THE UNIVERSITY OF TOKYO

Graduate School of Information Science and Technology
Department of Information and Communication Engineering

Master Thesis

Hidden Topic Modeling Approach for Online Review Quality Prediction and Classification

「潜在的トピック分布のアプローチを用いたオンライン
レビュー質の予測と分類」

Tran Quoc Hoan

February 2016

Thesis Supervisor . . . . . Hiroshi Esaki
Professor
Thesis Supervisor . . . . . Hideya Ochiai
Assistant Professor

Abstract

The importance of automatic quality assessment on online review is getting recognized, according to the rapid increase of the number of reviews, as well as the increase of reviews. In order to grasp review's quality, some online services adopt a system where users can evaluate or feedback against the reviews as a kind of crowdsourcing knowledge approach. It is recognized that this approach has fatal shortcomings: sparseness of voted data distribution and a richer-get-richer problem in which favor reviews are voted much frequently than others. Some other approaches using the information of helpful/unhelpful votes against reviews to conduct a hybrid of unsupervised and supervised leaning in the process of evaluation of reviews' quality. Each review is represented as features vector via unsupervised feature extractor, then a supervised learning machine is applied to these feature vectors against the reviews which have high number of votes. The output of this learning machine is numeric value quality of reviews.

In the previous works, many of feature types in representation of reviews are suggested. However, these kinds of features do not represent the essential information in review content which is the major factor in evaluating review's quality. Moreover, quality of a review is defined simply as helpfulness votes ratio led to bias understanding without probabilistic meaning of review's quality. Some of learning algorithms also focused on this ratio without considering the information of votes' population.

The contributions of this thesis are summarized as the following fivefold. Firstly, we exploit the hidden topics distribution information of all reviews text and propose a topics distribution in each stand-alone review as a new feature vector representation. Our study

reveals that the topics' distribution gives high understanding to the content that could be led to the decision in evaluating review's quality.

Secondly, we propose a probabilistic definition for review quality. Under this definition, the quality of a review is given with mathematical meaning and associated with a probabilistic framework.

Thirdly, we also propose supervisor prediction model (logistic regression and learning algorithm) based on probabilistic meaning of the review's quality. This model incorporates the votes' information for reviews into cost function and performs better than conventional methods.

Fourthly, we propose confident models combination method to take the most confident output when we apply learning machine to different type of features. The confident models' combination proposed in this research archives the best performance than any single type of features.

Finally, we demonstrate that our proposals achieve to deliver better accuracy regarding the evaluation of review's quality in real implemented system and with real review dataset. To our best knowledge, this is the first framework and application for the modeling of review quality and measuring the effect of feature design and learning algorithm.

Acknowledgments

First and foremost, I would like to express my sincere gratitude to Professor Hiroshi Esaki for giving me a great opportunity to be a part of his laboratory and a chance to study and conduct the interesting research. In particular, I am grateful for his time and effort he has spent to make my research's direction clear and objective as well as the encouragement he has given me over the years. He also pointed out the motivation of this research, gave me priceless advices and guidances, that is without his supervision my thesis would not be completed.

I am extremely grateful to Assistant Professor Hideya Ochiai for all of the invaluable advices and kind supports. He always gives kind considerations with many helpful suggestions that made the substantial improvements in my research. He has also led me to see things in scientific way with motivations and inspirations for growing up in research activities. Without his support, it is difficult for me to conduct the research in a proper method.

I really appreciate Assistant Professor Hirochika Asai and Assistant Professor Manabu Tsukada for their precious advices and suggestions on my research. Their vision in scientific research is admirable.

I would like to thank all members in Esaki Laboratory that always give kind supports and advices helping me through these years. Specially, I would like to give a millions thanks to Mr. Hiroki Nakagami and Mr. Tokaku Hiroshi for their invaluable friendships over many years since we joined the same department. I wish them every success in their future.

Finally, I would like to express all my sincere gratitude to my parents, my wife and my sister for their support and encouragement during the years of study. Without their unconditional love, none of this thesis would have been possible.

"This thesis is dedicated to my loving son who was born on October 28, 2015".

Contents

List of Figures

List of Tables

Chapter 1

Introduction

In this chapter, we discuss the motivation and background of this research. This chapter also presents the research problems, the target system, the contributions of our research, and the organization of this thesis.

1.1 Motivation

Online review is one of popular and important types of user-generated resources, where users can publish their experiences and opinions about products, events or services. As online activities continue to grow, the role of online reviews is expected to become increasingly important, especially in the decision-making process relating to users' actions in online system [1][2]. Some studies in [3][4][5] have continuously shown that the quantity and quality of consumer-generated reviews have a positive effect on purchaser's intentions and become major information source for consumers and marketers regarding to product quality. From the business view, the comprehensive and profound knowledge from reviews holds the successful key for some review portals associated with online commerce (Amazon.com) or online services (Yelp.com, Tripadvisor.com).

However, the big volume of online reviews today makes the process of extracting helpful information becoming more and more difficult. The quality of consumer-generated varies

and remains uncovered. In other words, users are encountering with mind confusing problem to find interesting and helpful opinion in mixtures of unhelpful or highly subjective and misleading information.

To deal with this problem, some review portal sites are providing a mechanism where users can evaluate or rate the helpfulness of a review (e.g. Amazon.com and Yelp.com). However, the disadvantage of provided mechanism is the top reviews attract more and more rating while more recent reviews are rarely read and thus not rated [6]. It is thus highly desirable to develop robust and reliable methods to evaluate the quality of reviews automatically. The system incorporating this method can discover and provide the most helpful reviews to the consumers with requirements of reducing time, effort and difficulties in acquiring high quality reviews.

1.2 Research Problem

To understand general problem in determination of online review quality, we firstly show an example of an online review page from Amazon.com in Figure 1-1. From the figure, we can see the product reviews are sorted by "most helpful" metric based on votes from previous readers on each review (e.g. *"343 of 349 people found the following review helpful"*). The potential reader may make use of previous votes information in two stages of his/her decision making: whether to read review or not, and then whether to buy product or not. After the reader has read the review, he/she can vote on the review to be "Helpful" or "Not Helpful" by answering YES/NO to the question from portal site "Was this review helpful to you?". The new vote information is accumulated again for voted review and is shown to other readers. This makes a cycle of processing.

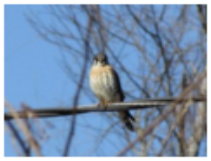
In general, using votes information for review as crowded-source knowledge is one of the most useful tools to build review quality measurement. However, this kind of knowledge is not often available for all reviews. Consider a situation when review stream is flowing fast thus one can miss voting for a new review. The review which does not have enough number

★★★★★ **High Quality with extended zoom-Love it!**

By Linda Harris on March 13, 2015

Style Name: Base | Color: Red | **Verified Purchase**

Way above expectations, I love this camera! I have already taken some wonderful pictures with it of wildlife in my area. I wanted a camera to take high quality pictures with extended zoom, without me putting a lot of thought into settings on the camera itself. If that is what you want from a camera, this is the one you have been looking for. Click or Mouse over the picture that I have included to see a larger version of it. I love the clarity of the subject itself and how the foreground and background are blurred. This picture was taken in the automatic setting using the Auto button, conveniently located on the back of the camera!



4 Comments | 343 of 349 people found this helpful. Was this review helpful to you? Yes No

Report abuse

★★★★★ **Good luck!**

By next24 on January 5, 2016

Style Name: Base | Color: Black

Buying a video camera can be a huge investment. I personally didn't want to just go to my local store and buy just any / or the latest model out there just because it was on sale that week... there is a huge variety of cameras (most people can use their phone) but if you need to take video for an extended period time, you should now what you're getting. There were many things to consider before purchasing; what was I using it for, what features I need, what format (HD or standard) and what recording medium.

While researching I found a page called justafax (you can google it). They did most of the work by compiling the top camcorder choices, compared them side by side with all their features and benefits. I just had to choose one according to my needs. It made my road to purchasing the right camcorder a lot smoother and in the end I got exactly what I needed.

I've told all my friends and family about because justafax.com took a load off my shoulders. Now I'm recommending it to all of you! Good luck!

Comment | 253 of 259 people found this helpful. Was this review helpful to you? Yes No

Report abuse

Figure 1-1: An example of an online review page from Amazon.com. Each review is associated with "helpful or not helpful votes" from readers. The potential reader may use this associated information to decide whether to read review. If he/she has read it, he/she can vote for review itself. This information is accumulated again and is shown to other readers.

of votes for itself could not be processed by method based on votes population. Zhang et. al. [7] have mentioned the "Words of few Mouths" phenomenon in real review portal system, where there is a large fraction of reviews only having feedbacks from very few users. Figure 1-2 plots the relationship between the number of reviews and number of votes for each review in Yelp review dataset provided in [8]. We can see that a small portion of the reviews attracts most of the feedbacks while a large number of reviews attract little feedbacks.

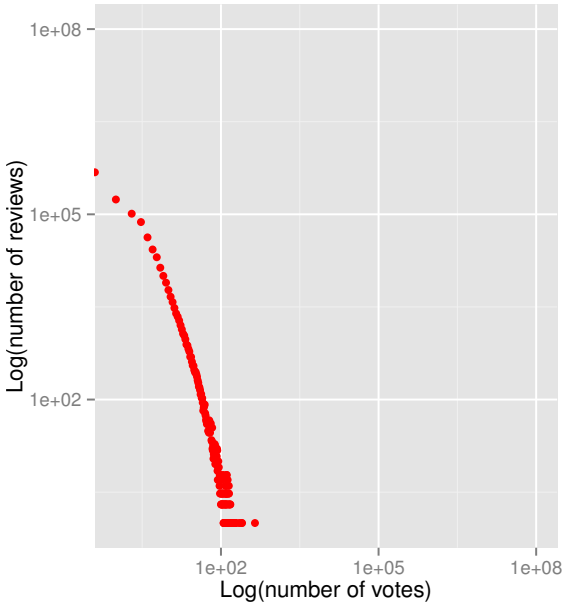


Figure 1-2: The relationship between the number of reviews and number of votes for each review in Yelp review dataset. A small portion of the reviews attracts most of the attention while a large number of reviews attract little attention ("Words of few Mouths" phenomenon).

In this thesis, we consider a typical target system called "ReQE" (Review Quality Evaluation) for solving two main tasks: review quality prediction and review quality classification. The first one includes quality definition and quality numeric inference, the second one includes quality class inference for the reviews that do not have enough number of votes for their helpfulness. ReQE can be combined with regular review portals or other recommendation systems to providing the most helpful reviews that can assist customers in better understanding their intended online decision. The general structure of ReQE is displayed in Figure 1-3.

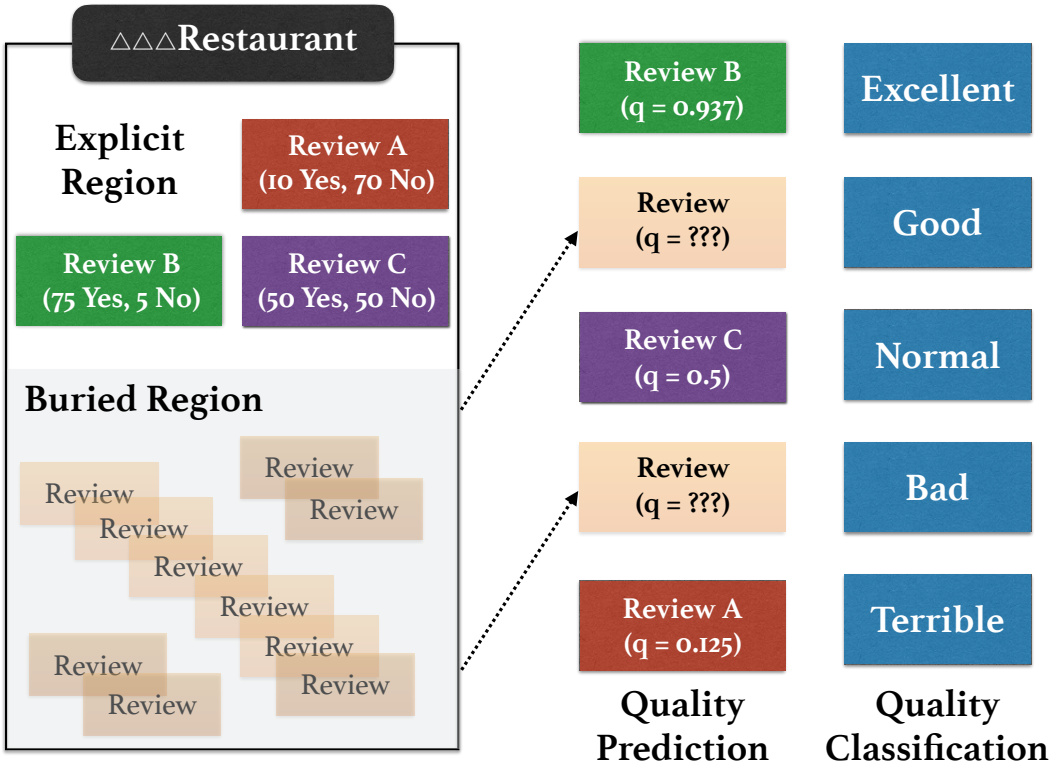


Figure 1-3: General structure of our target system (ReQE) solving for review quality prediction and classification task. ReQE uses reviews with many feedbacks in the explicit region as supervisor data and evaluates quality for the reviews that do not have enough number of votes in buried region.

1.3 Research Methodology

This thesis focuses on choosing suitable and useful feature set to represent each review in quality prediction and classification problem. And then we propose the state of the art in design learning method with this set of features to evaluate the quality of reviews.

An initial idea is treating each review as a stand-alone text document, extracting statistical textual features from the text and proposing a function based on these features for predicting/classifying review quality. However, in addition to statistical textual features, there is much more information available in review's text. We propose a simple idea that all reviews discussed the same number of topics but with the different proportion of topics in each review. We call this kind of topics as hidden topics and use topic modeling algorithm to extract these hidden topics and their proportion in each review.

We then build a probabilistic model with mathematical formulation and probabilistic definition of review quality. In this proposed framework, we study the problem of designing algorithm inferencing numeric value for review quality. We propose logistic regression method and demonstrate the superior performance with support vector regression as state of the art in the previous works.

We also consider review quality as discrete labels in five level classes "Terrible", "Bad", "Normal", "Good", "Excellent" and train a classifier on supervised data. The difference of classification task with prediction task is the class label reduces the bias of review numeric quality (e.g. some of votes on review is unreliable).

Finally, we consider the variation of results when applying different features types and propose a confident models combination method to archive better results. Our learning machine takes different feature types to produce different outputs but adopting only the most confident value in set of these outputs.

1.4 Contribution

The contribution of this research are as follows:

1. Hidden topics distribution as representation for review text. This feature type has the semantic meaning to infer quality of reviews and performs good performance in evaluation.

2. Probabilistic definition for review quality. Under this definition, the quality of a review is given with mathematical meaning and probabilistic framework.

3. Logistic regression and learning algorithm for predicting a review's quality. This method takes votes information for reviews into cost function and perform better than the previous state of the art - Support Vector Regression (SVR) method.

4. Models combination method to take the most confident output when apply learning machine to different types of features. The models combination proposed in this thesis shows the best performance than any single type of features.

5. Evaluating proposals and previous works in real reviews dataset and implementing in real recommendation service.

1.5 Organization of Thesis

This thesis is divided in six chapters, with detailed description is given below:

- Chapter 1 proposes the research problem, the target system and contribution of this thesis.

- Chapter 2 discusses existing works and related techniques using in this thesis.

- Chapter 3 gives proposals for research problem as hidden topic modeling for review feature extraction, probabilistic definition and mathematical framework for review quality, logistic regression using votes information to predict review quality as its helpfulness. This chapter also presents the simple but useful method of models combination to produce the confident output in tasks of predicting and classifying review quality.

- Chapter 4 introduces the implemented system in details.
- Chapter 5 delivers the experiments setup, the evaluation method and discussion in the experiments results.
- Chapter 6 gives conclusions for this thesis and discusses our future research direction in feature design, modeling algorithm and human labeling for ground-truth data.

1.6 Peer Reviews Publication

1. Hoan Tran Quoc, Hideya Ochiai, Hiroshi Esaki, "Hidden Topics Modeling Approach for Review Quality Prediction and Classification", *In Proceeding of IEEE 7th International Conference on Soft Computing and Pattern Recognition (SoCPaR 2015)*, Fukuoka, Japan, November 2015.

Chapter 2

Related Works

In this chapter we introduce some related works and fundamental techniques used in this thesis.

2.1 The Impact of Online Reviews Quality on Online Behavior

Many literature works investigate the important role of online reviews in the decision process of online action (e.g. shopping, booking hotels, etc...). Park et al. (2007) [3] find that consumers' intentions of purchasing product increase as the quantity of product reviews. Duan et al. (2008) [9] show that consumers often read product reviews when finalizing purchase decisions. And Hu et al. (2008) [4] point out that not only review ratings, but also the contextual information (e.g. reviewer's reputation) is important factor considering by consumers.

In addition, Park and Kim (2008) [5] present the variation types of reviews and their effect to consumers. Lee et al. (2008) [10] investigate the effect of negative online reviews on level of involvement consumers. Vermeulen and Seegers (2009) [11] also examine that positive reviews have a positive impact on consumer behavior in booking hotels. Moreover, in Park and Lee (2008) [12], they studied roles of the online consumer reviews in quality and

quantity aspects. They conclude that number of users on review websites increases rely on the quality of the reviews rather than on their quantity.

Based on these findings, it can be concluded that quality of online reviews has an influential impact to consumers' online behavior.

2.2 Review Helpfulness Prediction

Reviews keep playing an important role in decision-making but the large amount of available user-generated reviews today causes confusing to the users. The more popular product or service, the more it attracts reviews. However, a large proportion of these reviews is spam content or unhelpful opinion. For this reason, it is very important for review portal retailers to rank reviews and provide the most helpful or the highest quality reviews to consumers.

