and TV recommendation system.

Sentiment analysis is essentially the labelling of an expression within a text according to whether it provides a "positive" or "negative" attitude towards the subject in question (Nasukawa & Yi, 2003).

Sentiment analysis has also been an approach utilized within recommender systems. The basic user rating approach to collaborative filtering algorithms has been taken to a new level by incorporating textual information such as user reviews for preferential information. An example is the study of Leung, Chan and Chung (2006), who use sentiment analysis within collaborative filtering algorithms for recommender systems. Levi, Mokryn, Diot & Taft (2012) also make use of sentiment analysis for their hotel recommender system as well as Yang, Zhang, Yu &Wang (2013) for their personalised location recommender system.

The study by Wu, Tan, Zhai, Zhang, Duan and Cheng (2009) is about ranking the items according to sentiment analysis. They proposed an approach, SentiRank which integrates the sentiment orientations of the documents into the graph-ranking algorithm for cross-domain sentiment classification. They claim that test results show dramatic increase in performance of cross-domain sentiment classification. Another study proposes an experimental work on a new kind of domain specific aspect-level sentiment analysis of movie reviews (Singh, Piryani, Uddin & Waila, 2013). They have altered the aspect-oriented schema that analyses the textual reviews of a movie and assigns a sentiment label on each aspect and the scores on each aspect are aggregated and a net sentiment profile of the movie is generated for all parameters. They also used SentiWordNet scheme to compute document-level sentiment for each movie and compared results with the results obtained from Alchemy API. They claim that their results are more accurate than a single document-level sentiment analysis.

Wang and Wang (2015) built an opinion-enhanced user preference model with the idea of the higher the similarity between user opinions the more consistent preference between users are. Then they compare their experiment results and they claim their proposed algorithm outperforms the baseline methods. The other study about sentiment analysis is the one conducted by Leung, Chan and Chung (2006) which describes a rating inference approach to combining collaborative filtering with textual user reviews. The main idea of their approach is to detect user preferences which are expressed in textual reviews and map those preferences onto some rating scales so that it can be used by existing collaborative filtering algorithms.

Furthermore, the study by Pang and Lee (2008) is about opinion-oriented information-seeking systems that focus on methods that seek to address the new challenges raised by sentiment-aware applications. They use material on summarization of evaluative text and broader issues regarding manipulation, privacy and economic impact that gives rises to the improvement of opinion-oriented services. The study proposes a model that improves location recommendation by improving both user location and recommendation algorithm using sentiment analysis Yang et al. (2013). They first propose a hybrid user location model by using the user check-ins and text-based tips which are processed using sentiment analysis technique, and then they develop a location based social matrix factorization algorithm that considers both user influence and venue similarity influence. According to their experiment their proposed model surpass the up-to-date algorithms. The study by Pappas and Popescu-Belis (2013) is about sentiment analysis of user comments which do not have explicit rating labels. They propose a sentiment-aware nearest neighbor model (SANN) for multimedia recommendations over TED talks which makes use of user comments. They claim that it outperforms the several competitive baselines.

In another study, Ertugrul, Onal and Acarturk (2017) use Turkish tweets to analyze the effect of confident rates in sentiment analysis. They extract lexical properties, sentiment rates and emoticons. Classification and word embedding regression is used. They claim their study improves sentiment classification accuracy.

Opinion Mining is playing a major role to summarize customer reviews and make it easy for online customers to determine whether to purchase the products or not. Khan and Jeong (2016) proposed a supervised lazy learning model utilizing syntactic rules for the product features and opinion words extraction in subjective review sentences. Their lazy learning algorithm, i.e. K-NN with k=3 is used for the review sentences' classification into two classes: subjective, objective. The experiment shows that the proposed method can improve the performance of existing work in terms of average precision, recall and f-score for the extraction of opinion sentences and product features.

Researchers improve suggestion algorithms utilized in social networks by considering subjective parts of the suggested items, such as loyalty and cost, the affecting factors between social network clients, the social network client behavior with respect to their shopping in different item sections and the semantic sections of the products to be suggested (Margaris, Vassilakis, & Georgiadis, 2016).

2.3 Targeted advertising

Targeted advertising is a form of online advertising that uses sophisticated methods to target the most receptive audiences with certain traits, based on the product or person the advertiser is promoting. Targeted advertising uses recommendation systems as a tool to find the most relevant users.

In a study by Yang, Dia, Cheng and Lin (2006) a data mining framework that uses the social network concepts for the targeted advertising was proposed. Their approach finds out the subgroups from customers' social network which is extracted from customers' interaction data and they infer the probabilities of a selection of a product by a customer from transaction records. So, they generated a targeted advertising system and evaluated the improvements. Another study about targeted advertising is the one by Farahat and Bailey (2012) that measures the effectiveness of targeted advertising systems. They use several large-scale online advertising campaigns to test the effectiveness of the targeted advertising on clickthrough rates and brand-related searches. They found that the treatment effect on the targeted group is about twice as large for brand-related searches. Chickering and Heckerman (2003) used targeted advertising with inventory management. They propose a delivery system that maximizes click-through rate for given inventory-management constraints in the form of advertisement quotas. Their system utilizes predictive segments along with a linear program to provide the constrained optimization and uses a real web site (msn.com) to show the effectiveness of the system.

Another study about targeted advertising is the one that explains difficulties in generating targeted advertising in social media and monetizing activity (Mitra & Baid, 2009). They propose a model based on keyword clustering to generate targeted advertising. Their model works well on both web and social networks according to study. The study conducted by Moraga-González and Galeotti (2004) investigates a move game of targeted advertising and pricing in a market with many customer segments. They explore the ramifications of market segmentation on firm competitiveness.

The study by Kardan and Hooman (2013) proposes a framework which eases targeted advertisements in social network platforms by using social network information, previous advertisement and their status to have more precise information for recommender systems. They use a recommender system as a tool for target user selection by their interest and preferences. Their goal is to show the most effective advertisement in sidebar. Another study in targeted advertising is the one that was put forward by Bhatia and Hasija (2016) which proposes a model which combines social and spatial data of users to supply targeted advertisement. They obtain social data from user’s Facebook profile and location data is obtained with help of beacons. Another study is by Bimpikis, Ozdaglar and Yildiz (2016) which investigates a game-theoretic model of match between firms that could target their marketing budgets to customers embedded in a social network. They claim they supply a clear characterization of the best targeted advertising strategies and find out their dependence on the social network structure.

2.4 Network generation

The use of implicit networks in recommender systems has been studied in current literature such as the study of Xiao and Zhaoguan (2017) which is one of the first ones to create implicit network communities for recommender systems by examining user ratings. The study of Zhang et al. (2017) also examines user ratings but enhances the process by calculating the credibility of these ratings based on user rating frequency and deviation from normal behavior.

The novel study of Zhang, et al. (2016) is one of the first ones to incorporate implicit networks in online advertising. They propose a target selection framework online advertising by utilising social media user activities. They use implicit weighted brand-brand networks to extract relations between users and brands. They analyse community structures and network properties and then propose a framework to find target audiences. They developed an extended community detection algorithm in order to group closely related brands that have a specific brand in common. They also developed a global ranking algorithm to calculate brand scores to find out the most related brands. They then make use of sentiment analysis to uncover the users within their selected brands. Finally, they design a novel evaluation technique to test the effectiveness of targeting framework as well as contribute an important improvement in the identification of target audiences for specific brands.

There are studies such as that of Alotaibi and Vassileva (2016) that combines both explicit and implicit networks with the purpose of maximising user coverage while minimising recommendation accuracy loss. Alotaibi and Vassileva (2016) found that combining explicit and implicit networks increases both accuracy and recommendation coverage.

Another study about implicit network is Yang, Tang, Dai, Yang, and Jiang (2013) that proposes a temporal analysis technique to identify implicit relationships that supplement the explicit relationships identified through the social media interaction functions. Lipczak, Sigurbjornsson, and Jaimes (2012) study a large sample of Flickr user actions and compare tags across different explicit and implicit network relations, particularly they compare tag similarities in explicit networks (based on contact, friend, and family links), and implicit networks (created by actions such as comments and selecting favorite photos). They perform an in-depth analysis of these five types of links specifically focusing on tagging,and compare different tag similarity metrics.

In the study conducted by Ma, King and Lyu, (2011) a novel framework was proposed which naturally the users' tastes and their close friends' favors together based on implicit relations. They mention that their approach can be applied to very large datasets since it scales linearly with the number of observations according to complexity analysis. Then they evaluate the effectiveness of their proposed system.

2.5 PageRank

PageRank (PR) is an algorithm used by Google Search to rank web pages in their search engine results. The main idea behind this algorithm is to figure out the importance of a web page by considering the pages hyperlinking to it. According to Google, PageRank works by counting the number and quality of links to a page to determine a rough estimate of how important the website is. The underlying assumption is that more important websites are likely to receive more links from other websites.

There are thousands of applications of PageRank algorithm in the literature. In a study in the field of Web information bibliometrics, retrieval, sociometry and econometrics, the PageRank method is reviewed and linked to some updated previous methods (Franceschet, 2010). Another study about PageRank was conducted by Şora (2015) who proposes a PageRank based recommendation tool to classify the most important parts of a system. His approach depends on static dependencies structure of the system as a graph and ranking algorithm. They identify different dependency types and find the optimal way of building the system graph. The study of Wang, Liu and Zhao (2012) is about PageRank based group recommendation that proposes an algorithm to calculate importance of a member in the group and improve the aggregate function of individual preferences. Their algorithm takes the initiative to find the user needs. Their experiment shows the effectiveness of their algorithm which is claimed to improve the prediction accuracy of the group recommendation.

Another study which was conducted by Jiang and Wang (2010) is about PageRank based collaborative filtering that proposes a model which merges user rank as weight of a user based on PageRank into item similarities computing. They proposed three different PageRank based rank calculation methods. Then they evaluated the experimental results of their proposed models. The study by Kandiah and Shepelyansky (2012) proposes PageRank based opinion formation model and examines its rich properties on real directed networks of the universities of Cambridge and Oxford, LiveJournal and Twitter. In their study PageRank is used in weighting the opinion formation of linked electors. According to their study, LiveJournal and Twitter networks have a stronger tendency to a totalitarian opinion than universities networks.

2.6 Community detection

A community is defined as a subset of nodes within the graph such that there are more connections between the nodes than rest of the network. The detection of the community structure in a network is generally intended as a method for mapping the network into a tree that the leaves are the nodes and the branches join nodes or groups of nodes.

There are several community detection algorithms. Generally, they are classified into two categories; Overlapping and Non-overlapping. In community detection algorithms, the network divides into groups of nodes with dense connections internally and sparser connections between groups. In a non-overlapping community, every node is placed in only one community, whereas in an overlapping community, a node can be found in more than one community. Some of the community detection algorithms are Minimum-cut method, Hierarchical clustering, Girvan–Newman algorithm, Modularity maximization, Statistical inference and Clique-based methods. Of these algorithms, only Clique-based is an overlapping community detection algorithm, all others are non-overlapping.

In their study, De Meo, Ferrara, Fiumara and Provetti (2011) propose a novel strategy to discover the community structure of large networks based on Louvain method for network modularity optimization. They use a novel measure of edge centrality based on k-paths to implement their proposed strategy. Their algorithm computes the pairwise proximity between nodes of the network after the centrality ranking is calculated and then discovers the community structure adopting to their strategy inspired by the Louvain method. Experimental results outperform the other techniques and slightly improves the results of the original Louvain method. In another study, a novel community based social recommender system that uses implicit relations in social data to provide personalized recommendations based on communities generated from users' social interaction history with the items in the target domain was proposed (Fatemi & Tokarchuk, 2013). They evaluated their approach by using the Internet Movie Database (IMDb). They use social network graph of the movies based on common reviewers to model the generic network of interest and then communities are discovered. They also evaluate the results and their approach increases the accuracy according to their claim.

Another study by Lalwani, Somayajulu and Krishna (2015) proposes a social recommendation system by using collaborative filtering and community detection approaches. They use community detection for extracting the friendship relations between the users by analyzing user-user network and user-item based collaborative filtering for rating the prediction. They develop their approach using map-reduce framework. Then they compare the results with traditional collaborative filtering-based recommendation system. Parimi and Caragea (2014) propose an approach which combines community detection and Adsorption algorithm which is a neighborhood-based recommendation system for recommending items using implicit user preferences. They claim their approach delivers good results. The study by Ben Yahia, Bellamine Ben Saoud and Ben Ghezala (2014) propose an approach of collaboration recommendation that depends on a community detection technique to find potential collaborators that help in problem solving. Their contribution is combining two community detection approaches into one approach and then they use computational optimization technique to maximize this combined quality. In their work, Abdrabbah, Ayachi and Amor (2014) propose a novel architecture called Dynamic Community based collaborative filtering that combines the dynamic community detection algorithms and recommendations techniques. They evaluate the efficiency of the approach by comparing it with recommendation system based on static community detection and item-based collaborative filtering. They claim that experimental results show dramatic improvements of recommendation accuracy.

2.7 Node2vec

Node2vec is a kind of framework that provides representation learning. It learns consistent element representations for nodes. Many machine learning functions utilizes this learned features (Grover & Leskovec, 2016).

It is useful to learn representations from graphs for the machine learning applications. Not only these representation provides a better predictive accuracy but also it reduces the engineering effort.

The node2vec framework optimizes a neighborhood protecting goal to learn low-dimensional representations for nodes. The goal is flexible and the algorithm has different neighborhoods by using random walk. In particular, it gives an approach to adjust the investigation-misuse tradeoff that prompts representations complying with a range of equivalences from homophile to constructural identicalness.

Another study shows how the node2vec algorithm could be used to provide item suggestions by a method that based on graph embeddings Palumbo et al. (2018). They use node2vec on a learning graph produced from MovieLens and Dbpedia datafiles and utilize node similarities to produce item suggestions. According to results, node2vec simply surpass many collaborative filtering recommendations.

A spectral clustering-based collaborative filtering framework based on node2vec algorithm to resolve the challenges of sparsity and efficiency which is encountered by many recommendation systems was proposed in a study conducted by Chen et al. (2017). They represent interaction data as a bipartite network. They cluster users and items separately by using improved spectral clustering method and then they generate recommendations over the most frequent pairs of user-item clusters. Then they show the effectiveness of their framework.

The study by Liao, He, Zhang and Chua (2018) proposes a generic attributed social network embeddings framework that learns representations for social actors by protecting both the structural proximity and attribute proximity. Structural proximity shows the global network structure and the attribute proximity is about homophily effect according to their study. They finally compare the effectiveness of their proposed algorithm with link prediction algorithms.

In the survey implemented by Zhang, Yin, Zhu, and Zhang (2018) they perform a comprehensive review of the current literature on network representation learning in the machine learning and data mining field and they propose new classifications to categorize and summarize the state-of-the-art network representation learning techniques. They also perform empirical studies to compare the performance of representative algorithms and analyze their complexity.

CHAPTER 3

METHODOLOGY

In this thesis, three different recommendation system frameworks were developed to find the potential customers for online restaurants. Ranks framework utilizes network analytics to find the rank for each user, whereas the Senti framework utilizes sentiment analysis of the comments made by each user to find their corresponding rankings. The last framework uses a completely different approach from the first two and utilizes network embeddings to find potential customers directly from the similarities based on these embeddings.

Firstly, the dataset used within the thesis was collected from the Zomato website, which contains useful information regarding the proposed recommendation framework. After the data collection phase, the data was cleaned and formatted so that it can easily be consumed by the frameworks. Then, the dataset was divided into three parts which contain the data to be used for first training the models, training data, and then optimize framework parameters, validation data, and then test and evaluate the frameworks using the test data.

3.1 Data preparation

Zomato is an online platform catering to the needs of “foodies” by supplying them with information regarding restaurants and coffee shops all around the world (Figure 1).

Figure 1 Datasource
Source: https://www.zomato.com/istanbul, June 2019

Zomato is a critical cog in the restaurant ecosystem as it provides a review and rating system for restaurants as well as a local advertising platform used extensively by restaurants to get established with a wider audience (Raman, 2018). This is the basis for using the Zomato platform for the thesis as it is one of the best platforms for implementing a recommendation system aimed at finding target customers for restaurants.

All the relevant data about restaurants, users, reviews, photos and foods for the restaurants in Istanbul, Ankara and Izmir provinces of Turkey within the period of 01/01/2016 – 01/10/2018 where accessed via the Zomato API. These provinces contains most of the restaurant information of Zomato Turkey. Below, the attributes collected from the site together with an approximate number of records for each attribute is given:

i. Users (100 K)

ii. Restaurants (30 K)

iii. Reviews (400 K)

iv. Replies (27 K)

Among these attributes, user review is the most important attribute used in the frameworks because reviews provide the interaction between users and restaurants. The reviews were utilized to generate an interaction network and the ones derived from that. For example, if a user leaves a comment to a restaurant this creates a relation between the user and the restaurant. Also, if two users leave a review to the same restaurant, then this interaction is an implicit relation between these two users because this implies that they have similar tastes. Likewise, if a user reviews two different restaurants, then it may conclude that these two restaurants are related to each other. Accordingly using reviews data, three different networks were generated, two of them being implicit user and implicit restaurant networks, and one of them being an explicit user restaurant network.

Besides the existence of the reviews, the content of the reviews is also very important criteria to understand user positivity or negativity to a particular restaurant which may be an important feature in identifying the potential users.

After collecting the data, it was pre-processed by removing unreadable comments and non-Turkish comments. Also, all the review data before 2016 was removed from the dataset. The dataset was divided into three periods; training, validation and testing. Each period was created for different purposes. Training data was used for generating the recommendation model, whereas the validation data was used for optimizing the hyperparameters of the model. Testing data was used for evaluating the success of the models. As shown in Figure 2 training period were selected as 2016/01/01 – 2017/12/31, whereas validation and testing periods were chosen as 2018/01/01-2018/04/30 and 2018/05/01-2018/11/01, respectively.

Figure 2 Period selections

Having divided the reviews data into periods, there are thousands of user reviews in each period. A user can review the same or different restaurants in different periods. When a user reviews a restaurant for the first time in a period, this is the new user for that period. Figure 3 shows all reviews made by all users. Blue circles show the reviews given by the new users. Since we have thousands of restaurants and thousands of users, there are always thousands of new users.

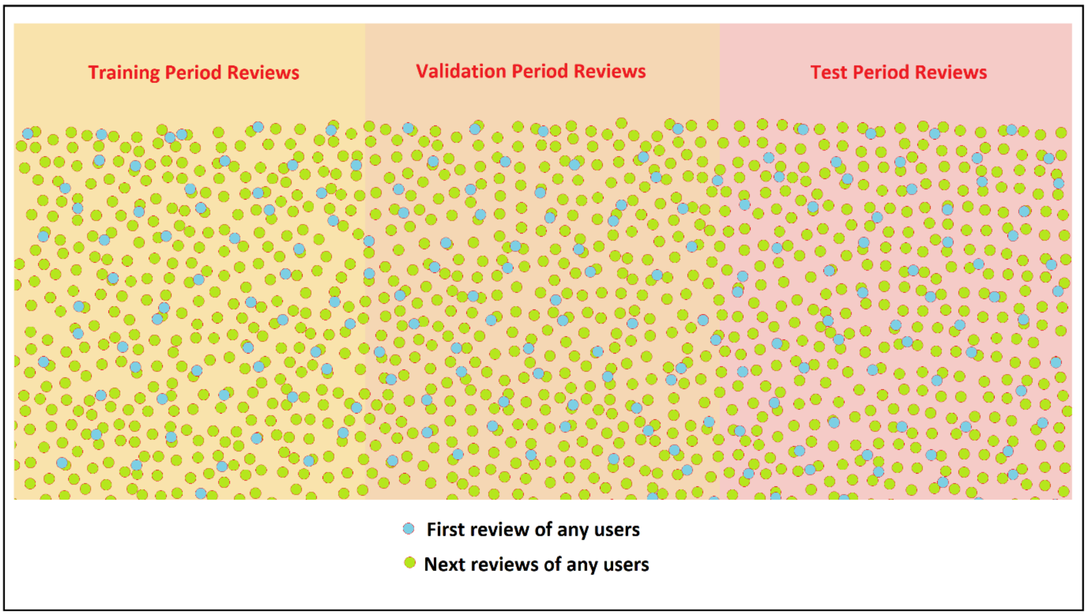

Figure 3 All reviews in all periods

The focal restaurant is the target restaurant for which the potential target users are to be identified who are likely to visit and review that restaurant. 300 different focal restaurants were used to create recommendation models and evaluate them in the frameworks. Focal restaurants were selected from the list of all restaurants by considering restaurants' review count. So, the top 300 restaurants were selected based on the review count in the training data as the focal restaurants. All frameworks were run one by one for each focal restaurant and each focal restaurant's success rates were evaluated.

For a single focal restaurant the goal was to predict users who will review that restaurant in the test period. These users were actual target users for a focal restaurant. Even if the goal was to try to find all actual target users it may not be possible to find all of them because some of the users were not available in the training period so the goal users could be only a part of the actual users. Figure 4 gives representations of all users who commented on a single focal restaurant in both the training period and test period. In Figure 4, red users show the Focal Users which were all users who commented on the focal restaurant in the training period and they already were in the database. Figure 4 also shows the actual target users which were all users who commented on the focal restaurant in the test period and distribution of the actual target users with different colors. Actual target users consist of three types of users; yellows are the goal target users; blues are new users and reds are re-reviewed users. New users mean they were not available in the training period, they didn't have any review for any restaurant in the training period and they began reviewing for the first time in the test period so new users couldn't be goal target users to predict in advance. Re-reviewed users were the users who commented on the same focal restaurant in both the training period and the test period so they also couldn't be the goal target users since they already reviewed the focal restaurant. Thus, only yellow points in Figure 4 could be goal target users to predict in the training period. All of success rate calculations of frameworks were based on goal target users.

