https://doi.org/10.35662/unine-thesis-2849

Université de Neuchâtel

Faculté des sciences

Institut d'informatique

Dual Recommendation Analysis

par

Mehdy Davary

Thèse

Présentée à la Faculté des sciences
pour l'obtention du grade de Docteur ès Sciences - Informatique

Acceptée sur proposition du jury :

Prof. Jacque Savoy, co-directeur de thèse
Université de Neuchâtel, Suisse

Prof. Hatem Ghorbel, co-directeur de thèse
Haute Ecole Arc Ingénierie, St-Imier, Suisse

Prof. Elöd Egyed-Zsigmond, rapporteur
L'institut national des sciences appliquées de Lyon, France

Dr. Valerio Schiavoni, membre
Université de Neuchâtel, Suisse

Soutenue le 4 septembre 2020

UNIVERSITÉ DE NEUCHÂTEL

Faculté des Sciences
Secrétariat-décanat de Faculté
Rue Emile-Argand 11
2000 Neuchâtel – Suisse
Tél : + 41 (0)32 718 21 00
E-mail : secretariat.sciences@unine.ch

IMPRIMATUR POUR THESE DE DOCTORAT

La Faculté des sciences de l'Université de Neuchâtel autorise l'impression de la présente thèse soutenue par

Monsieur Mehdy DAVARY

Titre:

“Dual Recommendation Analysis”

sur le rapport des membres du jury composé comme suit:

- Prof. Jacques Savoy, directeur de thèse, Université de Neuchâtel, Suisse
- Prof. Hatem Ghorbel, co-directeur de thèse, HES-SO, Neuchâtel, Suisse
- Dr Valerio Schiavoni, Université de Neuchâtel, Suisse
- Dr Elöd Egyed-Zsigmond, INSA-Lyon, France

Neuchâtel, le 8 septembre 2020

Le Doyen, Prof. A. Bangerter

Acknowledgements

A special thanks to my family. Words cannot express how grateful I am to my mother, my father and my daughters Melody and Maya for all of the sacrifices that you have made on my behalf. Your prayer for me was what sustained me thus far. I would also like to thank all of my friends who supported me in writing and incented me to strive towards my goal. At the end I would like to express appreciation to my beloved wife Malihe who spent sleepless nights with and was always my support in the moments when there was no one to answer my queries.

I would like to acknowledge my sincere gratitude to my thesis director Professor Jacques Savoy of the University of Neuchatel, for the continuous support of my Ph.D. study and related research, for his patience, motivation, and knowledge. His guidance helped me in all the time of research and writing of this thesis. I could not have imagined having a better advisor and mentor for my Ph.D. study.

Besides my advisor, I would like to extend my special appreciation and thanks to Professor Hatem Ghorbel of HE-Arc, my thesis co-director, who guided me in the elaboration of this thesis. He gave me an opportunity to join their team as intern and provided me with the necessary access to their laboratory and research facilities. Without their precious support it would not be possible to conduct this research.

I would also like to thank my committee members, Professor Elöd Egyed-Zsigmond, and Dr. Valerio Schiavoni for serving as my committee members even at hardship. I also want to thank you for letting my defense be an enjoyable moment, and for your brilliant comments and suggestions, thanks to you.

Neuchâtel, 4 septembre 2020

Abstract

This thesis will investigate the recommendation systems in a radically different perspective. First, instead of focusing and proposing related products/services to customers, we will analyze and suggest recommendations for the producers. Such recommendations taking the form of product suggestions or adjustments will be defined according to a given geographical region and time span. We will also generate maps based on product profiles for a given time period and geographical region.

Second, the targeted analytic system will be able to generate a description of both the different product facets and the entire product based on the customers' reviews and thus fulfill a database describing the products available and their quality in a given time period and a geographical region. Moreover, being able to detect and measure the polarity of written opinions, the system can generate a map for a given time period and a geographical region, showing the most successful products/services. We can complete this map by indicating the degree of similarity between successful (or unsuccessful) products.

Once this information is obtained, the system can detect product opportunities and proposes a map of alternatives indicating where and when products that were successful in the past and in other similar regions might have a success in another region.

Third, based on the social network between customers, we will determine the strength of the relationships between customers and define their degree of leadership. Based on this information, we can weight more precisely the different customers' reports, assuming that reviews written by leaders will have a stronger impact than those written by followers. Moreover, after being able to identify the leaders, we could determine how they can improve their status or leadership.

Keywords

Recommendation systems; opinion and polarity detection and evaluation; natural language processing; similarity measures; social graph analysis.

Résumé

Cette thèse de doctorat examinera les systèmes de recommandation dans une perspective radicalement différente. Premièrement, au lieu de cibler et de proposer des produits / services connexes aux clients, nous analyserons et suggérerons des recommandations pour les producteurs. Ces recommandations prenant la forme de suggestions de produits ou d'ajustements seront définies en fonction d'une région géographique et pour une période donnée. Nous générerons également des cartes basées sur les profils de produits pour une période donnée et une région géographique.

Deuxièmement, le système analytique ciblé pourra générer une description des différentes facettes du produit et de l'ensemble du produit sur la base des avis des clients et remplir ainsi une base de données décrivant les produits disponibles et leur qualité dans une période et une zone géographique données. De plus, étant capable de détecter et de mesurer la polarité des opinions écrites, le système peut générer une carte pour une période donnée et une région géographique, montrant les produits / services les plus réussis. Nous pouvons compléter cette carte en indiquant le degré de similarité entre les produits réussis (ou mal réussis).

Une fois cette information obtenue, le système peut détecter les opportunités de produits et proposer une carte des alternatives indiquant où et quand les produits qui ont réussi dans le passé et dans d'autres régions similaires peuvent avoir du succès dans une autre région.

Troisièmement, sur la base du réseau social entre les clients, nous déterminerons la force de la relation entre les clients et définirons leur degré de leadership. Sur la base de ces informations, nous pouvons pondérer plus précisément les différents rapports des clients, en supposant que les avis rédigés par les influenceurs auront un impact plus fort que ceux écrits par les abonnés. De plus, après avoir été en mesure d'identifier les leaders, nous pourrions déterminer comment ils peuvent améliorer leur statut ou leur leadership.

Mots-clés

Les systèmes de recommandation ; la détection et l'évaluation d'opinion et de la polarité ; traitement du langage naturel ; mesures de similarité ; analyse des réseaux sociaux.

Contents

List of Figures

List of Tables

Chapter 1 Main Research Theme and Context

1.1 Introduction

The users have first used the Web as a medium to access information where commercial search engines (e.g., Yahoo, Google) have played the first role. With the Web 2.0, they have transformed this communication channel to promote their ideas or simply to share their findings, opinions, and sentiments. This tendency towards an open new social medium is primarily textual (blogs) but people might also share their photos (photoblog as in Flickr), videos (vlog as in YouTube), music (MP3 blog) and audio (podcasting) files. Sharing information also means sharing customers' reviews on products / services (e.g., restaurants, hotels, books, movies, events are the most well-known, but we can also find customers' comments on many other products such as watches, cars, (e.g., epinions.com). These reviews are becoming an important tool to promote brands and services. It is known that before booking a hotel (restaurant), future customers search and read reviews on these target items or services.

Customers' reviews are therefore an important marketing aspect and are one component of the planned research and my doctoral dissertation. Users tend to also communicate (more or less intensively) with friends and relatives within social networks (e.g., Facebook, Instagram) or by broadcasting their feelings or commenting on their activities (e.g., Twitter, Yelp). With such social networks, it is possible to establish not only links between persons but to determine the intensity of their relationship or to define the degree of leadership of the different members of a given community. Of course, such links and their intensity vary over time. In those social networks, the locality principle is rather effective, meaning that groups of people strongly connected tend to live in the same geographical region. Statistics also show that the average number of friends within social networks varies by age group.

The planned analytic system will integrate the customers' reviews with their social networks to define the importance of each report more precisely. This integration will also be the basis for a more effective analysis of product opportunity.

1.2 Social Networks

Most social networks are based on the Web 2.0 and their services are provided by the Internet-based applications. To date, social networking websites have been defined in many ways, however the definition that seems more accurate is that they are generally online Web-based services. A platform or website in which a social relationship is conducted by users and where individuals can post their own comments, interests, backgrounds, and relationships and contents, and share their interests with friends and others.

Each active user on a social networking website has his own personal profile, with a unique identification for each account to overcome the problem of privacy and has access to many other services.

Currently, Facebook, Twitter, LinkedIn, Google Plus, Yelp and Instagram are a few of the most popular social networking websites. The main goal of social networking websites is to make the user's interaction as simple and efficient as possible and to be an effective social change system in our society. Users can share freely their information either with their friends or by publishing them publicly, and without being obliged to accept communications from others.

With the introduction of social networking websites, a new kind of relationship was formed between people. The formation and development of online communities allows users to communicate faster, but also to access to both reliable and unreliable online resources and to share them internationally, beyond the physical borders. The propagation of news starts by spreading and exchange of news, opinions, thoughts, pictures, songs and videos by users, just around a personalized wall of information that is populated by their friends. These aspects are the most well-known functions of cyberspace.

Facebook with more than 2.3 billion monthly active users, Instagram with over 1 billion monthly active users, Twitter with more than 321 million monthly users, Yelp with more than 178 million unique visitors every month and other social networks have created a new form of

communication so that people can communicate and have access to information at any time of the day without being seen by each other.

Even though, social networking websites do the work of providing newsworthy information to the user, people may accidentally or purposely share information which are not truthful, but which seems to be relevant and important. According to an article written by Jayson DeMers on the forbes.com website, "*recent studies have shown that an estimated 59 percent of social media users will share information without actually reading an article but will share information based on the title alone.*"

1.3 Opinion and polarity detection

It is no secret that online customer reviews can be significant, but just how important are they to a business? In today's Internet driven world, customers have more power than ever. The number of consumers who read and trust online reviews is increasing, and several consumers trust online reviews as much as a personal recommendation.

When asked about their experience with the purchased product, customers offer an opinion, which must be considered by the vendor. That is why the customers' opinion is very important when it comes to the quality of customer service you provide. After-sales support is a service that is provided after products or services have been sold. Most after-sales support involves a guarantee, warranty, upgrade, or repair service.

There are some products which cannot be sold without an effective planning for services after sale. For example, customers will not buy a new car if the manufacturing company does not propose a solid after-sales service. In the same way, a municipality for example is not going to buy the parts for its power or water transmission and distribution networks without a reliable after-sales service and support provided for long-term by the supplier firms and/or by the manufacturing companies.

Companies often ask clients to communicate their opinion, because the customer feedback plays an important role in shaping company's business. The customer's opinion expressed through reviews is especially more important when a business offers products or services without an after-sales service to its customers.

As we saw at the beginning of this chapter, we have seen the rising adoption of Web 2.0 technologies, where refers to websites that emphasize participatory on user-generated content, ease of use, and compatiblility with other products, systems, and devices for end users. In parallel with the growth of social networks, the use of words and phrases of an informal register by their members is favored. With emails, chat groups, and text messaging which are new means of communicating on the social networking sites came new words, confusing acronyms, lots of jargon, emoticons, and emojis, etc. For example, LOL is an initialism for "laughing out loud" and is a popular element of Internet slang.

Linguists love this type of new form of language because it is a new form of writing. It is not oral. It is not writing. It is like an email. For example, when someone makes an email, he can afford to make spelling mistakes but when he writes a formal letter, he cannot afford to make spelling mistakes.

The Web 2.0. is where everyone can talk. What makes Umberto Eco[1] say:

"Social media gives legions of idiots the right to speak when they once only spoke at a bar on Sunday after a glass of wine, without harming the community. Then they were quickly silenced, but now they have the same right to speak as a Nobel Prize winner. It's the invasion of the idiots."

As in public relations, everyone can talk on the internet. Today's customers do not hesitate to write negative reviews online when they have a bad experience with a business. That is even truer when there is no customer service to complain to about their problem. In addition to leaving critical reviews, customers also express their frustrations on social media, so their friends, family, colleagues, and the entire world can know.

Customers who had a genuinely bad experience with a restaurant or hotel, like to talk more about it but prefer talking a little less about restaurants that are excellent. When they are satisfied with a restaurant, a hotel, or a book, they talk to 4 or 5 people, but if they are unhappy, they talk to 20, 25 or 30 people! An interesting investigation is to verify this by comparing the overall opinion polarity of a review with the number of stars given to the restaurant.

[1] https://en.wikipedia.org/wiki/Umberto_Eco

For most customers, the after-sales service is very important because they want to get the value they paid for from a product or service, particularly when things turn out wrong. A large number of businesses are aware that a grate customer service is essential for them to keep their customers satisfied and on-board. Moreover, as time passes by, customers are looking for businesses that offer better after-sales services, and they favor businesses that keep pace. According to the American Express[2] 2017 Customer Service Barometer:

- Half of U.S. consumers have abandoned a purchase due to a poor service experience.
- Two-thirds of shoppers say they will spend more money (17% more, on average) with a business that provides consistently great customer service.
- One third of customers say they would look to switch to a competitor after a single bad service experience.

While in most cases the after-sales service is very important for business success, there are also some businesses that perform successfully without needing a strong customer service. We can underline “Amazon” selling books and CDs online; “Liability insurance” (also called third-party insurance) that protects the insured in the event he is sued for claims that come within the coverage of the insurance policy; and “Property insurance” that offers protection against material damage to customers’ property resulting from storms, fire, water, glass breakage and theft.

For example, a restaurant owner may choose to do not provide any after-sales service to his consumers, and this decision will not dramatically impact his business and sales. The same way, a successful hotel that delivers excellent quality service to customers, and service quality is considered the life of the hotel, may be still successful without considering an after-sales support. These are some good examples of businesses that work well but do not need to provide a strong after-sales service to their customers.

We can also mention Apple’s services business[3] (including iTunes) with almost no need for a strong after-sales service. They brought in over $10.9 billion during the most recent quarter that is higher than Apple's other business segments providing strong after-sales services. For

[2] https://www.americanexpress.ch/
[3] https://www.apple.com/

example, Apple provides complete care of iPhone for a period of one year from the date of purchase which is included in the Apple care. However, the Mac ($7.4 billion), iPad ($6.7 billion), or the collected “Wearables, Home, and Accessories” group of products ($7.3 billion) are less successful compared to “Apple’s online services business’ income”.

1.4 Research Objectives

The overall purpose of this Ph.D. thesis and my research is to design, implement and evaluate an analytic system able to provide useful and pertinent recommendations, not to the customers, but to the producers. To achieve this main objective, such an analytic system must first be able to extract opinions about products from a set of customer reviews. Based on the social network of those customers, the system will then be capable of detecting and weighing the social relationships between customers. Following this perspective, the objectives of our research proposal are:

1. To design and implement a distance-based model that can effectively determine the distance (or the similarity) between products based on products descriptions and customers’ opinions.
2. To develop and implement an analytic system able to discover where and when a new product/service can be launched according to customers’ reports and expectations.
3. Based on a social network of customers, and after determining the district communities and leaders, how can we weight the customers’ reports to reflect the influence factor of each customer?
4. Can we detect the most important leaders in a community able to accept and to make acceptable a new product/service in a given time period and geographical region?

As success criteria, we will evaluate the effectiveness of the fully automatic analytic tool by letting the system have access to two sources of information (customers’ reviews & the social network) until a predefined point in the past. From this training part, we can evaluate the product descriptions generated by the system by comparing them with the descriptions generated manually. The opinion and polarity detections can be evaluated using test collections (e.g., movie reviews).

1.4.1 Objective 1: To design and implement a distance-based model

The design and implementation of a similarity (or distance) measure to define the similarities between products. Such a measure must first be based on the facets and their values. Of course, each facet may have its own importance and thus having different weights. As a second important aspect, we need to consider the customer's reviews. The underlying problem is to define the most appropriate mix between these two components in a distance measure. Based on this measure, we can then cluster the different products to define homogeneous groups. The geographical information must be taken into account in this clustering process.

1.4.2 Objective 2: To develop and implement an analytic system

The primary goal of this objective is to investigate different methods of sentiment analysis and opinion mining techniques on online user reviews. Methods which include opinion summarisation, opinion retrieval, and fake reviews or ratings detection. The aim is to uncover useful information based on the geographic location of users and the target business. These methods are used to find specific information, such as identifying hidden patterns, measuring of correlation between the users, calculating the market growth rate (%) and understanding customer preferences. The opinon polarity experessed in customers' comprehensive reviews is used to predict whether a producer can start a new business/service idea (introduce new products) or launch a branch of his existing business in a new region. The objective is to develop and implement an analytic system which can analyze data sets and draw conclusions to help companies make informed business decisions based on the feedback and experince of customers in other similar regions.

1.4.3 Objective 3: To reflect the influence factor of each customer

Social network of customers are online communities of people who share a common interest. They have attracted millions of users and are growing at fastest rate during the past years. Many users have integrated social networks into their daily practices. A communication platform and an online social venue where people can connect and communicate. Similar to the "real" world, a social network is a place to find others with common interests and to share information with them.

In addition, we aim to determine the communities and their evantual leaders using social network analysis. The leaders and followers are identified to weight the customers' reports in

order to reflect the influence factor of each customer and their ability to influence other members' choices and opinions.

1.4.4 Objective 4: To detect the most important leaders in a community

Influential customers in a given community are determined by adapting existing algorithms in the social influence state of the art, for instance graph centrality measure algorithms. Results of this task should be used to weight customers' reports and hence improve the prediction of the analytic system.

We need to study whether influential consumers have a real influence in shaping the opinion of their communities at a future time? The study explores the temporal interpolation of the social influence in a given community regarding similar products commented on during a previous time period.

Chapter 2 State of the Art

2.1 Introduction

During the last few decades, with the rise of Facebook, Amazon, Yelp and many other such Web 2.0 services, recommender systems have taken more and more place in our lives. They are used in various domains and become unavoidable in our daily online navigation. In e-commerce for suggestting articles that could interest the buyers, in online advertisement and in political campaigns for proposing to users the right contents and matching their preferences.

Recommender systems are algorithms aimed at suggesting relevant items to users and can generate a huge amount of income when they are efficient and effective.

2.2 Recommender systems

Recommender systems (RS) automatically predict the rating or preference that a user would give to an item and suggest items to users which they may find appealing (Ricci et al., 2011). Classical approaches attempt to predict users' interests by using their rating history and usually rely on either collaborative filtering (Jannach et al., 2010), (Pan et al., 2008), (Paterek, 2007), or content-based filtering approaches (Pazzani & Billsus, 2007). Collaborative filtering approaches recommend items based on the user's past behavior as well as other users' choices that purchased similar items (e.g., Amazon[4], eBay[5]).

RS are one of the most widespread application of machine learning technologies used to generate and provide suggestions for items and other entities to the users by exploiting various strategies.

[4] https://www.amazon.com/
[5] https://www.ebay.com/

Machine learning algorithms in RS are typically classified into the following main categories: content-based, collaborative filtering methods, knowledge-based and hybrid recommendation approaches.

To achieve higher performance and overcome the drawbacks of traditional recommendation techniques, hybrid RS combine the best features of two or more recommendation strategies in different ways to benefit from their complementary advantages.

2.2.1 Content-based methods

Content-based filtering approaches utilize a series of discrete characteristics of an item in order to recommend additional items with similar properties. So, content-based methods are based on similarity of item attributes. In these methods, the utility of an item for a user is estimated based on the utilities assigned earlier by that user to items that are "similar" to the recommended item.

For example, in a song recommendation application, in order to recommend songs to a user, the content-based recommender system tries to understand the commonalities among the songs the user has rated highly in the past (melody, harmony, lyrics, orchestration, vocal character, etc.). Then, only the songs that have a high degree of similarity to the user's preferences would be recommended to the user.

Obviously, each method has its strengths and weaknesses and therefore its results need to be interpreted with caution. Here are some of the limits of content-based filtering algorithm:

2.2.1.1 Limited by the features

Content-based methods are limited by the features that are explicitly associated with the recommended items by the RS. Therefore, in order to have a sufficient set of features, the content must either be in a form that can be analyzed automatically by a software (e.g., structured text), or that a human annotator can interpret it manually and categorize it accordingly.

While the automatic feature extraction methods work well for the text documents, some other domains such as multimedia data (e.g., graphical images, audio, and video streams) have an inherent problem with automatic information retrieval methods. Moreover, often due to limitations of resources it is not practical to assign attributes manually.

2.2.1.2 Limited content analysis

Another problem of content-based RS lies with their limited content analysis. That is due to the fact that if two different items are represented by the same set of features, they are considered exactly as identical to the system and so are indistinguishable.

Therefore, since text-based documents are usually represented by their most important keywords, a well-written article and a badly written one cannot be distinguished by content-based systems if they both used the same terms.

2.2.1.3 Over-specialization

This is one of the most common problems faced by the content-based recommendation system. A good recommender system must suggest diverse items which content-based system lacks. When the system can only recommend items that score highly against a user's profile, the user is limited to being recommended items similar to those already rated. It gives nothing "surprised". It hinders the users from discovering something new and different. Users are recommended items they are already familiar with.

For example, a person with no experience with Greek cuisine would never receive a recommendation for even the greatest Greek restaurant in town.

In addition, the problem with over-specialization is not only that the content-based systems cannot recommend items that are different from anything the user has seen before. But also, in certain cases, items should not be recommended if they are too similar to something the user has already seen. For example, different news article describing the same event. Therefore, some content-based RS, filter out items not only if they are too different from user's preferences, but also if they are too similar to something the user has seen before.

2.2.1.4 New user problem

The user has to rate a sufficient number of items before a content-based recommender system can really understand user's preferences and present the user with reliable recommendations. Therefore, a new user, having very few ratings, would not be able to get accurate recommendations**.**

2.2.2 Collaborative RS

Unlike content-based recommendation methods, collaborative RS try to predict the utility of items for a particular user based on the items previously rated by other users. Most music-discovery systems have been social recommenders, also known as collaborative filtering systems. These systems know little about songs' inherent qualities. They just assume that if you and a group of other people enjoy many of the same artists, you will probably enjoy other artists popular with that group.

Here are some of the limits of collaborative filtering algorithm:

2.2.2.1 New user problem

It is the same problem as with content-based systems. In order to make accurate recommendations, the system must first learn the user's preferences from the ratings that the user makes.

2.2.2.2 New item problem

New items are added regularly to RS. Collaborative systems rely solely on users' preferences to make recommendations. Therefore, until the new item is rated by a substantial number of users, the recommender system would not be able to recommend it.

2.2.2.3 Sparsity

In any recommender system, the number of ratings already obtained is usually very small compared to the number of ratings that need to be predicted. Effective prediction of ratings from a small number of examples is important. Also, the success of the collaborative recommender system depends on the availability of a critical mass of users.

For example, in the movie recommendation system there may be many movies that have been rated only by few people and these movies would be recommended very rarely, even if those few users gave high ratings to them. Also, for the user whose tastes are unusual compared to the rest of the population there will not be any other users who are particularly similar, leading to poor recommendations.

One way to overcome the problem of rating sparsity is to use user profile information when calculating user similarity. That is, two users could be considered similar not only if they rated the same movies similarly, but also if they belong to the same demographic segment. For

example, gender, age, area code, education, and employment information of users in the restaurant recommendation application. This extension of traditional collaborative filtering techniques is sometimes called “demographic filtering”. Another approach that also explores similarities among users has been proposed in, where the sparsity problem is addressed by applying associative retrieval framework and related spreading activation algorithms to explore transitive associations among consumers through their past transactions and feedback.

2.2.2.4 Scalability

As the numbers of users and items grows the system needs more resources in order to give the most accurate recommendations to the users. Most of resources are used in the purpose of determining users of similar tastes, and items with similar attributes. It is one of the problems found in collaborative filtering approach.

2.2.2.5 Privacy

Privacy is also a big issue in context of demographic recommender systems. In order to give the most accurate recommendation to the user, the system must acquire the most appropriate information of user, including demographic data (age, sex, email-id, hobbies etc.), and data about the location of a particular user which may breach the privacy of the user.

2.2.3 Demographic Filtering Systems

It uses pre-existing knowledge of demographic information about the users and their opinions for the recommended items as a basis for recommendations. Demographic systems are stereotypical, because they depend on the assumption that all users belonging to a certain demographic group have alike taste or preference.

The advantage is that it does not require history of user ratings that are required by collaborative and content-based techniques. This is a quick, easy, and straight forward approach for making results based on few observations. However, regarding the security and privacy issue, gathering of complete user information is impractical and so can be considered as a disadvantage.

2.2.4 Context-Aware Recommender System

It is one of the most trending recommender systems these days. It helps in giving diverse and accurate recommendations to the user. The contextual information may include location of the user, identity of people around, date, season, temperature, and etc. The contextual information may be retrieved in a number of ways, including:

- Explicitly i.e. gathering information by asking the direct questions from the user. For example, a website may recommend songs to a user by asking the current mood of the user.
- Implicitly i.e. from the data or the environment.
- Inferring (Based on evidence or by reasoning).

2.2.5 Knowledge-based recommendation techniques

Recently, with the increase in popularity of social networks, a new filtering approach based on social information has emerged. Social information is derived substantially from users' profiles, friends and connections, and tracks of social activities such as social ratings. Further surrounding contextual information of social networking such as social influence, viral marketing and social trust are quantified and exploited in recommender systems (Barbieri et al., 2014).

Knowledge-based recommendation offers items to users based on knowledge about the users, items and/or their relationships. Usually, knowledge-based recommendations retain a functional knowledge base that describes how a particular item meets a specific user's need, which can be performed based on inferences about the relationship between a user's need and a possible recommendation. Case-based reasoning is a common expression of knowledge-based recommendation technique in which case-based RS represent items as cases and generate the recommendations by retrieving the most similar cases to the user's query or profile.

Ontology, as a formal knowledge representation method, represents the domain concepts and the relationships between those concepts. In computer science and information science, an ontology encompasses a representation, formal naming and definition of the categories, properties and relations between the concepts, data and entities that substantiate one, many or all

domains of discourse. Ontology has been used to express domain knowledge in RS. The semantic similarity between items can be calculated based on the domain ontology.

2.2.6 Hybrid approach

Hybrid approaches attempt to combine both collaborative and content-based approaches where additional content features are used to improve the efficiency of collaborative filtering and to avoid the cold start problem (Pazzani, 1999). Several recommendation systems use a hybrid approach by combining collaborative and content-based methods, which helps to avoid certain limitations of content-based and collaborative systems. Different ways to combine collaborative and content-based methods into a hybrid recommender system can be classified as follows:

- Implementing collaborative and content-based methods separately and combining their predictions.
- Incorporating some content-based characteristics into a collaborative approach.
- Incorporating some collaborative characteristics into a content-based approach.
- Constructing a general unifying model that incorporates both content-based and collaborative characteristics.

2.2.7 Cold start problem

Cold start problem can be classified into two categories, cold start of new items and cold start of new users. Cold start problem for an item occur when we do not have enough previous rating related to that item. Also, it is difficult to recommend items to new users as the system don’t have any information related to his past purchases or it might be possible that he has not rated any item yet, so his taste is unknown to the system.

2.2.8 Other problems

Commercial RS are facing with other problems. For example, before recommending an item, the system must verify that this item is available for purchase (is in stock?) and can be sent to the user. In addition, depending to the origin or destination of an item we may need to consider what is going to be shipped, where it is going, and so to whom it can be recommended.

2.3 Opinion detection and polarity evaluation

Given a short text (e.g., a customer's review in our context), we need to determine whether this text contains an opinion or not about the targeted product. When an opinion is detected, we need to determine its polarity (positive or negative). Of course, one review may contain more than one opinion, usually with an associated target part (e.g., "This smartphone is very good, but not the microphone").

To achieve this goal, various strategies have been proposed (Pang & Lee, 2008), (Liu, 2012). The most effective ones are related to machine-learning paradigm (Hastie et al., 2009), viewing the opinion and polarity detection as a text classification tasks (Sebastiani, 2002), (Boiy & Moens, 2009).

In this case, we need to first select the most effective set of textual features (Forman, 2003), (Liu & Motoda, 2008). For example, if the adjectives "good" or "bad" have dualy a polarity. For some features (isolated terms, sequences of words), the polarity can be determind without considering the target application. Other selected terms tend, however, to be domain specific (e.g., the term "television" has a negative polarity in a movie review but may have a positive one in a book review).

In a second stage, a classifier must be trained according to a set of training examples. As possible classification schemes, the naïve Bayes, max entropy or the SVM method (Joachims, 2002) appear to be good starting points (Pang et al. 2002), (James et al., 2013). When the training set is composed of a relatively large number of instances, such classifiers could be effective (and under the assumption that time evolution will not strongly affect the selected predictors (Hand, 2006)). Using a unigram model with isolated words extracted from the training examples, Pang et al. (2002) obtained an overall accuracy of 85% when classifying opinions on their movie reviews corpus. Other studies have reported slightly better performance levels when considering bigrams or trigrams of words, while other researches indicated only small and non-significant performance differences when comparing unigram with bigram models (Pang & Lee, 2008).

As a second main paradigm, we can predefine a priori a set of words (mainly adjectives and adverbs) introducing or denoting a positive or negative opinion. Of course, other POS might play an important role (e.g., verbs such as love or hate). Such a strategy can be enlarged by

considering a large list of terms with associated values expressing their polarity (e.g., SentiWordNet (Esuli & Sebastiani, 2006)) based on the General Inquirer (Stone, 1966) or on LIWC (Tausczik & Pennebaker, 2010). Of course, the domain must be taken into account while a term may have different polarities according to the underlying subject ("go read the book" has a positive polarity in a book review, but a negative one in a movie review).

As a third strategy to define a set of pertinent features, we can select a set of seed words having a clearly positive or negative polarity in a given domain. When analyzing the corpus, the terms co-occurring with those seed words can be extracted to enrich the feature set. Such enrichment can also be done by considering neighboring terms in a thesaurus structure such as WordNet (Hu & Liu, 2004). Following a similar method, Jijkoun et al. (2010) suggest another way to generate a corpus-based lexicon. The performance difference between such a lexicon and a general lexicon is however small, but in favour of a corpus-based lexicon.

In opinion and polarity detection, NTCIR[6] organizes evaluation campaigns to determine whether a sentence written in the English, Japanese or Chinese language contains an opinion or not (e.g., descriptive sentence). Those sentences were extracted from different newspapers. If an opinion is detected, we need to classify it as positive, negative or mixed (Zubaryeva & Savoy, 2008), (Zubaryeva & Savoy, 2010). For the English language, UniNE obtained very good results based on the SentiWordNet thesaurus.

For the French language, HE-Arc developed hybrid classifiers based on shallow linguistic features and bag of words to classify opinions at the review level of French movie reviews (Ghorbel & Jacot, 2011) and French discussion forums (Infrarouge.ch) (Ghorbel, 2012). Moreover, they have improved their classifier by integrating a multilingual approach to take advantage of English resources such as SentiWordNet[7].

They have also studied the subjectivity issue in question answering systems and classified opinionated questions in order to retrieve candidate answer passages from a transcribed Arabic political TV show in Arabic (Bayoudhi et al., 2014).

[6] http://research.nii.ac.jp/ntcir/ntcir-15/index.html
[7] http://sentiwordnet.isti.cnr.it/

During the TREC blog track, the first task was to retrieve blog posts written in English according to a given request. Then after determining posts relevant to the query, the participants needed to classify them as positive or negative (Fautsch & Savoy, 2009).

2.4 Information retrieval and text categorization

One of the UniNE[8] previous research was focused on information retrieval (IR) technology adapted for languages other than English (various TREC, NTCIR, CLEF and FIRE participations) (Savoy, 2005). UniNE has also worked on cross-lingual IR (Dolamic & Savoy, 2010) and on domain-specific IR (Fautsch & Savoy, 2010).

More recently, UniNE has conducted some research on text classification and more precisely on authorship attribution. In this perspective, they have designed and evaluated a classifier based on the specific vocabulary (Savoy, 2012). In this domain, they have conducted a comparative evaluation of various feature selection schemes (Savoy 2015a).

In another perspective, UniNE has studied the lexical content of US presidential speeches (both electoral and governmental) in order to detect the specific characteristics of each candidate or president (Savoy, 2010). In this case, the clustering method applied to those data are derived from phylogenetics approach in which the distance between entities are computed according to the Manhattan function. Based on all State of the Union addresses, they have also applied a text clustering approach based either on the stylistic features or on the topical aspects of these governmental speeches (Savoy, 2015b).

2.5 Social Networks

The problem of social networking has been studied in sociology and more recently in the context of online social networks (Carrington et al., 2005), (Borgatti et al., 2013). For example, with Facebook one can have texts written by the user and links with his friends. Previous research has tackled the problem from two perspectives: a content-based approach and a link-based approach (Agrawal & Zhai, 2012). The content-based approach makes use of text clustering methods to which is added network information as a side-information in order to im-

[8] https://www.unine.ch/

prove clustering performance (Angelova & Siersdorfer, 2006), (Zhang et al., 2008), Aggarwal, 2012).

Link-based approaches combine textual content with structural information in graph-based models to represent the mutual influence between nodes (users) in the network. In this context, one of the important research questions pointed out so far is the social influence in social networks. This is the issue of how to find a set of individuals to receive a piece of information (e.g., ad) such that, after the initial dissemination the number of peers who receive it is maximized (Kempe et al., 2003).

It has recently been observed that certain and usually small numbers of influential individuals is essential in a diffusion process such as product adoption through a social network (Kaplan & Haenlein, 2011). Related works have focused on how to find these persons/nodes and use them as seed nodes such that the influence in the network represented by the total number of affected nodes will be maximized. Basic processes utilize graph theory measures to determine a centrality measure such as degree, closeness, betweenness and eigenvector centrality to find out about positions of such nodes and groups in the network, as well as clustering coefficients of nodes and their egocentric networks (Wasserman, 1994), (Easley & Kleinberg, 2010). Other processes are modeled by the Independent Cascade Model (Dodds & Watts, 2007), (Goldberg et al., 2001) or the Threshold Model (Granovetter, 1978). These models rely on the assumption that every edge (connecting node u to node v) has a weight which represents the probability that node v will be affected if u was affected before (in the previous round). Based on these edge weights, influence values for nodes are calculated.

Other related studies further suggest that the reason behind social contagion/diffusion, besides influential individuals, is the large number of susceptible individuals. As a further step, we will study what the most relevant factors are in accepting a service or a product. Is it being suggested by an influential person or being suggested to a susceptible person or perhaps a combination of both? A recent related work studied this question in the context of accepting a movie application via Facebook (Aral & Walker, 2012).

In social network analysis, HE-Arc[9] have studied different centrality measure algorithms using the French discussion forum (Infrarouge.ch) and Twitter dataset to determine influential users and central/hot discussion topics. They have also studied the polarization dynamics in social networks using the same dataset by analyzing the users' disagreement from the perspective of the polarity of their opinion. They used metrics such as the network disagreement index (NDI) (Dandekar et al., 2013) to find out whether the opinion formation process is polarizing (opinions are more and more separated into poles) or not.

[9] https://www.he-arc.ch/

Chapter 3 Corpus

3.1 Introduction

Yelp is a business review and social networking website. The site has pages devoted to individual restaurants and other companies, where Yelp users can submit a review of their foods or services using a five-star rating system. Restaurants can also update contact information, hours and other basic listing information or add special deals. In addition to writing reviews, users can react to reviews as "useful", "funny" or "cool." Reciprocally, business owners can respond to reviews privately by messaging the reviewer or publicly on their profile page.

Since 2014, Yelp added some new features such as booking hotels (in collaboration with Hipmunk a travel planning search site), ordering and scheduling manicures, flowers, Golf courses, and legal consultations through their website.

According to Emarketed[10] a web marketing agency located in Los Angeles:

"Small or growing businesses need to take advantage of every opportunity they can to get more exposure and bring in more customers online. Yelp was created to help people find good local businesses that they can trust to deliver good products and customer service."

In an article Matt Ramage, founder of Emarketed, explains the five main reasons Yelp is helpful for business owners:

1. ***"Influence people at the right moment*** *- The moment when someone is looking for a business on Yelp is at a critical point when they are ready to make a purchase. If they see your Yelp page and read a positive review, they can be easily influenced to use your business and possibly continue to be a customer for years to come.*

10 https://www.emarketed.com/blog/5-reasons-yelp-important-business/

2. ***Respond and interact with customers*** *- Should you receive any negative reviews or complaints on your Yelp page, you can use the site as a way to address these issues either publicly or privately. This can be a powerful tool to make your business appear invested in each customer's experience and responsive to correct any potential issues.*
3. ***Create a brand image*** *- With a Yelp business listing you can add your own content so that you have more control over the way your business is presented.*
4. ***Use Yelp as an authority*** *- In terms of internet marketing, Yelp is useful for website rankings and getting your business seen online. In SEO terms, Yelp is considered an "authority" site meaning that it can carry more weight and help boost traffic to your site.*
5. ***Make it easier to find out about you*** *- With Yelp you can add listings that provide people with crucial information about you that they can see immediately when they search in Google. Your Yelp can list your location in Google Maps, your business hours and a description of the items you have. Restaurants can have a menu available for customers to view online."*

Most of the time, before buying anything from a business, people look at their reviews online first. By doing this, they double-check their reputation before contacting them. Similary, this is becoming more and more common that people read others reviews on a restaurent before going there for the first time. So, a good restaurant review can tell someone their new favorite place or help them avoid a gastronomic disaster.

3.2 The choice of dataset

In the age of social media, rating sites and social geolocation applications and their growing importance have metamorphosed the food critics.

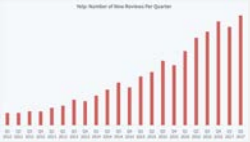
The mobile smart devices have become a global means of communication in our everyday lives. Local business reviews are more important than ever. Social geolocation makes it possible to look for nearby addresses and comment a restaurant from its smartphone.

In our research machine learning techniques used to conduct predictive analytics. Machine learning is a subset of Artificial Intelligence (AI) that uses a set of statistical tools to learn from data.

There are many community platforms which allow Internet users to evaluate and give their opinion on restaurants. Among them we can mention Foursquare, Yelp, and Tripadvisor which display literally millions of reviews of products and services, and that generally offer a good consensus.

Here is an overview of Foursquare, Yelp, and Tripadvisor most important features:

Table 3-1: The feature comparison table for Foursquare, Yelp and Tripadvisor

Features	FOURSQUARE	yelp	tripadvisor
Number of new reviews per quarter[11]			
Social geolocation	✔	✔	✕
Mobile App for Pros	✕	✔	✕
Management of his file: editing, adding photos	✔	✔	✔
Reply to comments as a restaurateur	✔	✔	✔
Alert email on new emails	✕	✔	✔
Creation of deals & special offers	✔	✔	✔
Detailed statistics on his file + reports emails	✔	✔	✔
targeted advertising	✔	✔	✔
Emailing tool for more reviews	✕	✕	✔
Open and long-term dataset	✕	✔	✕

For our research we choose Yelp dataset because:

- Yelp has been around longer and has a more established base of reviewers. It has been a steady performer over the years.
- With its open dataset “Yelp Dataset Challenge”, Yelp provides a valuable source for researchers in Natural Language Processing (NLP) to conduct researches such as sentiment analysis of textual data.

[11] https://www.restoconnection.fr/pourquoi-utiliser-tripadvisor-foursquare-et-yelp-pour-votre-restaurant/

- According to the data depicted in Table 3-1, Yelp has the largest number of features.
- Yelp's reviews contain a lot of metadata that can be mined and used to infer meaning, business attributes, and sentiment, to answer a lot of good questions such as: What's in a review? Is it positive or negative?

3.3 The fake reviews problem

Millions of users can indicate where they are and recommend outlets such as restaurants. However not all reviews are legitimate. Fake reviews written by real people are already common on review sites, but the amount of fakes generated by machines is likely to increase substantially.

Review sites have a responsibility to identify and take action against those who try to submit fake reviews. For example, Amazon, the leader in book retailers, has made it their mission to crack down reviews which fall under the label of "incentivized reviews[12]". According to Amazon, an "Amazon Verified Purchase" review means that the reader did in fact buy the book and has potentially read through it before posting a review.

Product reviews that are not marked "Amazon Verified Purchase" are valuable as well, but we either cannot confirm that the product was purchased at Amazon or the customer did not pay a price available to most Amazon shoppers. For unverified reviews, if not a fake review, then in most cases the reviewer received an advance copy of the book and was possibly on a launch team to support the book's release.

Mainly, there are three kind of fake reviews:

- Biased positive review: It is when someone connected with a business attempts to post a positive review of that business. It can also occur when a business offers its customers incentives, such as a free meal or a discount, to post reviews.
- Biased negative reviews: It is when someone submits a deliberately malicious review about a property in an effort to unfairly lower its ranking position or improperly discredit the property in some way. Most biased negative reviews come either from

12 A product review written by a consumer that received compensation, such as a discount or free product, for writing that review.

someone connected to a rival establishment, or from someone who is trying to blackmail a business by threatening to submit a false negative review.

- Paid reviews: This is when a business, employs the services of an individual or a company to boost its ranking position on rating sites with positive reviews. Moreover, the paid reviews could also be negative ones, targeting concurrent products or services.

3.4 Yelp challenge dataset (2014)[13] and its details for restaurant domain

In this section, we will present in more detail the Yelp Dataset (July 30, 2014) released for the academic challenge we used for this thesis.

3.4.1 An overview of the Yelp dataset structure

The dataset is provided in the form of separate Json objects and includes five objects.

Table 3-2: Structure of the 3 main tables in Yelp dataset

review		**business**		**user**	
funny:	int	attributes:	string	yelping_since:	string
useful:	int	business_id:	string	votes: {funny: 1, useful: 5, cool: 0},	string
cool:	int	full_address:	string	name:	string
user_id:	string	open:	boolean	review_count:	int
review_id:	string	hours:	string	user_id:	string
stars:	int	categories:	string	friends:	string
text:	string	city:	string	fans:	int
date:	string	review_count:	int	average_stars:	float
type:	string	name:	string	type:	string
business_id: string		neighborhoods:	string	compliments:	string
		longitude:	float	elite: string	
		state:	string		
		stars:	float		
		latitude:	float		
		type: string			

[13] https://www.yelp.com/dataset

Each file is composed of a single object type, one Json-object per-line. The unical identifiers of each review, each business and each user are used to link different tables' data:

1) Business profile objects - contains business data which includes information about the type of business, location (latitude and longitude), attributes, rating, categories, and business name, as well as a unique business identification.

Table 3-3: “business” object sample

full_address: 8308 Greenway Blvd\nMiddleton, WI 53562
hours: {"Monday": {"close": "23:00", "open": "05:00"}, "Tuesday": {"close": "23:00", "open": "05:00"}, "Friday": {"close": "00:00", "open": "05:00"}, "Wednesday": {"close": "23:00", "open": "05:00"}, "Thursday": {"close": "23:00", "open": "05:00"}, "Sunday": {"close": "23:00", "open": "06:00"}, "Saturday": {"close": "00:00", "open": "06:00"}}
categories: ["Burgers", "Fast Food", "Restaurants"]
city: Middleton
review_count: 6
name: McDonald's
state: WI
stars: 4.0
attributes: {"Take-out": true, "Wi-Fi": "free", "Alcohol": "none", "Caters": false, "Noise Level": "average", "Takes Reservations": false, "Delivery": false, "Parking": {"garage": false, "street": false, "validated": false, "lot": false, "valet": false}, "Has TV": true, "Good For": {"dessert": false, "latenight": false, "lunch": false, "dinner": true, "brunch": false, "breakfast": false}, "Attire": "casual", "Waiter Service": false, "Accepts Credit Cards": true, "Good for Kids": true, "Good For Groups": true, "Price Range": 1}

2) User profile objects - A user object includes information about the user identification, name, review count, votes, user's friend mapping, etc. As each user is also described by its list of friends, this dynamic data from can be used to study (in space and time) connectivity patterns which its result could be useful for social network reconstruction.

Table 3-4: “user” object sample

yelping_since: 2009-01
review_count: 172
fans: 7
average_stars: 3.66

3) Review content objects - contains full review text data associated with the reviewer's explicit feelings specified by a rating from 1 (negative) to 5 (positive) stars about the business reviewed. The object is connected with a specific business identification and user identification. Each review was subject to being labeled useful, funny, or cool by other reviewers.

Table 3-5: "review" object sample

votes: {"funny": 0, "useful": 1, "cool": 0} **stars**: 4 **date**: 2011-11-08 **text**: Great truck stop restaurant. I've had breakfast and dinner here and it has always been good. Huge portions and reasonable prices. The bakery I guess is legendary. They have doughnuts the size of your head! Eclairs the size of your arm and Cream Puffs that are about 10\" high. On our most recent trip in the area, stopped at the gas station attached to the restaurant fo some quick snacks and of course BAKERY.\n\nIf you are driving by and see the sign for The Pine Cone Restaurant - Take the next off ramp and stop it.

4) Check-in information - the check-in information sets describe the number of business check-ins per time and per day of the week (Monday to Sunday).

Table 3-6: "checkin" object sample

checkin_info: {"19-5": 1, "16-3": 1, "9-5": 1, "17-3": 1, "21-2": 1}

5) Tip content data - the tip information sets also include the tip text that a user would like to tell the world about a specific business, the date, and other users' "number of likes". Tips are shorter than reviews and tend to convey quick suggestions.

Table 3-7: "tip" object sample

text: Great food, huge portions and a gift shop and showers. **likes:** 0 **date:** 2012-05-16

3.4.2 Some statistics about the Yelp dataset

The dataset contains information for 42,153 businesses, in five cities from three countries. It has 31,617 check-in sets, 252,898 users, and 1,127,525 reviews throughout the cities of Edin-

burgh (U.K.), Waterloo (Canada), Phoenix (U.S.), Las Vegas (U.S.), and Madison (U.S.), and 403,210 tips for these businesses.

Figure 3-1: Geolocation of all five cities included in the Yelp dataset on the map

The dataset also contains 320,002 business attributes, and 955,999 edge social graphs (a graph that depicts personal relations of users). It spans a period of 10 years, from 2004 to 2014 (time dependent features), and with a rough estimation contains few billion words.

3.4.3 Five-star rating system

In the dataset's reviews content object, we count 1,127,525 reviews. Each text review is written by a user to describe his opinion about one business. Each time a user writes a text review, he also rates the business based on a scale of five, from 1 to 5 stars. This five-star rating system gives the user a lot of freedom without complication. The five stars are granular enough to create an understanding of how good something is without overloading the user with choices. The number of stars given to a business can mean different things in different cultures and countries and from one user to another. The following are what the restaurants' stars means to one of the Yelp reviewers[14]:

Table 3-8: What prompts someone to give to a 1-star review on Yelp for a restaurant?

5 stars	I thoroughly enjoyed it and will go out of my way to eat here regularly.
4 stars	I would go out of my way to eat here again.
3 stars	I would go again if I'm in the neighborhood, but I won't go out of my way for it.
2 stars	I would only eat here again if someone else really wanted to and was paying for it.
1 star	I would not eat here again, even if it's free.

14 According to Nadine F.: https://www.yelp.com/topic/washington-what-do-your-yelp-stars-mean

3.4.4 A restaurant dataframe by filtering all business

For our study, we are interested solely by 14,300 business which are categorized as food or restaurants. Among all reviews in the dataset, 706,404 are restaurant reviews. The following histograms compare the number of reviews in global (all businesses included) of each category of stars, to the number of restaurant reviews in the same rating category.

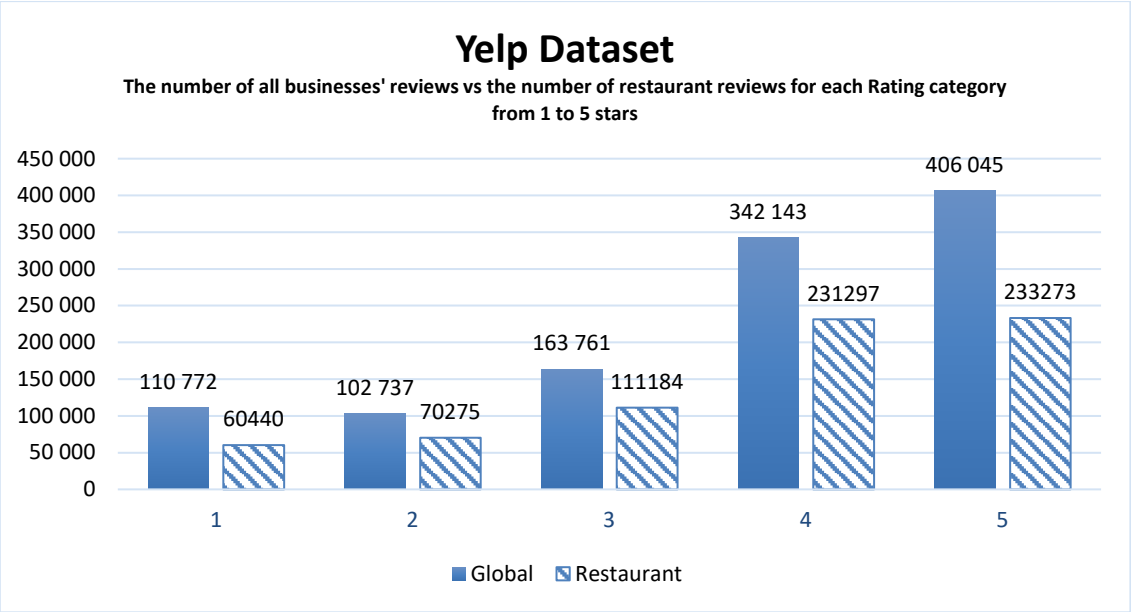


Figure 3-2: Yelp Dataset - The number of all businesses' reviews vs the number of restaurant reviews for each Rating category from 1 to 5 stars.

Figure 3-2 shows that more than 80% of restaurant reviewers rate a restaurent from 3 to 5 stars (3 stars: 15.74%, 4 stars: 32.74%, and 5 stars: 33.02%). Unexpectedly, from these statistics, we can clearly see that most of the review writers are rather satisfied clients. Only less than 19% of clients can be considered as unsatisfied (1 star: 8.56%, and 2 stars: 9.95%).

The same pattern repeats when it comes to a similar comparison but on a smaller level such as cities. Here are the statistics (in form of histograms) for five cities in U.S., Canada, and U.K. In the following figures we can observe that 464,321 of restaurant reviews (~ 66%) are for restaurants located in three U.S. cities of Phoenix, Las Vegas, and Madison. With the largest concentration of 320,100 of restaurant reviews (~45%) in Las Vegas.

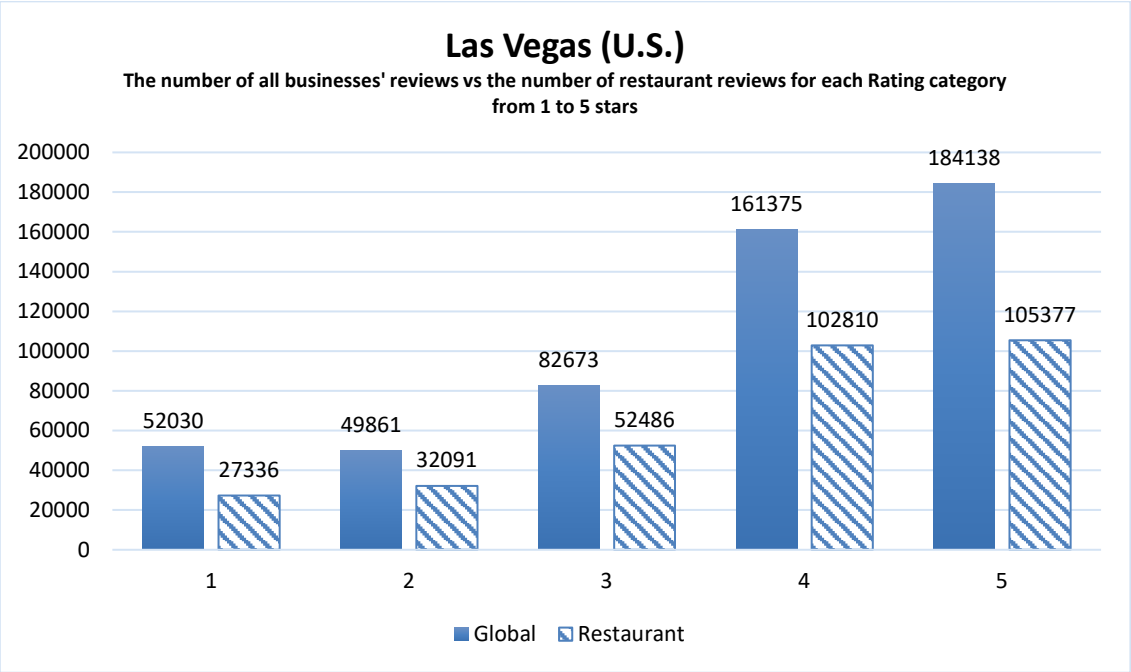


Figure 3-3: Las Vegas (U.S.) - The number of all businesses' reviews vs the number of restaurant reviews for each Rating category from 1 to 5 stars.

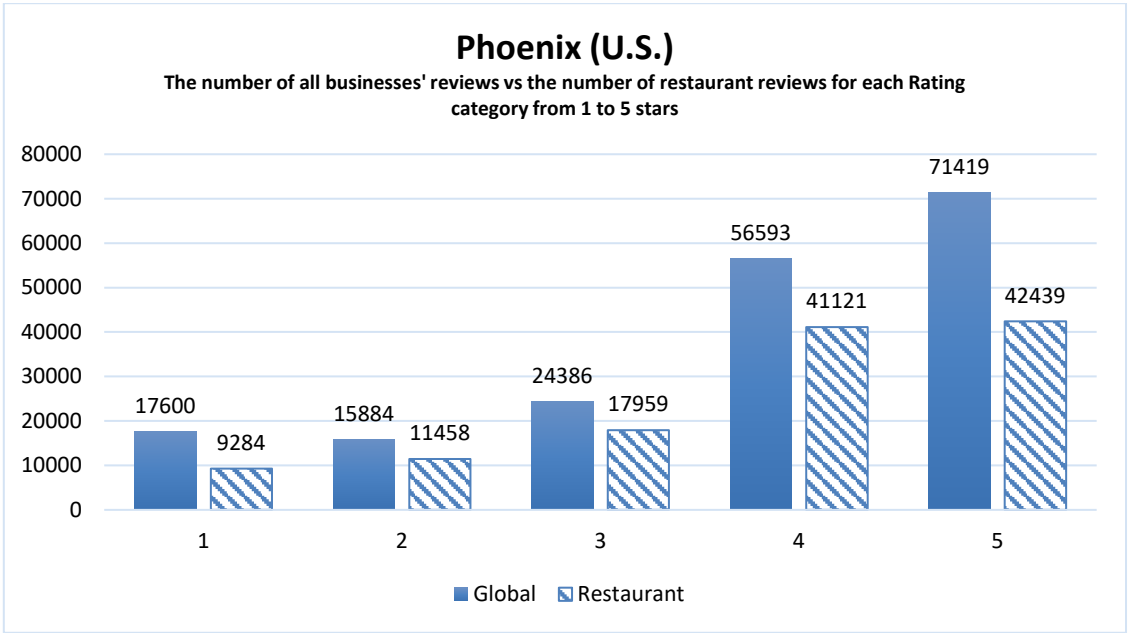


Figure 3-4: Phoenix (U.S.) - The number of all businesses' reviews vs the number of restaurant reviews for each Rating category from 1 to 5 stars.