Several previous works (Liu et al (2008) [13]; Danescu-Niculescu-Mizil et al (2009) [14]; Mudambi and Schuff (2010) [15]; Ghose and Ipeirotis (2011) [16]) have suggested that the helpfulness or quality of consumer-generated reviews should be defined as what degree a review help in making purchase decision. In [17], A. Spool presents an interesting survey resulted that the question "Was this review helpful to you?" helps to increase an estimate $2.7B revenue to Amazon.com annually.

Most of works on evaluating the quality and helpfulness of reviews has addressed the solution by treating each review as different sets of features and proposing a function based on these features for predicting its quality.

Kim et al. (2006) [18] propose an method to automatically identify review helpfulness by using Support Vector Regression (SVR) to train on three main features: the length of the review, the product rating and the tf-idf score in unigram model. Zhang and Varadarajan (2006) [19] incorporate a diverse set of features from statistical textual features in review text to build a review helpfulness regression model. Liu et al. (2008) [13] develop a nonlinear regression model for the helpfulness prediction. Their study on IMDB movie reviews dataset demonstrates that the helpfulness of a review depends on three important factors: the re-

viewer's expertise, the writing style of review, and the timeliness of the review. O'Mahony and Smyth (2010) [20] indicate that structural and readability features are useful for review helpfulness prediction. These works lack principles in probabilistic model and focus on statistics computed from review text but not the actual meaning of the words.

A recent work solves helpfulness prediction by interesting idea in combining with outer meta data as social context information (Lu and Tsaparas (2010) [21]). This approach is promising for the increasing of social relationships between reviewers. However, for general review portals, social context may be lacked or untrusted.

As semantic approaches, some of works are proposed recently. Martin and Pu (2014) [22] use a general lexicon of emotional words and derive the emotion-based helpful review prediction. Later, Yang et al. (2015) [23] show that helpful reviews are less emotional but could be converted to interpretable semantic features. They introduced two semantic features type: LIWC and General Inquirer (INQUIRER) for easy mapping from text to human sense, including emotions, writing styles. They also find that the models trained with semantic features are easier to be generalized to reviews of different domains and align well with human annotations.

Our study focuses on review quality as review helpfulness and proposes new semantic features set based on topic modeling for quality prediction and classification problem. Then we build a generalized framework with probabilistic approaching for mathematical formulation of our problem. Furthermore, our proposals are examined with real review datasets and implemented in self-built recommendation system.

2.3 Probabilistic Topics Modeling

There is a growing need to analyze large content of electronic documents. One of common tasks is topic modeling which organizes documents into groups of related underlaying semantic information called topics. Formally, a topic is defined as a probability distribution over terms in a vocabulary that represent the structure of generating a large number of words.

Probabilistic topic models find a low dimensional representation of document under the assumption that each word is generated from underlying topics. Understanding latent topics behind text is important in assessing document properties.

In this section, as dealing with review text, we firstly introduce the basic ideas of Latent Dirichlet Allocation (LDA) in Blei et al. (2003) [24], which is the most famous and most widely used topic model algorithm.

2.3.1 Latent Dirichlet Allocation

Latent Dirichlet Allocation (LDA) [24] is a Bayesian generative model that describes how the documents in a dataset were created. It is used as an unsupervised method to discover the underlying topics covered by a text document. LDA assumes that a corpus of text documents is just a collection of topics where each topic has some particular probability for generating a particular word. The particular probability is determined by looking at each training document as a "bag of words" from a distribution selected by Dirichlet process.

LDA Graphical Model

Traditional LDA can be represented by plate diagram (Figure 2-1) of graphical model that defines the pattern of conditional dependence between random variables. Unshaded and shaded circles display for latent random variables and observed random variables respectively. Edges represent dependencies between variables, and the rectangular plates indicate repetition. The parameters used in LDA model are summarized in table 2.1.

LDA generative model describes how each document obtained its words. Each topic β_i is defined as a multinomial distribution over a word dictionary with $|V|$ words drawn from a Dirichlet process $\beta_i \sim Dirichlet(\eta)$. The LDA generative process for a document d and number K of topics is described as following steps in [24].

LDA Generative Process

1. Randomly choose number of words N for document d from Poisson distribution: $N \sim Poisson(\lambda)$

Table 2.1: Parameters used in LDA model

Symbols	
α	Dirichlet parameter
η	Dirichlet parameter
θ_d	Topic proportion for d^{th} document
z_{dn}	Topic assignment of n^{th} word in d^{th} document
w_{dn}	n^{th} word in d^{th} document
β_k	Distribution of vocabularies in k^{th} topic
N	Number of words per document
D	Number of documents
K	Number of topics
V	Vocabulary set

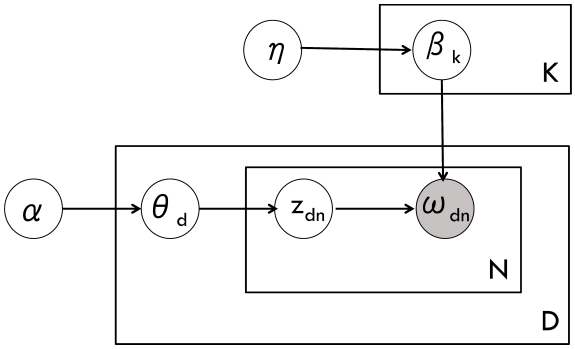


Figure 2-1: LDA plate diagram

2. Randomly choose a distribution over topics for document d from Dirichlet process: $\theta_{\mathbf{d}} \sim Dir(\alpha)$. $\theta_{\mathbf{d}}$ is the parameter for multinomial distribution.

3. For each n^{th} word in the document d:

   - Randomly choose topic z_{dn} from the distribution over topics: $z_{dn} \sim Multinomial(\theta_{\mathbf{d}})$
   - Randomly choose a word w_{dn} from one of $|V|$ words in the corresponding topic. The selected probability is defined as $p(w_{dn}|z_{dn}, \beta)$. β is a $K \times V$ matrix whose row $\beta_i \sim Dirichlet(\eta)$ and β_{ij} is the probability that j^{th} word in vocabulary assigned to topic i.

The generative process above for LDA corresponds to the following joint distribution of

the hidden and observed variables with its dependencies.

$$p(\boldsymbol{\beta}_{1:K}, \boldsymbol{\theta}_{1:D}, \boldsymbol{z}_{1:D}, \boldsymbol{w}_{1:D}) = \prod_{i=1}^{K} p(\boldsymbol{\beta}_i) \prod_{d=1}^{D} p(\boldsymbol{\theta}_d) \left(\prod_{n=1}^{N} p(z_{d,n}|\boldsymbol{\theta}_d) p(w_{d,n}|\boldsymbol{\beta}_{1:K}, z_{d,n} \right) \tag{2.1}$$

LDA Inference Process

Suppose we have a set of documents and some fixed number of K topics to discover and we do not know K topic distributions for our corpus. The LDA inference process learns the topic representation for each document and the words associated to each topic that best fit the corpus. The observed variable is the bag of words $\{w_{dn}\}$, we want to learn latent variables: $\boldsymbol{\beta}_k$ (distribution over vocabulary for topic k) and θ_{dk} (topic proportion of topic k in document d).

The inference problem returns to the problem of computing the conditional distribution of hidden factors given the observed documents (this is called posterior). The posterior of LDA is calculated as below.

$$p(\boldsymbol{\beta}_{1:K}, \boldsymbol{\theta}_{1:D}, \boldsymbol{z}_{1:D}|\boldsymbol{w}_{1:D}) = \frac{p(\boldsymbol{\beta}_{1:K}, \boldsymbol{\theta}_{1:D}, \boldsymbol{z}_{1:D}, \boldsymbol{w}_{1:D})}{p(\boldsymbol{w}_{1:D})} \tag{2.2}$$

The numerator in (2.2) is the joint distribution of all the random variables which can be computed easily from (2.1). The denominator is the marginal probability of the observations, which is the probability of seeing the observed corpus under any topic model. In theory, it can be computed by summing the joint distribution over every possible instantiation of the hidden topic structure. However, in practical, this sum is intractable[1].

The algorithms of topic modeling try to find an approximation form of (2.2) by finding an alternative distribution close to the true posterior over the latent topic space. This process could be implemented by Variational Bayesian approach in [24] [25], and collapsed Gibbs sampling approach in [26].

[1]The number of observed words in document collections is at least 10^3 order. Thus the sum which is over all possible ways of assigning each observed word to one of the topics is extremely hard to compute.

Chapter 3

Proposals

3.1 Hidden Topic Modeling Approach for Review Feature Extraction

The text of a stand alone review provides rich information about its quality. For example, Lu and Tsaparas (2010) [21] group the features for review's quality predictor into three different types.

Text-statistic features: The aggregate statistical features over the text, such as the review's length, the average length of a sentence, or the richness of the vocabulary.

Syntactic features: The statistical features based on the Part-Of-Speech (POS) tags of the words in the text, such as percentage of nouns, adjectives, punctuations, etc.

Sentiment features: The features that take into account the positive or negative sentiment of words in the review.

However, they are not enough to reveal the content of review which is the major factor to evaluate its quality. For example, consider two following reviews for a restaurant.

> **Review 1**: *This place is one of the best spots in this area. I came here as my first date. He was very sweet to me. We talked a lot about school, summer holidays,*

the newest movies of Tom Cruise. He's very intelligent and as he was speaking, I felt dizzy and hot. I could no longer focus on his words. I controlled myself to not say something stupid. It's wonderful day with me.

Review 2: *We ordered the omakase and truly enjoyed each dish and experienced topped off wagu beef that I'm still dreaming about! All the nigiri/sashimi was super tasty and fresh!*

We call these types of features proposed by Lu and Tsaparas in [21] as "explicit textual features". In the estimator using only explicit textual features, review 1 is evaluated with higher quality than review 2 for abundant words and plenty of positive opinions. However, review 1 mentions only about reviewer's boyfriend without useful information about restaurant. Review 2 with keywords like "omakase", "wagu beef", "nigiri/sashimi", "fresh", "tasty" is representative review that helps reader understand the characteristics of restaurant. Review 2 is better in this circumstance.

Our work bases on simple hypothesis that quality of a review depends on the content of this review. The reviews which discussed the same things or the same opinions about the same topics may have the same quality. A review could discuss one major topic or mixture of topics. It's more appropriate to use the distribution of topics in each individual review as features in quality evaluation. To figure out the distribution of topics in each individual review and in all reviews community, we use Latent Dirichlet Allocation (LDA) [24] method for topic modeling.

3.1.1 Extracting Features from Hidden Topics Distribution

As discussion of related works in chapter 2, we use LDA for process all reviews in dataset to produce K topics in which each topic is correspondence between word in vocabulary and

a numeric probability. Only nouns, adjective, adverb words of reviews are fed into corpus. The training process keeps a number of most frequent tokens for producing K topics.

In addition, another output of LDA is the topics distribution as K-dimensional vector for each review. However, to compare and make a distinct between two reviews which discuss the same topics with the same proportion, we make a modification for this output instead of using normalized features vector in LDA output. After training model, each word in this model will be displayed as K dimensional feature vector with element as probability for appearance of this word in corresponding topic. We sum up the values of all words in review for each dimension to construct representation of each review by histogram. The example of LDA output and our feature vector is illustrated in Figure 3-1.

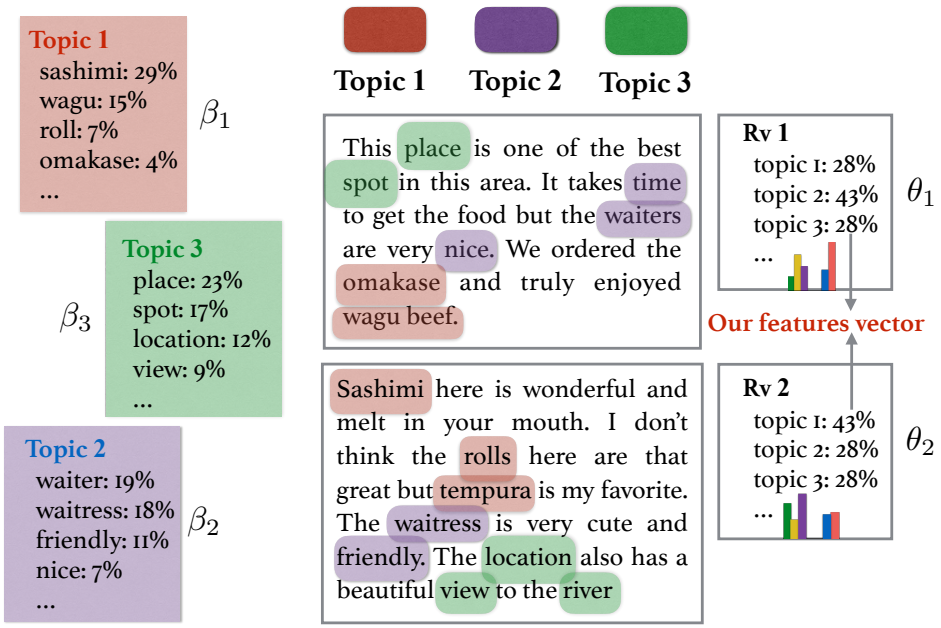


Figure 3-1: Illustration of LDA output: distribution of vocabulary in topics, topics assignment of each word in each review, and histogram of topic in each review as our feature vector

Figure 3-2 gives an illustration [1] for LDA output of Yelp Challenge Dataset [8]. The

[1] This illustration is made by using LDAvis tool. See details in [27]

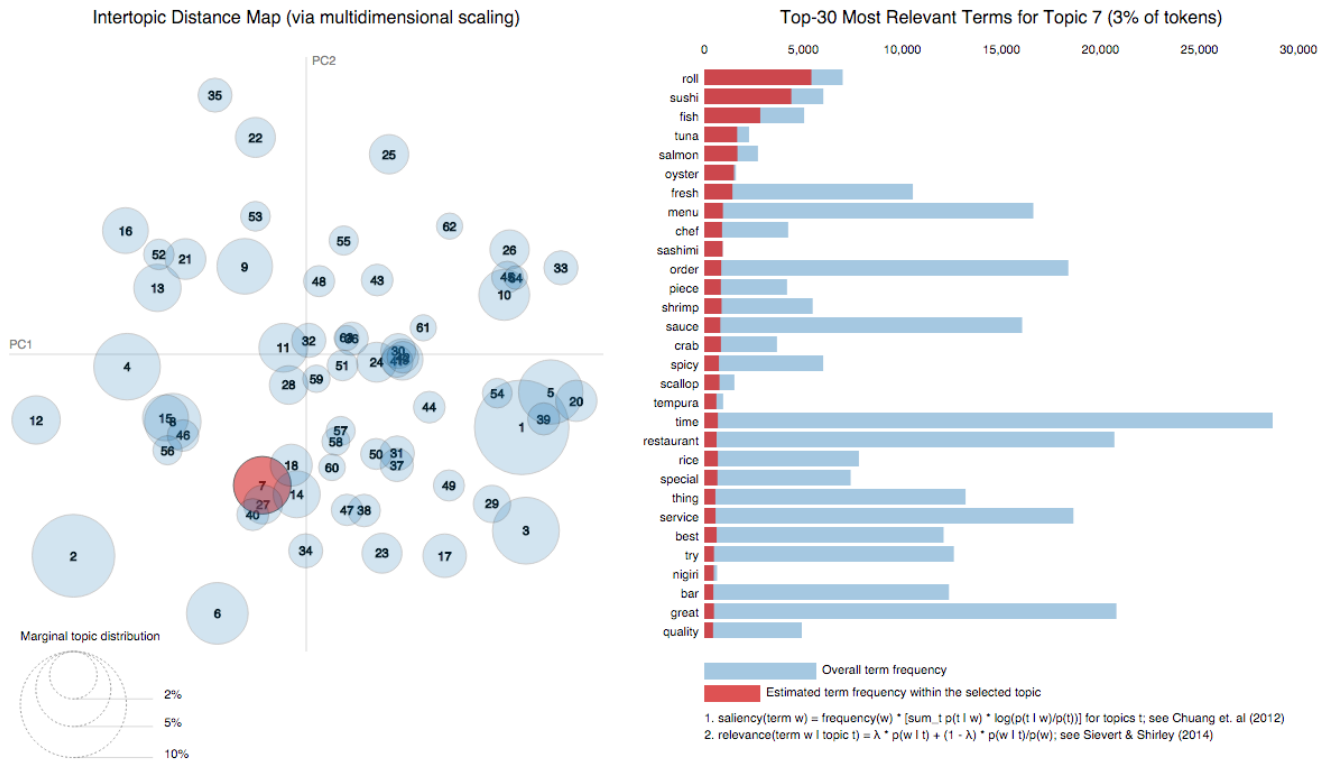


Figure 3-2: Illustration of LDA output for Yelp Challenge Dataset

left side of figure shows the topics as circles in the two-dimensional plane whose centers are determined by Jensen-Shannon divergence distance. The area of circle stands for the prevalence of topics, that is number of words from this topics by number of tokens in reviews collection. The right side is linked to the left side, showing top meaningful words in each topic. A pair of overlaid bars represents both the frequency of given term in selected topic and in all corpus. In this example, 64 topics are produced and as showing in figure, top 30 most relevant words for topic number 7 give us the general understanding about topic. This topic mentions about "Japanese food" with high evaluation as "best", "great", "special" appeared in the top-30 most relevant terms.

3.2 Probabilistic Modeling Approach for Review Quality Prediction

3.2.1 Problem Statement and Conventional Approaches

We consider the simple statement of predicting review quality as following.

