Ph.D. Dissertation of Linguistics

Estimating the Helpfulness of Product Reviews based on Review Information Types

리뷰 정보 유형에 기반한
상품평 유용성 평가

August 2016

Graduate School of Seoul National University

Department of Linguistics

Munhyong Kim

Abstract

Estimating the Helpfulness of Product Reviews based on Review Information Types

Munhyong Kim

Department of Linguistics

The Graduate School of Seoul National University

The sheer number of product reviews for any given product makes it impossible for potential customers to locate those reviews that will be helpful to them. This results in the need to automatically estimate the helpfulness of product reviews such that customers may locate the most helpful ones as quickly and easily as possible. Researchers have explored multiple ways of evaluating review helpfulness, but have mainly focused on how reviews deliver information, i.e., the length, sentiment aspect, readability, etc. However, we make the assumption that it is more important to consider what information reviews deliver to customers than how that information is delivered.

Therefore, this study investigates a way of extracting what information reviews deliver to estimate the helpfulness of those reviews.

To extract information that reviews contain, we categorized the review information types (RIT) for each sentence. When considering the information

target, information can be divided into background information about the reviewer's previous experience or expertise, core information about the product, peripheral information about non-product information, such as shipping or packaging, and none-relevant information. Overall information contains final purchasing decision, summary and recommendations.

Once the information type of each sentence is categorized, every sentence is converted into a topic dimension vector with the Latent Dirichlet Allocation. For each type of information, topic-based vectors are clustered to find similar-information holding clusters. Then, these clusters are used to extract what information each sentence delivers for sentences in product review test data.

The product reviews are collected for an e-book reader, outdoor tent, and jeans from Amazon.com. For each product domain, 200 reviews are chosen for training and testing for various experiments. The helpfulness score for reviews and review information type for each sentence are manually annotated for this study.

To begin with, we present to what extent it is possible to correctly predict the information type of each sentence through various classification experiments. The review information type of each sentence is predicted based on various features: such as bag-of-words, the position of the sentence in a review, and the form and part-of-speech tag for main subject, verb, and auxiliaries.

A preliminary experiment was conducted to foresee the possibility of using background information to predict the helpfulness of product reviews. This experiment result indicates that our approach with only background information performs as effectively as the features from previous studies.

The final experiments are to mainly show the effect of extracting what information is delivered compared with that of extracting how information is delivered on estimating the helpfulness of product reviews. Through various experiments, we proved that our approach of extracting what information is delivered can more accurately estimate the helpfulness of reviews than features related with how information is delivered.

Keywords : review helpfulness estimation, review information types, latent dirichlet allocation, topic-based approach, product review evaluation

Student Number : 2010-30873

Table of Content

Abstract i
Table of Content v
List of Tables ix
List of Figures xii
List of Equations xiii
1 Introduction 1
1.1 Research Summary 4
1.1.1 Review Information Types for Estimating Review Helpfulness 4
1.1.2 Problem Statement 7
1.1.3 Investigation of Hypotheses 10
1.2 Outline 11
2 Background 13
2.1 Predicting Helpfulness of Reviews 13
2.2 Factors for Review Helpfulness 15
2.2.1 Basic Factors 16
2.2.2 Readability 17
2.2.3 Subjectivity 18
2.2.4 Content 20
2.3 Summary 24
3 Extracting Information from Reviews 25
3.1 Review Information Types 25
3.1.1 Motivation 25
3.1.2 Introducing Review Information Types 26
3.1.3 Difficulties and Ambiguities 28
3.2 Finding Similar Information-bearing Sentences 31
3.2.1 Sentence Representations 32
3.2.2 Clustering Similar Information-bearing Sentences 35
3.3 The Summary of the Extracting Procedure 37

4 Preparing Product Reviews 40
4.1 Collecting Data 40
4.2 The Review Helpfulness Vote Score 42
4.3 Building Review Helpfulness Manual Score 45
4.3.1 Annotating Manual Helpfulness Score 45
4.3.2 Evaluation of Review Helpfulness Manual Score 48
4.4 Annotation of Information Types 53
4.5 Summary 55
5 A Preliminary Study: Introducing Background Information Type for Product Review Helpfulness 57
5.1 Task Description 57
5.2 Data Collection 58
5.3 Extracting Background Information 58
5.3.1 Pattern Matching for Background Information 59
5.3.2 Seed-based Information Extraction 60
5.3.3 Topic-based Information Extraction 61
5.3.4 Features 61
5.4 Experiments and Analysis 64
5.4.1 Experiment Setting 64
5.4.2 Model 64
5.4.3 The Evaluation Metrics 66
5.4.4 Results and Analysis 70
5.5 Summary 72
6 Recognition of Review Information Types 74
6.1 Task Description 74
6.2 Models 75
6.2.1 Unsupervised Clustering Methods 76
6.2.2 Supervised Learning Models 76
6.3 Features for Recognizing Information Types 80

6.4 Recognition of Review Information Types....81
6.4.1 The Results with Clustering Models....81
6.4.2 The Results with SVM model....83
6.4.3 The Results with CRF model....88
6.5 The Summary of Recognizing Information Types....91
7 Estimation of Review Helpfulness....93
7.1 Task Description and Restriction....93
7.2 Data Collection....94
7.3 Features for Estimating the Review Helpfulness....94
7.3.1 Baseline (BASE)....94
7.3.2 Features from Previous Studies....95
7.3.3 Product Aspect Keyword-based Features (ASPECT)....99
7.3.4 The Proportion of Information Types (INFO_TYPE)....101
7.3.5 The Semantics of Sentence Information....101
7.4 Experimental Setting....103
7.4.1 Evaluating Clustering Algorithms....104
7.5 Experiment Results....108
7.5.1 Gold Standard Ranking Validation....109
7.5.2 Sentence Representations....112
7.5.3 The Best Feature Combinations....114
7.5.4 Whole Document vs Separate Sentences....125
7.5.5 No Distinction on Information types....126
7.5.6 Review Helpfulness Evaluation with Predicted Sentence Information Types....127
7.5.7 The Product Domain Adaptation....129
7.6 Summary....130
8 Conclusions and Future Directions....133
8.1 Summary of Contribution and Results....134
8.1.1 Categorization of Information Types....134