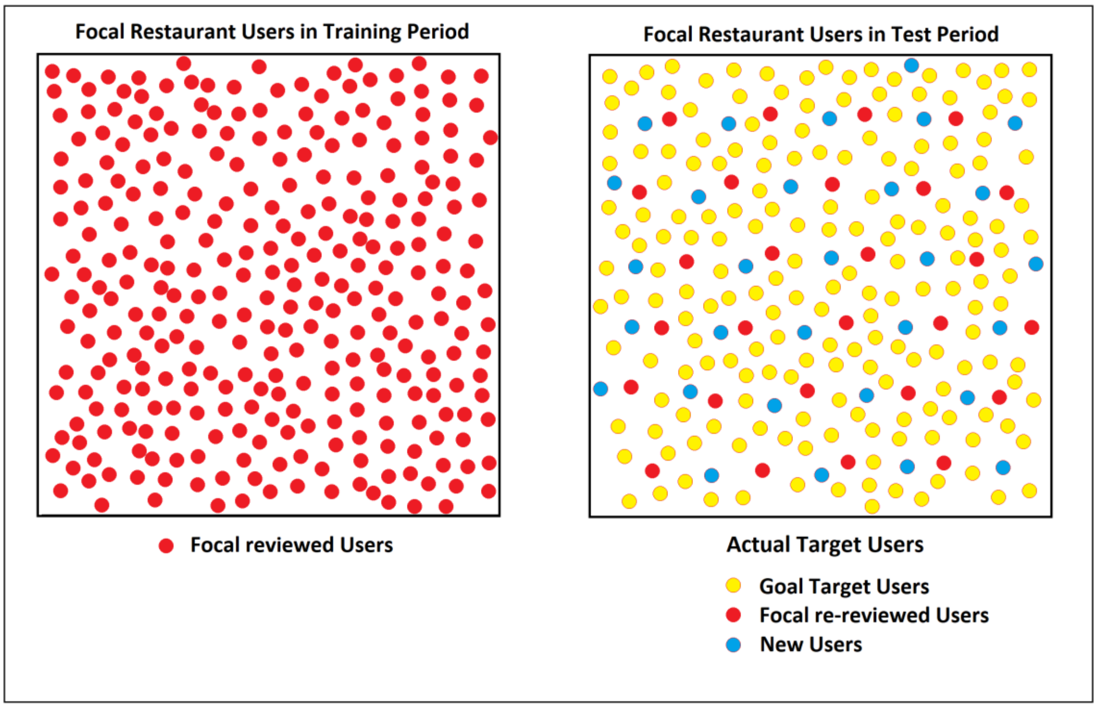

Figure 4 Focal restaurant user and actual users

Table 1 shows focal restaurants and their properties which were taken from the test and training data. It shows Focal users count, Goal user count, Actual user count, new user count and re-reviewed user count for the corresponding focal restaurant with real data.

For example, for the focal restaurant "Mendels", there were 298 reviews in the training period. 87 users commented on the restaurant in the test period and 54 of them were new users.There were no re-reviewed users for the focal restaurant so remaining 33 users were goal users which were available both training and test period. This thesis will try to predict these 33 goal users.

Table 1. Some Focal Restaurants and Their Properties

Focal Restaurant Name	Training User Count	Actual Target User Count	Goal Target User Count	New User Count	Re-reviewed User Count
Valonia	320	84	32	52	0
Mendels Çesme	298	87	33	54	0
Bazlama Kahvalti	279	94	41	53	0
Dürümcü Emmi	268	63	18	45	0
Biber Burger	249	91	33	58	0
Virginia Angus	226	61	19	42	0
Tatar Salim Döner Lokantasi	212	74	42	32	0
Karadeniz Döner	191	40	21	19	0
Tarihi Viktor Levi Sarap Evi	189	50	21	29	0
Baldir	175	67	21	46	0
Mini Eatery	173	37	16	21	0
Varuna Gezgin	153	43	21	22	0
Limonlu Bahçe	151	54	14	40	0
Tarihi Vefa Bozacisi	138	16	6	10	0
Ozzies Kokoreç	119	46	15	31	0

In the thesis, to predict goal target users three different frameworks are developed and all of them try to find as much as goal user as a future prediction. These frameworks are Ranks Framework, Senti Framework and Embeddings Framework.

First framework is Ranks Framework which computes the target users by combining many rank algorithms. Ranks Framework uses training data to calculate aggregated target user list which contains as many target user list as hyperparameters count for each focal restaurant and uses validation data to select the best hyperparameters by considering the success of each alternative target list. Finally, it uses the goal users of test data to evaluate the success of the framework.

Second framework is Senti Framework which computes target users by using sentiment analysis. Senti Framework uses training data to calculate target user list and doesn't use validation data. It compares the target users with goal users of test data and evaluates the success of the framework.

Third framework is Embeddings framework that computes the target users by their embedding vectors using training data. Embeddings are calculated with different hyperparameters and the best parameters are selected with validation data. This framework uses the direct relation between users and restaurants to get embeddings. Finally, it evaluates the success result of the framework with test data.

3.2 Network analysis

Networks are a relation of entities and in this case the entities are users and restaurants. The best relations need to be determined to be able to predict future relations of other entities. There are direct relations between entities, such as if a user bookmarks or follows a restaurant then we can clearly conclude that this restaurant and user are related to each other. Direct relations signify the explicit networks between entities. However, only using explicit networks is not enough to predict future relations. Indirect relations also come into play for better prediction. For example, let's have two users, A and B. User A has bookmarked restaurant R1 and R2, user B has also bookmarked restaurant R1. The resulting explicit network A→R1, R2 and B→R1 does not help us to predict which restaurant user B will bookmark in the future. We also need to use indirect relations of entities which signify implicit networks.

To use the above example, since user A and B bookmarked the same restaurant (R1) then we can say that they are related to each other (A→B). Likewise, we can conclude that user B is likely to bookmark restaurant R2 since user A has already bookmarked that restaurant. Finally, the more restaurants that user A and B have commonly bookmarked, the more powerful the relation they have, and the more precise future predictions can be made for these users.

3.2.1 Explicit networks

Explicit networks are formed by using direct relations between a user and a restaurant. In an explicit network, one node is a user and other one is a restaurant. For example, if a user comments, bookmarks, rates or likes a restaurant then we can use this interaction to make an explicit connection between the user and restaurant. As we can see in Figure 5, when a user gives a rating to a restaurant this relation creates an explicit network edge between user (node1) and restaurant (node2).

Figure 5 Explicit network

The weight of this edge is the rating given by user that is 3.5 in our example (Figure 6).

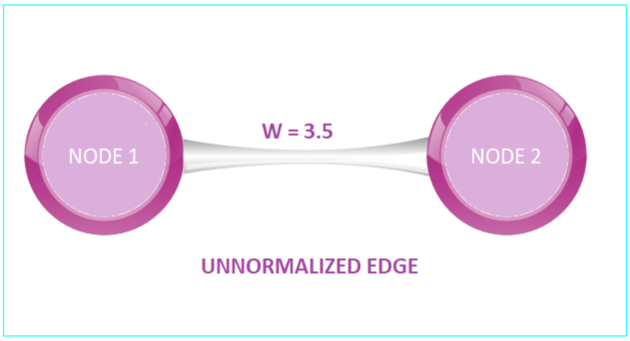

Figure 6 Unnormalized edge weight

In this way, using all user-restaurants interactions, a big explicit network of restaurants to users was formed. Then, the edge weights of this network were normalized by dividing all edge weights with maximum edge weights. There are 70,452 nodes and 176,682 edges in this network.

Explicit network was generated using training dataset and only Embeddings framework used this network.

3.2.2 Implicit user networks

In the implicit networks there is no direct relation between users or restaurants. In the implicit user network, nodes are users, edges are shared restaurants. If two users comment on the same restaurant then we can infer that there is a relation between two users as an implicit connection. In this connection, the edge weight is equal to the difference between the maximum rate value (5 in our case) and node rate differences. If there are many shared restaurants, then the total weight of the edge is sum of the all edge weights.

In the example given in Figure 7, we have two users (user 1 and user 2) who have two shared restaurants. Each user has given different ratings to the shared restaurants.

Figure 7 Implicit user network

To calculate these two users' total edge weight, we first calculate each edge weight and then sum up the all weights. Figure 8 shows the edge weight calculations of the network, the first edge weight is [5 – (5-4)] = 4 and second weight is [5 – (4-3)] = 4. So, the total unnormalized weight is equal to 8.

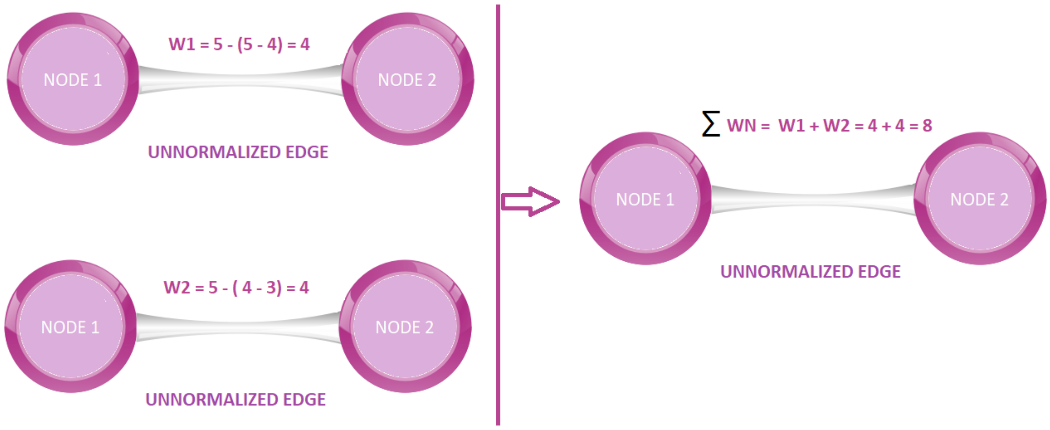

Figure 8 Implicit user-user network edge weights

Using all implicit user interactions like above, a large implicit user network was created. In this network there are 4,698,441 edges and 53,436 nodes. The Training dataset was used for this network.

3.2.3 Implicit restaurant networks

In the implicit restaurant network, nodes are restaurants, edges are shared users. If two restaurants are commented on by the same user, then we can infer that there is a relation between these two restaurants as implicit connection. In this connection, the edge weight is equal to the difference between maximum rate value (5 in our case) and node rate differences. If there are many shared users for the restaurant pair, then the total weight of the edge is the sum of all edge weights. In Figure 9 we have two restaurants (Rest A and Rest B) which have two shared users. Each restaurant is given different rating from users.

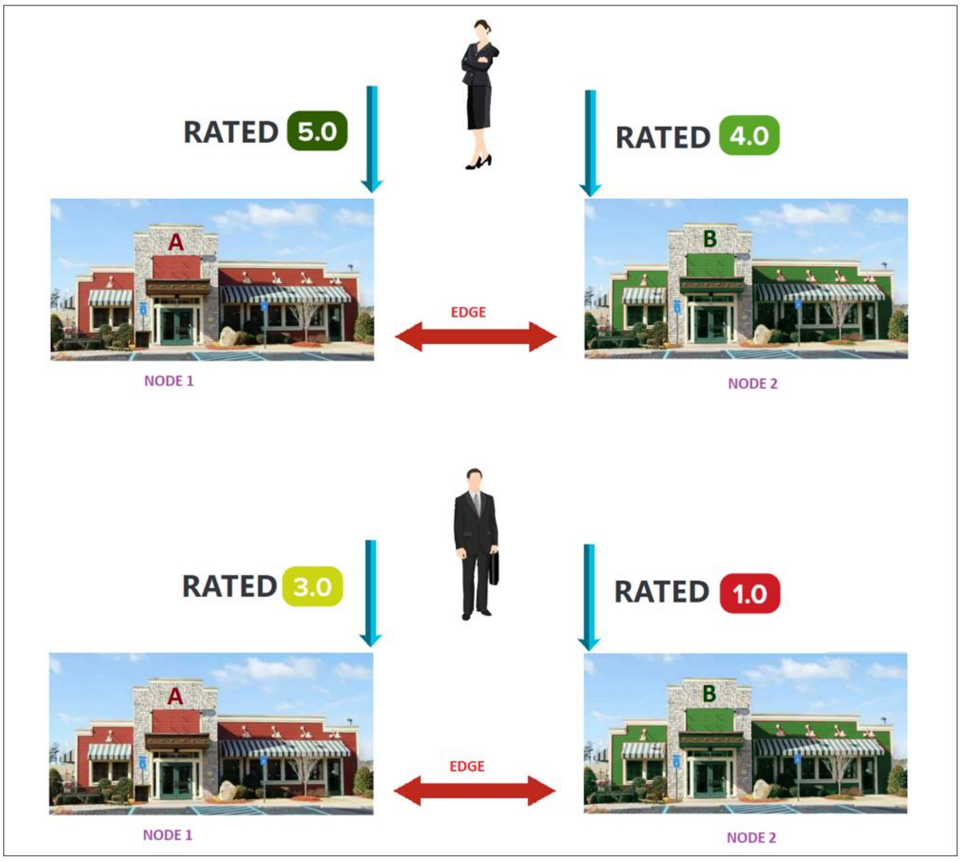

Figure 9 Implicit restaurant network

To calculate these two restaurants' total edge weight first every edge weight one by one was calculated then the all of them were summed. The first edge weight is [5 – (5-4)] = 4 and second weight is [5 – (3-1)] = 3. So, the total unnormalized weight is equal to seven for this Rest A and Rest B as it is shown in Figure 10.

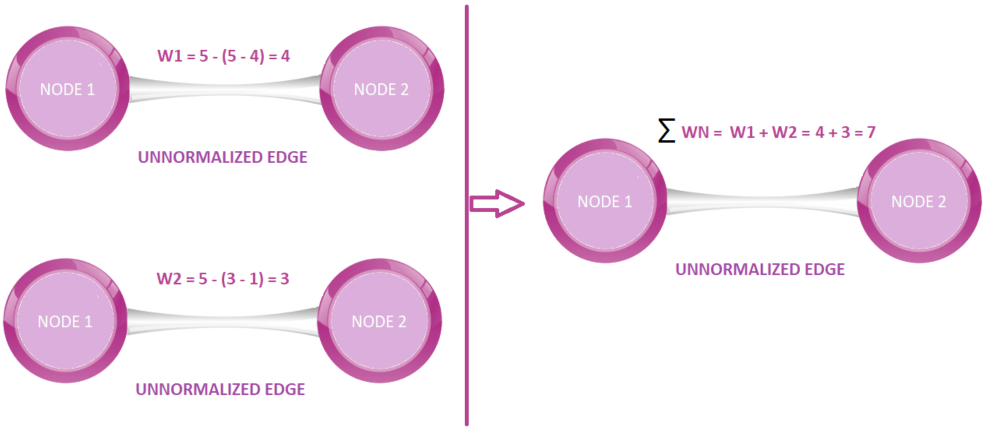

Figure 10 Implicit restaurant-restaurant network edge weights

Using all implicit user interactions like above, a large implicit restaurant network was created. In this network there are 3,191,203 edges and 14,627 nodes. Training dataset was used for this network.

3.2.4 Normalization and direction of networks

There were two different implicit networks which were user-user network and restaurant-restaurant network. All edge weights on this network were calculated when the networks were formed by user rates and shared object. Then all edge weights were normalized since edge weights can range from zero to hundreds. Edge normalization is simply done by dividing every edge weight by the maximum value of weights.

Also, directed networks for both implicit networks were generated. The direction of edges is dependent on edge weights and node weights

Figure 11 shows the normalization of the network for both implicit restaurant and user networks. Besides this, Figure 12 shows the direction of network steps with normalization, and this was done for both implicit networks.

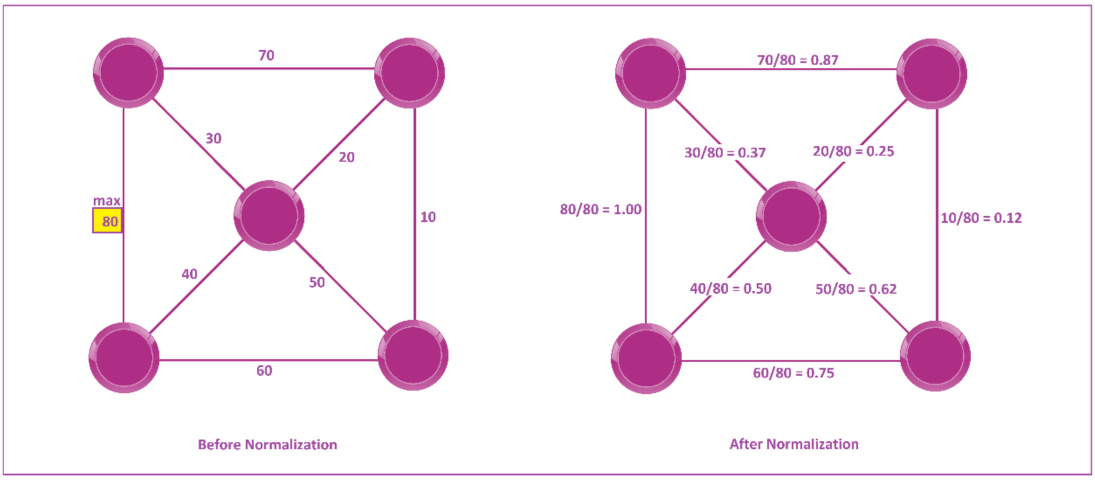

Figure 11 Normalization steps

Also, normalization of explicit network was done by dividing each edge weight with maximum of edge weights as shown in Figure 11.

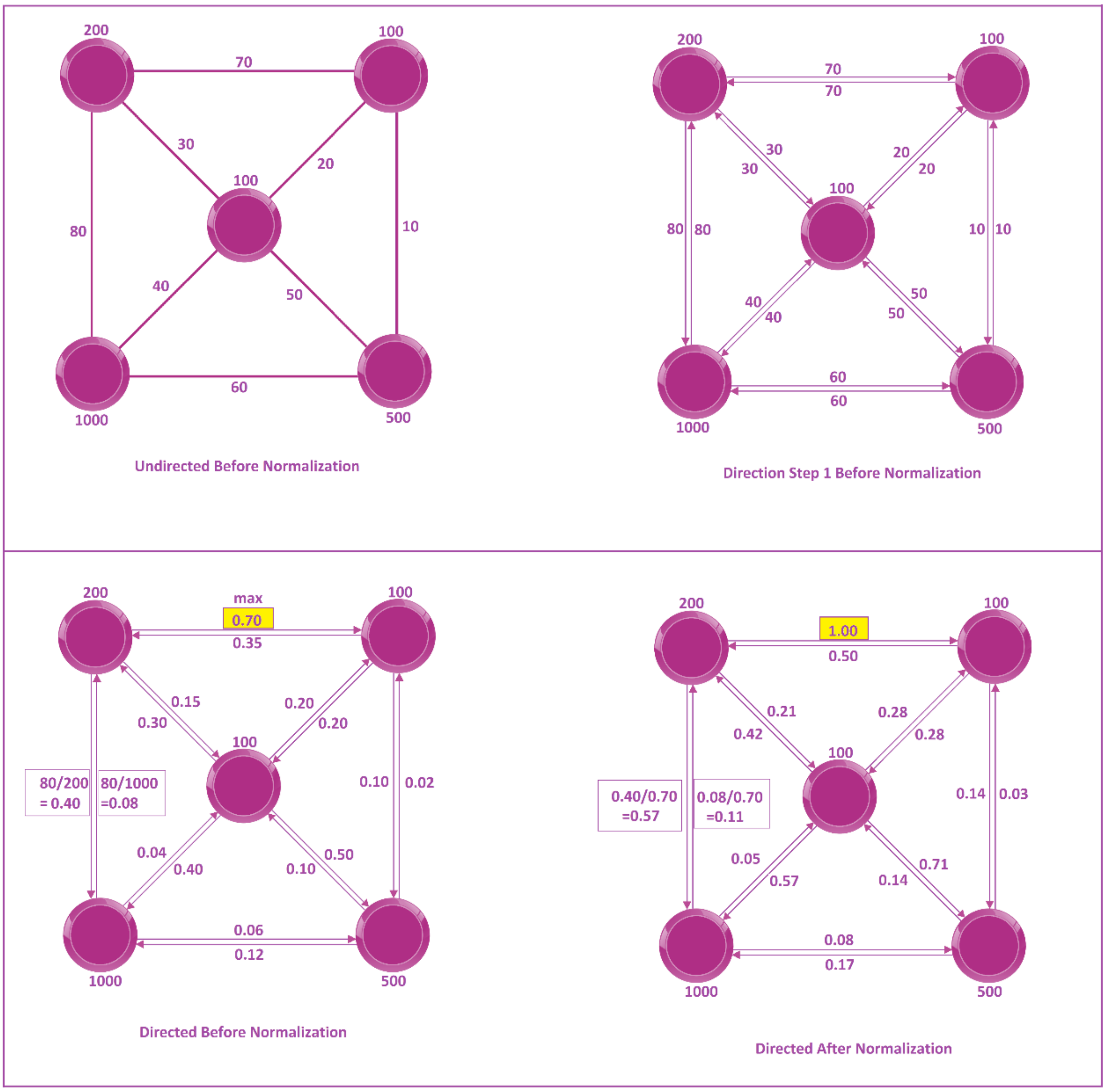

Figure 12 Direction with normalization steps

3.3 Frameworks

Table 2 shows the comparison of the frameworks in terms of used features in the thesis. Such as, Embeddings framework used explicit restaurant-user network in the whole process, all values and success results calculated with this network data. Also, Senti framework didn't need to use validation data since it didn't have any hyperparameters.

Table 2. Frameworks Feature Comparison

Feature	Ranks Framework	Senti Framework	Embeddings Framework
Validation Data	✓		✓
Implicit restaurant-restaurant network	✓	✓	
Implicit user-user network	✓	✓	
Explicit restaurant-user network			✓
Node2Vec			✓
PageRank	✓	✓	
Community Detection	✓	✓	

3.4 Ranks framework

There are many ways a user may interact with a restaurant such as liking, bookmarking or rating a restaurant or via leaving a comment. User comments were the primary data source for user restaurant interaction. In Ranks framework, both implicit restaurant and implicit user networks were used. PageRank values were calculated and community detection was performed for both of these networks. Using the extracted communities, a novel rank value for implicit user networks was computed which iscalled CommunityRank. It is a rank value that depends on focal user count in a community. CommunityRank was calculated for all users. Then PageRank and CommunityRank values were combined with two hyperparameters and calculated FinalRank values for the user. Hyperparameter optimization and final target user list were performed according to this FinalRank value. Accordingly, this framework is named Ranks framework.