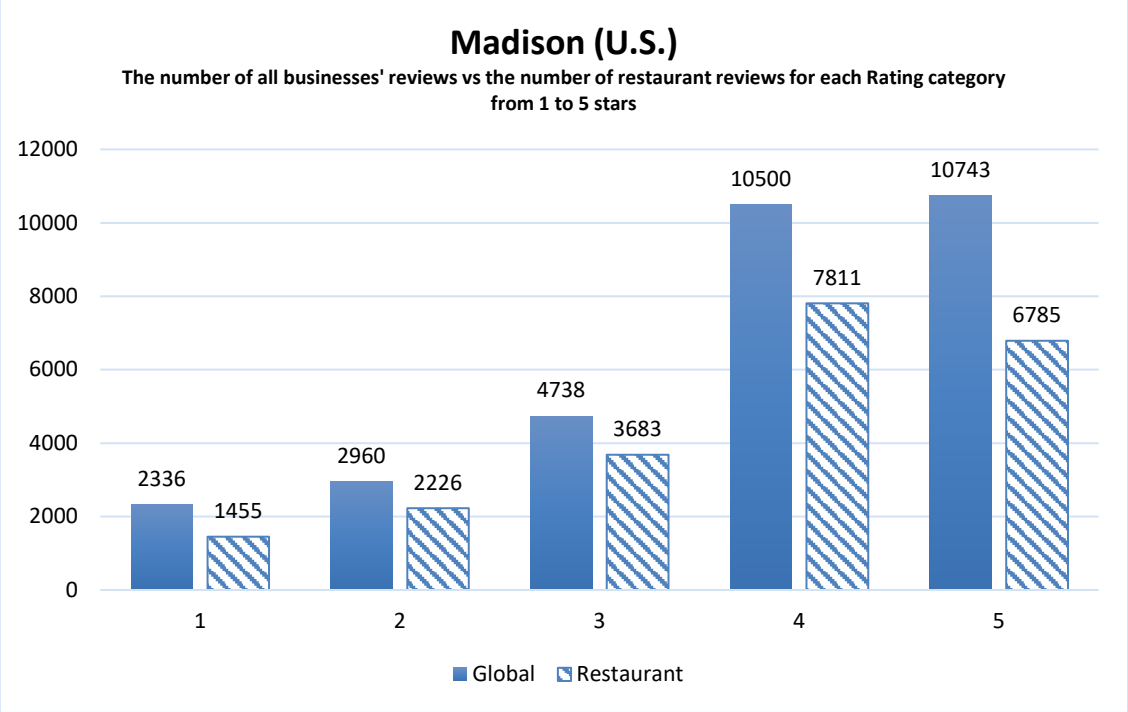


Figure 3-5: Madison (U.S.) - The number of all businesses' reviews vs the number of restaurant reviews for each Rating category from 1 to 5 stars.

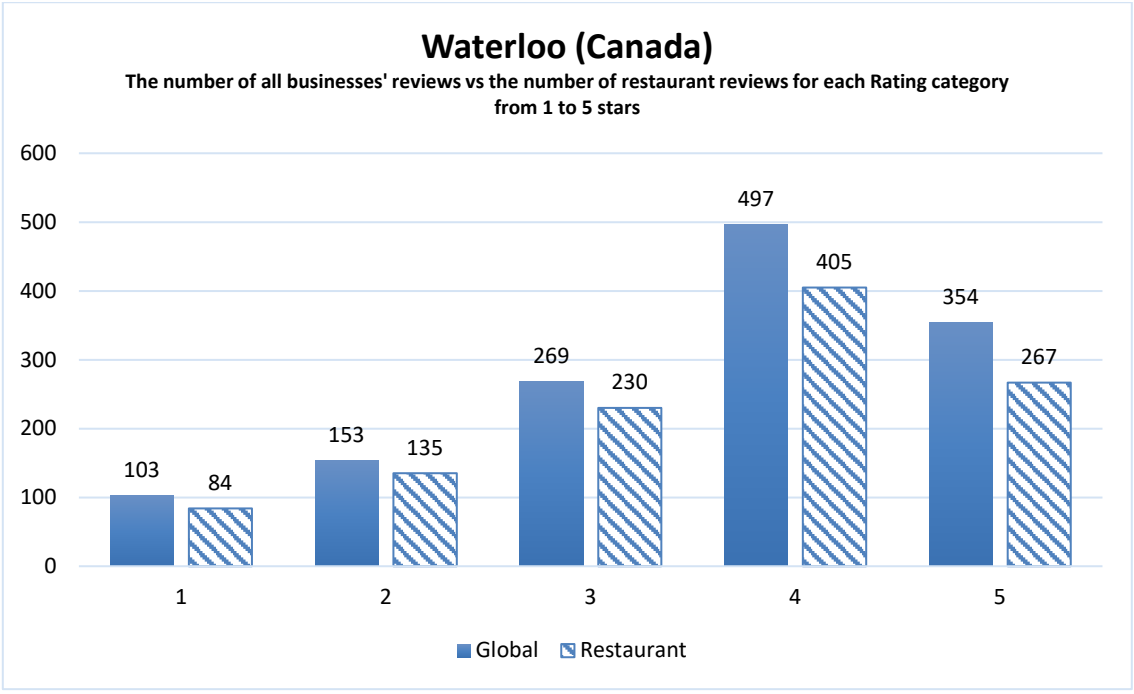


Figure 3-6: Waterloo (Canada) - The number of all businesses' reviews vs the number of restaurant reviews for each Rating category from 1 to 5 stars.

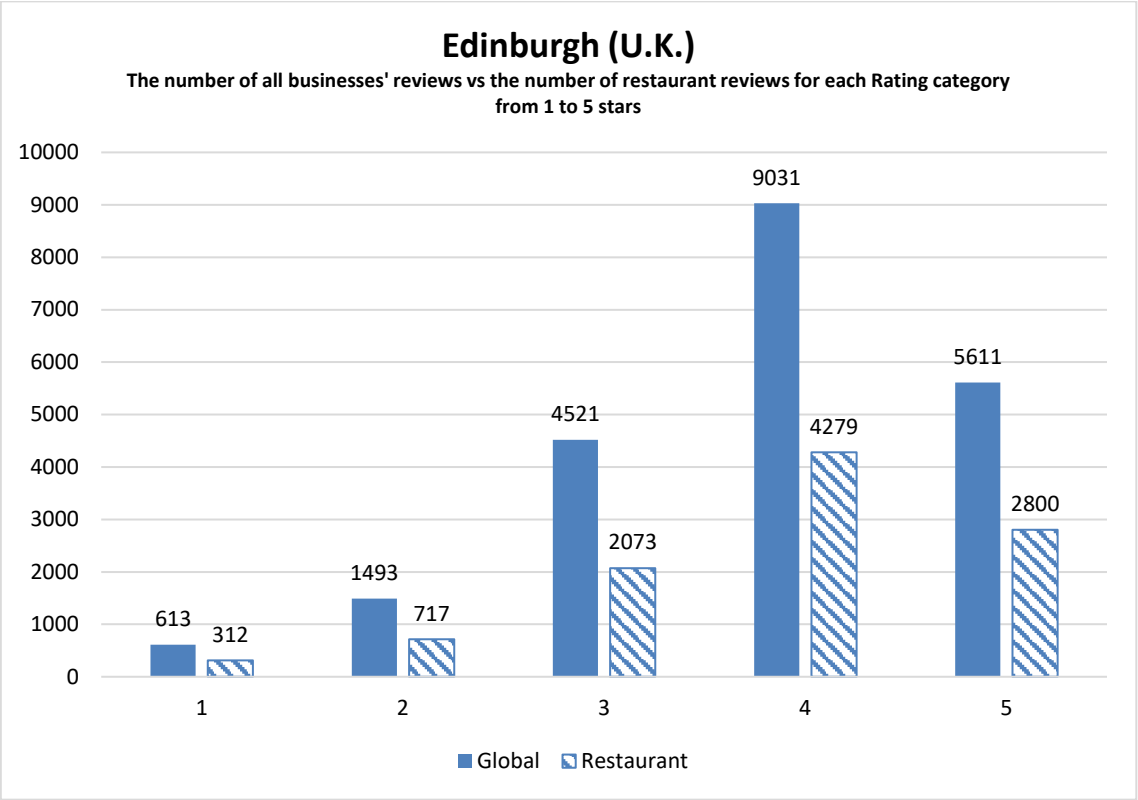


Figure 3-7: Edinburgh (U.K.) - The number of all businesses' reviews vs the number of restaurant reviews for each Rating category from 1 to 5 stars.

The Yelp dataset does not have much reviews for Canada and United Kingdom. As we can observe in Figure 3-6, we have only 1,121 (~0.16%) restaurant reviews for the city of Waterloo, Canada. Figure 3-7 depicts that the city of Edinburgh, United Kingdom has 10,181 (~1.45%) restaurant reviews in the dataset.

3.5 Summary

This chapter highlights Yelp challenge dataset (2014) details. It describes the dataset structure to make it easier the Information Retrieval (IR) needed for purposes such as creating a restaurant-specific domain dataframe.

In chapter four we discuss how to fully take advantage of Yelp Challenge Dataset reviews to generate a highly focused sentiment lexicon for feature-based opinion mining on restaurants.

Chapter 4 Restaurant-Specific Sentiment Lexicons

In this chapter, features and keywords search method is used to generate a highly focused sentiment lexicon for "Yelp Challenge Dataset" reviews on restaurants. Research objective in this chapter is to identify the specific components of a review that lead to a sentiment for a feature and then calculate the overall sentiment of the review.

A review can express an opinion about one or more aspects about a business depending on the experience of the user with respect to those aspects. In case of a restaurant, the food, the ambience, the service or even the discounts offered can often influence the user ratings. To enable a finer-grained analysis of sentiment of the review, we use a predetermined set of five most important features of a restaurent:

- Food
- Service
- Ambience
- Deals/Discounts
- Quality-Price Ratio

To identify aspects the users are most interested in, we need to find out which are the important features highlighted in each review. For this porpose, and contrary to the traditional sentiment analysis which provides an overall picture of a review that can be positive or negative, we employ feature-based opinion mining.

To process with less complexity the Yelp challenge dataset which is provided as JSON[15] files (semi structured data) we use Hive. For this porpose all five JSON data files are loaded into

[15] https://en.wikipedia.org/wiki/JSON

Hive tables which are sortable and searchable. Mainly we use HCatalog, Pig and Hive to load and process data.

After loading Yelp academic dataset review ".json" file and Yelp academic dataset business ".json" file into Hive tables, we generate a multi-label classification table (file in tab-separated values - TSV) using word bigrams and trigrams that contain a relative keyword to a given aspect according to their concordance hits and frequency.

Then by order we follow two aims:

- First to retrieve a list of word bigrams and trigrams which are "mostly used" in all reviews on "Restaurants" to express an opinion or sentiment about one of the five features given above and rate them from -5 to 5 (0 means neutral or not applicable). A rating of -5 is the most negative opinion on a given feature and the rate 5 is used for the most positively opinion on that feature.
- Second to classify each review separately the final TSV file may look like the following:

Table 4-1: Examples of word bigrams and trigrams

Reviews						
Words (bigrams and trigrams)	Frequency	Food	Service	Ambience	Deals/ Discounts	Quality-Price Ratio
absolutely delicious	3,562	5	0	0	0	0
excellent service	4,223	0	5	0	0	0
pleasant ambience	13	0	0	2	0	0
decently priced	676	0	0	0	0	3
worst service	1,155	0	-5	0	0	0
pretty good deal	426	0	0	0	3	0

In addition to our main tools (Hortonworks Sandbox, H2O, Neoj4 and Excel 2013 with Power View), we use a freeware corpus analysis toolkit for concordance and text analysis called "AntConc[16]".

[16] http://www.laurenceanthony.net/software/antconc/

As first step we try to find a maximum number of keywords (one single word) that may have been used by a user to describe a feature. To do so, we create a text file (we call it total_review_text) from the database which contains all the 706,404 reviews on “Restaurants”. From this point, this file is mainly used for searching keywords, as well as all their bigrams and trigrams.

Using SentiWordNet, we find the most used synonyms of each keyword and repeat the search for each synonym. The criteria of choosing a synonym is its frequency of use in the total_review_text.

4.1 Bigrams and patterns

In total we find 309 unique bigrams beginning with the word “very”: (frequency >= 5)

Table 4-2: Unique bigrams beginning with the word “very”

Bigrams with “very”	
How many times	**Bigram**
2362	very good
716	very nice
680	very tasty
665	very friendly
472	very well
286	very very
284	very fresh
…	…
Total: 14582	

In total we find 45 unique bigrams beginning with the word “really”: (frequency >= 10)

Table 4-3: Unique bigrams beginning with the word “really”

Bigrams with “really”	
How many times	**Bigram**
1613	really good
575	really enjoyed
415	really like
401	really liked

358	really nice
248	really great
218	really is
…	…
Total: 10333	

In total we find 45 unique bigrams beginning with the word “extremely”: (frequency >= 5)

Table 4-4: unique bigrams beginning with the word “extremely”

Bigrams with “extremely”	
How many times	**Bigram**
84	extremely friendly
43	extremely attentive
38	extremely fresh
36	extremely tasty
32	extremely flavorful
31	extremely helpful
29	extremely well
…	…
Total: 594	

There are many patterns that we can find, here are some:

Table 4-5: Example of other patterens

Some patterns	
How many times	**Pattern**
425	definitely be back
401	really liked
107	be back again
95	This is a great place
41	I really like this place

As one can see, all the most frequent bigrams indicate positive opinion about the underlying restaurant.

4.2 Example: “Delicious” to describe food feature

As an example, we search a keyword which might be used in a review to describe the “Food” feature of a restaurant. We choose the adjective “delicious”.

The keyword “*delicious*” is used 125,575 times in the total_review_text. According to Senti-WordNet its main synonyms are “*yummy*”, "*delightful*", “*delectable*”, “*scrumptious*” and “*toothsome*”.

The search result reveals that the synonym:

- The word “yummy” is used 30,652 times.
- The word “delightful” is used 4,282 times.
- The word “delectable” is used 2,135 times.
- The word “scrumptious” is used 1,936 times.
- The word “toothsome” is used only 107 times in the total_review_text.

Tokenization is the process of demarcating and possibly classifying sections of a string of input characters. The resulting tokens are then passed on to some other form of processing. The process can be considered a sub-task of parsing input. In Table 4-6, n-gram Type (unique values) and Tokens (occurrence frequency) are depicted. After text processing we find all the word bigrams and trigrams which are relative to these six words as following:

Table 4-6: statistical analysis of the keyword “delicious” and its main synonyms in the dataset

Total No. of times used / word	**"delicious"**	**"yummy"**	**"delightful"**
The word itself	125,575	30,652	4,282
bigram Type	423,393	164,109	38,216
bigram Tokens	2,305,792	579,771	77,365
trigram Type	1,214,527	385,375	64,261
trigram Tokens	2,305,791	579,770	77,364
	“delectable”	**“scrumptious”**	**"toothsome"**
The word itself	2,135	1,936	107

bigram Type	22,119	19,932	1,607
bigram Tokens	38,459	34,619	1,896
trigram Type	34,164	30,522	1,860
trigram Tokens	38,458	34,618	1,895

We filter and keep only the bigrams and trigrams which contain our main keyword “delicious” or one of its synonyms. Here are some examples:

Table 4-7: Examples of word bigrams and trigrams which contain the main keyword “delicious”

Rank	Freq	N-gram
1	24454	was delicious
2	16936	delicious and
4	13599	and delicious
5	12164	delicious the
6	12027	delicious i
9	8010	is delicious
10	7166	were delicious
17	5073	a delicious
20	4458	so delicious
21	4079	delicious food
22	4001	delicious but
27	3692	are delicious
29	3603	absolutely delicious
36	3182	delicious we
42	2776	delicious as
43	2736	delicious it
48	2623	very delicious
52	2543	delicious my
55	2366	the delicious
83	1698	delicious they
100	1502	of delicious

Rank	Freq	N-gram
1	6980	it was delicious
2	4667	was delicious and
6	3422	delicious and the
7	3396	food was delicious
8	3048	was delicious the
9	2869	was delicious i
10	2306	food is delicious
12	2084	fresh and delicious
13	1919	is delicious and
14	1832	and delicious the
15	1540	and delicious i
16	1499	which was delicious
18	1392	everything was delicious
19	1316	was absolutely delicious
20	1269	delicious and i
21	1264	were delicious and
22	1196	was so delicious
23	1175	was delicious but
26	1156	delicious as well
27	1146	delicious i had
28	1124	they were delicious

Table 4-8: Examples of bigrams and trigrams which contain “toothsome”, a synonym of the main keyword “delicious”

Rank	Freq	N-gram
1	19	toothsome and
2	18	and toothsome

Rank	Freq	N-gram
1	3	a nice toothsome
2	3	and toothsome the

4	10	a toothsome
7	7	is toothsome
11	6	toothsome the
18	4	more toothsome
19	4	the toothsome
21	4	toothsome but
35	3	nice toothsome
36	3	of toothsome
43	3	toothsome chew
47	3	yet toothsome
66	2	are toothsome
71	2	bit toothsome
74	2	but toothsome
79	2	crusty toothsome
102	2	perfectly toothsome
127	2	toothsome bread
128	2	toothsome crunch
129	2	toothsome crust
130	2	toothsome my

3	3	chewy and toothsome
4	3	is toothsome and
5	3	toothsome and fresh
6	3	toothsome and the
8	2	a bit toothsome
12	2	and toothsome my
13	2	bit more toothsome
15	2	moist and toothsome
16	2	nice toothsome crunch
19	2	perfect crusty toothsome
24	2	toothsome with a
29	2	with a toothsome
41	1	a crunchy toothsome
51	1	a little toothsome
56	1	a perfectly toothsome
58	1	a pretty toothsome
62	1	a really toothsome
66	1	a toothsome applesauce
67	1	a toothsome applewood

Table 4-9: Examples of bigrams and trigrams which contain “yummy”, a synonym of the main keyword “delicious”

Rank	Freq	N-gram
1	2895	yummy and
3	2789	was yummy
4	2746	yummy i
6	2357	so yummy
7	2325	and yummy
8	2130	yummy the
10	1465	very yummy
15	1120	yummy yummy
16	1108	is yummy
17	1096	a yummy
19	1068	super yummy
21	1054	were yummy
25	981	yummy food
26	966	really yummy

Rank	Freq	N-gram
1	709	it was yummy
3	541	yummy and the
4	480	was yummy and
6	452	was so yummy
8	396	so yummy i
11	347	yummy yummy yummy
12	325	so yummy and
13	304	was very yummy
14	292	fresh and yummy
15	276	was yummy the
16	269	was yummy i
17	254	is so yummy
19	251	was really yummy
20	228	food is yummy

27	951	yummy but
31	923	the yummy
38	719	yummy we
43	684	are yummy
48	637	of yummy
62	551	yummy my
64	541	yummy it

22	220	was yummy but
23	218	and yummy i
24	215	was super yummy
25	213	food was yummy
26	212	so yummy the
27	212	yummy and i
28	207	which was yummy

From this example we can easily determine that “delicious” keyword and “yummy”, "delightful", “delectable”, “scrumptious” synonyms are more often used by reviewers to describe “Food” feature, but “toothsome” is rarely used.

Once for each of our five feature / prediction / attribute the most often used keywords (and their synonymes) are found, we can generate a highly focused sentiment lexicon for “Yelp Challenge Dataset” reviews on Restaurants.

Then we should find a method to rate each word bigram and trigram as explained above from -5 to 5. The final table will include the total list of n-gram used in the total_review_text as well as their rating for each of the five features.

This sentiment lexicon can be used as training data for an Opinion Detection Model and later in a second step we can use this model to classify each of the 706, 404 reviews on “Restaurants” individually.

4.2.1 N-grams preceded by word “not”

Now that we have the data organized into bigrams and 3-grams, it’s easy to tell how often words are preceded by a word like “not”. How many n-grams (n = 2 and 3) starting with "Not"? It is important to know because if an adjective (such as "good") with a positive score in the SentiWordNet is preceded by the word "not" (such as "not good") then its score:

- should not be considered in the posscore calculation
- but must be added to the negscore calculation of this comment.

Table 4-10: N-grams preceded by word “not”

Number	bigrams	Number	3-grams
2015	not a	286	not a fan
1125	not too	256	not sure if
1097	not the	254	not the best
960	not to	243	not a big
796	not sure	195	not to mention
643	not only	176	not a huge
535	not as	174	not going to
527	not be	137	not on the
484	not that	133	not as good
451	not have	129	not too sweet
373	not bad	125	not sure what
350	not so	121	not to be
340	not just	108	not a bad
320	not really	102	not too much
301	not like	98	not in the
282	not even	93	not at all
264	not in	86	not sure how
239	not much	84	not bad at
239	not my	84	not so much
238	not disappoint	71	not sure why
204	not for	70	not be disappointed
202	not on	70	not only is
200	not going	69	not much of
196	not overly	69	not my favorite
191	not your	69	not that i
179	not enough	69	not your typical
179	not quite	66	not only was
168	not one	64	not for the
165	not very	64	not have been
160	not get	63	not a lot
152	not all	63	not too bad
152	not been	62	not have to
139	not at	62	not really a
135	not an	61	not only did
133	not great	61	not want to
130	not disappointed	57	not the case
110	not being	55	not in a
110	not greasy	53	not one of
104	not know	51	not be the
103	not what	51	not like the

As one can see, the word adjacent or the bigram appearing after a “not” is not always very informative. Moreover, the expression “not bad” could be recived as positive or maybe neutral. But a longer context is certainly needed.

4.3 Opinion profile construction from social media

This section's research is a collaboration between the Data Analytics Group (Prof. Hatem Ghorbel, William Droz, and Magdalena Punceva) at the university of Applied Sciences and Arts of Western Switzerland and the Department of English (Prof. Martin Hilpert) at the University of Neuchâtel.

Social Media such as Facebook, and Twitter are playing an increasingly important role in our daily lives. Especially the younger generation spends considerable time in these virtual environments, which indicates a likely further increase in popularity towards the future. Researchers in computer science have realized that the huge amounts of data accumulating in these environments are a resource that allows the extraction of useful knowledge.

However, their efforts to extract meaningful information from that data has been hampered by a lack of cooperation and the difficulty to obtain these data with researchers in the humanities, who have been studying how human beings convey meaning through language use.

Sentiment analysis, also known as opinion mining, generally aims to define the opinions and subjective expressions in terms of positive or negative sentiment [Liu2010]. Typically, sentiment analysis proceeds on the basis of opinionated texts produced on web-based social platforms. An approach that goes deeper than classifying texts into positive and negative sentiments is referred as feature-based sentiment analysis.

The goal of featured-based sentiment analysis is to extract the set of product features that are commented on, and the respective sentiment orientation for each feature. For example, in the expression “I find the hands of the Rolex watch luminous and great however its numerals are unfashionable”, the word “Rolex” denotes the product while “hands” and “numerals” are features that are commented upon. As the example shows, features may be evaluated differently, some positive, others negative. A realistic representation of opinions thus needs to be fine grained, establishing an opinion profile rather than a single polarity value. The current section builds upon the work of Hu and Liu [Hu2004] and [Ding2008], who summarize product re-

views in terms of central features of the reviewed products. Wei et al. [Wei2010] use a lexicon of positive and negative words to improve the effectiveness of product feature extraction. While we will adopt some of the methods, we believe that feature-based sentiment analysis ought to be based on a close analysis of the specific domain that the reviews address. For instance, restaurant reviews address elements such as the food, the service, the ambiance, and the price level.

In this section we analyze restaurant reviews in terms of semantic frames to provide an opinion profile that reflects customer satisfaction along several features. We first constructed a 'restaurant frame' that contains the features that matter to restaurant reviewers.

To represent in a concrete way a set of values we can calculate their mean and variance.

$$\overline{x} = \frac{(\sum x)}{n} \tag{4.1}$$

The median is known also as a measure of location, splitting the distribution in two equal parts. As a robust location estimation, modifying the smallest or biggest value would not change the value of the median. Thus, the median does not use all the information in the data and so it can be shown to be less efficient than the mean or average, which does use all values of the data. To calculate the mean, we add up the observed values and divide by the number of them.

The standard deviation is a summary measure of the differences of each observation from the mean. If the differences themselves were added up, the positive would exactly balance the negative and so their sum would be zero. Consequently, the squares of the differences are added. The sum of the squares is then divided by the number of observations minus one to give the mean of the squares, and the square root is taken to bring the measurements back to the units we started with.

The sum of the squares of the differences (or deviations) from the mean, is divided by the total number of observations minus one, to give the variance. Thus,

$$Variance = \frac{\sum(x-\overline{x})^2}{n-1} \tag{4.2}$$

Finally, the square root of the variance provides the standard deviation:

$$SD = \sqrt{\frac{\sum(x-\overline{x})^2}{n-1}} = \sqrt{Variance} \quad (4.3)$$

Available for research purposes, we have used the Yelp new challenge dataset and a statistical sum up is given in the following table.

Table 4-11: Yelp academic dataset description

Corpus size (reviews)	***706,404***		
	Mean	**Median**	**Std deviation**
Review size (sentences)	9.41	7	7.81
Review size (words)	128	94	117

4.4 Data sampling

Here we focus on the reviews for restaurants. We aim to predict the rating for a restaurant from the review text, and the restaurant's statistic. In sentiment analysis, training data is composed of reviews as the input, with the label indicating whether that piece of text is positive or negative. So, since we plan to label manually reviews, identifying and analyzing a representative sample is more efficient and cost-effective than surveying the entirety of the dataset. The complete review data includes 1,127,525 comments, from which 706,404 of them are restaurant reviews. In addition, users can rate reviews as either being useful, funny or cool.

	yelp.review.funny	yelp.review.useful	yelp.review.cool	yelp.review.user_id	yelp.review.review_id	yelp.review.stars	yelp.review.text
0	0	2	1	Xqd0DzHaiyRqVH3WRG7hzg	15SdjuK7DmYqUAj6rjGowg	5	dr. goldberg offers every
1	0	2	0	H1kH6QZV7Le4zqTRNxoZow	RF6UnRTtG7tWMcrO2GEoAg	2	Unfortunately, the frustra
2	0	1	1	zvJCcrpm2yOZrxKffwGQLA	-TsVN230RCkLYKBeLsuz7A	4	Dr. Goldberg has been n
3	0	0	0	KBLW4wJA_fwoWmMhiHRVOA	dNocEAyUucjT371NNND41Q	4	Been going to Dr. Goldbe
4	0	2	1	zvJCcrpm2yOZrxKffwGQLA	ebcN2aqmNUuYNoyvQErgnA	4	Got a letter in the mail la

Figure 4-1: Users can rate reviews as either being useful, funny or cool

We decide to choose 500 reviews for analyzing. First, to reduce the numbers of reviews on "Restaurants", we adopt four criterias in research sources selection as filtering factors:

- Review usefulness > 3
- Review coolness > 2
- Review stars > 3

- Business review_count > 5

This filtering reduces the numbers of restaurant reviews to 22,584 reviews. We take the top 500 reviews from this list.

Table 4-12: Data filtering for choosing a smaller number of restaurants from the Yelp dataset

	All businesses	**All restaurants**	**Restaurants** **r.useful > 3** **r.cool > 2** **r.stars > 3** **b.review_count > 5**
Review	1,127,525	706,404	22,584
Business	42,153		
User	252,898		
Tip	403,210		
Checkin	31,617		

Below is the code we use to filter the reviews data, to choose our 500 reviews on restaurants with the business review count bigger than 5.

Table 4-13: Hadoop Hive query for filtering the restaurant reviews data, to choose our 500 reviews

Hadoop Hive filter query
SELECT r.business_id, name, r.text, r.review_id, r.date, r.stars, r.funny, r.useful, r.cool, b.review_count FROM yelp.review r JOIN yelp.business b ON (r.business_id = b.business_id) WHERE categories LIKE '%Restaurants%' AND r.useful > 3 AND r.stars > 3 AND b.review_count > 5 AND r.cool > 2 Limit 500;

4.5 Feature-based opinion mining

The goal of this section is:

1. To detect terms at the sentence level of the reviews referring to the constructed restaurant features.

2. To construct word chunk within the sentence for each detected feature.
3. To compute the chunk polarity according to Textblob[17] module updated by empirical restaurant-based polarity words list.
4. To modify/enrich the restaurant profile accordingly.

We secondly conducted a fine-grained sentiment analysis corresponding to each restaurant features as expressed by customers. Automatic term classification of reviews' keywords was used to evaluate our methodology found to perform a recall of 52% and a precision of 67% over a manually labeled excerpt done by myself.

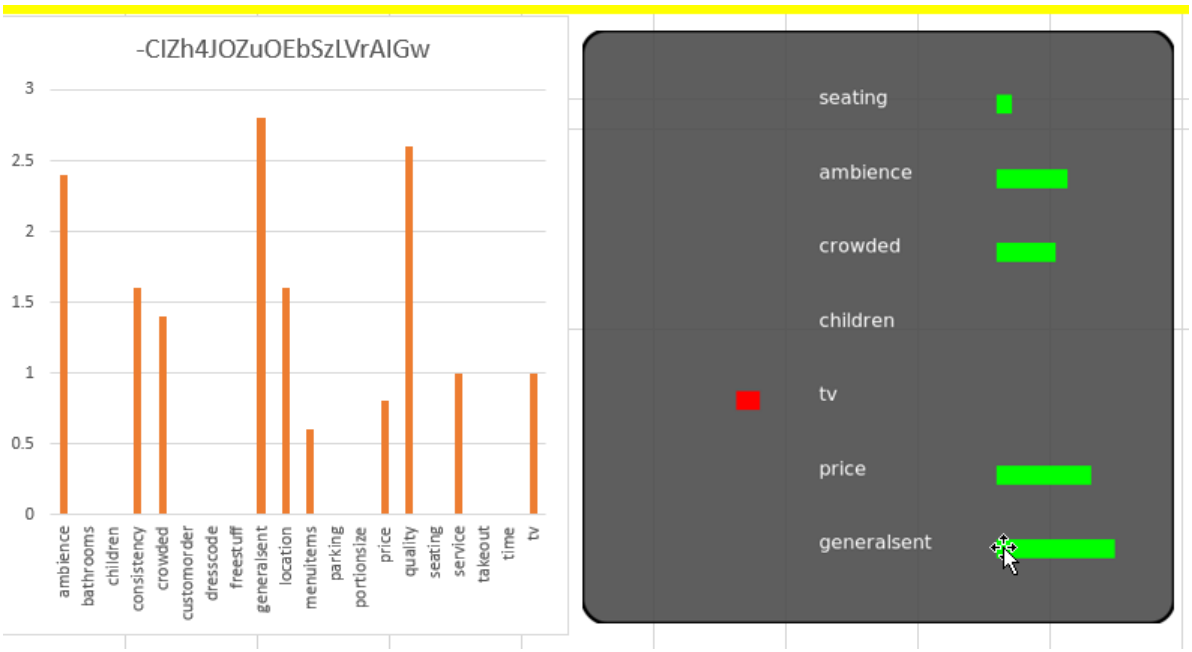


Figure 4-2: The manually labeled test versus the restaurants frame extraction algorithm.

[17] TextBlob is a Python (2 and 3) library for processing textual data. It provides a simple API for diving into common natural language processing (NLP) tasks such as part-of-speech tagging, noun phrase extraction, sentiment analysis, classification, translation, and more.

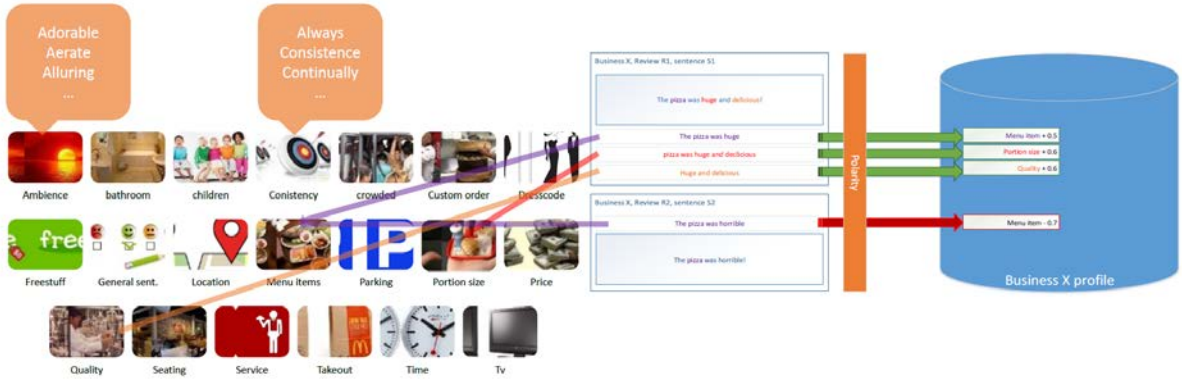


Figure 4-3: Extracting restaurant profile from features

Finally, we implemented a web-based tool capable of extracting restaurant profiles and summarizing the result to the final user.

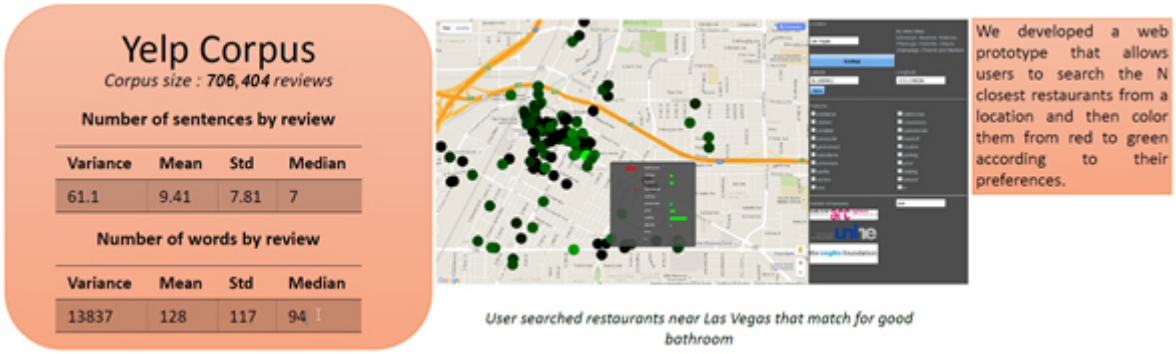


Figure 4-4: Search result for restaurants near Las Vegas that match for good bathroom.

4.6 Qualitative analysis

What do people expect when they go to a restaurant? What are the criteria against which they evaluate their experience? The most basic knowledge that comes into play is that a restaurant is a place that provides the service of cooking and serving food, for which the customers pay a price. Webpages such as Yelp distill this knowledge into a single measure of quality, which is expressed as a number of points or stars, and a single measure of cost, which reflects the prices that a restaurant charge. This is very compared and useful information.

However, it is merely the tip of an iceberg of much more detailed knowledge that could be mined through the analysis of restaurant reviews. For some customers, this information is used as a filter before rating the selected restaurants. In fact, there are different kinds of restaurant that cater to different audiences and different expectations. Depending on the kind of

restaurant that is being reviewed, different criteria are evaluated. Our research therefore aimed for a more comprehensive understanding of the restaurant experience, which led us to propose a list of categories that could matter to customers who write a restaurant review. The first step of the analysis was thus a qualitative approach to the reviews in our database, which consisted of a close reading of a selected subset of reviews and had as its aim the identification of recurring topics that are being evaluated. This analytical step resulted in a list of the following 22 criteria.

1. Food quality – How good is the food that is offered?

   Expectably, almost every review has at least one sentence about food quality, in which the reviewers comment on a specific food item and point out aspects about this item that they find noteworthy.

2. Menu item – What are the dishes that are offered?

   In general, the second most significant part of the reviews is the discussion of menu items. Most often without any evaluation in the sentence that mentions the item.

3. Returning customer – Does the reviewer visit this restaurant often?

   Many more comprehensive reviews include comments to the effect that the reviewer visits the restaurant often or will soon visit it again. This is evidently a sign of quality.

4. Ambiance – What is the feel of the restaurant?

   This is a fairly heterogeneous parameter. Reviewers mention furniture or décor, comment on whether the place is casual or fancy, themed in a certain way, or just very plain.

5. Service – Are customers well attended to?

   The quality of the service is another straight-forward continuous parameter. Many reviewers offer explicit comments that evaluate the service by the waiting staff.

6. Price – How expensive is the place?

   Reviewers commonly point out whether the food that is on offer is cheap or expensive and if it is worth the price.

7. General sentiment – Is this a good restaurant?

Some evaluative sentences are not inherently specific with regard to what makes a restaurant good or bad but still convey a general sentiment that applies to the restaurant that is reviewed.

8. Time – At what time of day do people go to this restaurant?

Some restaurants are lunch places, others are open late. Some restaurants may have different offerings at different times during the day, for instance lunch specials. Depending on the needs of a customer, this is information that can be very valuable.

9. Portion size – How generous are the servings?

Portion size is another frequent topic of discussion. Comments on portion size are quite variable in terms of the words that are used.

10. Consistency – Is the food consistently good?

Not all items on the menu may be good, or not all staffers in the kitchen may be up to the same standards. Reviewers often point out that a restaurant offers consistently good food.

11. Crowded – Are there lots of people or is it usually quiet?

Some reviewers comment on the level of traffic that a restaurant gets. Traffic is an indicator of quality, but sometimes customers try to avoid busy places.

12. Free items – Are customers offered free items?

American restaurants often offer free drink refills, chips, or appetizers. Some reviewers comment on these items.

13. Patrons – What kind of people can you expect to meet?

Some reviewers comment on the people that they see. For customers who are looking for a specific restaurant experience, i.e. a quiet dinner rather than a bar with a party atmosphere, this is crucial information as well.

14. TV – Is there a TV in the dining area?

Sports bars and lowbrow restaurants in the US tend to have TVs, some customers visit these restaurants in order to view sports matches or other particular TV shows.

15. Bathrooms – Are the bathrooms clean and well-maintained?

This can be a source of complaints. Comments vary on a continuum from bad to good.

16. Custom order – Can you order food items that are not on the menu?
This category is relevant for customers with dietary restrictions or who like their order without certain ingredients.

17. Dresscode – What do people wear?
This category is similar to ambiance, but more clearly delimited on the clothing that customers are expected to wear in a given restaurant.

18. Location – What are the surroundings of the restaurant like?
Some reviews include comments on the surroundings of the restaurant, how it can be reached, and closeness to landmarks, shops, parks, bus stops and metro stations.

19. Seating – What are the seating arrangements?
Some restaurants have booths, some have large tables, and some have a bar or a terrace.

20. Takeout – Can food be ordered to be taken out?
Some restaurants offer customers the option of packaging up food so that it can be taken home.

21. Parking – What are the parking arrangements?
Comments by restaurant reviewers sometimes indicate whether a restaurant has designated parking or whether there is valet parking. Especially in an American context, where most costumers arrive by car, this is important information.

22. Children – Is the restaurant child-friendly? Are there menu items for children?
Some reviews include information on how a restaurant caters to the needs and preferences of families with children, for instance by providing highchairs or food items specifically for children.

The above list of 22 categories represent what we call the restaurant frame. Each category describes a part of the experience that is associated with dining at a restaurant. Naturally, not every review relates to all of the categories, but in a large database of reviews, all of them will be represented to some extent. Expectably, certain categories, such as food quality and service, are strongly represented, whereas others, such as parking, are only mentioned rarely.

An obvious challenge in the analysis of restaurant reviews is to detect automatically what category a reviewer is talking about in a given sentence. To facilitate this step, which will be discussed in more detail in the next section, we created word lists for each category, so that for instance the words “car” and “garage” would be included in a list that provides cues for the parking category. The word lists were assembled on the basis of different kinds of key words including nouns, verbs and adjectives. We included key words that were present in the sample of reviews that we read, we added synonyms of these key words. Then we determined words that were overrepresented in natural language data that contained our previously established key words. In this way, each category was associated with lists of about 30 words. This forms what we call later the restaurant lexical database.

4.7 Automatic restaurant feature extraction

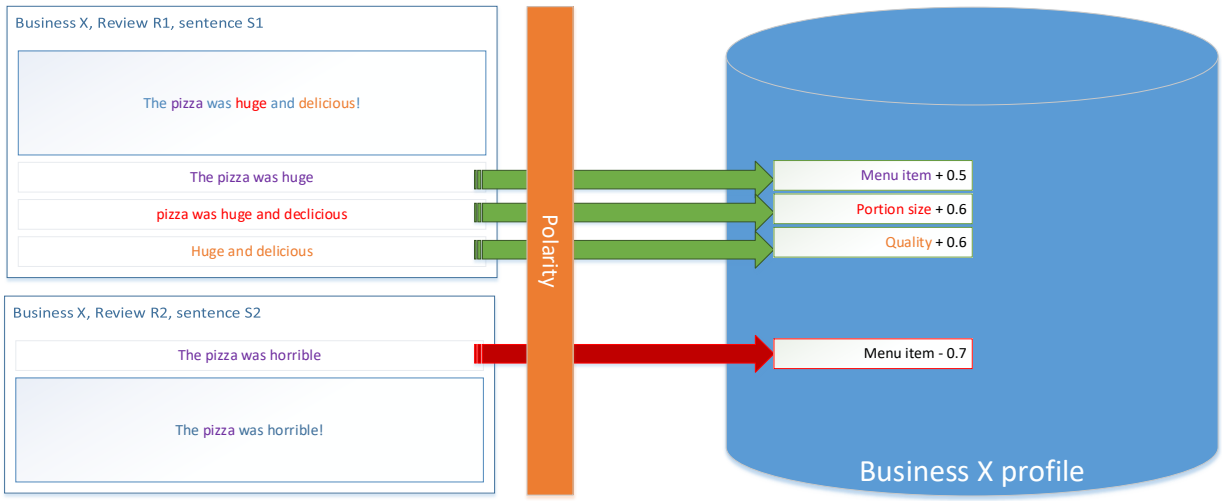


Figure 4-5: Feature extraction and polarity measure

The second computational step is the development of an algorithm that classifies each sentence according to its textual content, so that we end up with separate databases of sentences about food quality, service, ambience, accessibility, and other features described previously in the constructed lexical database. To achieve this goal, we conducted the following steps.

a. We detect terms at the sentence level of the reviews referring to the constructed restaurant features in the lexical database.
b. We construct word chunks within the sentence for each detected feature.

c. We compute chunk polarity according to TextBlob [Loria2014] module updated by empirical restaurant-based polarity words list.
d. We update the restaurant profile accordingly.

In Figure 4-5, we consider two sentences S1 and S2 from two different reviews on the same restaurant. S1 is divided into three separate chunks since three different entries are found in the lexical database corresponding to the terms “pizza”, “huge” and “delicious”. A chunk is constructed from a window of two words from both sides around the term. Each chunk is then associated to the appropriate restaurant feature according to the lexical database entry already found. Polarity measure is applied to all the terms of the chunk from TextBlob module. Adaptation to the restaurant case are annually applied to the polarity of certain lexical entries identified as irrelevant as calculated by TextBlob[18].

4.8 Demonstration: Opinion profile construction tool

In order to demonstrate our algorithm, a web-based interface illustrating the above functionalities are developed. A snapshot of the application is given in Figure 4-6.

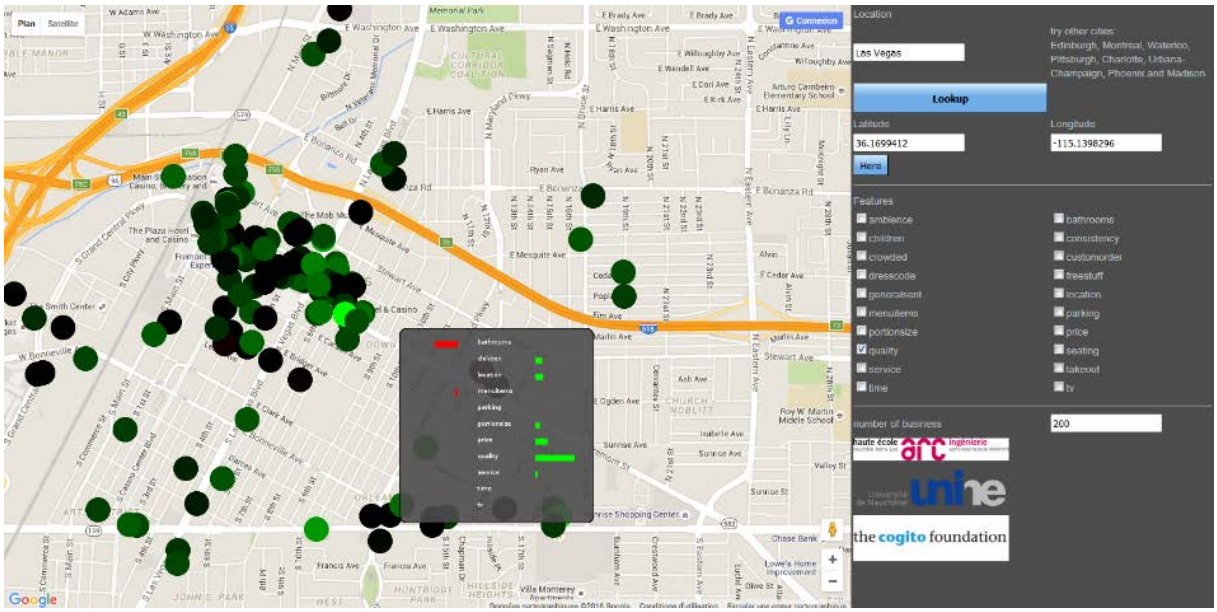


Figure 4-6: Opinion profile construction Web application

[18] https://textblob.readthedocs.io/en/dev/index.html

4.9 Evaluation and dissemination

Here we focuse on the evaluation of the restaurants frame extraction algorithm with manually labeled test corpus.

As a measure of quality, we choose a list of terms which is sufficiently abundant to cover as much as possible the related features. In all, to provide the opinion profile 706,404 restaurant reviews are automatically analyzed. We compared the opinion profiles of a random sample of 10 restaurants constructed by our tool to manual construction over a sample of 260 reviews. We denote that data as the test corpus. We used the metrics of recall and precision to measure the fraction of relevant sentiments that are extracted, and the fraction of extracted sentiments that are relevant, respectively.

	ambience	bathrooms	children	consistency	crowded	customorder	dresscode	freestuff	generalsent	location	menuitems	parking	portionsize	price	quality	seating	service	takeout	time	tv
-CIZh4JOZuOEbSzLVrAIGw																				
Greatest place on the mountain! Wonderful barstaff, great view, play pool, gaming, or have a party!	4								5	5					4					
Great tacos!!															4					
I've been going to this place for years It's definitely one of the best kept secrets in Sunrise Manor. Great atmosphere, great food, affordable drinks. It's not overcrowded and the employees and clientele are very cool. I'll continue to head to the "Cas" whenever I get a chance!	5			4	4				5					4	5		5			
Sunrise Casablanca is a nice little bar tucked away on the East Side of Las Vegas (I think it's actually apart of Sunrise Manor, NV but whatevs). I like to cruise by here to relax, sit at the bar, and watch a game on the TVs hanging above the bar itself. Not too crowded when I go in and still not crowded by the time I leave.. just how I like it. They have a full menu of bar food and also have Black Butte Porter, which is one of my favs, on tap. Nice bar, nice people.. definitely stop in to try it when you're in my neck o' the woods. Definitely a good time.	3			4	3				4	3	3									5
	2.4	0	0	2	1	0	0	0	3	2	1	0	0	1	3	0	1	0	0	1

Figure 4-7: Example of manually labelled text

Figure 4-8 shows a case example of the difference in restaurant profile extraction between human annotator and our tool. We notice a kind of conformance in “generalsent” and “location” features, however minor deviance in the rest of features.

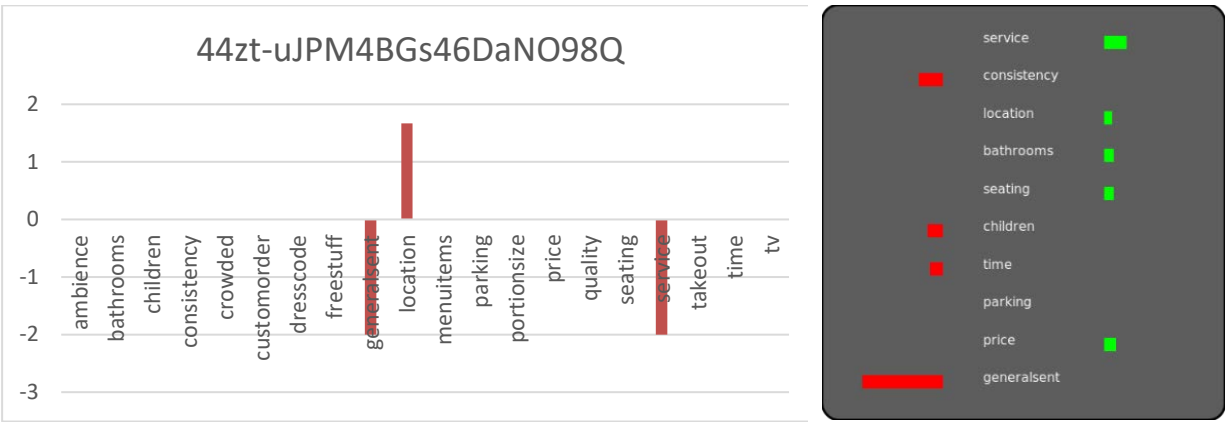


Figure 4-8: Manual feature annotation (left) vs automatic profile extraction (right) of a given restaurant

Results in column 1 of the Table 4-14 show recall and precision at the level of feature classification and polarity of such features. A ratio of 52% of recall indicates that 52% of the manually known features are extracted by the tool. Whereas a ratio of 67% of precision indicates that among all the detected features, 67% are conform to the manual features and hold the same polarity.

Columns 2 and 3 of the Table 4-14 depict the same recall and precision measures but separately on positive reviews and negative one, respectively. Results show that it is much easier to detect positively polarized features than negatively polarized ones. The main reason for that is that the sample we considered is found to be positively biased.

Table 4-14: Evaluation of the test corpus

Global evaluation over the test corpus

Business_id	Precision	recall
-CIZ...	0.4	0.57
1CfO...	0.71	0.5
44zt...	0.67	0.22
dgGp...	0.79	0.65
lxQ1...	0.88	0.7
kJ2a...	0.75	0.6
LIAF...	0.58	0.44
Oi8l...	0.79	0.61
smoG...	0.64	0.45
wMzo...	0.5	0.47
Mean	**0.67**	**0.52**

Do the restaurants have the *same* polarity and categories?

Evaluation over the positive reviews of the test corpus

Business_id	Precision	recall
-CIZ...	0.4	0.67
1CfO...	0.67	0.8
44zt...	1	0.2
dgGp...	0.85	0.85
lxQ1...	0.81	0.72
kJ2a...	1	0.58
LIAF...	0.67	0.5
Oi8l...	1	0.61
smoG...	0.75	0.4
wMzo...	1	0.53
Mean	**0.81**	**0.59**

for matching categories, what are the part of goodly polarized as positive.

Evaluation over the negative reviews of the test corpus

Business_id	Precision	recall
-CIZ...	0	0
1CfO...	1	0.2
44zt...	0.5	0.25
dgGp...	0	0
lxQ1...	0	0
kJ2a...	0	0
LIAF...	0.33	0.25
Oi8l...	0	0
smoG...	0.17	0.2
wMzo...	0	0
Mean	**0.2**	**0.09**

for matching categories, what are the part of goodly polarized as negative.

Finaly we try to find whether a sentence is Positive or Negative by following the next steps:

1. Retrieving the Parts of Speech (PoS) (verbs, nouns, adjectives, etc.) from the sentence using the Stanford NLP parser.
2. Using the SentiWordNet to find the positive and negative values related to each PoS.
3. Summing up the positive and negative values obtained to calculate a net positive and net negative value related to a sentence.

Table 4-15: Example of net positive and net negative calculation of a sentence from SentiWordNet

For example, from the sentence "The pizza was huge and delicious."				
# POS	ID	PosScore	NegScore	SynsetTerms
n	7873807	0	0	pizza#1
a	1387319	0	0.125	huge#1
A	1807964	0.75	0	delicious#1

The positive and negative values of two words huge and delicious in SentiWordNet are resulting:

Table 4-16: Sentiment Analysis results from SentiWordNet

Polarity	**Net positive**	**Net negative**
The text is pos.	**0.75**	**0.125**

In order to manipulate the dataset, we used Big Data tools such as Hadoop Hue and Hive for data storage and filtering.

Table 4-17: The list of terms for the twenty features describing the restaurant profile as constructed from Yelp

Feature	**Terms at the sentence level**	**Term count**
Ambience	adorable, aerate, alluring, clean, ...	153
Bathroom	bathroom, antibacterial, restroom, toilet, washroom, ...	53
Children	kids, safety, family, baby, playground, …	104
Consistency	always, consistence, continually, ...	22
Crowded	full, cramped, busy, noisy, ...	17

Custom order	adapted, custom, made-to-order, personalized, special, tailored, ...	8
Dress-code	baggy, black, casual, chic, clothes, code, drawer, dress, ...	51
Free stuff	complimenting, endless, free, freebies, gratis, refill, ...	10
General sent.	admirable, amazing, astonishing, astounding, best, brilliant, …	34
Location	airport, area, beach, bus, center, central, city, corner, downtown, ...	50
Menu items	appetizer, apple, apricot, asparagus, aspic, avocado, bacon, bagel, ...	546
Parking	ample, automobile, car, covered, driveway, driving, fee, garage, ...	35
Portion size	big, devoured, huge, portion, small, …	14
Price	affordable, cheap, cost, expensive, high-priced, inexpensive, price, ...	13
Quality	awesome, delectable, delicious, delight, eatable, enjoyed, flavorful, ...	29
Seating	armchair, bar, bench, divan, sofa, terrace, ...	32
Service	accommodating, approachable, attentive, caring, friendly, ...	51
Takeout	takeaway, aluminum, bag, carry-out, container, plastic, styrofoam, ...	16
Time	24-7, afternoon, all-night, am, breakfast, clock, dawn, daybreak, ...	25
Tv	angeles, baseball, basketball, cleveland, coach, detroit, …	33

4.10 Summary

In this chapter we study the “feature-based opinion analysis”. We explain how to conduct a fine-grained sentiment analysis corresponding to each restaurant features as expressed by customers, by detectting terms at the sentence level of the reviews referring to the constructed restaurant features. The restaurant reviews are analized to detect automatically what category a reviewer is talking about in each sentence. Finally, we demonstrate the use of our algorithm to perform automatic feature extraction, its classification and polarity detection.