8.1.2 Review Helpfulness Annotation....134
8.1.3 Features for Recognizing Review Information Types....135
8.1.4 Computational Modeling of Information Type Recognition....135
8.1.5 Features and Computational modeling for Estimating Review Helpfulness....136
8.2 Future Directions and Open Problems....137
8.2.1 Extraction of Sentence Information....137
8.2.2 Topic based Clustering....138
8.2.3 Remaining Practical Issues....138
8.2.4 Expandability of Review Information Types....139
REFERENCES....140
Appendix I. Product lists and Ids from Amazon.com....142
Appendix II. Regular patterns for finding background information of e-book reader reviews....144
Appendix III. Groups of product features for each product domain....146
국문초록....152

List of Tables

Table 1. Collected Data 41
Table 2. The Spearman Rank Correlation between Annotators (e-book reader).....51
Table 3. The Spearman Rank Correlation between Annotators (tent).....51
Table 4. The Spearman Rank Correlation between Annotators (jeans).....52
Table 5. The Agreement on the Ranks of 10 Reviews between Annotators.....52
Table 6. The Number of Information Types 54
Table 7. Agreement for judging information types of e-book reader reviews.....55
Table 8. Agreement for judging information types of tent reviews.....55
Table 9. Agreement for judging information types of jeans reviews.....55
Table 10. Product reviews used for this preliminary study 58
Table 11. Summary of Background Knowledge with Regular Patterns.....60
Table 12. Separate Feature Examination with helpfulness vote score (h_v).....71
Table 13. Separate Feature Examination with helpfulness manual score (h_m).....71
Table 14. The best combination of feature groups with h_m 72
Table 15. The Clustering Results of Review Sentences for an E-reader with DBSCAN.....82
Table 16. The Clustering Results for sentences with k-means (e-book reader reviews).....83
Table 17. The Clustering Results for sentences with k-means (tent reviews).....83
Table 18. The Result of Recognizing Information Types with SVM (e-book reader) 85
Table 19. The Result of Recognizing Information Types with SVM (Tent) 86
Table 20. The Result of Recognizing Information Types with SVM (Jeans).....87
Table 21. The Result of Recognizing Information Types with CRF (e-book Reader) 89
Table 22. The Result of Recognizing Information Types with CRF (Tent) 90
Table 23. The Result of Recognizing Information Types with CRF (Jeans).....90

Table 24. The Result of Recognizing Information Types with CRF (given previous information types) 91
Table 25. Collected Data 94
Table 26. Example of Product Aspect Keywords 99
Table 27. The Clustering Results of Core Information Sentences with DBSCAN 105
Table 28. Varying cluster numbers in Background sentences for e-book reader reviews 106
Table 29. Varying cluster numbers in Core sentences for e-book reader reviews 107
Table 30. Varying cluster numbers in Peripheral sentences for e-book reader reviews 107
Table 31. Varying cluster numbers in Overall sentences for e-book reader reviews 107
Table 32. Varying cluster numbers in Background sentences for tent reviews 107
Table 33. Varying cluster numbers in Core sentences of tent reviews 107
Table 34. Varying cluster numbers in Peripheral sentences of tent reviews 107
Table 35. Varying cluster numbers in Overall sentences of tent reviews 108
Table 36. Varying cluster numbers in Background sentences of jeans reviews 108
Table 37. Varying cluster numbers in Core sentences for jeans reviews 108
Table 38. Varying cluster numbers in Peripheral sentences for jeans reviews 108
Table 39. Varying cluster numbers in Overall sentences jeans reviews 108
Table 40. The different effects of features depending on the helpfulness vote score (h_v) 111
Table 41. The different effects of features depending on the manual helpfulness score with all three score dimensions (h_m) 111
Table 42. Different Effect of Sentence Information Representations 113
Table 43. The Best Performing Feature Combination for e-book reader Reviews 118
Table 44. The Best Performing Feature Combination for Tent Reviews 120
Table 45 The Best Performing Feature Combination for jeans reviews 121
Table 46. The Comparison of Document-based with Sentence-based Approach 126

Table 47. The Comparison of results with no information type distinction.....127
Table 48. Comparison of results with previous studies and given information type and results with predicted information types 128
Table 49. The result of product domain adaptation.....129

List of Figures

Figure 1. The Illustration of Review Information Types.....28
Figure 2. The Illustration of Extracting Information from Review Sentences.....37
Figure 3. The Reviews' Percentage Score (J. Liu et al., 2007) 42
Figure 4. Votes of the Top-50 Ranked Reviews (J. Liu et al., 2007).....43
Figure 5. Dependency on Publication Date.....44
Figure 6. The soft margin loss setting corresponds to the linear SV machine (Smola & Schölkopf, 2004).....66
Figure 7. The Algorithm to Calculate the Ranking Distance 69
Figure 8. Linear separating hyperplanes for the separable case. The support vectors are circled. (Burges, 1998) 78
Figure 9. Predicting the orientation of opinion sentences (Hu & Liu, 2004).....97
Figure 10. An example of feature extractions for how information is delivered and helpfulness estimation 124
Figure 11. An example of feature extractions of what information is delivered and helpfulness estimation 125

List of Equations

Equation 1. tf-idf formula.....33
Equation 2. Helpfulness Vote Score (h) 41
Equation 3. The Spearman’s Rank Correlation Coefficient 49
Equation 4. The linear function and conditions for ε-sensitive SVR 65
Equation 5. Optimization Problem with slack variables ($\xi i, \xi i*$) 65
Equation 6. The Kendall’s tau-b.....67
Equation 7. The Equation to Calculate the Ranking Distance 68
Equation 8. Formulas of Readability Measures 98

1 Introduction

The number of online reviews for a given product available at an online retailer, for instance Amazon.com, can number more than 30,000. When it comes to data from the entire web, including blogs, other retailers, and review sites, the number of reviews becomes uncountable. Thus, review ranking becomes an important service to ensure that consumers are able to identify and read only helpful reviews before making a purchasing decision, and thereby enhancing the customer experience.