Ranks Framework has three phases; Training, Validation and Test phases. Framework Outlines (Figure 13):

1. Network generation of weighted directed/undirected user and restaurant networks and normalization of the edge weights in these networks.
2. Community detection for both user and restaurant networks.
3. Global PageRank calculation for both user and restaurant networks.
4. Initial target user pool selection from all users who commented the focal related restaurant.
5. CommunityRank value calculation for all users in the initial target user pool.
6. FinalRank value calculated based on a combination of PageRank and CommunityRank values.
7. Selection of Aggregated Target users based on FinalRank values with all hyperparameters.
8. Determining the best parameter values that yield the most successful results according to validation period goal users.
9. Selection of the target users based on the chosen parameter values.
10. Evaluation of the final target/recommended user list considering the goal users found in the test data.

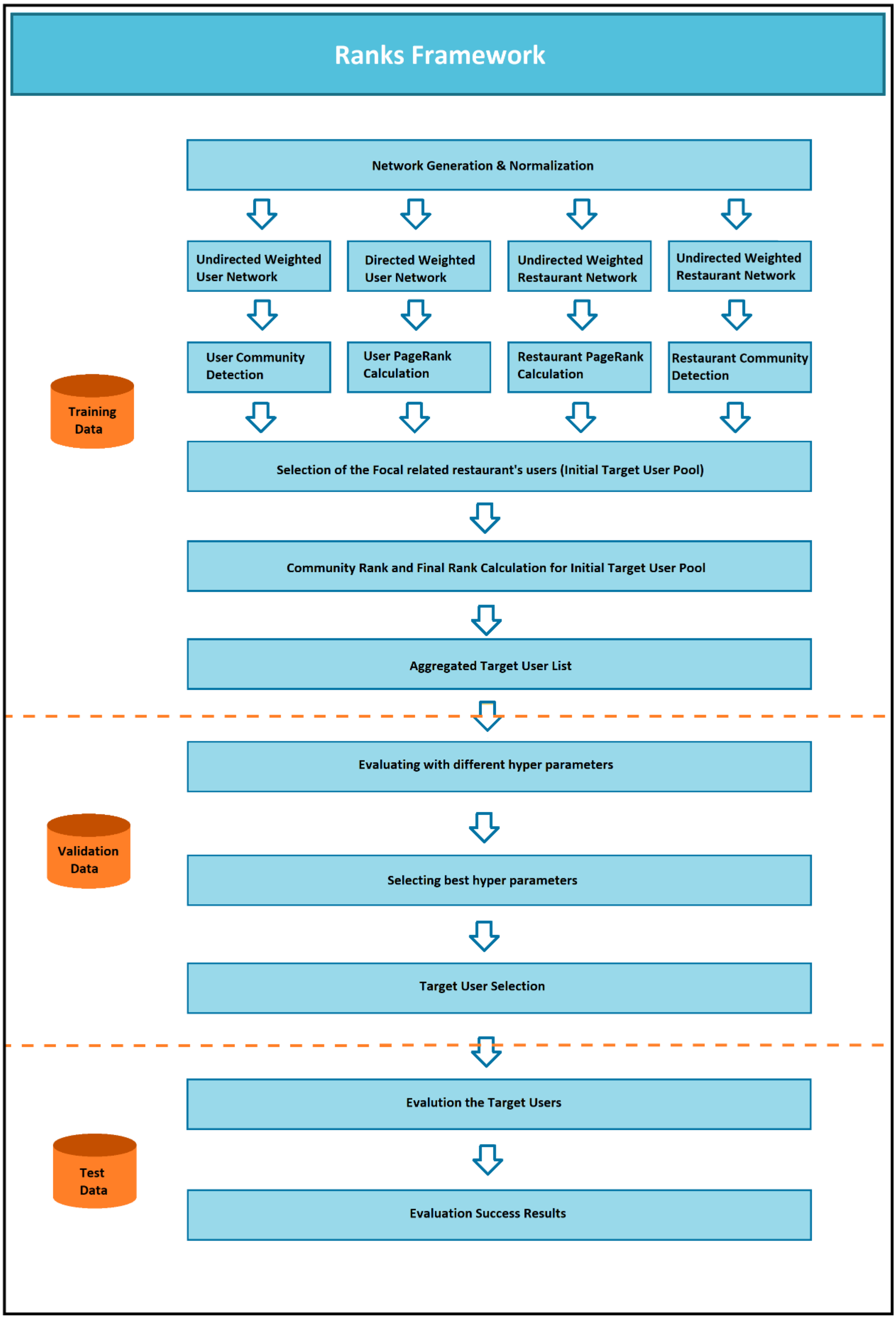

Figure 13 Ranks framework flowchart

3.4.1 Networks

In the Ranks framework, implicit user and implicit restaurant networks were used for user and restaurant ranking.

3.4.2 User and Restaurant pagerank

In the restaurant and user networks, every node needs to be weighted since every restaurant has a different influence on others. For example, In Figure 14 we have A, B and C restaurants and let the edge weights of the nodes be (A→B: 10), (A→C: 100), (B→C: 20). Here, both A and B have influence on restaurant C. How can we decide what would be the next target of restaurant C while restaurant A and B offer different users as the next target? The answer depends on the influence power of the relations. Because restaurant A and C have a stronger relation than B and C, we can conclude that restaurant A has more influence on Restaurant C and C is more likely to follow A. Likewise, the influential ranking of every node was calculated by using the PageRank algorithm. This algorithm was applied to the whole network data and the global rank values of each node (restaurants) were calculated.

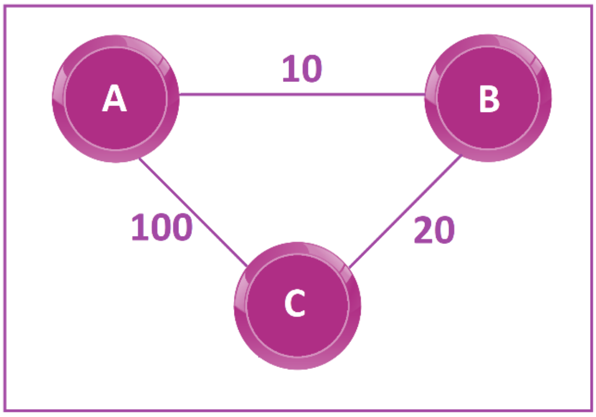

Figure 14 An example nodes for PageRank

For restaurants, the most related restaurants to the focal restaurant need to be determined. To do this, the community in which the focal restaurant resided was selected. Then, the global PageRank values of restaurants in the community were used to select the most related restaurants to the focal restaurant.

To find the PageRank values of the restaurants a program was developed. The code was implemented by using the Python NetworkX package library. To be able to calculate the PageRank values, the normalised directed weighted edges of the restaurant network previously generated was used as input. The default Alpha value which is a mandatory input parameter for calculation was specified as 0.9 which is a default value in the package. By using the alpha parameters and the directed network, the global PageRank algorithm was run for all restaurants.

3.4.3 User and restaurant community detection

There are thousands of users and restaurants and there are nodes with high node rank values, but they may never be related to each other. For example, let A be one of the most influential restaurants in one province, and B be one of the most influential restaurants in another. Since they are in different places, they may never be influential to each other. So, the networks need to be separated into clusters that are formed by nodes related to each other. These clusters are called “communities” and groups of related restaurants are found within these communities (Figure 15).

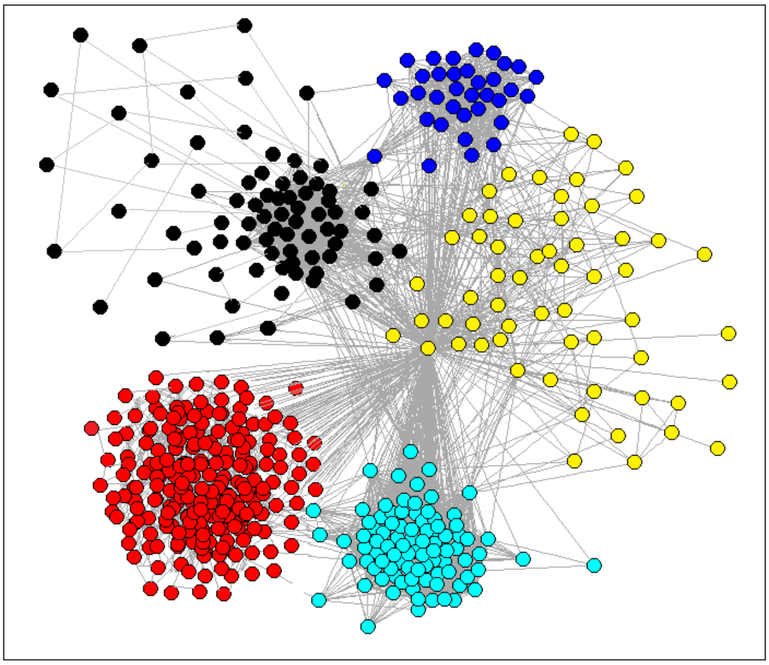

Figure 15 Community detection representation

The Louvain heuristics method that computes the partition of the graph nodes by maximizing the modularity was used. A program was developed to compute communities in Python language which takes undirected weighted networks as input and creates communities as output. This algorithm is a non-overlapping community detection algorithm.

Two different community detection algorithms, one for restaurants and one for users, were applied, the community detection algorithm was applied to the undirected weighted restaurant-restaurant network resulting in a total of 80 communities. This community detection is a non-overlapping community detection so any restaurant can only be found in one community. Every restaurant in the same community is related to each other. Restaurants related to the focal restaurant were determined by using this algorithm. The community detection algorithm was also applied to undirected weighted user-user network resulting in a total of 480 communities. This is also a non-overlapping community detection algorithm. The user community detection process was used to find the Focal Training User Count per community. Any community having a higher Focal Training User count were considered important to calculate the rank of users in that community.

3.4.4 Finding the initial target user pool

The goal was to predict target users for the focal restaurant. Implicit networks were generated and then many filtering and sorting operations were made on these networks to find target users. Before the final target user list, first the initial target user pool then the final target user list was determined from the users specified in this pool. Initial target user pool is the primary target user list which contains thousands of users filtered by focal related restaurants. The final target user list was selected from this pool by using FinalRank values of users. The initial target user list was used in both the Ranks Framework and Senti Framework and they will differ in selecting the final target user list from this pool.

Below finding of the initial target user pool is explained step by step. Firstly, it was started by selecting the focal restaurant from restaurant-restaurant network.

Figure 16 is a representation of the selected focal restaurant (red) on this implicit restaurant-restaurant network. It was selected randomly in the most commented restaurants in training data.

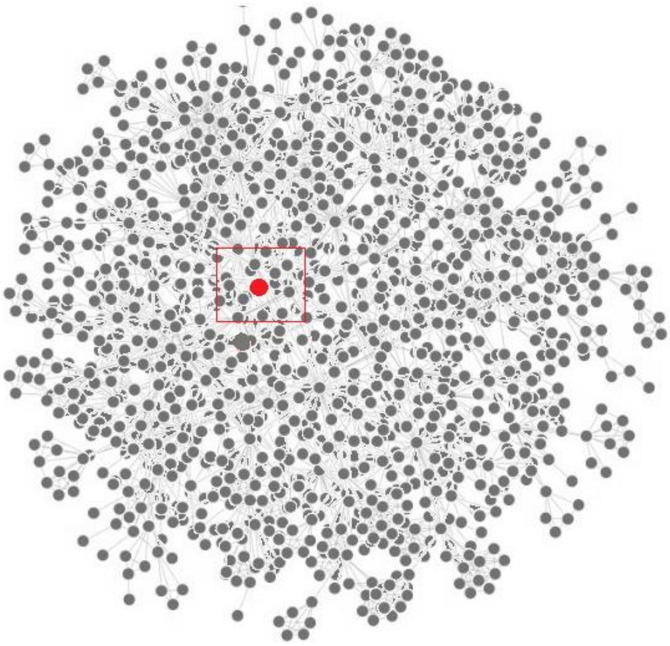

Figure 16 Implicit restaurant network with focal restaurant (red)

Then community detection was applied on this network and it divided the rest of the networks into communities. Some of the communities are shown in Figure 17. The community in which the focal restaurant is found is marked in red rectangular called the focal community.

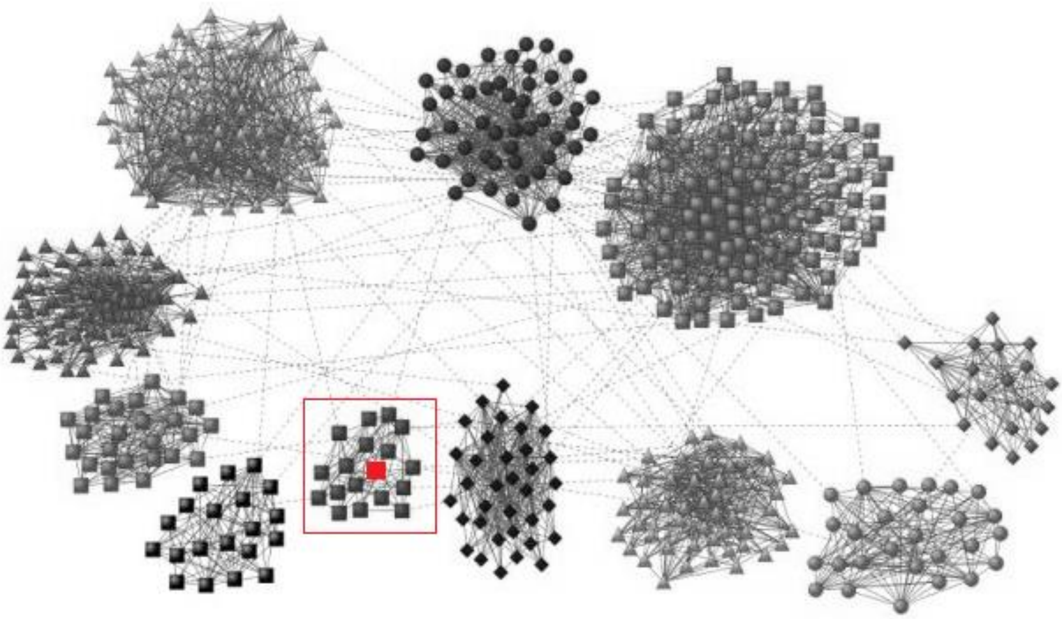

Figure 17 Community detection on implicit restaurant networks

Then, the top N most related restaurants to focal restaurant were selected by their PageRank value in the focal community and colored yellow as in Figure 18. These restaurants are Focal Related Restaurants. This focal related restaurant is important for the framework because their users were used to search final target users.

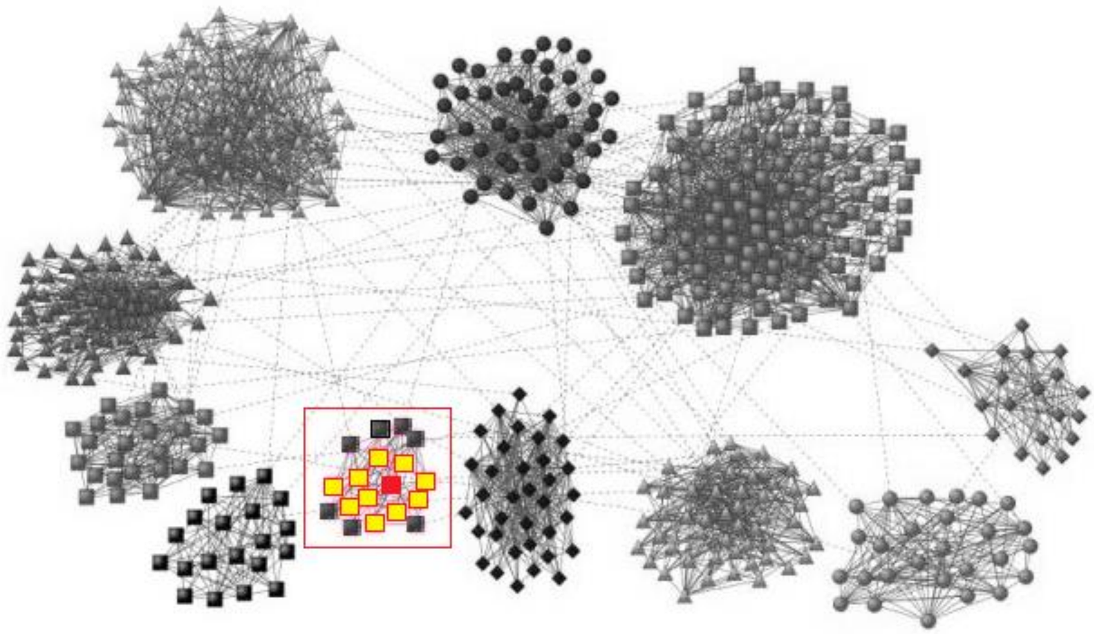

Figure 18 Focal related restaurants

All users who commented on the focal related restaurants were selected and these users formed the initial target user pool which was named focal related users. So, the final target users were selected from these focal related users. Since there were thousands of users in this initial target user pool, ranking algorithms were used to sort the users and select the most specific target users. Figure 19 shows the focal related user selection from the focal related restaurant. The red coloured users were the focal users who already commented on the focal restaurant, so they were not in the target user list. Yellow points were focal related users which is used for selection of the final target list from this pool. Both Ranks framework and Senti framework have the same steps until this step. After this, the frameworks differ from each other by using different user selection algorithms. Ranks framework uses FinalRank algorithm while the Senti framework uses SentiRank algorithm to select target users list.

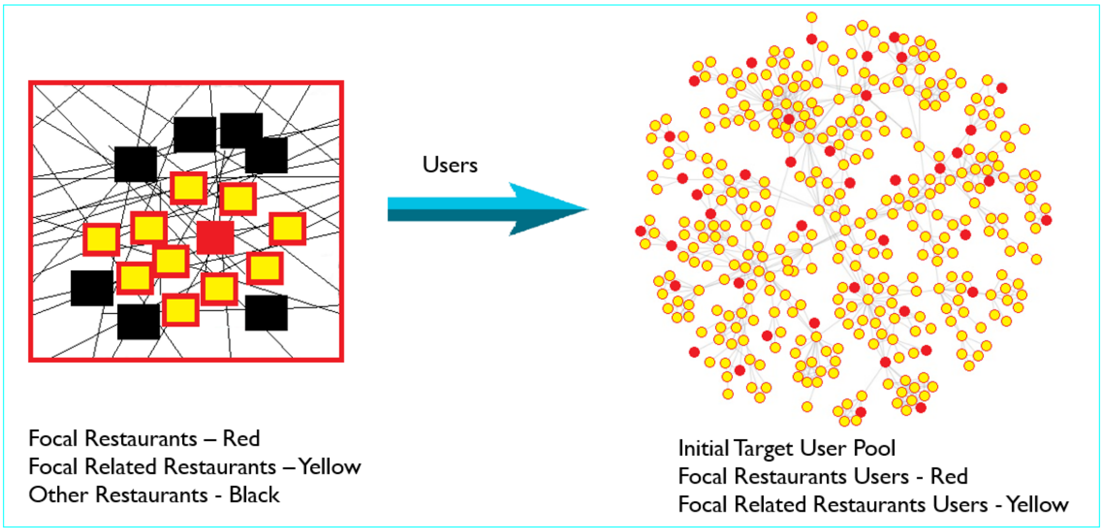

Figure 19 Focal related users

3.4.5 Communityrank Calculation

According to the Ranks framework, the importance of a community also depends on the Focal user (FU) count. FUs are all users who have reviewed the focal restaurant in the training period. The users who are in the same community as the FUs are more likely to review focal restaurants in the future, so the higher the FU count, the more importance is given to that community. This variable count was calculated for all the communities and then normalised before utilising. These normalized variables were named as the CommunityRank of communities. The users also have this rank value from the community they belong to and have been named as User CommunityRank. Normalization of CommunityRank values has two steps, firstly all rank values were divided by community user count and then the maximum value was found to divide all values by maximum value.

In the Ranks framework the FinalRank value of users were used to sort users in the initial target user pool and find the final target user list. To calculate FinalRank value theCommunityRank value of users in the initial target user pool was calculated. To calculate CommunityRank, the whole implicit user network was used and community detection was applied on this network. Then, user communities and initial target user pool users were combined to calculate CommunityRank. Figure 20 represents the whole implicit user network and shows how focal related users and focal users are seen in that implicit user network.

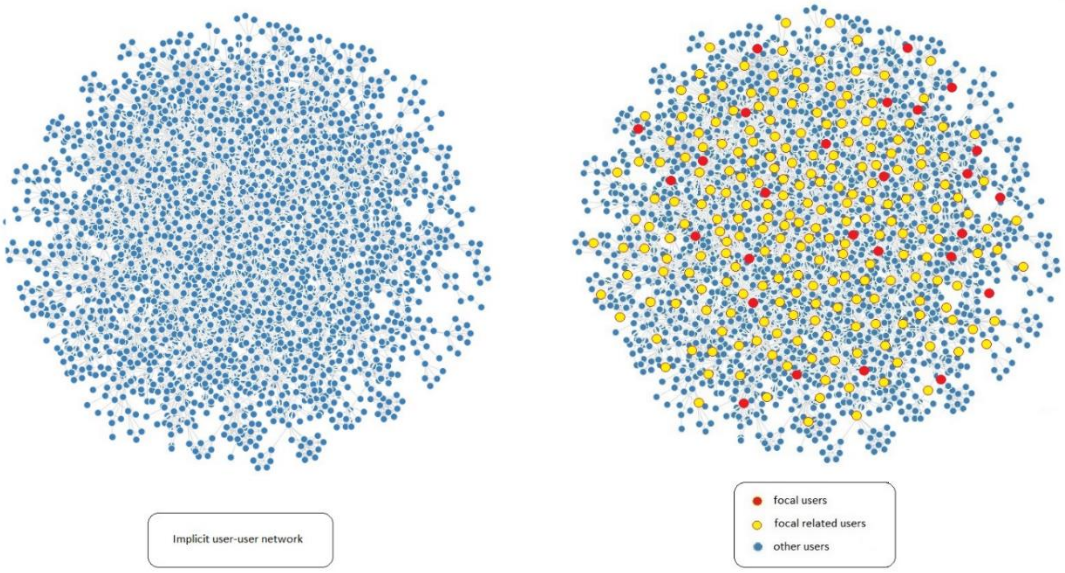

Figure 20 Implicit user network (left) and focal related users on it (right)

Community detection was applied to this network (Figure 21). Hundreds of user communities resulted from community detection process, some having few users in them others having thousands.