Problem statement. *Given a dataset of online reviews associated with collection of readers' votes on these reviews, determine how "helpful" a new review would be to the users when no (or little) votes information available on this review.*

We denote the set of all available reviews as $R_I := \{r_i : i \in I\}$ where I is a finite indexing set and each r_i $(i \in I)$ is a review document. Each review r_i is associated with reader's vote set $V_i = \{v_i^{(1)}, v_i^{(2)}, ..., v_i^{(N_i)}\}$ where each $v_i^{(k)}$, is a random variable taking value of 0 (UNHELPFUL vote) or 1 (HELPFUL vote). In particular, V_i could be display as union of two disjoint subsets V_i^+ and V_i^- corresponding to the HELPFUL votes and UNHELPFUL votes on review r_i. Then, we can denote $V_I := \{V_i : i \in I\}$ and $V_I^+ := \{V_i^+ : i \in I\}$, $V_I^- := \{V_i^- : i \in I\}$.

The review quality prediction problem can be considered as determining how helpful an arbitrary review r, not necessarily in R_I given R_I, V_I^+ and V_I^-.

Conventional approaches (e.g. [21], [22], [18], [28]) define the quality of review r_i as the proportion of helpfulness votes among the total number of votes on this review.

Definition 1. *Quality of review r_i is defined as the proportion of helpfulness votes among the total number of votes for this review.*

$$d_i = \frac{|V_I^+|}{|V_I|} \tag{3.1}$$

Based on this measure of helpfulness, previous works have addressed the solution to problem on inference the dependency of helpfulness vote proportion with document r_i. The dependency is described as a function that best fit the data by training supervised data set R_I, V_I.

Despite promising results reported for several cases, this definition of review quality has a limitation due to insufficient statistics of voting data. Using only the helpfulness ratio from user feedback votes without considering the population of readers could not tell us how the votes for the review are produced. For example, "3 out of 10 people found the review helpful" statement is clearly less meaningful than "300 of 1000 people found the review helpful" although they have the same helpful votes ratio as 0.3.

In the next sub-section, we will propose more robust and probabilistic definition of review quality and more principle approach for review quality prediction.

3.2.2 Probabilistic Framework

We denote $\mathcal{R}$ as the space of all reviews, $\mathcal{H}$ is a family of vote function mapping $\mathcal{R}$ to $\{0, 1\}$. We define the quality of a review r, denoted by q(r), as the probability on $(\mathcal{R}, \mathcal{H})$ that one vote for this review to be a helpful vote.

Definition 2. *Quality of review r is defined as the probability that one vote for this review to be a helpful vote, namely,*

$$q(r) = Prob[\mathbf{H}(r) = 1] \tag{3.2}$$

where $\mathbf{H}$ *is drawn at random from* $\mathcal{H}$.

Equivalently, the quality of review r can be described by the conditional probability distribution $p_{\mathbf{V}|\mathbf{R}}(v|r)$ where $\mathbf{R}$ is a random variable draw from review set $\mathcal{R}$, and $\mathbf{V}$ is the $\{0, 1\}$-value target set of random vote function draw from $\mathcal{H}$. Clearly that we have following relation formula: $p_{\mathbf{V}|\mathbf{R}}(v = 1|r) = q(r)$ and $p_{\mathbf{V}|\mathbf{R}}(v = 0|r) = 1 - q(r)$

In our framework, we consider a family $\mathbf{\Phi}_{\mathbf{V}|\mathbf{R}}$ of conditional distribution of $\mathbf{V}$ given $\mathbf{R}$ that represent the dependency between review and reader's vote opinion. Let $\mathbf{R_I}$ be a random set of reviews drawn from $\mathcal{R}$ where $\mathbf{I}$ is a finite indexing set. $\mathbf{V_I}$, $\mathbf{V_I^{(+)}}$, $\mathbf{V_I^{(-)}}$ is re-used as definition from previous section. We also denotes the index set of votes in $\mathbf{V_I}$ as $\mathbf{J}(i)$. For convenience, the notations used in this section is summarized in Table 3.1.

Table 3.1: Parameters used in probabilistic model

Symbols	
$\mathcal{R}$	the space of all review texts
$\mathcal{F}$	the space of low dimensional feature vector
$\mathcal{H}$	family of votes function mapping $\mathcal{R}$ to {0, 1}
$\boldsymbol{R}$	random variable of review text
$\boldsymbol{I}$	the index set of all available reviews
r_i	review with index $i \in \boldsymbol{I}$
$\boldsymbol{F}$	random variable of feature vector
$\boldsymbol{f}_i$	represented feature vector of review with index $i \in \boldsymbol{I}$
$\boldsymbol{V}$	random variable of votes
$\boldsymbol{V_I}$	the set of all votes for all available reviews
$\boldsymbol{V_I}^+$	the set of all HELPFUL votes for all available reviews
$\boldsymbol{V_I}^-$	the set of all UNHELPFUL votes for all available reviews
$\boldsymbol{V}_i$	the set of all votes associated with review index i
$\boldsymbol{V}_i^+$	the set of all HELPFUL votes associated with review index i
$\boldsymbol{V}_i^-$	the set of all UNHELPFUL votes associated with review index i
$\boldsymbol{J}(i)$	the index set of all available votes associated with review index i
$v_i^{(k)}$	random variable taking value of 0 or 1 draw from $\boldsymbol{V}_i$ with $k \in \boldsymbol{J}(i)$
N_i	number of votes for review with index $i \in \boldsymbol{I}$
h_i	number of HELPFUL votes for review with index $i \in \boldsymbol{I}$

The problem of review quality inference leads to the process of choosing a optimal distribution $p^*_{\mathbf{V}|\mathbf{R}}(v|r)$ from the family $\mathbf{\Phi}_{\mathbf{V}|\mathbf{R}}$ that satisfying:

$$p^*_{\mathbf{V}|\mathbf{R}} = argmax_{p_{\mathbf{V}|\mathbf{R}} \in \mathbf{\Phi}_{\mathbf{V}|\mathbf{R}}} \log p_{\mathbf{V}|\mathbf{R}}(\mathbf{V_I}|\mathbf{R_I}) \tag{3.3}$$

To factorize it, we consider two following assumptions.

Assumption 1: *Voters opinion on one review is independent with opinions on other reviews.*

Assumption 2: *All votes in the same review are independent.*

Then (3.3) will become:

$$p^*_{\mathbf{V}|\mathbf{R}} = argmax_{p_{\mathbf{V}|\mathbf{R}} \in \mathbf{\Phi}_{\mathbf{V}|\mathbf{R}}} \log \prod_{i \in \mathbf{I}} \prod_{j \in \mathbf{J}(i)} p_{\mathbf{V}|\mathbf{R}}(v_i^{(j)}|r_i) \tag{3.4}$$

$$= argmax_{p_{\mathbf{V}|\mathbf{R}} \in \mathbf{\Phi}_{\mathbf{V}|\mathbf{R}}} \sum_{i \in \mathbf{I}} \sum_{j \in \mathbf{J}(i)} \log p_{\mathbf{V}|\mathbf{R}}(v_i^{(j)}|r_i) \tag{3.5}$$

In general, solving this optimal problem is infeasible due to huge dimensionality of space $\mathcal{R}$. The common technique is used in this kind of problem is reducing the dimensionality of $\mathcal{R}$ into a low dimensional feature space $\mathcal{F}$. In details, assume that we have a mapping function g from $\mathcal{R}$ to $\mathcal{F}$ that $g(r)$ is a feature vector of review $r \in \mathcal{R}$. Choice of g is discussed in section 3.1 that represent each review as combination vector of hidden topics proportion in this review.

Our optimal problem is then modified to finding

$$p^*_{\mathbf{V}|\mathbf{F}} = argmax_{p_{\mathbf{V}|\mathbf{F}} \in \mathbf{\Phi}_{\mathbf{V}|\mathbf{F}}} \sum_{i \in \mathbf{I}} \sum_{j \in \mathbf{J}(i)} \log p_{\mathbf{V}|\mathbf{F}}(v_i^{(j)}|\mathbf{f}_i) \tag{3.6}$$

where $\mathbf{\Phi}_{\mathbf{V}|\mathbf{F}}$ is family of conditional distribution of $\mathbf{V}$ given $\mathbf{F}$ that represent the dependency between review's features and reader's vote opinion.

And then again, we can re-make definition for review quality $q(r)$ of review r as

$$q(r) = p_{\mathbf{V}|\mathbf{F}}(v = 1|\mathbf{f}) = 1 - p_{\mathbf{V}|\mathbf{F}}(v = 0|\mathbf{f}) \tag{3.7}$$

where $\mathbf{f}$ is a feature vector in $\mathcal{F}$ associated with review r. We could re-write $p_{\mathbf{V}|\mathbf{F}}(v|\mathbf{f}) =$

$q(r)^v(1-q(r))^{1-v}$ and our optimal problem 3.6 as:

$$p^*_{\mathbf{V}|\mathbf{F}} = argmax_{p_{\mathbf{V}|\mathbf{F}} \in \mathbf{\Phi}_{\mathbf{V}|\mathbf{F}}} \sum_{i \in \mathbf{I}} \sum_{j \in \mathbf{J}(i)} \log \left\{ q_i^{v_i^{(j)}} (1-q_i)^{1-v_i^{(j)}} \right\} \tag{3.8}$$

$$= argmax_{p_{\mathbf{V}|\mathbf{F}} \in \mathbf{\Phi}_{\mathbf{V}|\mathbf{F}}} \sum_{i \in \mathbf{I}} \{h_i \log(q_i) + (N_i - h_i) \log(1-q_i)\} \tag{3.9}$$

where $q_i = q(r_i)$ is quality of review r_i defined in (3.7) and N_i, h_i are the number of votes and helpful votes for review r_i respectively.

3.2.3 Logistic Regression for Review Quality Prediction

For conventional definition of review quality, the state of the art for prediction task is using Support Vector Regression (SVR) method (e.g Kim et al. (2006) [18]; [19]; Yang et al. (2015) [23]). However, this regression method considers only the helpful votes ratio in cost function for optimization problem.

In this subsection, we consider regression method for predicting review quality q(r) and incorporating both number of helpful votes and population of all votes in our cost function. First of all, we notice that we can think review quality definition in (3.7) as the same numeric value with conventional definition in 3.2.1 if the number of votes for this review is high enough. Particularly, we use the reviews which have number of votes higher than or equal ten as supervised data and helpful votes fraction as target learning for our regression problem.

Because the quality of review defined in (3.7) is the quantitative probability function taking value in range [0, 1], then it could be displayed as the logistic function.

$$q(r) = p_{\mathbf{V}|\mathbf{F}}(v=1|\mathbf{f}) = logistic(z) = \frac{1}{1+\exp(-z)} \tag{3.10}$$

where z is expressed as linear combination from features space

$$z = \mathbf{w}^T \mathbf{f} \tag{3.11}$$

The objective of logistic regression problem is finding the weights vector $\boldsymbol{w}$ that is commonly shared with all review r_i in supervised dataset. If we can inference the common weights vector $\boldsymbol{w}$, we can calculate the quality of arbitrary review r from its features vector $\mathbf{f}$ by formula (3.10) and (3.11). Notice that in (3.11), we include bias term in parameter vector $\boldsymbol{w}$ by adding a additional constant element in feature vector $\boldsymbol{f}$.

The optimization solution $\boldsymbol{w}^*$ will be defined as argument that maximize the sum log-likelihood function:

$$\mathcal{L}(\boldsymbol{w}) = \sum_{i \in \mathbf{I}} \{h_i \log(q_i) + (N_i - h_i) \log(1 - q_i)\} \tag{3.12}$$

or the mean log-likelihood function:

$$\mathcal{E}(\boldsymbol{w}) = \frac{1}{|I|} \sum_{i \in \mathbf{I}} \{h_i \log(q_i) + (N_i - h_i) \log(1 - q_i)\} \tag{3.13}$$

$$\mathcal{E}(\boldsymbol{w}) = \frac{1}{|I|} \sum_{i \in \mathbf{I}} \left\{ h_i \log(\frac{1}{1 + \exp(-\mathbf{w}^T \mathbf{f}_i)}) + (N_i - h_i) \log(\frac{1}{1 + \exp(\mathbf{w}^T \mathbf{f}_i)}) \right\} \tag{3.14}$$

where $\boldsymbol{I}$ is index set of sample reviews in supervised dataset, and h_i, N_i are number of helpful votes and total votes for review r_i respectively.

Mini-Batch Stochastic Gradient Descent for finding optimization

There are several ways to doing optimization for (3.14). One of the most common algorithms is batch gradient descent method in which the value of the object function can be increased via updating the parameters follow the gradient direction.

$$\boldsymbol{w}^{(t+1)} = \boldsymbol{w}^{(t)} + \eta_t \frac{\delta \mathcal{E}}{\delta \boldsymbol{w}} (\boldsymbol{w} = \boldsymbol{w}^{(t)}) \tag{3.15}$$

where η_t is a choice of step size (called learning rate) at iteration t. This update equation gives a gradual increase in the log-likelihood.

The gradient $\frac{\delta \mathcal{E}}{\delta \boldsymbol{w}}$ can be easily computed by using (3.10), (3.11) and (3.13).

$$\frac{\delta \mathcal{E}}{\delta \boldsymbol{w}} = \frac{1}{|I|} \sum_{i \in I} \left\{ (\frac{h_i}{q_i}) \frac{\delta q_i}{\delta \boldsymbol{w}} - (\frac{N_i - h_i}{1 - q_i}) \frac{\delta q_i}{\delta \boldsymbol{w}} \right\} \tag{3.16}$$

$$= \frac{1}{|I|} \sum_{i \in I} \{h_i - N_i q_i\} \boldsymbol{f}_i \tag{3.17}$$

If I takes values as all index in training data, to compute gradient in (3.16), it takes $O(nd)$ as the time complexity where n is the number of training examples and d is dimension of feature vector. Batch gradient works well incase of convex or relatively smooth error manifolds. However, with error manifolds that have lots of local minimum and maximum points, batch gradient often leads to optimization process stack in local region.

To deal with these disadvantages of batch gradient descent algorithm, stochastic gradient descent (SGD) is proposed as computing the gradient using a single sample (in our problem is single review). In this case, single sample is somewhat noisy but it helps to move the function out of local maximum (minimum) region. However, in practical, for effective computation, instead of using a single sample, SGD use a mini-batch of several samples in training process. To make a trade-off between batch gradient descent and SGD, mini-batch is adopted to average the noise out while tends to move dynamically. The basic idea of mini-batch SGD algorithm is presented in algorithm 1.

In addition, to prevent overfitting we also impose squared L_2 norm as a penalty on the magnitude of the parameters. Then our mean log-likelihood function will become:

$$\mathcal{E}(\boldsymbol{w}) = \frac{1}{|I|} \sum_{i \in \mathbf{I}} \left\{ h_i \log(\frac{1}{1 + \exp(-\mathbf{w}^T \mathbf{f}_i)}) + (N_i - h_i) \log(\frac{1}{1 + \exp(\mathbf{w}^T \mathbf{f}_i)}) \right\} - C\|\boldsymbol{w}\|^2 \tag{3.18}$$

The constant C quantifies the trade-off between maximizing the log-likelihood and making parameters to be close to zero.

Algorithm 1: Mini-batch SGD algorithm

Input: Training data set with format of $(\boldsymbol{f}_i, h_i, N_i)$ for n reviews
Output: Parameter $\boldsymbol{w}$ for fitting logistic model
Initialize vector of parameters $\boldsymbol{w}$ and leaning rate η_0, mini-batch size m, epoch number t=0;
while *not converge* **do**
  update t = t+1 ;
  update η_t by some schedule (e.g. decaying after number of epochs);
  randomly shuffle examples in training dataset;
  divide training dataset into mini-batches ;
  for *each mini-batch* **do**
    takes I as the set of index values for all reviews in mini-batch ;
    update $\boldsymbol{w}$ by (3.14) and (3.15)
  end
end

3.3 Confident Models Combination

Each feature has advantage in predict and classify quality of review in some aspects. However, the non-linear dependency of review quality with its features makes the simple combination of features types (joining vector) not good. In this section, we propose a method from the predicting and classifying result of each feature type to get the most confident result as output in general.

The idea of confident models combination is using Median Absolute Deviation (MAD) [29] to remove outliers in different predicted result. For a univariate dataset, the MAD is defined as the median of the absolute deviations from data's median. The MAD is a robust statistic and being more resilient to outliers in dataset than the standard deviation [2].

For each review r, assume that we have m predicted output $X_1, X_2, ..., X_m$ for its quality correspond with m feature types. MAD is calculated as the formula below.

$$\begin{aligned} X = &\quad X_1, X_2, ..., X_m \\ MAD = &\quad median_i(|X_i - median_j(X_j)|) \end{aligned}$$

[2]In the standard deviation, the distances from the mean are squared so large deviations affect more in the value and then the outliers can heavily influence.

If the sample data follows normal distribution with standard deviation σ, as showing in Figure 3-3, 99% of sample data lie in the range of $[mean - 3\sigma, mean + 3\sigma]$. All of sample lie outside the range of $[mean - 2\sigma, mean + 2\sigma]$ or $[mean - 3\sigma, mean + 3\sigma]$ could be considered outlier points. Similar with this idea, the outlier detection method using MAD considered the points outside $[median(X) - k \times MAD(X), median(X) + k \times MAD(x)]$ as outlier points. Then, our confident output could be defined as following formula.

$$Confident(X_1, X_2, ..., X_m) = median(X_i \quad | \quad |X_i - median(X)| < k \times MAD(X)) \quad (3.19)$$

In our experiments k is set as value from 1 to 3, depending on the dataset.