Although most e-commerce companies enable users to rank reviews based on product rating score or how recently a review was written, they do not sufficiently suggest helpful reviews to customers. For an enhanced review ranking service, Amazon.com allows customers to vote on the helpfulness of reviews, and reviews are then automatically ranked by their number of helpful votes. Unfortunately, this helpfulness vote system does not accurately rank reviews as a result of various biases (J. Liu, Cao, Lin, Huang, & Zhou, 2007). Additionally, most reviews do not receive enough helpfulness votes. For instance, among one product's 19153 reviews, only 561 reviews have more than 10 helpfulness votes, and more than 30% of the reviews do not have any votes. Therefore, a study of automatic evaluation of review helpfulness could be a breakthrough for a review ranking system.

There exist a lot of factors that are related to the helpfulness of reviews. Consider the following examples of snippets from e-book reader reviews.

(1) *The screen isn't as responsive or accurate with fingers.*

(2) *First the touch screen is **too** sensitive. It is constantly switching to something I never intended it to go to, such as word look up. **Just a shadow of your finger** is enough to trigger it. I am always **cussing** as I read because if I shift my fingers it skips to some other function or page.*

(3) *Responsiveness of the Touchscreen: as touchscreens go, it's pretty good, but a light tap doesn't always activate the page turn--I have to keep my finger on the screen a split-second longer, and have to press it a bit harder than I would like.*

(4) *They'd use an actual shipping box instead of just slapping mailing labels all over the retail box.*

(1), (2) and (3) are about the sensitivity of the touch screen. The first noticeable difference among them is the length of texts, the number of tokens to describe the same product aspect. People might judge (2) and (3) more helpful than (1) while describing the same product feature due to their length. The length of the reviews is the most common and naïve factor in estimating the helpfulness of product reviews in previous studies (Cao, Duan, & Gan, 2011; Chen & Tseng, 2011; Kim, Pantel, Chklovski, & Pennacchiotti, 2006; Korfiatis, García-Bariocanal, & Sánchez-Alonso, 2012; Y. Liu, Huang, An, & Yu, 2008; O'Mahony & Smyth, 2009; Zhang & Varadarajan, 2006).

Another difference between (2) and (3) is the intensity of polarity. The example in (3) expresses a negative polarity sentiment on the touch screen and (2) expresses the same idea with extremely negative expressions and exaggeration such as “too”,

“cussing” and “just a shadow of your finger”. A too intense polarity sentiment could degrade the helpfulness of the review. The sentimental aspect of product reviews can also be a factor of review helpfulness (Chen & Tseng, 2011; Ghose & Ipeirotis, 2007, 2011; Kim et al., 2006; Zhang & Varadarajan, 2006). The readability of a review can also be related with the review’s helpfulness (Cao et al., 2011; Ghose & Ipeirotis, 2007, 2011; Kim et al., 2006). These factors are all related with how reviews deliver information.

However, in (4), the sentence is about a shipping problem, not the product itself. It is important to note that informatin related to shipping may not be information that people are interested in. Though the length of the sentences about a shipping problem may increase the length of the review, it may not be crucial information related to a reader’s desire to buy the product. Thus, what the information is about in product review is crucial in estimating review helpfulness.

Moreover, the reason for judging (2) and (3) more helpful than (1) is not just the length of the description, but (2) and (3) also contain the information of the writer’s personalized experience that a reader may seek in product reviews. This factor is also related with what information the sentence contains.

We assume that review helpfulness is most dependent on the informativeness of review, in another words, what information a review delivers to the reader. To extract what information a review offers, we proposes categorizing review information types based on the target of the information and the effectiveness of that information. As can be seen in (4), the type of target information is important for estimating review helpfulness. We assume that the categorization of

information types would consequently help to accurately find what information sentences hold.

1.1 Research Summary

The goal of this research is **to examine the possibility of categorizing information types for a certain type of text such as a product review and its effect on recognizing the specific meaning of sentences to estimate product review helpfulness**.

In order to understand the influence of review information types on review helpfulness estimation, we define specific review information types and then study the effect of these on improving the estimation of product review helpfulness.

1.1.1 Review Information Types for Estimating Review Helpfulness

Up to this point, there have been no studies that have attempted to categorize the information types of product review sentences. Rather, it was observable from other studies that the helpfulness of reviews could be predicted by using the meta-data of reviewers (Chen & Tseng, 2011; Ghose & Ipeirotis, 2011; Y. Liu et al., 2008; O'Mahony & Smyth, 2009) or product aspect-indicating words (Chen & Tseng, 2011; Kim et al., 2006). For instance, Chen and Tseng (2011) measured the reputation of reviewers with the number of reviews written by the reviewer and the ranking of the reviewer. Ghose and Ipeirotis (2011) extracted whether the reviewer disclosed their information, such as real name, location, nickname or hobbies and the reviewer's history by calculating division of the total "yes" helpfulness vote

number of all the reviews by the reviewer with the total “yes + no” helpfulness vote number of all his reviews and additionally averaging all helpfulness scores (“yes” / “yes + no”) from each review. Explicit product aspect-indicating words are also applied in studies that estimate the review helpfulness (Chen & Tseng, 2011; Kim et al., 2006). These studies indicate that reviewer-related and product-related information is valuable information that review readers want to find from reviews.

We categorized the information type of review sentences based on the *target of the information* and *the influence of the information on review helpfulness*. When considering the information target, information can be divided into *background* information about the reviewer, *core* information about the product, *peripheral* information about non-product information and *none-relevant* information. The examples in (5) to (7) showed *background*, *core*, *peripheral* and *none-relevant* information respectively.

(5) I have had every kindle made, my two sons have Kindles as do my nephews and mom.

(6) The battery drains faster than previous models I had (even with the screen light AS LOW AS POSSIBLE)

(7) It was ordered on December 22 only because Amazon promised delivery on December 24; otherwise I would have bought it locally.

(8) All you do with your Down votes is discredit and undermine the Amazon Review system.

In (5), we can see that the reviewer has had sufficient experience of previous versions of the product, so he is able to write a better review. Though this type of information does not give any information about the product, it raises the credibility of the review to readers. This type of information, *background*, usually is placed at the beginning of the reviews and is composed usually of a few sentences.

In (6), the sentence describes how the reviewer has felt about the product's battery life. Battery life is one of the most important features for an e-book reader. This type of information, *core*, is what people want to find from reviews, thus it expectedly has more influence on a review's helpfulness than other information types. How deeply and widely this type of information is dealt largely decides the helpfulness of the review.