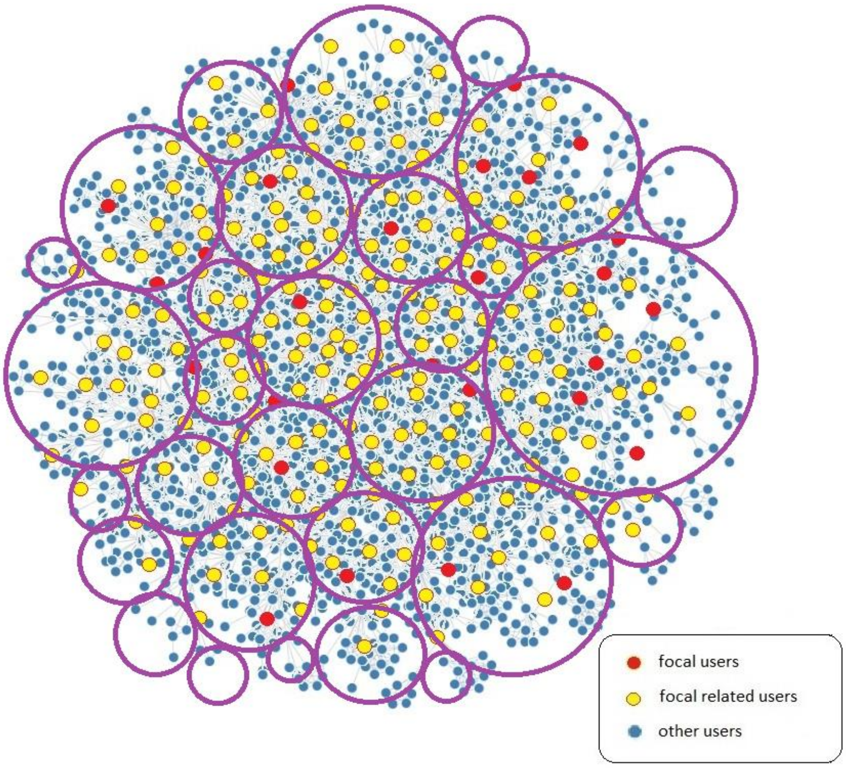

Figure 21 Community detection on implicit network

Having detected the communities in implicit user network, the CommunityRank value of each community was calculated. Figure 22 is an illustration for the user communities, all the circles are the communities and the CommunityRank values assigned to them according to focal user (red nodes) count in the circles. These CommunityRank values were assigned to all focal related users which are shown as yellow nodes in the circles, too. Yellow users are initial target user pool users. In this way the CommunityRank values for each user in the initial target user pool were calculated.

Four communities that contained focal users or focal related users were chosen randomly. Figure 22 shows these four communities. The first community has two focal users, the second has five focal users, third community doesn't have any focal users and fourth community has five focal users in it. According to these focal user counts, communities were assigned unnormalized CommunityRank values as 2, 5, 4 and 0 respectively.

For the first community, CommunityRank = 2 means every user in that community has a CommunityRank of 2.

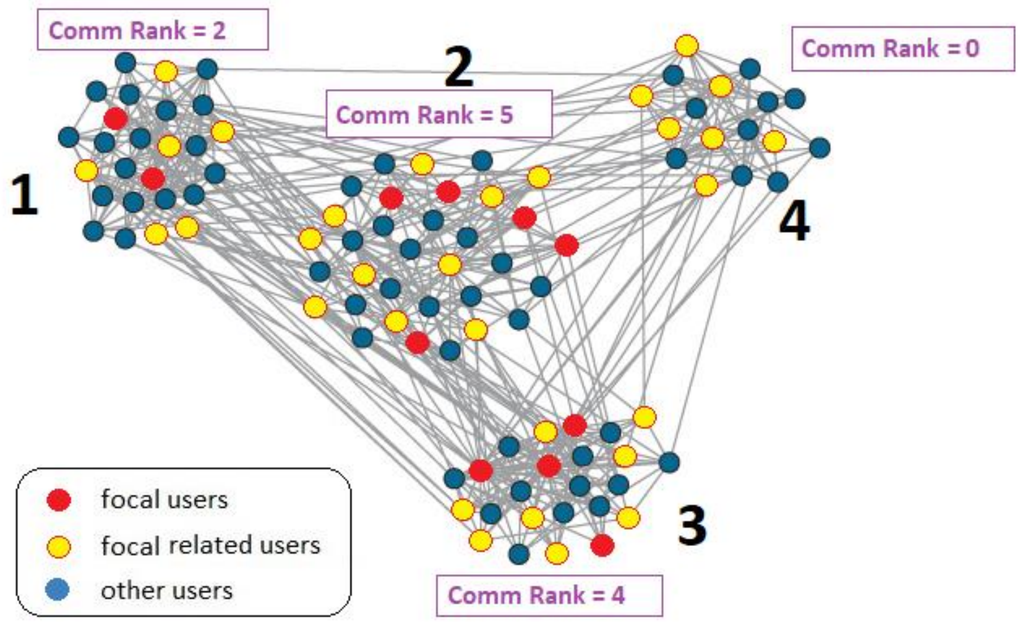

Figure 22 CommunityRank representation

After calculating all users' CommunityRank values each rank value was normalized. Figure 23 shows the normalization steps. There are two steps for normalization. First step calculates the division of CommunityRank with total user count in the community, so it equals to [2] / [26] = [0.08]. Second step calculates the normalization value by dividing the first step value with the maximum value so it is [0.08] / [0.17] = [0.47] (Figure 23).

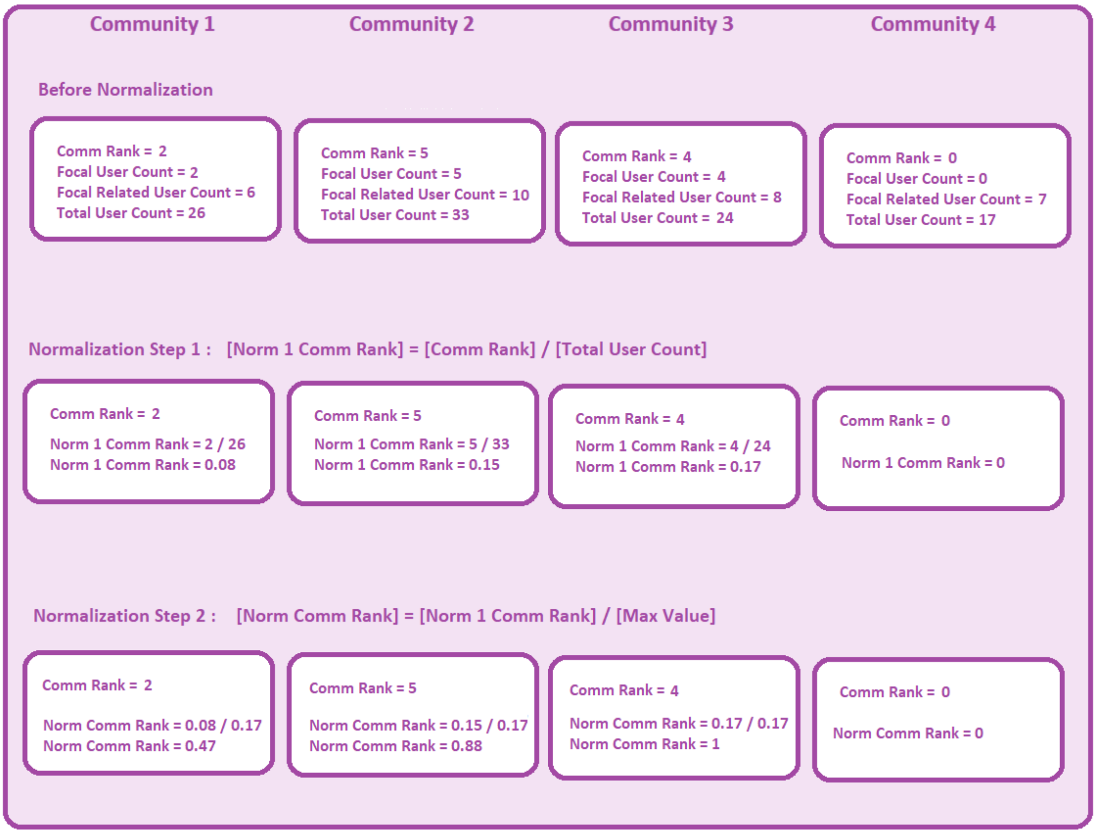

Figure 23 CommunityRank normalization

3.4.6 Finalrank Calculation

The final step for target user selection is the calculation of user FinalRank values. To select the target users, the new defined rank value of users was considered as FinalRank value. The FinalRank value for all users who are in the initial target user pool was calculated. According to this rank value, the top N target users list for each focal restaurant was selected. The user-final-rank value depends on the PageRank and CommunityRank values of users. Both PageRank and CommunityRank values of users who are in the initial target user pool have already been calculated. The previously calculated and normalized rank values were multiplied with some hyperparameter and the formulized FinalRank is shown in Figure 24.

α is a hyperparameter for PageRank and β is a hyperparameter for user CommunityRank. 17 different combinations of these hyperparameter values were used to find the best parameter pair in the validation period. After, all FinalRank values were calculated, each user FinalRank value was normalized by dividing it with maximum value. Normalization was done separately for each α/β pair.

$$\text{FinalRank} = \alpha * \text{PageRank} + \beta * \text{CommRank}$$

Figure 24 FinalRank formula

According to formula parameters, 17 different FinalRank values were calculated for each of user for each focal restaurant. So, there are 17 different Target User Lists for each focal restaurant. This is the aggregated target user list. With the help of Validation data, the best hyperparameter that gave the best result was determined by using Grid Search optimization technique.

Table 3. Top 10 Users Ordered By Finalrank for a Focal Restaurant

#	Focal Restaurant Name	User Name	Normalized FinalRank Value	Normalized CommRank	Normalised PageRank values
1	Midpoint	Seyyah ****	1.0000	1.0000	1.0000
2	Midpoint	Mehmet ****	0.8064	1.0000	0.7096
3	Midpoint	Candan ****	0.7721	1.0000	0.6581
4	Midpoint	Samet ****	0.7240	1.0000	0.5860
5	Midpoint	Fatih ****	0.6288	1.0000	0.4432
6	Midpoint	Melisa ****	0.6162	1.0000	0.4242
7	Midpoint	Entel ****	0.5977	1.0000	0.3966
8	Midpoint	Gezgin ****	0.5973	1.0000	0.3959
9	Midpoint	Hezarfen***	0.5921	1.0000	0.3882
10	Midpoint	Food ****	0.5868	1.0000	0.3802

Table 3 shows the top 10 user list according to FinalRank values calculated for a single hyperparameter (α/β = 2/1) for a single focal restaurant. According to this table and α/β pair these users were recommended as target users and the success result was confirmed by the validation data. However, values were calculated for only one α/β pair. The target user list for all α/β pairs was calculated and validated with validation data and the best α/β parameter was selected.

Table 4. α/β Rank Comparison for All Different User Selection List

#	Focal Restaurant Name	User Name	Normalized FinalRank Value	Alpha (α)	Beta (β)
6	Midpoint	Melisa ****	0.9943	1	100
6	Midpoint	Melisa ****	0.9779	1	25
6	Midpoint	Melisa ****	0.9477	1	10
6	Midpoint	Melisa ****	0.9040	1	5
6	Midpoint	Melisa ****	0.8081	1	2
6	Midpoint	Melisa ****	0.7697	2	3
6	Midpoint	Melisa ****	0.7383	5	6
6	Midpoint	Melisa ****	0.7121	1	1
6	Midpoint	Melisa ****	0.6859	6	5
6	Midpoint	Melisa ****	0.6545	3	2
6	Midpoint	Melisa ****	0.6162	2	1
11	Midpoint	Melisa ****	0.5202	5	1
12	Midpoint	Melisa ****	0.4766	10	1
15	Midpoint	Melisa ****	0.4464	25	1
16	Midpoint	Melisa ****	0.4299	100	1
16	Midpoint	Melisa ****	0.4242	1	0

For example one of the users from the table was selected, the user "Melisa ****" which is in 6th place in Table 3 then all 17 different FinalRank values were calculated according to hyperparameters for the same focal restaurant. Table 4 shows the FinalRank results of the user “Melisa” for each α/β pairs. The leftmost column in the table shows the rank of the user in the target user list for that α/β. For example, the user “Melisa” is 12^{th} place in the Top 100 list when α/β is “10/1” with corresponding FinalRank. Likewise the user is in 6^{th} place with the FinalRank calculation using most of the α/β pairs.

3.4.7 Evaluation of finalrank Framework in Validation Period

In Ranks Framework hyperparameters were optimized using validation data and grid search algorithm. To do so the success rates of all focal restaurants were examined. Success rate shows how many of the recommended target users commented on the focal restaurants in the validation period.

For every focal restaurant, the top N target user list was suggested for each α/β pair. Even though different N values (top 50, top 100, top 250 …) were used, the top 100 user list was used to evaluate success rates.

3.4.8 Actual target users

Actual target users are the users who will comment on the focal restaurant in future periods (validation and test period). Table 5 shows the focal restaurant “Istanbul Modern Café & Restaurant” and users who commented on this restaurant in the validation period. There are seven total users in the list who commented on the focal restaurant and one of them is a special user because this user (Elmashan***) is a new user so that this user has entered the system in the validation period and was not available in the system in the training period thus this user never would be in the target list. The other six users were sought after with Ranks framework and six is the Goal user count.

Table 5. Actual and Goal Users in Validation Period

Focal Restaurant Name	User Name	User ID	New User
Istanbul Modern Cafe & Restaurant	Canberk ****	42975043	No
Istanbul Modern Cafe & Restaurant	Elmashan ****	51276217	Yes
Istanbul Modern Cafe & Restaurant	Food ****	33571838	No
Istanbul Modern Cafe & Restaurant	Grace ****	35950593	No
Istanbul Modern Cafe & Restaurant	Kaim ****	35098874	No
Istanbul Modern Cafe & Restaurant	Mc ****	36428899	No
Istanbul Modern Cafe & Restaurant	Pelin ****	32541749	No

3.4.9 Target user selection in validation period

Table 6 shows some of the users suggested by Ranks Framework with hyperparameter $\alpha/\beta = 2/1$. Only the top 20 users of the target users for the focal restaurant have been listed and it can be seen that only one user is successfully recommended (8^{th} place). So, the success count is 1 for the top 20 target user selection for that restaurant.

Table 6. Top 20 Suggessted Users by the Ranks Framework

#	Focal Restaurant Name	User Name	User ID	Normalized FinalRank Value
1	Istanbul Modern Cafe & Res.	Blabla ****	43372826	1.0000
2	Istanbul Modern Cafe & Res.	Cemre ****	18758707	0.9978
3	Istanbul Modern Cafe & Res.	Mmm ****	30547307	0.9255
4	Istanbul Modern Cafe & Res.	Seyyah ****	2429694	0.9224
5	Istanbul Modern Cafe & Res.	Sevde ****	17124193	0.8613
6	Istanbul Modern Cafe & Res.	Azra ****	17116843	0.8489
7	Istanbul Modern Cafe & Res.	Ahu ****	17070793	0.8080
8	Istanbul Modern Cafe & Res.	Food ****	33571838	0.7786
9	Istanbul Modern Cafe & Res.	Sera ****	16140381	0.7428
10	Istanbul Modern Cafe & Res.	Erdi ****	1603185	0.6821
11	Istanbul Modern Cafe & Res.	Gurmeli ****	2139112	0.6722
12	Istanbul Modern Cafe & Res.	Mehmet ****	5709121	0.6709
13	Istanbul Modern Cafe & Res.	ISIL ****	19470664	0.6447
14	Istanbul Modern Cafe & Res.	Gamze ****	8944651	0.6420
15	Istanbul Modern Cafe & Res.	Berat ****	16206537	0.6326
16	Istanbul Modern Cafe & Res.	Miray ****	37096740	0.6314
17	Istanbul Modern Cafe & Res.	Zafer ****	11493371	0.6265
18	Istanbul Modern Cafe & Res.	Candan ****	30252304	0.6264
19	Istanbul Modern Cafe & Res.	Pelin ****	29999892	0.6198
20	Istanbul Modern Cafe & Res.	Berkin ****	27806356	0.6158

If I top 1000 were suggested for the same restaurant, then four of the goal users coult be found successfully. Table 7 shows the four users with their rank in the top 1000 list. For example, the user “Food” is 8th place in the list, the user “Kaim” is 37th place in the order and the user “Grace” is 55th place in the target list order and the user “Pelin” is in 956th place in the top 1000 target user list.

Table 7. Goal Users in the Target User List

Rank	Restaurant Title	User Name	User ID	Normalized FinalRank Value
8	Istanbul Modern Cafe & Restaurant	Food ****	33571838	0.7786
37	Istanbul Modern Cafe & Restaurant	Kaim ****	35098874	0.5379
55	Istanbul Modern Cafe & Restaurant	Grace ****	35950593	0.5076
956	Istanbul Modern Cafe & Restaurant	Pelin ****	32541749	0.3631

3.4.10 Success results in validation period

Recall success rates are calculated by the formula:

[Recall Success Rate] = [Successful found user] / [Goal user count].

For the previous examples according to Table 5, only one user prediction was successful when the goal user count was six, then the success rate for this top 20 user suggestion:

[Successful found user] / [Goal user count] = 1 / 6 = 16.6% success rate.

If the top 1000 users for the restaurant “Istanbul Modern Café & Restaurant” was suggested, then four of the goal users would have been predicted successfully. Table 7 shows the successful users with their rank in the top 1000 list. The success rate of this top 1000 selection is:

[Successful found user] / [Goal user count] = 4 / 6 = 66.6 %

As seen above, one example of focal restaurant was analyzed to explain success rates. Likewise, Table 8 shows the most successful top 20 focal restaurants with their success rates. $\alpha / \beta = 2 / 1$ was used here, too. In Table 8, Actual User Count means the number of total users that commented on the focal restaurant in the validation period. The Goal User Count shows how many of the actual users are also available in the training period to try to predict in advance. The Success Count shows the user count successfully found for the focal restaurant. Success Rate shows the rate percentage between Success Count and Goal User Count.

According to Table 8 it can be said that the Ranks framework is really successful for many restaurants. For the restaurant "Istanbul Modern Cafe & Restaurant" the top 100 users were suggested to find six goal users and three of them were found successfully. While Recall success rate is 50.0%, precision success rate is 3.0 %.

Table 8. Top 20 Focal Restaurant with Success Rates (α / β = 2 / 1)

Focal Restaurant	Actual User Count	Goal User Count	Success Count	Success Rate (%)
Borisin Yeri	6	4	2	50.00
Istanbul Modern Cafe & Res	7	6	3	50.00
Spago - St. Regis Istanbul	7	4	2	50.00
Sahan	9	5	2	40.00
Kirinti	6	5	2	40.00
San Marcos Caffe	11	8	3	37.50
Akin Restoran	19	3	1	33.33
Ankara Sereserpe Köftecisi	9	6	2	33.33
Carls Jr.	4	3	1	33.33
Cup of Joy	3	3	1	33.33
Pizzeria Pera	8	3	1	33.33
Rumeli Çikolatacisi	10	6	2	33.33
Latife Türk Kahvecisi	7	3	1	33.33
Agapia Meyhane	7	3	1	33.33
Kase No.16	14	9	3	33.33
Big Chefs	9	6	2	33.33
Dardenia Fish & Sushi	8	3	1	33.33
Köskeroglu Baklava	19	10	3	30.00
B.blok	31	17	5	29.41
Tarihi Sultanahmet Köftecisi	11	7	2	28.57

3.4.11 Best hyperparameter selection

Until now, for all the results calculated above $\alpha / \beta = 2 / 1$ was used.

The grid search algorithm was employed to find the best hyperparameter. 17 α / β pairs were used. Increasing the hyperparameter count increases the run time of all frameworks. Even with these 17 hyperparameters it took one day to complete all work of the framework. So, these 17 values were chosen carefully to cover all important cases. Values {0, 1, 2, 3, 5, 6, 10, 25, 100} were taken and distributed below as α / β pairs

[(1,0) (0,1) (1,1) (10,1) (1,10) (100,1) (1,100) (2,1) (1,2) (5,1) (1,5) (2,3) (3,2) (25,1) (1,25) (5,6) (6,5)]

Table 8 shows the success rates of just 20 focal restaurants which also "α / β" pair is equal to "2 / 1" in there. But there are 300 focal restaurants and 17 different α / β pairs. So, to find the best α / β pair, the average rate of all the success rates for all of the focal restaurants was calculated.

If the average success rate for Table 8 is calculated, it would be found that the average success rate is equal to the sum of all success rates divided by 20 that is equal to [722.11 / 20] = 36.10 % for $\alpha / \beta = 2 / 1$. Likewise, the sum of success rates was calculated and divided by 300 and then the average values for each α / β pair was calculated. Table 9 shows average success rates of all focal restaurants for each different α / β pair.