In chapter five the Social Network Analysis, opinion mining of reviews text and the friendship information from the Yelp dataset provided on users, are used to determine the strength of the relationship between customers. Finaly we create few graphs representing the Yelp dataset friendships network, to analyze and to depict friendship patterns.

Chapter 5 Social Network Analysis

5.1 Introduction:

Getting users to follow their accounts is a primary objective for many domains with a social media presence such as advertisement, community health campaigns, administrative science, business and even politics.

The huge quantities of data available in a social media, and the many interrelationships among those data offer opportunities for extracting large amounts of valuable information about their structure, evolution, and internal processes. Unfortunately, the huge volume of that information renders it almost unusable without applying methodologies which high light the relevant information for a given aspect such as social connections between users.

Social media seems to be growing by leaps and bounds every day. Almost with each new social network a unique terminology is created. A collection of specific terms used for the interaction between users, the sharing of information, collaboration, opinions, and fun. If we are not familiar with the specific terminology associated with a social network, it is nearly impossible to analyze the related data and to derive meaningful conclusions. So before going further, we provide the definition of some of the most important terms relevant to our social network analysis.

5.1.1 Who is a social media follower?

In a social network, a follower represents a user who has chosen to subscribe to the account of another user (a personality or a brand), and to see all of that user's posts in his content feed. Generally, a follower supports and admires a particular person or his set of ideas. We find this term especially on Twitter and Instagram social networks unlike Yelp, Facebook or Linkedin whose networks are respectively friends and contacts.

We often measure the level of leadership and the popularity of a person or a company to its number of followers.

5.1.2 Who is a social media influencer or leader?

A social media influencer is a person who has established credibility in a specific area, has access to a huge number of followers and fans, and can persuade them to act based on his recommendations. An influencer has the tools and authenticity to attract many viewers consistently and can motivate others to expand their social reach. An influencer may be anyone from a blogger to a food critic, to a celebrity or to a political leader.

5.1.3 What is influencer marketing?

The Internet opened new business perspectives for professionals all over the world. Social media influencers have dedicated and engaged groups of followers on social media.

Influencer marketing has become a new form of collaboration between a business and an influential person to promote something. It could be a product, service, or campaign. It offers the ability to generate extra income for an average person such as a social media influencer.

There is a direct relationship between their income and the number of followers and fans the influencer has, the amount of engagement their posts generally garner, and the fit of the advertisement with their brand and following.

5.1.4 Why detect a leader in a social network?

In the digital era, social media gives the opportunity for businesses to foster and sustain huge communities of online users around their product and services. Thus, identifying social media leaders and followers is important because leaders capable of generating persuasive reviews can change the attitudes of followers and make them accept the opinion provided, influencing their intention to purchase a given product.

5.1.5 How to detect a leader in Yelp?

To detect an influencer, we must take into account many key factors, a) mainly the number of people who are following him: the number of friends on Yelp, likes on Facebook, followers on Twitter, and connections on LinkedIn, b) the quality of his followers: Are the right people following him? He may have many followers, but if only a few reflect his target market, his

efforts become valueless, c) the number of people that are engaged by him: An influencer can have many followers and only be engaging a very small number. An engaged follower's value increases by ten times.

5.2 Friendship links between Yelp users

Our objective in this part of research is to use the friendship information from the Yelp dataset provided on his customers, to determine the strength of the relationship between customers. Based on this information, we can weight more precisely the different customers' reports, assuming that reviews written by leaders will have a stronger impact than those written by followers. Moreover, after being able to identify the leaders, we plan to determine how leaders can improve their status or leadership.

For this purpose, first, we search the users' database to establish a complete list of all friendship links between users. Thanks to this information, not only we can determine the exact number of each user's friends, but we can also verify if all users in the database are included in our complete list of friendship links or not. For example, we can identify those users with no friendship links at all.

We retrieve the information and data from Yelp academic dataset. As a result, we obtain a list of 1,911,998 of entries. Each entry contains two columns. The first with the user identification of a person and the second with the user identification of one of his friends. This means that a user with "n" friends is listed on "n" entries, each time with only one of his friends' "user identification".

By analyzing of the results list we can observe that the list contains bidirectional friendship links. This means that when in a line of the list we observe that A is a friend of B, always in another line we can also find that B is a friend of A.

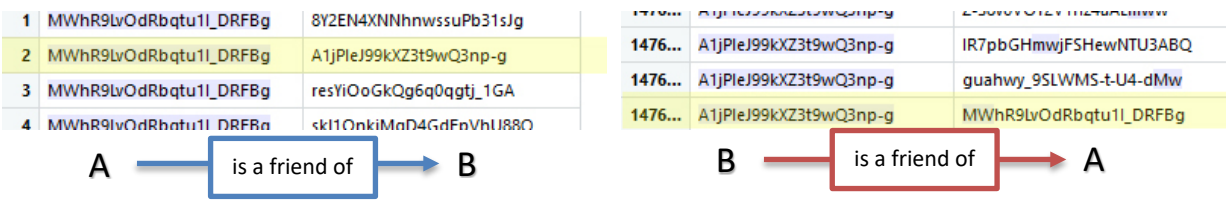


Figure 5-1: Example of bidirectional friendship in Yelp dataset

This brings us to the conclusion that if we take the “user identification” of a user as a node, the maximum number of potential nodes in this list can be the half of the number of all links which is 955,999. However, this could be true if each user had only a unique friend relationship with one other user.

Here comes another question to be answered. Does our friendship-links list covers all the users listed in the Yelp academic dataset? Otherwise saying, are there also users in the database with no friends at all? This is a question that can be answered by keeping and counting all unique “user identification” of users in our list of friendship links and comparing them with the number of users existing in the main dataset.

In our Yelp dataset we have 252,898 users in total. Exactly 129,530 of users, or 51.22 % of them, are friend with no one (or we do not have information about their friends in our dataset), and so these users cannot be used for the analysis of friendship.

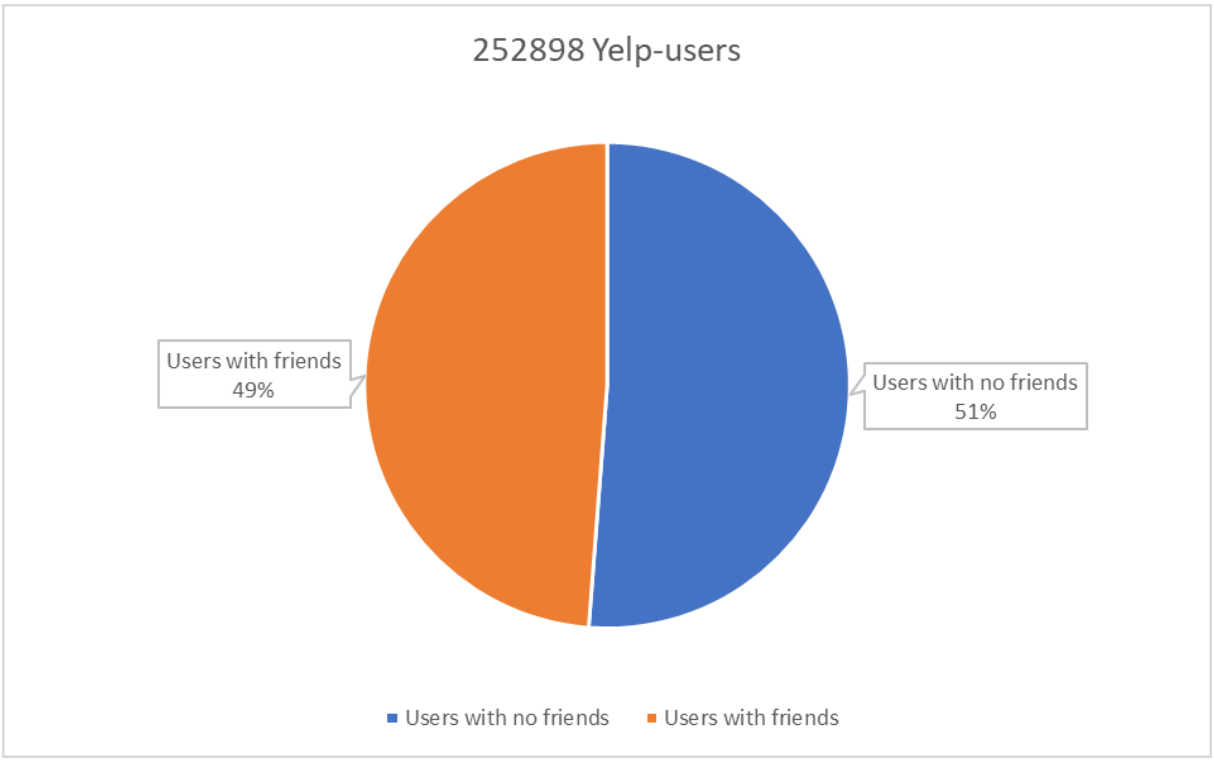


Figure 5-2: Graphical representation of the Yelp dataset user's friendship percentage

To continue our analysis, we are interested in the friendship links between users who have reviewed the same restaurant. To start analysis this hypothesis we plan to investigate few examples, first we choose the restaurant most reviewed in the dataset.

This restaurant is *Mon Ami Gabi* in Las Vegas with 4,084 reviews. Since a user may have written more than one review for this restaurant, we create a list of unique users who have commented this restaurant. This leads us to a list of 3,615 unique users. We find that the maximum number of reviews written by the same user for this restaurant is eight. Only one user has produced eight reviews, one has four reviews, and four users have each three reviews for this restaurant. Then we find 62 users with only two reviews each and all the rest of users, thus 3,547 users have only one review.

Table 5-1: Number of reviews per unique user who commented "Mon Ami Gabi" in Las Vegas

Mon Ami Gabi in Las Vegas with 4084 reviews		
Number of users	**Number of reviews**	**% of total 3,615 users**
1	8	0.028%
1	4	0.028%
4	3	0.111%
62	2	1.715%
3547	1	98.119%

By comparing the list of 3,615 unique users who have reviewed *Mon Ami Gabi* to the general list of friendships which contains more than 1.9 million links, we can create a complete list of 11,570 friendship links of all users who have reviewed this restaurant.

At this step, we have a list of users who are friends and have reviewed, at least once, *Mon Ami Gabi* restaurant. However, to improve the result and to also reflect the social tie strength between two users (weak links, compared to strong links), it is necessary to add a weight factor for each friendship links.

To find which value is more significant to demonstrate how strong a friendship link is between two Yelp's users, and can be used as weight factor, we propose the following two approaches.

5.2.1 Common number of reviews for the same business as weight

As the weight, we can take the common number of reviews made by two friends for the same restaurant. Here is an example:

Table 5-2: Common number of reviews made by two friends as weight

	Number of reviews for the same business
User	15
His friend	1
Common number of reviews made by two friends	1

In this case, we observe that for more than 98% of cases, the common number of reviews made by two friends is equal to “one”. Thus, this value cannot be used as weight because it is not significant enough to show the friendship strength. Of course, no statictics can be computed with a simulation.

5.2.2 Average number of reviews for the same business as weight

Alternatively, as the weight, we can also take the average number of reviews made by two friends for the same business. In this case, we obtain numbers from 1 to 8.

Table 5-3: Avregae number of reviews made by two friends as weight

	Number of reviews for the same business
User	15
His friend	1
Avregae number of reviews made by two friends	8

This second value is much more significant and useful as weight. It can be used to show the friendship strength. In the following example, the avregae number of reviews made by A and B for the same restaurant is 4. So, we consider that the friendship link between A and B is stronger than the link between A and C or D.

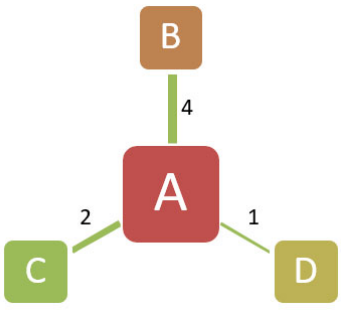


Figure 5-3: Average number of reviews made by two friends used as their friendship link weight.

5.3 Friendship patterns

To analyze and to find friendship patterns, we create few graphs representing the Yelp dataset friendships network. For each link, the identification of a user is used as the source node, and the identification of his friend is used as the target node. Then, the average number of reviews for the same business is used as the weight of their friendships link (edge weight). We use a software called Gephi[19] to calculate the user relationship and its sterenth according to the average number of their comments on the same restaurant.

Nodes: 526	Edges: 606
Modularity: 0.802	Average Degree: 1.152
Network Diameter: 7	Average Path Length: 2.925

Figure 5-4: graph representing the Yelp dataset friendship network of 526 Nodes.

[19] a visualization and exploration software for all kinds of graphs and networks. https://gephi.org/

In the next example, we repeat the same analysis for some other businesses with a bigger number of reviews and so with a longer user's friendship list.

Nodes: 1,252	Edges: 9,038
Modularity: 0.477	Average Degree: 7.22
Network Diameter: 10	Average Clustering Coefficient: 0.149
Average Path Length: 3.564	

Figure 5-5: Graph representing the Yelp dataset friendship network of 1,251 Nodes.

As we can easily observe in the above two graphs, the higher the number of edges, the less the graph is understandable.

Finally, to produce the Figure 5-6, we use 12,000 edges. However, the result is not analyzable at all and in addition the software takes a very long computation time to produce this graph. As conclusion, these results are not useful enough to let us explore the friendship information further in a conclusive perspective.

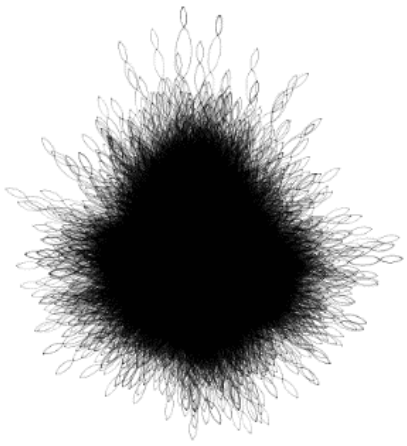


Figure 5-6: Graph representing the Yelp dataset friendship network with 12,000 edges.

In the above examples the modularity, network diameter, average path length, Average degree, and average clustering coefficient is calculated using Gephi[20]. Below is a brief description of each measure:

5.3.1 Modularity

Modulaity measures how well a network decomposes into modular communities. A high modularity score indicates sophisticated internal structure. This structure, often called a community structure, describes how the network is compartmentalized into sub-networks. These sub-networks (or communities) have been shown to have significant real-world meaning. Randomizing the algorithm can produce a better decomposition resulting in a higher modularity score, however randomizing will increase computation time.

5.3.2 Network diameter

The diameter of a network refers to the length of the longest of all the computed shortest paths between all pair of nodes in the network. In other words, the diameter is the maximal distance between all pairs of nodes. (i.e. How far apart are the two most distant nodes).

[20] https://gephi.org/

5.3.3 Average path length

The average path length is the average graph-distance between all pairs of nodes. Average path length is one of the three most robust measures of network topology, along with its clustering coefficient and its degree distribution.

Some examples are the average number of clicks which will lead you from one website to another, or the number of people you will have to communicate through, on an average, to contact a stranger. It should not be confused with the diameter of the network, which is defined as the longest geodesic, i.e., the longest shortest path between any two nodes in the network.

The average path length distinguishes an easily negotiable network from one, which is complicated and inefficient, with a shorter average path length being more desirable. However, the average path length is simply what the path length will most likely be. The network itself might have some very remotely connected nodes and many nodes, which are neighbors of each other.

In a real network like the Internet, a short average path length facilitates the quick transfer of information and reduces costs.

5.3.4 Average degree

The degree of a node in a graph is defined as the number of edges that are incident on that node. Produces distribution plot of node in-degrees (user “popularity”).

5.3.5 Average clustering coefficient

The clustering coefficient, when applied to a single node, is a measure of how complete the neighborhood of a node is. When applied to an entire network, it is the average clustering coefficient over all of the nodes in the network.

The clustering coefficient, along with the mean shortest path, can indicate a "small-world" effect. For the clustering coefficient to be meaningful it should be significantly higher than in version of the network where all of the edges have been shuffled.

The neighborhood of a node, u, is the set of nodes that are connected to u. If every node in the neighborhood of u is connected to every other node in the neighborhood of u, then the neigh-

borhood of u is complete and will have a clustering coefficient of 1. If no nodes in the neighborhood of u are connected, then the clustering coefficient will be 0.

5.4 Identifying influencers and finding their degree of leadership

At one point, the common assumption was that people with lots of followers on social media were automatically influencers.

To find leaders among the people who have written reviews for the same restaurant we choose the top three restaurants that have the highest number of reviews in the dataset.

1. *Mon Ami Gabi* in NV has 4137 comments but only 88 people have more than one comment, which is 2.1%.
2. *Earl of Sandwich* in NV has 3517 comments but only 86 people have more than one comment, which is 2.4%.
3. *Wicked Spoon* in NV has 3352 comments but only 76 people have more than one comment, which is 2.26%.

For each of these three restaurants we create a file with two columns. The file contains information about the date that a review was written, the user identification, and review identification.

We use user identification and review identification as nodes. To establish links between the nodes, we use information about the comments that were written by each user for the same restaurant. Considering the importance of opinion leadership in social networks, identifying importance of nodes in a network is an active problem. In order to identify an opinion leader by using social network analysis, the study used network centrality measures for degree centrality, closeness, and betweenness. Below is a brief description of each measure:

5.4.1 Centrality

In graph theory and network analysis, indicators of centrality identify the most important vertices within a graph. Applications include identifying the most influential person(s) in a social network, key infrastructure nodes in the Internet or urban networks, and super-spreaders of disease.

5.4.2 Degree centrality

Degree centrality is defined as the number of links incident upon a node (i.e., the number of ties that a node has). The degree can be interpreted in terms of the immediate risk of a node for catching whatever is flowing through the network (such as a virus, or some information). In the case of a directed network (where ties have direction), we usually define two separate measures of degree centrality, namely indegree and outdegree. Accordingly, indegree is a count of the number of ties directed to the node and outdegree is the number of ties that the node directs to others. When ties are associated to some positive aspects such as friendship or collaboration, indegree is often interpreted as a form of popularity, and outdegree as gregariousness.

5.4.3 Closeness centrality

In a connected graph, the normalized closeness centrality (or closeness) of a node is the average length of the shortest path between the node and all other nodes in the graph. Thus, the more central a node is, the closer it is to all other nodes.

5.4.4 Betweenness centrality

Betweenness is a centrality measure of a vertex within a graph. Betweenness centrality quantifies the number of times a node acts as a bridge along the shortest path between two other nodes.

5.4.5 Eigenvector centrality

Eigenvector centrality (also called eigencentrality) is a measure of the influence of a node in a network. It assigns relative scores to all nodes in the network based on the concept that connections to high-scoring nodes contribute more to the score of the node in question than equal connections to low-scoring nodes. Google's PageRank and the Katz centrality are variants of the eigenvector centrality.

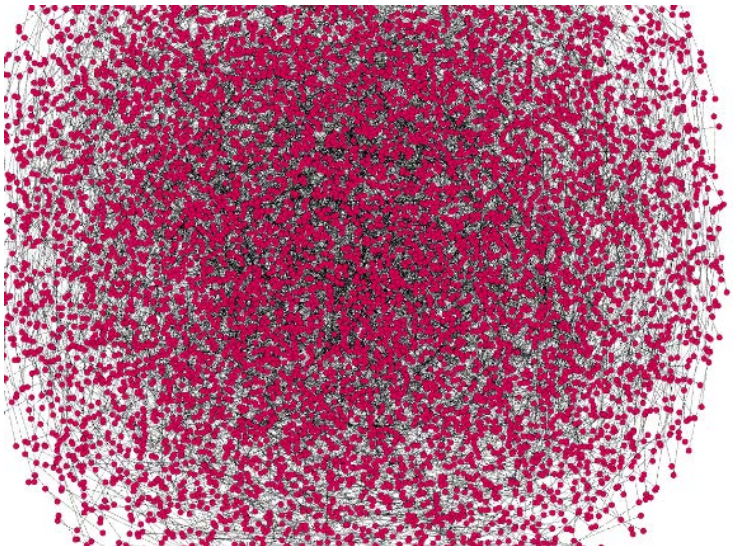


Figure 5-7: A Network of users who posted reviews for the same restaurant

We notice that most of the time people have written a single comment for the same restaurant and there are only 2% of users with more than one comment on the same restaurant.

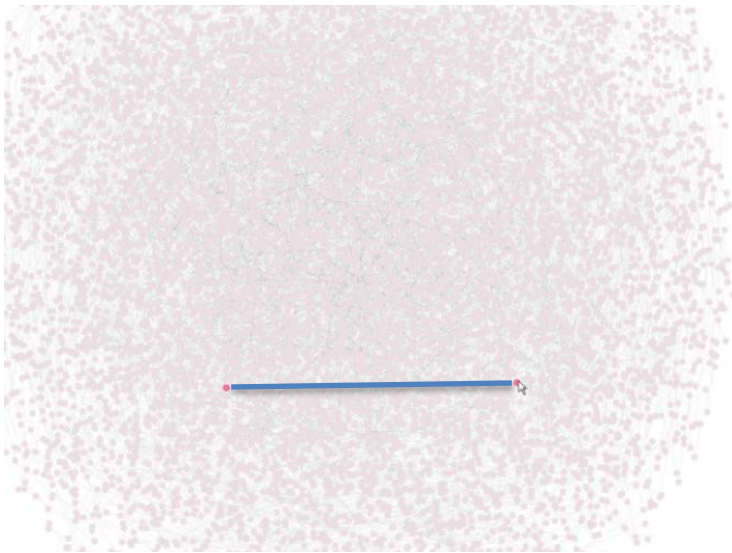


Figure 5-8: Two users with each one comment on the same restaurant

We mix our three files (combining them in the same file) but with different color codes. Surprisingly, we observe that there are some users who have written, on the same date, reviews for two of or even three of these restaurants!

_3xvZsoCjJeqcsrPOlCN_Q	zt1TpTuJ6y9n551sw9TaEg	14/01/2014	6M21aesrivtOGjOJ-j_A4Q
_3zG5xg6-ymGoflTH8VN_Q	4bEjOyTaDG24SY5TxsaUNQ	07/05/2013	zkC8zKn2eAXMaC7S3Bjw5A
_48n732DXmYfrfgq-YlKGw	2e2e7WgqU1BnpxmQL5jbfw	11/01/2012	cV7A34RMEBj48z5n6W3DPQ
_4bvJDHjSn7LwFIky6-ewQ	zt1TpTuJ6y9n551sw9TaEg	27/02/2011	tVh3X4UJiu3YRAXp-Fo8Jg
_4lqpCYCqOQzbB6xQGGhrQ	2e2e7WgqU1BnpxmQL5jbfw	05/09/2012	MR8qyNnICfcGGiK6U5_jhg
_4NhIw5wc9zfmvlZPVWk1w	2e2e7WgqU1BnpxmQL5jbfw	17/09/2011	1lQ1nw390RJDS2Sy_J8FSg
_5NDxrFC8UEe2Wj42kyOkg	4bEjOyTaDG24SY5TxsaUNQ	25/02/2008	KGXayQgVQwo-kYA7P_bdpg
_86XAPZt5S9R1KGXn_lHXw	2e2e7WgqU1BnpxmQL5jbfw	23/11/2011	7kMzwQ9kaFimLxOr6p457Q
_972suHZy7F6a8__Snolew	4bEjOyTaDG24SY5TxsaUNQ	15/05/2014	OAnXFt4H-RX_1QOzjl3-hg
_9Sj8L_XbVgSmExcvDLHiQ	zt1TpTuJ6y9n551sw9TaEg	07/05/2013	kMiy8Bp0HvKyTcYYhZcVtg
_AmzHunCMB_RzMyrmiWUJw	zt1TpTuJ6y9n551sw9TaEg	06/12/2013	LSwL4bc3OdZDkl7SIV03mQ
_bBX2Gj1n69vjv8MwQMA6g	2e2e7WgqU1BnpxmQL5jbfw	25/06/2013	BZnxKIB51FsXeISXhq_Rlg
_bBX2Gj1n69vjv8MwQMA6g	zt1TpTuJ6y9n551sw9TaEg	25/06/2013	zYmcoXvAlfjzfpoqnjFpbg
_BGuV3vbsrETs-bo8oir9A	zt1TpTuJ6y9n551sw9TaEg	29/12/2013	qmAwzQWdHtsz81fOCwtlwQ
_bHzW18Vg1gxTMoR1fFZhA	2e2e7WgqU1BnpxmQL5jbfw	25/08/2012	yG_ZPN7wxPQG_Gu6n333sA
_bkymE0DMONSAAsUSByDLw	zt1TpTuJ6y9n551sw9TaEg	14/11/2011	DQpBTNkWqsSgQIxhlrsDlA
_BZqAWDj0CuNoq6wf1MPiw	4bEjOyTaDG24SY5TxsaUNQ	17/07/2013	pseRq8Q-BsEngJm-zXPXGQ

Figure 5-9: A combination of information on users and the date they reviewed these three restaurants

After identifying these users by their identification, we check the Yelp dataset to find whether these users have written even more comments on the same date?

The result is even more surprising! One user was able to write 35 reviews the same day. As we can see in the fifth line of the next table, a single user has commented 35 restaurants on 04.02.2014 and then he has commented 38 restaurants the day after on 05.02.2014.

Table 5-4: Example of a same user who has commented many restaurants on the same dates

user_id	Comments for different restaurants	date	Comments for different restaurants	date
1	3	12.06.2014		
1	3	24.06.2013	3	25.06.2013
1	11	29.07.12		
1	15	27.12.2010		
1	35	04.02.2014	38	05.02.2014

By further research we find that some users have commented up to 75 restaurants on the same day, and that is why we decide to investigate opinion spam in reviews.

5.4.6 The opinion spam detection problem

The problem of opinion spam was introduced by investigating supervised learning techniques to detect fake reviews (Jindal & Liu, 2008). Opinions such as online reviews are the main source of information for customers to help gain insight into the products they are planning to buy. Customers write reviews to provide feedback by sharing their experience, either bad or

good, with others. Their experiences impact businesses for the long term either positively or negatively.

We are going to consider few elements to verify whether an opinion is fake or not.

5.4.6.1 Verification of the geographical location of restaurants

First, we check the geographical location of the restaurants that our suspected users with many reviews written either on the same day or in a short period of time. We check if physically it was possible for them to be present at these restaurants and to eat in each of them during the same day or even 2-3 days?

So, the main two questions to be answered are: a) what are restaurants for whom these reviews are written? b) How far these restaurants are located from one another?

Next two figures show the results for one of the users with many reviews written on the same day.

Even Though we can take some conclusions from these results, but this cannot help us to generalize our conclusion and apply it for all the users in the dataset, in order to group reviews to two categoris of fake and not fake!

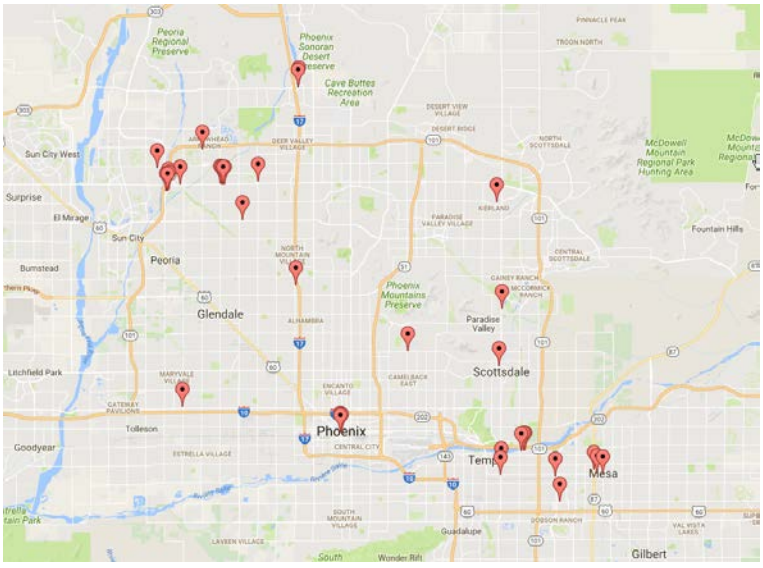


Figure 5-10: The geographical location of the restaurants that the above user has commented on the same dates

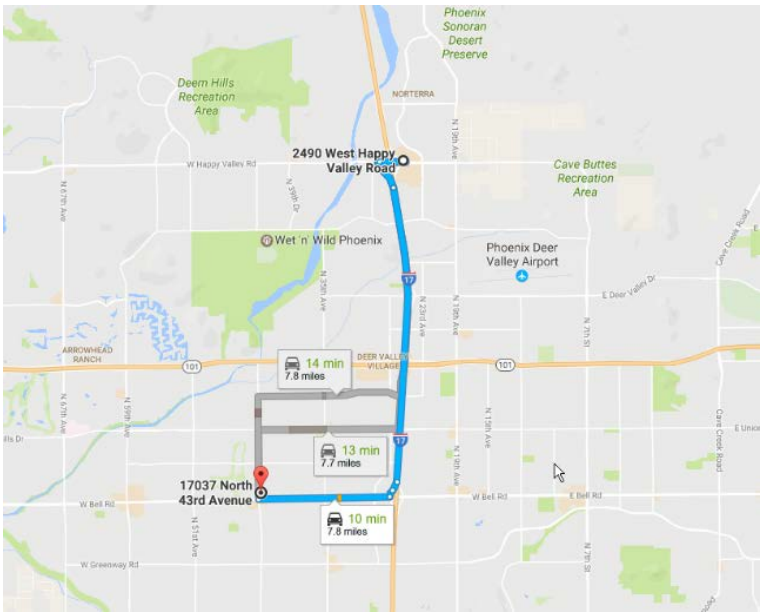


Figure 5-11: Example of distance between two restaurants who has been commented by the same user on the same date

5.4.6.2 ID verification of the customer

The Yelp dataset is providing us with a great part of their data but not all; so, all comments are not included in the dataset and we have only access to 1/9 of all the information on reviews' text.

That is why, we identify those users who have the biggest number of comments and who are also included in our database. Then we check the date ranges between their comments, and we notice that they are much more regular and there are not many comments with the same date, written by the same user! We check the polarity of the opinions and try to detect if a restaurant pays a user for the review.

5.4.6.3 Are these reviews biased?

In this part we try to check if users who have written many comments on the same date, try to support a restaurant (or some kind of restaurant chain) or if they try to discriminate a restaurant or a group of restaurants! To verify this, we check the number of stars given by each to a restaurant at the same time they wrote their review.

Our hypothesis is that in case of discrimination or support, was will have a lot of 1 or 5 stars! Here is an example of a histogram we are expecting in case that a user's reviews are biased:

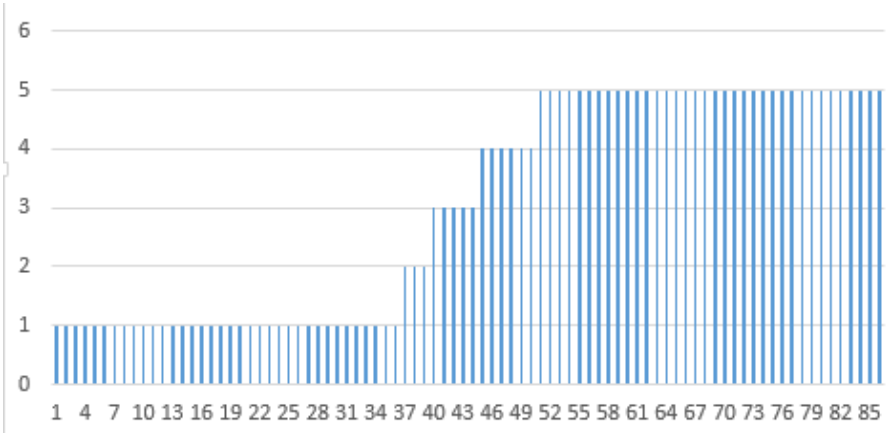


Figure 5-12: example of a histogram in case of discrimination or support. Axe Y: stars and axe X: number of reviews

After verification of star rating of some of these suspected users, the distribution of the number of stars related to their reviews is shown in the following histograms:

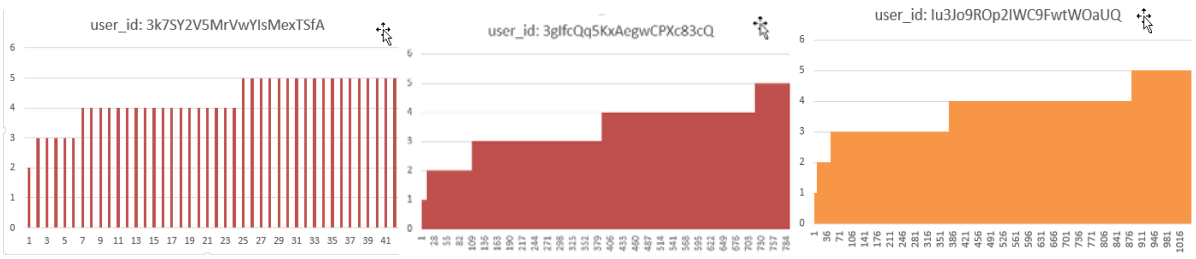


Figure 5-13: Real distribution of the number of stars for three random users

As we can see, the result does not show that these users are trying to give more benefit to a restaurant nor to discriminate them! The distribution of stars is very regular.

5.4.6.4 Kullback–Leibler divergence to identify suspicious users

Divergence of Kullback-Leibler is a measure of dissimilarity between two probability distributions P and Q. For discrete probability distributions P and Q defined on the same probability space, the Kullback–Leibler divergence between P and Q is defined to be

$$D_{KL}(P||Q) = \sum_i p(i) log \frac{p(i)}{Q(i)} \qquad (5.1)$$

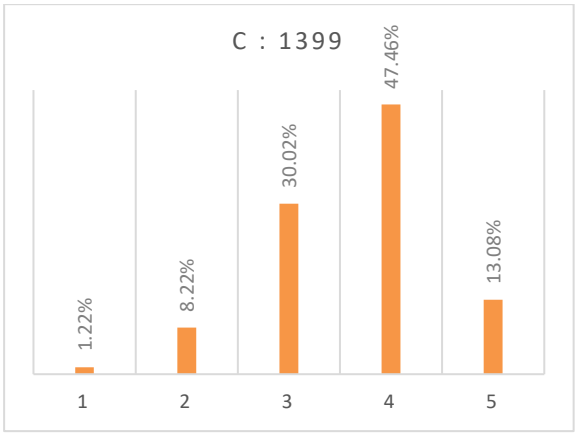
In the simple case, a Kullback–Leibler divergence of 0 indicates that the two distributions in question are identical.

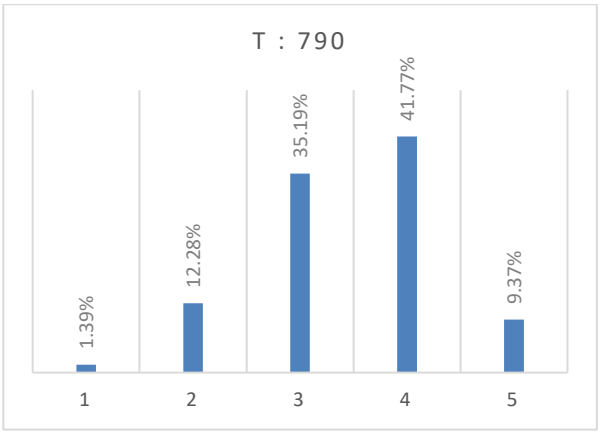
For example, we can consider two of users above. The first user with 1,399 comments (Orange histogram) and the second user with 790 comments (Red histogram). To compare these

two distributions, we take the same number of reviews (the first 790) from each distribution and calculate their KL divergence.

By looking at these two histograms of star distributions, we can say that these two distributions are very similar and so we are expecting to have a very small Kullback–Leibler divergence!

Table 5-5: The KL divergences KL (C || T) and KL (T || C)



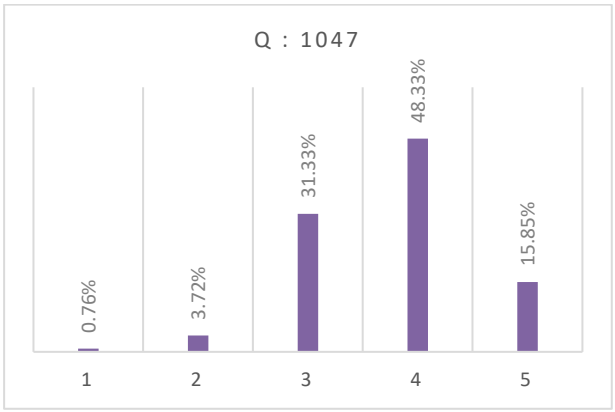


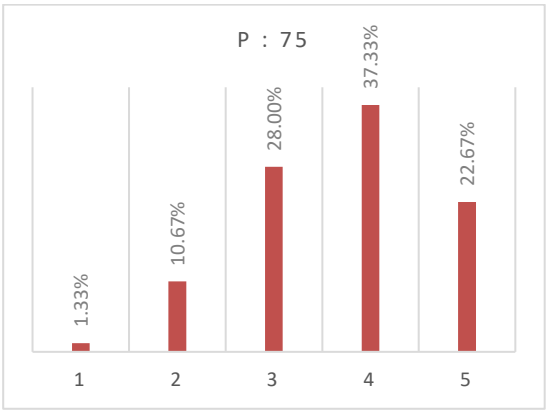
C					**T**			
1399					790			
1	17	1.22%	1.22		1	11	1.39%	1.39
2	115	8.22%	8.22		2	97	12.28%	12.28
3	420	30.02%	30.02		3	278	35.19%	35.19
4	664	47.46%	47.46		4	330	41.77%	41.77
5	183	13.08%	13.08		5	74	9.37%	9.37
	1399	100.00%				790	100.00%	
The KL divergences **KL (C || T)** and **KL (T || C)** are calculated as follows								
KL (C || T)								**3.16726198**
KL (T || C)								**3.23481856**

In order to separate legitimate forms of feedback from false reviews, we continue our investigation and try to find other cases that the use of the KL divergence may be more meaningful. If the result shows a lot of dissimilarity, we can make further research to verify if one of the two distributions represents a fake user.

Here is another case:

Table 5-6: The KL divergences KL (P||Q) and KL (Q||P)





Q					**P**			
1046					75			
1	8	0.76%	0.76		**1**	1	1.33%	1.33
2	39	3.72%	3.72		**2**	8	10.67%	10.67
3	328	31.33%	31.33		**3**	21	28.00%	28
4	506	48.33%	48.33		**4**	28	37.33%	37.33
5	166	15.85%	15.85		**5**	17	22.67%	22.67
	1047	100.00%				75	100.00%	
The KL divergences **KL (P||Q)** and **KL (Q||P)** are calculated as follows								
KL (P||Q)					**11.75**			
KL (Q||P)					**8.63**			

* Because the number of reviews with one star in P was zero, we added one to each group! (P has written all his 75 reviews during only two days)

As a conclusion, we cannot answer the question of "What percentage of Yelp reviews are fake?" In my opinion, the above results cannot be considered as a sophisticated system that easily detects fake reviews. To find if a review is accurately depicting the customer's experience, we require a case-by-case analysis.

5.5 Summary

In chapter five we establish a complete list of all friendship links between users. We are interested in the friends who have reviewed the same restaurant. To indicate the strength of their

friendship link, we calculate the average number of reviews made by two friends for the same restaurant and use this value as the link weight. Finally, to distinguish legitimate forms of feed-back from false reviews, the opinion spam detection problem is investigted.

The chapter six will investigate the sentiment analysis and opinion mining of reviews in a much larger scale. A semi-automatic data pre-processing method is used to correct the whole reviews text. SentiWordNet is used for Sentiment Analysis. The polarity of the first sentence of each review in whole corpus is detected and compared to the polarity of the comment. A further step is taken toward the automatic category detection and extraction for sentiment classification and to predict the star ratings from the review text.

Chapter 6 Sentiment Analysis and Opinion Mining

6.1 Introduction

In chapter 4, we studied the “feature-based opinion analysis”. In this chapter, the research objective is sentiment analysis (SA) and opinion mining in a larger scale.

Sentiment Analysis is a natural language processing problem. Opinion mining, or sentiment analysis refers to the use of text analysis to extract and quantify affective states and subjective information.

Yelp allows consumers to rate and review products and services from small businesses, such as restaurants, they have visited. Opinions and sentiments which are expressed in each review by the customers are important.

The review text is considered as an unstructured data. This text must be analysed to find whether the review expresses a positive or negative opinion about a restaurant or simply an objective or descriptive one. We will ignore the descriptive only comments.

For the analysis of the subjectivity with methods and tools of the sentiment analysis, the classification of the polarity of documents is a familiar method: texts which contain the comments on the products, for example, can be divided in two groups: positive and negative reviews.

There are many questions that deserve our reflection, and to which we can provide both possible and reasonable answers using tools of the natural language processing. Here are some of these questions:

- In general, do consumers use positive or negative words to write their comment?
- Is the sentiment that is expressed in the first sentence of the comment is dominant?

- Do the stars assigned by the user to a restaurant, faithfully reflect his sentiment which is expressed in his review about the restaurant?
- Are there users who are always in a positive attitude? Are the negative reviews eliminated (or reduced) by the site on which the notice was posted?
- Can we automatically recognize the authors? Is there specific “writing patterns”?
- Is there a way to automatically filter fake (malicious) comments and identify their authors, even if the same person uses many different identifiers?

Later in this chapter by using the results obtained from sentiment analysis and opinion mining, we try to answer some of the above questions.

6.2 A lexical resource for opinion mining

Opinion lexicons are resources that associate sentiment polarity for words. A word which indicates the presence of an opinion is a subjective term, and in opposite a neutral word does not indicate any opinion and so is an objective term.

SentiWordNet 3.0 is one of these lexicons that assigns to each synset of WordNet 2.0 a triplet of score (Positive, Negative, Objective) describing how strongly the terms contained in synset enjoy each of the three properties.

The method used to develop SentiWordNet is based on the quantitative analysis of the glosses associated to synsets, and on the use of the resulting vectorial term representations for semi-supervised synset classification. SentiWordNet is freely available for research purposes.

6.2.1 Different platforms are available. We can cite:

- The Hortonworks Sandbox[21] which is a single node implementation of the Hortonworks Data Platform (HDP). It is a personal, portable Hadoop environment.
- H2O[22] on Hortonworks Data Platform which is a fully Open Source Predictive Analytics Platform.

[21] https://www.cloudera.com/downloads/hortonworks-sandbox.html

- Neo4j[23] which is a Graph Database that stores data in a Graph, with Nodes. Neo4j uses Cypher queries to work with graph data.

6.3 Data preparation for data mining

The data preparation (data pre-processing) phase covers all activities to construct the final dataset from the initial raw data in order to prepare the data for further processing. The Yelp dataset contains around 1.2 million text reviews from users on businesses, as well as their rating. All 1,127,525 review texts are used for data mining.

The review texts are generally noisy, and incomplete. It implies that raw data tends to be corrupt, have missing values or attributes, outliers, or common grammatical errors. Data preparation stage resolves such kinds of data issues to ensure the dataset used for modeling stage is acceptable and of improved quality.

6.3.1 Text quality issue

After retrieving all reviews from the Yelp dataset, as it was expected, we noticed that there are many grammatical and orthographic mistakes in the reviews' texts. In fact, for different reasons such as interruption, distraction, or just for being non-native to English, users have committed some minor mistakes in their commentaries. These mistakes which are varying from missing spaces, absent end of sentence punctuations, and spelling faults could reverse the polarity of a review from positive to negative, or inversely.

6.3.2 Review text cleaning

As it is more important to know what the user exactly wants to express in his comments, we decided to correct the text grammatically, even before starting other usual treatments (further processings).

We are expecting that some grammatical correction of mistakes in the customers' reviews would be helpful to obtain more adequate results. For example, because of the missing space after the end of the sentence, the string "good.Food" was considered as one word, and so as an

[22] https://www.h2o.ai/products/h2o/
[23] https://neo4j.com/

unknown term. After adding the missing space, the two words "good" and "food" were considered separately and were therefore easily recognized.

A semi-automatic method is used to correct the whole review text, by adding the missing punctuations or spaces, replacing double spaces, breaking the attached words, and correcting some spelling faults. Then we process again the corrected text. Not a big surprise, the new results are much more adequate. Thanks to this correction:

- As depicted in the last row of Table 6-1, we obtained a 2.59% increase in total of word count for the top 33 most repeated words.
- We count 712,622 distinct strings, before the text correction. After, we find only 390,909 distinct strings. This means that 321,713 strings were broken to meaningful words (or existing strings) and are grouped with them respectively. In other words, the text cleaning reduces approximately 45.14 % the number of unknown words.

Table 6-1: Top 33 most repeated words counted before and after the text cleaning

Word	**Word count before correction**	**Word count after correction**	**Improvement**	**%**
food	557,661	573,716	16,055	2.88%
good	515,914	534,896	18,982	3.68%
place	454,150	462,241	8,091	1.78%
great	336,656	345,898	9,242	2.75%
service	263,465	277,759	14,294	5.43%
time	234,105	240,379	6,274	2.68%
back	213,673	218,019	4,346	2.03%
ordered	174,993	176,834	1,841	1.05%
restaurant	166,994	172,523	5,529	3.31%
chicken	156,702	159,142	2,440	1.56%
dont	154,285	156,901	2,616	1.70%
order	150,557	153,189	2,632	1.75%
menu	149,639	153,736	4,097	2.74%
nice	143,866	147,194	3,328	2.31%
im	140,501	146,569	6,068	4.32%
love	135,558	137,545	1,987	1.47%
ive	128,893	133,669	4,776	3.71%

pretty	126,328	127,103	775	0.61%
didnt	123,830	124,432	602	0.49%
delicious	118,686	124,927	6,241	5.26%
eat	115,388	117,163	1,775	1.54%
pizza	113,425	115,957	2,532	2.23%
sauce	110,294	113,391	3,097	2.81%
vegas	109,388	113,713	4,325	3.95%
cheese	108,041	109,978	1,937	1.79%
bar	103,565	105,906	2,341	2.26%
salad	102,279	104,334	2,055	2.01%
lunch	101,785	104,227	2,442	2.40%
fresh	98,933	100,941	2,008	2.03%
people	97,608	99,367	1,759	1.80%
meal	96,727	101,161	4,434	4.58%
made	94,740	95,215	475	0.50%
make	94,415	95,102	687	0.73%
Total	**5,793,044**	**5,943,127**	**150083**	**2.59%**
Improvement percentage				**2.59%**

6.3.3 Stop-words removal

In computing and sentiment classification, we must also consider stop-words, which are the most common words such as "the", "a", "an", "is", and "at", used in English and which are useful grammatically. However, stop-words do not have enough descriptive power to distinguish the polarity of a document and that is why we filter them before processing the data.

It is important to mention that there is no single universal list of stop-words used by all NLP tools. We use a list of stop-words that contains 571 words (provided by the Cornell SMART system).

After stop-words removal from the whole review’s text, the remained text is composed of 36,861,723 words (tokens) with a total of 375,662 distinct words (string).

6.3.4 Filtering data

The frequency of words differences comments to comments. The most frequently occurring word is “food”. It is repeated 573,716 times. To reduce the number of 375,662 distinct words, we filter the data by removing stop-words, and keep only words appearing at least 10 times.

This takes us to keep only 44,277 words. Surprisingly, these 44,277 words covers 35,495,577 words (tokens) which represent almost 96.3% of total number of words used to compose the whole reviews' texts. Each of 44,277 words in this list, were compared with words in Senti-WordNet. As result, a dictionary of words which were found in SentiWordNet was created. We create two versions of this file. One with of all synsets "SynsetTerms" (#1 = the first sense, #2 = the second sense, …, #N = the Nth sense), that a word belongs to and second with only its most commonly used form (only #1 = the first sense).

Table 6-2: Example of dedicated SentiWordNet dictionary (#1) with for "Steak"

SynsetTerms	PosScore	NegScore	ObjScore	# POS	ID
best#1	0.75	0	0.25	a	227507
bone#1	0	0	1	a	295924
rare#1	0.25	0	0.75	a	488561
cooked#1	0	0	1	a	615757
delicious#1	0.75	0	0.25	a	1807964
good#1	0.75	0	0.25	a	1123148
hot#1	0	0	1	a	1247240
juicy#1	0	0.375	0.625	a	1368793
great#1	0	0	1	a	1386883
nice#1	0.875	0	0.125	a	1586342
thick#1	0	0	1	a	2410393
tough#1	0	0.75	0.25	a	2448437
tender#1	0.5	0	0.5	a	2448889
swiss#1	0	0	1	a	2960975
best#1	0.25	0	0.75	n	127531
large#1	0	0	1	n	5096191
good#1	0.5	0	0.5	n	5159725
small#1	0	0	1	n	5559023
sirloin#1	0	0	1	n	7658958
swordfish#1	0	0	1	n	7785887
nice#1	0	0	1	n	8937251
great#1	0	0	1	n	10145081
best#1	0.5	0	0.5	r	188137
small#1	0	0	1	r	225971
big#1	0.25	0	0.75	r	226054
large#1	0	0	1	r	386393

buffalo#1	0	0.125	0.875	v	1780010
round#1	0	0	1	v	1858910
tender#1	0	0	1	v	2296615
flank#1	0	0	1	v	2694106

These dictionaries which contain all the information needed for opinion mining, were integrated as searchable SQL tables in Hadoop. The dictionary with the most common sense of a word (#1) has 18,977 rows.

Figure 6-1: Word cloud of the top 100 words of the reviews' texts, after stop-words removal

6.4 Statistical data analysis of the Yelp reviews

Preliminary to data mining is data understanding in which data is analysed to get familiar with, and to discover first insights into it. We perform various statistical operations and quan-

titative researches in order to quantify the data and to identify the proper methods to organize and analyse the data, and effectively present the results. As we can depict in Table 3-2 the review database contains full review text including the identification of the user who wrote the review and the identification of the business for whom the review is written.

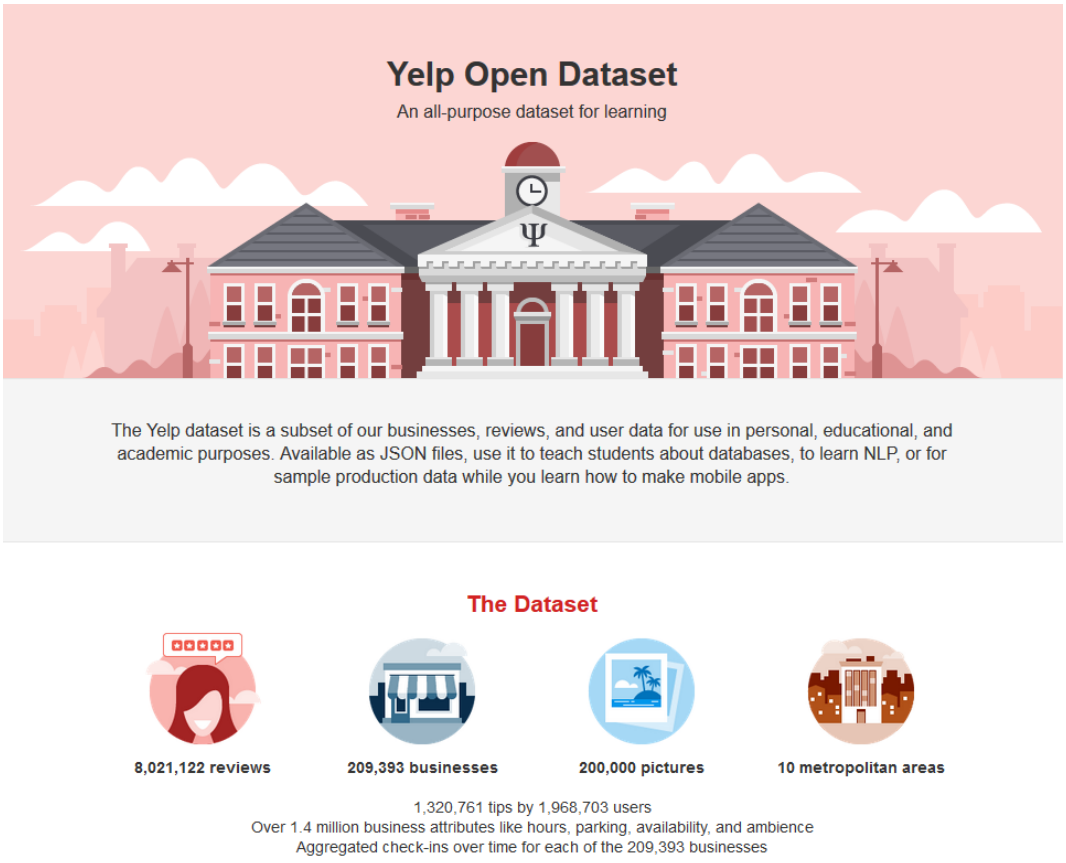


Figure 6-2 : Information about latest version (2020) of the Yelp challenge dataset

Yelp reviews are varying in terms of length, content, writing style and usefulness because they are written by different reviewers. In most reviews, the total number of words is greater in negative comments than in positive comments, (i.e., the total number of words is an important feature to detect the polarity of the review (R.T. Anchiêta et al., 2015)). Here we are planing to predict a business rating based on user-generated reviews texts. That is why we will consider three aspects of reviews and their structure.