In (3), the sentence is about the shipping, which is not about the product itself. It is peripheral. This type of information is not unhelpful, and readers will think more negatively about the review as the length of this type of information becomes longer.

The last information type, *none-relevant*, is information that is not related at all, which is not what people look for from product reviews, illustrated in (8). Non-relevant type information negatively influences the review helpfulness.

As seen in the examples of background, core, peripheral and non-relevant information, the categories are separated not only based on the target of the information but also the different tendency of effect on the review's helpfulness.

For this reason, the influence of the information on review helpfulness derives another category: overall judgment. Reviewers often include final remarks on the

product or summaries of the review in the beginning or the end of their review as in (9). These sentences are helpful, but as the length of the review increases, they decrease in helpfulness, unlike the core information we explained above. Even though the target of the information of this overall judgment is the product, this is different from core information in the influence on the review's helpfulness. Readers are unlikely to enjoy reading the same but lengthy content that could be infered from the overall review.

(9) Regarding the rest, Kindle is a wonderful product which I couldn't live without it.

1.1.2 Problem Statement

The question "what is a helpful review?" should be addressed before analyzing the helpfulness of product reviews. It is not an easy question that can be answered simply, since everyone is different in assessing the helpfulness of reviews. Nevertheless, we can at least categorize the factors on the helpfulness of reviews into two different dimensions: **how the information is delivered** and **what information is delivered**. A review can be helpful if it offers sufficient information about the experience of the product in depth, but could be unhelpful if it contains too many grammatical mistakes, unsuitable vocabulary, or exaggeratedly negative expressions that make the reader uncomfortable. Which factor is most influential on the helpfulness of reviews is not a matter that can be proved. It varies depending on the individual. **This study is interested mainly in capturing what information a review provides** to estimate the review helpfulness. To recognize

the information each review gives, this study proposes to categorize the Review Information Types (RIT) for review sentences so that the possible semantic space is reduced, consequently improving the clustering result of similar information-bearing sentences. By finding these similar information-bearing clusters from reviews of the training data, we extract what information each review provides. Therefore, the goal of the thesis is as stated below:

Goal: To Analyze what information a review provides for improving product review helpfulness estimation

To pursue the goal of the study, we initiate explorations into categorizing information types of review sentences by human experts, suggesting formal definitions of review information types and discussing the difficulties of applying these categories to the real-world data. Based on the review information types, it becomes possible to extract information from sentences by clustering similar-information bearing sentences (Chapter 3). Our proposed approach is examined by conducting review helpfulness estimation experiments. The process of collecting review data and annotating their helpfulness score and review information types is dealt with in Chapter 4. Then, a preliminary study is introduced which examined the effect of extracting background information from reviews on predicting review helpfulness to see the possibility of using review information types (Chapter 5). The main experiment is divided into two phrases, recognizing review information types (Chapter 6) and estimating review helpfulness based on the types (Chapter 7). In pursing the goal of this study, we investigate the following hypotheses.

1. The review helpfulness score can be reliably annotated by trained human annotators to be used as a gold standard to learn a review helpfulness estimation model.
2. The information types of review sentences can be reliably annotated by trained human annotators.
3. The information types of review sentences can be automatically recognized by classification models.
4. Categorizing information types of sentences helps to reduce the semantic space so that finding similar information-bearing sentences within each information type becomes more achievable.
5. Similar information-bearing sentence clusters for each information type can be used to find what specific information a review gives.
6. Extracting what specific information a review contains enables the creation of an improved model to predict the review helpfulness.
7. Finding similar information-bearing sentences within separate information types is more effective than that with no information categories on estimating review helpfulness.
8. Extracting information from each review sentence and aggregating them is more effective than extracting information from the whole document on estimating review helpfulness.

1.1.3 Investigation of Hypotheses

The hypotheses described above can be tested directly by some metrics or indirectly from experimental results.

The first hypothesis is tested by agreement evaluations between scores from annotators. The metric to evaluate the agreement is the Spearman rank correlation coefficient (ρ) (Myers, Well, & Lorch, 2010). This score can measure to what extent the ranks of the product reviews differ to determine the feasibility of annotating the scores of review helpfulness.

In order to test the second hypothesis, review sentences are randomly chosen to have their review information tyesp annotated by two annotators. The extent to which the two annotators agree on the information types of sentences indicates to what extent the categorization of review information types is appropriate and practically possible.

We examine the third hypothesis through a series of experiments to determine the review information types of sentences with various models. Supervised and unsupervised models are examined to predict information types with features that distinguish one information type from another.

The fourth hypothesis is difficult to directly measure by mathematical evaluation metrics since making a gold standard of similar information bearing sentence groups is a near impossible task due to the vagueness of sentence similarity. However, we assume that better clustering results in improved quality of extracting specific information from sentences, in turn leading to improved performance in estimating review helpfulness.

The fifth hypothesis, that similar information-bearing sentence clusters can be used to extract what specific information a review gives, does not need to be proven, but can be demonstrated with examples.

In order to test the sixth hypothesis, other approaches should be compared. In this study, two previous studies and other possible alternative approaches are compared to our approach on predicting review helpfulness.

To examine the seventh hypothesis, we conducted an experiment that compares the performance of ranking reviews by separating review information types with that of unifying all review information types into one. The result of this experiment will be introduced in Chapter 7.

The last hypothesis that our suggested approach is based on, which is that extracting information from each sentences and aggregating them for the review is more effective than extracting information from the whole document in estimating review helpfulness, is tested in Chapter 7 by conducting an experiment to extract information from the whole document without any consideration of information types.

1.2 Outline

We introduce previous studies in Chapter 3 that have examined various features and conditions on review helpfulness estimation. Those studies are categorized depending on the features they focused on.

In order to explain our approach to predicting review helpfulness, review information types are formally defined and the difficulties of applying these

categories to real world data are discussed in Chapter 3. Additionally, we propose an approach that extracts information from sentences based on similar information-bearing sentence clusters within each review information type.

Chapter 4 explains what and how product review data is collected and the necessity of building a gold standard for the review helpfulness score. This chapter reports the evaluation result of manually annotating the review helpfulness score in depth. In addition, the annotation results of the review information types are evaluated in terms of agreement.

Chapter 5 addresses the question of whether introducing background information into review helpfulness evaluation improves the review ranking quality, compared to other studies and possible naïve approaches. This was a preliminary study conducted before examining all the information types in depth.