According to Table 9, hyperparameter value of $\alpha / \beta = 5 / 1$ has the best average success rate so $\alpha / \beta = 5/1$ chosen as optimized α / β hyperparameter and it was used in all calculations of the test period success rates.

Table 9. α / β Success Rates

ALPHA (α)	BETA (β)	Average Success Rate of 300 Focal Restaurant (%)
5	1	7.8511
2	1	7.7330
10	1	7.5654
3	2	7.5400
25	1	7.4504
5	6	7.4466
1	1	7.4466
6	5	7.4281
2	3	7.4023
1	2	7.3660
100	1	7.3282
1	5	7.2583
1	0	7.1054
1	10	6.9824
1	25	6.7039
1	100	6.6522
0	1	1.7330

3.4.12 Success results in test period

Table 10 shows the success rate of 20 of the focal restaurants in the test period. With these results, it can be said that the Ranks framework is quite successful. For example, for the focal restaurant Zencefil this framework successfully predicted the half of the goal user count. These success results were calculated by recommending top 100 target user list by their FinalRank.

Table 10. Top 20 Focal Restaurants Test Period Success Result

Focal Restaurant	Actual User Count	Goal User Count	Success Count	Success Rate (%)
Zencefil	18	4	2	50.00
Frankie Istanbul - The Sofa	4	2	1	50.00
Karadeniz Pide Kebap Salonu	12	4	2	50.00
Daily Dana Burger & Steak	6	5	2	40.00
Beyaz Firin & Brasserie	10	5	2	40.00
Bunco	13	8	3	37.50
Baylan	29	8	3	37.50
Çukurcuma 49	16	8	3	37.50
Viyana Kahvesi	28	11	4	36.36
Safran Pub & Meyhane	13	6	2	33.33
Yirmibir Kebap	9	3	1	33.33
OT	9	3	1	33.33
Yakup 2 Restaurant	8	3	1	33.33
Dirty Hands	15	6	2	33.33
Kruvasan	30	9	3	33.33
Tatbak	11	6	2	33.33
Sekerci Cafer Erol	24	10	3	30.00
Bira Fabrikasi	10	7	2	28.57
Balkan Lokantasi	13	7	2	28.57
Rose Marine	25	11	3	27.27

3.4.13 Precision – Recall results

Precision is the percentage of the recommended user that successfully recommended by our framework. On the other hand, Recall is the percentage of goal users that are successfully recommended by our framework. In the test results given above, there were three variables; Success Count, Goal User Count and Recommended user count. Recommended user count is 100 in the calculations above.

Success rates were calculated according to the Recall formula by comparing the successful user count with the goal user count. Otherwise, if it was Precision metrics, successful user count would be compared with total recommended user count. To be able to draw the precision-recall curve all success rates were calculated with different recommended user counts. Recommended user count is equal to how many of the target users selected when a recommendation for any focal restaurant was made. Until now the top 100 target user list was used in all calculations.

All the success results and hyperparameter selection parts were done with using “Top 100” best target user selection. That is, in recommendation target list, for any focal restaurants only the top 100 of them were selected and success rates were calculated accordingly. Besides top 100 user count, many different target user count were selected that ranging from top 50 to top 1000 and many different success rates were calculated for all of them. Also, only Recall metric was used to to calculate all previous success results since all previous calculations considered only goal user count. In addition to Recall metric, Precision metric success results were also calculated for all focal restaurants. Below Precision and Recall success rates are shown for some of the focal restaurants with the top N target user selections separately and together.

Table 11 shows the Precision metric success rates for 20 focal restaurants with target user selection count from top 10 to top 800. In presicion metric, success rates are decreasing as recommended user count increase because increase rate of denomitor is much bigger than the increase rate of correspondent goal user count to predict.

Table 11. Precision Success Rates (%)

Focal Restaurant Name	Top 100	Top 200	Top 300	Top 400	Top 500	Top 600	Top 700	Top 800
B.blok	6.00	4.00	3.33	2.50	2.00	2.00	1.71	1.50
Viyana K.	4.00	2.00	1.33	1.00	1.00	0.83	0.71	0.63
Akali	4.00	3.50	2.33	2.25	1.80	1.67	1.57	1.50
Asuman	4.00	2.50	2.33	2.00	1.60	1.67	1.43	1.25
Mendels	3.00	2.00	1.33	1.00	0.80	0.83	0.86	0.75
Baylan	3.00	1.50	1.33	1.00	0.80	0.67	0.57	0.50
Rose Marine	3.00	2.00	1.33	1.00	0.80	0.67	0.57	0.50
Meshur Dond	3.00	1.50	1.00	0.75	0.60	0.67	0.57	0.63
Varuna G.	3.00	1.50	1.00	0.75	0.60	0.50	0.43	0.38
Kruvasan	3.00	1.50	1.33	1.00	0.80	0.83	0.71	0.63
Bunco	3.00	1.50	1.00	0.75	0.60	0.50	0.43	0.38
Mua Gelatieri	3.00	1.50	1.33	1.00	0.80	0.67	0.57	0.50
Çukurcuma	3.00	1.50	1.00	0.75	0.60	0.50	0.43	0.38
Sekerci Cafer	3.00	2.00	1.67	1.50	1.20	1.17	1.00	0.88
Mangerie	2.00	1.50	1.33	1.00	0.80	0.67	0.57	0.50
Karadeniz D.	2.00	1.50	1.00	0.75	0.80	1.00	0.86	0.75
Kasibeyaz	2.00	1.50	1.00	0.75	0.60	0.67	0.71	0.75
Bira Fabr.	2.00	1.00	1.00	0.75	0.60	0.67	0.57	0.50
MOC Ist.	2.00	1.50	1.67	1.25	1.00	0.83	0.71	0.75
Divan	2.00	1.00	1.00	0.75	0.60	0.67	0.57	0.50
Beşiktaş Pilav	2.00	1.50	1.00	0.75	0.60	0.50	0.43	0.38

Table 12 also shows the same data with Recall metric success rates. In recall metrics success rate is getting increase as recommended user count increase as it is shown in the Table 12.

Table 12. Recall Success Rates (%)

Focal Restaurant Name	Top 100	Top 200	Top 300	Top 400	Top 500	Top 600	Top 700	Top 800
B.blok	15.38	20.51	25.64	25.64	25.64	30.77	30.77	30.77
Viyana Kahvesi	36.36	36.36	36.36	36.36	45.45	45.45	45.45	45.45
Akali	9.30	16.28	16.28	20.93	20.93	23.26	25.58	27.91
Asuman	9.09	11.36	15.91	18.18	18.18	22.73	22.73	22.73
Mendels	9.09	12.12	12.12	12.12	12.12	15.15	18.18	18.18
Baylan	37.50	37.50	50.00	50.00	50.00	50.00	50.00	50.00
Rose Marine	27.27	36.36	36.36	36.36	36.36	36.36	36.36	36.36
Meshur Dond.	16.67	16.67	16.67	16.67	16.67	22.22	22.22	27.78
Varuna Gezgin	14.29	14.29	14.29	14.29	14.29	14.29	14.29	14.29
Kruvasan	33.33	33.33	44.44	44.44	44.44	55.56	55.56	55.56
Bunco	37.50	37.50	37.50	37.50	37.50	37.50	37.50	37.50
Mua Gelatieri	27.27	27.27	36.36	36.36	36.36	36.36	36.36	36.36
Çukurcuma 49	37.50	37.50	37.50	37.50	37.50	37.50	37.50	37.50
Sekerci Cafer	30.00	40.00	50.00	60.00	60.00	70.00	70.00	70.00
Mangerie	15.38	23.08	30.77	30.77	30.77	30.77	30.77	30.77
Karadeniz D.	9.52	14.29	14.29	14.29	19.05	28.57	28.57	28.57
Kasibeyaz	11.76	17.65	17.65	17.65	17.65	23.53	29.41	35.29
Bira Fabrikasi	28.57	28.57	42.86	42.86	42.86	57.14	57.14	57.14
MOC Istanbul	15.38	23.08	38.46	38.46	38.46	38.46	38.46	46.15
Divan Brasserie	25.00	25.00	37.50	37.50	37.50	50.00	50.00	50.00
Besiktas Pilav	20.00	30.00	30.00	30.00	30.00	30.00	30.00	30.00

Table 13 shows the comparison of the precision and recall average success rates. There were 300 focal restaurants and the average success rates of all the focal restaurants with different user recommendation count are shown in Table 13.

Table 13. Precision-Recall Average Success Rates (%) Comparison

Metrics	Avg Top 100	Avg Top 200	Avg Top 300	Avg Top 400	Avg Top 500	Avg Top 600	Avg Top 700	Avg Top 800
PRECISION	0.91	0.78	0.66	0.57	0.49	0.45	0.4	0.37
RECALL	9.61	15.98	20.22	23.15	24.85	26.75	28.04	29.67

Lastly, the recommended user count was extended to the top 1000 user selection and then the precision recall curve was drawn as can be seen in Figure 25. Points shows the Top N user selection.

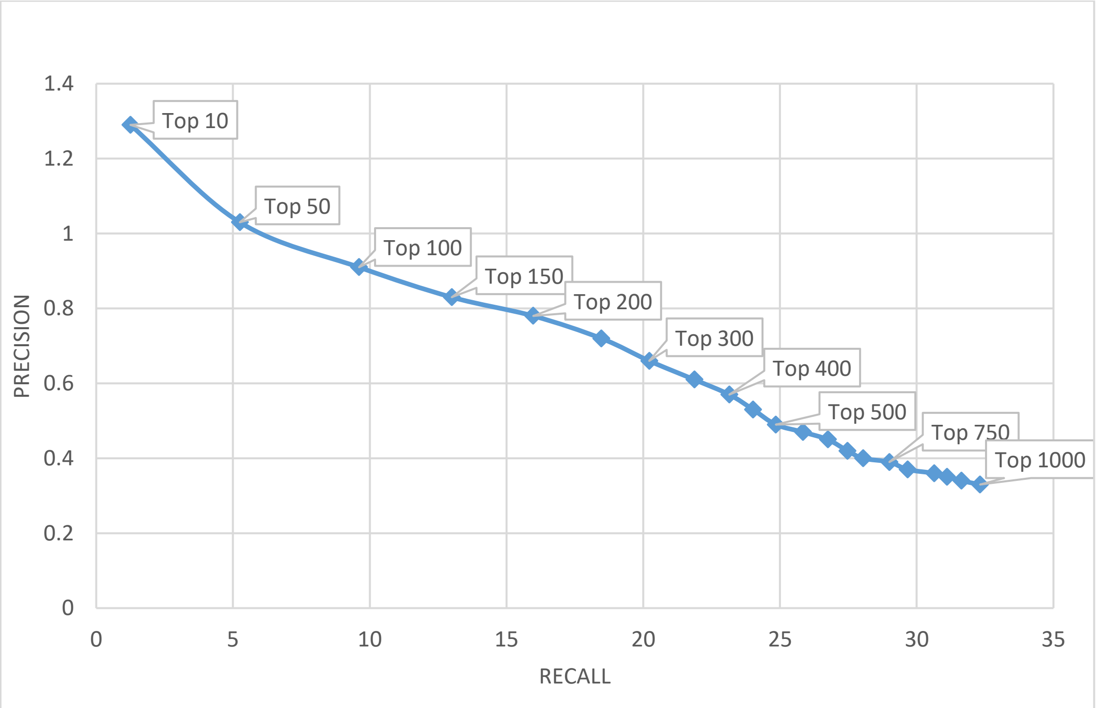

Figure 25 Ranks framework precision-recall curve

3.5 Senti framework

In the Senti Framework the same steps were conducted as the Ranks Framework until finding the initial target user pool. They both used the same cleansed dataset, restaurant network, PageRank, communities of restaurants, focal related restaurants and same initial target user pool. It differs from the Ranks Framework especially on the selection part from the initial user pool. This framework uses SentiRank value of users to select them so that's why it called this framework Senti framework. SentiRank was calculated for each user review by using a computer program which is capable of analyzing Turkish sentences.

Users review a restaurant, leave a comment about that restaurant with a rating. If a user likes the restaurant, he/she leaves a positive comment with a high rating, otherwise makes a negative comment. Users' positivity or negativity can be understood by applying sentiment analysis on user comments. User rates given by the users to the restaurant can also be used to understand user positivity or negativity. In this framework user comments were used to understand user positivity or negativity and gave rank according to user sentiment. SentiRank values can be calculated reviews based, user based, or community user based by context they are used. In this framework user SentiRank values were calculated for the users who were in the initial target user pool according to reviews for any of the focal related restaurants.

In this approach, there were only Training and Test periods, there was no need for the Validation period. In the Training period, some of the steps were similar to those in the Ranks Framework. Senti Framework outlines:

1. Data pre-processing that consists of turning all review data into useful condition and language.

2. Network generation and normalization that generated weighted/directed/undirected networks for both users and restaurants by using user review data.

3. Community detection and PageRank for only the Restaurants network using undirected-weighted restaurant network and directed-weighted restaurant network respectively.

4. In light of restaurants' communities and PageRank values, restaurants related to the focal restaurants are specified. Related restaurants are the most similar restaurants to the focal restaurants based on common user count. Also, all users that reviewed these related restaurants are selected for the initial user pool.

5. Sentiment analysis and SentiRank values are calculated by using user comment positivity for the restaurants. Related restaurant users are then ranked by their SentiRank.

6. The top K target users are selected from the initial user pools by their SentiRank value and these K target users are target users.

7. Target users are evaluated and analysed with goal users of test data and success rates are stated.

Figure 26 shows the overall steps of the Senti Framework which is SentiRank approach recommendation system.

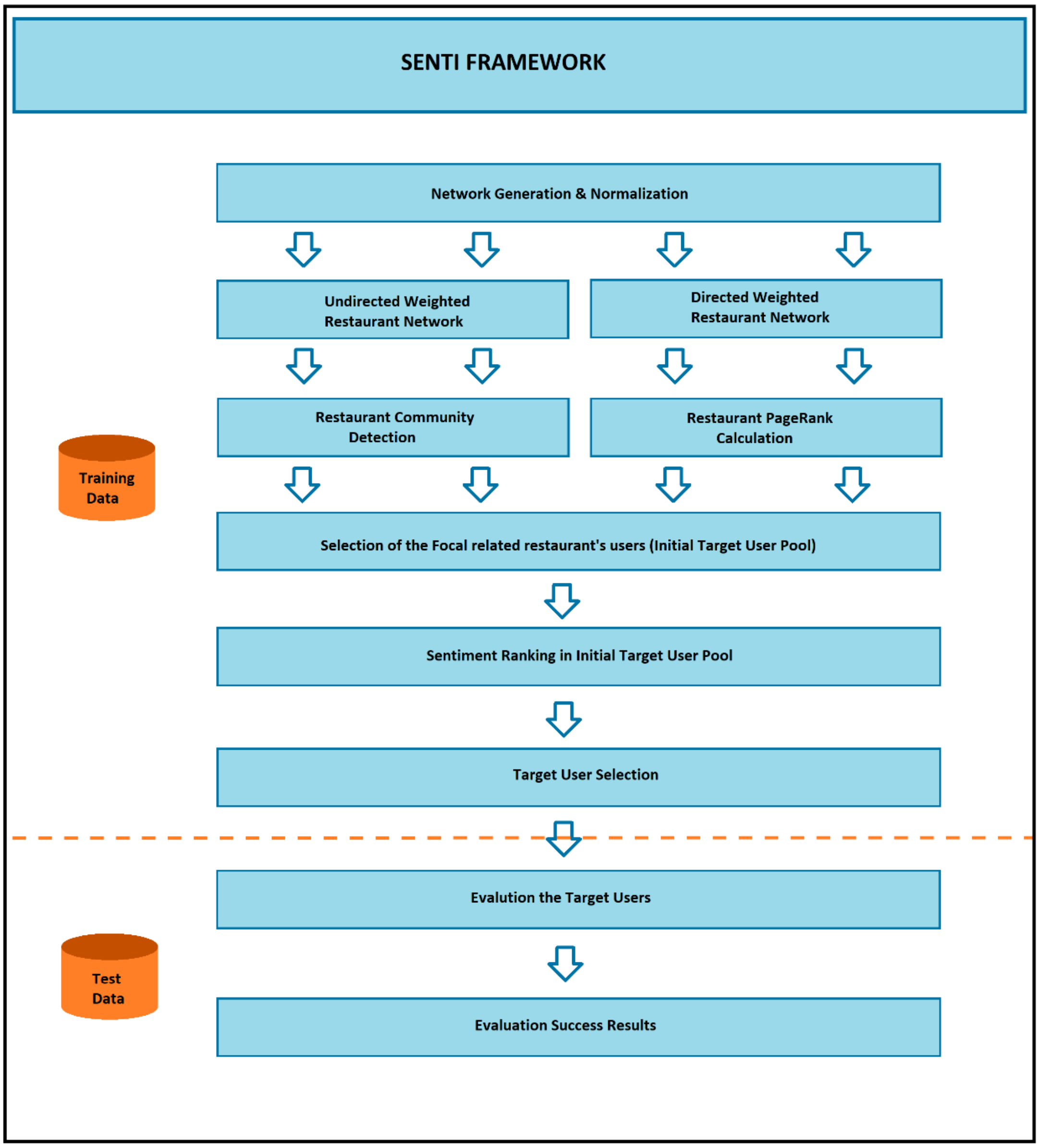

Figure 26 Senti framework flowchart

3.5.1 Networks

The same networks were used with Ranks Framework until initial target user pool. Figure 27 shows all focal related users which shows the users who commented on the focal related restaurants. These are the users which were ranked with SentiRank and top N of them was selected as the final target user list for this framework.

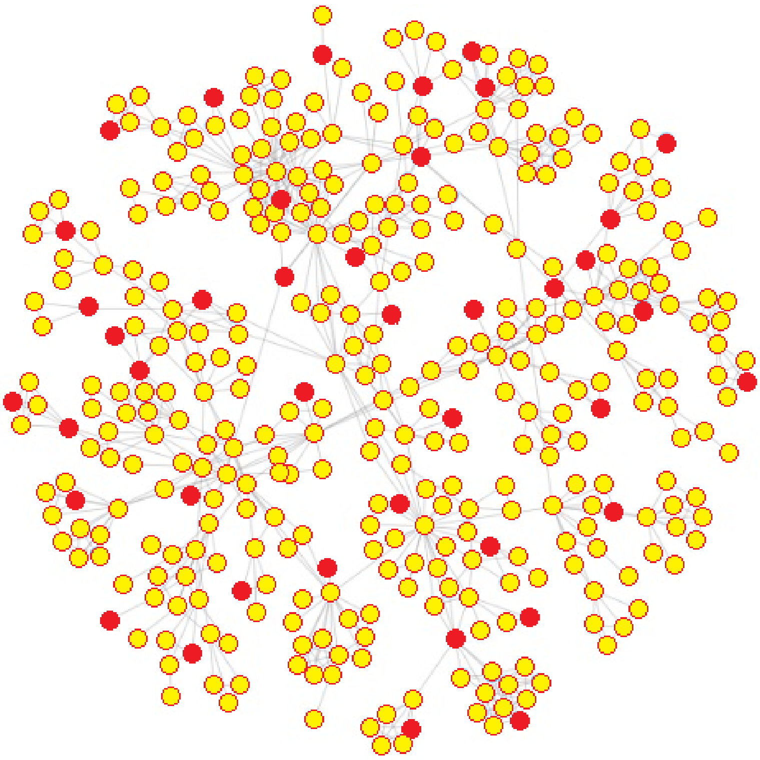

Figure 27 Initial target user pool (yellow)

3.5.2 Sentiment analysis

The sentiment analysis software which was used in this thesis is called SentiStrength V2.2. The Turkish word dictionary used with this software is developed by Vural et al. (2013) on the base of the largest open source Turkish natural language processing library called “Zemberek”, which is commonly used in Open Office and Libre Office software.

Since the SentiStrength program only takes single text string as input it was not used directly. Instead, a new program was developed which extended the SentiStrength program and the new version is able to process thousands of comments. In order to find the strength of the sentiments, the program first reads all the review data and calls the SentiStrength program for each review one by one and gets the corresponding sentiment value which is named as the “sentiment strength” of a review. For each review, customer ratings given to the restaurants was also collected that are the subjects of the reviews.

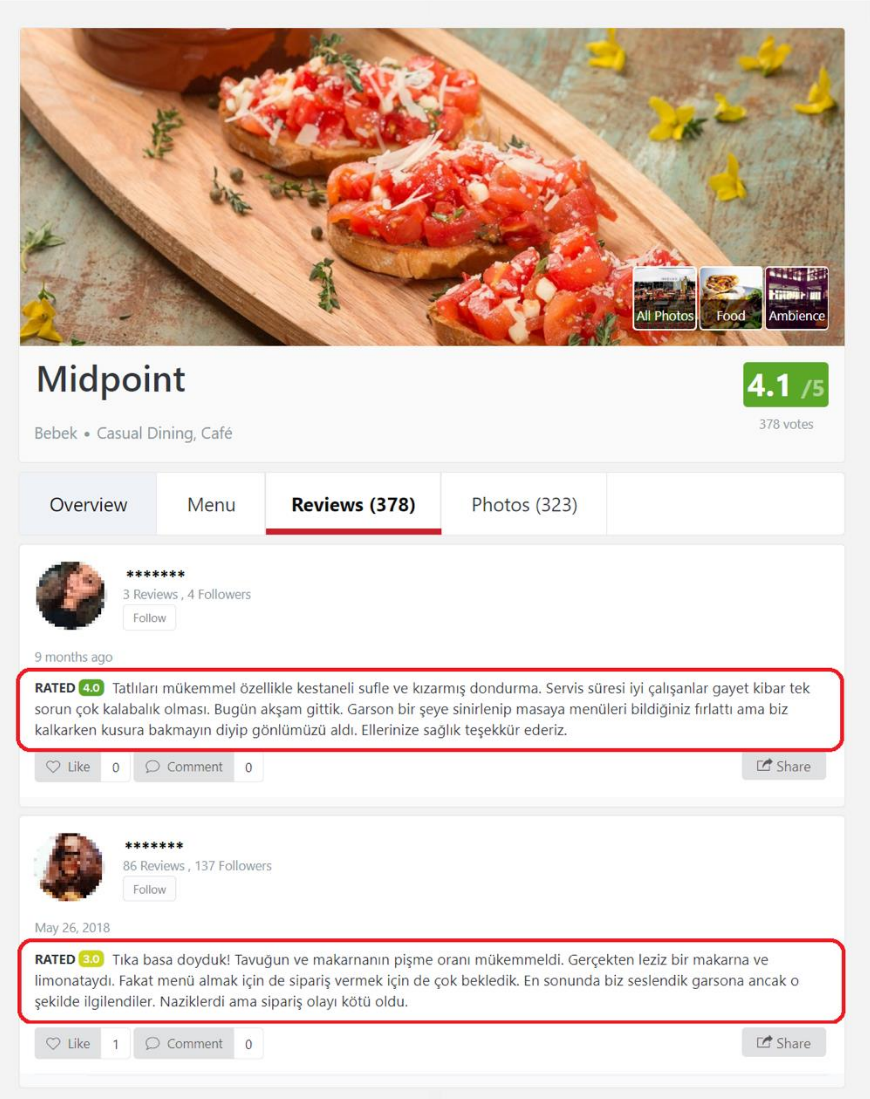

Figure 28 Some comments for a restaurant

For example Figure 28 shows two of the comments written for one of the restaurants. If these comments were used with our developed program then it would give these results:

Review 1:“Tatlıları mükemmel özellikle kestaneli sufle ve kızarmış dondurma. Servis süresi iyi çalışanlar gayet kibar tek sorun çok kalabalık olması. Bugün akşam gittik. Garson bir şeye sinirlenip masaya menüleri bildiğiniz fırlattı ama biz kalkarken kusura bakmayın diyip gönlümüzü aldı. Ellerinize sağlık teşekkür ederiz."