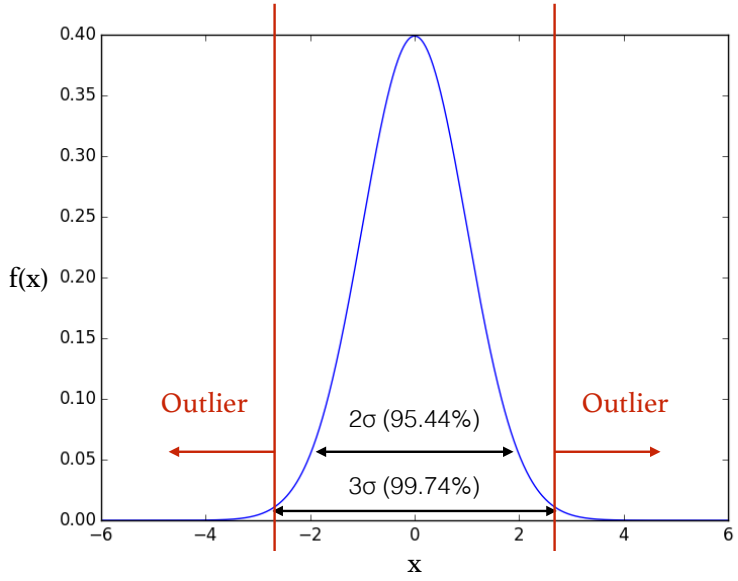


Figure 3-3: The outliers detection method using standard deviation

Chapter 4

Implemented System

In this section, we describe in general structure of our application. Further more, we focus on main part of this thesis's contribution - ReQE as our target system for processing and evaluating quality of review.

4.1 Application

Our application is a recommendation service with review portal built on the open-source machine learning server PredictionIO [30]. Our application has API for collecting and retrieving data, web-based interface and smartphone-based interface to interact with end-users. The general structure of application is described in Figure 4-1.

In our application, data are stored in MongoDB database and connect with PredictionIO server through Rail framework. However, main contribution of this thesis is backend part - ReQE connect with database for processing and evaluating quality of reviews.

4.2 Review quality evaluating system - ReQE

The detail of ReQE is described in Figure 4-2. ReQE solves two main tasks: review quality prediction and review quality classification. The first one includes quality definition and

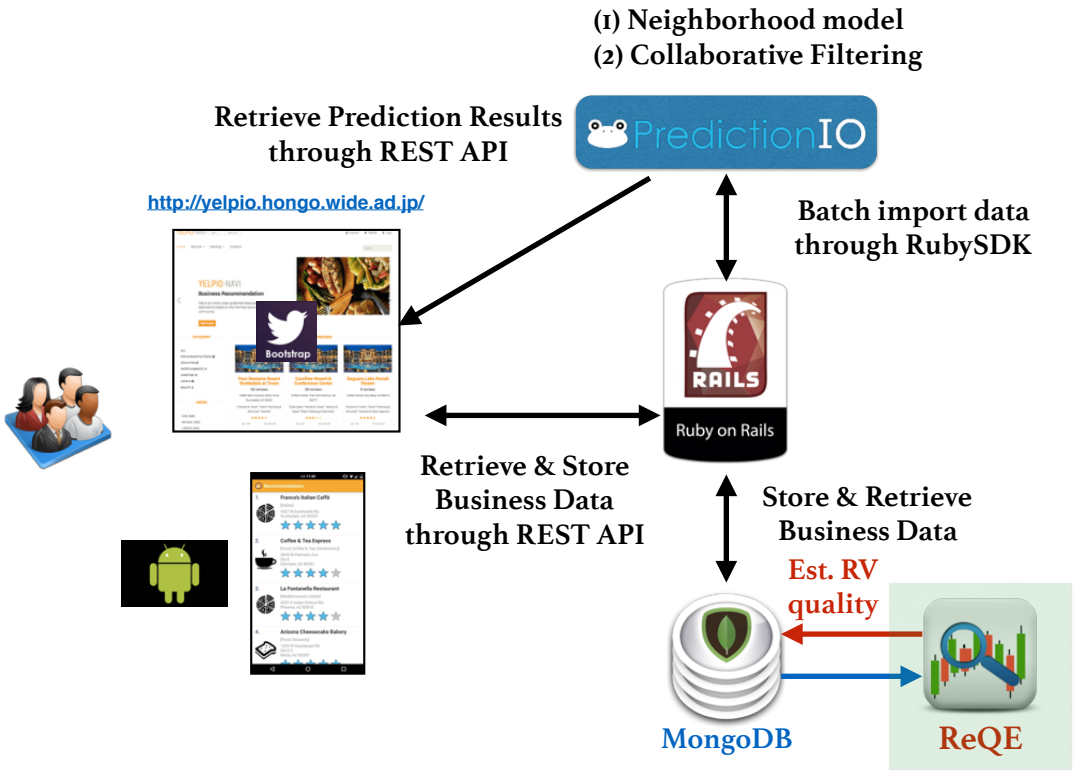


Figure 4-1: General structure of our application. Main contribution of this thesis is ReQE part for processing and evaluating quality of review.

quality numeric inference, the second one includes quality classify and class inference for the reviews that do not have enough number of votes for their helpfulness. The main idea of ReQE is using the reviews that do have sufficient number of votes as supervised data to inference quality of the reviews that do not have enough number of votes for their helpfulness. As showing in Figure 4-2, there are four main modules of ReQE which are the focus points in our proposals.

Module 1: Feature Extractor

Represent each review text as fixed-dimensional feature vector. We proposal the hidden topic distribution in each review as represented feature vector but other feature types could be used in this module (see 5.2).

Module 2: Quality Definition

Define what is review's quality. A conventional definition is discussed in 3.2.1 but we proposed more typical probabilistic approach in 3.2.2.

Module 3: Learning Model

Use the reviews that do have sufficient number of votes as supervised data to train model to predict or classify quality. Support Vector Regression is typically adopted in this module but our proposal focuses on logistic regression with incorporating of votes population in optimize function.

Module 4: Confident Inference Model

If the feature extractor produces multi types of feature, then corresponding with each feature type, each learned model in module 3 will produce different output. Module 4 plays the role of taking the most confident output when apply learned model to different type of features.

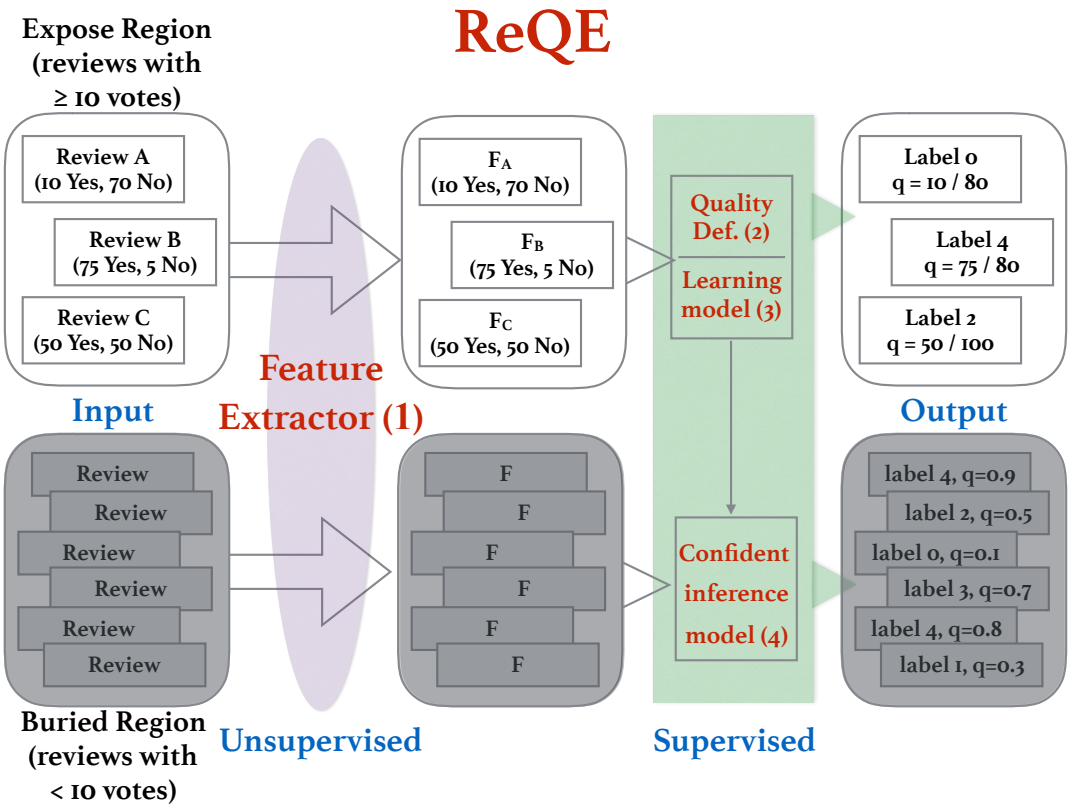


Figure 4-2: Details of ReQE - our proposed review quality evaluating system. Our proposals focus on four main modules: feature extractor (1), quality definition (2), learning model (3) and confident inference model (4).

Chapter 5

Experiments & Evaluation

In this chapter, firstly we describe the properties of dataset, set of extracted features, and then introduce experiments setup with evaluation methods. Finally, we go in details for discussion on performance of our proposals.

5.1 Dataset

We based our research on three datasets extracted from different product and service review websites. The first dataset contains reviews from Yelp, a social platform of restaurant reviews. The dataset is formed of 990,627 reviews extracted from the website for the Yelp Dataset Challenge [8]. The second one is crawled by Ganesan [31] from Trip Advisor, which is one of the largest social media platforms for hotel reviews. Finally, our last dataset is constructed from Amazon Review Dataset (McAuley, (2014)[32]), which includes nearly 143.7 million reviews spanning from May 1996 to July 2014.

In Amazon Review Dataset, a subset of 7,973,051 reviews from 14 categories: *Home_and_Kitchen, Sports_and_Outdoors, Automotive, Baby, Beauty, Office_Products, Clothing_Shoes_and_Jewerly, Toys_and_Games, Musical_Instruments, Pet_Supplies, Health_and_Personal_Care, Books, Electronics, Movies_and_TV*, are chosen in this research. For each category, we select the top 500 products with the most reviews and then include all reviews related to the selected

products for analysis.

In all the dataset, a review contains at least three basic information:

- Rating for the item (product, service,...).
- Review text.
- Helpful and unhelpful votes for the review itself (e.g: 20/30, 20 helpful votes out of 30 votes)

Quality of review can be approximated by using helpfulness rating if the number of votes for review is sufficiently high. In our experiments, the reviews with at least 10 votes form the automatic ground-truth labeled dataset. Table 5.1 shows some of the main characteristics of datasets.

Table 5.1: Statistics of the datasets

Dataset (category)	# reviews	# selected reviews ($\geq$ 10 votes)	Avg. of #words
Yelp	990,627	39,250 (3.96%)	152.62
TripAdvisor	240,060	40,741 (16.97%)	157.10
Home+Kitchen	581,889	23,627 (4.06%)	83.33
Sports+Outdoors	249,999	9,037 (3.61%)	72.10
Automotive	158,703	6,256 (3.94%)	66.61
Baby	253,334	7,371 (2.91%)	87.34
Beauty	265,114	11,316 (4.27%)	69.96
Office Products	257,430	13,634 (5.30%)	83.49
Clothing Shoes + Jewelry	358,619	7,686 (2.14%)	55.13
Toys + Games	249,177	9,077 (3.64 %)	75.50
Musical Instruments	119,580	6,522 (5.45%)	74.70
Pet Supplies	317,240	7,725 (2.44%)	81.73
Health + Personal Care	473,134	17,222 (3.64%)	73.59
Books	1,451,288	80,965 (5.58%)	80.45
Electronics	1,146,942	26,790 (2.34%)	73.46
Movies + TV	859,915	61,233 (7.12%)	80.04

5.2 Features

In this thesis, we only consider review text (without metadata) for evaluating review's quality. Thus, we focus on text-based features. Features used in previous related work, namely Structure and Syntactic (SaS) (Kim et al. (2006) [18]; Lu and Tsaparas (2010) [21]; Xiong and Litman (2011) [33]), Unigram tf-idf score (TF-IDF) (Kim et al. (2006) [18]; Xiong and Litman (2011) [33]; Agarwal et al. (2011) [34]), GALC emotion (Martin and Pu (2014) [22]), LIWC and General Inquirer (INQUIRER) (Yang et al. (2015) [23]) are considered as baselines. The characteristics of each feature type are described below.

SaS: the features are grouped in to three different types (Table 5.2)

- *Text-statistic features*: The aggregate statistical features over the text, such as the review's length, the average length of a sentence, or the richness of the vocabulary.
- *Syntactic features*: The statistical features based on the Part-Of-Speech (POS) tags of the words in the text, such as percentage of nouns, adjectives, punctuations, etc.
- *Sentiment features*: The features that take into account the positive or negative sentiment of words in the review.

Table 5.2: Structure and Syntactic Features (SaS)

Feature Name	Type	Feature Description
NumToken	Text-Stat	Total number of tokens
NumSent	Text-Stat	Total number of sentences
UniqWordRatio	Text-Stat	Ratio of unique words
SentLen	Text-Stat	Average sentence length
POS:NN	Syntactic	Ratio of nouns
POS:JJ	Syntactic	Ratio of adjectives
POS:COMP	Syntactic	Ratio of comparatives
POS:V	Syntactic	Ratio of verbs
POS:RB	Syntactic	Ratio of adverbs
POS:FW	Syntactic	Ratio of foreign words
POS:CD	Syntactic	Ratio of numbers
PosSEN	Sentiment	Ratio of positive words
NegSEN	Sentiment	Ratio of negative words

TF-IDF: We build a unigram vocabulary without stopwords and including the top 1024 frequent words from set of reviews. Then each review is represented as 1024 dimension features vector whose elements are *tf-idf* weighting for each appeared term in vocabulary.

GALC (Geneva Affect Label Coder) (Scherer (2005) [35]) proposes to recognize 38 affective states commonly distinguished by words in natural languages by parsing text databases for these words and their synonyms (searching for roots of the words). Similar to [22], we construct a feature vector with the number of occurrences of each emotion plus one additional dimension for non-emotional words. Thus our constructed feature vector will contain 39 elements.

Table 5.3: Geneva Affect Label Coder (GALC) example

State	Words
Amiration/Awe	admir*, ador*, awe*, dazed,dazzl*,..., wonder*, worship*
Anger	anger, angr*, cross*, enrag*, furious,..., wrath*, wrought*
...	...
Guilt	blame*, contriti*, guilt*, remorse*, repent*
...	...
Positive	agree*, excellent, fair, fine, good, nice, positiv*
Negative	bad, disagree*, lousy, negativ*, unpleas*

LIWC (Linguistic Inquiry and Word Count) (Pennebaker et al. (2007) [36]) is a dictionary which helps to determine the degree of feelings, personality and emotions of any text in language dimensions. Each word in LIWC is assigned 1 or 0 for each dimension. Similar to [23], we sum up the values of all words in review for each dimension to construct representation of each review by histogram. Dictionary used in our experiment is LIWC2007 English version which contains 4,476 words and 64 dimensional language.

INQUIRER (Stone et al. (1963) [37]) is a dictionary in which words are grouped in semantic categories. For example, *abnormal* is mapped to tags NEG and VICE. The dictionary contains 182 semantic TAGS and a total of 7,444 words. Similar to [23] and LIWC representation, we compute the histogram of each category (semantic TAG) for each review.

TOPICS: is our proposed feature vector. We construct a corpus for training by loops

Table 5.4: LIWC codeword-degree mapping

1	funct	17	preps	131	cogmech	149	sexual
2	pronoun	18	conj	132	insight	150	ingest
3	ppron	19	negate	133	cause	250	relativ
4	i	20	quant	134	discrep	251	motion
5	we	21	number	135	tentat	252	space
6	you	22	swear	136	certain	253	time
7	she/he	121	social	137	inhib	354	work
8	they	122	family	138	incl	355	achieve
9	ipron	123	friend	139	excl	356	leisure
10	article	124	humans	140	percept	357	home
11	verb	125	affect	141	see	358	money
12	auxverb	126	posemo	142	hear	359	relig
13	past	127	negemo	143	feel	360	death
14	present	128	anx	146	bio	462	assent
15	future	129	anger	147	body	463	nonfl
16	adverb	130	sad	148	health	464	filler

Table 5.5: LIWC dictionary example

Word	Code
abandon	125, 127, 130, 131, 137
abdomen*	146, 147
...	...
miss	11, 14, 125, 127, 130
...	...
zoloft	146, 148
zz*	463

Table 5.6: General Inquirer dictionary example

Word	Semantic Tag
alienate	Negativ, Affil, Hostile, SocRel
beseech	Negativ, Weak, Submi, Active
possess	Strong, Power
reject	Negativ, Ngtv, Hostile, Strong, Active, SocRel
sharp	Strong, Ovrst, Quality

through all the reviews in each dataset, split the reviews into tokens, remove stopwords and keep only noun, adjective and adverb words. We use *ldamodel* package in Gensim Python Library to train our model, keeping only the 10,000 most frequent tokens (but not appear at a half number of reviews) and using 64 topics. The core estimation code of *ldamodel* package is based on the *onlineldavb.py* script by M. Hoffman [25] using Variational Bayesian method and online learning approach. The reviews were processed in "batches" and the topic model was updated incrementally after processing each batch. After training model, each word in this model will be displayed as 64 dimensional feature vector with element as probability for appearance of this word in corresponding topic. We sum up the values of all words in review for each dimension to construct representation of each review by histogram.

Table 5.7: Statistics of the datasets

Feature type	Characteristics	Number of dimension	Training data dependency	Semantic meaning	Online training	Cross category usable
SaS	Structure & Syntactic	13	No	×	○	○
TF-IDF	Discriminitating tf-idf score	1024	Yes	×	×	×
GALC	Emotion	39	No	△	○	○
LIWC	Linguistic Inquiry (Psychological)	64	No	△	○	○
INQUIRER	Personal & critical	182	No	△	○	○
TOPICS	Latent topics distributions	64	Yes	○	○	△

ConfAll: is our proposed confident models combination for different feature types. The learning machine is applied to all types of features but taking the most confident value as the predicting output.

5.3 Experiments

5.3.1 Experiments setup

For our proposals, we are interested in the effectiveness of logistic regression algorithm and TOPICS features comparing with previous work. We perform two types of experiments: prediction (regression) and classification.