In Chapter 6, the automatic recognition of review information types is examined with various supervised and unsupervised models. The recognition of information types would influence the overall performance of review helpfulness estimation.

The experiment estimating review helpfulness is conducted with all information types used as features in Chapter 7. This experiment is conducted first given the review information types to determine to what extent it is possible to correctly estimate review helpfulness. Then, the result of automatically recognizing review information types is given for the review information types of sentences to see if it is practically applicable for a complete, automated system of ranking reviews in terms of their helpfulness.

Finally, Chapter 8 summaries the contributions of this study and discuss possible ways to expand this result and approach to other fields.

2 Background

2.1 Predicting Helpfulness of Reviews

The helpfulness of product reviews can be predicted in various ways. One intuitive way of predicting review helpfulness is to classify helpful and unhelpful reviews, a binary classification. In Ghose and Ipeirotis (2007) and Ghose and Ipeirotis (2011), they experimentally define a review as helpful when useful votes / total votes $\geq$ 0.6. Then, from a trained regression model, they predicted the helpfulness of product reviews and classified whether or not the review is helpful. O'Mahony and Smyth (2009) also tried to classify the helpfulness of reviews with a threshold of 75% positive helpful votes out of the total votes. Though it is intuitively simple to make it a binary problem, the nature of review helpfulness is not suitable for a binary task.

In addition, review helpfulness could be measured in ordinal values or multi-class categories. Chen and Tseng (2011) predicted review helpfulness using a multi-class classification system: high-quality, medium-quality, low-quality, duplicate and spam. To classify the quality-related multi-classes, they applied the one-versus-all multi-class support vector machine. Cao et al. (2011) divided the review helpfulness into 8 ordinal values (“0” to “7”, “0” to “6” for the number of votes and “7” for “7 or more”). They used ordinal logistic regression models (OLR), which is an extension of the binary logistic regression model. The binary logistic regression model can accommodate only 2 ordinal values, but OLR can use cumulative logits with ordinal dependent variables. Thus, the dependent variable

for this study is the ordinal values indicating how many votes each review receives. Ordinal values and multi-classes used to predict the helpfulness of reviews are not feasibly applicable to product reviews, since the number of reviews for one product is sometimes more than 10,000. The reviews that belong to each ordinal value or class are still too many to look up for potential consumers. When it comes to world-wide-web data, these approaches cannot properly filter the reviews.

This leads to the conclusion that, to estimate review helpfulness, a regression model that can continuously line up product reviews is appropriate. In fact, the majority of previous studies applied regression models to solve this problem (Ghose & Ipeirotis, 2007, 2011; Kim et al., 2006; Korfiatis et al., 2012; Y. Liu et al., 2008; O'Mahony & Smyth, 2009; Zhang & Varadarajan, 2006).

While O'Mahony and Smyth (2009) classified the helpfulness of reviews into helpful or unhelpful categories, they additionally attempted to rank reviews in terms of prediction confidence score from a classification model. Moreover, Ghose and Ipeirotis (2007, 2011) first approximated a linear regression model for the relationship between the helpfulness and the informativeness of reviews and then classified their helpfulness with a proper threshold. Korfiatis et al. (2012) also built a linear regression model to find the relationship between the qualitative characteristics of the review text, review helpfulness and the impact of review helpfulness on the review score. Y. Liu et al. (2008) adopted the radial basis function, which is better suited for data that cannot be approximated with a linear function. The regression models above can approximate an equation that predicts the helpfulness of reviews, but are restricted in the types of independent variables, allowing only continuous values.

Kim et al. (2006) and Zhang and Varadarajan (2006) both used the support vector regression model (SVR), which allows the use of any type of independent variables. Thus, this model can be used to predict a continuous dependent variable with various independent variables: binary, categorical, ordinal, or continuous features. In this study, we applied the SVR model to learn and predict the helpfulness of reviews. A more detailed explanation of this model will be presented in Chapter 5.

2.2 Factors for Review Helpfulness

To predict review helpfulness, previous studies have focused on various aspects of product reviews: the words used in reviews, the readability or style, sentimental properties, or even meta-data of the reviews. The factors can be divided into in-text factors that can be extracted from the review text content and out-text factors that can be extracted from the metadata of reviews. For a highly-developed online system, such as Amazon.com or TripAdvisor.com, various kinds of metadata is available for reviews. The star-rating score on the product, the release-date, the number of review for the product and the metadata about reviewers' characteristics or history have been widely used to extract factors that are effective in estimating review helpfulness (Cao et al., 2011; Chen & Tseng, 2011; Ghose & Ipeirotis, 2007, 2011; Kim et al., 2006; Korfiatis et al., 2012; Y. Liu et al., 2008; O'Mahony & Smyth, 2009). However, these metadata are only available on some particular systems, consequently, it is difficult to use the same type of information for

predicting the helpfulness of product reviews on the world-wide web. Therefore, in this study, the only factors considered are those from the review text's content.

2.2.1 Basic Factors

There are basic factors that are generally assumed to influence the helpfulness of product reviews. Using the frequency of all word types is commonly referred to as a bag-of-words feature. This feature is one of the most basic features for most natural language processing tasks. Since this includes all word type frequencies, using this feature is highly effective for various tasks. At the same time, however, it also has too much noise, meaning the same words can be used both in a helpful review and an unhelpful review.

Kim et al. (2006) included unigram and bigram tf-idf weighted frequencies as features and, through a review ranking experiment, they reported that the unigram frequency is one of the most effective features.

As a basic factor, it can be assumed that the length of the review is important for the helpfulness of reviews. The length of the reviews can be measured by the number of tokens and sentences. This is the most widely assumed feature in previous studies (Cao et al., 2011; Chen & Tseng, 2011; Kim et al., 2006; Korfiatis et al., 2012; Y. Liu et al., 2008; O'Mahony & Smyth, 2009; Zhang & Varadarajan, 2006).