- The number of reviews per user
- The number of words per comment

- The number of sentences per comment

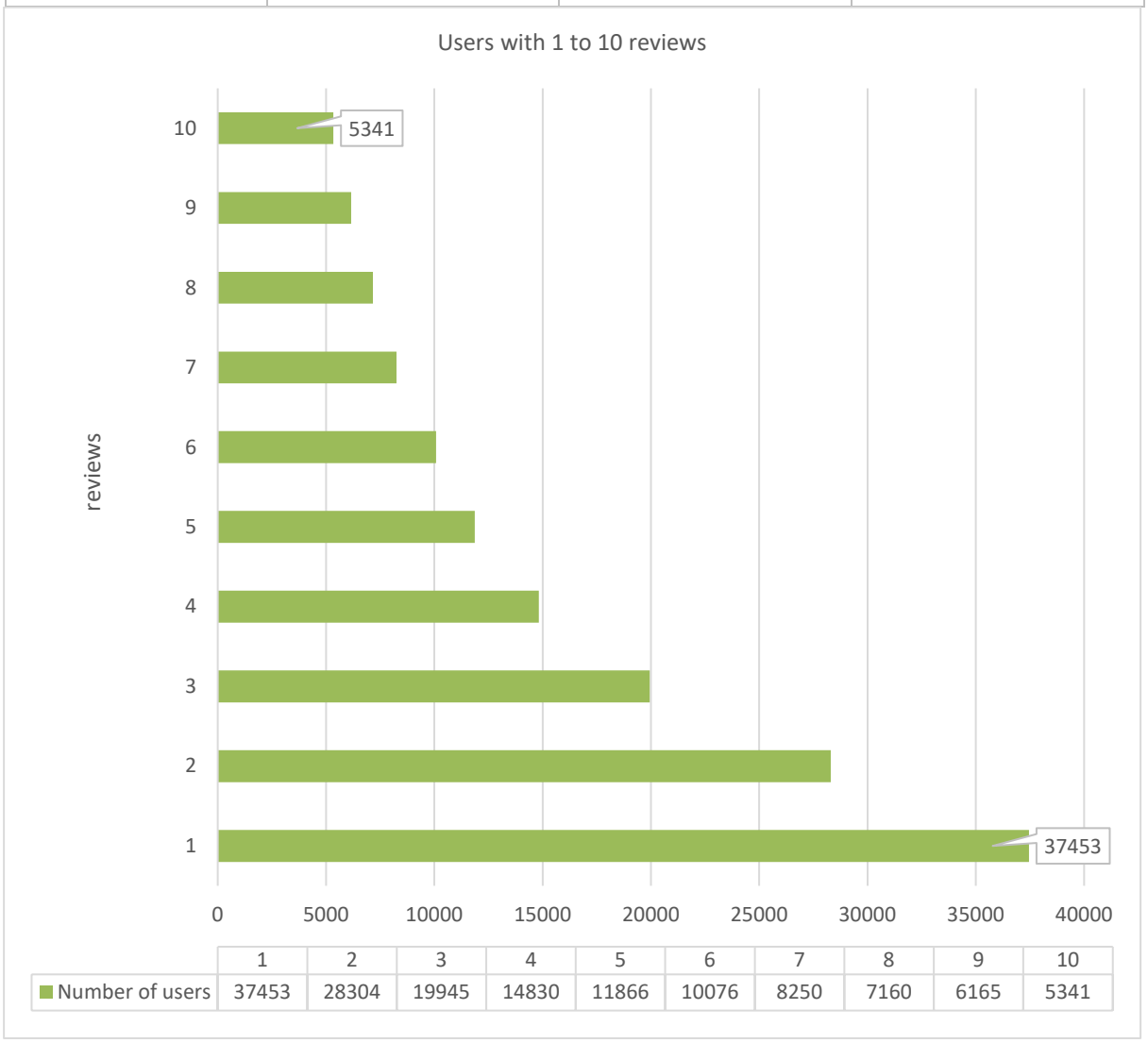
It is important to know that in this chapter we use all reviews in the entire corpus for analysis purposes. The data interpretation of the reviews of 252,898 users is summarized in the following subsections.

6.4.1 Charts and statistics for the number of reviews per user

We have found that 5,341 users have written 10 reviews and 37,453 only one. In addition, we have 202 users with 100 reviews. We obtain an average of 36 reviews per user.

Table 6-3: The analysis of the Yelp review count of users

Variance	Average	The standard deviation	Median
10286.0908	36.2	101.4	7



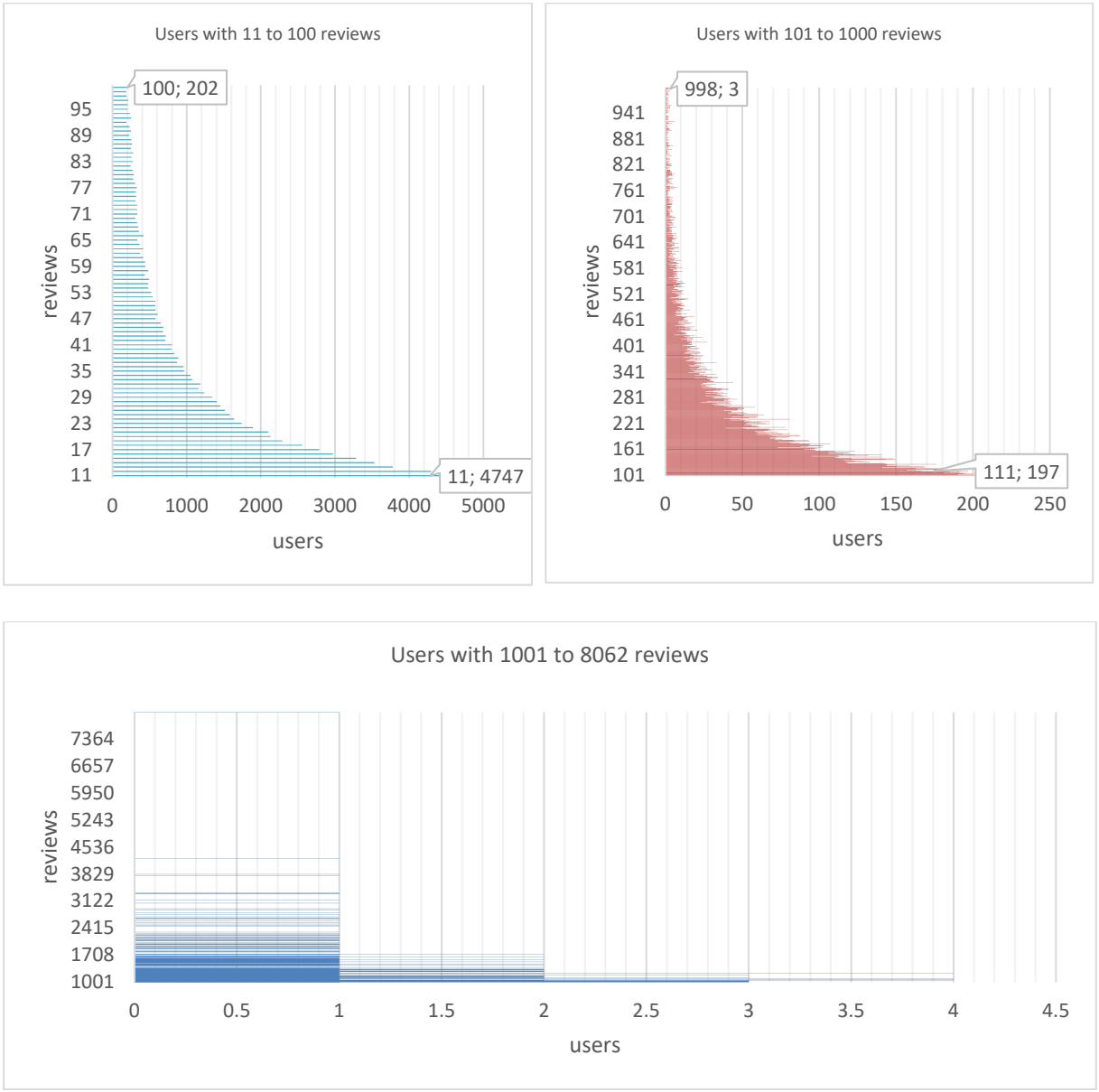


Figure 6-3: Graphical representation of number of reviews per unique user

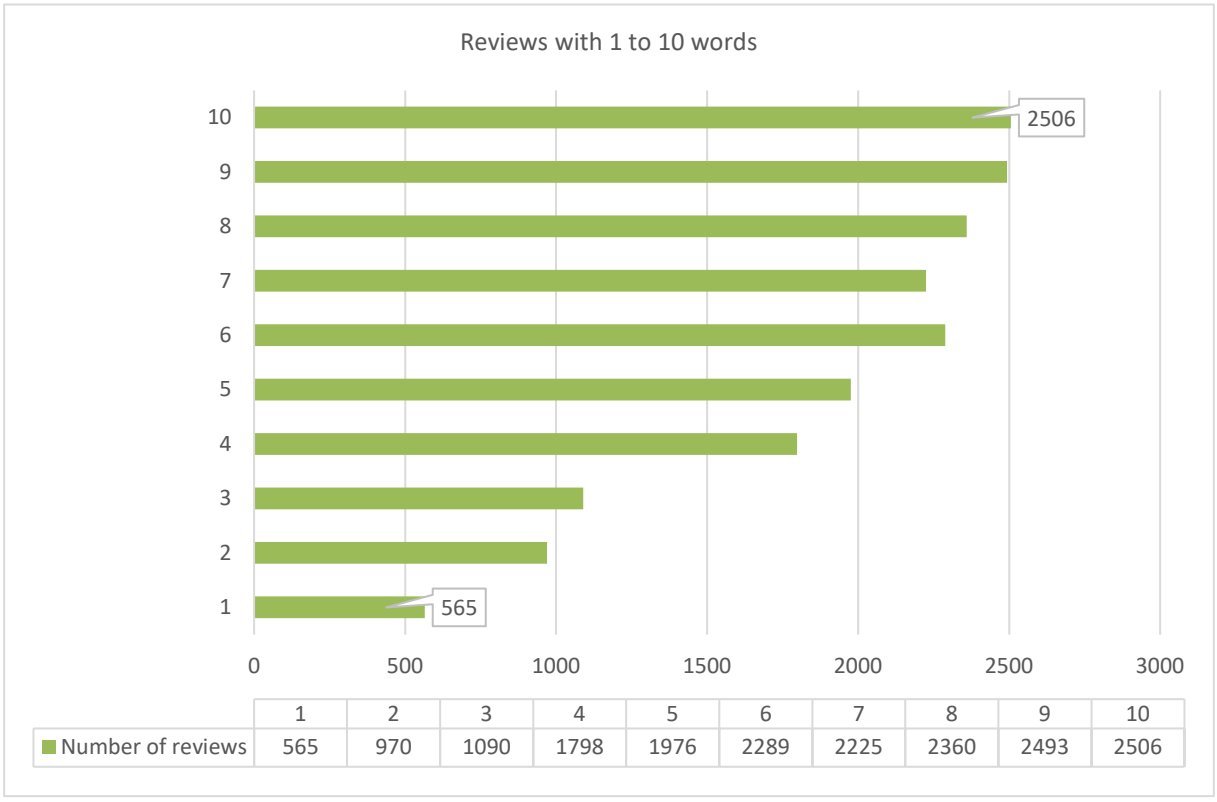
Only 395 (0.16 %) users have more than 1000 reviews, while 149,390 (59.05 %) of users have only 10 or less reviews. Then 81,002 (32.03 %) of users have written between 11 and 100 reviews and the rest of 22,110 (8.74 %) users have between 101 and 1000 reviews.

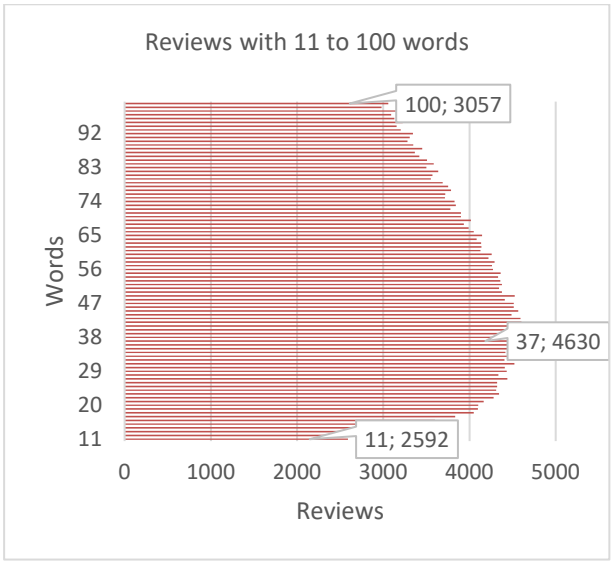
6.4.2 Charts and statistics for the number of words per comment

Here we also analyze the number of words per review. Average number of words per comment is 128.4. The median is 94.

Table 6-4: The analysis of the number of words per review

Variance	Average	The standard deviation	Median
13837.0505	128.5	117.6	94





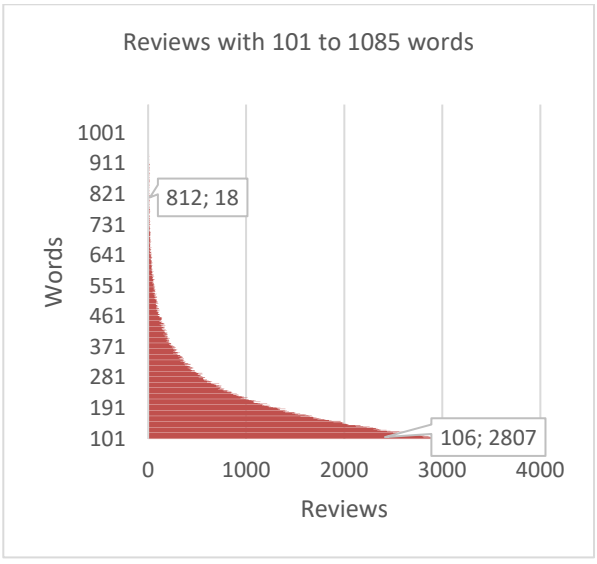


Figure 6-4: Graphical representation of number of words per review

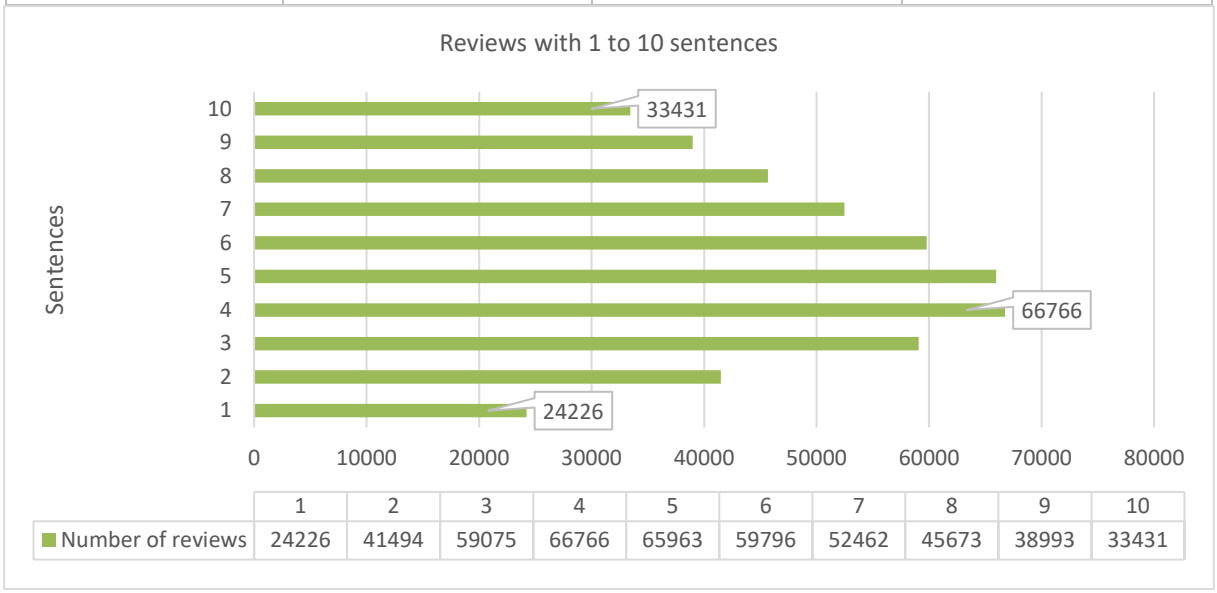
This study shows that 47.15% of reviews are longer than 101 words. Then 50.26 % of reviews contain between 11 and 100 words, while only 2.59% of reviews contain 10 or less words.

6.4.3 Charts and statistics for the number of sentences per comment

To answer the question “How many sentences are in a review?” the number of sentences in each review was counted. We use the NLTK data package which includes a pre-trained Punkt tokenizer for English. Average number of sentences per comment is 9.4. The median is 7.

Table 6-5: The analysis of the number of sentences per review

Variance	Average	The standard deviation	Median
61.0624091	9.4	7.8	7



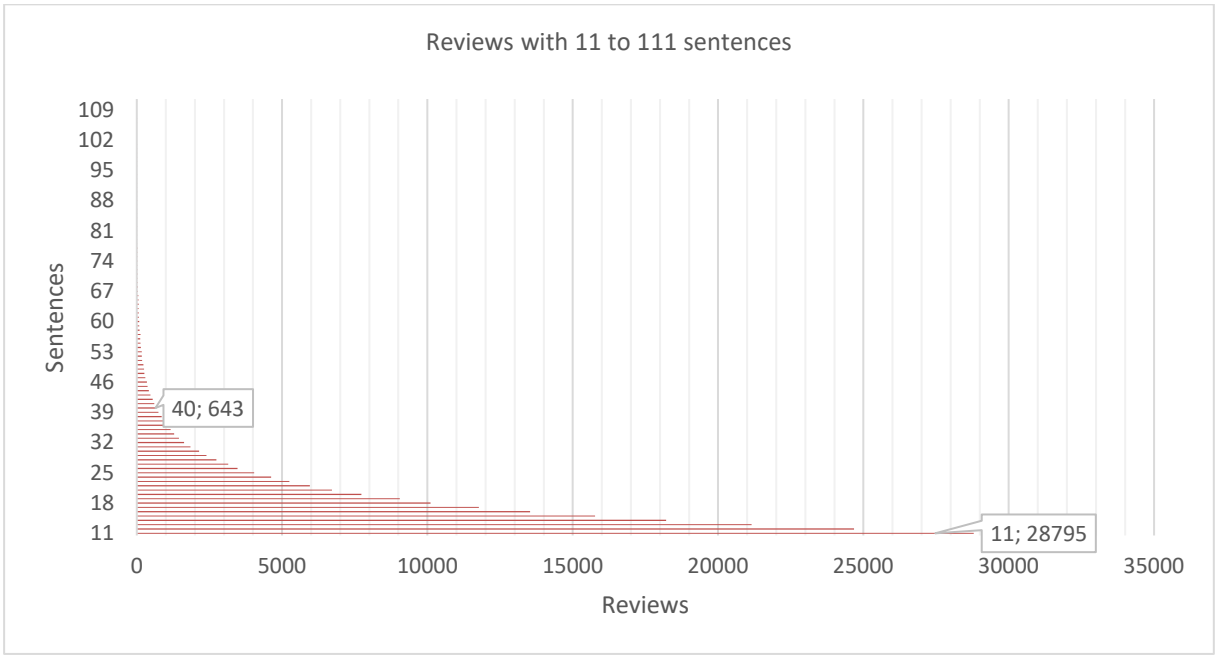


Figure 6-5: Graphical representation of number of sentences per review

This study shows that 31.42 % of reviews are longer than 10 sentences. However, 68.57 % of reviews contain only 10 or even less sentences.

6.5 Sentiment classification of Yelp reviews

To classify a review, the most important indicators of sentiments are opinion words. These are words that are commonly used to express positive or negative sentiments. For example, "*tasty, appetizing, scrumptious, yummy, luscious, delectable, and mouth-watering"* are positive sentiment words to describe "*delicious meals"*, and "*stale, nasty, rancid, inedible, unpalatable, tasteless, and bad"* are negative sentiment words as for example in "*meals tasting bad or lacking flavour"*.

6.5.1 Resources for analyzing the sentiments and opinions expressed in review texts

Due to high resources needed for sentiment polarity calculation of all the 1,127,525 reviews, we had to use SentiWorldNet and Hadoop on "Amazon Web Services". A high-level data processing language called Apache Pig Latin[24], is used for the sentiment polarity calculation of all reviews. We presented the complete Apache Pig Latin code that processes the sentiment polarity calculation in *Appendix D*.

6.5.2 Using SentiWordNet for Sentiment Analysis

To classify a review, every word which may represent an opinion must be detected. To calculate the polarity of words we use SentiWordNet v3.0.0 as a reference tool. The SentiWordNet is the result of the automatic annotation of all the synsets of WordNet according to the notions of "positivity", "negativity", and "objectivity".

The SentiWordNet is an enhanced lexical resource explicitly devised for supporting sentiment classification and opinion mining applications, which contains a list of thousands of English words/terms, and Part of Speech (PoS), which have been attributed a score of positivity, negativity, and objectivity ranging from 0 to 1. The sum of the positive, negative and objective scores is equal to one. This means that the positive and negative scores can have values between zero (the minimum value) and one (the maximum value).

[24] https://www.cloudera.com/tutorials/beginners-guide-to-apache-pig.html

However, there are very few numbers of words with the maximum polarity (positive or negative) value. Table 6-6 shows the compelete list of all these words.

Table 6-6: Example of words of SentiWordNet with maximum polarity

# POS	PosScore	NegScore	Obj.	SynsetTerms
a	**1**	0	0	unsurpassable#1
a	**1**	0	0	soft#18 mild#3 balmy#2
a	**1**	0	0	estimable#1
a	**1**	0	0	good#3
a	**1**	0	0	homological#1 homologic#1
a	**1**	0	0	good#6
a	**1**	0	0	respectable#2 honorable#4 good#4 estimable#2
a	**1**	0	0	sensational#1
a	**1**	0	0	mean#4
a	**1**	0	0	splendid#2 first-class#1 fantabulous#1 excellent#1
a	**1**	0	0	topping#1 top-hole#1 top-flight#1
n	**1**	0	0	wonderfulness#1 admirableness#1 admirability#1
n	**1**	0	0	praise#1 kudos#1 extolment#1 congratulations#1
n	**1**	0	0	first-rater#1
n	**1**	0	0	researcher#1 research_worker#1 investigator#1
n	**1**	0	0	happiness#1 felicity#2
n	**1**	0	0	walking_on_air#1 seventh_heaven#1 cloud_nine#1 blissfulness#1 bliss#1
v	**1**	0	0	like#2
v	**1**	0	0	love#2 enjoy#3
a	0	**1**	0	henpecked#1 dominated#2
a	0	**1**	0	unfortunate#3
a	0	**1**	0	abject#2
a	0	**1**	0	sorry#2 sad#3 pitiful#2 lamentable#1
a	0	**1**	0	unsound#5 unfit#3 bad#10
a	0	**1**	0	scrimy#1
a	0	**1**	0	tawdry#2 shoddy#1 cheapjack#1
n	0	**1**	0	shitwork#1 scut_work#1
n	0	**1**	0	abduction#1
n	0	**1**	0	disrespect#1 discourtesy#1
n	0	**1**	0	worst#1
n	0	**1**	0	hound#2 heel#3 dog#4 cad#1
n	0	**1**	0	motormouth#1
n	0	**1**	0	angriness#1 anger#2
v	0	**1**	0	mislead#1 misguide#1 misdirect#2 lead_astray#2

The SentiWordNet is structured by SynsetTerms. One word can belong to many synsets, and each synset has its own positive polarity score, negative polarity score and objectivity score. With the assumption that reviews are rather subjective text, we ignore the objectivity scores and only consider negativity and positivity of each word for finding the strongest sentiment words.

Not only, a word may have more than one PoS, but also SentiWordNet lists multiple senses of words for each of its PoS. This information gives us different possibilities to calculate the overall positive score and overall negative score of a review text.

First, we consider computing a sentence score by by summing-up scores of individual words appearing in it. However, since we want to compare several review texts, this score is not reliable, as a longer review may naturally have a higher number of positive words. Therefore, we need to inspect the averages. Having this in mind, to predict the opinion expressed in a review, we will investigate two different approaches. Both approaches are based on the polarity score values provided by SentiWordNet.

6.5.2.1 Approache #1: Considering PoS

PoS indicates the property of a word, thus it can be utilized to calculate sentiment scores. The polarity score is extracted according to the PoS the term belongs to and is matched to each of the terms in the review. The PoS considered in this study are adjective (a), noun (n), adverb (r) and verb (v). In this method, after PoS tagging, only those terms that are a, n, r or v are searched in SentiWordNet. In this way, terms to be considered are reduced and therefore all senses are not considered. This method is a better approach for small, and medium datasets analysis. We use this method in chapter 4 while only 500 reviews were chosen for the research purposes.

In this method, we can consider two different ways to calculate the score of a term. By considering:

A) The score of the sense #1 of a term at PoS level. Sense #1 represents most common sense. For example, if the word "good" occurs as noun in a sentence then only score of the noun sense of "good#1" is used.

Table 6-7: Example of using the score of the sense 1 of a term at PoS level

Noun
{"POSpeech":"n","PosScore":"0","NegScore":"0","good"}:" #4" {"POSpeech":"n","PosScore":"0.875","NegScore":"0","good"}:" #2" {"POSpeech":"n","PosScore":"0.5","NegScore":"0","good"}:" #1" {"POSpeech":"n","PosScore":"0.625","NegScore":"0","good"}:"#3"
PosScore = 0.5 NegScore = 0

In this case we split SentiWordNet to four files and keep only the sense #1 of each term. Each file regroups only one of the four (a, n, r, and v) PoS terms.

B) The average score at PoS level. For example, if the word "delicious" occurs as adjective in a sentence then average score of all adjective senses of "delicious" is used.

Table 6-8: Example of using the average score of all term occurrences at PoS level

Adjective
{"POSpeech":"a","PosScore":"0.75","NegScore":"0.25","delicious"} {"POSpeech":"a","PosScore":"0.75","NegScore":"0","delicious"}
PosScore = Average (0.75; 0.75) = **075** **NegScore** = Average (0.25; 0) = **0.125**

6.5.2.2 Approache #2: Considering all senses of a term without taking PoS into account

The entire 1.2 million reviews in the Yelp dataset are used for our research and opinion mining. With a large volume of data, when PoS is considered, the system gets very complex quickly and it is hard to maintain. To make optimal use of available computational resources, reduction of the processing time is achieved by using the average score of all synsets “SynsetTerms”, that a word belongs to.

The advantage of this strategy is that, despite of having more than one million reviews analysed, we still overcome to assign a positive score, and a negative score to each review. Further the same process is repeated to calculate the positive and the negative polarity scores of the first sentence of each review.

Thus, the polarity score of each term is the avrage of PosScores and the avrage of NegScores of all its occurrences identifed in the SentiWordNet. As we mentioned, in this approche PoS is not considered. For example, if the word "large" occurs as adjective in a sentence, then its PoS is ignored and instead the average scores (positive and negative) of all senses of " large" across different PoS is used.

Table 6-9: Example of using the average scores of all term occurrences in SentiWordNet

Adjective
{"POSpeech":"a","PosScore":"0.5","NegScore":"0","large"}
{"POSpeech":"a","PosScore":"0","NegScore":"0.25","large"}
{"POSpeech":"a","PosScore":"0.25","NegScore":"0","large"}
{"POSpeech":"a","PosScore":"0.5","NegScore":"0","large"}
{"POSpeech":"a","PosScore":"0.25","NegScore":"0.125","large"}
{"POSpeech":"a","PosScore":"0.125","NegScore":"0","large"}
{"POSpeech":"a","PosScore":"0","NegScore":"0","large"}
Adverb
{"POSpeech":"r","PosScore":"0","NegScore":"0.25","large"}
{"POSpeech":"r","PosScore":"0","NegScore":"0","large"}
{"POSpeech":"r","PosScore":"0","NegScore":"0","large"}
Noun
{"POSpeech":"n","PosScore":"0","NegScore":"0","large"}
PosScore = Average (0.5; 0; 0.25; 0.5; 0.25; 0.125; 0; 0; 0; 0; 0) = **0.147** **NegScore =** Average (0; 0.25; 0; 0; 0.125; 0; 0; 0.25; 0; 0; 0) = **0.056**

In this case we do not split SentiWordNet and keep it as it is, with all senses of each term.

Table 6-10: Few lines of SentiWordNet v3.0.0

# POS	ID	PosScore	NegScore	SynsetTerms
a	1740	0.125	0	able#1 (#1 = the first sense of "able")
a	2098	0	0.75	unable#1
a	2312	0	0	dorsal#2 abaxial#1

a	2527	0	0	ventral#2 adaxial#1
…				
n	1024968	0.25	0	standing_operating_procedure#1 standard_procedure#1 standard_operating_procedure#1 sop#3
n	1025254	0.375	0.125	lockstep#1
n	1025411	0	0	stiffening#1
n	1025563	0	0.125	red_tape#1 bureaucratic_procedure#1
…				
r	296836	0.25	0	mysteriously#1 enigmatically#1 cryptically#1
r	297023	0.25	0	cryptographically#1
r	297112	0.375	0	cutely#1 cunningly#1
r	297319	0.125	0	shortly#3 short#7 curtly#1
…				
v	2769900	0	0.375	storm#4
v	2770019	0	0.125	squall#3
v	2770170	0	0.125	storm#3
v	2770362	0	0.125	bluster#1

6.5.3 Implementation of approache #2 with *TF-IDF* as weighting factor

Our key objective is to calculate an overall contextual polarity for each review.

To calculate the normalized positive polarity of a comment *D* according to the terms *i = 1, 2, ... m*, we apply the formula (6.1) in which *TFi-IDF* indicates the term frequency-inverse document frequency of the term *i* in the review text and *wt +i* the average value of the positive polarity of this term in SentiWordNet. Similarly, the normalized negative polarity is measured by the equation (6.2) in which *wt-i* indicates the average value of the negative polarity of this term in SentiWordNet.

In other words, first we identify all occurrences of word *i* and their *PosScore* and *NegScore* in SentiWordNet, and then we calculate the avrage of *PosScores* of all senses of word *i* as *wt +i* and the avrage of *NegScores* of all senses of word *i* as *wt-i*.

$$Normalized\_PosScore(D) = \sum_{i=1}^{m} nTF_i\_IDF \cdot wt +_i \qquad (6.1)$$

$$Normalized\_NegScore(D) = \sum_{i=1}^{m} nTF_i\_IDF \cdot wt -_i \qquad (6.2)$$

$$wt +_i = \frac{1}{n}\sum_{i=1}^{n} PosScore(i) \quad (6.3)$$

$$wt -_i = \frac{1}{n}\sum_{i=1}^{n} NegScore(i) \quad (6.4)$$

To normalize the polarity scores for each review, we divide them by the number of words in that review. In other words, we use nTF_i instead of TF_i. Therefore, the equations (6.5) and (6.6) can be used to calculate the polarity of a comment *D* without normalization.

$$PosScore(D) = \sum_{i=1}^{m} TF_i\_IDF \cdot wt +_i \quad (6.5)$$

$$NegScore(D) = \sum_{i=1}^{m} TF_i\_IDF \cdot wt -_i \quad (6.6)$$

With

- *nTFi-IDF*: Normalized *TF(i,d)×IDF(i,D)*,
- *TFi-IDF*: Unnormalized *TF(i,d)×IDF(i,D)*,
- *IDF(iD)*: Inverse document frequency
- $wt +_i$*:* The average positive sentiment score of the word *i*,
- $wt -_i$*:* The average negative sentiment score of the word *i*,
- *n:* Total number of all synsets in SentiWordNet, that word *i* belongs to.

6.5.3.1 Term Frequency-inverse Document Frequency

TF-IDF is a short term for the Term Frequency-Inverse Document Frequency formula that aims to define the importance of a word within a review.

Term Frequency (*TF*), as implied by the name, indicates the frequency of a specific word within a review. As a variant, *nTF* is calculated by dividing the number of times the word occurs in a review by the total number of words within the same review.

nTF(i,d) is the term frequency of a word i in a review d

$$nTF(i,d) = \frac{TF(i,d)}{\sum_k TF(k,d)} \quad (6.7)$$

Table 6-11 depicts the TF and nTF of some of the words used in a review which is formed by 209 words.

Table 6-11: Term frequency of few words from a review

Review ID	Word count	word	TF	nTF
---fHfISisMXB-fv7Nl_SA	209	i	6	0.0287
---fHfISisMXB-fv7Nl_SA	209	at	1	0.0047
---fHfISisMXB-fv7Nl_SA	209	by	1	0.0047
---fHfISisMXB-fv7Nl_SA	209	if	1	0.0047
---fHfISisMXB-fv7Nl_SA	209	of	3	0.0143
---fHfISisMXB-fv7Nl_SA	209	or	1	0.0047
---fHfISisMXB-fv7Nl_SA	209	don	2	0.0095
---fHfISisMXB-fv7Nl_SA	209	for	4	0.0191
---fHfISisMXB-fv7Nl_SA	209	expensive	1	0.0047

As it is depicted in Table 6-12 our Pig Latin code *(Appendix D)* that processes the sentiment polarity calculation output/results four polarity scores for each review. Two normalized scores (Normalized_PosScore and Normalized_NegScore) and two scores (PosScore and NegScore) without normalization.

Table 6-12: Two examples from 1,127,525 lines in the table

Calculated review's polarity table		
review_id	GeLxz2wbCTaNAgGebNBz6A	9AxAw1QRBvtq0YAvCbgunA
sentence_count	7	4
word_count	173	43
user_id	__26EDQ1FacBdY4gChHsuA	__26EDQ1FacBdY4gChHsuA
date	24.01.10	29.01.10
stars	5	5
posscore	15.435	11.787
normalized_posscore	0.089	0.274
negscore	13.742	7.878
normalized_negscore	0.079	0.183
business_id	UPgqmSaO5zOWNKczVwakLA	33aOj9haiHsULCUmFOZ7uQ

Inverse document frequency (*IDF*) shows how frequently a word is used within a collection of reviews. This is a corrective metric that aims to level down the importance of common words and various function words and to give more prominence to meaningful words.

The *IDF* is the logarithmically scaled fraction of the reviews that contain the word. It is calculated by the *ln* (natural logarithm) of "the number of reviews divided by the number of reviews that contain the word *i*".

$$IDF(i, D) = ln \frac{N}{|\{d \in : i \in d\}|} \quad (6.8)$$

With

- *N*: Total number of reviews in the corpus $N = |D|$,
- $|\{d \in : i \in d\}|$*:* Number of reviews where the word *i* appears (i.e., $TF(i, d) \neq 0$). If the term is not in the corpus, this will lead to a division-by-zero. It is therefore common to adjust the denominator to $1 + |\{d \in : i \in d\}|$.

The product of these two metrics is the *TF-IDF* formula that indicates the relevance of a word to the review. The larger is the *TF-IDF* value - the more relevant (important) is the word to the review. In short, we are somehow counting the occurrence of each word in a review and weight the importance of those words and calculate a polarity score for that review.

TF-IDF is calculated as:

$$TF\_IDF(i, d, D) = TF(i, d) \times IDF(i, D) \quad (6.9)$$

6.5.3.2 Example of TF-IDF calculation

The review corpus has 1,127,525 reviews in total. The word "reaaaaallllyyyyy" occurs only in two reviews. Therefore, the *IDF* for this word is calculated as following:

$$iDF(\text{reaaaaallllyyyyyy}, D) = ln \frac{1{,}127{,}525}{2} = 13.2405$$

The first review is composed of 49 words and it is the following:

"Daaaamnnnnnn.. it's french japanese fusion with really really really reaaaaallllyyyyyy amazing food & drinks. It's inside the MGM hotel run by japanese sushi chiefs. We had our bachlorette dinner here before we went out for the night. bill came out to about $1200 but.. SUPER worth it. hahahha. damnnn.."

The second review is composed of 43 words.

"Went there for lunch. The wait was not that bad. It only took us ten minutes to get a seat outside. The atmosphere was great especially when the bellagio fountain is right across the street. Food was decent, but it was reaaaaallllyyyyyy slow."

So, the *nTF* of the word "reaaaaallllyyyyy" for each review is calculated as following:

$$nTF(\text{reaaaaallllyyyyyy}, " - HIt - RBCa - Rk2oHSKwHP7g") = \frac{1}{49} = 0.0204$$

$$nTF(\text{reaaaaallllyyyyyy}, "psbYscC8TxIvDXssPSu2sA") = \frac{1}{43} = 0.0232$$

The same way we can calculate the weighting factor *TF-IDF* for the word "reaaaaallllyyyyy" in each of these two reviews.

$$TF\_IDF(\text{reaaaaallllyyyyyy}, D) = 13.2405 * 0.0204 = 0.2702$$

$$TF\_IDF(\text{reaaaaallllyyyyyy}, D) = 13.2405 * 0.0232 = 0.3079$$

As we can observe that the weighting factor *TF-IDF* for the same word is different for these two reviews.

At the beginning of this chapter we asked few questions. We want to find answers to the following two questions.

- Can we automatically recognize the authors? Is there specific "writing patterns"?
- Is there a way to automatically filter fake (malicious) comments and identify their authors, even if the same person uses many different identifiers?

We found a very high *IDF* value for the word "reaaaaallllyyyyy", because it occurs only in two reviews. Knowing that, we investigate to verify whether these two reviews are written by the same user (user identification). To do so, we retrieve the user identification of each reviewer as well as the identification of businesses that these two reviews were written for.

Although the word "reaaaaallllyyyyy" is very rare and it is used only in two reviews, the result of investigation is not concluding. Neither the identifications of writers nor the identifications of restaurants for which these two reviews were written are matching.

At this point, we concluded that due to the restaurant specific vocabulary used by most of the Yelp users and the fact that reviews are rather short documents (the average number of words per comment is 128.4), it will be difficult to answer the above questions with certainty. Further investigations are needed to confirm this preliminary observation. For instance, in the above example we (human verification required) can easily observe that the text of these two reviews have quite different writing styles and conclude that they are not written by the same author.

6.5.4 Is the sentiment that is expressed in the first sentence of the comment dominant?

One of the questions to which we would like to find an answer is to find out whether the polarity of the first sentence of a review is representing the polarity of the whole comment? To the best of my knowledge, this question has not yet been investigated considering the entire corpus of reviews from the Yelp dataset.

This is an interesting question to be answered, because if turns out that in the vast majority of cases the polarity of the first phrase of a review represents the polarity of whole review's text, then we can (as an option in a web application) speedup the opinion mining process time, by analyzing only the first sentence of the review instead of runing the process for all its sentences.

To investigate this question, first we split all reviews to sentences, and separate the first sentence of each review. All 1,127,525 reviews (the entier dataset) is used for this investigation. Then, we create a new database table in which the first sentence is stored.

Table 6-13 depicts few lines of the database created for this purpose.

Table 6-13: Few lines of the table with the first sentence of each review

First sentence of the comment
(Big greasy burgers!)
(What a great local joint on the southwest.)
(Ate lunch at Beach Cafe yesterday.)
(I really can not believe all the good reviews of this place.)
(I'm going to keep it short and sweet for once.)
(This is one of my favorite breakfast spots.)
(Today my office treated us to a lunch catered by Beach Cafe.)

(Has a California beachfront feel to the restaurant.)
(I can't remember how many times my friends and I have been here after cycling or working out.)
(I've gone here twice, both times for breakfast.)
(Someone brought a menu for Beach Cafe into my work a while back, and I was in the mood for something new, so I decided to try them out.)
(A neighborhood cafe with an extensive menu and cute atmosphere - love it!)
(This place is cute, swimsuits hang above in surf style decor.)
(Back in the summer of 2010, my father-in-law suggested this place.)
(Been living around this place for several years and never noticed it until I saw a small sign on the side of the road that caught my eye.)
(Best place in Vegas!)
(Not a fan.)

Then, to predict each sentence polarity, we use the same method of sentiment analysis, that we used for opinion mining of reviews. As a result of text processing, an overall contextual polarity is calculated also for the first sentence of each review.

Much to our surprise, the polarity values of first phrases are calculated only for 1,109,227 reviews. It is difficult to determine exactly why polarity values of first sentence of 18,298 reviews are missing. One of the possible causes of this may be that the review text begins with an empty phrase. For example, an end point "." at the beginning of a review may have been considered by the system as its first sentence. A phrase which contains no words. Anyways, regardless of the exact reasons of this phenomenon, we decide to ignore these missing 18,298 reviews (1.6%) and continue our investigation with the rest of results (98.4%).

It is imperative to note that we will only compare the polarity (positiveness or negativeness) of a review versus its first sentence's polarity. In Fact, the normalization does not affect overall polarity of a sentence in sense of positiveness or negativeness. In other words, a sentence which is positive does not become negative after the normalization of its PosSore and NegScore. That is why we do not need to use the normalized values for this part of analyse.

To calculate the overall polarity of a comment (or of its first sentence), we subtract its negative polarity score value from its positive polarity score.

$$SenGeneral(D) = PosScore(D) - NegScore(D) \qquad (6.10)$$

Accordingly, the following three different outcomes are possible:

$SenGeneral(D) > 0$ the overall polarity is positive

$SenGeneral(D) = 0$ the overall polarity is neutral

$SenGeneral(D) < 0$ the overall polarity is negative

As mentioned at this step the polarity "value" itself is not important. What is important to us, is to know whether a review is positive, neutral, or negative and then to verify if its first sentence has the same polarity. So, to answer this question, by using the review identifications, we compare the overall contextual polarity of each review *SenGeneral(D)* with the one of its first sentence *SenGeneral(D_{fs})*.

In total, the overall polarity of 1,109,227 reviews' first sentence were compared to the polarity of their respective review. As shown in Table 6-15, in 771,997 cases these two polarities match. In other words, as Figure 6-6 depicts, for 70% of reviews the sentiment that is expressed in their first sentence is dominant.

Table 6-14: Comparison of polarity of a review with the polarity of its first sentence

Y (polarity is the same)	**N (polarity is not the same)**
771,997	337,230

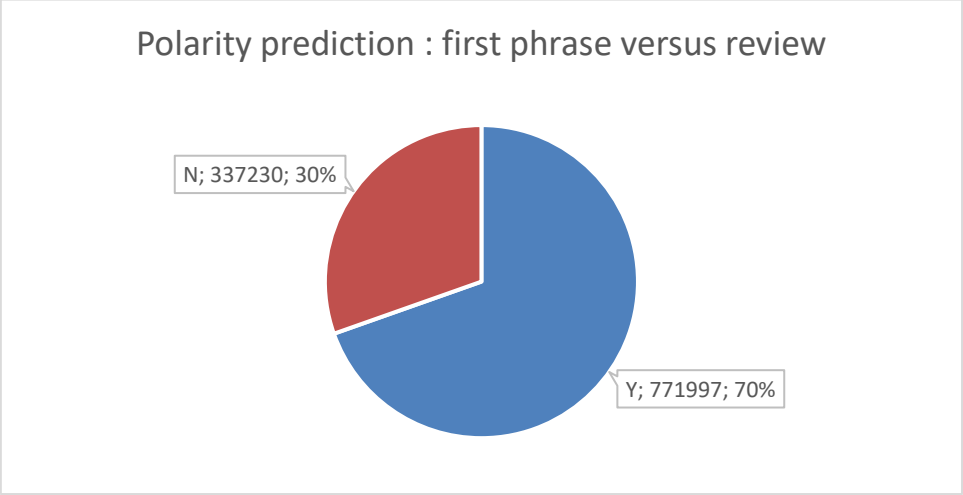


Figure 6-6: Polarity of reviews versus the polarity of their first sentence

Even though in 70% of cases the first sentence polarity is the same as the polarity of review itself, but from my perspective this percentage is not big enough to ignore the computing errors which may cause by considering only the first sentence of review for sentiment analysis.

6.5.5 Example analysis of a review

Below is just an example text that was analyzed in our study. The text length is 70 words. After processing the review data, we have for each review, its original text, the same text after removal of stop-words, the PoS tagging of text, and finally the review text broken to distinct sentences.

Stanford Log-linear PoS Tagger (java) was used. This is a piece of software that reads text and assigns PoS to each word (and other token), such as noun, verb, adjective, or adverb. A popular procedure to reduce the noise of textual data is to remove stop-words by using pre-compiled stopword lists. We use a list of English stop-words that contains 571 words.

Splitting text into sentences might look like a simple task but it's not. For example, a period (full stop) is often used to signify an abbreviation. Thus, when a period occurs in a place that is obviously not the end of a sentence, or if a sentence ends with an abbreviation followed by a period, can cause the fail of sentence splitter code.

The solution is to match and capture the abbreviations and replace them momentary by their meaning, before splitting the text into sentences. Bellow are some of the abbreviations we replaced before splitting.

```
abbreviations = {'dr.': 'doctor', 'Dr.': 'doctor', 'mr.': 'mister', 'Mr.': 'mister', 'bro.': 'brother', 'bro': 'brother', 'mrs.': 'mistress', 'Mrs.': 'mistress', 'ms.': 'miss', 'Ms.': 'miss', 'jr.': 'junior', 'sr.': 'senior', 'i.e.': 'for example', 'Sr.': 'senior', 'e.g.': 'for example', 'vs.': 'versus', 'A.M.': 'ante meridiem', 'P.M.': 'post meridiem'}
```

We use an NLTK implementation (Loper et al., 2009) of the Punkt algorithm (Kiss and Strunk, 2006) for English that achieves a very high accuracy on a broad range of texts. This tokenizer divides a text into a list of sentences by using an unsupervised algorithm to build a model for abbreviation words, collocations, and words that start sentences.

6.5.5.1 The original text of review

“Worth a stop for quick refreshment & meal. Solid restaurant with easy access from the 60 vic Gold Canyon. Entrees sit around $11 +\- $2 with complete selection of Mexican beer to complement. What's notable - shrimp (taco, burrito or fried). Fajita's also score high. Flavorful salsa What's not - flan. If your expecting a small delicate dome shaped custard with carmel sauce you be disappointed... Served like pie with crust and whipped cream.”

6.5.5.2 The same text after removal of stop-words

“Worth stop quick refreshment & meal. Solid restaurant easy access 60 vic Gold Canyon. Entrees sit $11 +\- $2 complete selection Mexican beer complement. What's notable - shrimp (taco, burrito fried). Fajita's score high. Flavorful salsa What's - flan. If expecting small delicate dome shaped custard carmel sauce disappointed... Served like pie crust whipped cream.”

6.5.5.3 The Parts of Speech

((Worth,NNP),(stop,NN),(quick,JJ),(refreshment,NN),(meal,NN),(restaurant,NN),(easy,JJ),(access,NN),(vic,NN),(Gold,NN),(Canyon,NNP),(sit,VB),(complete,JJ),(selection,NN),(Mexican,JJ),(beer,NN),(complement,VB),('s,VB),(notable,JJ),(shrimp,NN),(taco,NN),(burrito,NN),(fried,VB),(also,RB),(score,VB),(high,JJ),(salsa,NN),('s,VB),(not,RB),(flan,NN),(expecting,VB),(small,JJ),(delicate,JJ),(dome,NN),(shaped,VB),(custard,NN),(carmel,NN),(sauce,NN),(be,VB),(disappointed,VB),(Served,VB),(pie,NN),(crust,NN),(whipped,VB),(cream,NN))

6.5.5.4 The review text broken to distinct sentences

{(Worth a stop for quick refreshment & meal.),(Solid restaurant with easy access from the 60 vic Gold Canyon.),(Entrees sit around $11 +\- $2 with complete selection of Mexican beer to complement.),(What's notable - shrimp (taco, burrito or fried).),(Fajita's also score high.),(Flavorful salsa What's not - flan.),(If your expecting a small delicate dome shaped custard with carmel sauce you be disappointed...),(Served like pie with crust and whipped cream.)}

6.5.5.5 Predicted polarity values

After using equations (6.1), (6.2), (6.5) and (6.6) we obtain four polarity score values for each of 1,127,525 reviews. Table 6-15 shows the predicted polarity score values for the example review given above.

Table 6-15: Polarity score values for this review

PosScore	NegScore	Normalized_PosScore	Normalized_NegScore
12.495	9.371	0.178	0.133

6.6 In general, do consumers use positive or negative words to write their comment?

It is important to emphasise that, lexicon level polarity assignment is not easy, especially because the polarities of individual words are highly dependent on their context. Therefore, how to assign polarity to each word in the lexicon of positive and negative terms?

So instead of counting the number of positive words and negative words used by Yelp users, we consider whole review corpus as a single document. Then, as each word has its PosScore and NegScore, we consider that the sum of all PosScores versus the sum of all NegSocres are good indictores to find out whether Yelp users use words with more positivty or negativty while writing their reviews.

As the output of opinion mining process of Yelp review corpus, we calculated four polarity score values for each of 1.2 million comments. Two of them are values of positive and negative scores, and the other two are the normalized values of these scores. Likewise, four polarity score values are calculated for each of reviews' first sentence.

Table 6-16: Sum of polarity score values for all reviews

Sum of PosScores	Sum of NegScores	Sum of normalized PosScores	Sum of normalized NegScores
19,196,738	14,937,883	160,904.8	114,534.7

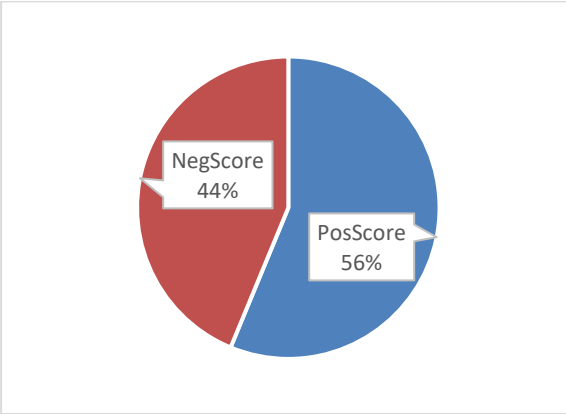


Figure 6-7: Sum of PosScores versus Sum of NegScores

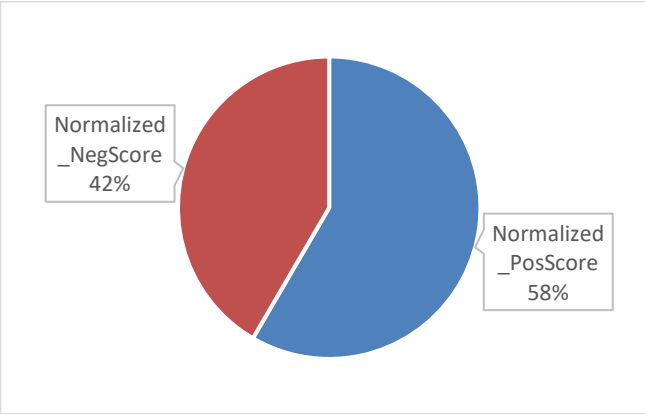


Figure 6-8: Sum of normalized PosScores versus Sum of normalized NegScores

The Figures 6-7 and 6-8 demonstrate that the "Sum of PosScores" of words used by Yelp consumers are higher than their "Sum of NegScores". We could say that Yelp users use rather words with higher positivity score to write their comment. We repeat the same operation for the polarity score values obtained from the sentiment analysis of reviews' first phrase.

Table 6-17: Sum of polarity score values for all reviews' first phrase

Sum of PosScores	Sum of NegScores	Sum of normalized PosScores	Sum of normalized NegScores
3,170,273	2,209,288	332,638.1	206,869.6

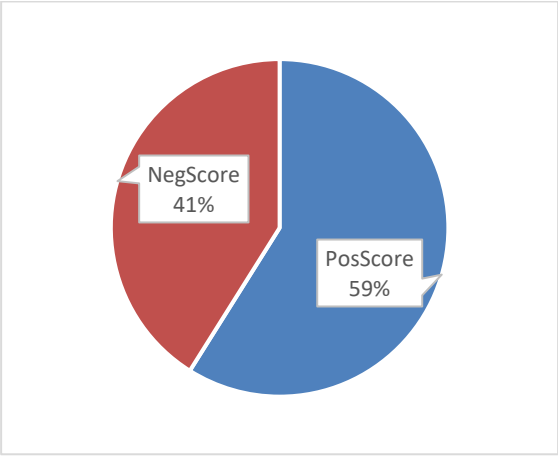


Figure 6-9: Sum of PosScores versus Sum of NegScores

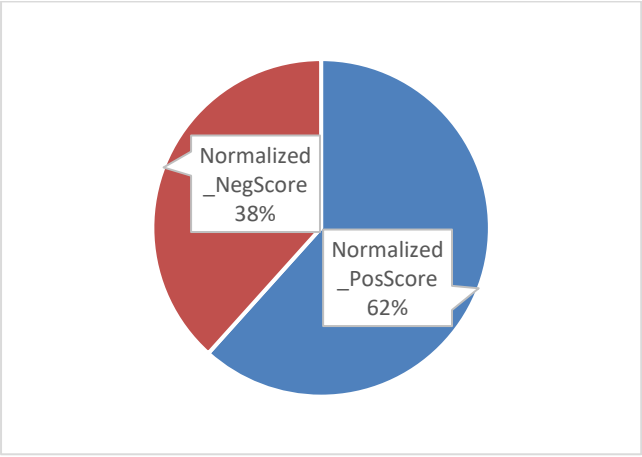


Figure 6-10: Sum of normalized PosScores versus Sum of normalized NegScores

The results above confirm that Yelp customers use even more (~4%) words with a higher positivity to write the first sentence of their review than for redaction of the comment itself.

6.7 Star ratings versus sentiment analysis of restaurant reviews

In this section, we only consider all restaurant reviews. Thus, a total of 706,404 restaurant reviews are analyzed. Due to the large sample size, most of the conclusions in this section are statistically very significant.

We want to compare sentiment analysis results of a restaurant review with its respective star ratings. The main objective of this section is to determine whether the stars assigned by the user to a restaurant faithfully reflect his sentiment which is expressed in his review?

It is important to understand that, the polarity prediction of a review is independent from the number of stars submitted by its author. As explained before, the polarity prediction score is calculated by the opinion mining of the review's text itself. However, Yelp star ratings data reflects the user's judgment which is expressed numerically by himself on a rating scale from 1 to 5.

First, according to their star ratings (1 to 5), we separate all restaurant reviews to five distinct groups. Table 6-19 depicts the distribution of the number of reviews in each group.

Table 6-18: Statistics for star ratings from 1 to 5

Star retings	**1**	**2**	**3**	**4**	**5**	**Total**
Number of reviews concerned	60,417	70,261	111,189	231,301	233,236	706,404
%	8.60%	9.90%	15.70%	32.70%	33.00%	100%

While 464,537 (65 %) of users have attributed 4 or 5 stars to restaurants, only 130,678 (18.5%) of users have attributed 1 or 2 stars to restaurants. Our first observation is that the most reviewers are happy customers who share their positive experiences online. It is interesting to see whether the predicted polarities of reviews will confirm this observation.