In regression task, the experiments try to use extracted features (including our proposal - TOPICS) to estimate review quality by logistic regression (our proposal) and compare with SVR (Support Vector Regressor) with RBF kernel.

In classification task, we define five categories of quality that represent the different helpful votes ratio's ranges from user feedback votes (Table 5.8). We use the Scikit-learn [38] and Chainer [39] framework in classifying reviews into these categories by different features types and different classify algorithms. In this thesis, to focus on proposed feature types, we used a neural network with two hidden layers (128 and 256 hidden units) in our classification task. The training is performed by Chainer framework with dropout technique [40] to prevent overfitting.

Table 5.8: Five categories of review quality

Review categories	**Helpful votes ratio r**
`Terrible`	$0 \leq r < 0.2$
`Bad`	$0.2 \leq r < 0.4$
`Normal`	$0.4 \leq r < 0.6$
`Good`	$0.6 \leq r < 0.8$
`Excellent`	$0.8 \leq r \leq 1$

For each dataset, we only consider subset of reviews which have number of votes higher than or equal 10. Because number of votes for each review in these dataset is sufficient, we could consider the helpfulness ratio as ground-truth data for our evaluation. Then we randomly divide this dataset into training set (80%) and test set (20%). Fifty random partitions are considered for all type of experiments.

5.3.2 Evaluation method

Performance of regression task is evaluate by Root Mean Square Error (RMSE) for regression task and accuracy score (percentage of true classified cases) for classification task. RMSE is defined as:

$$RMSE = \sqrt{\frac{1}{\mathcal{T}} \sum_{i \in \mathcal{T}} (q_i - t_i)^2} \tag{5.1}$$

where q_i is the predicted quality for review i and t_i is the ground-truth value, and $\mathcal{T}$ is the index set of test dataset.

Another useful evaluation metric is used here is Spearman's correlation coefficients for ranking performance. One of the most important output in processing reviews in real system is ranking review. Then, after each experiment of regression, the testing reviews are ranked according to their predicted quality score. The Spearman's correlation coefficient ρ is defined below.

$$\rho = 1 - \frac{6 \sum_{i \in \mathcal{T}} (x_i - y_i)^2}{|\mathcal{T}|(|\mathcal{T}|^2 - 1)} \tag{5.2}$$

where x_i is the rank of review i according to the quality predicted value by algorithm (logistic regression or SVR), and y_i is the rank of review i according to the helpful vote fraction of review i obtained from dataset (notice again that number of votes for each review in these dataset is sufficient (≥ 10), we could consider the helpfulness ratio as ground-truth data for our evaluation).

The average with each evaluation metric (as $RMSE$, $Spearman - corr$, $accuracy$) can be computed across all random partitions to obtain the overall performance of algorithm. In addition, we perform t-test (with significant level is 0.05) to determine an algorithm is significantly better than another algorithm or not. The details of this test could be found in appendix A.

For the meaning of RMSE (lower is better) and Spearman's correlation score (higher is better), we could summary evaluation method for comparison between logistic regression and SVR in Table 5.9.

Table 5.9: Hypothesis score (significant level = 0.05) between Logistic Regression (LGR) and Support Vector Regression (SVR)

Metric (m)	$\bar{m}(LGR) > \bar{m}(SVR)$	$\bar{m}(LGR) = \bar{m}(SVR)$	$\bar{m}(LGR) < \bar{m}(SVR)$
`RMSE`	-1	0	1
`Spearman-Corr`	1	0	-1

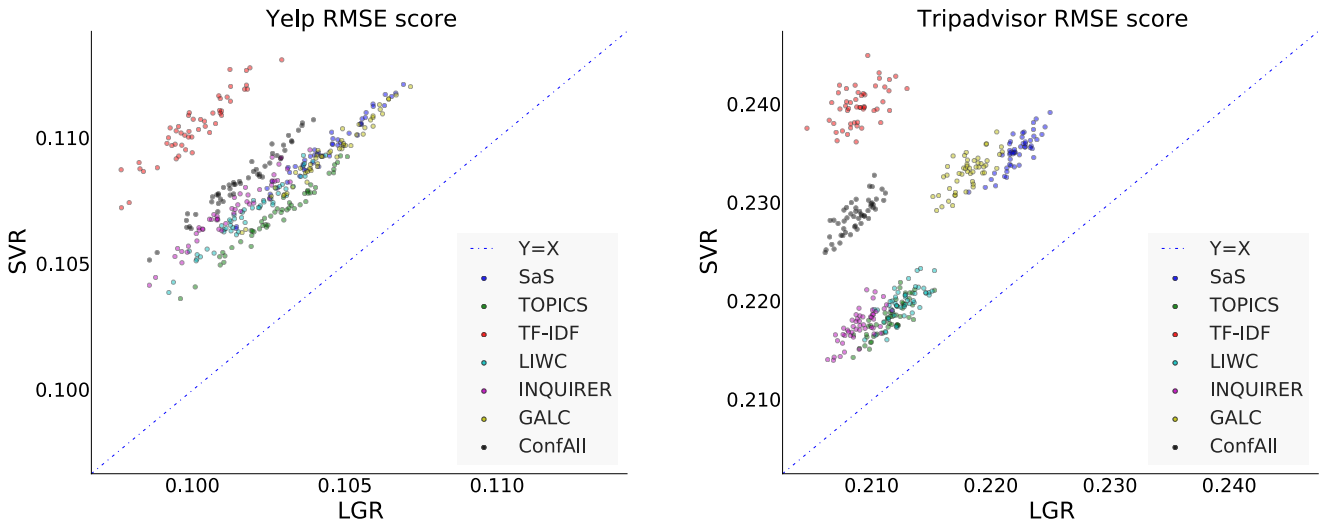
5.3.3 Performance of logistic fitting model

To verify the performance of our proposal - logistic fitting model, we examine the evaluation metric of 50 experiments for each dataset, and each feature type. We compare the evaluation metric of Logistic Regression (LGR) with Support Vector Regression (SVR). In each pair of dataset and feature type, we perform a grid-search for finding good parameter for prediction model.

Figure 5-1, **??**, 5-3, 5-4 show a set of scatter plots that compare RMSE score (smaller is better) between proposed LGR and SVR. Each point in each plot corresponds to the one experiment with pair of dataset and feature type. In most of datasets, the points are above y=x diagonal line thus showing a performance advantage of LGR over SVR.

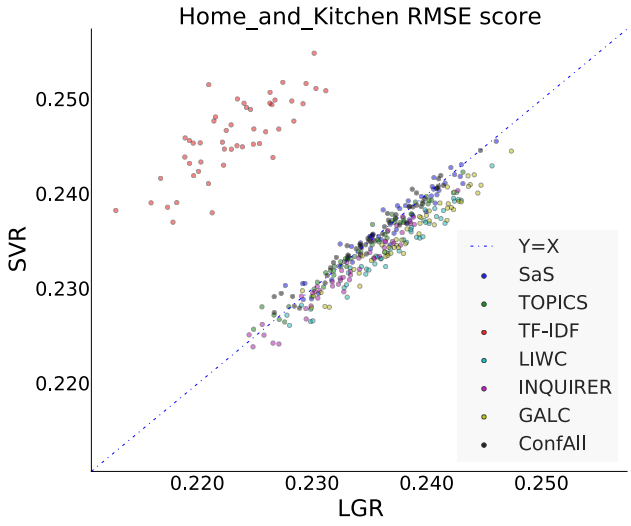
Figure 5-5, 5-6, 5-7, 5-8 show a set of scatter plots that compare Spearman score (higher is better) between proposed LGR and SVR. Each point in each plot corresponds to the one experiment with pair of dataset and feature type. In most of datasets, the points are below y=x diagonal line thus showing a performance advantage of LGR over SVR.

Figure 5-9 shows a heat-map represent hypothesis score between LGR and SVR in each evaluation metric and each pair of feature and dataset (WHITE means LGR = SVR, BLUE means LGR is better than SVR, RED means SVR is better than SVR in our hypotheses with significant level = 0.05). As we can see in this heat-map, in each evaluation metric and each pair of feature and dataset, LGR has a better performance over SVR.

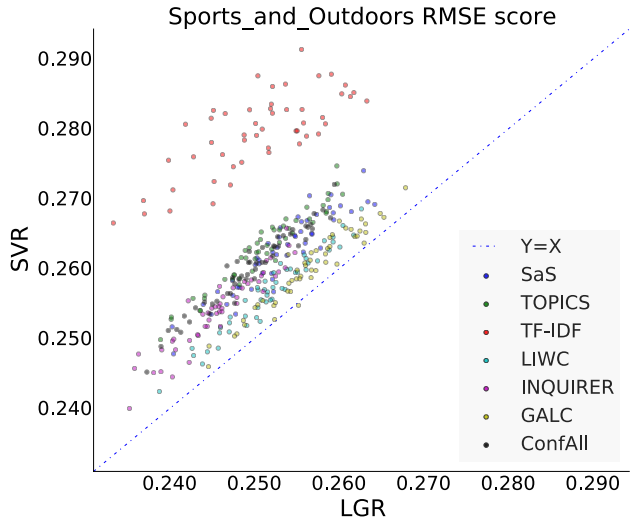


(a) Yelp

(b) TripAdvisor

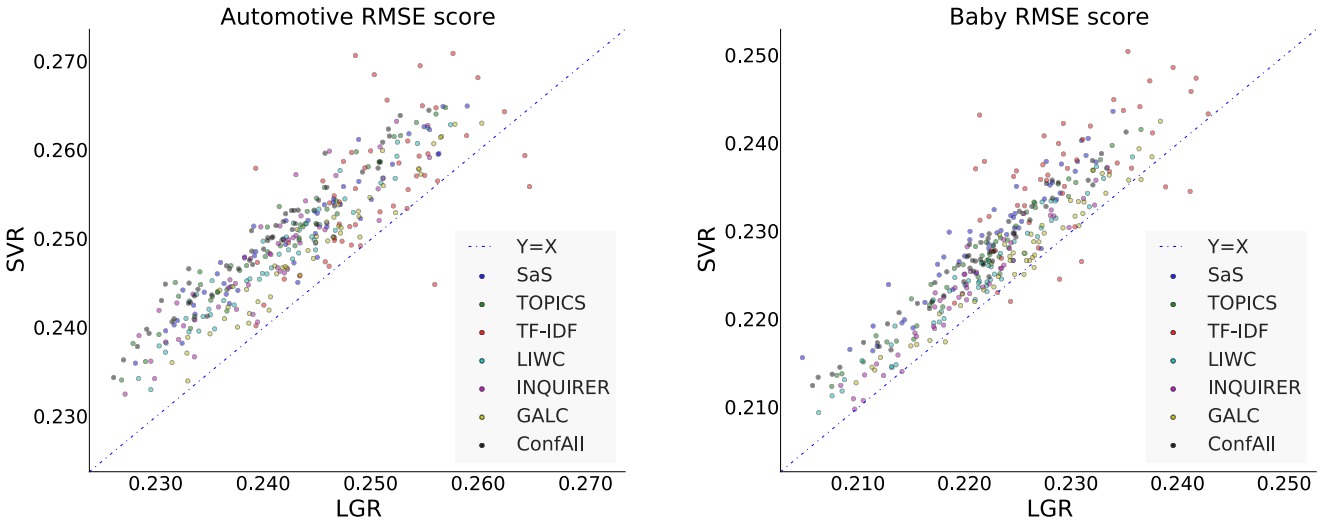


(c) Home and Kitchen



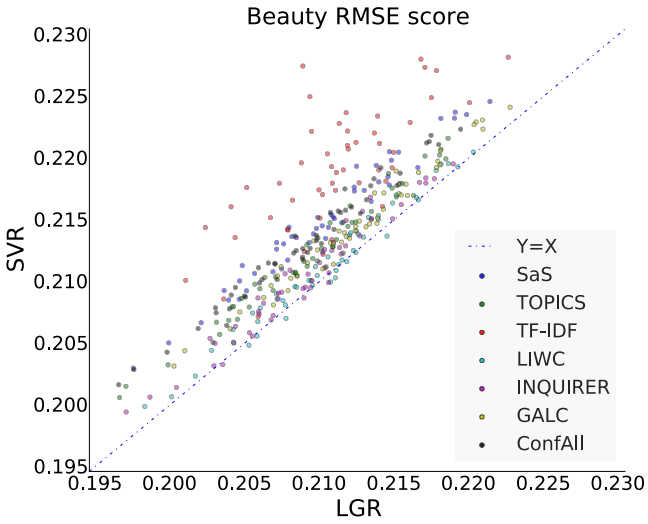
(d) Sports and Outdoors

Figure 5-1: (A) Scatter plot of experiments RMSE score in comparison between LGR and SVR. If the data point is above y=x line, LGR is better and vice versa.

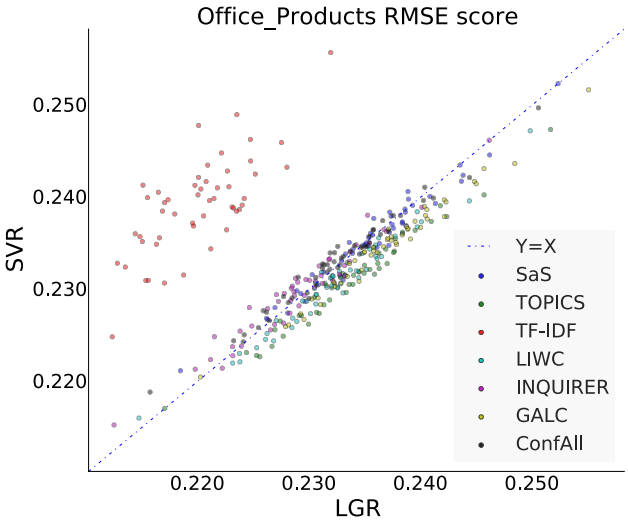


(a) Automotive

(b) Baby

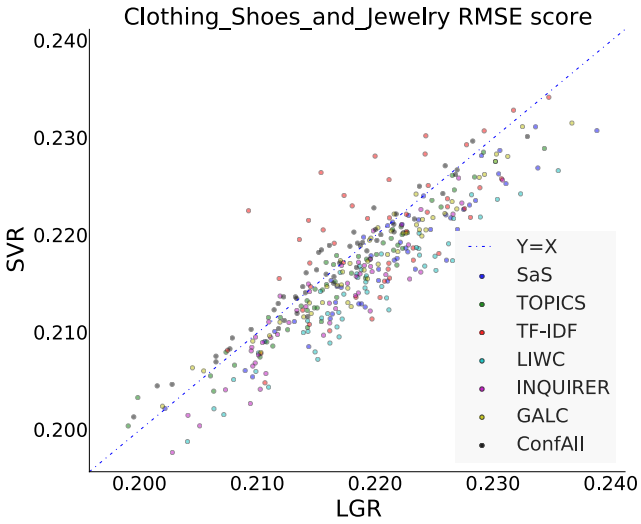


(c) Beauty

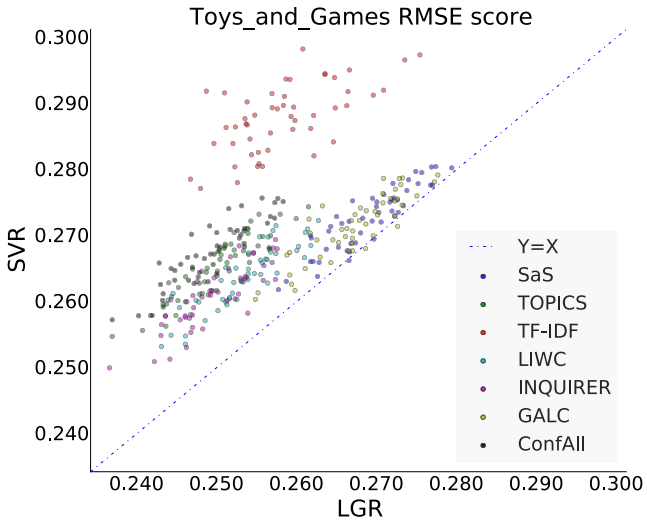


(d) Office Products

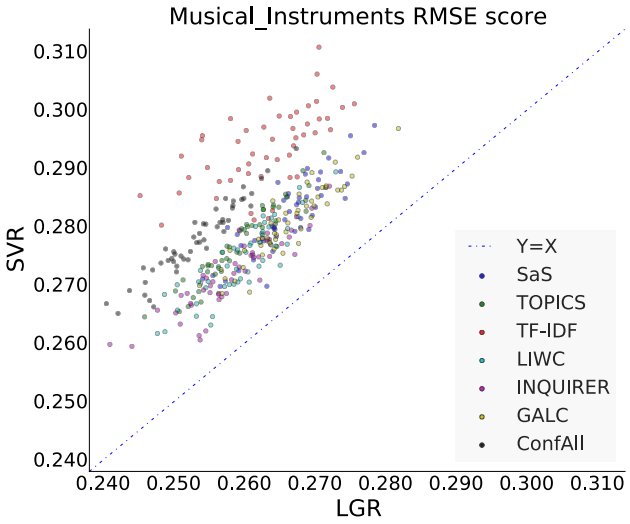
Figure 5-2: (B) Scatter plot of experiments RMSE score in comparison between LGR and SVR. If the data point is above y=x line, LGR is better and vice versa.