Review 2: "Tıka basa doyduk! Tavuğun ve makarnanın pişme oranı mükemmeldi. Gerçekten leziz bir makarna ve limonataydı. Fakat menü almak için de sipariş vermek için de çok bekledik. En sonunda biz seslendik garsona ancak o şekilde ilgilendiler. Naziklerdi ama sipariş olayı kötü oldu."

SentiRank = Positive Rank – Negative Rank

For the first comment SentiRank calculation is:

Positive Rank = 4

Negative Rank = 3

SentiRank = 4 – 3 = 1

For the second comment SentiRank calculation is:

Positive Rank = 4

Negative Rank = 2

SentiRank = 4 – 2 = 2

These are the review based SentiRank values for any reviews. But it was needed to find user based SentiRank values since a user could give many reviews for many restaurants. First, SentiRank was calculated for every review given to the focal related restaurants. Then all reviews were grouped based on users and the sum of all SentiRank values of each users were calculated. This total rank value is the user's SentiRank value. The SentiRank of users also depended on the focal restaurants because all reviews given to focal related restaurants and obtained according to the focal restaurant were calculated and summed.

Having found all user based SentiRank values, all SentiRank values normalized by dividing them with the maximum of Sentirank values.

3.5.3 Actual target users

First a focal restaurant was selected and was specified who would be their actual user and who would be the goal users. Then who could be recommended as target users for this focal restaurant was shown. Lastly, successfully predicted users recommended by the framework were shown.

Table 14. Actual and Goal User for a Focal Restaurant

Focal Restaurant	User Name	User ID	New User
Beyaz Firin & Brasserie	Ahmet ****	56941517	YES
Beyaz Firin & Brasserie	Alp ****	39024140	NO
Beyaz Firin & Brasserie	Burak ****	17226547	NO
Beyaz Firin & Brasserie	Burcu ****	30113809	NO
Beyaz Firin & Brasserie	Gurman ****	76877552	YES
Beyaz Firin & Brasserie	Idil ****	19782286	YES
Beyaz Firin & Brasserie	Müge ****	17974837	NO
Beyaz Firin & Brasserie	Naz ****	53917522	YES
Beyaz Firin & Brasserie	Tadim ****	37093596	NO
Beyaz Firin & Brasserie	Yami ****	18102046	YES

In Table 14, there are 10 actual users meaning they all commented on the focal restaurant in the test period. But only five of them were goal users because the other five of them were new users who just entered to the system in test period so they couldn’t be predicted in the training period. It was tried to predict and recommend these Goal users as much as possible.

3.5.4 Target user selection

Table 15 shows the top 20 of recommended target users for the same focal restaurant.

Table 15. Recommended Target User List for the Focal Restaurant

#	Restaurant Title	User Name	Normlized SentimentRank	User ID
1	Beyaz Firin & Brasserie	Alp ****	1.0000	31385855
2	Beyaz Firin & Brasserie	Büsra ****	0.9057	34823531
3	Beyaz Firin & Brasserie	Seyyah ****	0.8868	2429694
4	Beyaz Firin & Brasserie	Efil ****	0.7170	20022559
5	Beyaz Firin & Brasserie	Emre ****	0.5849	16231765
6	Beyaz Firin & Brasserie	Kadiköy ****	0.5849	34304419
7	Beyaz Firin & Brasserie	Kübra ****	0.5660	38022269
8	Beyaz Firin & Brasserie	Fatih ****	0.5660	34125624
9	Beyaz Firin & Brasserie	Gizem ****	0.5472	16036301
10	Beyaz Firin & Brasserie	Yasasin ****	0.5283	35571612
11	Beyaz Firin & Brasserie	Dogan ****	0.5094	30529730
12	Beyaz Firin & Brasserie	Derya ****	0.5094	37211477
13	Beyaz Firin & Brasserie	Miiika ****	0.4906	29020165
14	Beyaz Firin & Brasserie	Candan ****	0.4717	30252304
15	Beyaz Firin & Brasserie	Mmm	0.4528	30547307
16	Beyaz Firin & Brasserie	Azra ****	0.4528	17116843
17	Beyaz Firin & Brasserie	Tansel ****	0.4528	42915062
18	Beyaz Firin & Brasserie	Necmeddin ****	0.4528	36591195
19	Beyaz Firin & Brasserie	Alp ****	0.4528	39024140
20	Beyaz Firin & Brasserie	Yagmur ****	0.4528	31592513

Table 16 shows the successful recommendations with their ranks. For example "Müge" is a goal user for the focal restaurant and she was ranked as 295th in the list, so at least 295 users need to be recommended for "Müge" to be successfully predicted. According to the table below, if the top 100 of the target user list were selected, then two of the goal users would be found, whereas if the top 500 were selected, four out of the five goal users would be predicted successfully.

Table 16. Goal Users in the Target User List

#	Restaurant Title	User Name	User ID	Normalized SentiRank Value
19	Beyaz Firin & Brasserie	Alp ****	39024140	0.4528
46	Beyaz Firin & Brasserie	Tadim ****	37093596	0.3207
295	Beyaz Firin & Brasserie	Müge ****	17974837	0.1320
477	Beyaz Firin & Brasserie	Burcu ****	30113809	0.0943

3.5.5 Success rates in test period

Success rate can be measured with two different metrics, Precision and Recall type success rates.

First it was shown that Recall metric success rates for some focal restaurants then precision metric success rates were calculated with different selection counts and then both of them were compared with a precision – recall curve.

Table 17 shows the top 20 of the recommended target users for the same focal restaurant. In Table 17, the success results of only 20 of the recommended users will be evaluated out of the top 100. Even if only the top 20 of the target user list were selected, one of the users, “Alp ***” in the 19th place would be successfully predicted. The success rate considering only 20 users can be calculated as:

(Recall) Success Rate = [Successful Prediction Count] / [Goal User Count]

Success Rate = 1 / 5 = 20 %

Table 17. Successful Focal Restaurants with Their Success Rates

Focal Restaurant	Actual User Count	Goal User Count	Success Count	Success Rate (%)
Beyaz Firin & Brasserie	10	5	2	40.00
Çukurcuma 49	16	8	3	37.50
Safran Pub & Meyhane	13	6	2	33.33
Yakup 2 Restaurant	8	3	1	33.33
Yirmibir Kebap	9	3	1	33.33
Zübeyir Ocakbasi	8	6	2	33.33
Balkan Lokantasi	13	7	2	28.57
Somunarasi	22	14	4	28.57
Rumeli Çikolatacisi	14	4	1	25.00
Karga Bar	8	4	1	25.00
Malta Köskü	15	4	1	25.00
Karadeniz Pide Kebap	12	4	1	25.00
Minoa	31	8	2	25.00
Chaya Galata	13	8	2	25.00
Çigdem Pastanesi	10	4	1	25.00
Bridge Restaurant & Cafe	23	4	1	25.00
Aida - Vino E Cucina	21	8	2	25.00
Yagcioglu Pastaneleri	9	4	1	25.00
Metet Közde Döner	31	12	3	25.00
Bunco	13	8	2	25.00

Table 17 shows the most successful focal restaurant in terms of success rates. These success rates calculated by using the top 100 of user selection and Recall method. The success rates change when the selected user count increases.

3.5.6 Precision – Recall results

As it is explained in the Ranks Framework, precision and recall consider different aspects of success results. One looks at Goal user count, while the other looks at how many users are recommended to the restaurant when it comes to calculating success rates.

Table 18 and Table 19 show the Precision and Recall success rates separately. The table columns show the different recommendation user counts. For example for the focal restaurant “Asuman” if 100 users were recommended then Precision success rate is 4%, Recall success rate is 9.09%. If 400 users were to be recommended this would yield a Precision success rate of 1.75 % and 15.91% for Recall. As it can be seen, Precision success rates decrease as the recommendation count increases while Recall success rates increase.

Table 18. Precision Success Rates (%)

Focal Restaurant Name	Top 100	Top 200	Top 300	Top 400	Top 500	Top 600	Top 700	Top 800
Asuman	4.00	2.50	1.67	1.75	1.40	1.33	1.29	1.13
Somunarasi	4.00	2.00	1.33	1.00	1.00	0.83	0.71	0.75
Virginia Angus	3.00	1.50	1.00	0.75	0.60	0.50	0.43	0.38
Akali	3.00	2.00	2.33	2.25	1.80	1.67	1.43	1.25
Burger Yiyeli	3.00	2.00	1.33	1.25	1.00	0.83	0.86	0.75
Çukurcuma	3.00	1.50	1.00	0.75	0.60	0.50	0.43	0.38
Mendels	3.00	3.00	2.00	1.50	1.40	1.17	1.14	1.00
Fornello Piz.	3.00	1.50	1.33	1.00	1.20	1.00	1.00	0.88
Varuna Gez.	3.00	2.00	1.33	1.00	0.80	0.67	0.57	0.50
Metet Közde	3.00	2.00	1.33	1.00	0.80	0.67	0.57	0.50
MOC Istanbul	2.00	1.50	1.00	1.25	1.00	0.83	0.71	0.63
Walters Coffee	2.00	1.00	1.00	1.00	0.80	0.67	0.71	0.63
Bunco	2.00	1.00	0.67	0.75	0.60	0.50	0.43	0.38
Beyaz Firin	2.00	1.00	1.00	0.75	0.80	0.67	0.57	0.50
Tatar Salim	2.00	1.00	1.00	1.50	1.20	1.17	1.00	1.00

Table 19. Recall Success Rates (%)

Focal Restaurant Name	Top 100	Top 200	Top 300	Top 400	Top 500	Top 600	Top 700	Top 800
Asuman	9.09	11.36	11.36	15.91	15.91	18.18	20.45	20.45
Somunarasi	28.57	28.57	28.57	28.57	35.71	35.71	35.71	42.86
Virginia Angus	15.79	15.79	15.79	15.79	15.79	15.79	15.79	15.79
Akali	6.98	9.30	16.28	20.93	20.93	23.26	23.26	23.26
Burger Yiyelim	13.64	18.18	18.18	22.73	22.73	22.73	27.27	27.27
Çukurcuma	37.50	37.50	37.50	37.50	37.50	37.50	37.50	37.50
Mendels	9.09	18.18	18.18	18.18	21.21	21.21	24.24	24.24
Fornello Piz	10.34	10.34	13.79	13.79	20.69	20.69	24.14	24.14
Varuna Gezgin	14.29	19.05	19.05	19.05	19.05	19.05	19.05	19.05
Metet Köz Döner	25.00	33.33	33.33	33.33	33.33	33.33	33.33	33.33
MOC Istanbul	22.22	33.33	33.33	55.56	55.56	55.56	55.56	55.56
Walters Coffee	11.76	11.76	17.65	23.53	23.53	23.53	29.41	29.41
Bunco	25.00	25.00	25.00	37.50	37.50	37.50	37.50	37.50
Beyaz Firin	40.00	40.00	60.00	60.00	80.00	80.00	80.00	80.00
Salim Döner	4.76	4.76	7.14	14.29	14.29	16.67	16.67	19.05

Table 20 shows the comparison of the precision and recall average success rates. There was a total of 300 focal restaurants and the average success rates of all the focal restaurants were taken. For example, according to Table 20 when top 500 users were recommended for each focal restaurant and success resuls calculated for each of them and then average success rate were calculated for the whole 300 focal restaurants and results was equal to 0.43 % for Precision metrics and 21.44 % for Recal metrics.

Table 20. Precision-Recall Average Success Rates (%) Comparison

Metrics	Avg Top 100	Avg Top 200	Avg Top 300	Avg Top 400	Avg Top 500	Avg Top 600	Avg Top 700	Avg Top 800
PRECISION	0.74	0.64	0.55	0.49	0.43	0.38	0.35	0.32
RECALL	7.76	13.02	16.92	19.8	21.44	22.9	24.01	25

Lastly, the recommended user count was extended to include the top 1000 providing the precision recall curve given below (Figure 29). According to Figure 29, Precision success rates was decreasing when recommended user count increased, otherwise Recall success rates were increasing when recommended user count increased.

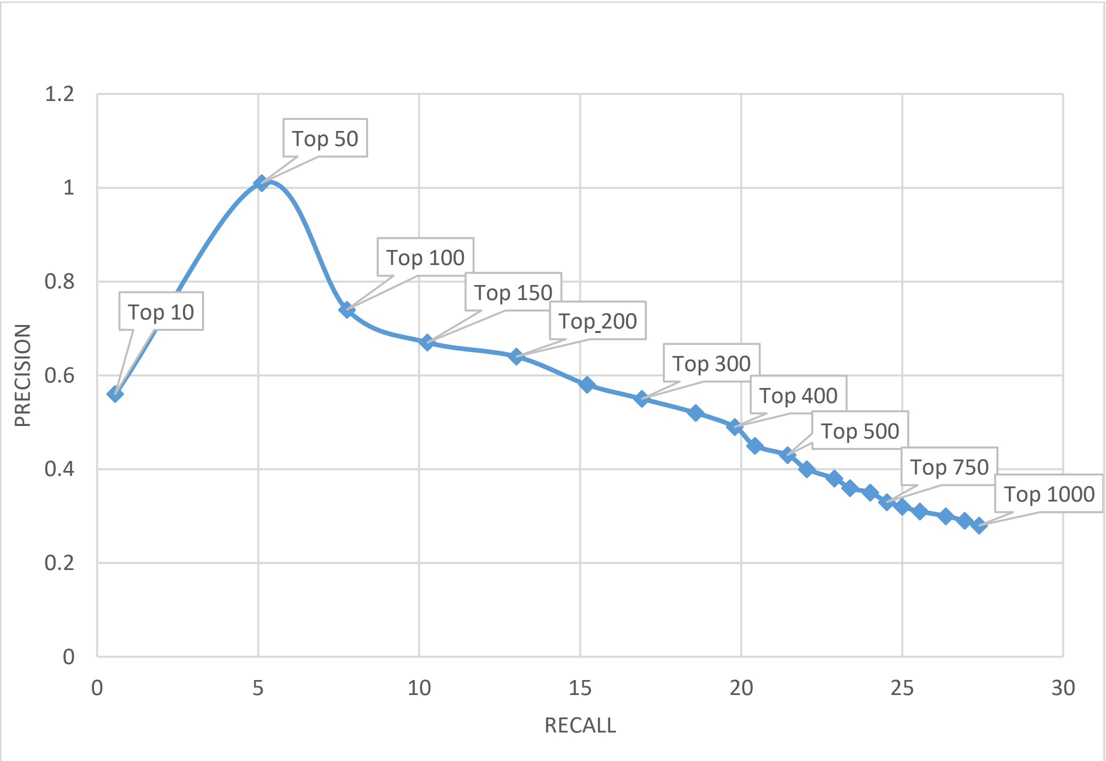

Figure 29 Senti framework precision-recall curve

3.6 Embeddings framework

The Embeddings Framework differs completely from the first two frameworks. The network structure, ranking and sorting algorithms of this framework are different than the others. Both of the previous networks used two networks, one being the user-user network, the other the restaurant-restaurant network. In both networks all nodes were either users or restaurants since it was an implicit network. Implicit networks comes from implicit relations between entities and the previous framework recommendations were suggested by power of this implicit relations. However, the Embeddings Framework did not use implicit networks, instead used explicit network. Explicit network is a cross-entity network, which nodes of one explicit network consists of both restaurants and users. So, with this direct relation between restaurants and users target users for the focal restaurants were predicted. The Embeddings framework has three phases.

Embeddings Framework outline:

1. Data pre-processing that consists of turning all review data into useful condition and language.
2. Undirected Restaurant-User explicit network generation and normalization
3. Node2vec Embeddings
4. Embeddings Restaurant to user similarity
5. Aggregated Target user selection with hyperparameters
6. Determining the best parameter values that yield the most successful results according to validation period goal users.
7. Selection of the target users based on the chosen hyperparameter values.
8. Success rate evaluation and analysis of the target users with goal users of the test data.

Figure 30 shows the overall Embeddings Framework which is embeddings approach recommendation system.

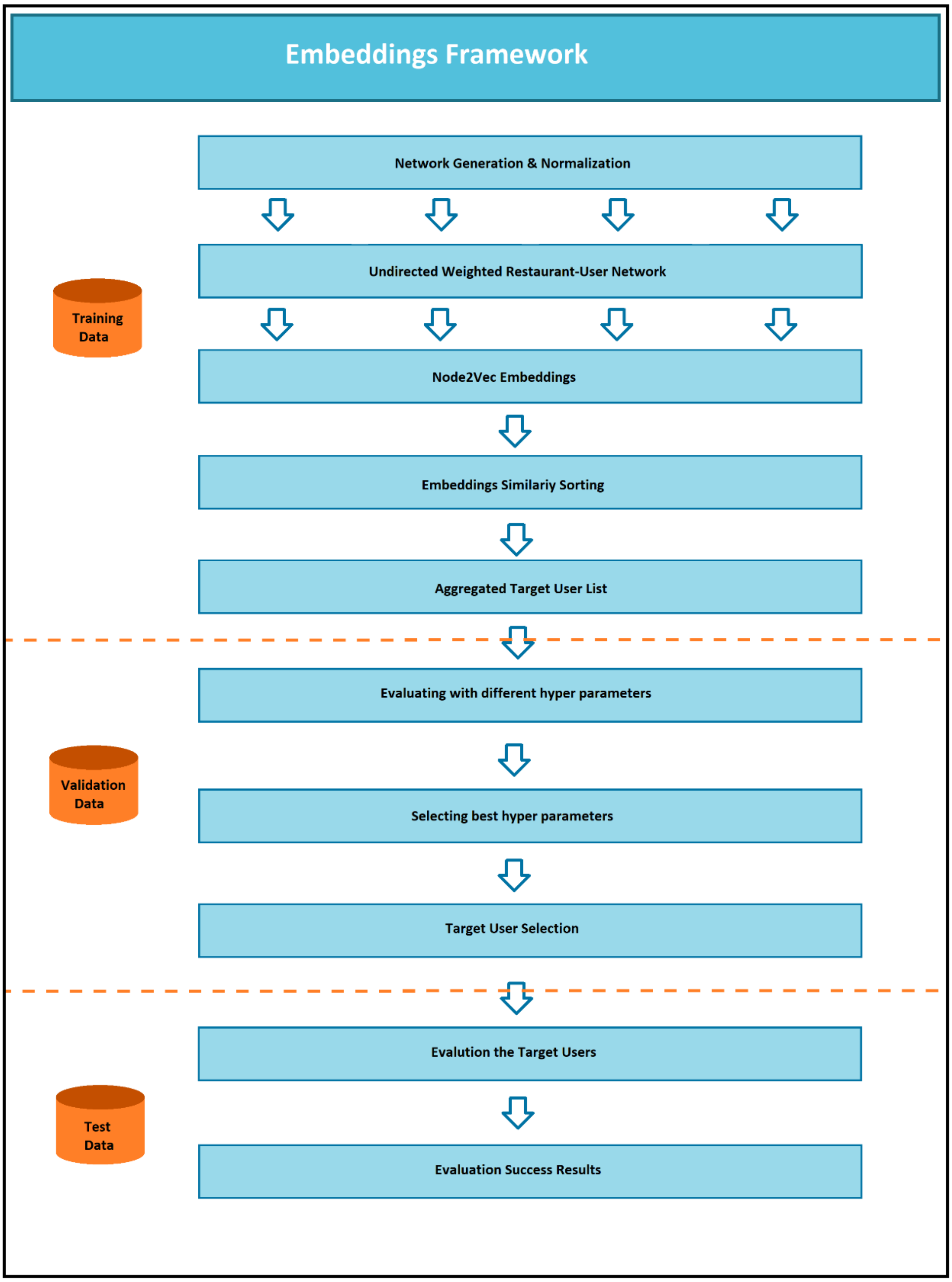

Figure 30 Embeddings framework flowchart

3.6.1 Networks

This framework utilized explicit networks. Explicit networks are formed by the direct interaction between user and restaurant. For example, when a user comments, likes or bookmarks a restaurant then there is a direct relation between the user and the restaurant. The more a user likes a restaurant the stronger the relation gets. The user review rate was taken as the relation weight. The same logic was applied to all restaurant and users subsequently generating the explicit restaurant–user network.

The aforementioned explicit restaurant user network can be seen in Figure 31. Community detection and PageRank algorithms were not applied in this network. The Restaurant-user explicit network is an undirected normalized explicit network.