As explained earlier in this chapter, we use equeation (6.10) to calculate the *SenGeneral(D)* which is the overall polarity of comment *D*.

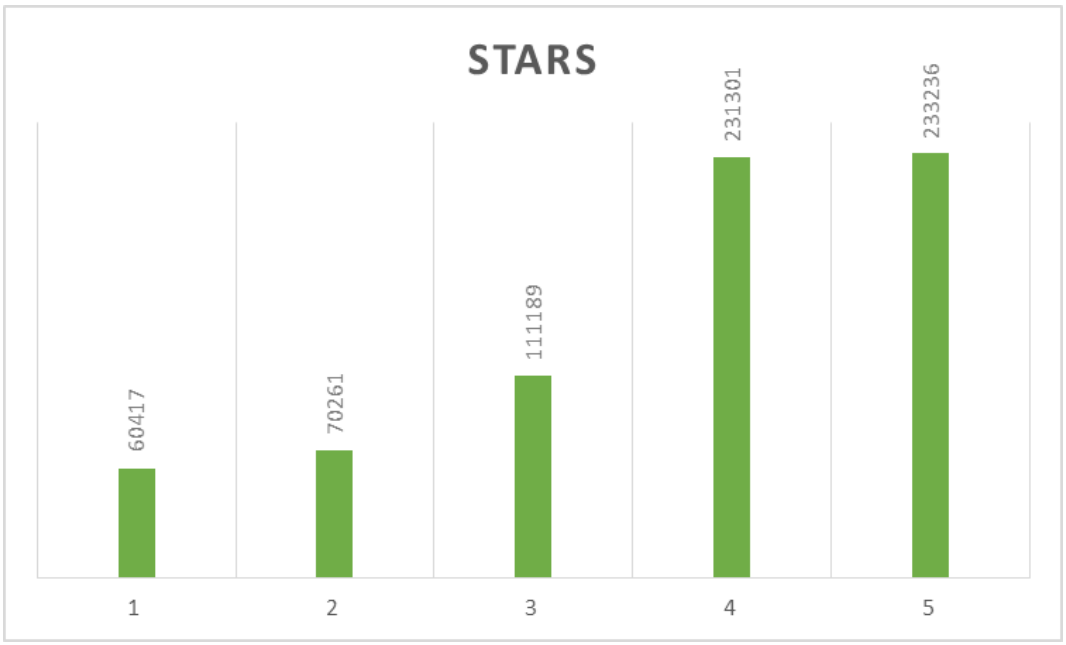


Figure 6-11: Statistics for stars for 1 to 5 stars of of 706,404 restaurant reviews

We notice that these predicted polarity values are floating-point numbers which vary between -110 and 120, however the star ratings are integer numbers between 1 and 5. Thus, we have two different kinds of numerical data, one integer and one floating-point. That is why we need to find a way to compare them.

$$-110 <= SenGeneral(D) <= 120 \quad (6.11)$$

Before searching the most effective method for comparing these two sets of values, we also need to take in consideration the following:

1. Each time a user writes a new review he also gives a new star rating to the business. So, the number of stars given to the same business by the same user can be different at each time.
2. Yelp in his dataset provides us only with about a tenth of all reviews. That is why we may not find always all the reviews written by the a given user but just a part of them. This makes it very difficult (in some cases impossible) to verify and/or compare the number of stars with the polarity of reviews written by the same user for the same business in a periodic manner.

At this stage, we want to predict the star rating of a review based only on its texts overall polarity *SenGeneral(D).* Predicting star ratings is important for better understanding of how subjective restaurant ratings are.

By attributing a value from 1 to 5 to each review's overall polarity value, in fact we are predicting its star rating given by the user. Obviously, we try to make our star rating prediction $\hat{R}u, i$ for user u and review i to be as close as Ru, i, the star rating user gives. In statistics, the Mean Absolute Error (MAE) is a measure of difference between two variables —that is, the absolute difference between the predicted values and the actual value.

$$MAE = \frac{1}{|N|} \sum_{i=1}^{N} \left| Ru, i - \hat{R}u, i \right| \quad (6.12)$$

$$Accuracy\ rate = \frac{correctly\ predicted\ class}{total\ testing\ class} \quad (6.13)$$

We propose two different methods for predicting the star ratings, and the *MAE* is employed to compare the performance of each approach.

6.7.1 Linear approach

Our first approach is to divide the range of *SenGeneral(D)* values in a linear manner into 5 equal intervals and assign our own stars, from 1 to 5, to each data range.

For stars (1 to 5) we cut all the values between -110 and 120 to five equal ranges of each 46.

SenGeneral(D): {-110; -64; -18; 28; 74; 120}

Then for each review, according to the range its *SenGeneral(D)* belongs to, we assign a star rating from 1 to 5.

Figure 6-11 depicts the results for star rating distributions for all of 705,404 comments. As we can clearly notice in the chart, most comments (98.52% or 695,959) are classified under 3 stars. This linear approach does not result a good distribution of stars. We calculate the *MAE =1.248*, the *Accuracy rate =16.06%*, and observe that only for 113,459 reviews, the predicted star ratings are correct.

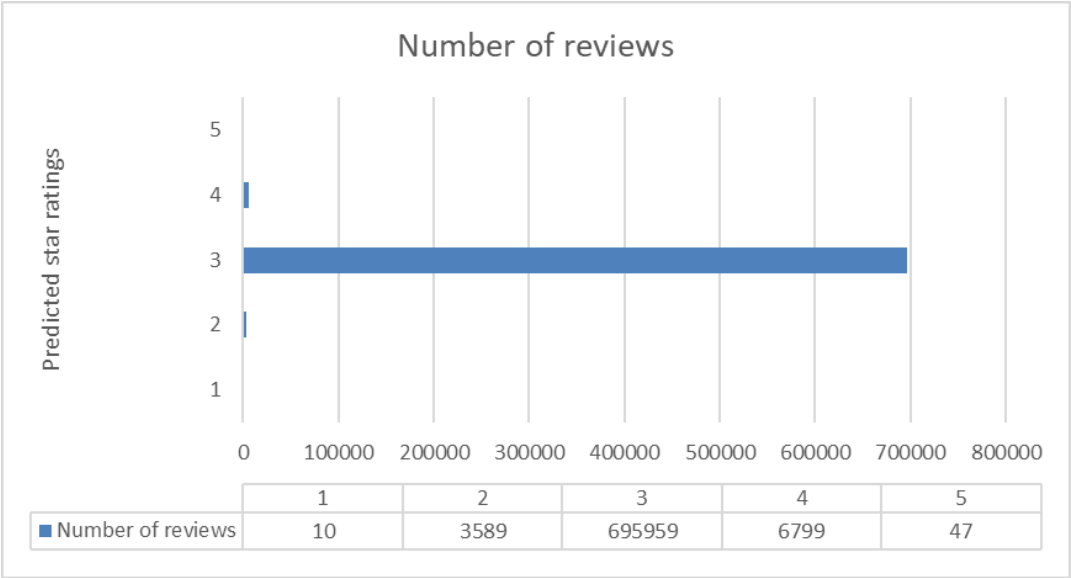


Figure 6-12: Distribution of stars to SenGeneral(D) values by a linear approch

6.7.2 Nonlinear approach

Our second approach is to separate these values in a nonlinear manner. As our first trial and error, we use the statistics for 1 to 5 stars (Figure 6-13) and find the intervals which result practically the same statistics.

SenGeneral(D): {-110; -4.22; -1.06; 1.17; 5.41; 120}

Here are the new distribution results and the chart for all comments.

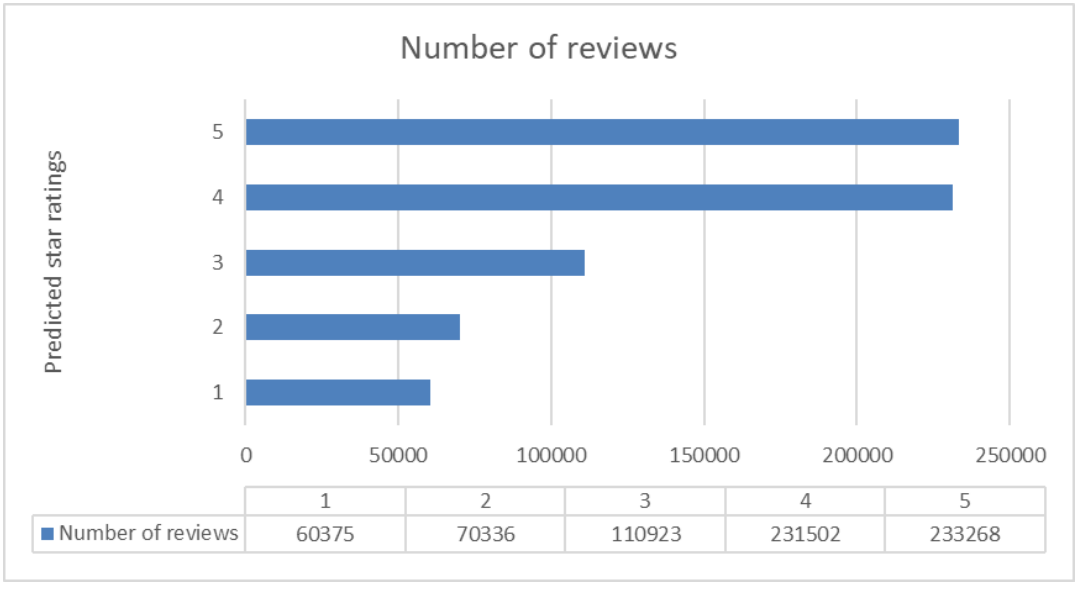


Figure 6-13: Smallest MAE, with much the same distribution of stars to actual star ratings by a nonlinear approch

We calculate the *MAE =0.999*, the *Accuracy rate =32.99%*, and observe that for 233,057 reviews, the predicted star ratings are correct. This sounds still like a low percentage but is pretty good for such a simple model using so huge and unstructured textual data.

By repeating cycle of operations (iterative process), we try to predict better than before and to come closer to the actual star rating. The best accuracy rate is achived by the following intervals.

SenGeneral(D): {-110; -4; -3.9; -3.89; 0.5; 120}

We observe that for (36.32% or 256,369) reviews, the predicted star ratings are correct. However, as we can see from Figure 6-14, the star ratings distribution is not identical for all points, and very few reviews are classified under the 2 or 3 stars (the classification system error is big). In addition, the calculated *MAE =1.06* is bigger than before.

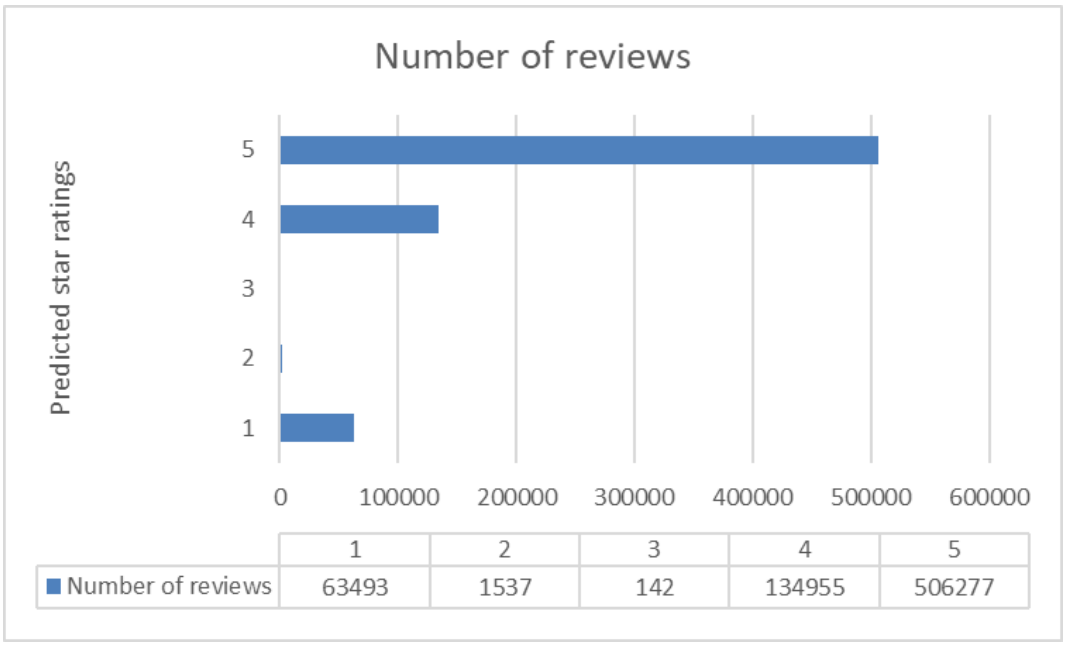


Figure 6-14: Best distribution of stars to SenGeneral(D) values by a nonlinear approach

As the smallest *MAE* (0.999) is achieved by *SenGeneral(D): {-110; -4.22; -1.06; 1.17; 5.41; 120}* intervals, we consider these results as acceptable.

6.7.3 Confusion matrix

Since the prediction of star ratings based on review texts with a zero-error tolerance gave mediocre results (accuracy less than 33%), we are interested to investigate whether with an error tolerance of 1 star we obtain a better accuracy.

In the field of machine learning and specifically the problem of statistical classification, a confusion matrix, also known as an error matrix, is a specific table layout that allows visualization of the performance of an algorithm. Each row of the matrix represents the instances in a predicted class while each column represents the instances in an actual class (or vice versa). All correct predictions are located in the diagonal of the table (highlighted in bold), so it is easy to visually inspect the table for prediction errors, as they will be represented by values outside the diagonal.

Table 6-19 depicts the confusion matrix, with two dimensions ("predicted star ratings" and "actual star ratings"), for the results of our first distribution of stars to *SenGeneral(D)* values using a nonlinear approach with the *MAE = 0.999*.

Table 6-19: Confusion matrix

		Actual star ratings				
		1	2	3	4	5
Predicted star ratings	1	**22,122**	14,915	9,837	8,462	5,039
	2	14,404	**13,217**	13,508	16,663	12,544
	3	10,240	13,012	**19,495**	34,461	33,715
	4	9,117	17,219	35,971	**82,740**	86,455
	5	4,534	11,898	32,378	88,975	**95,483**

Figure 6-15 depicts that, if we tolerate one-star estimation error, then we calculate the *Accuracy rate =75.70%*, and observe that for 534,758 reviews, the predicted star ratings are fairly accurate.

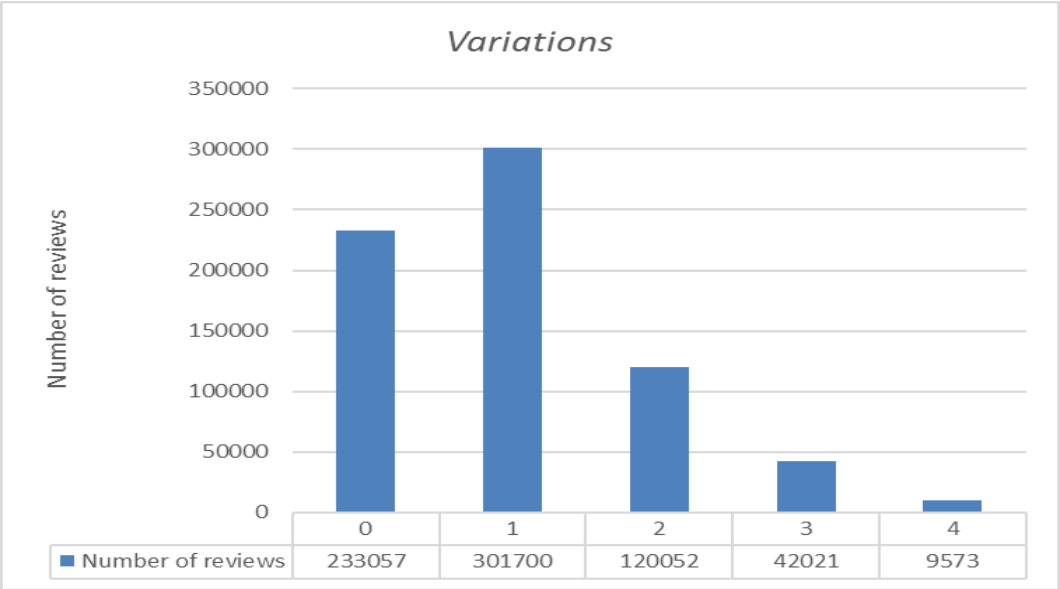


Figure 6-15: Number of reviews with star ratings prediction errors, varying between 0 and 4

6.7.4 A comparison of explicit and implicit measures of opinions

Another way of verifying whether the stars assigned by a user to a restaurant, faithfully reflect his sentiment which is expressed in his review, would be to consider each actual star ratings group separately. In other words, we consider only reviews with similar star rating (e.g., 1 star) at once. Then we count how many reviews of each group are considered as positive and how many of them are considered as negative by our prediction. Here again, the polarity score value *SenGeneral(D)* is used to seprate each group of reviews to two categories of positive and negative reviews. A (Score -) means that the calculated overall polarity of review is negative.

$$Score - : SenGeneral(D) \ < 0 \qquad (6.13)$$

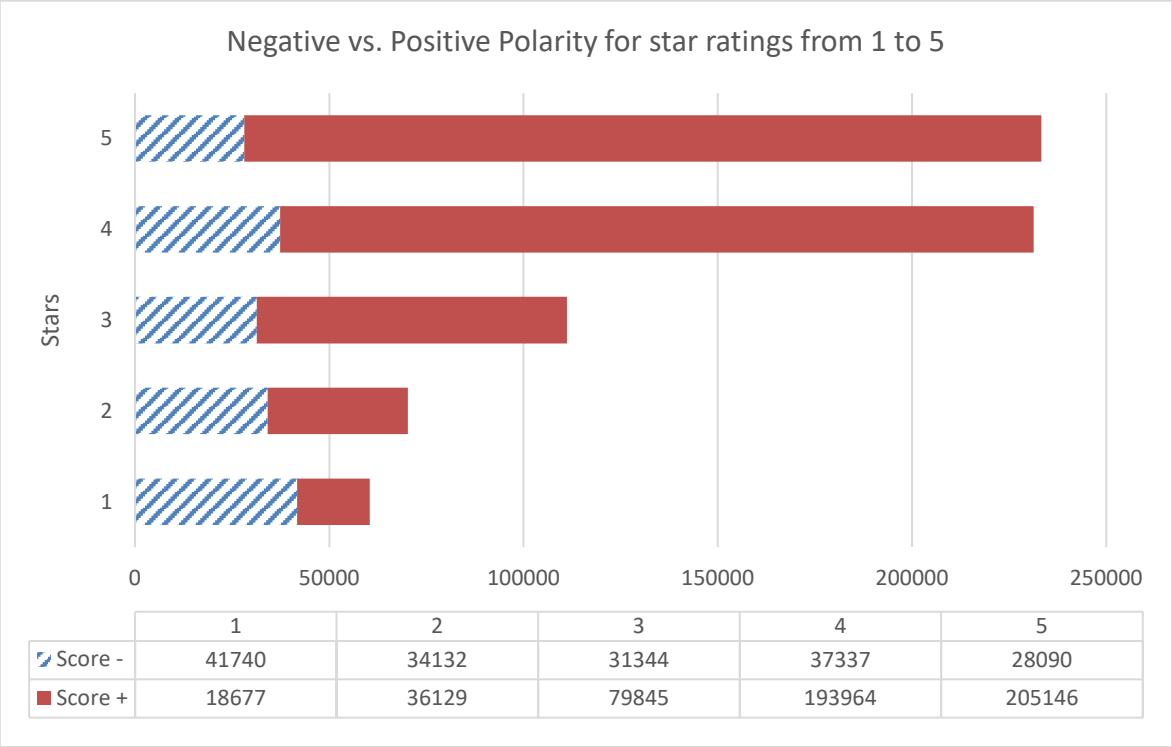
A (Score +) means that the calculated overall polarity of review is positive.

$$Score + : SenGeneral(D) > \ 0 \qquad (6.14)$$

Therefore, for each star ranking we count the number of predicted positive and negative reviews belonging to its class. Then we compare the number of "*Score+*" reviews with the number of "*Score-*" reviews within the same class.

By doing this comparison we aim to find out whether there is a correlation between the actual star ratings and the predicted polarities and to see how strong the predicted polarities repre-

sent each ranking class? In the Figure 6-16 we can see that for the total number of 60,417 reviews with 1-star ratings, there are 41,740 reviews with a negative polarity prediction (*Score-*) vs. 18,677 reviews with a positive polarity prediction (*Score+*).

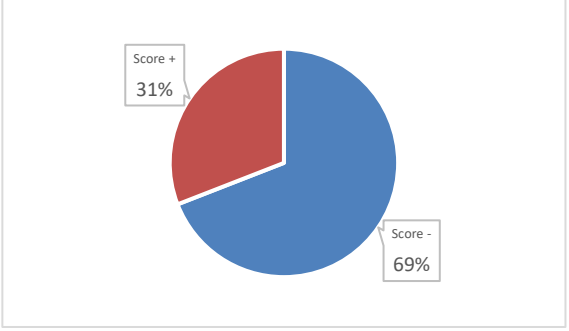


	1	2	3	4	5
Score -	41740	34132	31344	37337	28090
Score +	18677	36129	79845	193964	205146

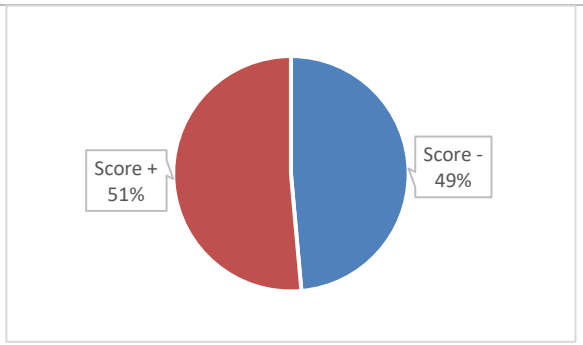
Figure 6-16: Negative vs. Positive Polarity for stars from 1 to 5

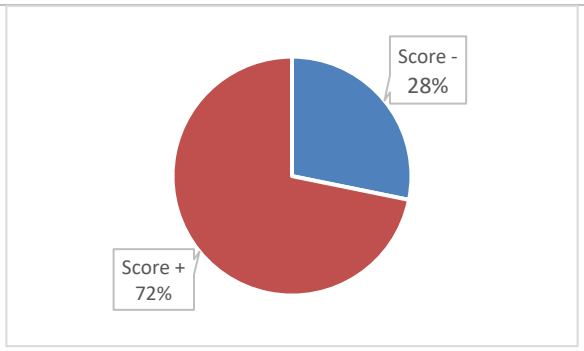
Table 6-20 depicts the relation between each star rating class and the number of positive and negative reviews in that class. For example, 84% of those users who gave four stars to a business, have also wrote a review with a positive polarity.

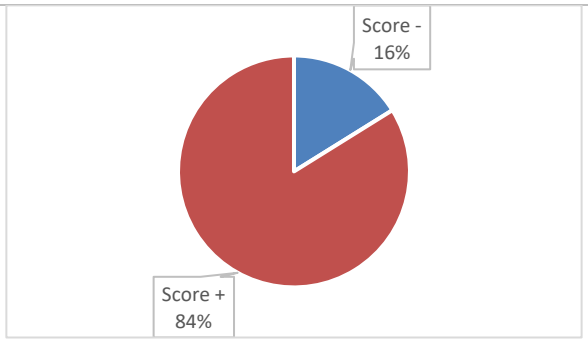
Table 6-20: Comparing the polarity of a review with its number of stars from 1 to 5

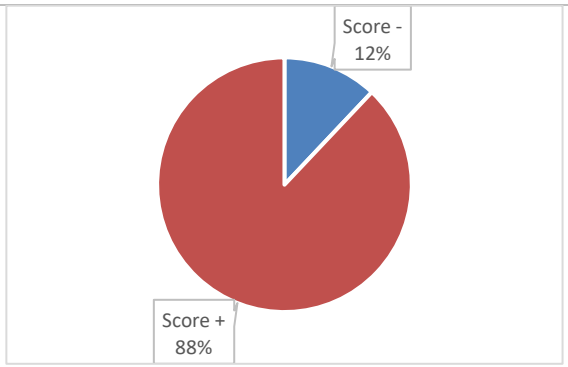
Stars (Integer)	Score -	Score +	Total	Score – vs. Score +
1 star	41,740	18,677	60,417	
The result interpretation: 69% (41,740 is 60% of 60,417) of those users who gave one star to a business, have also wrote a review with a negative polarity				

2 stars	34,132	36,129	70,261
The result interpretation: 49% of those users who gave two stars to a business, have also wrote a review with a negative polarity			
3 stars	31,344	79,845	111,189
The result interpretation: 72% of those users who gave three stars to a business, have also wrote a review with a positive polarity			
4 stars	37,337	193,964	231,301
The result interpretation: 84% of those users who gave four stars to a business, have also wrote a review with a positive polarity			
5 stars	28,090	205,146	233,236
The result interpretation: 88% of those users who gave five stars to a business, have also wrote a review with a positive polarity			
Total	**172,643**	**533,761**	**706,404**
	24 %	76%	100%



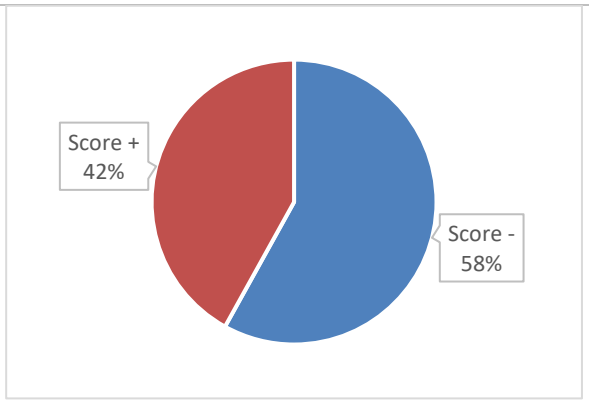
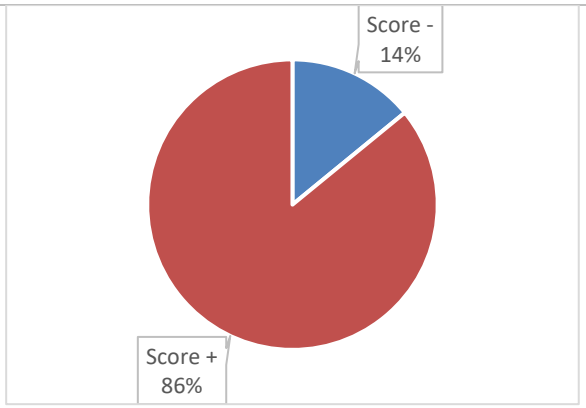






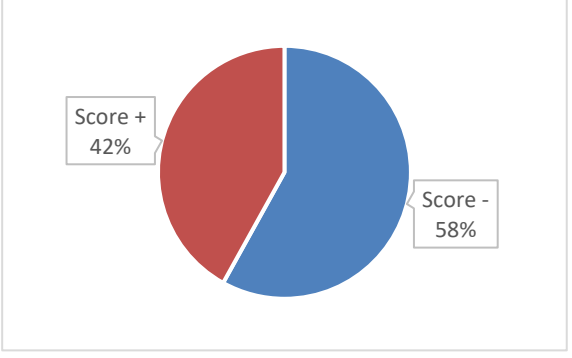
The Table 6-21 depicts that 58% of users consider 1-star or 2-stars ratings as negative, and 86% of them consider 4-stars or 5-stars ratings as positive.

Table 6-21: Comparing the normalized polarity of a review with its number of stars for (1 or 2) and (4 or 5)

Stars (Integer)	Score -	Score +	Total	Score – vs. Score +
1 or 2 stars	75,872	54,806	130,678	
The result interpretation: 58% of those users who gave one or two stars to the business, have wrote a review with a negative polarity				
4 or 5 stars	65,427	399,110	464,537	
The result interpretation: 86% of those users who gave four or five stars to the business, have wrote a review with a positive polarity				

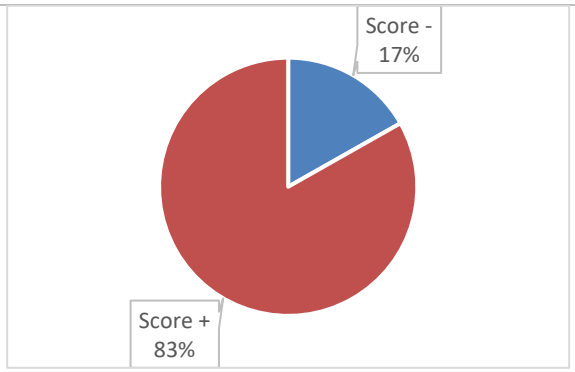
Alike, as it can be observed in Table 6-22, most of the Yelp customers consider the 3-stars to 5-stars ratings as positive. In fact, 83% of those users who gave three to five stars to a business, have wrote a review with a positive polarity.

Table 6-22: Comparing the normalized polarity of a review with its number of stars for (1 or 2) and (3 to 5)

Stars (Integer)	Score -	Score +	Total	Score – vs. Score +
1 or 2 stars	75,872	54,806	130,678	
The result interpretation: 58% of those users who gave one or two stars to a business, have wrote a review with a negative polarity				

3 to 5 stars	96,771	478,955	575,726

The result interpretation: 83% of those users who gave three to five stars to a business, have wrote a review with a positive polarity



6.8 Category detection and extraction for sentiment classification

Our challenge becomes even more interesting when it comes to detect different categories which are raised in a review and find out the polarity of these categories.

Therefore, we push further our research, and we make sentiment analysis not only for a review but also for each specific category raised in that review. First, a list of seed words which are believed to best represent each category is compiled. Then for each list we create a dictionary of words including their polarity score value attributed by the most commonly used meaning of the word in SentiWordNet. The complete dictionary of all words (#1 = the first sense) which were found in SentiWordNet has 18,977 rows, including all PoS (a, n, r, and v) terms.

Table 6-23: Row count vs. Seed word count of words for 13 categories

No.	Attribute (category)	Row count in dictionary of (#1) words	Seed word count
1	ambience	123	147
2	food	209	579
3	hamburger	8	21
4	lunch	102	117
5	meals	122	138
6	noises	51	91
7	pizza	48	41
8	quality price ratio	309	381
9	restaurants	106	119
10	sandwiches	37	41
11	seafood	19	26
12	service	64	173
13	steak	30	35

As it is depicted in Table 6-24, we create a separate list of specific seed words for each category which will be used for the category detection. For each category, we also create a list of positive sentiment words and a list of negative sentiment words used to describe the polarity of each category. For example, words *"beef, beer, berry, biscuit, black-berry, blueberry, bread, break-fast, and broccoli"* are attribute words for the category *"Food"*. Then, words *"divine, enjoyable, excellent, ex-quisite, extraordinary, and fantastic"* are positive sentiment words, and words *"tasteless, undercooked, un-derwhelming, unpalatable, andunpleasant"* are negative sentiment words used to describe the *"Food"* category.

Here we separate the task of "category detection" from the task of "opinion mining". First, we search all the seed word lists to detect whether a sentence is talking (or is covering) one or more categories.

Then, as we did for the first sentence of each review, we use equations (6.1), (6.2), (6.5) and (6.6) to determine whether each phrase of review is negative or positive. By combining the results from the category detection and opinion mining in sentence level, not only we can detect the existence of a category in a sentence but also, we can predict its respective polarity.

It is important to note that the category detection results are very depending to the quality of the specific seed words of each category.

Table 6-24: Food category and its specific sentiment words (positive and negative)

Food category
Attribute words
alcohol, appetizers, apple, apricot, asparagus, bacon, bake, banana, barbecue, basil, batter, beans, beef, beer, berry, biscuit, blackberry, blueberry, bread, breakfast, broccoli, brownie, brunch, burger, butter, cake, candy, caramel, carrot, caviar, cellar, cheese, cheesecake, chicken, chili, chips, chocolate, citron, cocktails, coffee, cook, cookie, cornbread, course, crab, cream, crepe, cucumber, cupcake, curry, dates, dessert, din-ner, dish, dressing, drink, eat, egg, eggplant, entree, entrée, fillet, fire, fish, flour, food, fork, fries, garlic, gastronomy, glasses, gluten, grapefruit, gravy, ham, hamburger, honey, hummus, ice, ingredients, jam, jelly, juice, kebab, ketchup, kettle, kiwi, lamb, lard, lasagna, legumes, lemon, lemonade, lentils, lunch, lychee, mac, macaroni, main, margarine, marshmallow, mayonnaise, meat, meatball, melon, menu, meringue, milk, milkshake, minerals, mocha, mock, mozzarella, muffin, mushroom, mustard, nectar, nectarine, nonalco-holic, noodles, oil, olive, omelet,

onion, options, orange, order, oregano, oven, pancake, papaya, parsley, pasta, pepperoni, persimmon, pizza, popcorn, pork, portions, potato, potatoes, pudding, quiche, quinoa, raisin, raspberry, ravioli, refreshments, restaurant, rice, roast, saffron, salad, salads, salami, salt, sandwich, sauce, sausage, selection, shrimp, side, snack, soda, sorbet, sorghum, soup, spaghetti, spicy, spinach, steak, sugar, sushi, sweet, taco, tail, take-out, taste, tea, tequila, toast, toffee, tofu, tomato, torte, tortilla, turkey, vanilla, vegetable, vegetarian, vinegar, vodka, wafer, waffle, wasabi, water, watermelon, wine, yogurt

Positive Sentiment Words

addicting, appealing, appetizer, appetizing, aromatic, authentic, balsamic, bittersweet, cheesy, chocolaty, creamy, crispy, crumbly, crunchy, crusty, decent, delectable, delicious, delightful, delish, divine, enjoyable, excellent, exquisite, extraordinary, fantastic, fiery, finger-licking, flavorful, flavorsome, fresh, fried, fruity, good, gooey, grainy, great, hearty, heavenly, impressive, infused, juicy, lemony, light, lip-smacking, lovely, marvelous, mashed, meaty, mellow, mild, mouthwatering, multi-layered, mushy, nectarous, nice, nutty, oily, oniony, outstanding, overwhelming, peppery, perfect, pleasant, pleasing, real, redolent, refreshing, rich, roasted, robust, salty, salubrious, sapid, satisfactory, satisfying, savory, scrumptious, seasoned, smokey, smooth, special, spicy, sweet, sweet-and-sour, tasty, tender, terrific, toasted, toothsome, warm, whipped, wonderful, yummy, zesty, zingy

Negative Sentiment Words

awful, bland, cold, disappointing, inedible, light, limited, mold, nasty, notfresh, notgood, notthebest, off, rancid, rotten, salty, slimy, smell, soggy, sour, stale, tasteless, undercooked, underwhelming, unpalatable, unpleasant, unsatisfactory, unwashed, watery, withered

Some of #1 polarity scores from SentiWordNet, for words in food category

SynsetTerms	PosScore	NegScore	ObjScore	PoS	id
appealing#1	0.25	0	0.75	a	170358
aromatic#1	0	0	1	a	2641378
bitter#1	0	0	1	n	7889814
creamy#1	0	0	1	a	373811
cold#1	0	0.75	0.25	a	1251128
delicious#1	0.75	0	0.25	a	1807964
brown#1	0	0	1	v	320246

course#1	0	0	1	v	2067540
divine#1	0.375	0	0.625	v	2107588
fresh#1	0.375	0.625	0	a	1067694
good#1	0.75	0	0.25	a	1123148
good#1	0.5	0	0.5	n	5159725
extraordinary#1	0.625	0	0.375	a	1675190
fruity#1	0	0.25	0.75	a	2397119
appealing#1	0.25	0	0.75	a	170358

The objective is to investigate each review text to detect one or more of the 13 attributes (Table 6-23) and its/their polarity. Review texts are already automatically broken to distinct sentences (see section 6.5.5). We have also a list of the seed words for each category. Here is an example of text analysis, for category and polarity detection in sentence level.

By finding matching words from the list of the seed words, first we attribute one or more categories to each sentence. Then we calculate for each sentence its overall *SentGeneral(D)*. The results of analysis are depicted in Table 6-26.

Some seedwords in sentence level which probably will help us to detect a category are highlighted in bold.

{(**Worth** a stop for **quick refreshment** & **meal**.),(Solid restaurant with **easy access** from the 60 vic Gold Canyon.),(**Entrees** sit around $11 +\- $2 with complete selection of **Mexican beer** to complement.),(What's **notable** - **shrimp** (**taco**, **burrito** or **fried**).),(**Fajita**'s also score **high**.),(**Flavorful salsa** What's **not** - **flan**.),(If your expecting a **small delicate dome** shaped **custard** with **carmel sauce** you be **disappointed**...),(**Served** like **pie** with **crust** and **whipped cream**.)}

Table 6-25: The results of analysis for category detection and sentiment classification

Category detection	
Sentence	Category
Sent1	Meals; Service
Sent2	Restaurants
Sent3	Quality price ratio

Polarity prediction						
	A	B	C	D		
Sentence	PosScore	NegScore	Normalized PosScore	Normalize NegScore	A - B	C - D
Sent1	0.972	0.227	0.139	0.032	0.746	0.107
Sent2	0.961	0.663	0.096	0.066	0.297	0.030

	Food
Sent4	Food
Sent5	Food
Sent6	Food
Sent7	Food
Sent8	Service; Food

Sent3	0.828	0.167	0.075	0.015	0.661	0.060
Sent4	0.520	0.000	0.065	0.000	0.520	0.065
Sent5	0.347	0.116	0.069	0.023	0.231	0.046
Sent6	0.000	2.339	0.000	0.390	2.339	0.390
Sent7	1.148	2.272	0.077	0.151	1.125	0.075
Sent8	0.948	0.242	0.119	0.030	0.706	0.088

Finally, we combine the above two results from sentence level, and create a table with categories and their polarity for each review.

Table 6-26: The combined results of analysis for category detection and sentiment classification

Sentence	Text	SenGeneral(D)	Polarity	Category
Sent1	Worth a stop for quick refreshment & meal.	0.746	Score+	Meals; Service
Sent2	Solid restaurant with easy access from the 60 vic Gold Canyon.	0.297	Score+	Restaurants
Sent3	Entrees sit around $11 +\- $2 with complete selection of Mexican beer to complement.	0.661	Score+	Quality price ratio; Food
Sent4	What's notable - shrimp (taco, burrito or fried).	0.520	Score+	Food
Sent5	Fajita's also score high.	0.231	Score+	Food
Sent6	Flavorful salsa What's not - flan.	-2.339	Score-	Food
Sent7	If your expecting a small delicate dome shaped custard with carmel sauce you be disappointed...	-1.125	Score-	Food
Sent8	Served like pie with crust and whipped cream.	0.706	Score+	Service; Food

6.9 Summary

In this chapter, for the purpose of obtaining results which are statistically very significant, a total of 706,404 restaurant reviews are analyzed. For verifing if the stars assigned by a user to a restaurant, faithfully reflect his sentiment which is expressed in his review, we propose a review rating prediction framework. The confusion matrix which is used to compare "predicted star ratings" versus "actual star ratings" depicts that the predicted star ratings are fairly accurate.

In chapter seven, to predict the best placement for a business or food category, and to identify clusters of cities with similar traits, we use a collection of economic, demographic, social, and housing indicators. Then the comparison of predicted star ratings for different food categories over time is used to depict their evolution during a given period and in a specific region. Finally, for analyzing the competition a proportional symbol map is applied. The size of each circle is related to the average star raings of the restaurant and represents visually the location and differences in classification of competing restaurants.

Chapter 7 Recommendation over time

Very different from other existing recommender systems, as our main objective we want to suggest a recommendation system to also guide producers. To the best of my knowledge this corresponds to a new approach to recommender systems. The objective is to help a business owner to decide about his next investment in terms of choice of food category and/or to find the best emplacement for his new restaurant in a city or in a state.

7.1 Conceptualization

Conceptualization is the most important activity in the development of a system. Even though in order to demonstrate its feasibility most of the programming components of the system have already been tested (the proof of concept - demonstration by concrete example), the realization and implementation of a complete and working platform based on our recommender system falls outside the scope of this thesis.

In this chapter I will rather focus on the progression of the concept of recommender system, followed by performance expectations that make use of this system.

The system is relying on business owner to provide his preferences. The business owner enters the necessary information about his business project to the system, the information such as the category of his new restaurant and at least one potential place to implement his new business; the main task of system would be:

- to identify the same exact kind or the similar business,
- to identify the cities or regions, which are similar to the place chosen by the business owner, and to calculate the business success over time of the place or its peer regions,

- to predict the best placement for that business or food category, among the potential places either entered as preference by the business owner or one of its peer cities that identified by data comparison.

7.2 Peer city identification

Peer cities are cities that are experiencing similar trends or challenges. Identifying a city's peers can give needed context to producers who plan to expand their business in a new city. Selecting peer regions and performing comparisons are a useful analysis method for benchmarking, measuring, and predicting the growth of a business in a new region.

The first step to identifying a peer region is to determine what a business's economic and creative goals are and then look for a place that has a similar outlook. Other factors to be taken into consideration include the population size, the growth of the workforce, location to nearby regions, and whether it has similar business sectors.

There are many statistical datasets available, for different parts of the world, to help us find similar regions. We can name few of them here:

- For the United States, in 2017, the Federal Reserve Bank of Chicago developed the Peer City Identification Tool (PCIT) to help aspiring cities identify potential peers based on four key themes.

- For Europe countries and regions, and in order to provide a detailed picture of the diverse EU territories and to monitor EU regional policy targets, Eurostat has developed a range of statistics based on different classifications and typologies.

- For the Asia, Geo-Economic Dataset for Asia (GEDA)[25] covers 16 countries and regions in East Asia (i.e., ASEAN10, China, Japan, Korea, Taiwan, India, and Bangladesh) at the sub-national level and includes population, area, and Gross Domestic Product (GDP) by industry for 2005.

- For the Africa, IDE-JETRO, Institute of Developing Economies of Japan External Trade Organization[26], have compiled a geo-economic dataset for Africa, Europe, and

25 https://www.ide.go.jp/English/Data/Geda.html
26 https://www.ide.go.jp/English.html

the United States. For Africa, they use satellite imagery, land cover data and population data, and the number of mines to divide sectoral GDP at the national level to each subnational region. For Africa they cover the following regions/countries: Uganda, Ethiopia, Eritrea, Ghana, Gabonese Republic, Cameroon, Gambia, Kenya, Cote d'Ivoire, Democratic Republic of the Congo, Republic of the Congo, Zambia, Zaire, Sierra Leone, Zimbabwe, Sudan, Swaziland, Senegal, Somalia, Tanzania, Togo, Nigeria, Niger, Burkina Faso, Burundi, Botswana, Madagascar, Malawi, Mali, Mozambique, Liberia, Rwanda, and Lesotho.

The first step to identifying a peer region is to determine what a region's economic and creative industry goals are and then look for a place that has a similar outlook. We use a collection of economic, demographic, social, and housing indicators to identify clusters of cities with similar traits. Once a city's peer cities are identified, we combine this information with the retrieved and/or computed information from the Yelp dataset by performing Sentiment Analysis on reviews over time. Our recommender system then can predict the success degree of a business both for the base city of producer's choice, and for its peer cities. This prediction will serve as base for producer, so he can compare and find the best degree of success for his project.

In this chapter to illustrate the peer city identification possibility from these kinds of statistical datasets, we will first make a complete demonstration of the PCIT dataset and later will briefly explor the Eurostat dataset.

The PCIT tool uses a collection of economic, demographic, social, and housing indicators to identify clusters of cities with similar traits. All data and images can be exported, and the full underlying dataset of 960 U.S. cities and 28 indicators is available for download[27]. Table 7-1 depicts the available data for the two cities of "*Abilene*" and "*Akron*".

Table 7-1: Two examples of 960 cities in the dataset

city	**Abilene**	**Akron**
state	Texas	Ohio
fips	4,801,000	3,901,000

27 https://www.chicagofed.org/~/media/others/region/pcit/2018-peer-city-data-full.xlsx?la=en

Median Family Income	54,564	45,018
Owner Occupied	53.61486	51.04936
High 30 Gross Rent	48.46629	52.98673
Housing Pre 1980	62.86472	85.09879
Vacancy Rate	12.74238	13.51869
Bach Plus 25 up	22.03567	20.21382
Families with Children	49.30589	48.88217
Total Population	122,612	198,508
White	59.04234	59.5306
Foreign Born	6.050794	5.521188
Median Monthly Housing Costs	794	728
Labor Force 16 up	60.3005	62.30284
Unemployment 16 up	5.726673	11.16072
Manuf	4.840471	14.47389
Median Household Income	44,108	35,240
Poverty	13.47499	19.77468
BWDI	34.83399	48.63926
HWDI	32.33192	31.09872
proj_uid	abilene_TX	akron_OH
Home Affordability	2.30117	2.270148
Poverty Change 2000	2.613663	5.736901
Change LFP 00	-2.79754	-1.81039
Manuf 1970 Change	-59.8198	-62.3989
Family Income 2000 Change	-6.62267	-21.6934
Percent Pop 2064	60.35706	61.61112
Population Change 2000	5.763823	-8.55284
Display name	Abilene, Texas	Akron, Ohio
Log Total Population	11.71678	12.19858
Av Gini 2011 2014	0.334714	0.314237
Pre-Post Recession Change	0.026439	0.01862
City longitude	-99.756	-81.5185
City latitude	32.43567	41.07669
geo_flag	Place	Place

A theme-based framework organizes indicators around key issues. The PCIT dataset uses four key themes, allowing users to explore a variety of potential peers. In addition, it generates peer group median values for each variable, as well as the median for all cities in the dataset, enabling comparison across and within the cities. This perspective can provide further context, especially in identifying areas in which the subject city might deviate from its peers,

which can serve to highlight particular business challenges or opportunities. The four key themes of PCIT dataset: equity, resilience, the outlook, and housing, are listed below:

1. The “Equity” category identifies cities based on inclusion, access, and diversity. It considers a city's racial and socioeconomic composition.

2. “Resilience” addresses economic diversification by considering conditions and trends in manufacturing employment, labor force participation, and unemployment.

3. The “Outlook” theme explores cities that are considered similar based on factors such as immigration, family composition, age structure, and changes in population.

4. “Housing” addresses the regional real estate trends with metrics on home ownership, renting, competitiveness of housing stock, and housing vacancies.

7.2.1 Methodology

To identify peer cities, we perform a hierarchical cluster analysis on all 960 cities, using the variables included in a selected theme. A theme (category) is simply a combination of variables (indicators) that we know to be relevant to a data research criterion. For example, “Housing” theme measures housing affordability, tenure, and age of the housing stock.

A cluster analysis is a way of grouping data based on the similarity of responses to several variables. Thus, when for example “Housing” is selected as theme, only those variables which are related to this theme are considered and compared for the peer city identification.

A cluster analysis can be imagined as treating each subject “city and its data” as a "point" in space. The analysis then proceeds to identify "neighbors" for each city, by computing and comparing the distance (according to the chosen theme and thus its variables) between a pair of cities in two clusters, and these "neighbors" are city’s peers.

7.2.2 Cluster distance measures methods

Hierarchical clustering involves creating clusters that have a predetermined ordering from top to bottom. There are two types of hierarchical clustering, divisive (top-down) and agglomerative (bottom-up).

7.2.2.1 Divisive method

In divisive clustering method we assign all the observations to a single cluster and then partition the cluster to two least similar clusters. Finally, we proceed recursively on each cluster until there is one cluster for each observation.

7.2.2.2 Agglomerative method

In agglomerative clustering method we assign each observation to its own cluster. So, each subject city and its data is considered as a cluster. Then, compute the similarity (e.g., distance) between each of the clusters and join the two most similar clusters. Finally, repeat steps two and three until there is only a single cluster left.

7.2.3 Some of the most frequently used methods of hierarchical agglomerative clustering

The agglomerative clustering is the most common type of hierarchical clustering used to group objects in clusters based on their similarity. In order to decide which objects/clusters should be combined or divided, we need methods for measuring the similarity between objects. There are several methods of hierarchical agglomerative clustering. Here are a few:

- Single linkage
- Complete linkage
- Average linkage
- Ward's method

We opt for one of the most accurate methods – the Ward's method which is a criterion applied in agglomerative hierarchical cluster analysis. Here below the Ward's method is explained in detail.

7.2.3.1 Ward's method

In statistics, Ward's method is a criterion applied in hierarchical cluster analysis. Ward's minimum variance method is a special case of the objective function approach originally presented by Ward, J. H., Jr. (1963). Ward suggested a general agglomerative hierarchical clustering procedure, where the criterion for choosing the pair of clusters to merge at each step is based on the optimal value of an objective function. This objective function could be "any function

that reflects the investigator's purpose." Many of the standard clustering procedures are contained in this very general class. To illustrate the procedure, Ward used the example where the objective function is the error sum of squares, and this example is known as Ward's method or more precisely Ward's minimum variance method.

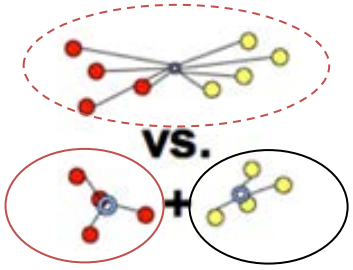


Figure 7-1: At each step, we find the pair of clusters that leads to minimum increase in total within-cluster variance after merging

The nearest neighbor chain algorithm can be used to find the same clustering defined by Ward's method, in time proportional to the size of the input distance matrix and space linear in the number of points being clustered.

The minimum variance criterion - Ward's minimum variance criterion minimizes the total within-cluster variance. To implement this method, at each step, we find the pair of clusters that leads to minimum increase in total within-cluster variance after merging. This increase is a weighted squared distance between cluster centers. At the initial step, all clusters are singletons (clusters containing a single point). To apply a recursive algorithm under this objective function, the initial distance between individual objects must be (proportional to) squared Euclidean distance.

The initial cluster distances in Ward's minimum variance method are therefore defined to be the squared Euclidean distance between points. Here X_i and X_j are two cluster centers and d_{ij} is the squared distance between them:

$$d_{ij} = d(\{X_i\}, \{X_j\}) = \|X_i - X_j\|^2 \qquad (7.1)$$

7.2.4 Choice of the clustering method

Finaly, to identify of approximately equal size clusters as output (for example, 5 to 15 cities identification as peer cities for the focus city), the Ward's clustering method is used for peer city identification. This method minimizes the variance across all variables in a given group.

Specifically, Ward's method minimizes the sum of the squared errors across all variables within a cluster, at each step of the clustering procedure. Each variable in each theme is normalized to have a standard deviation of 1.

If a cluster produces only a small number of results, the program instead uses the ranked values instead of the normalized values, which tends to produce more evenly distributed groups, but does not allow for easy distinction between extreme outliers and more typical cities. The cluster containing the focus city is expanded before the peer cities are presented for ease of explanation and verification, by including any other cities that have all variables fall between the cluster's maximum and minimum values on each variable.

7.2.5 Ward’s method for hierarchical cluster analysis with IBM SPSS

To illustrate the procedure of the Ward’s method, we use a sample of dataset. Table 7-2 contains only ten cities and eleven variables from the PCIT dataset which in total includes 960 cities and 28 indicators.

Table 7-2: a sample of 10 cities from the PCIT dataset

city / state	Cambridge / Massachusetts	Dallas / Texas	Hayward / California	Indianapolis / Indiana	Las Vegas / Nevada	Madison / Wisconsin	Ottumwa / Iowa	Portland / Oregon	San Diego / California	Waterloo / Iowa
Families with children	42.86	55.71	52.25	51.65	50.66	46.77	50.47	47.58	49.78	48.27
Family income change	24.38	-18.70	-8.39	-23.98	-18.45	-6.20	-7.74	2.73	3.59	-11.46
Home affordability	7.58	3.15	5.94	2.79	3.63	3.84	1.92	5.47	7.17	2.42
Labor force 16 up	69.07	68.10	66.74	67.55	63.67	72.45	62.90	69.55	67.32	66.13
Median family income	107897	48566	73165	54107	60078	81936	50237	75394	80241	55228
Median household income	83122	45215	68138	43101	50882	56464	38090	58423	68117	44146
Median monthly housing costs	1800	938	1547	859	1052	1107	641	1238	1573	723
Owner occupied	36.86	41.88	50.99	52.95	52.10	47.65	67.44	53.08	46.51	63.15
Poverty	8.40	19.36	10.26	16.09	12.57	8.20	16.00	10.55	10.33	12.68

Total population	108757	1278433	154507	8E+05	613295	246034	24709	620589	1374812	68357
Unemployment 16 up	5.43	6.81	7.97	9.06	10.61	4.81	7.42	7.53	7.73	8.31

In order to create an example of hierarchical cluster analysis which uses Ward's method, we use IBM SPSS Statistics 20.0 software.

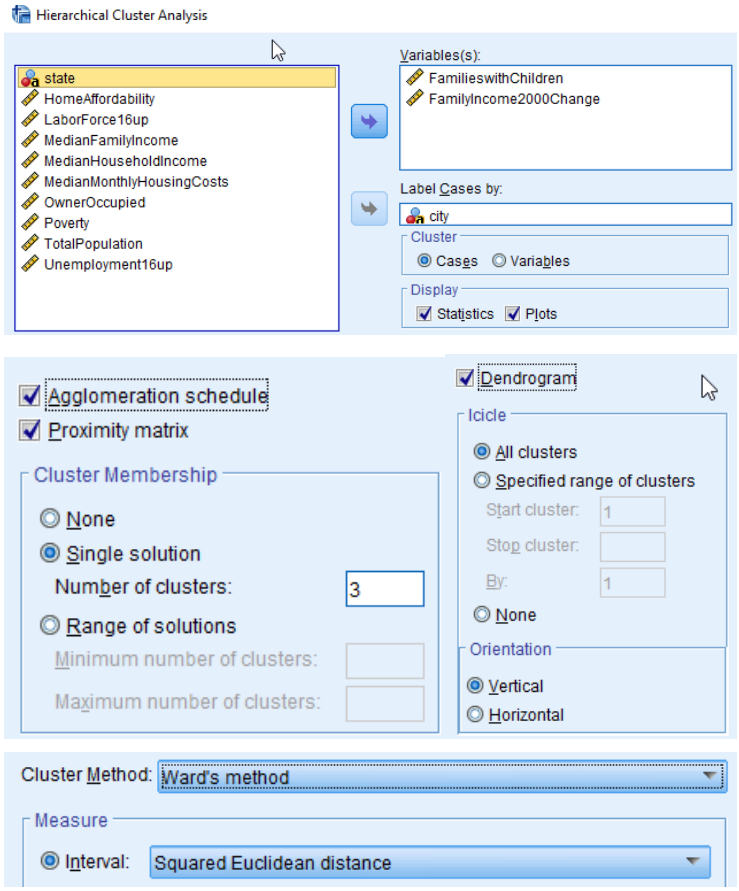


Figure 7-2: IBM SPSS configurations for sample data hierarchical cluster analysis

After loading the sample data from the Table7-2 to the IBM SPSS software, we configure the software as shown in the Figure 7-2, so the calculation takes into consideration only two variables of "*families with children*" and "*family income change*". The Ward's method is chosen for the hierarchical cluster analysis and squared Euclidean distance as interval. We setup the software to output a single solution by identify three clusters. To illustrates the arrangement of the clusters produced by the cluster analysis, we setup as output, also a dendrogram diagram, an agglomeration schedule table and the pairwise comparison matrix (proximity matrix or the dissimilarity matrix).