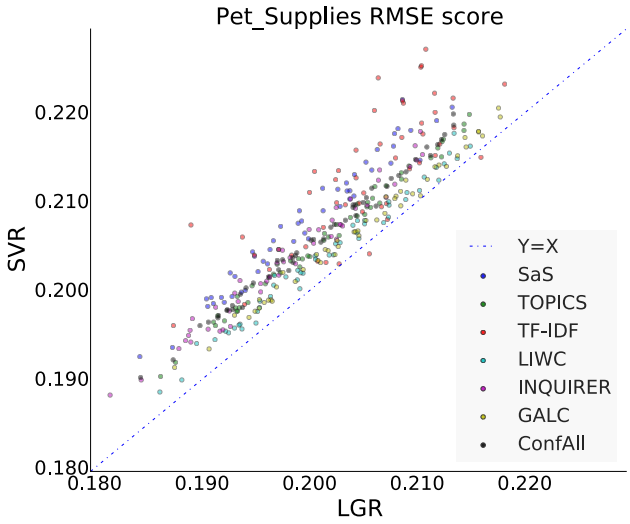
(a) Clothing Shoes and Jewelry



(b) Toys and Games

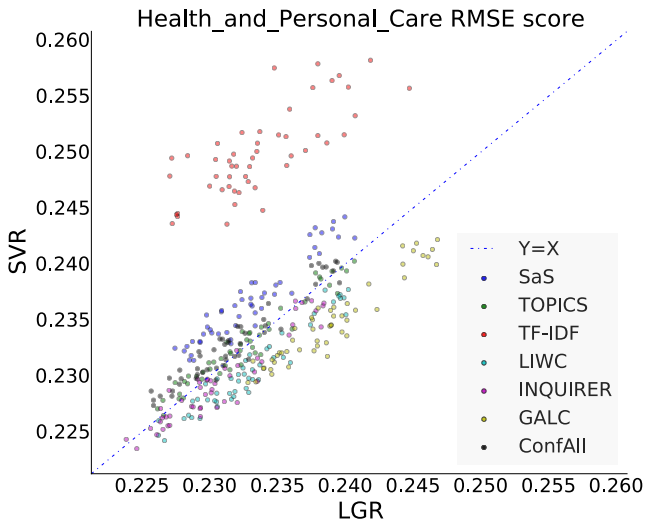


(c) Musical Instruments

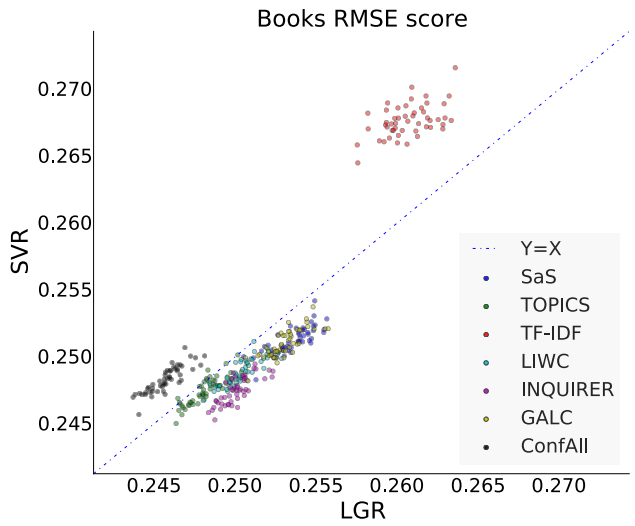


(d) Pet Supplies

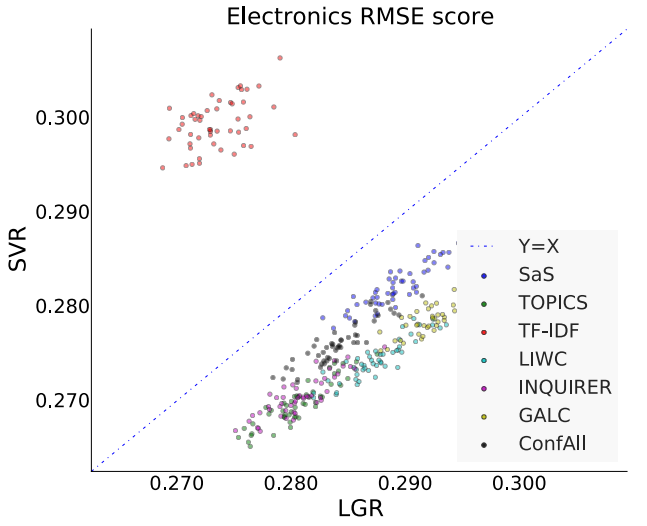
Figure 5-3: (C) Scatter plot of experiments RMSE score in comparison between LGR and SVR. If the data point is above y=x line, LGR is better and vice versa.



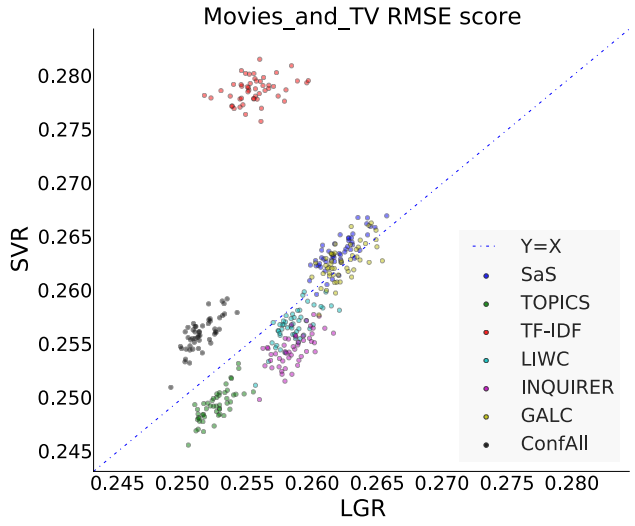
(a) Health and Personal Care



(b) Books

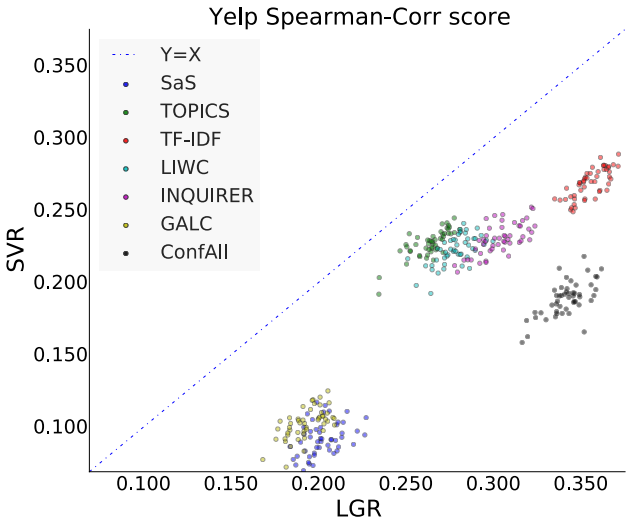


(c) Electronics

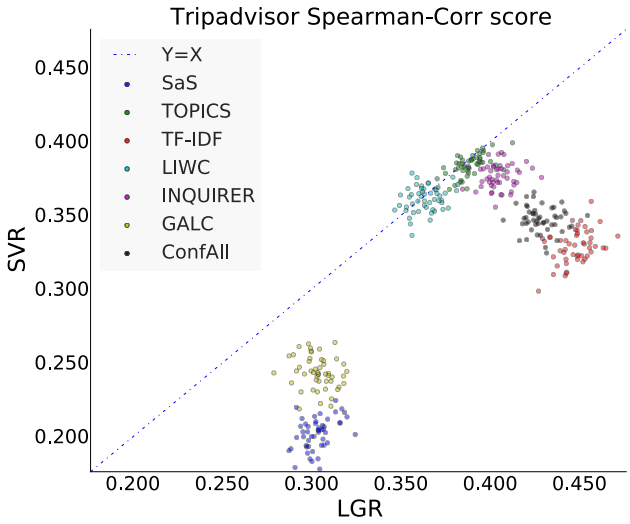


(d) Movies and TV

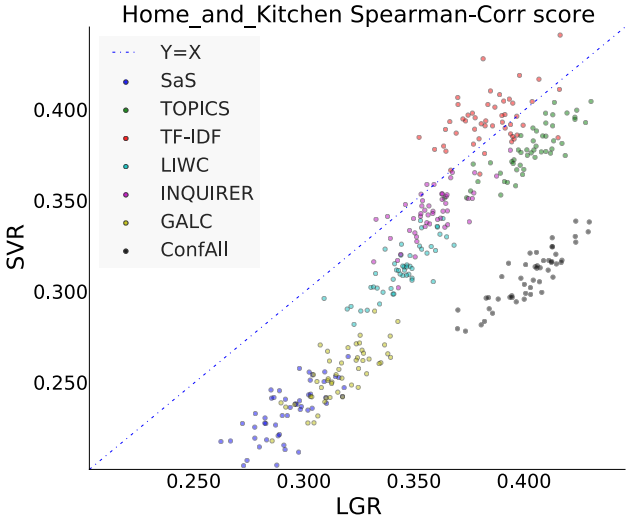
Figure 5-4: (D) Scatter plot of experiments RMSE score in comparison between LGR and SVR. If the data point is above y=x line, LGR is better and vice versa.



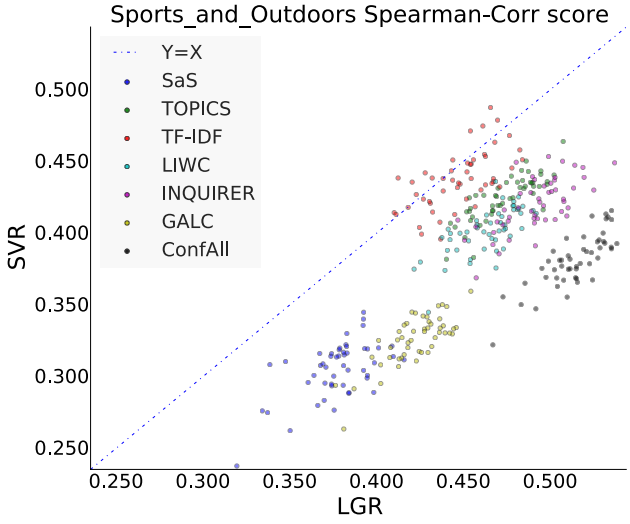
(a) Yelp



(b) TripAdvisor

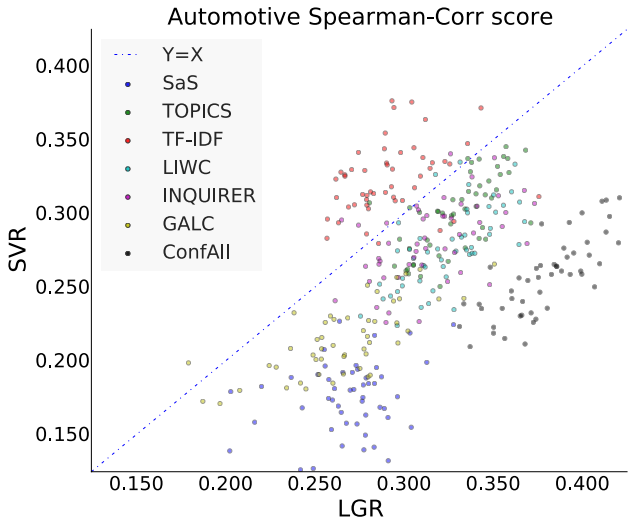


(c) Home and Kitchen

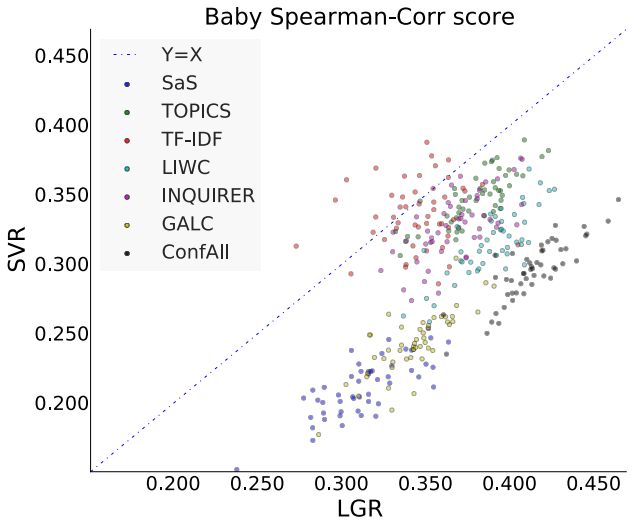


(d) Sports and Outdoors

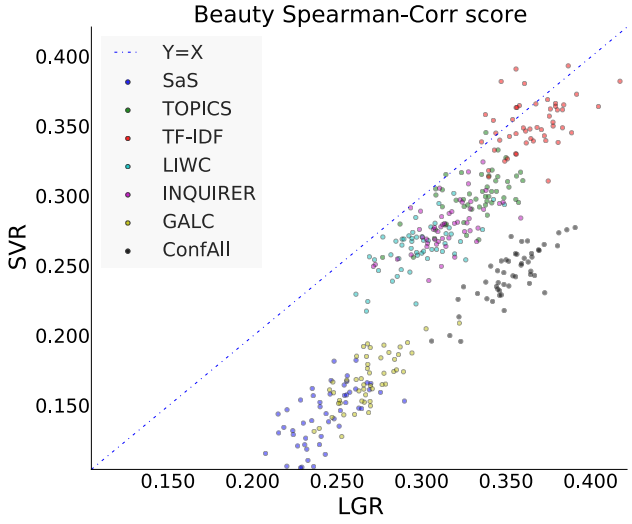
Figure 5-5: (A) Scatter plot of experiments Spearman-corr score in comparison between LGR and SVR. If the data point is below y=x line, LGR is better and vice versa.



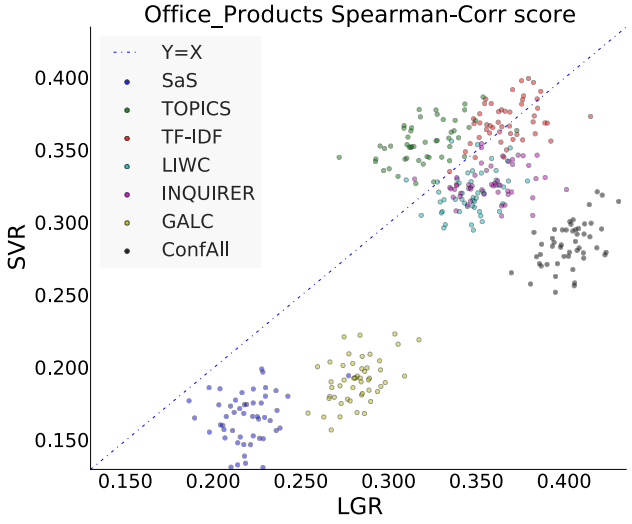
(a) Automotive



(b) Baby

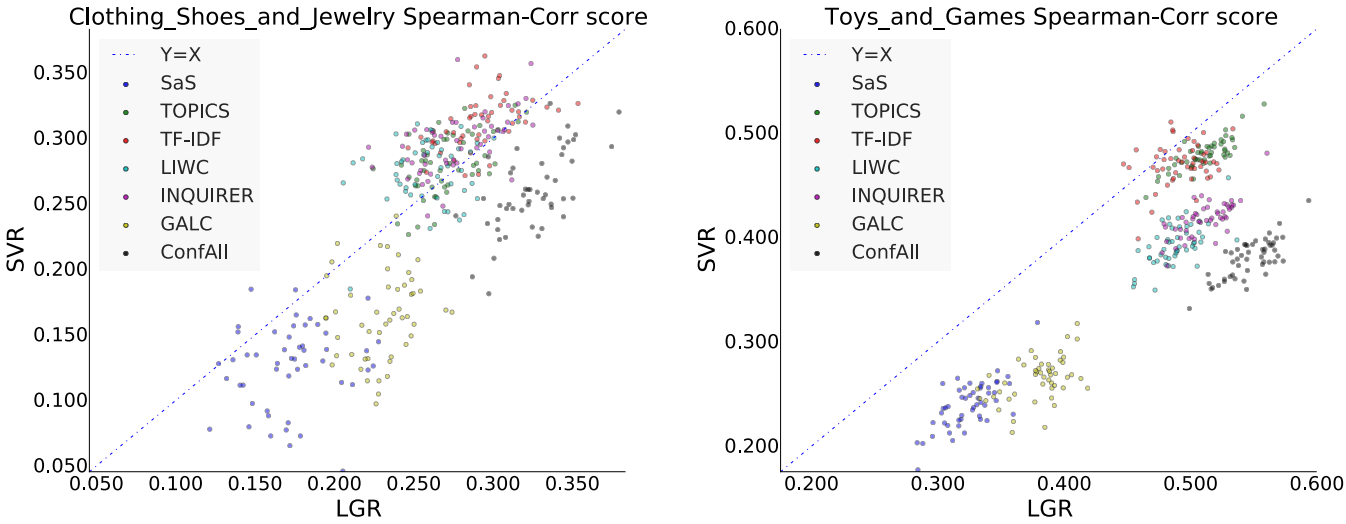


(c) Beauty



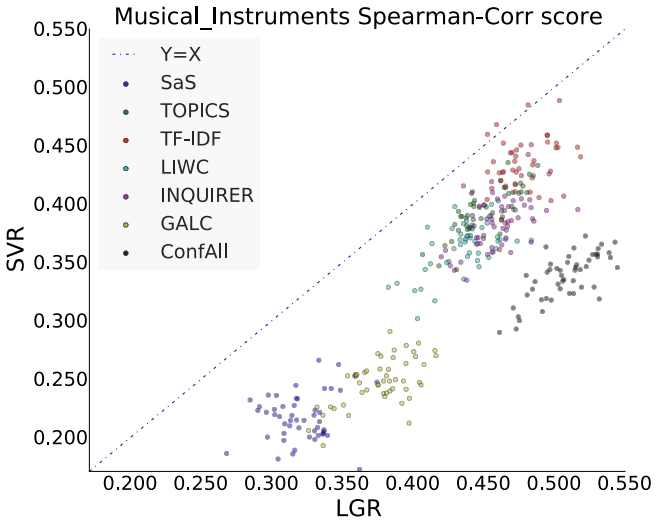
(d) Office Products

Figure 5-6: (B) Scatter plot of experiments Spearman-corr score in comparison between LGR and SVR. If the data point is below y=x line, LGR is better and vice versa.