Syntactic categories, such as noun, verb, adjective, etc, can be used as features that influence review helpfulness. Kim et al. (2006) calculated the percentage of parsed tokens, the percentage of nouns and verbs, verbs conjugated with 1st person, and the percentage of adjective or adverb tokens. Zhang and Varadarajan (2006)

used the counts of proper nouns, numbers, modal verbs, interjections, comparatives and superlatives, wh-determiners, wh-pronouns, possessive wh-pronouns and wh-adverbs to predict review helpfulness. Lastly, Y. Liu et al. (2008) encoded the number of words with each part-of-speech tag as a feature. These syntactic category features are also commonly used in the field of natural language processing, not only for review evaluation.

2.2.2 Readability

Another factor indicating how the information is delivered is readability. The readability of a text is to measure how difficult or easy a text is to read. This factor is examined in depth with various metrics.

Cao et al. (2011) investigated the effect of readability, referred to as writing style in the study on review helpfulness by using the average characters per word, average words per sentence, the number of words in pros, cons and summary sections, the number of words in the title, the number of 1-letter words in the review, 2 to 9-letter words in the review and 10 or more-letter words in the review. In the evaluation of the review ranking experiment, they reported that using all the features above was significantly effective when combined with other features. Specifically, the number of 4-letter words and the number of words in cons are positively effective for the review helpfulness and the number of words in titles is negatively effective. Among these, the number of words in cons is more related with the direction of review opinion, not a pure stylistic factor.

Ghose and Ipeirotis (2007, 2011) measured the readability of product reviews with the number of spelling errors in the review, the Automated Readability Index

(ARI), the Gunning-Fod index, the Coleman-Liau index, the Flesch Reading Ease score, the Flesch-Kincaid Grade Level and the Simple Measure of Gobbledygook score for the review. For a detailed explanation of calculating each metric, see DuBay (2004). These metrics are used to calculate the readability of reviews and are grouped into a readability feature group. Through parameter fitting with a linear regression model, and learning a classification model that predicts if a review is helpful or unhelpful, they concluded that readability is one of the influencing factors for review helpfulness. Again, these readability metrics are examined in Korfiatis et al. (2012) by regression models, approximating the review helpfulness. They reported that the readability of reviews has a greater effect than the review length on the helpfulness ratio of a review.

2.2.3 Subjectivity

How subjectively or objectively a review is written is an important factor that can measure how information is delivered. Too negatively or positively written reviews can be intuitively judged as biased, thus influencing the helpfulness of the reviews.

Kim et al. (2006) used positive and negative sentiment words from General Inquirer Dictionary[1] describing products or product features, but concluded that simply counting the word list is not more significantly effective than other factors. Kim et al. (2006) additionally measured the percentage of question sentences and

[1] http://www.wjh.harvard.edu/~inquirer/homecat.htm

exclamation marks and reported they did not show a significant improvement in estimating review helpfulness.

Zhang and Varadarajan (2006) assumed that a good review has subjective judgment on objective observation, so measured the lexical similarity between customer reviews and their product specification, and the lexical similarity between customer reviews and their editorial reviews. Additionally, they measured the lexical subjectivity by using the list of subjective adjectives in Wiebe (2000) and Hatzivassiloglou and Wiebe (2000) respectively and the list of strong subjective nouns and weak subjective nouns in Thelen and Riloff (2002). In order for the learned model to generalize well, the total occurrences of words in each list are used as features. The subjectivity based on lexical similarity and subjectivity dictionaries is shown to have a very limited or minor influence on review helpfulness.

Ghose and Ipeirotis (2007, 2011) calculated the probability of a sentence being subjective by building a classifier following (Pang & Lee, 2004). Instead of classifying the subjectivity of sentences, it used the probability from the classification model and calculated the average of probability for a sentence to be subjective over all sentences with the average probability and the standard deviation of the probability being tested as features. With a regression model taking the helpfulness score as the dependent variable, average probability of sentence subjectivity is proved to have a significant effect on the review helpfulness; highly subjective reviews are perceived as less helpful.

In Chen and Tseng (2011), review subjectivity is measured following (Hu & Liu, 2004) to extract sentiment sentences. The following features are the objectivity feature group in Chen and Tseng (2011):

- The number of opinion sentences, positive sentences, negative sentences, and neutral sentences in a review.
- The percentage of opinion sentences, positive sentences, negative sentences, and neutral sentences in all sentences of a review.
- The percentage of positive sentences and negative sentences in all opinion sentences of a review.
- The cosine similarity between the tf-idf vectors of a review and the product description.

This feature dimension is examined as a group and compared with other factors for review helpfulness and concluded to be one of the most effective factors. The cosine similarity between the tf-idf vectors of a review and the product description is a feature, examined also in Zhang and Varadarajan (2006), but was reported as an ineffective feature. As can be seen from previous studies, the subjectivity of review texts is an important factor for estimating the review helpfulness, which is related with how the information is delivered.

2.2.4 Content

The content of reviews is the factor that this study is most related with due to the assumption that people read product reviews to seek information they need before

making a decision. However, the extraction of the review content is not simple. Firstly, review content is related with the product aspects or features, so extraction of the review content can be achieved by using the list of product aspects. Kim et al. (2006) automatically extracted product features from a Pro/Cons listing from Epinion.com, then counted the number of lexical matches that occured in the review for each product feature. This feature itself performs as well as the unigram bag-of-word feature, according to the experiment result. This indicates how effective using the product aspect word list is in extracting the review content and working for estimating review helpfulness.

In Chen and Tseng (2011), the review content is measured in depth with three different feature dimensions: relevancy, completeness, and appropriate amount of information. The relevancy dimension is based on the assumption that a helpful review should provide a large amount of product information, and the completeness dimension is based on the idea that informative reviews should be wide ranging and cover many different product features and specifications. Lastly, the Appropriate-Amount-of-Information dimension begins with the assumption that a high quality review should include a great deal of product information to help readers judge the value of a product.

- Relevancy: The number of the product names(f1), brand names (f2), website names (f3), and other product names (f4) mentioned; The percentage of times the product name occurs(f5), brand names (f6), website names (f7), and other product names (f8); The number of opinion sentences containing the product name (f9), brand names(f10),

website names (f11) and other product names (f12); The percentage of opinion sentences containing the product name (f13), brand names(f14), website names (f15) and other product names (f16)

- Completeness: the number of different product features (f17), brand names (f18), websites (f19), and product names (f20) mentioned in a review.
- Appropriate Amount of Information: The number of product features (f21), opinion-bearing words (f22), words (f23), sentences (f24), and paragraphs (f25) in a review; The average frequency of product features in a review (f26); The number of sentences that mention product features in a review (f27).