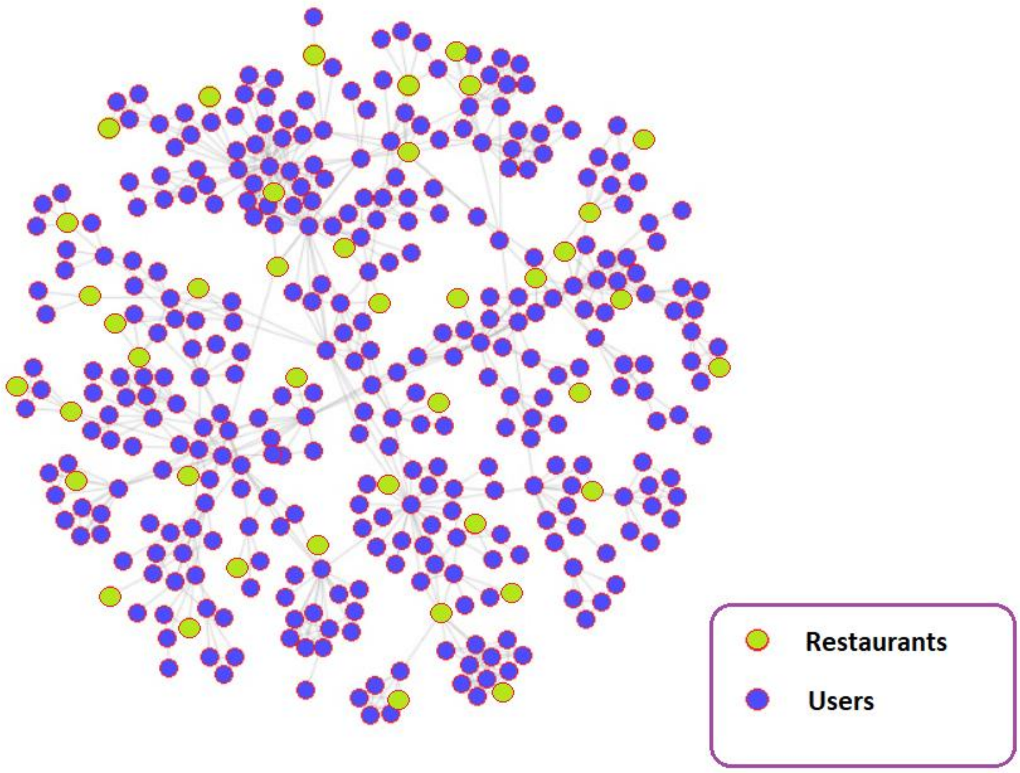

Figure 31 Explicit restaurant-user network

Having had a network with restaurants and users in it, a method needed to be found to extract the most related user nodes for any restaurant node to be able to recommend target users for that restaurant. Node2Vec embeddings algorithm was used to find the similarities of nodes. Since this network is an explicit restaurant-user network, nodes were flagged to determine whether it is a restaurant or user. Because the node2vec algorithm cannot identify whether a node is a user or restaurant, the first focal restaurant node was selected and only user nodes from similar nodes to the focal restaurant node were seleted.

3.6.2 Embeddings

By using the Node2vec algorithm Node2vec embeddings were generated for each node in the network. The Python implementation of the Node2vec algorithm was used.

Having generated embeddings, another Python program was developed to find the most similar top 1000 users for each focal restaurant according to node2vec similarity. Similar users were sorted by their embeddings and ranked. Table 21 shows the top 20 users for a focal restaurant.

3.6.3 Best hyperparameter selection

Special node2vec walk probability hyperparameters p and q were used when generating the node2vec embeddings. These parameters are node2vec model transition parameters that effect random walk probabilities. P controls the probability to go back to the previous node and q controls the probability to explore undiscovered nodes of the network. Using different hyperparameters changes the embedding vectors also changing the target user list. The following five hyperparameter pairs were used to calculate five different target user lists; (p,q)= { (1, 1), (0.25, 4), (4, 0.25), (0.5, 2), (2, 0.5) }.

The Node2vec embedding computation process is a long process and it takes two days to complete for a single run. Therefore, the five parameters were chosen carefully and the node2vec embeddings were run for them separately.

The success results of the hyperparameters were evaluated with the help of the validation data and the (0.25, 4) hyperparameter pair was found to yield the best results. This parameter pair was used for the success result evaluation in the test period.

3.6.4 Actual target users

Table 21 shows the actual and goal users of a focal restaurant. All of them are actual users. The users whose attribute “New User” is “NO” are the Goal Users, others are new users that are unpredictable users. All users commented on the focal restaurant in the test period, but only old users are available in both the test and the training period.

Table 21. Focal Resaurant Actual and Goal Users

Focal Restaurant	User Name	User ID	New User
Mangerie	Ahmet ****	54575336	YES
Mangerie	Asli ****	31940609	NO
Mangerie	Atalay ****	50861670	YES
Mangerie	Gokce ****	31034540	NO
Mangerie	Aylik ****	30779194	NO
Mangerie	Bahadir ****	39302192	NO
Mangerie	Bekir ****	40105364	NO
Mangerie	Buket ****	33364905	YES
Mangerie	Carlos ****	34292604	NO
Mangerie	Deniz ****	50953606	YES

3.6.5 Target user detection in test period

Table 22 shows the top 20 target user list that was recommended for the focal restaurant. They were sorted by their similarity vector value. Unfortunately, none of the users in the top 20 list commented on the focal restaurant in the test period. In the top 100 user selection list, there were two users who commented on the focal restaurant.

Table 22. Target User List for the Focal Restaurant

#	Restaurant Title	User Name	User ID	Similarity Vector
1	Mangerie	Kadir ****	3647941	0.32
2	Mangerie	Emre ****	41153394	0.30
3	Mangerie	Güliz ****	36984068	0.28
4	Mangerie	Yeliz ****	48112208	0.27
5	Mangerie	Dilara ****	32876327	0.26
6	Mangerie	Mustafa ****	38773264	0.26
7	Mangerie	Angirem ****	32869385	0.26
8	Mangerie	Duygu ****	34039755	0.25
9	Mangerie	Zeynep ****	40114408	0.25
10	Mangerie	Mehmet ****	29637448	0.25
11	Mangerie	Helin ****	34230918	0.24
12	Mangerie	Mnesnmz ****	33329429	0.24
13	Mangerie	Burak ****	38352544	0.24
14	Mangerie	Damla ****	33260781	0.24
15	Mangerie	Basak ****	18281269	0.24
16	Mangerie	Irem ****	38641914	0.24
17	Mangerie	Glnr ****	37946191	0.24
18	Mangerie	Cigdem ****	37070870	0.24
19	Mangerie	Ela ****	17371885	0.24
20	Mangerie	Yummyin ****	45868592	0.24

Table 23 shows the successful predictions within the top 100 target users recommended for the focal restaurant. Rank column shows the users rank in the Target user list. According to table, there were at least 2388 user recommendation needed to predict the user Dilara successfully. So, only 2 users would predict successfully if top 100 target user list were selected.

Table 23. Goal Users in the Target User List

Rank	Restaurant Title	User Name	User ID	Similarity Vector
77	Mangerie	Gun ****	39155661	0.1832
83	Mangerie	Carlos ****	34292604	0.1816
1213	Mangerie	Pervan ****	19056949	0.0715
2213	Mangerie	Asli ****	31940609	0.0442
2388	Mangerie	Dilara ****	1425309	0.0405
3557	Mangerie	Bekir ****	40105364	0.0196

3.6.6 Success results in test period

With using best hyperparameters success results were calculated. Table 24 shows the most successful top 20 focal restaurants in terms of correct recommendation. In Table 24 success rates were calculated by using Recall metrics. According to the table, the success count is equal to 1 for most of the focal restaurants, this means that only one of the goal users were predicted correctly. Also, the results were calculated by using the top 100 target user selection. There are same restaurants in the Table 24 which shows the different outlets of the same restaurant. According to test period results, Embeddings framework had the lowest success rates among the test period results of the all frameworks.

Table 24. Success Rates of Top 20 Focal Restaurants

Focal Restaurant	Actual User Count	Goal User Count	Success Count	Success Rate (%)
Meloon Coffee & Food	18	6	1	16.67
Tatbak	11	6	1	16.67
Zübeyir Ocakbasi	8	6	1	16.67
Mangerie	41	13	2	15.38
Cadiköy	26	7	1	14.29
Cookshop	12	7	1	14.29
Cookshop	20	7	1	14.29
Dobbys Burger Place	15	7	1	14.29
Kropka Coffee	14	7	1	14.29
Kydonia	11	7	1	14.29
Serez Dondurmacisi	12	7	1	14.29
Taslihan Restaurant	16	7	1	14.29
Yeniköy Kahvesi	14	7	1	14.29
Cafe Esmer Chef	68	8	1	12.50
Chef Mezze	32	8	1	12.50
Hane Çikolata & Kahve	28	8	1	12.50
Hüsnü Ala Cafe	30	8	1	12.50
Oba Restaurant	22	8	1	12.50
Thales Bistro	26	8	1	12.50
Frango	18	9	1	11.11

3.6.7 Precision – Recall results

Previous results are calculated according to Top 100 target user selection. Success rates were calculated according to the correct predictions within the top 100 recommended user count. Besides this, many different recommended user count ranging from the top 10 to the top 1000 were recalculated.

Precision considers selected target user count whereas Recall considers goal user count when it comes to success rate calculations. So, increasing the target user count increases the Recall success rates but decreases the precision success rates.

Table 25 show the Precision type success rates with many different top target user counts. Focal restaurants are sorted by precision of the top 100 selection.

Table 25. Precision Success Rates (%)

Focal Restaurant Name	Top 100	Top 200	Top 300	Top 400	Top 500	Top 600	Top 700	Top 800
Mangerie	2.00	1.00	0.67	0.50	0.40	0.33	0.29	0.25
Çesme Bazlama	2.00	1.50	1.00	0.75	0.80	0.83	0.71	0.63
Brasserie Noir	1.00	0.50	0.33	0.25	0.20	0.17	0.14	0.13
Happy Moons	1.00	0.50	0.33	0.25	0.20	0.33	0.29	0.25
Midpoint	1.00	0.50	0.33	0.25	0.20	0.17	0.14	0.13
Kireçburnu Firini	1.00	0.50	0.67	0.50	0.40	0.33	0.29	0.25
Basta! Street Food	1.00	0.50	0.33	0.50	0.40	0.33	0.29	0.25
Dardenia Fish	1.00	0.50	0.33	0.25	0.20	0.17	0.14	0.13
Borsam Tasfirin	1.00	1.00	0.67	0.50	0.40	0.33	0.43	0.38
Burger Yiyelim	1.00	1.00	0.67	0.50	0.40	0.50	0.43	0.38
Ozzies Kokoreç	1.00	0.50	0.33	0.25	0.20	0.17	0.14	0.13
Zübeyir Ocakbasi	1.00	0.50	0.33	0.25	0.20	0.17	0.14	0.13
Tuzla Balikçisi	1.00	0.50	0.33	0.25	0.40	0.33	0.29	0.25
Zuma	1.00	0.50	0.67	0.75	0.60	0.50	0.43	0.38
Hüsnü Ala Cafe	1.00	0.50	0.33	0.50	0.60	0.50	0.57	0.50

Table 26 shows the Recall metrics success rates results for some of the focal restaurants. Table 26 also shows the success results for many different user recommendation count which ranged from top 100 to top 800. Same restaurant with Table 25 were used in the Table 26.

Table 26. Recall Success Rates (%)

Focal Restaurant Name	Top 100	Top 200	Top 300	Top 400	Top 500	Top 600	Top 700	Top 800
Mangerie	15.38	15.38	15.38	15.38	15.38	15.38	15.38	15.38
Çesme Bazlama	4.88	7.32	7.32	7.32	9.76	12.20	12.20	12.20
Brasserie Noir	10.00	10.00	10.00	10.00	10.00	10.00	10.00	10.00
Happy Moons	6.25	6.25	6.25	6.25	6.25	12.50	12.50	12.50
Midpoint	5.26	5.26	5.26	5.26	5.26	5.26	5.26	5.26
Kireçburnu Firini	10.00	10.00	20.00	20.00	20.00	20.00	20.00	20.00
Basta! Street Food	5.00	5.00	5.00	10.00	10.00	10.00	10.00	10.00
Dardenia Fish	9.09	9.09	9.09	9.09	9.09	9.09	9.09	9.09
Borsam Tasfirin	8.33	16.67	16.67	16.67	16.67	16.67	25.00	25.00
Burger Yiyelim	4.55	9.09	9.09	9.09	9.09	13.64	13.64	13.64
Ozzies Kokoreç	6.67	6.67	6.67	6.67	6.67	6.67	6.67	6.67
Zübeyir Ocakbasi	16.67	16.67	16.67	16.67	16.67	16.67	16.67	16.67
Tuzla Balikçisi	4.76	4.76	4.76	4.76	9.52	9.52	9.52	9.52
Zuma	8.33	8.33	16.67	25.00	25.00	25.00	25.00	25.00
Hüsnü Ala Cafe	12.50	12.50	12.50	25.00	37.50	37.50	50.00	50.00

The precision recall results in Table 25 and 26 show the success rates for each focal restaurant. The combination of all 300 focal restaurants were calculated and the average success rates of the all focal restaurants was taken. Table 27 shows the average success rates results of precision and recall together.

Table 27. Precision-Recall Average Success Rates (%) Comparison

Metrics	Avg Top 100	Avg Top 200	Avg Top 300	Avg Top 400	Avg Top 500	Avg Top 600	Avg Top 700	Avg Top 800
PRECISION	0.10	0.10	0.10	0.10	0.10	0.10	0.09	0.09
RECALL	0.82	1.85	2.96	3.83	4.65	5.53	5.98	6.60

Lastly Figure 32 gives the precision recall curve for all different target user selections. Precision-Recall curve was drawn by using average success rates for all restaurants according to different top N recoomendation target user selection. According to Figure 32, Embeddings framework had the worst success rates among all frameworks.

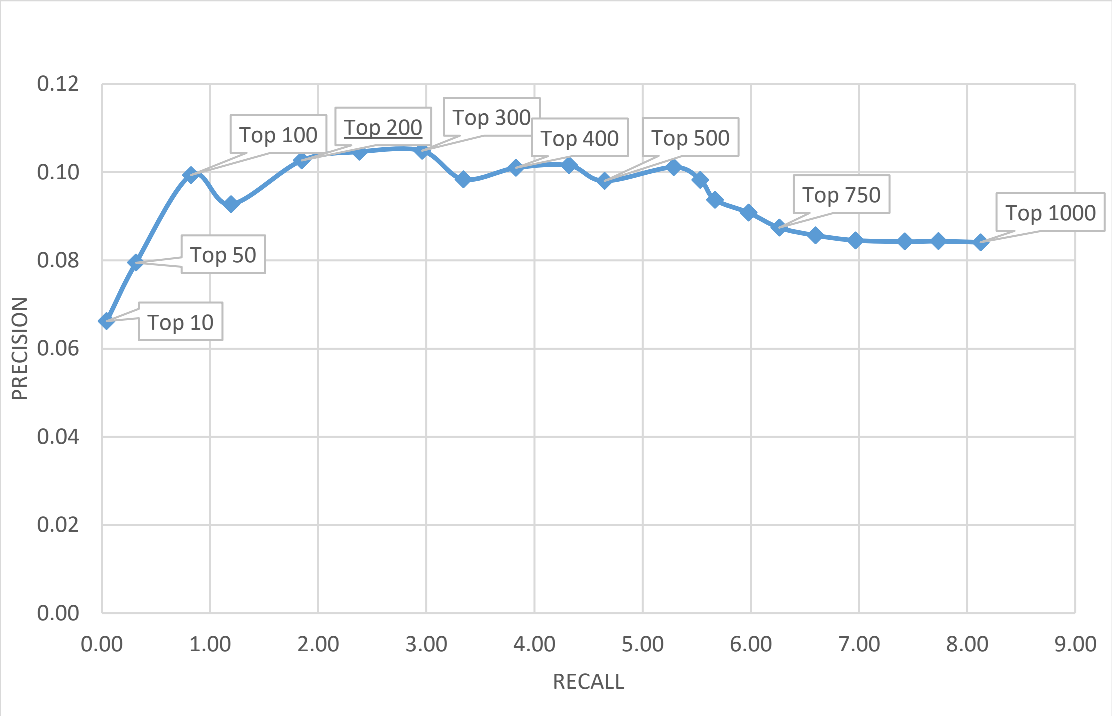

Figure 32 Embeddings framework precision-recall curve

CHAPTER 4

RESULTS AND FINDINGS

As explained before there were two types of success rates calculated, Recall and Precision type success rates. Recall type success rate is the ratio between goal user count and successfully predicted user count for any given focal restaurant. Precision type success rate is the ratio between recommended target user count and successfully predicted user count for any given focal restaurant. Recommended target user list was selected with different counts and it ranged from Top 10 to Top 1000 target user count. Goal target users were the users who actually reviewed the focal restaurants only in the validation/test period and who were also available in the training period. For a focal restaurant, all users who really reviewed the focal restaurant after the training periods could contain new users whose first review is available after the training periods for any restaurants thus these new users were excluded while success rates were calculated since there were no chance to predict them. Users who reviewed the focal restaurants in both the training period and after training periods were also excluded since they were already available. The overall purpose was to correctly predict as many users to recommend to a focal restaurant. So, target users should review the focal restaurant in the test period while being present but not having reviewed the focal restaurant in the training period.

Table 28 shows the comparison of the success rates of the three frameworks by their Top 100 and Top 1000 target user selection. In the table, focal restaurants are sorted by their success rates of the Top 1000 target user selection. Also, 10 of each successful focal restaurants of every frameworks are selected.

Table 28. Three Frameworks Success Rates (%) Comparison

Focal Restaurant Title	Top 100 Recommendation Ranks Framework	Senti Framework	Embedding Framework	Top 1000 Recommendation Ranks Framework	Senti Framework	Embedding Framework
Moda Aile Çay	0.00	0.00	0.00	100.00	50.00	0.00
Brew Coffee	16.67	16.67	0.00	83.33	83.33	33.33
Beyaz Firin	40.00	40.00	0.00	80.00	80.00	0.00
Must Nisantasi	25.00	0.00	0.00	75.00	75.00	0.00
Cookshop	0.00	0.00	14.29	28.57	14.29	71.43
Leman Kültür	10.00	10.00	0.00	30.00	70.00	0.00
Sekerci Cafer	30.00	10.00	0.00	70.00	60.00	10.00
MOC Istanbul	22.22	22.22	0.00	66.67	55.56	11.11
Yirmibir Kebap	33.33	33.33	0.00	33.33	66.67	0.00
Safran Pub	33.33	33.33	0.00	66.67	50.00	16.67
Kruvasan	33.33	11.11	0.00	66.67	33.33	0.00
Cookshop	22.22	11.11	0.00	66.67	33.33	11.11
Seraf Rest	9.09	18.18	0.00	63.64	45.45	0.00
Daily Dana	40.00	20.00	0.00	60.00	60.00	0.00
Baltepe Pasta	20.00	20.00	0.00	60.00	20.00	0.00
Bira Fabrikasi	28.57	14.29	0.00	57.14	57.14	0.00
Kydonia	0.00	0.00	14.29	57.14	14.29	28.57
P.F. Changs	0.00	0.00	11.11	55.56	22.22	33.33
Mums Cafe	7.69	15.38	0.00	53.85	46.15	0.00
Tatbak	33.33	0.00	16.67	33.33	50.00	16.67
Zübeyir Ocak	16.67	33.33	16.67	50.00	33.33	16.67
ANY	12.50	12.50	0.00	37.50	50.00	0.00
Hüsnü Ala Caf	0.00	0.00	12.50	37.50	25.00	50.00
Baylan	37.50	12.50	0.00	50.00	50.00	12.50
MOC Istanbul	15.38	7.69	7.69	46.15	30.77	15.38
Mangerie	15.38	7.69	15.38	30.77	38.46	15.38
Meloon Coffe	0.00	0.00	16.67	16.67	16.67	33.33
Cookshop	0.00	0.00	14.29	28.57	14.29	28.57
Elbet Steak	0.00	0.00	7.69	23.08	7.69	23.08

Figure 33 shows the precision-recall curves for all the frameworks combined. Orange is Ranks framework, grey is Senti framework and blue is Embeddings Framework curve. All the poings on the lines show the top N user selection count which started from Top 10 to Top 1000. According to the figure, Ranks Framework provides the best solution.