7.2.5.1 City cluster

A city cluster is a collection of cities which are "similar" between them and are "dissimilar" to the cities belonging to other clusters.

Cases					
Valid		Missing		Total	
N	Percent	N	Percent	N	Percent
10	100.0	0	.0	10	100.0

a. Ward Linkage

Figure 7-3: Case processing summary

Identifying a city's peers, does not mean that the cities are the same, but simply highlighting cities that are experiencing similar trends and challenges. So, it is important to note that "Peer ≠ Same".

To identify a city's peers, first we choose "*Las Vegas, Nevada*" as the base city, and then we select as indicators, "*families with children*" and "*family income change*" to identify clusters of cities with similar traits. Here is the output of IBM SPSS distance matrix. The matrix is symmetric, meaning that the numbers on the lower half will be the same as the numbers in the top half.

Table 7-3: Proximity matrix of ten cities

	Squared Euclidean Distance									
Case	1:Cambridge	2:Dallas	3:Hayward	4: Indianapolis	5:Las Vegas	6:Madison	7:Ottumwa	8:Portland	9:San Diego	10:Waterloo
1:Cambridge	.000	2021.038	1162.408	2415.887	1895.238	950.775	1090.031	490.775	480.040	1314.145
2:Dallas	2021.038	.000	118.158	44.328	25.568	235.944	147.402	525.598	532.106	107.703
3:Hayward	1162.408	118.158	.000	243.230	103.648	34.789	3.575	145.724	149.795	25.295
4:Indianapolis	2415.887	44.328	243.230	.000	31.555	339.688	264.925	730.217	763.673	168.033
5:Las Vegas	1895.238	25.568	103.648	31.555	.000	165.020	114.632	458.286	486.633	54.498
6:Madison	950.775	235.944	34.789	339.688	165.020	.000	16.069	80.557	105.021	29.901
7:Ottumwa	1090.031	147.402	3.575	264.925	114.632	16.069	.000	118.230	129.016	18.715
8:Portland	490.775	525.598	145.724	730.217	458.286	80.557	118.230	.000	5.597	202.115
9:San Diego	480.040	532.106	149.795	763.673	486.633	105.021	129.016	5.597	.000	229.018
10:Waterloo	1314.145	107.703	25.295	168.033	54.498	29.901	18.715	202.115	229.018	.000

This is a dissimilarity matrix

We have highlighted two numbers in this matrix. The distance between the 2 closest cities to *Las Vegas*. These two cells are the intersections of Las Vegas with Dallas and Indianapolis, in 2-dimensional space (each dimension representing a different variable).

7.2.5.2 Ward linkage

Table 7-4 depicts the cluster combinations by stage. In stage one the coefficient is calculated for clusters three and seven. In stage four the coefficient is calculated for clusters three, seven and ten, and so on.

Table 7-4: Agglomeration schedule

Stage	Cluster Combined		Coefficients	Stage Cluster First Appears		Next Stage
	Cluster 1	Cluster 2		Cluster 1	Cluster 2	
1	3	7	1.788	0	0	4
2	8	9	4.586	0	0	7
3	2	5	17.370	0	0	6
4	3	10	31.444	1	0	5
5	3	6	47.669	4	0	7
6	2	4	68.701	3	0	8
7	3	8	249.386	5	2	8
8	2	3	772.696	6	7	9
9	1	2	1877.460	0	8	0

As we can depict in the Figure 7-4 the three clusters begin to be formed (identified) only from stage six and up.

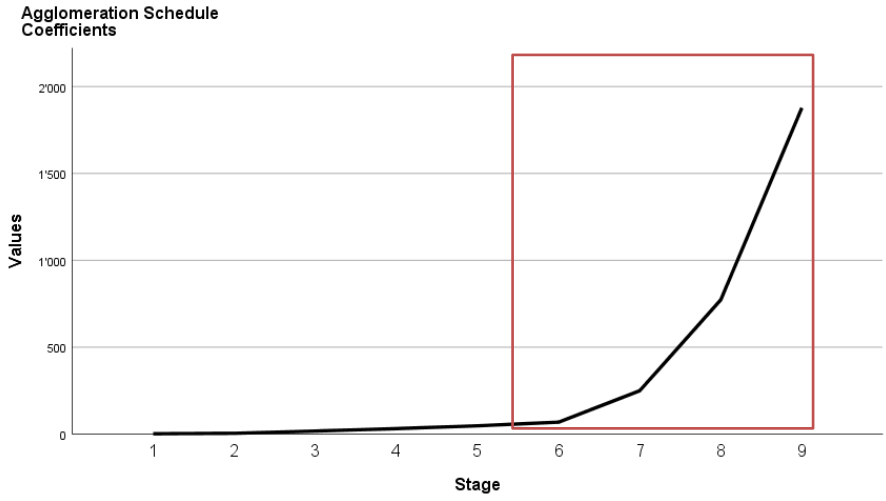
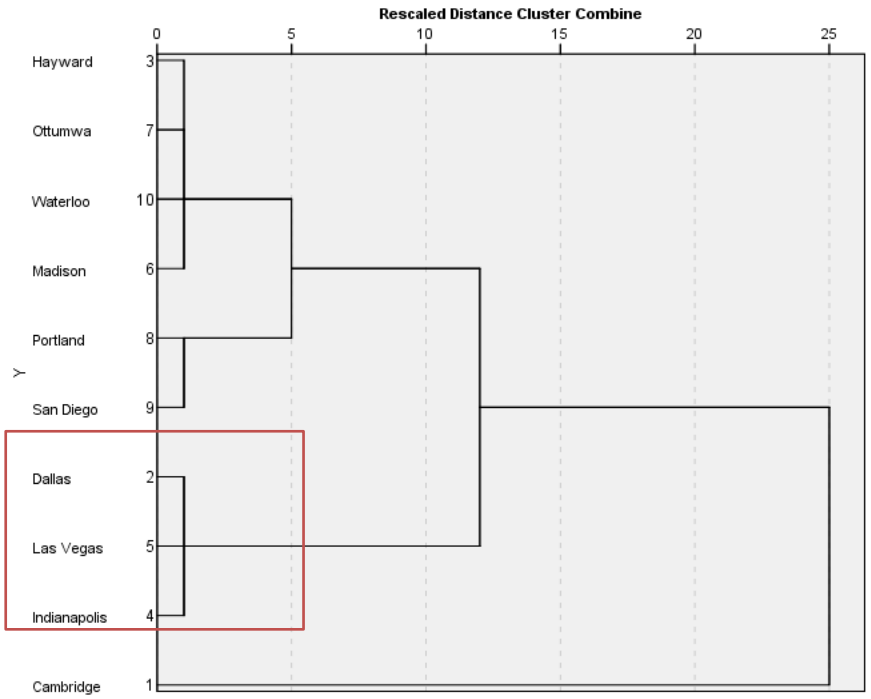


Figure 7-4: Clusters are formed from stage 6 and up

Figure 7-5 depicts a diagram representing a tree (dendrogram) which illustrates the arrangement of the clusters produced by the analyses.



Cluster Membership

Case	3 Clusters
1:Cambridge	1
2:Dallas	2
3:Hayward	3
4:Indianapolis	2
5:Las Vegas	2
6:Madison	3
7:Ottumwa	3
8:Portland	3
9:San Diego	3
10:Waterloo	3

Figure 7-5: Dendrogram using Ward linkage and the output cluster membership list

As it was configured in IBM SPSS software, "*families with children*" and "*family income change*" were the two variables chosen for this automatic clustering and as a result three distinctive clusters were identified. *Dallas*, *Indianapolis*, and *Las Vegas* are members of the same cluster (number 2), and so are considered as peer cities.

7.2.6 Example of peer city identification by theme

In the example below, *Las Vegas*, *Nevada* is selected as the base city and outlook is the theme. Outlook theme explores signs of city's demographic and economic future such as percent of foreign-born, percent of families with children, percent of population between the ages of 20 and 64, and the total population.

In Figure 7-6, the cities highlighted in red are the peer cities of *Las Vegas*. In a similar manner, one of the other three themes of equity, resilience, or housing can be selected to see how it aligns and deviates compared to peer cities.

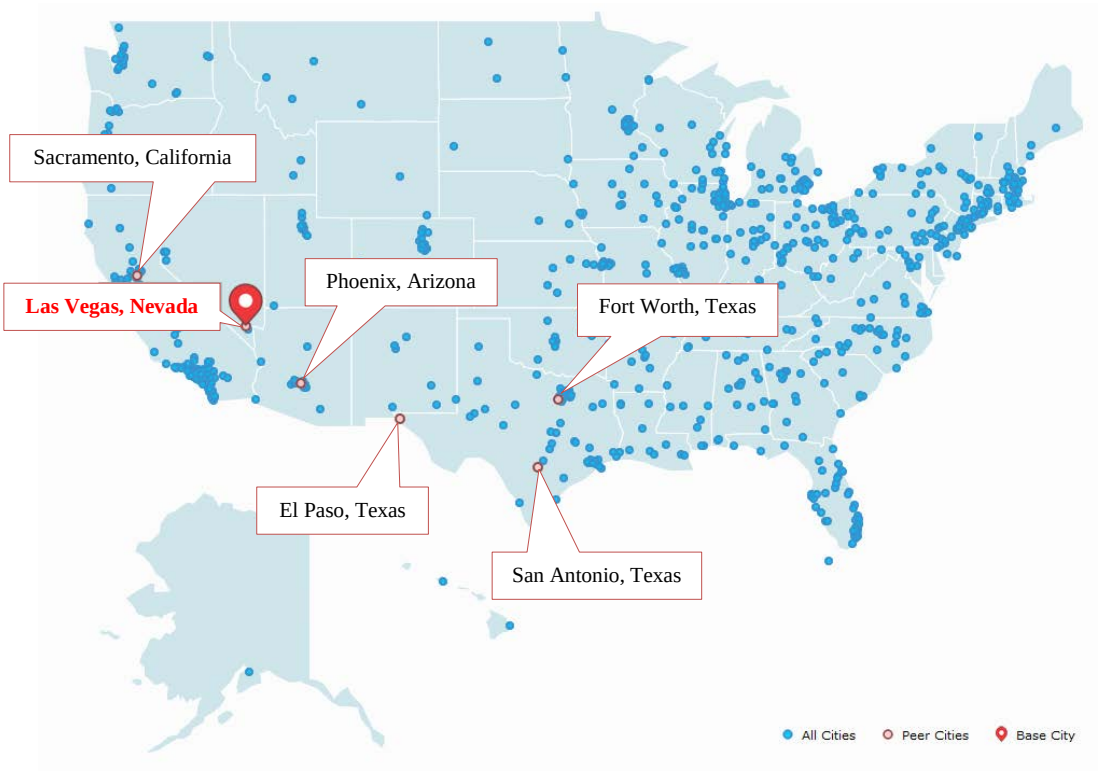


Figure 7-6: Las Vegas, Nevada is the base city and cities highlighted in red are its peer cities

Table 7-5 depicts the results for *Las Vegas* peer cities when outlook peer group is selected. Depending to the variable (or a combination of variables) under the same theme which might be more important for a business, the system will suggest one of the five different cities of *Fort Worth (Texas)*, *Phoenix (Arizona)*, *El Paso (Texas)*, *San Antonio (Texas)*, or *Sacramento (California)* as the perfect sister city for *Las Vegas (Nevada)*.

Table 7-5: The outlook theme explores cities that are considered similar to *Las Vegas*

Outlook Peer Group						
Peer cities	% foreign-born	% change in population, 2000-2017	% of families with children	**% of population 20-64**	Total population	Share of metro-politan area population
PCIT-960 Median	12.1%	8%	49.6%	60.1%	72,045	6.1%
Peer Group Median	20.4%	24%	53.8%	59.8%	756,698	31.9%
Fort Worth, Texas	16.9%	56.2%	58.7%	**59.6%**	835,129	11.8%
Phoenix, Arizona	19.6%	19.2%	55.8%	60.6%	1,574,421	34.5%

El Paso, Texas	24.5%	20.3%	54%	57.7%	678,266	80.9%
San Antonio, Texas	14.2%	27.7%	53.5%	60%	1,461,623	61.5%
Sacramento, California	22.6%	20.3%	53.3%	61.8%	489,650	21.6%
Las Vegas, Nevada	21.2%	29.9%	50.1%	**59.6%**	621,662	29.4%

In this example, as depicted in Figure 7-7, we can observe that *Fort Worth* city in *Texas* is the perfect peer for *Las Vegas* in *Nevada* when it comes to their percentage (59.60%) of residents between the ages of 20 and 64.

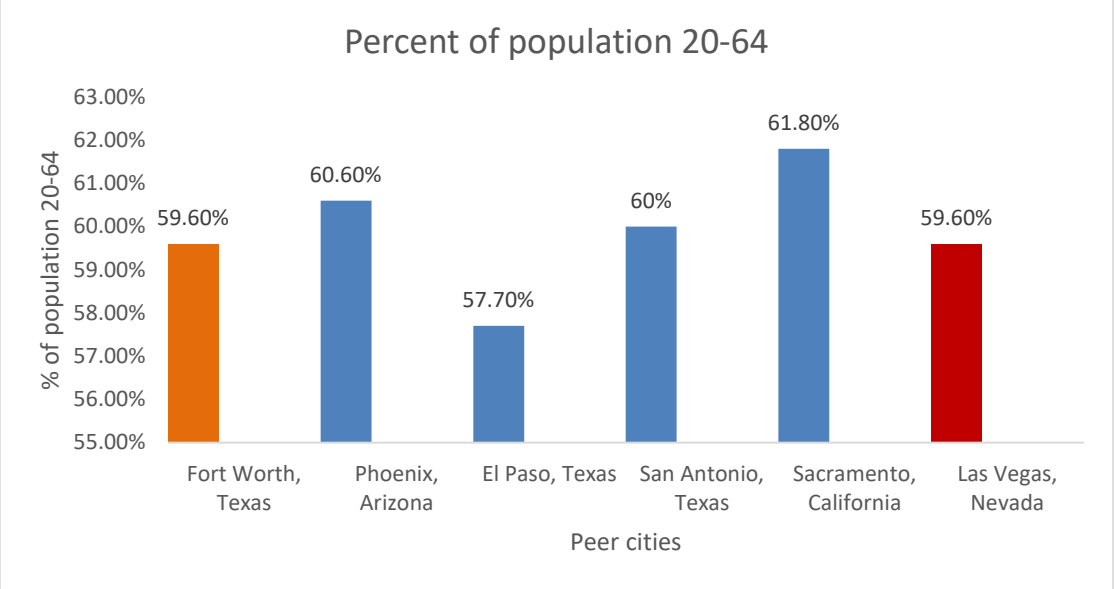


Figure 7-7: The percent of population 20-64 in Las Vegas is compared to its peer cities

7.2.7 Eurostat statistical dataset

With over 4,600 datasets containing more than 1.2 billion statistical data values, Eurostat is a mine of statistical information and covers all areas of European society. Eurostat is the statistical office of the European Union situated in Luxembourg. Its mission is to provide high quality statistics for Europe.

7.2.7.1 What kind of information is available?

In order to provide a detailed picture of the diverse EU territories and to monitor EU regional policy targets, Eurostat has developed a range of statistics based on different classifications and typologies. These include data for:

- regions
- cities and greater cities
- metropolitan regions
- rural areas and regions

Specific geographies such as coastal regions, mountain regions, border regions or island regions are also covered. For example, as depicted in Figure 7-8, concerning gender balance the city of Edinburgh is in position 28 out of 179 regions in United Kingdom. There are 105.4 women per 100 men.

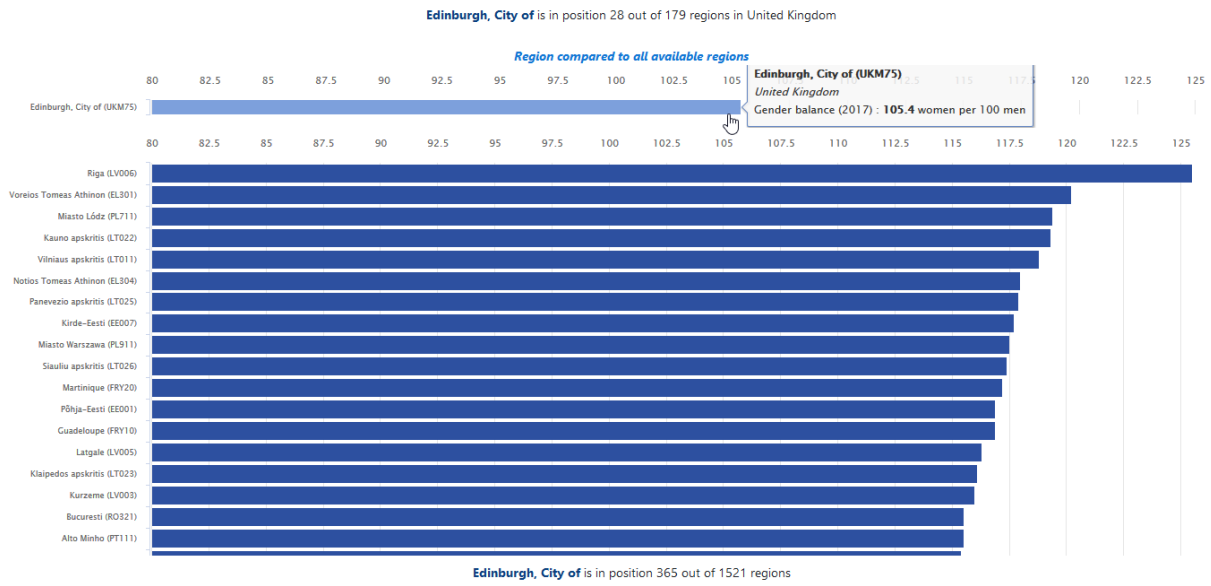


Figure 7-8: Gender balance of the city of Edinburgh in United Kingdom

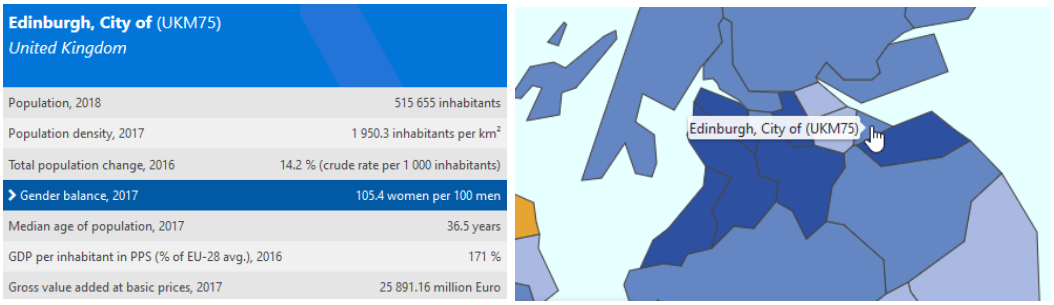


Figure 7-9: Population statistics of Edinburgh in United Kingdom

Eurostat's complete database or individual datasets can be downloaded[28].

On the bulk download you will find:

- all information updated twice a day, at 11:00 and 23:00,
- the datasets in tsv (tab separated values[29]) and sdmx format, which can be easily used to import the data in a tool of your choice,
- guidelines on how to automate the download of datasets,
- a manual containing all detailed information on the bulkdownload facility,
- the table of contents that includes the list of the datasets available,
- the "dictionnaries" of all the coding systems used in the datasets (Table 7-6).

Table 7-6: Example of one of the coding systems used in Eurostat dataset

TOTAL	**Total housing**
H_CV	Conventional housing
DW	Conventional dwellings
DW_OC	Occupied conventional dwellings
DW_OWN	Conventional dwellings occupied by the owner
DW_COOP	Conventional dwellings occupied by a member of the owning cooperative
DW_RENT	Conventional dwellings occupied by a tenant
DW_OTH	Conventional dwellings occupied under other type of ownership
DW_NOC	Unoccupied conventional dwellings
DW_SEC	Conventional dwellings reserved for seasonal or secondary use
DW_VAC	Vacant conventional dwellings
CLQ	Collective living quarters
CLQ_PRISA	Adult prisons
CLQ_PRISJ	Juvenile prisons
H_NCV	Non-conventional housing (incl. homelessness)
H_OTH	Non-conventional housing (excl. homelessness)
HMLS	Homelessness
H_OC	Total occupied housing (excl. homelessness)

28 https://ec.europa.eu/eurostat/estat-navtree-portlet-prod/BulkDownloadListing
29 "Tsv" files are flat files that include a "tab delimited" sequence of values in each line instead of one value per line/record.

Taking in consideration the very big amount of data available in the Eurostat database, it is essential to optimize the data by filtering it before going further. In other words, to find peer cities we need to compare only the data which is useful for a given business or restaurant category. For example, if a business has family-oriented products (category of food), the producer may give more consideration to the data available for the households' statistics.

There are a variety of data under the household's category, such as the population by household composition and number of children, employment by household composition, and working status within households. Table 7-7 depicts some examples of the Eurostat data navigation tree for the household's statistics.

Table 7-7: Example of data under the household's category in the Eurostat dataset

Households statistics - LFS series (lfst_hh)
Population by household composition and number of children or age of youngest child (lfst_hh_p) • Number of persons by sex, age groups, household composition and working status (1 000) (lfst_hhindws) • Number of persons by sex, age groups, household composition and educational attainment level (1 000) (lfst_hhinded) • Number of adults by sex, age groups, number of children, age of youngest child and working status (1 000) (lfst_hhacwnc) • Number of adults by sex, age groups, number of children, age of youngest child and educational attainment level (1 000) (lfst_hhacednc) • Number of adults by sex, age groups, number of children, age of youngest child and household composition (1 000) (lfst_hhaceday)

In addition, the Eurostat dataset enables us to benchmark the base city with other European peer cities with similar population size, income and employment, hence giving insights to develop our future business. Peer cities can also be represented by the same color on the map.

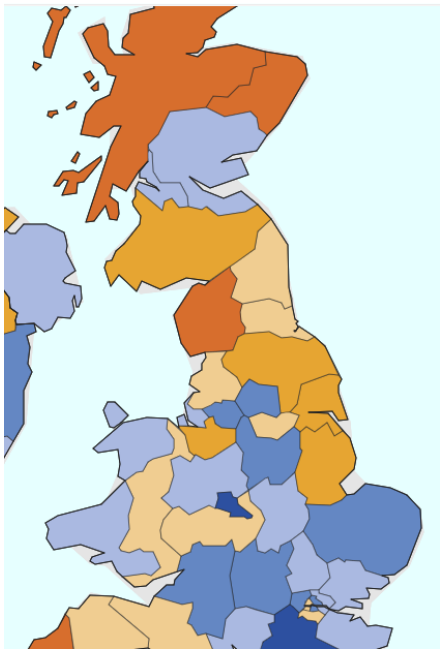


Figure 7-10: Eample of peer cities identified and presented by different colors in United Kingdom.

Even though, the very detailed nature of Eurostat's dataset allows a more focused comparison of cities in order to identify peer cities, the big number of variables may be confusing and less user friendly when it comes to identifying the most efficient variables to start with.

Therefore, one of the first steps of the implementation phase, would be simplifying (e.g. By feature selection) the end user interface and the preference entry page of the recommender system for the end users (producers).

One solution may be to compare different existing worldwide statistical datasets and try to identify a part of each dataset, as well as their set of variables, to create a single homogeneous dataset for all countries of the world.

In this way, we can manage custom themes to regroup specific indicators. For example, the “Resilience” theme can be used to describe economic change and labor market conditions and will only group those variables which are relevant to this theme.

7.3 Characteristics of our recommender system

Small business owners need guidance on various marketing and business services. Developing a Recommendation System that uses best practices and data to make recommendations to

the business owners on what will work best for them and their specific goals might seem like a difficult task.

Albert Einstein is quoted as having said that "*if he had one hour to save the world, he would spend fifty-five minutes defining the problem and only five minutes finding the solution*". This quote does illustrate an important point: before jumping right into solving a problem, we should step back and invest time and effort to improve our understanding of it.

So, let us break down the process into smaller steps, by pointing out some of the essential characteristics of our recommender system:

- The goal of our recommender system is to provide innovative and relevant information to help a business owner start a restaurant business.
- Food concepts, competition and location are the main domain characteristics in which recommendation takes place.
- A user-friendly interface where the controls and information are laid out in an intuitive and consistent way, and which is understandable and simple to navigate.
- Filters such as demographic, star ratings (current and predicted), and review date are used for soliciting data and for creating recommendations.
- To avoid overwhelming business owners with useless data and to assist them in focusing on interpreting the information, results (outputs) are presented in form of an analytical dashboard.

To examine the problem from a strictly business point of view, we choose a pragmatic approach as a problem-solving framework, by putting ourselves in business owner's shoes.

Finding the right kind of restaurant to open is a mixture of deciding what kind of business you want to run (restaurant concepts), selecting the right location (city), knowing your competition (competitors star ratings and location), and staying within your budget.

7.4 Preparing data for over time analysis

To begin, we need to identify relevant variables for which a statistical analysis of change in their value over time is possible. For our research we are interested in following data:

- Part of dataset which can be used to create time series (such as star ratings and predicted star ratings).
- Any data which can be considered as the geographic location information of a reastaurent (e.g., city, state).
- Information to spot a business competitive advantage (the business overall average star).

As mentioned earlier, consumers are given the opportunity to leave a business review, which includes choosing a grade from one star (poor) to five stars (excellent). Note that the overall business stars (rounded to half-stars) is the value of variable at the moment that the snapshot of Yelp dataset (Dataset, 2014) was created. Therefore, we consider it as a constant value (performance and quality indicator). It can be used for classification of a business.

The date of reviews each business has received over time is a variable which can also be considered as time information for other variables. We sort the reviews by date, so an older review appears before a newer one. Grouping these variables by year or quarter can help us to see how (according to the reviewers) business performances are changing by time. Are they improving or lagging behind?

To create the sample dataset for this study, as depicted in Table 7-8, we merge some data from the two tables of reviews and business based on business identification column, and we add a column to store the predicted star ratings data, which were calculated earlier (see section 6.7.3).

Table 7-8: Example of sample dataset

Review and business		**Concept**	**Competition**	**Location**
Review star rating	4	X		X
Review predicted star rating	5	X		X
Review date	2016-03-09	X		X
The city of the business	Las Vegas	X		X
The state of the business	CA	X		X
The business categories	["Mexican", "Burgers", etc.]	X		X
The business latitude	37.7817529521		X	
The business longitude	-122.39612197		X	
The business star rating	4.5		X	

7.5 Methodology

Now that we have our dataset ready, let us see how to use them for recommendation of each three main domain characteristics of food concepts, competition, and location.

7.5.1 Selecting a food concept

A total number of 42,153 businesses are provided in the Yelp database. Among them 21,892 have "Restaurants" as category, 10,878 have "Food" as category, and 27,952 have "Food OR Restaurants" as category. Therefore, there are many different types of restaurant concepts to choose from, when planning a new restaurant. The same issue applies when it comes to buy a franchise. Table 7-9 depicts a partial list of categories for restaurants.

Table 7-9: The non-exclusive Yelp category list for restaurants

Afghan	Cheesesteaks	French	Live Raw Food	Shanghainese
African	Chicken Wings	Fruits & Veggies	Malaysian	Shaved Ice
American New	Chinese	Gastropubs	Mediterranean	Singaporean
American Traditional	Colombian	Gelato	Mexican	Slovakian
Arabian	Comfort Food	German	Middle Eastern	Soul Food
Argentine	Creperies	Gluten-Free	Modern European	Soup
Asian Fusion	Cuban	Greek	Mongolian	Southern
Australian	Czech	Halal	Moroccan	Specialty Food
Bangladeshi	Delis	Hawaiian	Pakistani	Steakhouses
Barbeque	Desserts	Herbs & Spices	Patisserie Cake Shop	Street Vendors
Belgian	Dim Sum	Himalayan Nepalese	Persian Iranian	Sushi Bars
Brasseries	Do-It-Yourself Food	Hot Dogs	Peruvian	Szechuan
Brazilian	Dominican	Hot Pot	Pizza	Tapas Bars
Breakfast & Brunch	Donuts	Ice Cream & Frozen Yogurt	Polish	Tapas Small Plates
British	Egyptian	Indian	Portuguese	Tex-Mex
Buffets	Ethiopian	Indonesian	Poutineries	Thai
Burgers	Ethnic Food	Irish	Pretzels	Turkish
Burmese	Falafel	Island Pub	Ramen	Ukrainian
Cafeteria	Fast Food	Italian	Russian	Vegan
Cajun Creole	Filipino	Japanese	Salad	Vegetarian
Cambodian	Fish & Chips	Korean	Salvadoran	Venezuelan
Canadian New	Fondue	Kosher	Sandwiches	Vietnamese
Cantonese	Food Court	Laotian	Scandinavian	
Caribbean	Food Stands	Latin American	Scottish	
Caterers	Food Trucks	Lebanese	Seafood	

In order to help a business owner to make the right business decision, we can use the quarterly comparison of the average star ratings data of two concepts or two franchises. There are many ways of preparing and comparing these sets of values. For example, we can compare them in general or just for a specific period, and for a given city.

Below is an example of comparing “Burgers” as a restaurant category in general with four of particular restaurants of the same category.

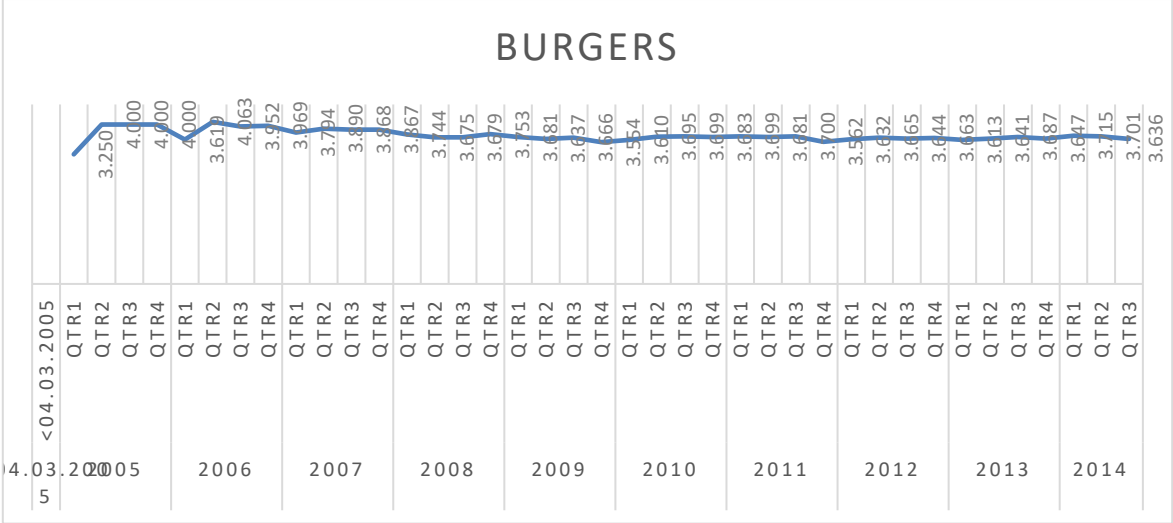
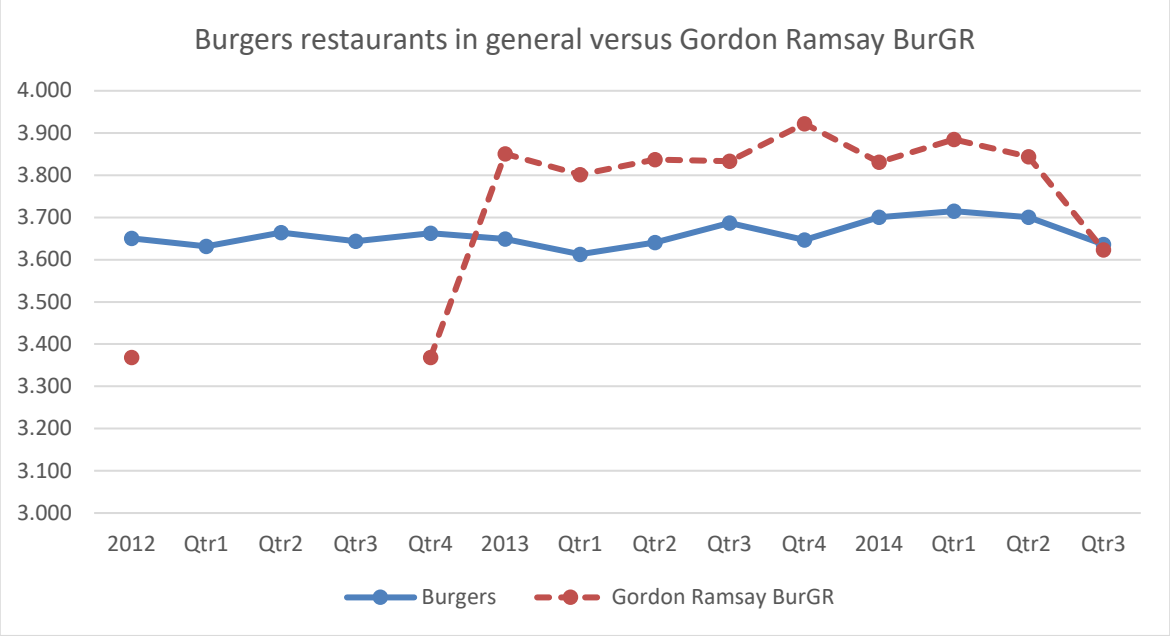


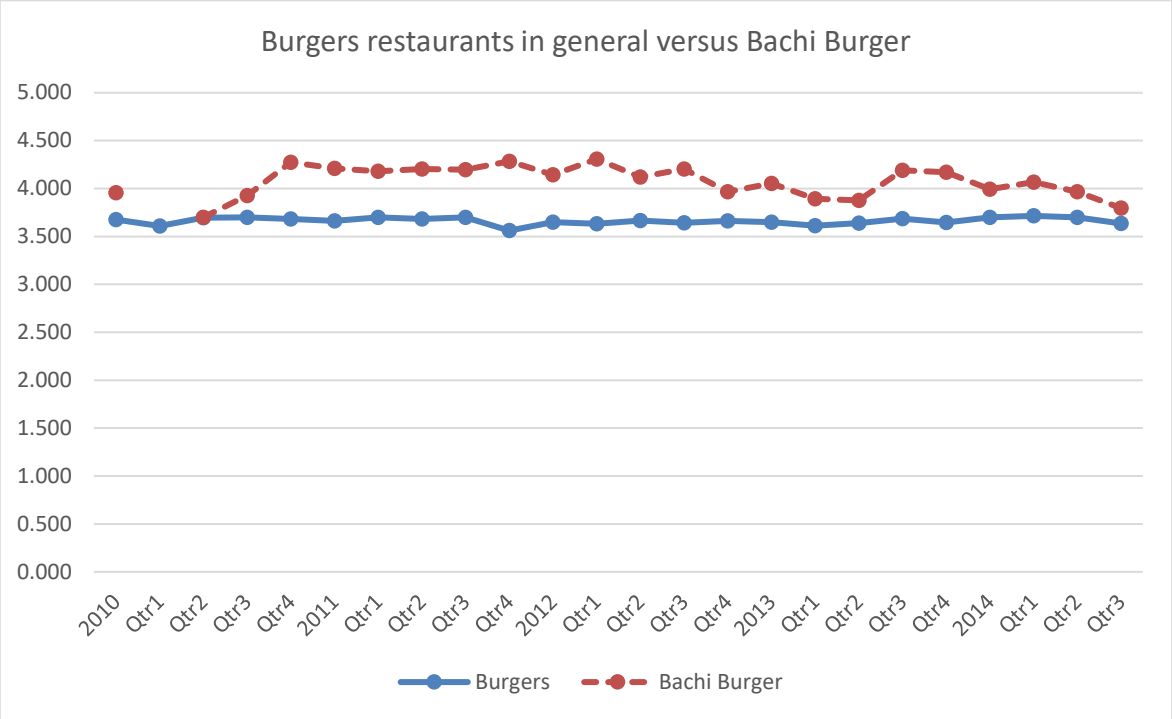
Figure 7-11: Over time average of star ratings for “Burgers” in general

	Burgers	Gordon Ramsay BurGR
2012	3.651	**3.368**
Quarter1	3.632	
Quarter2	3.665	
Quarter3	3.644	
Quarter4	3.663	**3.368**
2013	3.649	**3.851**
Quarter1	3.613	**3.801**
Quarter2	3.641	**3.837**
Quarter3	3.687	**3.833**
Quarter4	3.647	**3.922**
2014	3.701	**3.831**
Quarter1	3.715	**3.885**
Quarter2	3.701	**3.844**
Quarter3	3.636	**3.624**



Correlation coefficient: 0.234

Figure 7-12: Average star ratings of Burgers restaurants in general versus Gordon Ramsay BurGR

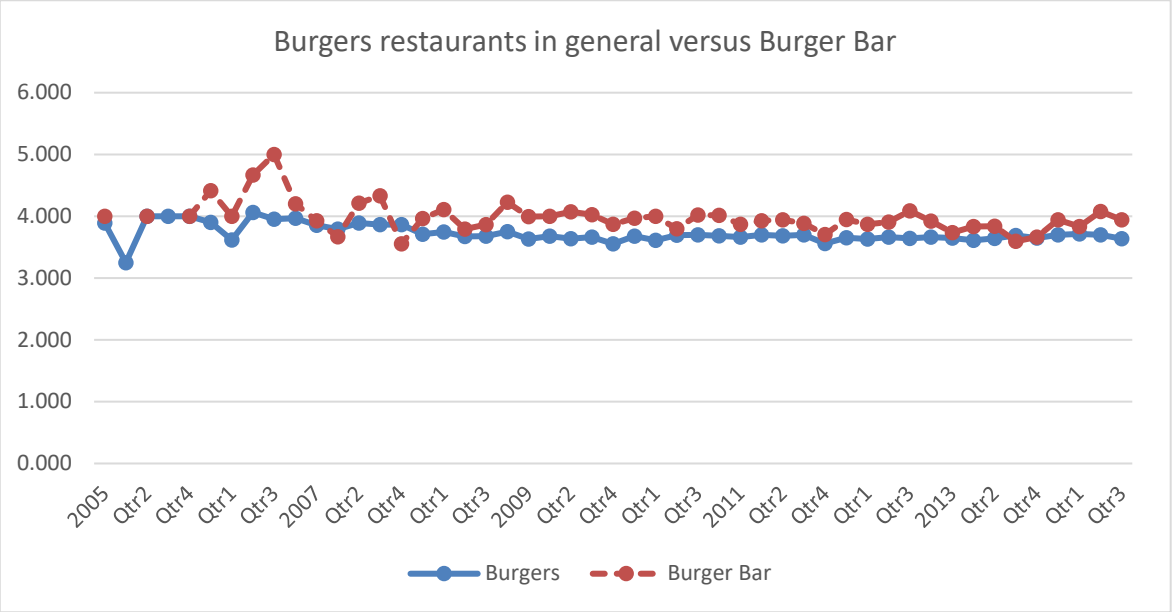


Correlation coefficient: -0.164

Figure 7-13: Average star ratings of Burgers restaurants in general versus Bachi Burger

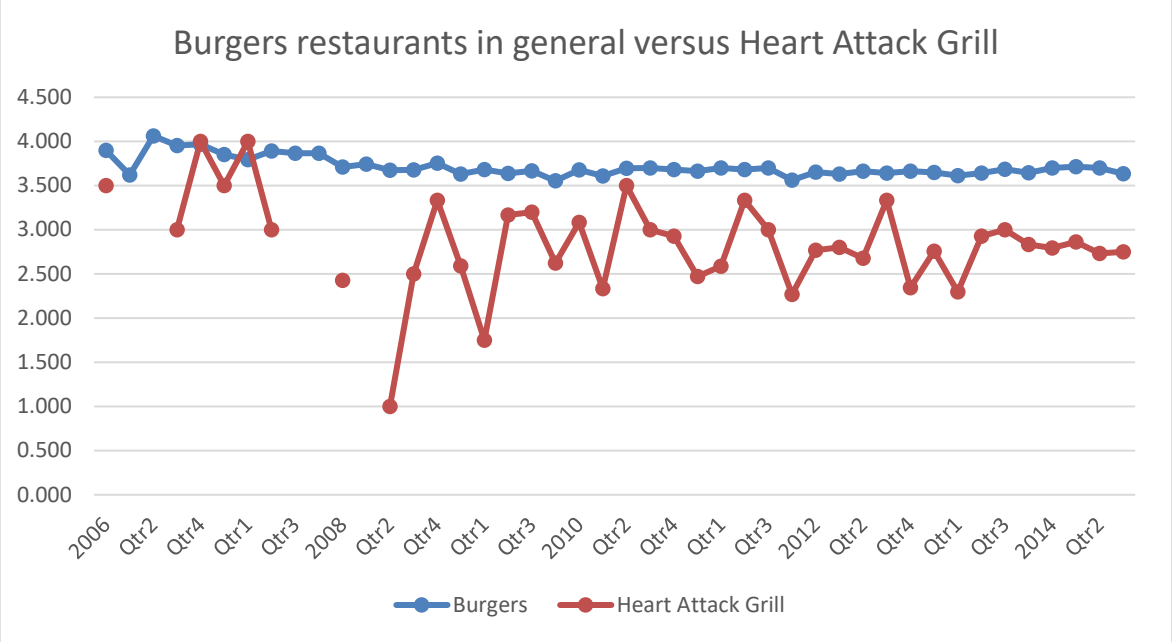
In each case, for an interactive analysis and comparison of categories, the Pearson's correlation coefficient is computed between the values of quarterly average star ratings of Burgers

restaurants in general, and the values of quarterly average star ratings of each restaurant.



Correlation coefficient: 0.550

Figure 7-14: Average star ratings of Burgers restaurants in general versus Burger Bar



Correlation coefficient: 0.5069

Figure 7-15: Average star ratings of Burgers restaurants in general versus Heart Attack Grill

As a second example, we can imagine that a business owner is hesitating among four different food categories such as “Pizza”, “Mexican”, “Chinese”, and “Fast Food”. In Table 7-10, these four food categories are compared according to the number of their overall star ratings. This

information could be interpreted as a helpful indicator of the success of a food category beyond the previous approach. As depicted in Table 7-10, there are 577 pizzerias with an overall star rating of 4. (More examples in *Appendix C*)

Table 7-10: Comparision of four food categories

Stars	Pizza	Mexican	Chinese	Fast Food
1	23	7	4	**37**
1.5	38	15	25	**134**
2	131	68	60	**256**
2.5	230	209	180	**395**
3	404	436	334	**454**
3.5	604	**680**	462	540
4	**577**	532	325	382
4.5	192	**229**	88	163
5	24	**32**	19	22
Total	2223	2208	1497	2383

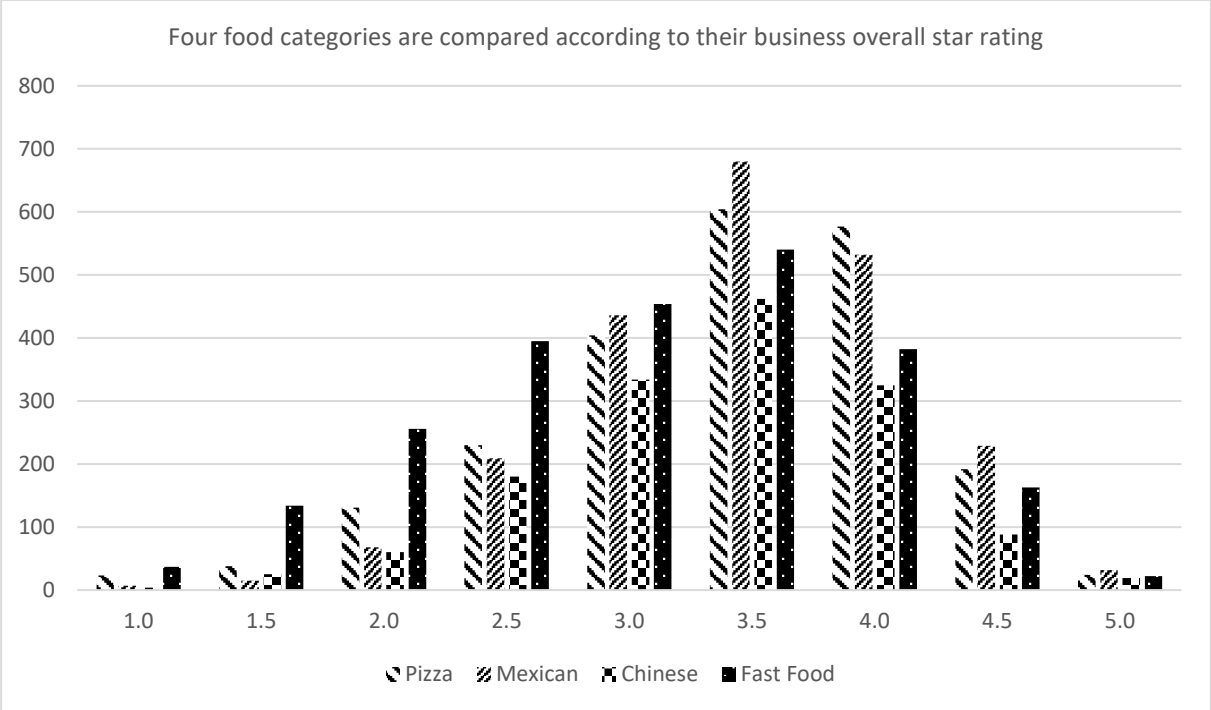


Figure 7-16: Overall business star rating for Pizza, Mexican, Chinese and Fast Food categories

As Figure 7-16 depicts, Chinese with 2 or more overall star ratings is a rarer event that Mexican with the same ratings. Fast Food category has always a higher rating compared to Chinese. Pizza and Mexican have highest ratings for 3.5 and 4 overall star ratings.

Now we can go one step further by comparing the success of these four food categories over time. For each category, we do this comparison, from 2006 to 2014, for Las Vegas in Nevada, based on either development of star ratings or predicted star ratings. As we can see from Figure 7-17, the over time comparison of star ratings depicts that from third quarter of 2009 the Pizza food category has the highest ratings, in Las Vegas, Nevada.

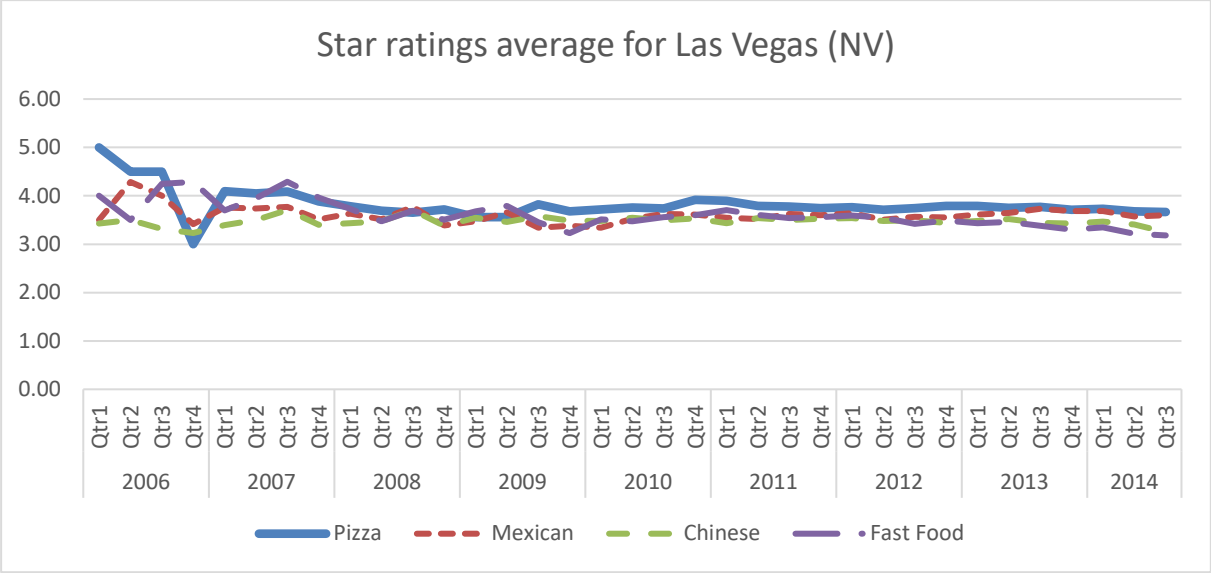


Figure 7-17: Comparing star ratings over time for Pizza, Mexican, Chinese and Fast Food categories

In Figure 7-18, the over time comparison of predicted star ratings depicts that by order, Pizza and Mexican food categories have the highest ratings, in Las Vegas, Nevada.

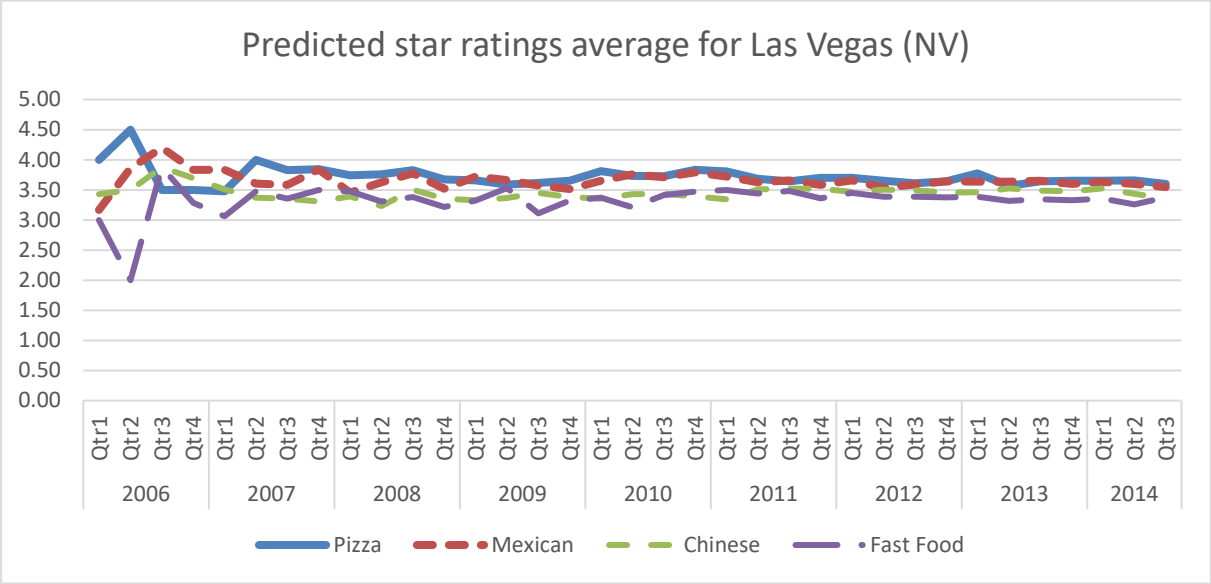


Figure 7-18: Comparing predicted star ratings over time for Pizza, Mexican, Chinese and Fast Food categories

The cold start problem could happen when the system does not have any form of data on a new type of restaurent.

7.5.2 Analyzing the competition

Are competing restaurants located nearby? To answer this question, we use three values of our data. The business latitude, longitude, and overall star ratings constant values. Like hotel ratings, the overall star ratings can be used as a performance and quality indicator to classify restaurants, accordingly, as perceived by their reviewers. The latitude and longitude features are used in order to draw geodetic features of a restaurant from the dataset.

Finally, a proportional symbol map is applied. It uses the above three values to represent visually the location and differences in classification of competing restaurants. Figure 7-19 depicts all the existing Burgers restaurants in Madison (WI) with 3.5, 4 or 4.5 overall ratings.

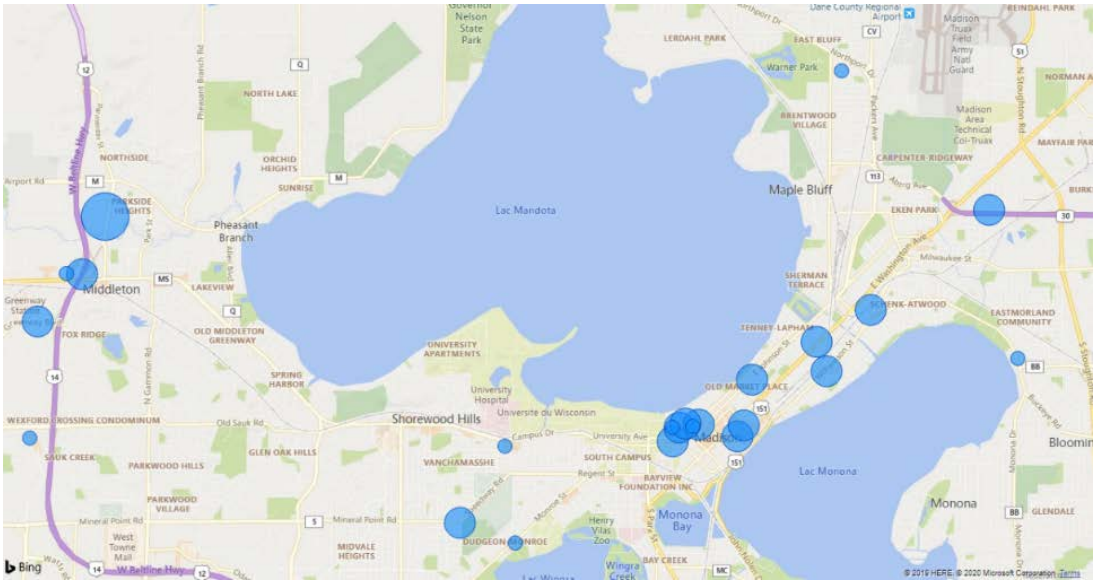


Figure 7-19: Burgers in Madison, in Wisconsin, with an overall star ratings of 3.5, 4 or 4.5

7.5.2.1 Proportional symbol maps

A proportional symbol map is a type of thematic map that uses map symbols that vary in size to represent a quantitative variable. For example, in a population proportional symbol map, San Francisco will have a larger dot than Las Vegas because it has a larger population.