The three review content-related feature dimensions are examined by experiments on review helpfulness estimation. Among the three dimensions, only the Appropriate-Amount-of-Information features are proved to be included in the best feature combinations. Though using the review content-related wordlist is effective in estimating the review helpfulness, it is not practically possible to extract the word lists for all products.

Though O'Mahony and Smyth (2009) suggested measuring the ratio of uppercase and lowercase characters to other characters in the review text and the ratio of uppercase to lowercase characters in the review text to extract the amount of property nouns in the review, which are usually product names, brand names or specific product features, this feature was not judged as highly effective according the feature evaluation result.

Another method of extracting review content is to use topic modeling of sentences, converting the term-document space to topic-document space. The term-document matrix has the problem of containing a zero count for most elements, but making a too high dimensional space, which causes complexity in calculus. This problem is resolved by reducing the high term dimension to a relatively small number of topic dimensions. One of the famous approaches is Latent Semantic Indexing (LSI) (Deerwester, 1988). It uses a mathematical technique called Singular Value Decomposition (SVD). This technique reduces the dimension of terms by summing up terms that occur in a similar context into groups. SVD technically decomposes a matrix (A), which is a term-document matrix in this case, into three matrices; the first matrix (T), a term-topic matrix, describes the original row entities, the second matrix (S), a topic-topic matrix, is a singular value diagonal matrix containing scaling values for the three matrices to be multiplied and thus reconstructed into the original matrix (A), and the third matrix (D), a topic-document matrix, is the original column entities. The second singular value matrix (S) is the topic space that represents the content of documents.

Cao et al. (2011) applied the LSA approach to extract the content of reviews and estimate the helpfulness of reviews. The experiment result shows that among the feature groups of basic features and stylistic characteristic features and the LSA-based content features, the LSA-based approach is proved to demonstrate the best performance on predicting the review helpfulness.

This topic-based modeling for converting the word frequency space into topic space has the advantage of not using a fixed list of words to extract the review content and effectiveness in extracting what information a review delivers. For this

reason, our approach for this study is based on another topic-modeling approach, introduced in a later chapter.

2.3 Summary

This chapter introduces the previous review helpfulness studies. First, we introduced ways of estimating review helpfulness from previous studies. The estimation of review helpfulness can be accomplished by using largely a classification model or a regression model. Due to the practical issues involved in applying the task to real-world review data, we chose to do the regression task for our study. We also categorized the factors used in estimating review helpfulness from previous studies into basic features, such as the length of reviews and the bag-of-word features, readability, subjectivity and content-related features. The factors can be additionally divided into two dimensions: how information is delivered and what information is delivered. Other than content-related features, all other factor categories primarily focus how information is delivered by the reviews. Only the content-related features can extract what information a review delivers. Since the purpose of this study is to show what information a review delivers is more effective than the how information is delivered, the factors from previous studies need to be tested and compared to the approach of this study by a series of experiments.

3 Extracting Information from Reviews

This chapter introduces the entire blueprint of extracting information that product review sentences hold to train a model with findings of the more or less helpful information from product reviews. Firstly, each sentence in product reviews is classified into different information types depending on the target of the information. Within each group of classified sentences, sentences are converted into topic vectors and clustered into similar information-bearing groups. These groups are later used to extract the features related with the information the sentences hold.

3.1 Review Information Types

3.1.1 Motivation

The information in a review text is relatively predictable. Personal experience about a purchased item is one type of information that commonly appears in most reviews. The description about various aspects of a product is usually the most prominent part of a review. Additionally, complaints about shipping service might also appear in a review. Product review information could be categorized by relying on the real world knowledge of products.

One may assume that a certain information type is more preferable than others. For instance, people seek information related with specific aspects of the product rather than how long the shipping takes or the attitude of an employee at the

customer service. It indicates that the helpfulness of product reviews can be estimated by what kind of information the review offers.

Therefore, this study suggests the following Review Information Types (RIT): 1) core, 2) peripheral, 3) background, 4) overall, 5) non-relevant information. Each type of information will be explained in detail with examples.

3.1.2 Introducing Review Information Types

Core Information: This type of information is about the product itself. Reviewers provide their personal experience with aspects of the product to help other customers obtain personalized product information that cannot otherwise be determined before purchase.

Peripheral Information: Reviewers sometimes only complain about shipping or customer service, which is often not helpful for other customers who want to know about the product itself. This information is to be considered separate from core information.

Background Information: Reviewers tend to provide information about their previous experience with other similar products or previous versions of a given product. This reviewer-related information is provided to give credibility to their reviews, which is not directly connected with review quality. However, we assume that people are more likely to trust reviews with background information.

Overall Information: Reviewers offer an overall judgment about the product. General judgment does not seem to help as much as information about specific aspects of a product. Though this information is about the product itself, it is to be considered separately.

Non-relevant Information: Sometimes reviews are filled with sentences that bear no direct relation to the product in question.

For an illustrative example of review information types and real-data examples, see Figure 1 and examples (10) - (13) below.

(10) **Background Information**. This model is my 4th or 5th Kindle. I am stating this after having read three books, so I am not in the “learning curve”. I have had every kindle made, my two sons have Kindles as do my nephews and mom.

(11) **Core Information**. The battery drains faster than previous models I had (even with the screen light AS LOW AS POSSIBLE). The “touch” feature to change pages is not good for me. Kindle support has been good about listening to my issues, but, they are generally unresponsive after that.

(12) **Peripheral Information**. This rating is not about the Kindle, it is about Amazon. It is a nightmare trying to buy a book with the gift card.

(13) **Overall Information**. I thought it would be superb but I am disappointed with it. IT'S GOING BACK TO AMAZON TOMORROW.