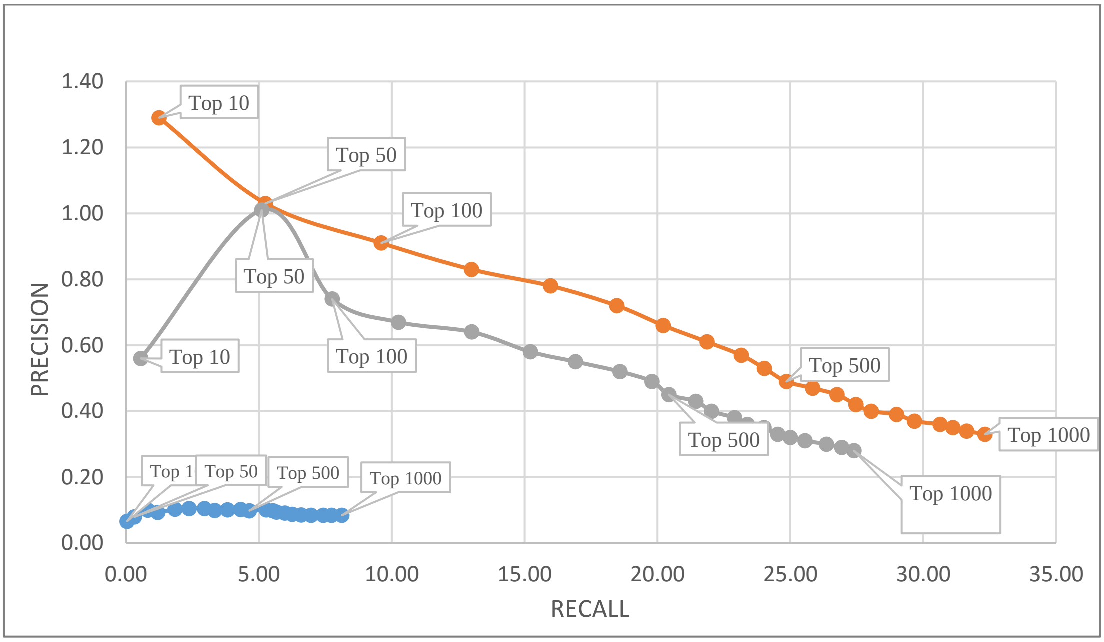

Figure 33 Precision recall curve for 3 frameworks

In the thesis, it is found out that PageRank is more important than CommunityRank according to α / β hyperparameter results in the validation period. User PageRank values are calculated globally and by using implicit networks. CommunityRank prediction is based on an assumption that if a user is in the same community with focal restaurant users then is likely to go that focal restaurant in the future. Also, the reason why CommunityRank is less important than PageRank in the Ranks framework was the network size, distribution of community users and communities count. For some focal restaurants, community detection only gives a few communities with very high user counts in them such as 10.000 users in one community and 15.000 users in another community. This causes the only two different CommunityRank values for all users which reduces the selection chance of any users in one of the communities. However, if there are hundreds of user communities with just hundreds of users homogeneously distributed in them, then this helps the system to select different users in different communities and will end up with very successful results.

So, the results show us that PageRank connection of users is more important than community connection in terms of future predictions in the framework.

According to the test result, it can be stated that Ranks framework is better than Senti framework and Embeddings Framework by their success rates. Ranks framework uses PageRank and CommunityRank combination for user selection but Senti framework uses SentiRank of users for user selection, and Embeddings Framework uses the explicit node2vec similarities for target user selection.

SentiRank is formed by using content and meaning of the users' reviews thus the users whose SentiRank is similar to each other can give a similar comment to the same restaurant according to the meaning of review content but cannot show whether or not both users give any comment. SentiRank is more related to meaning similarities of user comments than existence of user comments. SentiRank also shows us whether the user who was recommended to a focal restaurant will like the restaurant with a high chance.

PageRank and CommunityRank values are formed by using implicit networks of users which are generated by user comments that are given to the same restaurant. This rank is more related to the existence of a comment than the meaning of comments. So, PageRank and CommunityRank are more precise indicators of target users of a restaurant than SentiRank. Findings of both frameworks support this claim.

The Embeddings Framework is the worst among the three frameworks since it was formed by using explicit relations between restaurants and users. The Embeddings Framework is more dependent on direct relations of the user and restaurants, so this makes it more difficult to find indirect connections between entities.

According to customer purpose different framework can be used. If a restaurant wants to be recommended the users who will really like the food, then Senti framework can be used. However, if the customer care about the users who will come to the restaurant regardless of liking the food then Ranks framework should be used.

CHAPTER 5

LIMITATIONS

There were some limitations that may have prevented more precise and accurate results.

Firstly, the sparse and very few goal user count affects the results. There was only 400.000 user reviews for Istanbul in Zomato for 30.000 restaurants. Even if 5 million-edged networks were generated, it drops a few goal users per focal restaurant thus making it very difficult to predict these goal users. So, for a good prediction there should be 500-1000 average reviews per restaurant and then larger and stronger network can be generated with this data that will provide the perfect data source for prediction.

We have only used the review data of Istanbul province downloaded from Zomato and using only Istanbul data was not enough for a good prediction. Even if Istanbul data wast not enough it was still one of the best cities in the world in terms of comments distribution per restaurant. Because we also checked the other countries like America, Germany, India and France for comment and restaurant counts and even if some of they have much more restaurants in their country they still fall behind the Istanbul in terms of user comments per restaurants.

The Zomato API did not provide users that only gave ratings without leaving a comment. It is generally time consuming and laborious to write down your opinions for every restaurant you visited or evaluated but it is easier to give a rating, so the existence of a user's rating data can be more valuable since it can show the behavior of average people who have little interest in reviewing restaurants.

In addition, people generally tend to not review or rate a restaurant that they visited, the reason behind is the amount of review data. Also, by its very nature reviewing restaurants on a daily basis in social networks is not preferable when compared with Facebook-like social media. Thus, a majority of user reviews which could be crucial in terms of recommendation may have been missed.

The problem of Cold start is that sometimes most of the actual target users can be new users for a focal restaurant and information for these users are not available in the training period lowering the success rate so much so that it almost drops 50% before starting the experiment.

CHAPTER 6

CONCLUSIONS AND FUTURE WORK

The aim of this study was to develop and test a system that can recommend customers to companies for targeted advertisement in the most effective and efficient way. There are three different frameworks developed and Ranks framework is the most successful framework among them according to prediction accuracy. The Ranks framework reached a maximum successful prediction rate of 50% and 9.61% on average when there are 100 target users suggested to restaurant. Senti framework reached a maximum of 40% and on average 7.76% success rates. The Embeddings framework is the worst framework among them and only reached maximum success rate of 15% and 0.82% on average.

There are several implications of this system. On the business side; using this system, companies can correctly identify target audiences and lower their advertisement costs, thus increasing their overall profit margins. On the customer side; the increased usage of such a system by companies will help clear out “noisy advertisements” for customers that are not interested in a product/service. Customers will also become aware of more specific items that fit their needs. This, in turn, will increase the level of customer satisfaction and consumption.

As a future work, there are many user and restaurant data in the data source that were not used. For example; restaurant average rates, cuisine, location, average cost, restaurant photo likes, restaurant extra features, user photos, user followers, user review replies, user bookmarks, user places and user Zomato rates. This information can be analyzed, cleansed and used with different frameworks according to purpose.

Also, these frameworks can be used in different subjects with different datasets, such as Facebook, Instagram or YouTube data to recommend users to online brands.

REFERENCES

Abdrabbah, S. B., Ayachi, R., & Amor, N. B. (2014). Collaborative filtering based on dynamic community detection. *Dynamic Networks and Knowledge Discovery*, 85.

Alotaibi, S., & Vassileva, J. (2016). Personalized Recommendation of Research Papers by Fusing Recommendations from Explicit and Implicit Social Network. *UMAP (Extended Proceedings).*

Anderson, M., & Magruder, J. (2012). Learning from the crowd: Regression discontinuity estimates of the effects of an online review database. *The Economic Journal, 122*(563), 957-989.

Ben Yahia, N., Bellamine Ben Saoud, N., & Ben Ghezala, H. (2014). Community-based collaboration recommendation to support mixed decision-making support. *Journal of Decision Systems, 3*(23), 350-371.

Bhatia, V., & Hasija, V. (2016, July). Targeted advertising using behavioural data and social data mining. *2016 Eighth International Conference on Ubiquitous and Future Networks (ICUFN)* (pp. 937-942). IEEE.

Bimpikis, K., Ozdaglar, A., & Yildiz, E. (2016). Competitive targeted advertising over networks. *Operations Research, 3*(64), pp. 705-720.

Bobadilla, J. E., Serradilla, F., & Hernando, A. (2009). Collaborative filtering adapted to recommender systems of e-learning. *Knowledge-Based Systems, 4*(22), pp. 261-265.

Bozanta, A., & Kutlu, B. (2018). Developing A Contextually Personalized Hybrid Recommender System. *Mobile Information Systems*, pp. 1-13. doi:https://doi.org/10.1155/2018/3258916

Bozanta, A., & Kutlu, B. (2019). HybRecSys: Content-based contextual hybrid venue recommender system. *Online First in Journal of Information Science*, pp. 1-15.

Burke, R. (2002). Hybrid recommender systems: Survey and experiments. *User modeling and user-adapted interaction, 4*(12), pp. 331-370.

Burke, R. (2007). Hybrid web recommender systems. *The adaptive web*, pp. 377-408.

Chen, J., Tang, Y., Li, J., Mao, C., & Xiao, J. (2013, October). Community-based scholar recommendation modeling in academic social network sites. *International Conference on Web Information Systems Engineering* (pp. 325-334). Berlin, Heidelberg: Springer.

Chen, J., Wu, Y., Fan, L., Lin, X., Zheng, H., Yu, S., & Xuan, Q. (2017, December). Improved spectral clustering collaborative filtering with Node2vec technology. *2017 International Workshop on Complex Systems and Networks (IWCSN)* (pp. 330-334). IEEE.

Chen, L., & Pu, P. (2007, January). Hybrid critiquing-based recommender systems. *Proceedings of the 12th international conference on Intelligent user interfaces* (pp. 22-31). ACM.

Chickering, D. M., & Heckerman, D. (2003). Targeted advertising on the web with inventory management. *Interfaces*, pp. 71-77.

Christakou, C., Vrettos, S., & Stafylopatis, A. (2007). A hybrid movie recommender system based on neural networks. *International Journal on Artificial Intelligence Tools, 5*(16), 771-792.

De Campos, L. M., Fernández-Luna, J. M., Huete, J. F., & Rueda-Morales, M. A. (2010). Combining content-based and collaborative recommendations: A hybrid approach based on Bayesian networks. *International Journal of Approximate Reasoning, 7*(51), 785-799.

De Gemmis, M., Lops, P., Musto, C., Narducci, F., & Semeraro, G. (2015). Semantics-aware content-based recommender systems. In *Recommender Systems Handbook* (pp. 119-159). Boston, MA: Springer.

De Meo, P., Ferrara, E., Fiumara, G., & Provetti, A. (2011, November). Generalized louvain method for community detection in large networks. *2011 11th International Conference on Intelligent Systems Design and Applications* (pp. 88-93). IEEE.

Di Noia, T., Mirizzi, R., Ostuni, V. C., Romito, D., & Zanker, M. (2012, September). Linked open data to support content-based recommender systems. *Proceedings of the 8th international conference on semantic systems* (pp. 1-8). ACM.

Ekstrand, M. D., Riedl, J. T., & Konstan, J. A. (2011). Collaborative filtering recommender systems. *Foundations and Trends® in Human–Computer Interaction, 2*(4), pp. 81-173.

Ertugrul, A. M., Onal, I., & Acarturk, C. (2017, June). Does the strength of sentiment matter? A regression based approach on Turkish social media. *International conference on applications of natural language to information systems* (pp. 149-155). Cham: Springer.

Farahat, A., & Bailey, M. C. (2012, April). How effective is targeted advertising? *Proceedings of the 21st international conference on World Wide Web* (pp. 111-120). ACM.

Fatemi, M., & Tokarchuk, L. (2013, September). A community based social recommender system for individuals & groups. *2013 International Conference on Social Computing* (pp. 351-356). IEEE.

Franceschet, M. (2010). PageRank: Standing on the shoulders of giants. arXiv preprint arXiv:1002.2858.

Grover, A., & Leskovec, J. (2016, August). node2vec: Scalable feature learning for networks. *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining* (pp. 855-864). ACM.

Gurini, D. F., Gasparetti, F., Micarelli, A., & Sansonetti, G. (2013). A Sentiment-Based Approach to Twitter User Recommendation. 1066. RSWeb@ RecSys.

Herlocker, J. L., Konstan, J. A., Terveen, L. G., & Riedl, J. T. (2004). Evaluating collaborative filtering recommender systems. *Transactions on Information Systems (TOIS), 1*(22), pp. 5-53.

Huang, Z., Chung, W., Ong, T. H., & Chen, H. (2002, July). A graph-based recommender system for digital library. *Proceedings of the 2nd ACM/IEEE-CS joint conference on Digital libraries* (pp. 65-73). ACM.

Jiang, F., & Wang, Z. (2010, October). Pagerank-based collaborative filtering recommendation. *International Conference on Information Computing and Applications* (pp. 597-604). Berlin, Heidelberg: Springer.

Kandiah, V., & Shepelyansky, D. L. (2012). PageRank model of opinion formation on social networks. *Physica A: Statistical Mechanics and its Applications, 22*(391), pp. 5779-5793.

Kardan, A. A., & Hooman, M. (2013, April). Targeted advertisement in social networks using recommender systems. *7th International Conference on e-Commerce in Developing Countries: with focus on e-Security* (pp. 1-13). IEEE.

Lalwani, D., Somayajulu, D. V., & Krishna, P. R. (2015, October). A community driven social recommendation system. *2015 IEEE International Conference on Big Data (Big Data)(* (pp. 821-826). IEEE.

Lee, D. H., & Brusilovsky, P. (2017). Improving personalized recommendations using community membership information. *Information Processing & Management, 5*(53), pp. 1201-1214.

Leung, C. W., Chan, S. C., & Chung, F. L. (2006, August). Integrating collaborative filtering and sentiment analysis: A rating inference approach. *Proceedings of the ECAI 2006 workshop on recommender systems*, (pp. 62-66).

Leung, C. W., Chan, S. C., & Chung, F. L. (2006, August). Integrating collaborative filtering and sentiment analysis: A rating inference approach. *Proceedings of the ECAI 2006 workshop on recommender systems*, (pp. 62-66).

Levi, A., Mokryn, O., Diot, C., & Taft, N. (2012, September). Finding a needle in a haystack of reviews: cold start context-based hotel recommender system. *Proceedings of the sixth ACM conference on Recommender systems* (pp. 115-122). ACM.

Li, H., Cui, J., Shen, B., & Ma, J. (2016). An intelligent movie recommendation system through group-level sentiment analysis in microblogs. *Neurocomputing*(210), pp. 164-173.

Liao, L., He, X., Zhang, H., & Chua, T. S. (2018). Attributed social network embedding. *IEEE Transactions on Knowledge and Data Engineering, 12*(30), pp. 2257-2270.

Lops, P., de Gemmis, M., & Semeraro, G. (2011). Content-based Recommender Systems: State of the Art and Trends. In F. Ricci, L. Rokach, B. Shapira, & P. Kantor (Eds.), *Recommender Systems Handbook.* Boston, MA: Springer.

Luca, M. (2016). Reviews, reputation, and revenue: The case of Yelp. com.

Ma, H., King, I., & Lyu, M. R. (2011). Learning to recommend with explicit and implicit social relations. *ACM Transactions on Intelligent Systems and Technology (TIST), 3*(2), p. 29.

Mahata, A., Saini, N., Saharawat, S., & Tiwari, R. (2016, December). Intelligent Movie Recommender System Using Machine Learning. *International Conference on Intelligent Human Computer Interaction* (pp. 94-110). Springer, Cham.

Meteren, R., & Van Someren, M. (2000, May). Using content-based filtering for recommendation. *Proceedings of the Machine Learning in the New Information Age: MLnet/ECML2000 Workshop*, (pp. 47-56).

Mitra, P., & Baid, K. (2009, July). Targeted advertising for online social networks. *2009 First International Conference on Networked Digital Technologies* (pp. 366-372). IEEE.

Moraga-González, J. L., & Galeotti, A. (2004). A model of strategic targeted advertising.

Nasukawa, T., & Yi, J. (2003, October). Sentiment analysis: Capturing favorability using natural language processing. *Proceedings of the 2nd international conference on Knowledge capture* (pp. 70-77). ACM.

Palumbo, E., Rizzo, G., Troncy, R., Baralis, E., Osella, M., & Ferro, E. (2018, June). Knowledge graph embeddings with node2vec for item recommendation. *European Semantic Web Conference* (pp. 117-120). Cham: Springer.

Pang, B., & Lee, L. (2008). Opinion mining and sentiment analysis. *Foundations and Trends® in Information Retrieval, 1-2*(2), pp. 1-135.

Pappas, N., & Popescu-Belis, A. (2013, July). Sentiment analysis of user comments for one-class collaborative filtering over ted talks. *Proceedings of the 36th international ACM SIGIR conference on Research and development in information retrieval* (pp. 773-776). ACM.

Parimi, R., & Caragea, D. (2014, December). Community detection on large graph datasets for recommender systems. *2014 IEEE International Conference on Data Mining Workshop* (pp. 589-596). IEEE.

Pazzani, M. J., & Billsus, D. (2007). Content-based recommendation systems. *The adaptive web*, pp. 325-341.

Raman, P. (2018). Zomato: a shining armour in the foodtech sector. *Journal of Information Technology Case and Application Research*, 1-21.
Raman, P. (2018). Zomato: a shining armour in the foodtech sector. *Journal of Information Technology Case and Application Research*, 1-21.
Rohani, V. A., Kasirun, Z. M., & Ratnavelu, K. (2014, December). An enhanced content-based recommender system for academic social networks. *2014 IEEE Fourth International Conference on Big Data and Cloud Computing* (pp. 424-431). IEEE.

Salehan, M., & Kim, D. J. (2016). Predicting the performance of online consumer reviews: A sentiment mining approach to big data analytics. *Decision Support Systems*(81), pp. 30-40.

Sarwar, B. M., Karypis, G., Konstan, J. A., & Riedl, J. (2001). Item-based collaborative filtering recommendation algorithms. *Www*(1), pp. 285-295.

Schafer, J. B., Frankowski, D., Herlocker, J., & Sen, S. (2007). Collaborative filtering recommender systems. *The adaptive web*, pp. 291-324.

Singh, V. K., Mukherjee, M., & Mehta, G. K. (2011, December). Combining collaborative filtering and sentiment classification for improved movie recommendations. *International Workshop on Multi-disciplinary Trends in Artificial Intelligence* (pp. 38-50). Berlin, Heidelberg: Springer.

Singh, V. K., Piryani, R., Uddin, A., & Waila, P. (2013, March). Sentiment analysis of movie reviews: A new feature-based heuristic for aspect-level sentiment classification. *2013 International Mutli-Conference on Automation, Computing, Communication, Control and Compressed Sensing (iMac4s)* (pp. 712-717). IEEE.

Şora, I. (2015, May). A PageRank based recommender system for identifying key classes in software systems. *2015 IEEE 10th Jubilee International Symposium on Applied Computational Intelligence and Informatics* (pp. 495-500). IEEE.

Taghipour, N., & Kardan, A. (2008, March). A hybrid web recommender system based on q-learning. *Proceedings of the 2008 ACM symposium on Applied computing* (pp. 1164-1168). ACM.

Taneja, A., Gupta, P., Garg, A., Bansal, A., Grewal, K. P., & Arora, A. (2016, December). Social graph based location recommendation using users' behavior: By locating the best route and dining in best restaurant. *2016 fourth international conference on Parallel, distributed and grid computing (PDGC)* (pp. 488-494). IEEE.

Tran, T., & Cohen, R. (2000, July). Hybrid recommender systems for electronic commerce. *Proc. Knowledge-Based Electronic Markets, Papers from the AAAI Workshop Technical Report WS-00-04. 40.* AAAI Press.

Unit Working Paper No. 12-016. (n.d.). Retrieved from https://papers.ssrn.com/sol3/papers2.cfm?abstract_id=1928601

Vural, A. G., Cambazoglu, B. B., Senkul, P., & Tokgoz, Z. O. (2013). A framework for sentiment analysis in turkish: Application to polarity detection of movie reviews in turkish. *Computer and Information Sciences III*, pp. 437-445.

Wang, J., Liu, Z., & & Zhao, H. (2012). Group Recommendation Based on the PageRank. *JNW*, *12*(7), pp. 2019-2024.

Wang, W., & Wang, H. (2015). Opinion-enhanced collaborative filtering for recommender systems through sentiment analysis. *New Review of Hypermedia and Multimedia*, *3-4*(21), pp. 278-300.

Wen, Y., Guo, L., Chen, Z., & Ma, J. (2018, April). Network embedding based recommendation method in social networks. *Companion Proceedings of the The Web Conference 2018* (pp. 11-12). International World Wide Web Conferences Steering Committee.

Wu, Q., Tan, S., Zhai, H., Zhang, G., Duan, M., & Cheng, X. (2009, September). SentiRank: Cross-domain graph ranking for sentiment classification. *2009 IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology. 1*, pp. 309-314. IEEE.

Xiao, L., & Zhaoquan, G. (2017, October). Modeling Implicit Communities in Recommender Systems. *International Conference on Web Information Systems Engineering* (pp. 387-402). Springer, Cham.

Yang, D., Zhang, D., Yu, Z., & Wang, Z. (2013, May). A sentiment-enhanced personalized location recommendation system. *Proceedings of the 24th ACM conference on hypertext and social media* (pp. 119-128). ACM.

Yang, D., Zhang, D., Yu, Z., & Wang, Z. (2013, May). A sentiment-enhanced personalized location recommendation system. *Proceedings of the 24th ACM Conference on Hypertext and Social Media* (pp. 119-128). ACM.

Yang, W. S., Dia, J. B., Cheng, H. C., & Lin, H. T. (2006, January). Mining social networks for targeted advertising. *Proceedings of the 39th Annual Hawaii International Conference on System Sciences (HICSS'06). 6*, pp. 137a-137a. IEEE.

Zhang, D., Yin, J., Zhu, X., & Zhang, C. (2018). Network representation learning: A survey. *IEEE transactions on Big Data*.

Zhang, K., Bhattacharyya, S., & Ram, S. (2016). Large-Scale Network Analysis for Online Social Brand Advertising. *Mis Quarterly*, *4*(40).

Zhang, S., Salehan, M., Leung, A., Cabral, I., & Aghakhani, N. (2018). A Recommender System for Cultural Restaurants Based on Review Factors and Review Sentiment.

Zhang, S., Salehan, M., Leung, A., Cabral, I., & Aghakhani, N. (2018). A Recommender System for Cultural Restaurants Based on Review Factors and Review Sentiment.

Zhang, Z., Zhao, W., Yang, J., Nepal, S., Paris, C., & Li, B. (2017, October). Exploiting Users' Rating Behaviour to Enhance the Robustness of Social Recommendation. *International Conference on Web Information Systems Engineering* (pp. 467-475). Springer, Cham.

Zheng, Z., Ma, H., Lyu, M. R., & King, I. (2009, July). Wsrec: A collaborative filtering based web service recommender system. *2009 IEEE International Conference on Web Services* (pp. 437-444). IEEE.