The main idea behind proportional symbol maps is that a larger symbol means “more” of something at a location.

7.5.3 Find the perfect location for a new restaurant

Location is also key to the success of a new restaurant. Entering a market that’s oversaturated with dining options or where the population is on the decline can spell disaster for a fledgling restaurant.

Let us say that a restaurant owner in U.S. is planning to open a new branch of his restaurant. Before investing, he uses our recommendation system to identify the best city to start his new restaurant. The business owner enters his preferences into the system. At least one restaurant category and the name of one region (city) in which he is interested.

First the system will identify the peer cities of the base city indicated by the producer. Then, it will create a simplified version of the Table 7-8, by keeping only the date, star ratings, and predicted star ratings data for the city of preference and all its peer cities.

As an example, let us assume that, Las Vegas, in Nevada, is selected by a “Sushi Bars” owner as the base city and that the cities of Phoenix, in Arizona and Madison, in Wisconsin are identified by the system as its peer cities.

According to whether the quarterly average of star rating and their respective dates form regular time intervals or not, we are taken to follow one of the two following approaches:

7.5.4 Approach #1: Valid time series are formed

In this case, for each city, we use the autoregressive–moving-average (ARMA) time series forecasting model to predict future values of star ratings (or predicted star ratings) based on their previously stored values. Table 7-11 depicts forecast values of star ratings for “Sushi Bars” in Madison.

Table 7-11: Forecast results of star ratings for “Sush Bars” in Madison, WI

Timeline	Forecast	Confidence Interval
31.03.2014	3.66	0.94
30.06.2014	3.65	0.97
30.09.2014	3.64	1.00
31.12.2014	3.62	1.03

31.03.2015	3.61	1.05
30.06.2015	3.60	1.08

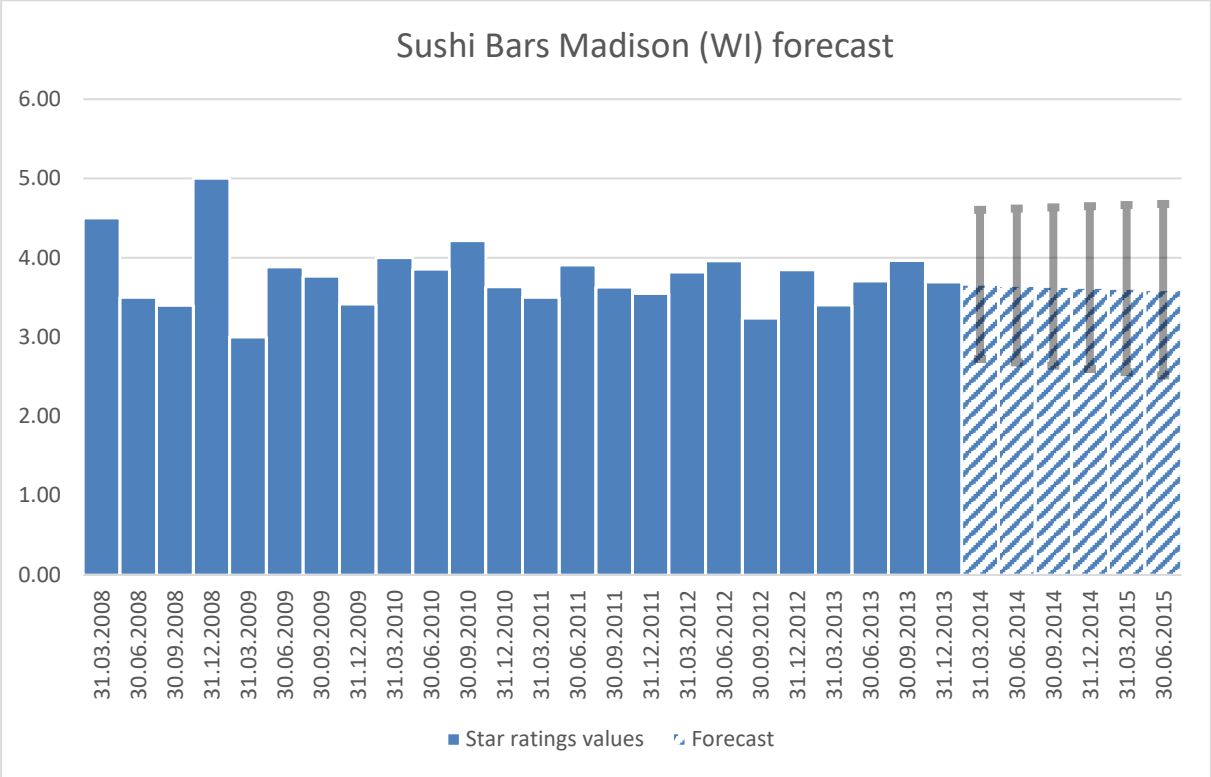


Figure 7-20: Time series forecasting model to predict future values of star ratings

The vertical error bars in Figure 7-20 depict the confidence interval of forecasting values. The same principle applies to other cities or food categories, and their results can be compared to predict their business success and failure.

The Mean Absolute Error (MAE) is a common measure of forecast error in time series analysis (see 6.7). In order to evaluate the forecast accuracy, the forecast is compared against a corresponding observation of what actually occurred. A small value indicates a better score, a perfect score is zero.

Table 7-12: Forecast verification of star ratings for “Sush Bars” in Madison, WI

Quarter	Real average star rating value	Forecast average star rating value
01.01.2014	3.53	3.66
01.04.2014	3.43	3.65
01.07.2014	3.08	3.64
MAE = 0.3		

In view of forecast verification, even though we have already their real values in our dataset, we forecast the average star ratings also for the first three quarters of 2014. Table 7-12 depicts that we calculate a small *MAE* of *0.3*, which indicates a good score. In my experience, autoregressive–moving-average (ARMA) model work best with monthly, quarterly, or yearly data. Below is an introduction to ARMA time series forecasting model which is used to predict star ratings values.

7.5.4.1 Introduction to time series forecasting

Any metric that is measured over "regular" time intervals forms a time series.

In the statistical analysis of time series, autoregressive–moving-average (ARMA) models provide a parsimonious description of a stationary stochastic process in terms of two polynomials, one for the autoregression (AR) and the second for the moving average (MA).

ARMA models are a popular and flexible class of forecasting model that utilize historical information to make predictions. This type of model is a basic forecasting technique that can be used as a foundation for more complex models.

Time series analysis can be used in a multitude of business applications for forecasting a quantity into the future and explaining its historical patterns. In our case, provided that the time intervals of time series are regular, and there are not missing data, it can be used for predicting the expected quarterly average of star ratings for a business in a given city. Between others, this information can be a good indicator to help a business owner decide about the location of his future restaurant.

Given a time series of data X_t, the ARMA model is a tool for understanding and, perhaps, predicting future values in this series. The AR part involves regressing the variable on its own lagged (i.e., past) values. The MA part involves modeling the error term as a linear combination of error terms occurring contemporaneously and at various times in the past. The model is usually referred to as the ARMA(p,q) model where p is the order of the AR part and q is the order of the MA part.

7.5.4.2 Autoregressive model

The notation AR(*p*) refers to the autoregressive model of order *p*. The AR(*p*) model is written

$$X_t = c + \sum_{i=1}^{p} \varphi_i X_{t-i} + \varepsilon_t \qquad (7.2)$$

where $\varphi_1, \ldots, \varphi_p$ are parameters, c is a constant, and the random variable ε_t is white noise. Some constraints are necessary on the values of the parameters so that the model remains stationary. For example, processes in the *AR(1)* model with $|\varphi_1| \geq 1$ are not stationary.

7.5.4.3 Moving-average model

The notation MA(*q*) refers to the moving average model of order *q*:

$$X_t = \mu + \varepsilon_t + \sum_{i=1}^{q} \theta_i \varepsilon_{t-i} \qquad (7.3)$$

where the $\theta_1, \ldots, \theta_q$ are the parameters of the model, μ is the expectation of X_t (often assumed to equal 0), and the $\varepsilon_t, \varepsilon_{t-1}, \ldots$ are again, white noise error terms.

7.5.4.4 ARMA model

The notation ARMA(*p*, *q*) refers to the model with *p* autoregressive terms and *q* moving-average terms. This model contains the AR(*p*) and MA(*q*) models,

$$X_t = c + \varepsilon_t + \sum_{i=1}^{p} \varphi_i X_{t-i} + \sum_{i=1}^{q} \theta_i \varepsilon_{t-i} \qquad (7.4)$$

Note that the model above assumes non-seasonal series. The purpose of seasonal adjustment is to remove systematic calendar-related variation associated with the time of the year, that is seasonal effects. For example, if December's sales are typically 150% of the normal monthly value (based on historical data), then each December's sales would be seasonally adjusted by dividing by 1.5. Since we consider over time star ratings data as non-seasonal series, we do not need to de-seasonalize the series before modeling.

It is essential to highlight here that the ARMA model can be used if the timeline is evenly spaced.

7.5.5 Approach #2: Valid time series are not formed

The Yelp dataset used for this study is a subset of their businesses, reviews, and user data. Thus, it is obvious that sometimes the missing values (e.g., date and value of star ratings) are observed, and time series are therefore not formed by default.

A linear regression's equation is $y = \alpha + \rho x$, where x is the explanatory variable and y is the dependent variable. The slope or regression coefficient of the line is ρ, and α is the y-intercept (the value of y when $x = 0$)

Our approach #2 is to use a linear relationship as a simple method to predict the average value of star ratings (y) for a given date (x) using the regression line. If the ρ and the α of that regression line are known, then we can plug in a value for x and predict the average value for y.

First, to reduce the number of missing data, instead of using each star ratings date, we use the monthly average of star ratings. However, the problem is not completely resolved, and from time to time we observe missing data. In time series data, if there are missing values, there are two ways to deal with the incomplete data: ignoring the missing value or replacing the missing data.

Taking in consideration that star ratings data is rather normally distributed data, and the fact that these missing monthly average values are rare, we decide to treate missing values by replacing each of them with the mean value of the star ratings of all reviews in that food category for each city.

Let us start by checking the accuracy of this method with the following scenario. We suppose that Las Vegas is selected by a business owner as the base city. The cities of Phoenix and Madison are identified by the system as its peer cities. In addition, the producer asks the system to suggest him the most successful food categories. As filtering criteria, he chooses the food categories with the highest number of reviews in these three cities.

The system finds 17 categories of the most highly reviewed restaurant. The data for the base city and all its peers are stored in a table. Table 7-13 depicts the list of these categories in alphabetical order for two cities of Las Vegas and Phoenix.

Table 7-13: Restaurant categories with highest review counts

	Las Vegas (NV)				Phoenix (AZ)			
	M	T	Training set 80%	Test set 20%	M	T	Training set 80%	Test set 20%
American New	3.71	40599	33008 81.30%	7591 18.70%	3.82	48848	39494 80.85%	9354 19.15%
American Traditional	3.52	42151	34685 82.29%	7466 17.71%	3.53	38848	31650 81.47%	7198 18.53%
Asian Fusion	3.75	10597	8332 78.63%	2265 21.37%	3.67	8461	6733 79.58%	1728 20.42%
Breakfast Brunch	3.87	28375	22558 79.50%	5817 20.50%	3.81	29974	23742 79.21%	6232 20.79%
Buffets	3.41	25175	21700 86.20%	3475 13.80%	3.52	7081	6072 85.75%	1009 14.25%
Burgers	3.69	21521	17077 79.35%	4444 20.65%	3.65	19844	15977 80.51%	3867 19.49%
Chinese	3.46	19907	16320 81.98%	3587 18.02%	3.58	16944	13945 82.30%	2999 17.70%
Fast Food	3.47	11007	8983 81.61%	2024 18.39%	3.46	9782	7706 78.78%	2076 21.22%
Italian	3.69	24504	19872 81.10%	4632 18.90%	3.87	33604	27793 82.71%	5811 17.29%
Japanese	3.86	25800	20619 79.92%	5181 20.08%	3.68	12537	10189 81.27%	2348 18.73%
Mexican	3.60	24775	18920 76.37%	5855 23.63%	3.64	43455	35103 80.78%	8352 19.22%
Pizza	3.75	19341	14871 76.89%	4470 23.11%	3.81	35700	29006 81.25%	6694 18.75%
Sandwiches	3.84	17127	13515 78.91%	3612 21.09%	3.87	29586	24573 83.06%	5013 16.94%
Seafood	3.81	19099	15747 82.45%	3352 17.55%	3.79	11819	9458 80.02%	2361 19.98%
Steakhouses	3.93	31329	26155 83.48%	5174 16.52%	3.70	12900	10594 82.12%	2306 17.88%
Sushi Bars	3.82	21094	16851 79.89%	4243 20.11%	3.68	16140	13420 83.15%	2720 16.85%
Thai	3.86	11019	8931 81.05%	2088 18.95%	3.87	8411	6871 81.69%	1540 18.31%

- M: Mean of star ratings of the category for the city which is used to replace missing data
- T: The total number of reviews of the category for the city
- “Training set” is all reviews dated before 2014 and the “Test set” is all reviews of 2014

We filter reviews by category and city, resulting 51 distinct data groups. For example, all 35,700 reviews for pizza in Phoenix are grouped under the same table. For each group we calculate *M* the mean of star ratings. Then each *M* is used to replace the missing monthly average of star ratings values within its group.

For each data group, the data is partitioned into a training set and a test set. We separate the data to two sets according to reviews date. All reviews from January 2004 to December 2013 (120 months) are used as training set and all reviews from January 2014 to July 2014 (7 months) are used as test set. Depending to the date of reviews in each group, the poursentage of training set and test set vary slightly from case to case. Approximately 80% of the data is used for training and the remaining 20% for testing.

Among others, the training data of each group is used to calculate its monthly average star ratings values. If needed, the missing data in each group is replaced by the value of its respective *M*. So, for each group we obtain 120 values, that to say one per month. These values are used to create a scatter plot per group. Scatter plots are used to identify relationships between monthly average star ratings values and the date variables. Regression lines, or best fit lines, are a type of annotation on scatterplots that show the overall trend of a set of data.

Linear regression is a statistical method for modeling the relationship between two variables. The sign of a regression coefficient tells you whether there is a positive or negative correlation between each independent variable and the dependent variable. A positive (+) coefficient indicates that as the value of the independent variable increases, the mean of the dependent variable also tends to increase. A negative (-) coefficient suggests that as the independent variable increases, the dependent variable tends to decrease.

Finally, for each group we forecast the monthly average star ratings of next seven months (from January to July 2014). We compare the forecast values with their real values. The *MAE* is calculated to measure the forecast error in each case. Table 7-14 depicts a comparison of the *MAE* values and shows the distribution of prediction accuracy for all 51 groups.

As we can see clearly in Table 7-14, the best performance (with the smallest *MAE* of 0.038) is obtained for the “Sandwiches” food category in Phoenix and the worst result (with the biggest *MAE* of 0.662) is obtained for the “Fast Food” food category in Madison.

Table 7-14: Comparison of the *MAE* values for 17 food categories in three cities

City / **Food category**	Las Vegas (NV)	Phoenix (AZ)	Madison (WI)
	Mean absolute error (MAE)		
American New	0.086	0.048	0.172
American Traditional	0.151	0.047	0.146
Asian Fusion	0.084	0.099	0.323
Breakfast & Brunch	0.156	0.079	0.186
Buffets	0.044	0.074	0.656
Burgers	0.061	0.113	0.137
Chinese	0.066	0.080	0.213
Fast Food	0.240	0.107	**0.662**
Italian	0.065	0.130	0.339
Japanese	0.159	0.132	0.299
Mexican	0.066	0.108	0.298
Pizza	0.055	0.093	0.228
Sandwiches	0.043	**0.038**	0.107
Seafood	0.111	0.139	0.368
Steakhouses	0.099	0.121	0.287
Sushi Bars	0.150	0.145	0.308
Thai	0.084	0.195	0.241

These *MAE* values indicate that our most accurate forecast is for the “Sandwiches” food category in Phoenix. The combination of this information with the relatively high predicted star ratings (depicted in Table 7-15), allows us to recommend them to the business owner. Of course, the final interpretation of these results is done by producer himself.

Table 7-15: Real values versus predicted values of monthly average star ratings for “Sandwiches” food category in Phoenix

2014	Real values	Predicted values
January	3.87	3.84
February	3.88	3.84
March	3.88	3.84
April	3.76	3.83
May	3.81	3.83
June	3.78	3.83
July	3.83	3.83
***MAE* = 0.038**		

The results of the two scenarios, with maximum accuracy and minimum accuracy, are described in more detail below.

7.5.5.1 An example of situation that the prediction has the minimum MAE

In this example, we analyse the "Sandwiches" food category in Phoenix. In total we have 29,586 reviews in this group. We use 24,537 (83%) of reviews as training set. As Figure 7-21 depicts we find a slightly negative slop for the training set.

Regression Statistics	
Multiple R	0.179
R Square	0.032
Adjusted R Square	0.024
Standard Error	0.291
Observations	120
y-intercept	4.0219
Regression coefficient	-0.0015

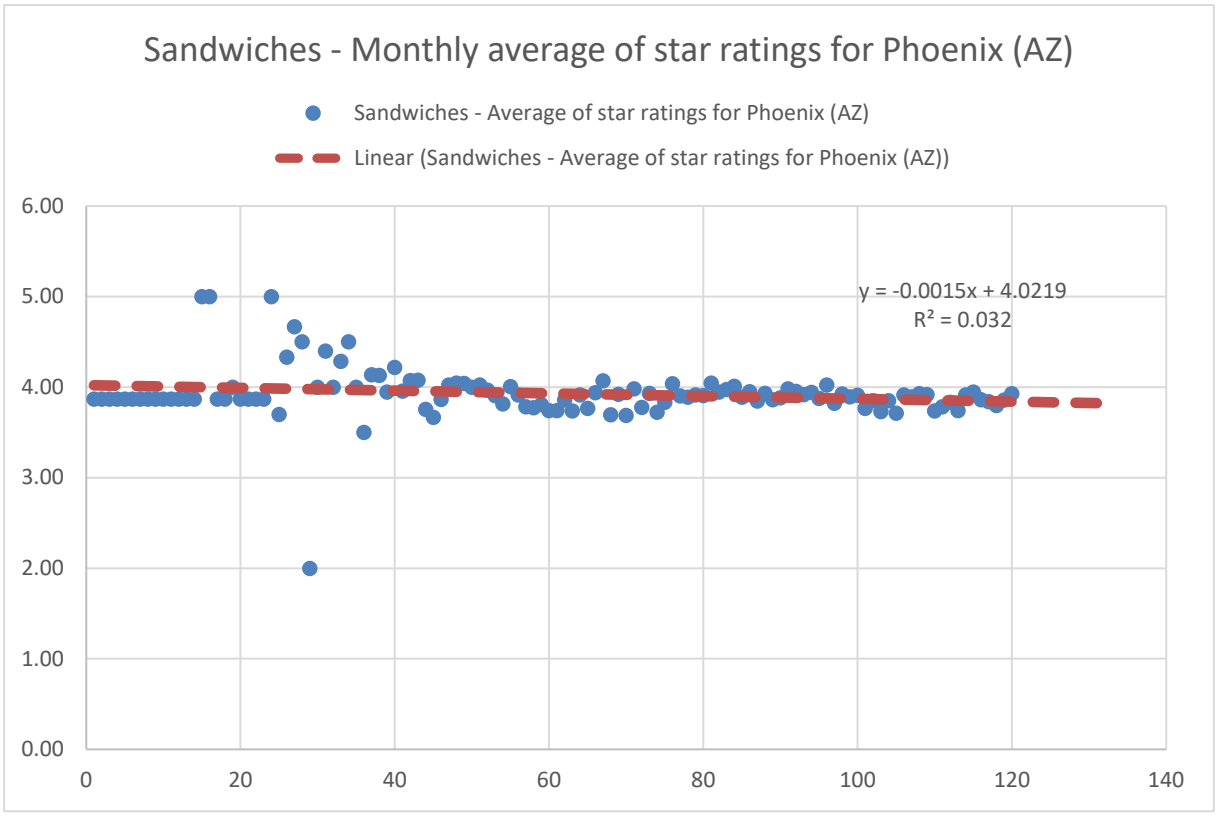


Figure 7-21: Regression Statistics 83% of data for "Sandwiches" food category, Phoenix

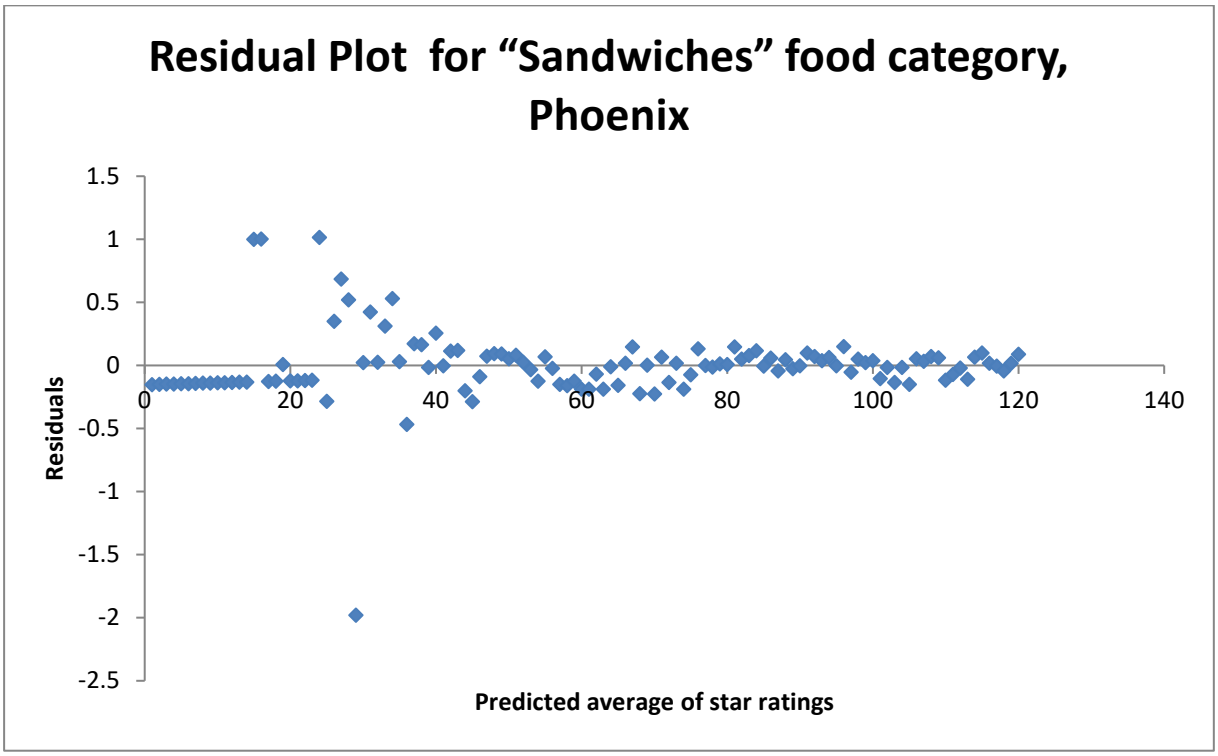


Figure 7-22: Residual Plot for "Sandwiches" food category, Phoenix

In the plot (Figure 7-22), each point represents the average star ratings value for one month, where the prediction made by the model is on the x-axis and the accuracy of the prediction is on the y-axis. The distance from the line at 0 is how bad the prediction was for that value. Positive values for the residual (on the y-axis) mean the prediction was too low, and negative values mean the prediction was too high; 0 means the guess was correct.

7.5.5.2 An example of situation that the prediction has the maximum MAE

In this second example, we analyse the "Fast Food" food category in Madison. Here, in this group we have in total only 330 reviews. For training set we use 254 (77%) of reviews. As Table 7-16 depicts the predicted values are rather far from the real values.

Table 7-16: Real values versus predicted values of monthly average star ratings for "Fast Food" food category in Madison

2014	Real values	Predicted values
January	3.86	3.01
February	4.13	3.00
March	3.58	3.00
April	3.73	2.99

May	3.08	2.99
June	3.25	2.99
July	2.00	2.98
***MAE* = 0.662**		

Figure 7-23 depicts that we find a negative slop for the first 77% of data. The random pattern of the predicted values in Figure 7-24 indicates that our linear model provides a decent fit to the data. However, as we can see in this case the accuracy of our prediction is very low.

Regression Statistics	
Multiple R	0.1701
R Square	0.0289
Adjusted R Square	0.0207
Standard Error	0.7981
Observations	120
y-intercept	3.4828
Regression coefficient	-0.0039

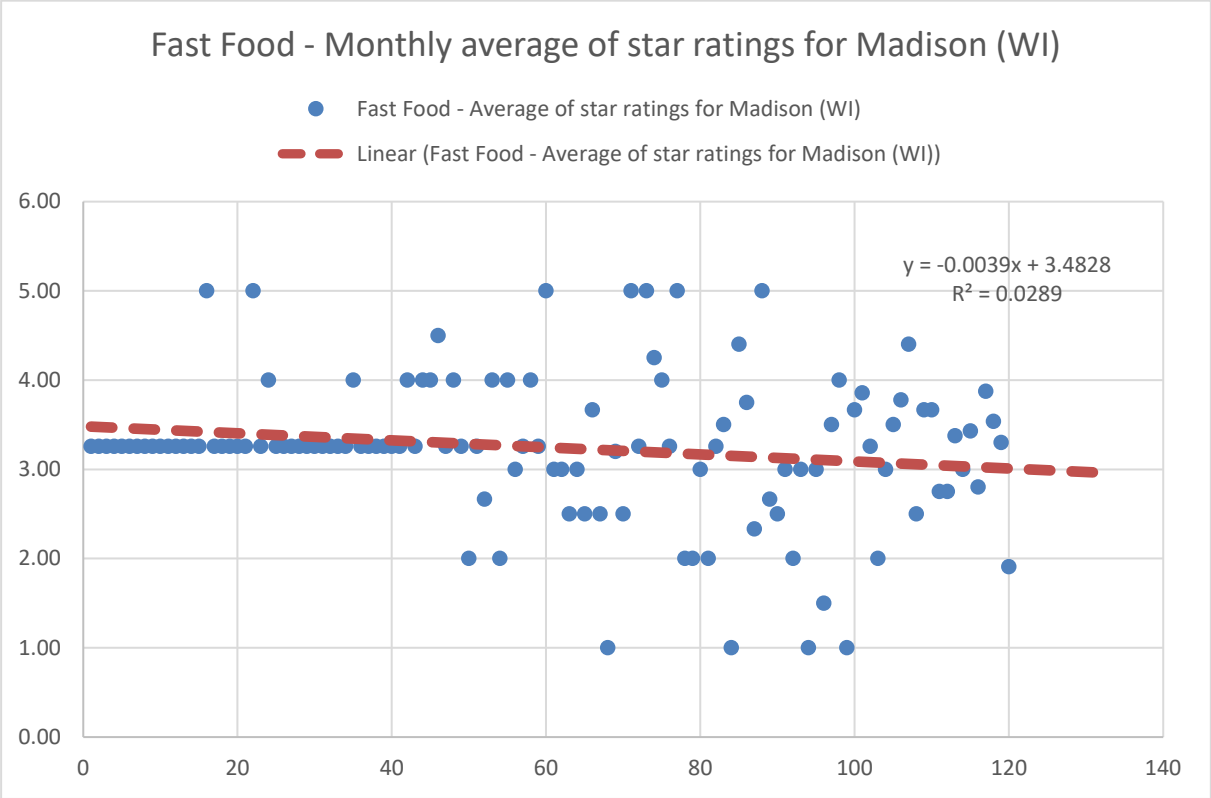


Figure 7-23: Regression Statistics 77% of data for “Fast Food” food category, Madison

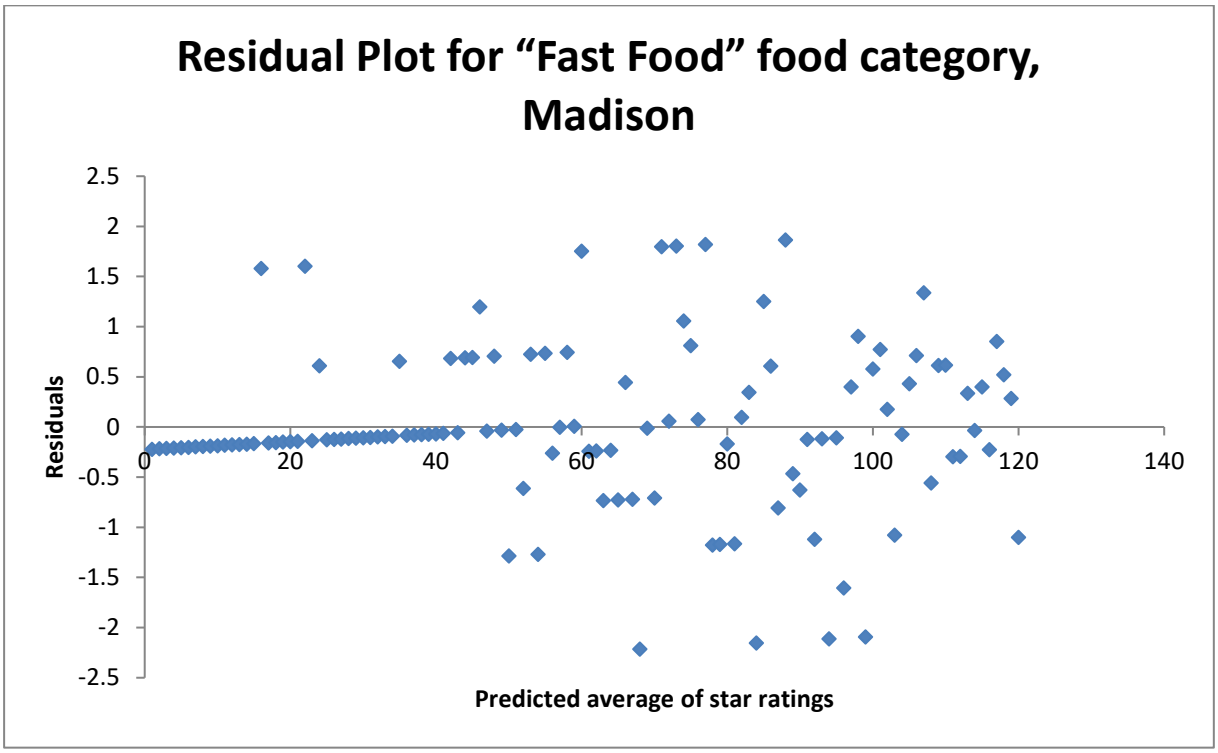


Figure 7-24: Residual Plot for “Fast Food” food category, Madison

Finaly, the linear regression can be used to compare overtime values of star ratings versus predicted satr ratings. As an example, the Figure 7-25 depicts this comparison for “Sushi Bars” food category in Madison.

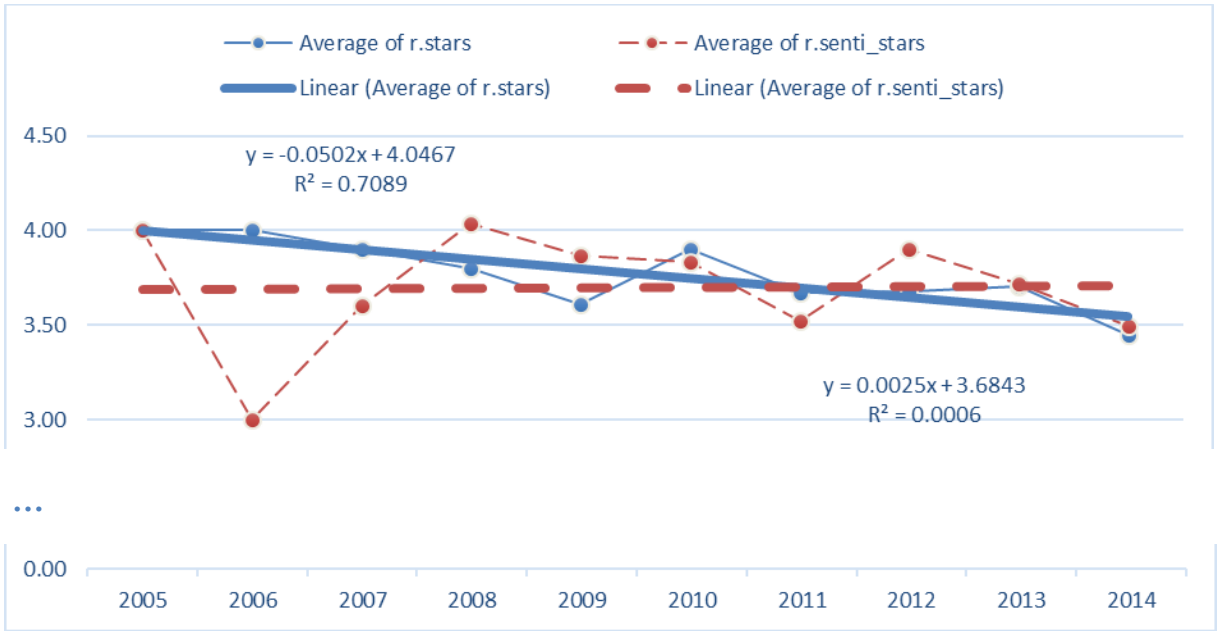


Figure 7-25: Average of star ratings versus the average of predicted star ratings for “Sushi Bars” in Madison (WI)

7.6 Summary

In chapter seven we develope pragmatic approaches to discover where and when a new restaurant can be launched. We use concrete examples to demonstrate the ability of recommender system for selecting food concepts, analyzing the competition, and identifying the perfect location to enable an entrepreneur to make correct decisions when starting a food business.

Chapter eight is the conclusion of my work. It will clearly state the answer to the main research question and will review the key points of the thesis.

Chapter 8 Conclusion

We presented the state of the art for recommender systems, social networking analysis, opinion detection, information retrieval, text categorization, and polarity evaluation. Then we proposed a taxonomy of current approaches for each.

Recommender systems made a significant progress over the last decade when numerous contents based, collaborative and hybrid methods were proposed and several “industrial-strength” systems have been developed.

The Web 2.0 is now a major medium of communication in our society and therefore, an element of our socialization. With the explosive growth of social media on the Web, businesses are increasingly relying on opinion mining methods to analyze the content of these media (e.g., Yelp reviews and star ratings) to achieve most effective ways to make thoughtful, and informed decisions. Forecasting the success or failure of a new restaurant in a given city or region is one of the domains focused on in this thesis.

This thesis includes a detailed study of different approaches for feature extraction and feature opinion classification, their strength and limitations and latest research challenges in feature-based opinion mining. Our research is different from most existing work, as we did not only consider user’s previous star ratings, but also the content and subjectivity of their previous review text.

In addition, we focused on the computational analysis of the reviews’ textual content of sentiment analysis combined with social network data to develop a collaborative and content-based recommendation system for Yelp and to investigate the recommendation systems in a significantly different point of view. We analyzed and provided pertinent recommendations for the food producers (restaurant owners), rather than targeting and suggesting products or services to their customers.

Our approach was to mine the text messages of reviewers (customers and non-customers) of a restaurant to find an indication of how happy (e.g., a polarity score was predicted based on the opinion mining of the review's text) they are with their dining experiences and what restaurant features they may be interested in.

We applied natural language processing techniques and handled the problem from an information extraction perspective and hence settled on a rule-based restaurant features extraction approach. Rules were extraction patterns from first the syntactic structure of reviews and second from ontologies describing classes, attributes, and relations between concepts.

We designed an analytic system able to extract restaurant features and opinions from a set of customer reviews. Then we implemented a distance-based model that can effectively determine the distance (or the similarity) between products using the products' descriptions and customers' opinions. We evaluated the products descriptions generated by the system by comparing them with the description generated manually.

The opinion and polarity detections were evaluated using the Yelp test collection reviews. Our first task was to preprocess the data. More precisely, we used as target application the restaurants and we preprocessed the reviews to remove very short ones (e.g., a review limited to "bad restaurant") or those not written in English.

We also identified and chose the reviewers (e.g., removing persons without or with a single review) and selected reviews corresponding to those selected persons, restaurants, and inside the defined time span and geographical regions (e.g., we ignored restaurants with no or a single review).

As an important step in the data mining process, we checked for spelling, and grammatical errors and we corrected them. We found that the text cleaning increased by 45.14 % the number of known words by the system and therefore we illustrated the importance of data cleaning.

Well-known processes in information retrieval were applied such as removal of some stopwords, punctuations, and stemming. Moreover, some filters were applied such as ignoring words having a low or very high occurrence frequency. More advanced features (e.g., using

PoS information and giving more importance to adjectives and adverbs) were investigated every time that it was suitable.

We used the SentiWordNet 3.0 thesauri to assign the polarity to each word. Of course, we enriched the representation with a few rules related to the negation, the presence of some adverbs (e.g., very) or connectives (e.g., and, but).

During the opinion polarity classification process on Yelp restaurant reviews corpus, we retrieved a list of keywords, as well as all their bigrams and trigrams which are “mostly used” to express an opinion or sentiment. However, we found that, there is no significant difference in performance when using a combination of bigrams, unigrams and trigrams instead of only unigrams model.

Our next computational step was the development of an algorithm that classifies each sentence according to its textual content, so that we ended up with separate databases of sentences about food quality, service, ambience, accessibility, and other features described previously in the constructed lexical database.

We focused on the evaluation of the restaurants frame (22 categories) extraction algorithm with manually labeled test corpus. We demonstrated how reviews for restaurants can be automatically classified into relevant categories. An automatic term classification of reviews’ keywords was used to evaluate our methodology found to perform a recall of 52% and a precision of 67% over a manually labeled excerpt.

As a classification tool, we adopted NLTK classifier, which supports classification, tokenization, stemming, tagging, parsing, and semantic reasoning functionalities. It is developed by Steven Bird and Edward Loper in the Department of Computer and Information Science at the University of Pennsylvania.

The study explored the temporal interpolation of the social influence in a given community regarding similar restaurants commented on during a previous time period. We considered the impact of the social networks in the prediction of the analytic system. Using the Yelp data, a social network was built based on the user profile of the reviewers. In addition to the reviewer network and the geographical proximity (or profile) network, we constructed the friendship network from the Yelp data.

The aim was to associate these networks in order to define an *affinity network* from which we determine customers' communities and influential customers taking into account the temporal dimension. We studied this social network connecting people posting reviews on the same restaurants. The magnitude of the relationship between two persons settled as a function of the number (or average number) of reviews they have in common for the same business. Moreover, the orientation was defined according to the polarity of those common reviews.

As one of our objectives, the influential customers in a given community were determined by adapting existing algorithms in the social influence state of the art, for instance graph centrality measure algorithms. Results were supposed to be used to weight customers' reports and hence improve the prediction of the analytic system.

We performed graph data analysis and visualization to find out how the interested users are connected and derived conclusions about the relationships between them, e.g., are they connected in many small groups or in a few large groups, are they in isolated peripheral groups or in central well connected parts of the graph, are they dispersed through the network or clustered together.

Next, we were interested to identify influencers and finding their degree of leadership, in other words to identify which nodes should receive a recommendation of a product, in order to maximize the impact. However, due to lack of information about Yelp customers friendship links, this investigation did not result in the expected results.

We briefly investigated into fake reviews and fake Yelp accounts. We found that, without a case-by-case analysis, our results are not reliable to detect fake reviews, and so we cannot answer the question of "What percentage of Yelp reviews (or users) are fake?" without doubt.

Each review was parsed to extract the passages matching a restaurant facet. The feature weighting scheme was limited to binary (presence / absence) because the customer reviews are usually rather short. For each detected facet, the polarity was detected (which can be nil if the text excerpt is simply a description). To achieve this, the features able to predict the polarity of a short text were defined. Pertinent facets (or restaurant features) were defined such as type (Thai, burger, Italian, …), service, food, price/value, atmosphere, space for children, etc. For each facet, we identified semi-automatically the strings corresponding to them in the reviews. Having defined the polarity of different facets, the system is able to determine the

overall polarity of the entire review. As a result, if we tolerate one-star estimation error, then we calculate the *Accuracy rate =75.70%*, and observe that for 534,758 reviews, the predicted star ratings are fairly accurate.

Unlike other studies in this domain, we took account of the temporal information (to be able to generate an evolution of a given food category or a set of related categories in a given region). Then we designed and partially implemented a similarity (or distance) measure to define the similarities between cities. Such a measure was based on four key themes and their variables. Of course, each variable has its own importance and thus having different weights. As a second important aspect, we considered the customer's reviews. The underlying problem was to define the most appropriate mix between these two components in a distance measure.

In order to identify peer cities, we highlighted some statistical datasets available, for different parts of the world (e.g., United States, Europe countries and regions, Asia, and Africa). Based on this measure, we then clustered (the geographical information was taken into account in this clustering process) the different cities to define peer cities.

We built an analytic system using the software components able to generate an overview of the evolution of the different food categories, for a given region and time span. Moreover, when comparing two distinct regions, the system provides a view of their differences and thus were used to suggest market opportunities.

A deeper analytic system was designed, able to answer questions such as "Where are my best competitors?", "In which restaurant features, my competitors encounter the most often difficulties?".

Finally, we developed an analytic system able to discover where and when a new product or service can be launched according to customers' reports and expectations. It was clearly demonstrated (through concrete examples) that, our recommender system can be used for selecting food concepts, analyzing the competition, and identifying the perfect location (city) to enable a business owner to make correct decisions when opening a new restaurant. We also showed how the results of this study can be incorporated (e.g., in form of an analytical dashboard) into Yelp's existing website.

References

Abassi, A., Chen, H. & Salem, A. (2008). *Sentiment Analysis in Multiple Languages: Feature Selection for Opinion Classification in Web Forums.* ACM-Transactions on Information Systems, vol. 26, nº 3.

Aggrawal, C.C., & Zhai, C.X. (2012). *A Survey of Text Clustering Algorithms*. In C.C. Aggrawal, & C.X. Zhai (Eds), Mining Text Data, Berlin: Springer-Verlag.

Aggarwal, C.C., Zhao, Y., & Yu, P. S. (2012). *On Text Clustering with Side Information.* In proceedings of the International Conference on Data Engineering ICDE.

Amini, M.-R. & Gaussier, E. (2013). *Recherche d'information. Applications, Modèles et Algorithmes, Fouille de données décisionnel et Big Data.* Paris, Eyrolles.

Amorim, R.C. de (2015). *Feature Relevance in Ward's Hierarchical Clustering Using the Lp Norm*. Journal of Classification. 32 (1): 46–62. doi:10.1007/s00357-015-9167-1.

Anchiêta, R.T. & al. (2015). *Using Stylometric Features for Sentiment Classification*. 16th International Conference, CICLing 2015, Cairo. 194.

Angelova, R., Siersdorfer, S. (2006). *A neighborhood-based approach for clustering of linked document collections.* CIKM Conference.

Aral, S. and Walker, D. (2012). *Identifying Influential and Susceptible members of Social Networks.* In Science.

Barbieri, N., Manco, G., & Ritacco, E. (2014). *Probabilistic Approaches to Recommendations.* Synthesis Lectures on data Mining and Knowledge Discovery. St Rafael (CA), Morgan & Claypool Publishers.

Bayoudhi, A., Ghorbel, H., Hadrich Belguith, L. (2014). *Focus Definition and Extraction of Opinion Attitude Questions.* 9th International Conference on Application of Natural Language to Information Systems (NLDB 2014): 224-227A. Montpellier, France.

Bing Liu (2010). *Sentiment Analysis and Subjectivity.* Invited Chapter for the Handbook of Natural Language Processing, Second Edition.

Boiy, E., & Moens, M.F. (2009). *A Machine Learning Approach to Sentiment Analysis in the Multilingual Web texts.* Information Retrieval, 12(5), 526-558.

Borgatti, Stephen P., Everett, Martin G., and Johnson, Jeffrey C. (2013). *Analyzing Social Networks.* Thousand Oaks, CA: Sage Publishing.

Carrington, P.J., Scott, J., & Wasserman, S. (Eds) (2005). *Models and Methods in Social Network Analysis.* Cambridge (UK), Cambridge University Press.

Chávez, E., Navarro, G., Baeza-Yates, R., & Marroquin, J. (2001). *Searching in Metric Space.* ACM Computing Surveys, 33(3), 273-321.

Cormack, R. M. (1971). *A Review of Classification*. Journal of the Royal Statistical Society, Series A, 134(3), 321-367.

Dandekara, Pranav, Goelb, Ashish, and Leec, David T. (2013). *Biased assimilation, homophily, and the dynamics of polarization.* In Proceedings of the National Academy of the United States of America (PNAS), Vol 110 no 15.

Dodds, P. and Watts, D. (2007). *Universal behavior in a generalized model of contagion.* In Phys Rev Lett, 92(21): 21870.

Dolamic, L., & Savoy J. (2010). *Comparative Study of Indexing and Search Strategies for the Hindi, Marathi and Bengali Languages.* ACM – Transactions on Asian Language Information Processing, 9(3),

Easley, D., & Kleinberg, J. (2010). *Networks, Crowds, and Markets. Reasoning about a Highly Connected World.* Cambridge (UK), Cambridge University Press.

Esuli, A. & Sebastiani, F. (2006). *SentiWordNet: A Publicly Available Lexical Resource for Opinion Mining.* In Proceedings *LREC-06*, Lisbon, 417-422.

Fautsch, C., Savoy, J. (2009). *UniNE at TREC-2008: Fact and Opinion Retrieval in the Blogosphere.* Proceedings *TREC-2008*, NIST Publication #500-277.

Fautsch, C., & Savoy, J. (2010). Adapting the *tf idf* Vector-Space Model to Domain-Specific Information Retrieval. *Proceedings ACM-SAC*, 1708-1712.

Forman, G. (2003). *An Extensive Empirical Study of Feature Selection Metrics for Text Classification.* Journal of Machine Learning, 3(3):1289-1305.

Ghorbel, H. and Jacot, D. (2011). *Sentiment analysis of French movie reviews.* Studies in Computational Intelligence - Advances in Distributed Agent-Based Retrieval Tools 361, 97–108.

Ghorbel, H. (2012). *Cross-lingual Sentiment Analysis in Online Discussion Forums SocInfo.* The 4th International Conference on Social Informatics, Lausanne, Switzerland.

Goldberg, J., Libai, B. and Mulle, E. (2001). *Talk of network: A complex systems look at the underlying process of word-of-mouth.* In Marketing Letters, pages, 221-223.

Gordon, A. D. (1999). *Classification, 2nd Edition*. Chapman and Hall, Boca Raton.

Milligan, G. W. (1979). *Ultrametric Hierarchical Clustering Algorithms*. Psychometrika, 44(3), 343–346.

Granovetter, M. (1978). *Threshold Models of Collective Behavior.* In American Journal of Sociology, (83):1420-1443.

Hand, D.J. (2006). *Classifier Technology and the Illusion of Progress.* Statistical Science, 21(1), 1-14.

Hastie, T., Tibshirani, R., & Friedman, J. (2009). *The Elements of Statistical Learning*. Springer, New York.

Hu, M., & Liu, B. (2004). *Mining Opinions Features in Customer Reviews.* Processing AAAI, 755-760.

James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). *An Introduction to Statistical Learning. With Applications in R.* Springer, New York.

Jannach, Dietmar, Zanker, Markus, Felfernig, Alexander, and Friedrich, Gerhard (2010). *Recommender Systems: An Introduction (1st ed.).* Cambridge University Press, New York, NY, USA.

Jannach, D., Zanker, M., Felfernig, A., & Friedrich, G. (2011). *Recommender Systems. An Introduction.* Cambridge (UK), Cambridge University Press.

Jijkoun, V., de Rijke, M., & Weerkamp, W. (2010). *Generating Focused Topic-specific Sentiment Lexicons.* Proceedings of the 48th international Meeting of the Association for Computational Linguistics, 585-594.

Joachims, T. (2002). *Learning to Classify Text Using Support Vector Machines. Methods, Theory and Algorithms*. London: Kluwer.

Kaplan, A. M. and Haenlein, M. (2011). *Two hearts in three-quarter time: How to waltz the social media/viral marketing dance.* Business Horizons, 54(3), 253-263.

Kaufman, L., & Rousseeuw, P.J. (1990). *Finding Groups in Data: An Introduction to Cluster Analysis*. Hoboken: Wiley Interscience.

Kempe, David, Kleinberg, Jon, Tardos, Eva (2003). *Maximizing the Spread of Influence through a Social Network.* In KDD, pages 137-146.

Kiss, Tibor, & Strunk, Jan (2006). *Unsupervised Multilingual Sentence Boundary Detection*. Computational Linguistics 32: 485-525.

Liu, H., & Motoda, H. (ed.). (2008). *Computational Methods of Feature Selection*. Boca Raton: Chapman & Hall / CRC.

Liu, B. (2012). *Sentiment Analysis and Opinion Mining*. Boca Raton: Morgan & Claypool Publishers.

Loria, S. (2014). *TextBlob: simplified text processing. Secondary TextBlob: Simplified Text Processing.* http://textblob.readthedocs.io .

Mani, I., & Maybury, M.T. (1999). *Advances in Automatic Text Summarization*. Cambridge (MA): The MIT Press.

Manning, C.D., & Schütze, H. (2000). *Foundations of Statistical Natural Language Processing*. Cambridge (MA): The MIT Press.

Manning, C., Raghavan, P., Schütze, H. (2008). *Introduction to Information Retrieval*. Cambridge (UK), Cambridge University Press.

Macdonald, C., Ounis, I., Soboroff, I. (2008). *Overview of the TREC-2007 blog track.* Proceedings TREC-2007, NIST Publication #500-274, 1-13.

Macdonald, C., Ounis, I., Soboroff, I. (2010). *Overview of the TREC-2009 blog track.* Proceedings TREC-2009, NIST Publication #500-278, 1-13.

Mouine, M. & Lapalme, G. (2012). *Using Clustering for Personalize Visualization.* Proceedings IEEE Conf. on Information Visualization, 258-263.

Newman, M. E. J. (2012). *Communities, modules and large-scale structure in networks.* Nature Physics 8, 25-31.

Nugues, P.M. (2014). *An Introduction to Language Processing with PERL and Prolog*. 2nd Ed., Springer-Verlag, Berlin.

Ounis, I., Macdonald, C., Soboroff, I. (2009). *Overview of the TREC-2008 blog track.* Proceedings TREC-2008, NIST Publication #500-277, 1-11.

Pan, R., Zhou, Y., Cao, B., Liu, N., Lukose R., Scholz M., and Yang Q. (2008). *One class collaborative filtering.* IEEE International Conference on Data Mining (ICDM).

Pang, B., Lee, L., & Vaithyanathan, S. (2002). *Thumbs up? Sentiment Classification using Machine Learning Techniques.* Proceedings *EMNLP*, 79-86.

Pang, B., & Lee, L. (2008). *Opinion Mining and Sentiment Analysis.* Foundations and Trends in Information Retrieval, 2(1-2).

Paterek, A. (2007). *Improving regularized singular values decomposition for collaborative filtering.* In Knowledge Discovery and Data Mining (KDD) cup and workshop.

Pazzani, Michael J., Billsus, Daniel (2007). *Content-Based Recommendation Systems.* The Adaptive Web: 325-341.

Pazzani, Michael J. (1999). *A Framework for Collaborative, Content-Based and Demographic Filtering.* Artif. Intell. Rev. 13(5-6): 393-408.

Ricci, F., Rokach, L., Shapira, B., Kantor, P.B. eds. (2011). *Recommender Systems Handbook.* Springer.

Savoy, J. (2005). *Comparative Study of Monolingual and Multilingual Search Models for Use with Asian Languages.* ACM Transactions on Asian Languages Information Processing, 4(2), 163-189.

Savoy, J. (2010). *Lexical Analysis of US Political Speeches.* Journal of Quantitative Linguistics, 17(2), 123-141.

Savoy, J. (2012). *Authorship Attribution Based on Specific Vocabulary.* ACM – Transactions on Information Systems, 30(2), Article #12.

Savoy, J. (2015a). *Comparative Evaluation of Term Selection Functions for Authorship Attribution.* Digital Scholarship in the Humanities (previously: Literary & Linguistic Computing), to appear.

Savoy, J. (2015b). *Text Clustering: An Application with the State of the Union Addresses.* Journal of the American Society for Information Science and Technology, to appear.

Sebastiani, F. (2002). *Machine Learning in Automatic Text Categorization.* ACM Computing Survey, 14(1), 1-27.

Seki, Y., Evans, D.K., Ku, L.W., Chen, H.H., Kando, N., Lin, C.Y. (2007). *Overview of opinion analysis pilot task at NTCIR-6.* Proceedings *NTCIR-6,* National Institute of Informatics, 265-278.

Snyder, B., & Barzilay, R. (2007). *Multiple Aspect Ranking using the Good Grief algotihm.* Proceedings HLT-NAACL, 300-307.

Sokolova, M. & Lapalme, G. (2011). *Learning Opinion in User-Generated Web Content*. Natural Language Engineering, 17(4), 541-567.

Stone, P.J. (1966). *The General Inquirer: A Computer Approach to Content Analysis.* The MIT Press, Cambridge (MA).

Tausczik, Y.R., & Pennebaker, J.W. (2010). *The Psychological Meaning of Words: LIWC and Computerized Text Analysis Methods.* Journal of Language and Social Psychology, 29(1), 24-54.

Ward, J. H., Jr. (1963). *Hierarchical Grouping to Optimize an Objective Function*. Journal of the American Statistical Association, 58, 236–244.

Wasserman, S., & Faust, K. (1994). *Social Network Analysis.* Methods and Applications. Cambridge (UK), Cambridge University Press.

Wei, C.P., Chen, Y.M., Yang, C.S. and Yang, C.C (2010). *Understanding what concerns consumers: a semantic approach to product feature extraction from consumer reviews.* Information Systems and E-Business Management pp. 149-167.

Wilson, T., Hoffmann, P., Somasundaran, S., Kessler, J., Wiebe, J., Choi, Y., Cardie, C., Riloff, E., Patwardhan, S. (2005). *OpinionFinder: A System for Subjectivity Analysis.* Proceedings HLT/EMNLP, Vancouver (BC), 34-35.