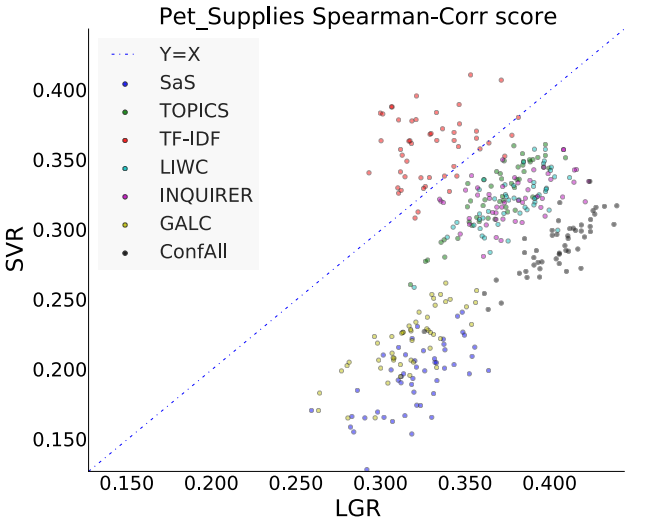


(a) Clothing Shoes and Jewelry

(b) Toys and Games

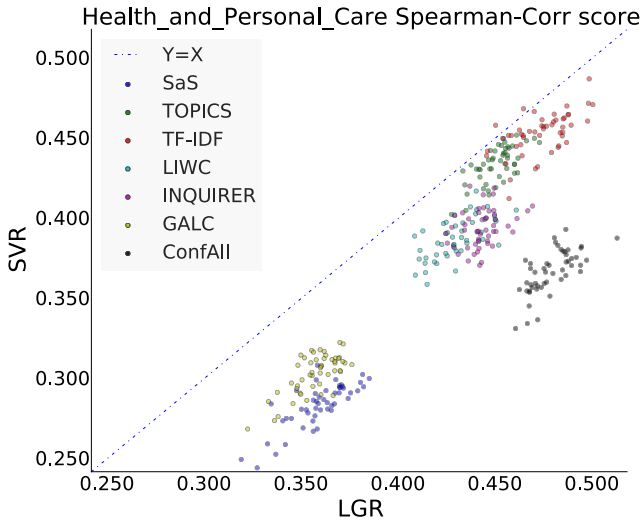


(c) Musical Instruments

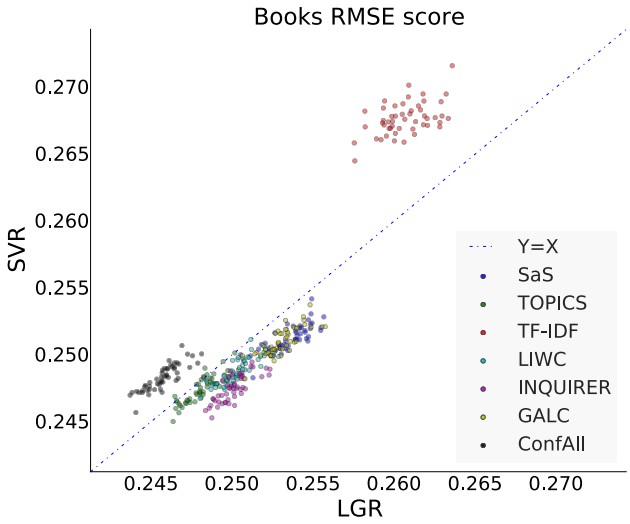


(d) Pet Supplies

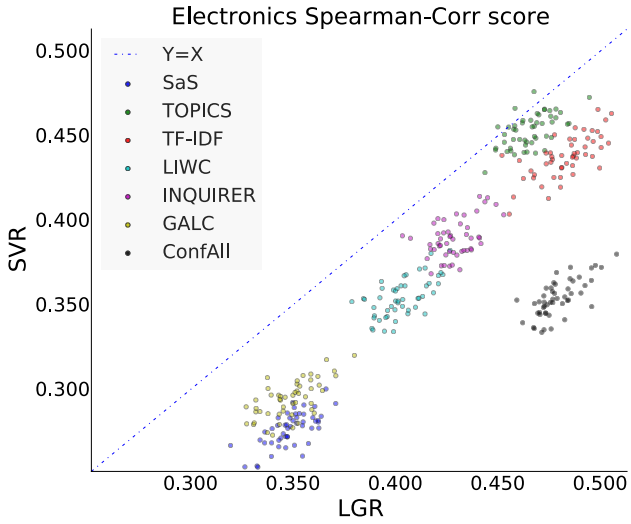
Figure 5-7: (C) Scatter plot of experiments Spearman-corr score in comparison between LGR and SVR. If the data point is below y=x line, LGR is better and vice versa.



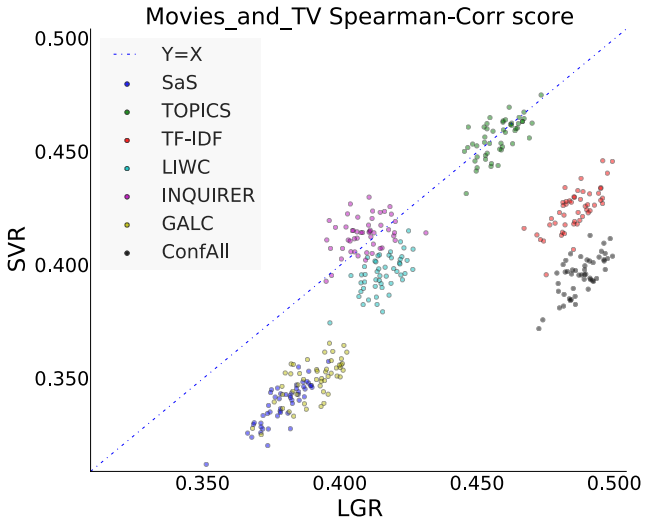
(a) Health and Personal Care



(b) Books

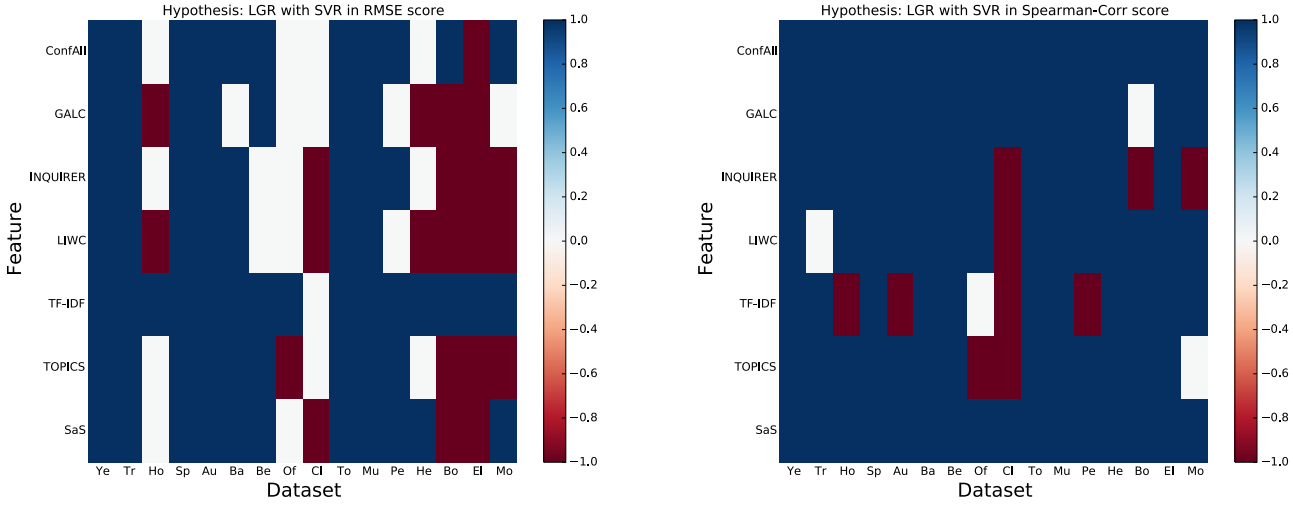


(c) Electronics



(d) Movies and TV

Figure 5-8: (D) Scatter plot of experiments Spearman-corr score in comparison between LGR and SVR. If the data point is below y=x line, LGR is better and vice versa.



(a) RMSE

(b) Spearman correlation

Figure 5-9: Heat-map of hypothesis score between LGR and SVR in each evaluation metric and each pair of feature and dataset (WHITE means LGR and SVR cannot be distinguished, BLUE means LGR is better than SVR, RED means SVR is better than SVR in our hypotheses with significant level = 0.05). Our proposed LGR gives the better performance in most of datasets and feature types.

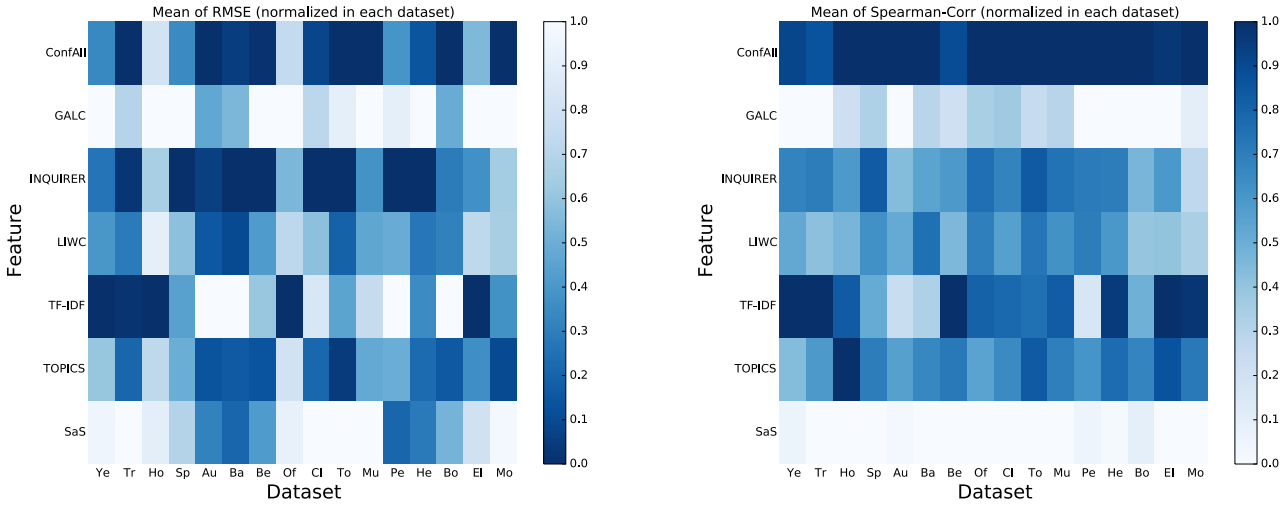
5.3.4 Performance of proposed features and confident model combination

RMSE and Spearman correlation coefficient mean score of 50 experiments using our proposed logistic regression are given in Table 5.10 and Table 5.11 respectively. Each column corresponds to the model trained by a feature or confident model combination of features, while each row corresponds to one dataset (or category). The lowest RMSE and highest Spearman-corr achieved in each row (each dataset) are marked in bold.

To make more details in comparison between features, Figure 5-10 shows a heat-map of evaluation metric(RMSE and Spearman-corr) in Table 5.10 and Table 5.11 for each pair of dataset and feature in our proposed logistic regression results. In each dataset, the evaluation value is normalized from [min, max] into [0, 1] range. Smaller RMSE is better and higher Spearman-corr is better thus BLUE is better in our color bar.

In prediction task, our proposed TOPICS feature has 0.97% higher RMSE on average than INQUIRER, and 2.45% lower correlation coefficient than TF-IDF, the best baseline features in each evaluation metric. However, TOPICS feature makes fairly trade-off evaluation in compare with other baseline features. In addition, our confident model combination of features has the best (lowest) overall RMSE and (highest) Spearman correlation coefficient score on average.

In classification task, the mean accuracies for fifty experiments setup of feature type and dataset are summarized in Table 5.12. Our proposed confident model combination of features has the best (highest) accuracy over all dataset with average is 62.5%, while our proposed TOPICS feature has the second best performance on average (61.1%). We also see that the classification algorithm with any features is not good for Books, Movies and TV dataset (about $34 - 39\%$). Unlike other dataset, these datasets have a broad range in products and review's quality depends highly on reader's personal favor more than profound content or vocabulary richness of review text. These characteristics give the problem of designing good features, especially in TOPICS feature cause many of words in each topic are character names in books or in movies.



(a) RMSE

(b) Spearman correlation

Figure 5-10: Heat-map of evaluation metric (RMSE and Spearman-corr) for each pair of dataset and feature in our proposed logistic regression results. In each dataset, the evaluation value is normalized from [min, max] into [0, 1] range. Smaller RMSE is better and higher Spearman-corr is better thus BLUE is better in our color bar. Our proposed TOPICS features give fairly good performance while our confident model combination of features produced the best performance in most of datasets.

Table 5.10: Logistic Regression Results in RMSE score (smaller is better)

Dataset	SaS	TOPICS	TF-IDF	LIWC	INQUIRER	GALC	ConfAll
Yelp	0.1044	0.1029	**0.1002**	0.1019	0.1013	0.1046	0.1017
TripAdvisor	0.2223	0.2116	0.2091	0.2127	0.2091	0.2182	**0.2088**
Home Kitchens	0.2370	0.2344	**0.2232**	0.2372	0.2334	0.2386	0.2357
Sports	0.2534	0.2513	0.2507	0.2522	**0.2460**	0.2566	0.2497
Automotive	0.2432	0.2413	0.2505	0.2415	0.2405	0.2448	**0.2398**
Baby	0.2224	0.2220	0.2298	0.2214	**0.2205**	0.2255	0.2210
Beauty	0.2105	0.2096	0.2112	0.2105	**0.2091**	0.2125	**0.2091**
Office Product	0.2357	0.2337	**0.2202**	0.2323	0.2293	0.2370	0.2327
Clothing	0.2219	0.2172	0.2209	0.2194	0.2159	0.2202	**0.2164**
Toys	0.2692	0.2505	0.2586	0.2534	**0.2496**	0.2674	**0.2496**
Musical Instrument	0.2665	0.2597	0.2633	0.2596	0.2585	0.2665	**0.2535**
Pet Supplies	0.1996	0.2016	0.2051	0.2016	**0.1982**	0.2045	0.2009
Health	0.2332	0.2327	0.2338	0.2331	**0.2309**	0.2391	0.2320
Books	0.2537	0.2482	0.2608	0.2505	0.2502	0.2532	**0.2458**
Electronics	0.2892	0.2807	**0.2736**	0.2878	0.2809	0.2932	0.2843
Movies and TV	0.2622	0.2527	0.2557	0.2587	0.2586	0.2625	**0.2516**
Average	0.2328	0.2281	0.2292	0.2296	**0.2270**	0.2340	**0.2270**

Table 5.11: Logistic Regression Results in Spearman-Corr score (higher is better)

Dataset	SaS	TOPICS	TF-IDF	LIWC	INQUIRER	GALC	ConfAll
Yelp	0.2031	0.2648	**0.3561**	0.2791	0.3045	0.1938	0.3427
TripAdvisor	0.3039	0.3893	**0.4484**	0.3644	0.4055	0.3044	0.4295
Home Kitchens	0.2940	**0.4034**	0.3854	0.3450	0.3589	0.3178	0.4030
Sports	0.3784	0.4749	0.4485	0.4644	0.4926	0.4231	**0.5152**
Automotive	0.2696	0.3305	0.2933	0.3253	0.3164	0.2668	**0.3798**
Baby	0.3143	0.3810	0.3466	0.3898	0.3691	0.3438	**0.4146**
Beauty	0.2438	0.3320	**0.3664**	0.2998	0.3166	0.2696	0.3532
Office Product	0.2201	0.3217	0.3679	0.3478	0.3592	0.2826	**0.4036**
Clothing	0.1734	0.2725	0.2928	0.2594	0.2775	0.2305	**0.3258**
Toys	0.3257	0.5120	0.4933	0.4889	0.5121	0.3809	**0.5482**
Musical Instrument	0.3222	0.4527	0.4777	0.4389	0.4614	0.3784	**0.5093**
Pet Supplies	0.3216	0.3704	0.3313	0.3789	0.3795	0.3172	**0.4045**
Health	0.3601	0.4517	0.4747	0.4311	0.4441	0.3581	**0.4800**
Books	0.3246	0.3674	0.3536	0.3467	0.3520	0.3180	**0.3908**
Electronics	0.3495	0.4665	**0.4839**	0.4037	0.4294	0.3493	0.4800
Movies and TV	0.3798	0.4583	0.4862	0.4160	0.4095	0.3911	**0.4884**
Average	0.2990	0.3906	0.4004	0.3737	0.3868	0.3203	**0.4293**

Table 5.12: Accuracy in classification task using neural network (higher is better)

Dataset	SaS	TOPICS	TF-IDF	LIWC	INQUIRER	GALC	ConfAll
Yelp	0.580	0.601	0.583	0.609	0.606	0.580	**0.612**
TripAdvisor	0.670	0.672	0.640	0.673	0.674	0.672	**0.691**
Home Kitchens	0.749	0.748	0.720	0.748	0.746	0.749	**0.758**
Sports	0.650	0.649	0.621	0.649	0.641	0.647	**0.661**
Automotive	0.711	0.706	0.681	0.707	0.686	0.710	**0.721**
Baby	0.600	0.600	0.574	0.599	0.575	0.597	**0.613**
Beauty	0.492	0.518	0.515	0.506	0.501	0.497	**0.523**
Office Product	0.792	0.789	0.762	0.791	0.783	0.792	**0.812**
Clothing	0.706	0.705	0.678	0.703	0.687	0.706	**0.716**
Toys	0.553	0.564	0.528	0.559	0.551	0.548	**0.578**
Musical Instrusment	0.624	0.615	0.595	0.615	0.610	0.619	**0.639**
Pet Supplies	0.788	0.782	0.758	0.784	0.774	0.786	**0.801**
Health	0.505	0.532	0.526	0.520	0.528	0.512	**0.543**
Books	0.342	0.381	0.375	0.362	0.373	0.335	**0.399**
Electronics	0.511	0.521	0.516	0.515	0.518	0.509	**0.541**
Movies and TV	0.336	0.385	0.363	0.356	0.366	0.344	**0.392**
Average	0.601	0.611	0.590	0.606	0.601	0.600	**0.625**

Chapter 6

Conclusion and Future Work

In this chapter, we give a conclusion for our thesis and mention about possible future research direction.