X. Ding, B. Liu and P. S. Yu (2008). *A holistic lexicon-based approach to opinion mining.* In Proceedings of the Conference on Web Search and Web Data Mining (WSDM).

Zhang, Hu, X., Zhou, X. (2008). *A comparative evaluation of different link types on enhancing document clustering.* ACM SIGIR Conference.

Zubaryeva, O., & Savoy, J. (2008). *Opinion and Polarity Detection within Far-East Languages in NTCIR-7.* In Proceedings NTCIR-7, NII publication (National Institute of Informatics), Tokyo, 318-323.

Zubaryeva, O., & Savoy, J. (2010). *Opinion Detection by Combining Machine Learning & Linguistic Models.* In Proceedings *NTCIR-8*, NII publication (National Institute of Informatics), Tokyo, 221-227.

Curriculum Vitae

Expériences

Caritas Jura	septembre 2018 – today
Chargé de projet en informatique, Delémont	
Archives cantonales jurassiennes (ArCJ)	juillet 2017 – février 2018
Ingénieur Informatique Senior, Porrentruy	
Fondation Mossadegh - Bibliothèque d'Iranologie	juin 2016 – décembre 2016
Assistant de Direction, Genève	
Haute Ecole Arc Ingénierie	octobre 2015 – mars 2016
Développeur Hadoop et ingénieur de recherche scientifique, St-Imier	
INDC Sàrl	juillet 2015 – août 2015
Gestionnaire du projet - Développeur PHP / JQuery, Delémont	
Services industriels de Delémont	août 2012 – décembre 2012
Gestionnaire du projet, Delémont	
ICT Generics SA	juillet 2009 – juin 2012
Fondateur de startup et Manager, Le Noirmont	
Fonds mondial de solidarité numérique (FSN)	août 2005 – mai 2009
Directeur informatique et Ingénieur system, Genève	
L'Union internationale des télécommunications (UIT)	octobre 2001 – février 2004
Responsable Informatique, Genève	
Université de Genève	septembre 1995 – septembre 2001

Assistant d’enseignement et de recherche au Centre Universitaire d'Informatique, Genève

Formation

2014–en cours	Doctorant en Informatique (Dual Recommendation Analysis)	Université de Neuchâtel
1997-2003	Doctorant en System d’Information (Enseignement à distance et E-learning en utilisant la technologie de la réalité virtuelle)	Université de Genève, Sans Diplôme
1995-1996	Diplôme d’études Supérieures en System d’Information (Visualisation et Communication Infographiques)	Université de Genève
1994-1995	Diplôme postgrade en Informatique (Infographiques)	École Polytechnique Fédérale de Lausanne (EPFL)
1988-1993	Diplôme d’ingénieur en Électronique et Télécommunication (Option satellite)	Université technique d'Istanbul (İTÜ)
1987-1988	Langue et littérature anglaises et les études de théologie	Université du Moyen-Orient de Beyrouth (MEU)
1986-1987	Langue et littérature turques (Faculté des langues, d'histoire et de géographie)	Université d'Istanbul (İÜ)

Langues

Français	Lu, écrit, parlé
Anglais	Lu, écrit, parlé
Allemand	Débutant
Persan	Lu, écrit, parlé (maternel)
Turc	Lu, écrit, parlé
Hébreu ancien, Japonais, Russe et Arabe	Notion

Appendix A

Yelp Dataset JSON

yelp_academic_dataset_business.json

```
{
 "business_id":"encrypted business id",
 "name":"business name",
 "neighborhood":"hood name",
 "address":"full address",
 "city":"city",
 "state":"state -- if applicable --",
 "postal code":"postal code",
 "latitude":latitude,
 "longitude":longitude,
 "stars":star rating, rounded to half-stars,
 "review_count":number of reviews,
 "is_open":0/1 (closed/open),
 "attributes":["an array of strings: each array element is an attribute"],
 "categories":["an array of strings of business categories"],
 "hours":["an array of strings of business hours"],
 "type": "business"
}
```

yelp_academic_dataset_review.json

```
{
```

```
 "review_id":"encrypted review id",
 "user_id":"encrypted user id",
 "business_id":"encrypted business id",
 "stars":star rating, rounded to half-stars,
 "date":"date formatted like 2009-12-19",
 "text":"review text",
 "useful":number of useful votes received,
 "funny":number of funny votes received,
 "cool": number of cool review votes received,
 "type": "review"
}
```

yelp_academic_dataset_user.json

```
{
 "user_id":"encrypted user id",
 "name":"first name",
 "review_count":number of reviews,
 "yelping_since": date formatted like "2009-12-19",
 "friends":["an array of encrypted ids of friends"],
 "useful":"number of useful votes sent by the user",
 "funny":"number of funny votes sent by the user",
 "cool":"number of cool votes sent by the user",
 "fans":"number of fans the user has",
 "elite":["an array of years the user was elite"],
 "average_stars":floating point average like 4.31,
 "compliment_hot":number of hot compliments received by the user,
 "compliment_more":number of more compliments received by the user,
 "compliment_profile": number of profile compliments received by the user,
 "compliment_cute": number of cute compliments received by the user,
 "compliment_list": number of list compliments received by the user,
 "compliment_note": number of note compliments received by the user,
```

"compliment_plain": number of plain compliments received by the user,
"compliment_cool": number of cool compliments received by the user,
"compliment_funny": number of funny compliments received by the user,
"compliment_writer": number of writer compliments received by the user,
"compliment_photos": number of photo compliments received by the user,
"type":"user"
}

yelp_academic_dataset_checkin.json

{
"time":["an array of check ins with the format day-hour:number of check ins from hour to hour+1"],
"business_id":"encrypted business id",
"type":"checkin"
}

yelp_academic_dataset_tip.json

{
"text":"text of the tip",
"date":"date formatted like 2009-12-19",
"likes":compliment count,
"business_id":"encrypted business id",
"user_id":"encrypted user id",
"type":"tip"
}

Appendix B

Words to describe the restaurant

acclaimed	decent	favorite	in	major
affordable	different	few	independent	many
air	down	fine	indian	mexican
american	downtown	finer	inexpensive	more
asian	elegant	finest	interesting	most
best	end	first	international	multiple
better	ethnic	food	italian	nearby
casual	even	foreign	japanese	new
celebrated	excellent	formal	known	nice
certain	exclusive	french	korean	nicer
charming	expensive	friendly	kosher	nicest
cheap	fabulous	german	large	night
chic	famous	good	larger	notch
chinese	fanciest	great	leading	numerous
class	fancy	greek	little	oldest
crowded	fashionable	hottest	local	only

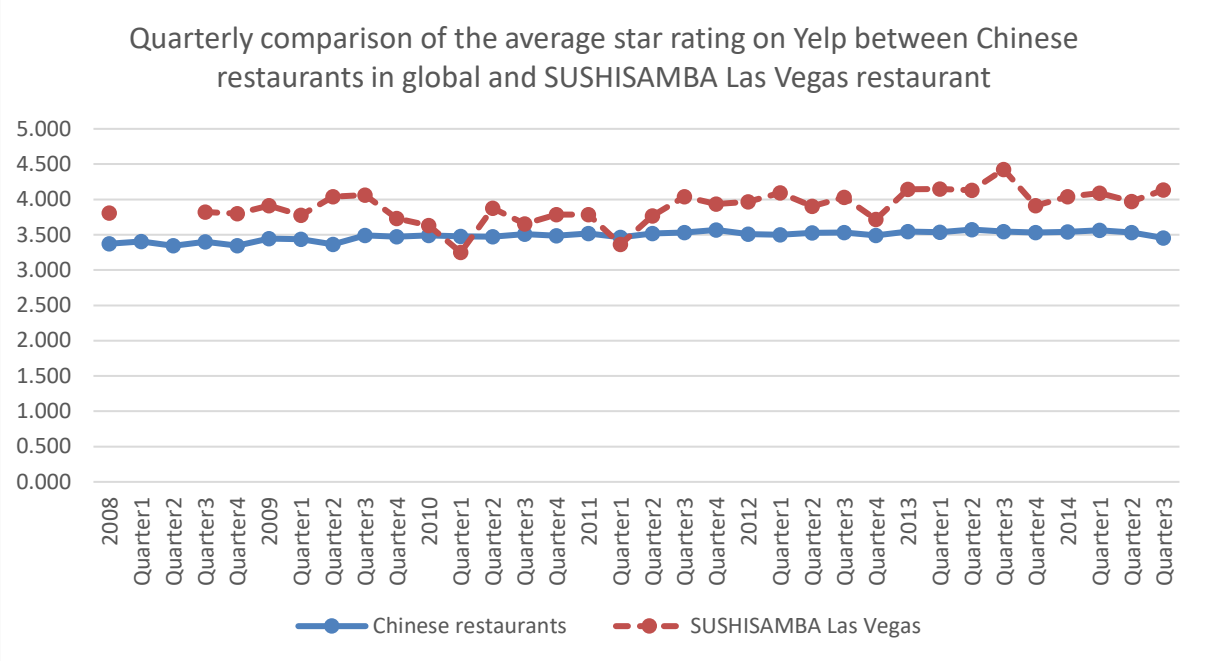
other	pricey	several	style	upscale
outdoor	public	simple	stylish	various
outstanding	quality	site	successful	vegetarian
own	rate	small	such	vietnamese
owned	renowned	smart	superb	wonderful
parisian	romantic	sophisticat-ed	themed	
popular	run	spanish	top	
posh	same	star	traditional	
priced	service		trendy	

Appendix C

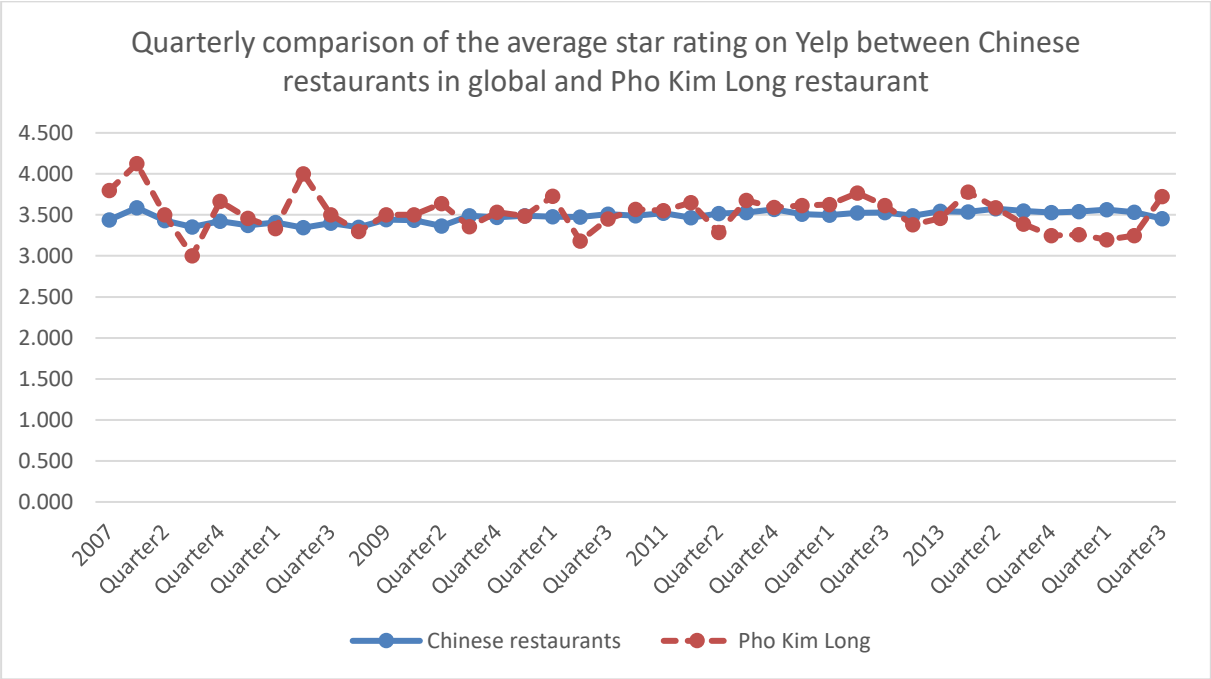
Average star ratings of a category of restaurant in general versus a specific restaurant in that category

	Chinese restaurants	SUSHISAMBA Las Vegas
2008	3.372	**3.810**
Quarter1	3.406	
Quarter2	3.344	
Quarter3	3.399	**3.824**
Quarter4	3.347	**3.800**
2009	3.444	**3.911**
Quarter1	3.435	**3.778**
Quarter2	3.364	**4.038**
Quarter3	3.490	**4.061**
Quarter4	3.472	**3.731**
2010	3.489	**3.630**
Quarter1	3.477	**3.250**

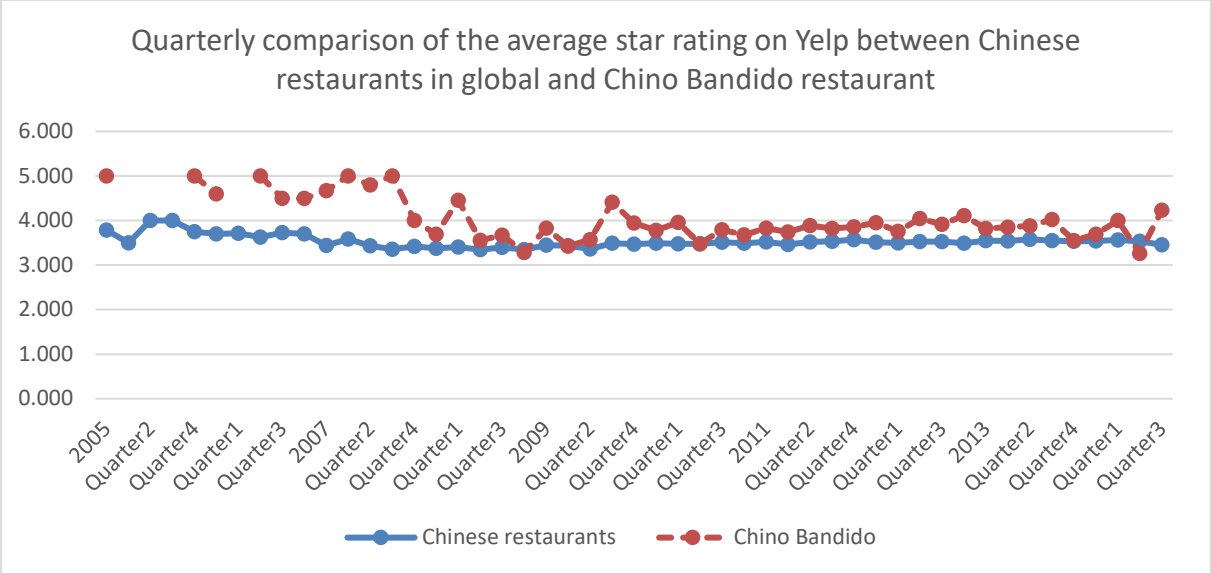
Quarter2	3.474	**3.875**
Quarter3	3.510	**3.653**
Quarter4	3.488	**3.788**
2011	3.520	**3.785**
Quarter1	3.465	**3.364**
Quarter2	3.517	**3.767**
Quarter3	3.531	**4.041**
Quarter4	3.569	**3.933**
2012	3.511	**3.966**
Quarter1	3.498	**4.093**
Quarter2	3.526	**3.904**
Quarter3	3.530	**4.030**
Quarter4	3.490	**3.719**
2013	3.546	**4.142**
Quarter1	3.537	**4.148**
Quarter2	3.574	**4.132**
Quarter3	3.548	**4.426**
Quarter4	3.531	**3.914**
2014	3.539	**4.039**
Quarter1	3.563	**4.089**
Quarter2	3.534	**3.974**
Quarter3	3.454	**4.136**



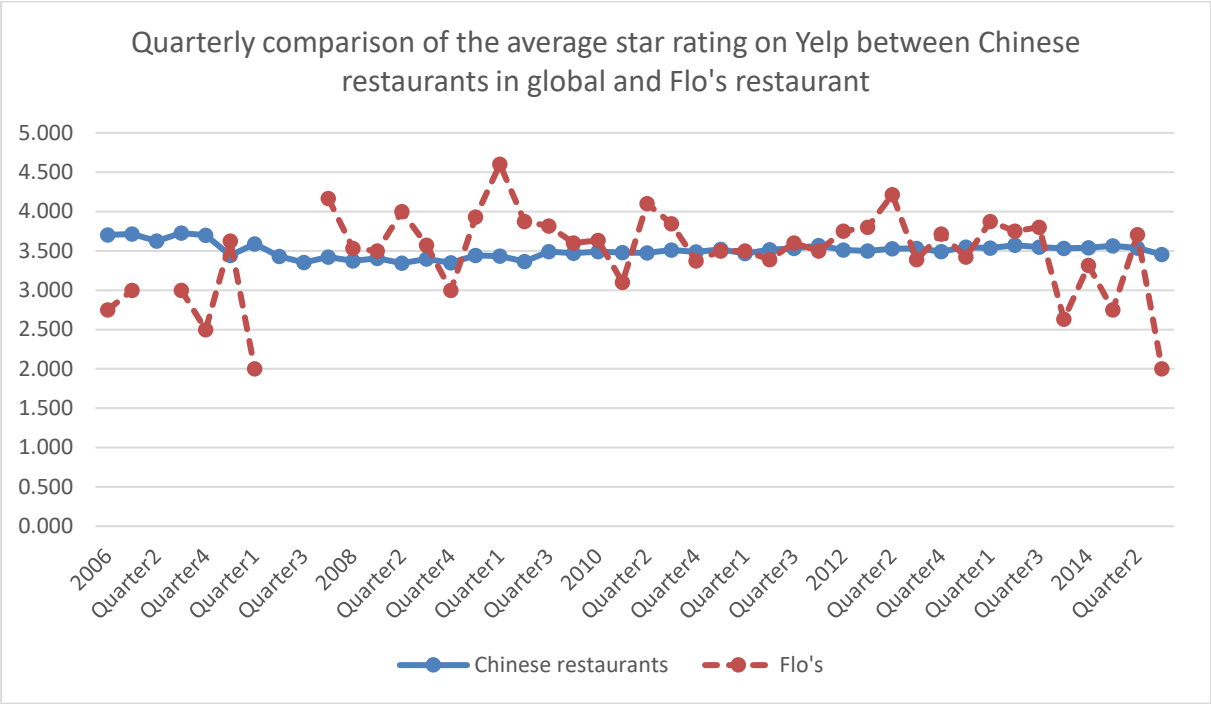
Correlation coefficient 0.336196968



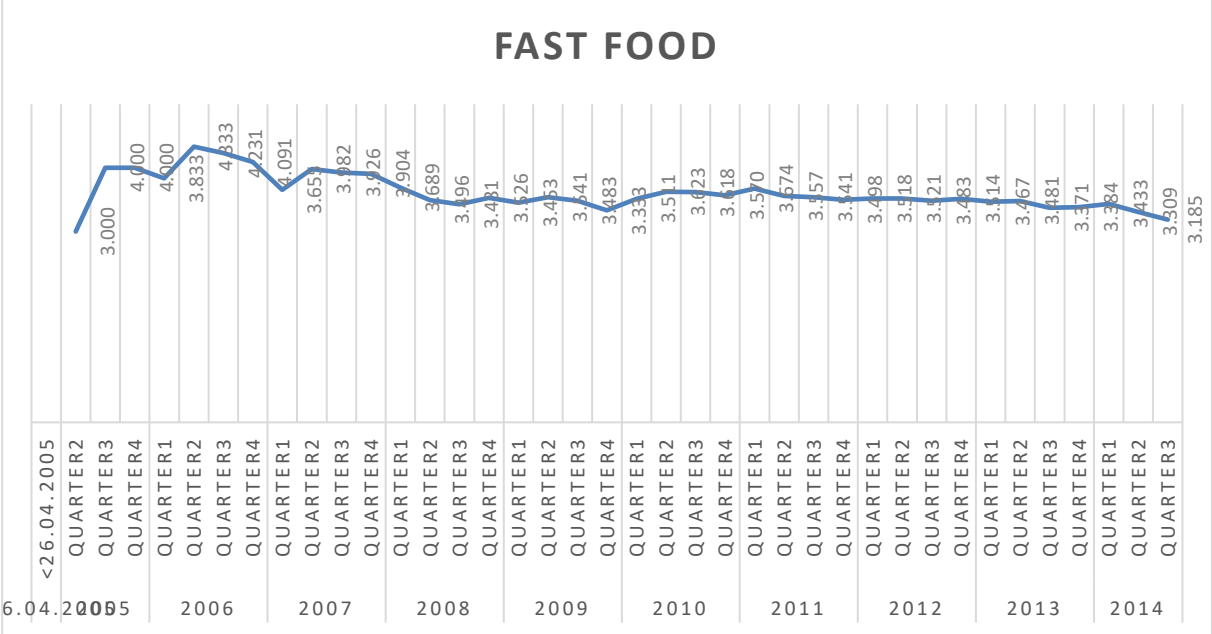
Correlation coefficient 0.072672002



Correlation coefficient 0.475664458

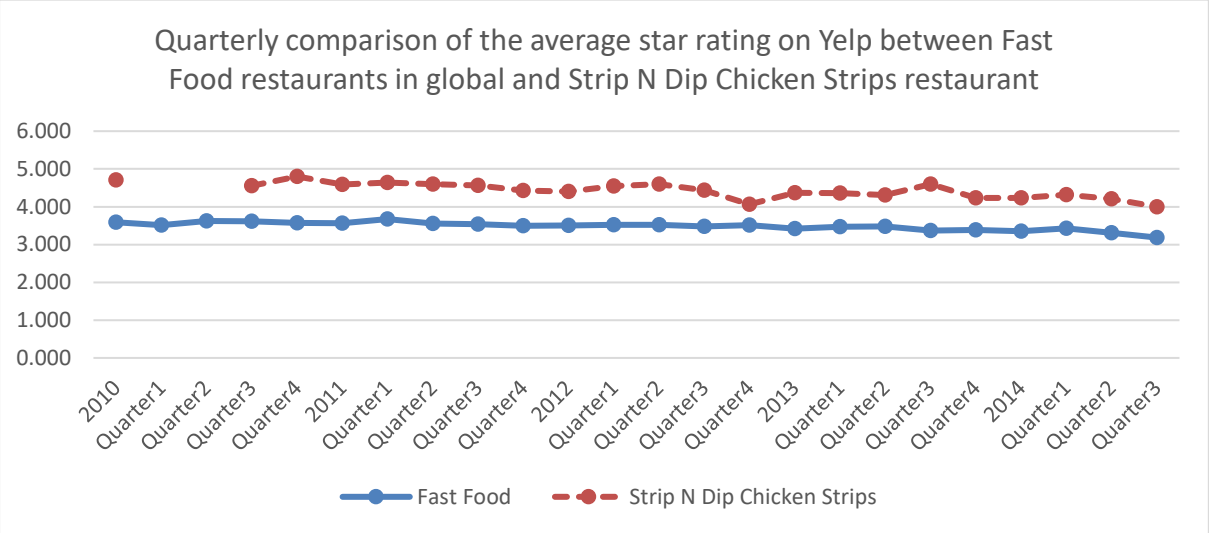


Correlation coefficient -0.427599118

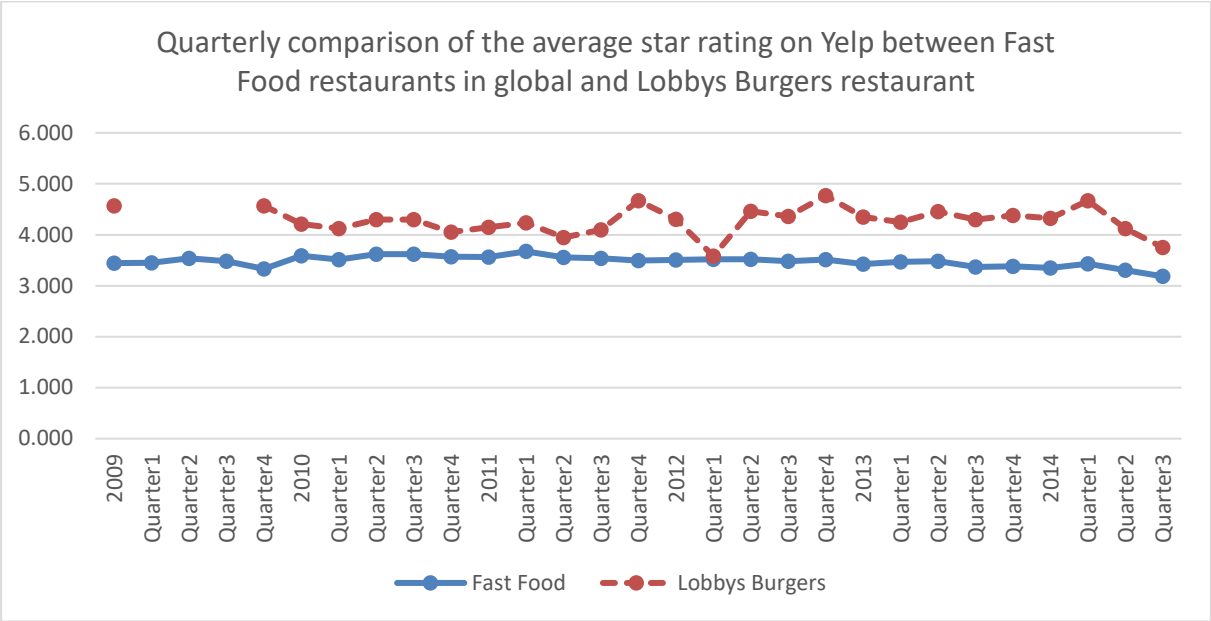


	Fast Food	Strip N Dip Chicken Strips
2010	3.589	**4.708**
Quarter1	3.511	
Quarter2	3.623	
Quarter3	3.618	**4.556**

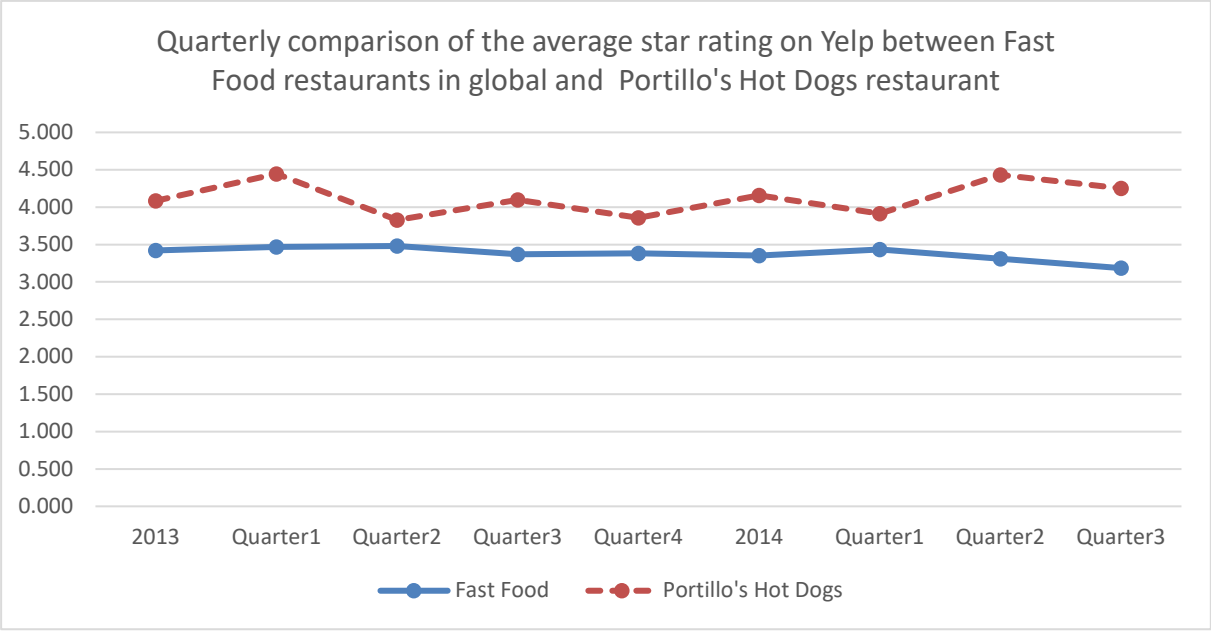
Quarter4	3.570	**4.800**
2011	3.566	**4.589**
Quarter1	3.674	**4.640**
Quarter2	3.557	**4.600**
Quarter3	3.541	**4.567**
Quarter4	3.498	**4.429**
2012	3.509	**4.403**
Quarter1	3.518	**4.550**
Quarter2	3.521	**4.600**
Quarter3	3.483	**4.438**
Quarter4	3.514	**4.063**
2013	3.424	**4.369**
Quarter1	3.467	**4.364**
Quarter2	3.481	**4.308**
Quarter3	3.371	**4.600**
Quarter4	3.384	**4.233**
2014	3.352	**4.235**
Quarter1	3.433	**4.316**
Quarter2	3.309	**4.206**
Quarter3	3.185	**4.000**



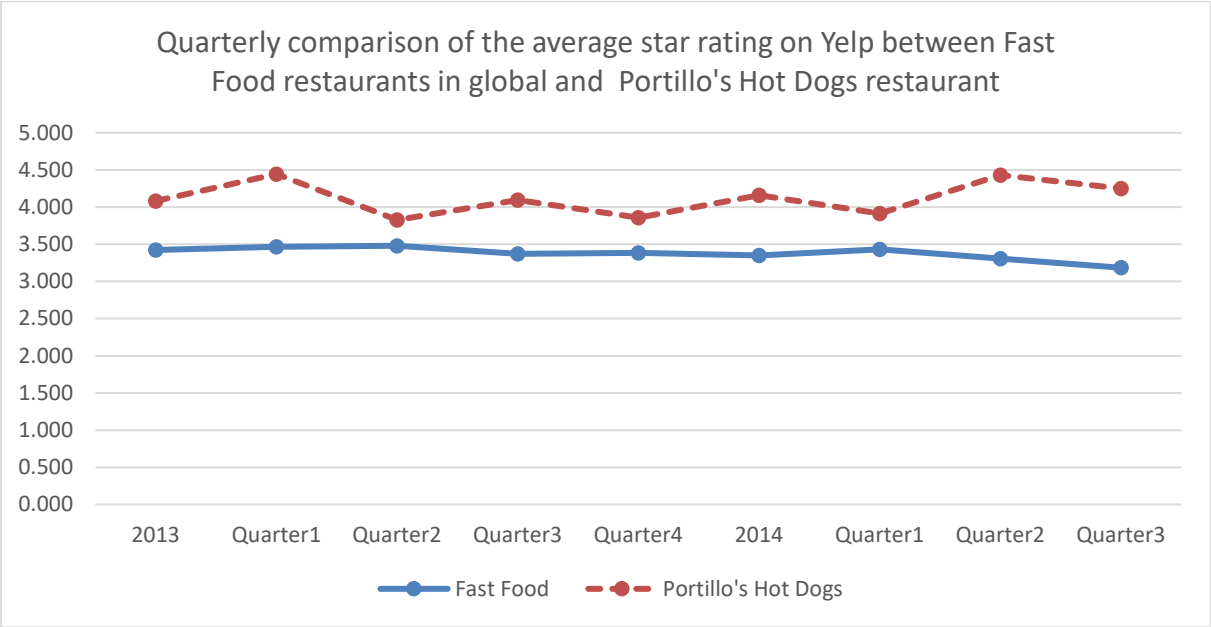
Correlation coefficient 0.73557366



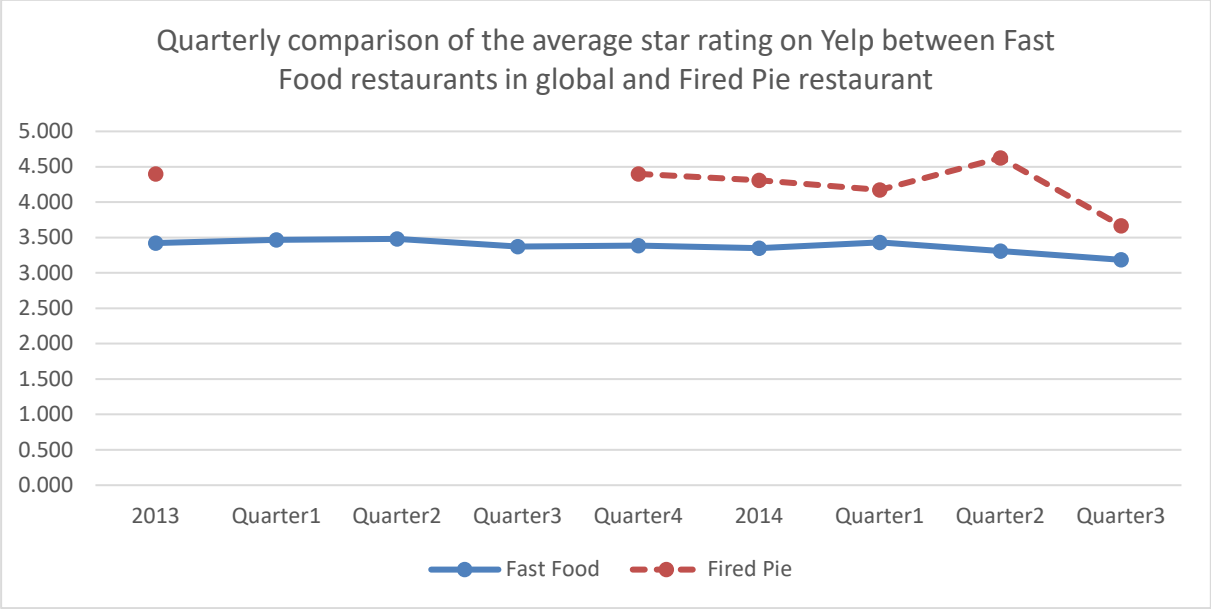
Correlation coefficient 0.015967853



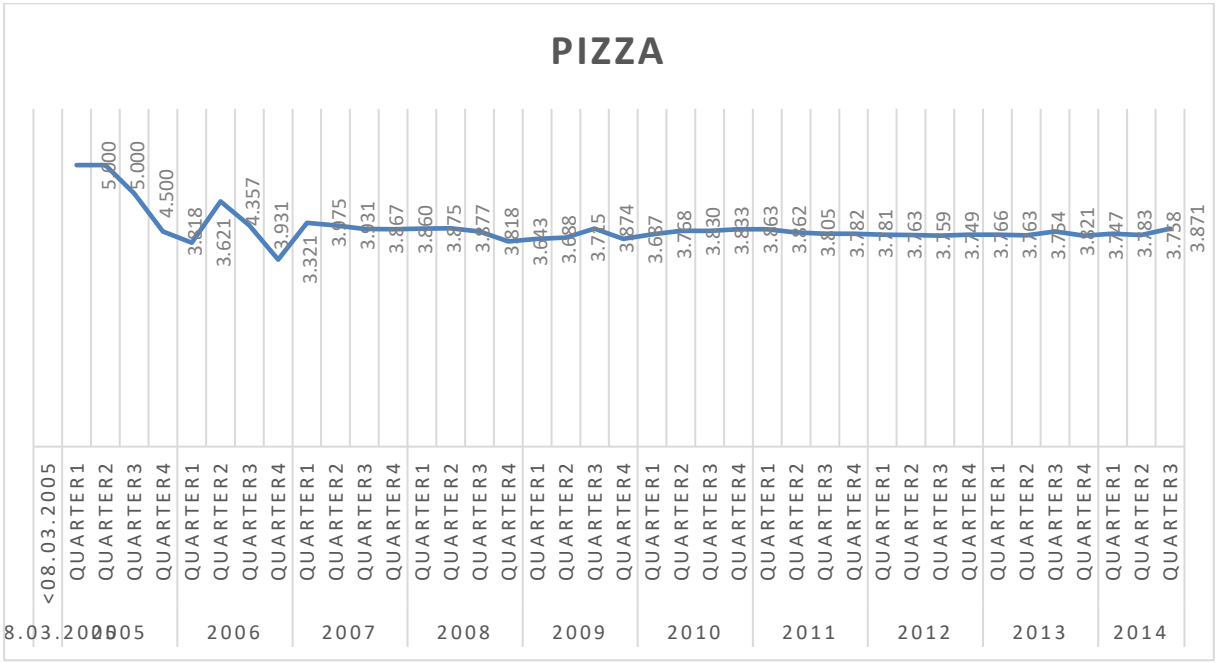
Correlation coefficient -0.376629129



Correlation coefficient -0.376629129

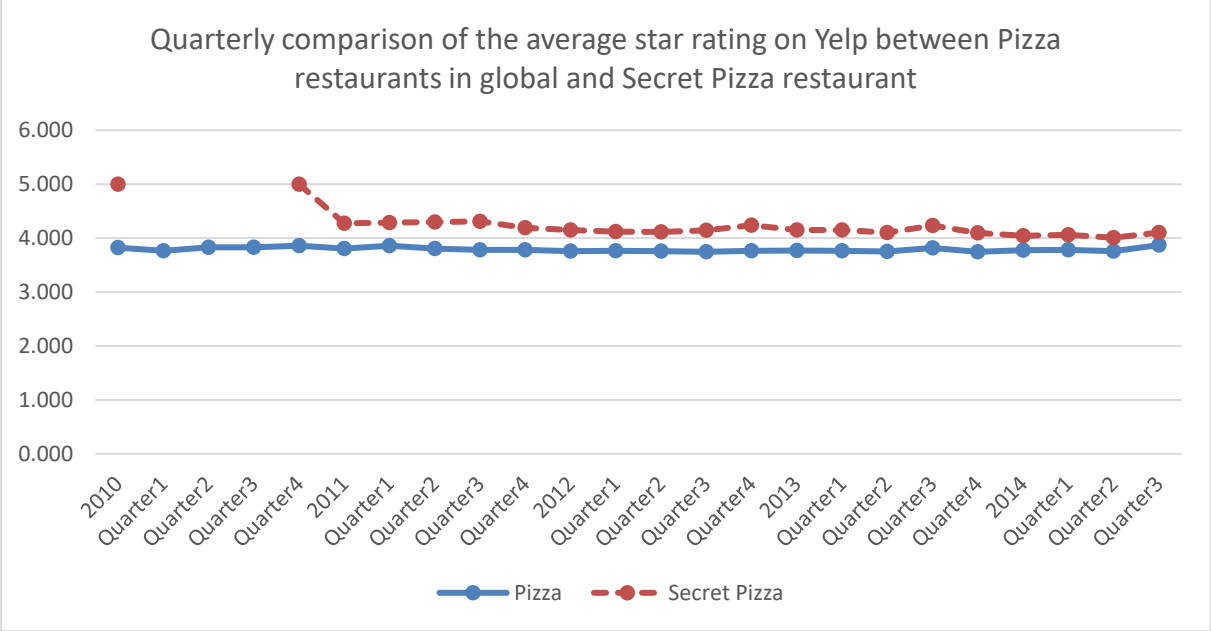


Correlation coefficient 0.605540493

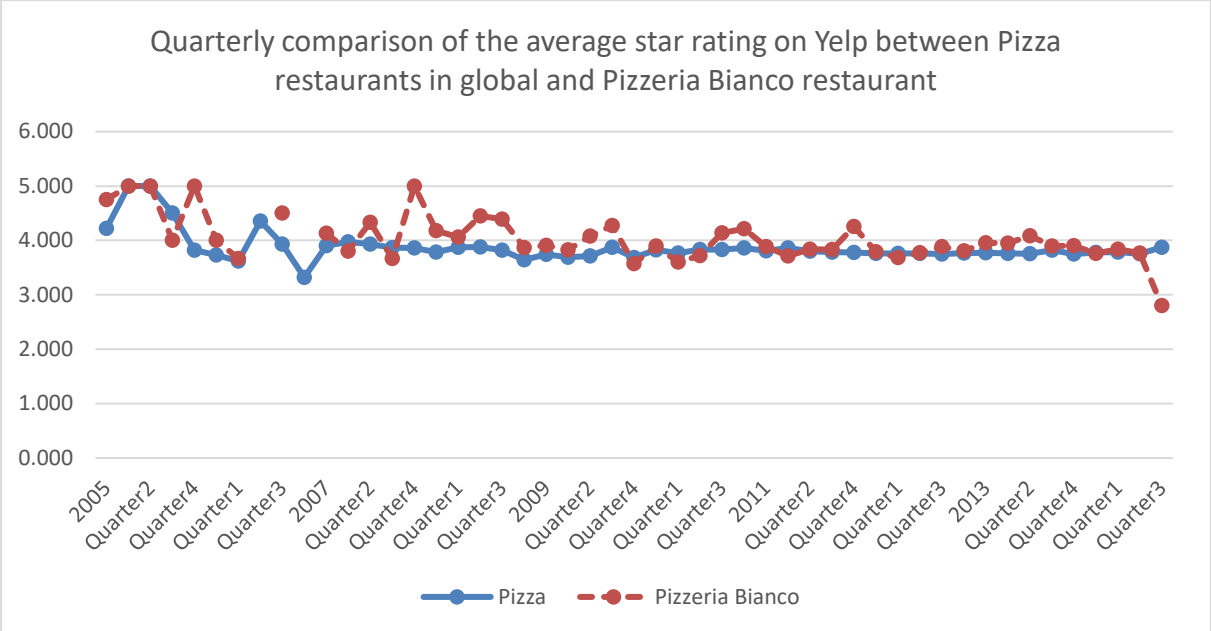


	Pizza	Secret Pizza
2010	3.826	**5.000**
Quarter1	3.768	
Quarter2	3.830	
Quarter3	3.833	
Quarter4	3.863	**5.000**
2011	3.807	**4.273**
Quarter1	3.862	**4.286**
Quarter2	3.805	**4.302**
Quarter3	3.782	**4.314**
Quarter4	3.781	**4.194**
2012	3.759	**4.149**
Quarter1	3.763	**4.123**
Quarter2	3.759	**4.117**
Quarter3	3.749	**4.146**
Quarter4	3.766	**4.242**
2013	3.772	**4.152**
Quarter1	3.763	**4.153**
Quarter2	3.754	**4.103**
Quarter3	3.821	**4.237**
Quarter4	3.747	**4.100**
2014	3.780	**4.043**
Quarter1	3.783	**4.063**

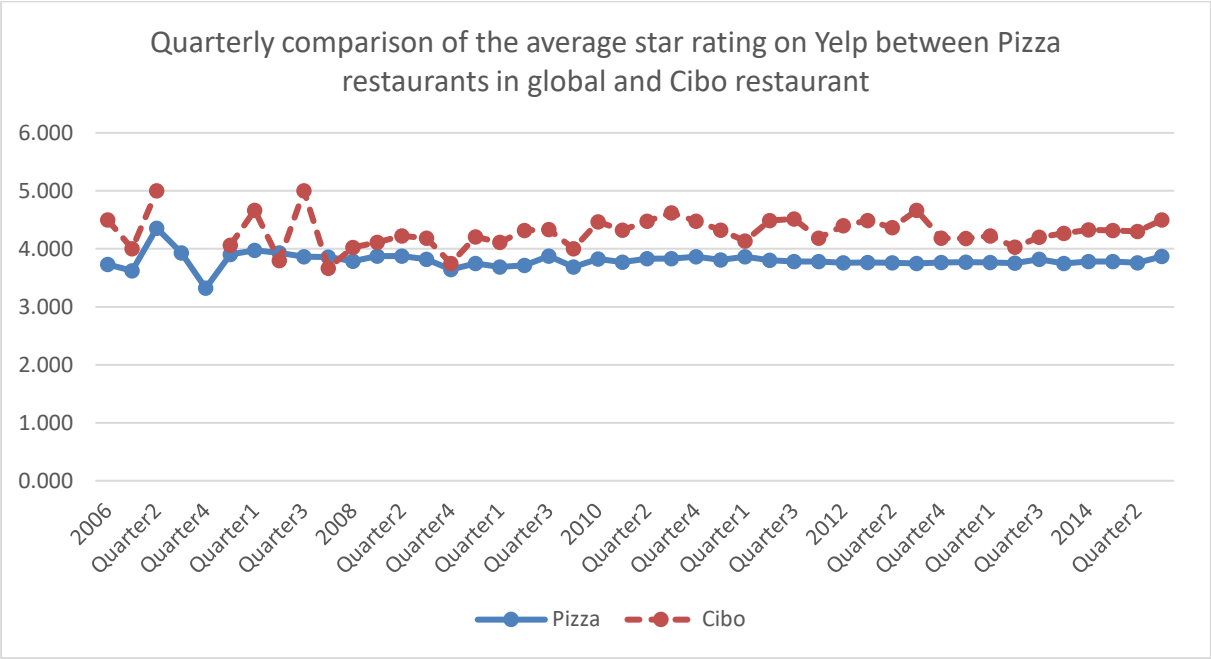
Quarter2	3.758	**4.008**
Quarter3	3.871	**4.103**



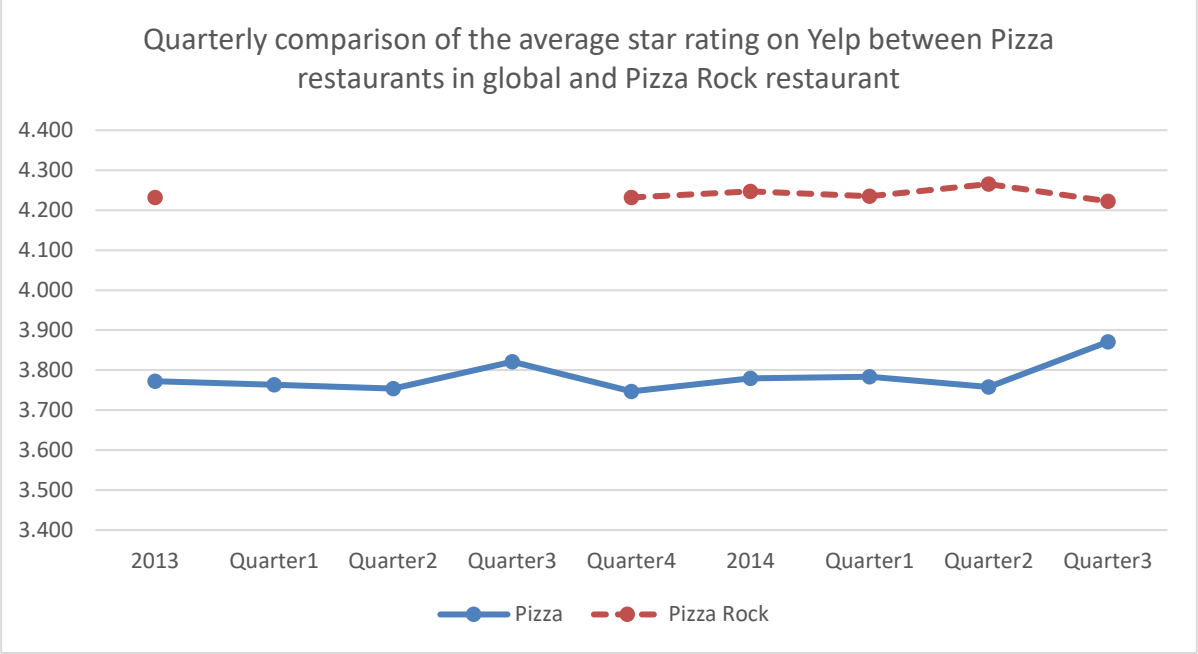
Correlation coefficient 0.558793661



Correlation coefficient 0.544685686



Correlation coefficient 0.439583635



Correlation coefficient -0.550523454

Appendix D

The complete Pig Latin code that processes the sentiment polarity calculation

```
-- Here these few lines of comments show how to use either directly " HCatalog", how to load a "JSON" file or load the first sentence of each review from the yelp.first_tuples_from_bag table we created erlier. When need are uncommented to be used.
-- REGISTER /user/oozie/share/lib/lib_20150424130956/hcatalog/hive-hcatalog-core-0.14.0.2.2.4.2-2.jar;
-- RAW = LOAD 'yelp_academic_dataset_review_short_POS_split_text_first.tsv/part-m-00000' using PigStorage() AS (fname:chararray, text:chararray);
-- Note : using the review.json as it is, is not possible. There is not enough place on the HDFS for this calcule.
-- RAW = load 'yelp_academic_dataset_review_short.json' using JsonLoader('votes:map[],user_id:chararray,fname:chararray,stars:int,date:chararray,text:chararray,type:chararray,business_id:chararray');
A = LOAD 'yelp.first_tuples_from_bag' USING org.apache.hive.hcatalog.pig.HCatLoader();
-- RAW = LOAD '/user/mdavary/yelp_academic_dataset_review/03_POS_split_into_sentences.tsv/part-m-00001' USING PigStorage('\t') AS (funny:int, useful:int, cool:int, user_id:chararray, fname:chararray, stars:int, text:chararray, textnstw:chararray, pos:chararray, sentences:chararray, date:chararray, type:chararray, business_id:chararray);

-- Assumes data format is <docname>\t<doctext>
-- take review_id as fname and text as text

RAW = LOAD 'Corpus_of_Yelp_reviews.txt' USING PigStorage('\t') AS (fname:chararray, text:chararray);

RAW = foreach A generate review_id as fname, first_tuple as text;

DOCUMENTS = FOREACH RAW GENERATE fname, TOKENIZE(REPLACE(LOWER(text), '[^a-zA-Z]+', ' ')) as word;
WORDS_IN_DOC = FOREACH DOCUMENTS {
        d_word = DISTINCT word;
        GENERATE d_word as words;
}

-- Flatten everything so we have <document>, <word> pairs
ONLY_WORDS = FOREACH WORDS_IN_DOC GENERATE flatten(words) as word;

-- Re-group so we can COUNT
WORDS = GROUP ONLY_WORDS BY word;
COUNTS = FOREACH WORDS GENERATE group, COUNT(ONLY_WORDS) as count;

-- Determine the actual length of the document in case we want to adjust TF for document length
DOC_LEN_RAW = FOREACH DOCUMENTS GENERATE fname, flatten(word);
RAW_GROUP = GROUP RAW ALL;
```

```
RAW_COUNT = FOREACH RAW_GROUP GENERATE COUNT(RAW) as docs;
RAW_DOC_GROUP = GROUP DOC_LEN_RAW BY fname;
DOC_LENGTHS = FOREACH RAW_DOC_GROUP GENERATE group as fname, COUNT(DOC_LEN_RAW) as length:long;

-- If a word occurs in more than 95% of documents, drop it.
INTERESTING_COUNTS = FILTER COUNTS BY count < .95*(double)RAW_COUNT.docs;
SORTED = ORDER INTERESTING_COUNTS BY count ASC;

-- TFIDF : term frequency–inverse document frequency
IDF = FOREACH SORTED GENERATE group as word, LOG((double) RAW_COUNT.docs/(double)count) as idf;
DOCS_JND = JOIN DOCUMENTS by fname, DOC_LENGTHS by fname;
DOC_WORD = FOREACH DOCS_JND GENERATE DOCUMENTS::fname as fname, length, flatten(word) as word;
WC_GROUPED_BY_DOC = GROUP DOC_WORD BY (fname, length, word);
TF = FOREACH WC_GROUPED_BY_DOC GENERATE flatten(group), COUNT(DOC_WORD) as tf, (dou-
ble)COUNT(DOC_WORD)/group.length as la_tf;

PRE_TFIDF = COGROUP TF by word, IDF by word;

PRE_PRE_TFIDF = FOREACH PRE_TFIDF GENERATE flatten(TF), flatten(IDF);

-- Generate both normal and length-adjusted weight.
TFIDF = FOREACH PRE_PRE_TFIDF GENERATE fname, TF::group::word as word, (double)tf*idf as weight, la_tf*idf as la_weight;

SENTIMENT_RAW = LOAD 'SentiWordNet_3.0.0_20130122.txt' using PigStorage() AS (pos:chararray, id:chararray, pos_score:double,
neg_score:double, obj_score:double, synset_term:chararray, gloss:chararray);

SENTIMENT_ALL = FOREACH SENTIMENT_RAW GENERATE flatten(TOKENIZE(synset_term)) as synset_term, pos_score,
neg_score;

SENTIMENT_DATA = FOREACH SENTIMENT_ALL GENERATE REGEX_EXTRACT(synset_term, '([^#]+)#([0-9]+)', 1) as synset,
pos_score, neg_score;

SENTIMENT = COGROUP TFIDF BY word, SENTIMENT_DATA BY synset;

SENTIMENT_TMP = FOREACH SENTIMENT GENERATE flatten(TFIDF), flatten(SENTIMENT_DATA);

SENTIMENT_GROUPED = GROUP SENTIMENT_TMP BY (fname, word, weight, la_weight);

-- Drop the word here, because we don't actually need it any more.

SENTIMENT_AVG = FOREACH SENTIMENT_GROUPED GENERATE group.fname as fname:chararray,
group.weight*AVG(SENTIMENT_TMP.pos_score) as positive:double, group.weight*AVG(SENTIMENT_TMP.neg_score) as nega-
tive:double, group.la_weight*AVG(SENTIMENT_TMP.pos_score) as la_positive:double,
group.la_weight*AVG(SENTIMENT_TMP.neg_score) as la_negative:double;

DOCS_GROUPED = GROUP SENTIMENT_AVG BY fname;

--
```

```
-- Everything from here down is specific to the data set and document naming convention I used, replace it with your own logic.
--

TOTAL_SENTIMENT = FOREACH DOCS_GROUPED GENERATE group as fname, REGEX_EXTRACT_ALL(group, '([0-9]+)-([0-9]+)-([0-9]+)-(.+).txt') as details:(year:chararray, month:chararray, day:chararray, type:chararray), SUM(SENTIMENT_AVG.positive) as pos:double, SUM(SENTIMENT_AVG.negative) as neg:double, SUM(SENTIMENT_AVG.la_positive) as la_pos:double, SUM(SENTIMENT_AVG.la_negative) as la_neg:double;

STORE IDF INTO 'senti-idf-review_all';
STORE TF INTO 'senti-tf-review_all';
STORE TFIDF INTO 'senti-tfidf-review_all';
STORE TOTAL_SENTIMENT INTO 'senti-sentiment-review_all';

B = FOREACH TOTAL_SENTIMENT GENERATE REPLACE(REPLACE(details.type, '10[qk]', '10q/k'), '.*earnings-transcript.*', 'earnings-statement') as type, fname, details.year as year, details.month as month, details.day as day, pos as pos_y_coord, neg as neg_y_coord, la_pos, la_neg;
STORE B into 'sentiment-final-review_all';
```