6.1 Conclusion

In this thesis, we formulate the problem of evaluating review's quality which is an important task in online review portals. We hypothesize that quality of review is an underlying property of review text and a review could be represented by features of topics information in review content. We develop a hybrid (unsupervised and supervised) method to predict reviews' quality using primarily hidden topics distribution (our proposed features) in the review.

Furthermore, we propose a probabilistic model with mathematical formulation and probabilistic definition to capture the probabilistic meaning of review's quality. And in corresponding with this definition, we then develop a logistic regression method using helpful-unhelpful votes population for review to learn a model to inference review's quality as review's helpfulness. The reviews which have sufficient number of votes for themselves could be used as supervised data for our learning machine.

We further propose a confident model combination of features to archive better results when using different types of features. Our learning machine takes different type of feature

sets to produce different predicting values but adopt only the most confident value in set of predictions.

Finally, we implement our proposals in real application with key concept of system in processing and evaluating review's quality. We also validate our proposal in some real review datasets.

We have found that our proposed logistic regression using votes population in learning model is state of the art in review's quality prediction task. Furthermore, our proposed TOPICS features give the trade-off performance in review quality prediction by two kinds of evaluation metric: RMSE(root mean square error between predicted value and true quality value) and Spearman correlation coefficient (in ranking evaluation metric). Finally, the combination of models (corresponding to different types of features) output a confident value is state of the art in predicting and classifying task compare with any single feature type.

6.2 Future Work

The quality evaluation of online reviews is a relatively new but important research problem in the domain of text mining and knowledge discovery. In this section, we discuss some possible future research directions to improve the effectiveness of our framework.

6.2.1 Feature Setting

In this thesis, we have provided many of feature types including our proposal (TOPICS, FeatureCombination) and previous researches (SaS, TF-IDF, LIWC, INQUIRER, GALC). The common point in these settings is the unsupervised process to extract feature. We plan to construct a supervised process to design feature with target learning is the review's quality based on votes information. This supervised process is a combination of two stages in our systems: feature extraction and quality learning, makes an appropriate design for feature representation of review correspond with its quality.

It is easy to realize that this supervised feature design process is appropriate only for

our proposal feature type. We could apply the idea of supervised latent Dirichlet allocation (Mcauliffe and Blei, 2007 ([41])) in which the model accommodates the helpful votes ratio in a maximum-likelihood procedure for parameter estimation of topics.

Another direction would be to consider in our work is exploiting the meta feature as information of reviewer (e.g. number of past reviews, average rating for review,...) with premise that the quality of a review depends on reviewer's quality. In addition, we could add regularize condition such as author consistency (a reviewer that writes high quality of review is likely to writing good reviews), then the objective function becomes:

$$\mathcal{L}^*(q) = \mathcal{L}(q) + \eta \sum_{u \in \boldsymbol{U}} \sum_{r_i, r_j \in \mathcal{R}_u} (q(r_i) - q(r_j))^2 \tag{6.1}$$

where $\boldsymbol{U}$ is reviewer set and $\mathcal{R}_u$ denote for the set of reviews by reviewer u in $\boldsymbol{U}$.

In this thesis, we have developed method based on all reviews on all items (products, restaurant,...) of a category dataset and design for common feature properties. However, the variation of items' characteristics need for dynamic feature design such as the feature represent for item's profile. Our learning model may play a process not for mixture all items but for each specific item. However, this approach has disadvantages of collecting sufficient supervised reviews data for each specific item and need to examine carefully in the future.

6.2.2 Bayesian modeling the relation between review features and voter opinions

In this thesis, we have proposed a probabilistic definition for review quality as probability of helpful vote for this review given review features and modeled this term as logistic function of linear combination for features. Then we perform maximum likelihood estimation from this definition. The relation between review features $\boldsymbol{f}$ and voter opinions v in our thesis could be summarized in below equation.

$$p(v = 1|\boldsymbol{f}) = \frac{1}{1 + \exp(-\boldsymbol{w}^T \boldsymbol{f})} \tag{6.2}$$

In the future, we plan to model the dependency of review features and voter opinions based on Bayesian model. From Bayes theorem, we have

$$p(v = 1|\boldsymbol{f}) = \frac{p(v = 1, \boldsymbol{f})}{p(\boldsymbol{f})} \propto p(\boldsymbol{f}|v = 1)p(v = 1) \tag{6.3}$$

and

$$p(v = 0|\boldsymbol{f}) = \frac{p(v = 0, \boldsymbol{f})}{p(\boldsymbol{f})} \propto p(\boldsymbol{f}|v = 0)p(v = 0) \tag{6.4}$$

If we display $p(\boldsymbol{f}|v = 1)$ and $p(\boldsymbol{f}|v = 0)$ by some probabilistic distribution (like mixture of Gaussian process) with parameter $\boldsymbol{\theta}_1$ and $\boldsymbol{\theta}_2$. We could form a optimization problem as finding $\boldsymbol{\theta}_1$ and $\boldsymbol{\theta}_2$ to maximize $p(\boldsymbol{f}_i|\boldsymbol{V}_i)$. Then we could use the density function of votes given features defined in 6.3 and 6.4 to evaluate review quality.

6.2.3 Human labeling as ground-truth data

In this thesis, we used the ground-truth target learning is helpful votes ratio of the reviews those have enough votes for themselves. However, these votes information are sometimes bias and unreliable (e.g. spam or mis-match votes) thus our proposed features are consistent with semantic meaning of review quality but not for learning model. To deal with this bias, we plan to make human labeling and scoring for review's quality and validate our proposed features with this ground-truth.

Appendix A

Compare Between Means

This appendix explains how to conduct a hypothesis test for compare between two means of two sample sets. There are four steps of this approach.

State the Hypotheses

Firstly, we need to state a *null hypothesis* and its exclusive statement: *alternative hypothesis*. Table A.1 shows three combinations of null and alternative hypotheses to compare between mean of one population μ_1 vs. the mean of another population μ_2.

Table A.1: Combination of null hypothesis and alternative hypothesis

Null hypothesis	**Alternative hypothesis**	**Number of tails**
$\mu_1 = \mu_2$	$\mu_1 \neq \mu_2$	2
$\mu_1 \geq \mu_2$	$\mu_1 < \mu_2$	1
$\mu_1 \leq \mu_2$	$\mu_1 > \mu_2$	1

The first combination using two-tailed test, since we could reject the null hypothesis if the finding sample fall into extreme side of sampling distribution. The other combinations are one-tailed tests, since we invest only one side of sampling distribution to reject the null hypothesis.

Choose Method of Analysis

In this step, we need to make clear the method of accept or reject the null hypothesis by following factors.

- Significant level: any value between 0 and 1 can be used but 0.01, 0.05 or 0.10 are often used.

- Test method: the two-sample t-test is adopted to determine the difference between means found in sample is significantly different from the hypothesized different between means.

Analyze Sample Data

In this step, we need to find the standard error, degrees of freedom, test statistic and the P-value associated with the test statistic.

- Standard error of the sampling distribution:

$$SE = \sqrt{\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}} \tag{A.1}$$

  where s_1, n_1, s_2, n_2 are the standard deviation, the size of sample 1 and sample 2 respectively.

- Degrees of freedom is computed as the nearest integer of:

$$DF = \frac{(s_1^2/n_1 + s_2^2/n_2)^2}{\frac{(s_1^2/n_1)^2}{n_1 - 1} + \frac{(s_2^2/n_2)^2}{n_2 - 1}} \tag{A.2}$$

- T-score:

$$t_0 = \frac{\bar{x_1} - \bar{x_2}}{SE} \tag{A.3}$$

  where $\bar{x_1}, \bar{x_2}$ are the means of sample 1 and sample 2 respectively.

- The P-value is the probability of observing a sample statistic as extreme as the test statistic. In this method, t distribution is used with degrees of freedom (DF) is com-

puted before.

$$p_{val} = \quad P(|t| < t_0) \text{ if null hypothesis is } \mu_1 = \mu_2 \quad \text{(two tailed test)} \tag{A.4}$$

$$p_{val} = \quad P(t <= t_0) \text{ if null hypothesis is } \mu_1 \geq \mu_2 \quad \text{(one tailed test)} \tag{A.5}$$

$$p_{val} = \quad P(t >= t_0) \text{ if null hypothesis is } \mu_1 \leq \mu_2 \quad \text{(one tailed test)} \tag{A.6}$$

Interpret Results

If the P-value is smaller or equal the significant level, we can reject the hypothesis. If the P-value is greater than the significant level, we cannot reject the null hypothesis. It means that the null hypothesis could be accepted in our hypothesis test.

Bibliography

[1] Nielsen-Online. 81 percent of online holiday shoppers read online customer reviews, according to nielsen online., 2008. Available online on December 2015 at http://www.nielsen-online.com/pr/pr-081218.pdf.

[2] A. Palmer. Web shoppers trust customer reviews more than friends., September 2009. Available online on December 2015 at http://www.adweek.com/news/advertising-branding/web-shoppers-trust-customer-reviews-more-friends-100313.

[3] Do-Hyung Park, Jumin Lee, and Ingoo Han. The effect of on-line consumer reviews on consumer purchasing intention: The moderating role of involvement. *Int. J. Electron. Commerce*, 11(4):125–148, July 2007.

[4] Nan Hu, Ling Liu, and Jie Jennifer Zhang. Do online reviews affect product sales? the role of reviewer characteristics and temporal effects. *Inf. Technol. and Management*, 9(3):201–214, September 2008.

[5] Do-Hyung Park and Sara Kim. The effects of consumer knowledge on message processing of electronic word-of-mouth via online consumer reviews. *Electron. Commer. Rec. Appl.*, 7(4):399–410, December 2008.

[6] Jingjing Liu, Yunbo Cao, Chin-Yew Lin, Yalou Huang, and Ming Zhou. Low-quality product review detection in opinion summarization. EMNLP 2007, June 2007.

[7] Richong Zhang, Thomas Tran, and Yongyi Mao. Opinion helpfulness prediction in the presence of words of few mouths. *World Wide Web*, 15(2):117–138, 2012.

[8] Yelp. Yelp dataset challenge., 2015. Available online on October 2015 at http://www.yelp.com/dataset_challenge.

[9] Wenjing Duan, Bin Gu, and Andrew B. Whinston. Do online reviews matter? - an empirical investigation of panel data. *Decis. Support Syst.*, 45(4):1007–1016, November 2008.

[10] Jumin Lee, Do-Hyung Park, and Ingoo Han. The effect of negative online consumer reviews on product attitude: An information processing view. *Electron. Commer. Rec. Appl.*, 7(3):341–352, November 2008.

[11] Ivar E. Vermeulen and Daphne Seegers. Tried and tested: The impact of online hotel reviews on consumer consideration. *Tourism Management*, 30(1):123 – 127, 2009.

[12] Do-Hyung Park and Jumin Lee. ewom overload and its effect on consumer behavioral intention depending on consumer involvement. *Electron. Commer. Rec. Appl.*, 7(4):386–398, December 2008.

[13] Yang Liu, Xiangji Huang, Aijun An, and Xiaohui Yu. Modeling and predicting the helpfulness of online reviews. In *Proceedings of the 2008 Eighth IEEE International Conference on Data Mining*, ICDM '08, pages 443–452, Washington, DC, USA, 2008. IEEE Computer Society.

[14] Cristian Danescu-Niculescu-Mizil, Gueorgi Kossinets, Jon Kleinberg, and Lillian Lee. How opinions are received by online communities: A case study on amazon.com helpfulness votes. In *Proceedings of the 18th International Conference on World Wide Web*, WWW '09, pages 141–150, New York, NY, USA, 2009. ACM.

[15] Susan M. Mudambi and David Schuff. What makes a helpful online review? a study of customer reviews on amazon.com. *MIS Q.*, 34(1):185–200, March 2010.

[16] Anindya Ghose and Panagiotis G. Ipeirotis. Estimating the helpfulness and economic impact of product reviews: Mining text and reviewer characteristics. *IEEE Trans. on Knowl. and Data Eng.*, 23(10):1498–1512, October 2011.

[17] A. Spool. The magic behind amazon's 2.7 billion dollar question., March 2009. Available online on December 2015 at https://www.uie.com/articles/magicbehindamazon/.

[18] Soo-Min Kim, Patrick Pantel, Tim Chklovski, and Marco Pennacchiotti. Automatically assessing review helpfulness. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*, EMNLP '06, pages 423–430, Stroudsburg, PA, USA, 2006. Association for Computational Linguistics.

[19] Zhu Zhang and Balaji Varadarajan. Utility scoring of product reviews. In *Proceedings of the 15th ACM International Conference on Information and Knowledge Management*, CIKM '06, pages 51–57, New York, NY, USA, 2006. ACM.

[20] Michael P. O'Mahony and Barry Smyth. Using readability tests to predict helpful product reviews. In *Adaptivity, Personalization and Fusion of Heterogeneous Information*, RIAO '10, pages 164–167, Paris, France, France, 2010. LE CENTRE DE HAUTES ETUDES INTERNATIONALES D'INFORMATIQUE DOCUMENTAIRE.

[21] Yue Lu, Panayiotis Tsaparas, Alexandros Ntoulas, and Livia Polanyi. Exploiting social context for review quality prediction. In *Proceedings of the 19th International Conference on World Wide Web*, WWW '10, pages 691–700, New York, NY, USA, 2010. ACM.

[22] Lionel Martin and Pearl Pu. Prediction of helpful reviews using emotions extraction. In *Proceedings of the Twenty-Eight AAAI Conference on Artificial Intelligence*, AAAI '14.

[23] Yaowei Yan Yinfei Yang and Forrest Bao Minghui Qiu. Semantic analysis and helpfulness prediction of text for online product reviews. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing*, ACL-IJCNLP 2015, pages 38–44. Association for Computational Linguistics, July 2015.

[24] David M Blei, Andrew Y Ng, and Michael I Jordan. Latent dirichlet allocation. *the Journal of machine Learning research*, 3:993–1022, 2003.

[25] Matthew Hoffman, Francis R. Bach, and David M. Blei. Online learning for latent dirichlet allocation. In J.D. Lafferty, C.K.I. Williams, J. Shawe-Taylor, R.S. Zemel, and A. Culotta, editors, *Advances in Neural Information Processing Systems 23*, pages 856–864. Curran Associates, Inc., 2010.

[26] T. L. Griffiths and M. Steyvers. Finding scientific topics. *Proceedings of the National Academy of Sciences*, 101(Suppl. 1):5228–5235, April 2004.

[27] Carson Sievert and Kenneth Shirley. Ldavis: A method for visualizing and interpreting topics. In *Proceedings of the Workshop on Interactive Language Learning, Visualization, and Interfaces*, pages 63–70, Baltimore, Maryland, USA, June 2014. Association for Computational Linguistics.

[28] Yang Liu, Xiangji Huang, Aijun An, and Xiaohui Yu. Modeling and predicting the helpfulness of online reviews. In *Proceedings of the 2008 Eighth IEEE International Conference on Data Mining*, ICDM '08, pages 443–452, Washington, DC, USA, 2008. IEEE Computer Society.

[29] Christophe Leys, Christophe Ley, Olivier Klein, Philippe Bernard, and Laurent Licata. Detecting outliers: Do not use standard deviation around the mean, use absolute deviation around the median. *Journal of Experimental Social Psychology*, 49(4):764 – 766, 2013.

[30] PredictionIO. Open-source machine learning server, 2014. Available online on September 2014 at https://prediction.io/.

[31] Kavita Ganesan and ChengXiang Zhai. Opinion-based entity ranking. *Information retrieval*, 15(2):116–150, 2012.

[32] Julian McAuley, Rahul Pandey, and Jure Leskovec. Inferring networks of substitutable and complementary products. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '15, pages 785–794, New York, NY, USA, 2015. ACM.

[33] Wenting Xiong and Diane Litman. Automatically predicting peer-review helpfulness. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: Short Papers - Volume 2*, HLT '11, pages 502–507, Stroudsburg, PA, USA, 2011. Association for Computational Linguistics.

[34] Deepak Agarwal, Bee-Chung Chen, and Bo Pang. Personalized recommendation of user comments via factor models. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '11, pages 571–582, Stroudsburg, PA, USA, 2011. Association for Computational Linguistics.

[35] Klaus R. Scherer. What are emotions? and how can they be measured? *Social Science Information*, 44(4):695–729, 2005.

[36] JW Pennebaker, ME Francis, and RJ Booth. *Linguistic inquiry and word count [computer software]*. Mahwah, NJ: Erlbaum Publishers, 2001.

[37] Philip J. Stone and Earl B. Hunt. A computer approach to content analysis: Studies using the general inquirer system. In *Proceedings of the May 21-23, 1963, Spring Joint Computer Conference*, AFIPS '63 (Spring), pages 241–256, New York, NY, USA, 1963. ACM.

[38] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[39] Seiya Tokui, Kenta Oono, Shohei Hido, and Justin Clayton. Chainer: a next-generation open source framework for deep learning. In *Proceedings of Workshop on Machine Learning Systems (LearningSys) in The Twenty-ninth Annual Conference on Neural Information Processing Systems (NIPS)*, 2015.

[40] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15:1929–1958, 2014.

[41] Jon D. Mcauliffe and David M. Blei. Supervised topic models. In J.C. Platt, D. Koller, Y. Singer, and S.T. Roweis, editors, *Advances in Neural Information Processing Systems 20*, pages 121–128. Curran Associates, Inc., 2008.