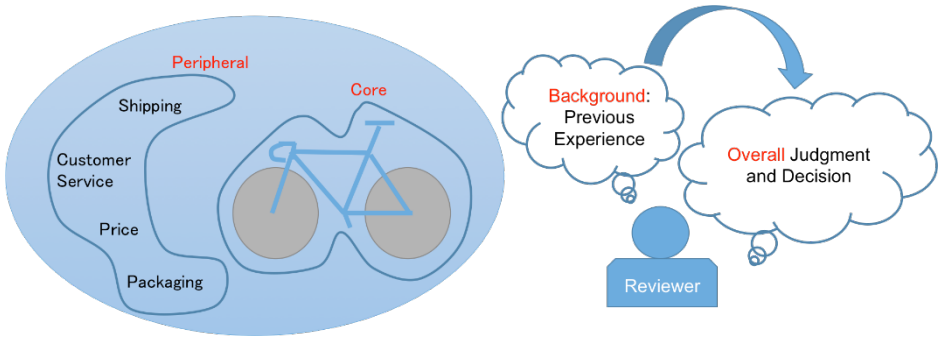

Figure 1. The Illustration of Review Information Types

3.1.3 Difficulties and Ambiguities

Despite the distinction between the information types, there are practical difficulties and ambiguities in judging the information types for review sentences. In real world review data, a sentence sometimes contains more than one information type. In that case, it is practically impossible to automatically pinpoint a range that matches only one information type in the sentence. Thus, this study assumed that one sentence corresponds to only one information type. Since one sentence contains only one information type in most cases, the problematic cases were up to the judgment of the annotator.

Additionally, there are ambiguous cases that are difficult to categorize into one type of information. Background information is considered to be, by definition, sentences involved with the reviewer's previous experience or expertise. See the example sentences in (14).

(14) **I brought it up to northern Minnesota for a camping trip. My friend and I decided to sleep in this tent.** We set it up, which was surprisingly easy, then we went to sleep. That night there was a thunderstorm. We had the rain-fly on all night. I woke up at about 4 a.m, and around the inside of our tent was an inch of water.

These sentences are a part of an outdoor tent review. The bolded sentences indicate that the reviewer had an experience with the product, which could give more credibility to the review. At the same time, these sentences help the reader to understand in what circumstances the tent was tested, relating to the features of the tent. The information in these sentences are ambiguous between core and background. For this case, the purpose of the sentence is not just to give the reliability of the review to the readers, but to explain the circumstance the product was tested and judged. If the bolded sentences are considered as background sentences, the underlined sentences additionally should be considered as background sentences. In fact, the underlined sentences are rather about the circumstance of using the product, helping to better understand aspects of the product. Therefore, the bolded sentences are treated as core information, considering the prior purpose of the sentences.

Therefore, background information is restricted to only those sentences which indicate the reviewer's expertise, his previous experiences with similar products, or rich experience on the product, dedicated to giving credibility to their reviews.

The overall information includes the overall judgment, summaries about the product, the final decision of another purchase, or the recommendation to other consumers. It is expected that the words from overall information type sentences

will overlapped with the words from core information sentences. The reason for making the distinction between the core and overall information is that the effect of overall information sentences is different from that of the core information sentences. The example sentences annotated as overall judgments are shown from (15) to (18), categorized depending on their characteristics. Reviewers sometimes give the overall judgment about the product at the beginning as in (15), then describes specific support on the judgment. This type of knowledge is usually short, less than two sentences in most cases. However, if a reviewer repeats the overall judgments over and over, this does not make the review helpful. The same is true for the recommendation, summary, and final decisions. On the other hand, sentences explaining specific aspects of the product in detail contribute to the overall helpfulness of the review. This is the reason for separating theses sentences from the core type, though the words in (15) and (17) can appear in core type sentences as well.

(15) Overall Judgments at the Beginning or the End.

Overall it has been a good tent.

This tent is good, if it isn't raining.

I give this tent 4 stars for the following reasons.

(16) Recommendation to Reviewers

I would recommend this to a friend.

Overall I would highly recommend this tent for weekend getaways.

(17) Summaries

Bottom line: This seems like more of a nice/warm climate tent (beach, desert, etc.), rather than a cool mountains/forest tent.

(18) Final Decision

I've rated this tent 5 stars, even though I need to return it, because it broke after first use.

3.2 Finding Similar Information-bearing Sentences

Though information types are given to every sentence, the quantity of each information type in reviews is not sufficient to evaluate the helpfulness of product reviews. Within the same information type, the specific meaning of sentences should be extracted. For instance, knowing how specifically the review deals with the clarity of the display, the duration of the battery life, or the accuracy of the touch screen contributes to the estimation of the helpfulness of product reviews on an e-book reader. These specific meaning of sentences can be found through various approaches.

Information extraction from sentences is closely related with the field of formal semantics and syntax in linguistics. By using predicate logic or syntactic structures, linguists not only try to find the structure of the language, but also the language independent representation of sentences. With an ideal linguistic representation, the similarity between the meanings of sentences can be captured. However, it is practically impossible to apply theoretical representation to all natural language sentences.

To extract the meaning of sentences, this study examined various approaches from the keyword-based approach to topic-based representation approaches.

3.2.1 Sentence Representations

Other than the keyword-based approaches mentioned above, the most naïve approach is using the frequencies of word types in a sentence to extract the meaning of sentences, commonly referred as the bag-of-words approach. It assumes that the meaning of sentences can be represented by the amount of times words occur in a sentence. Then the similarity between sentences is measured by the co-occurring words in the sentences. However, the pure frequency of words in a sentence is not appropriate for representing the meaning of the sentence due to the fact that the degree of contribution to the whole meaning of the sentence is not the same for every word in the sentence. For this reason, a weighting method is applied to the pure word frequency vector.

3.2.1.1 TFIDF

Among words in a sentence, some words have more importance in the meaning of the sentences than others, such as articles, prepositions, etc. These words are referred to stopwords. These words are extremely common in natural language, so the frequencies of those words are higher than other important words, which in turn deteriorates the meaning of sentences based on word frequency.

The term frequency-inverse document frequency (TFIDF) is a numeric statistic to weight the importance of a word in a document. It is commonly used to filter out stopwords and find more important terms in a document. For instance, “the” is extremely common, so its importance is over-emphasized. Thus, the inverse document frequency decreases the weight of words that occur frequently in all

documents and increase the weight of words that rarely occur. It is based on the assumption that unimportant words occur frequently across all documents, but important keywords in a document occur only in the document. There are variations of calculating the tf-idf weight, but in this study we utilized the most common algorithm, given below:

$$\text{TFIDF WEIGHT} = f_{t,d} \times \log(1 + \frac{N}{n_t})$$

EQUATION 1. TF-IDF FORMULA

In the Equation 1, $f_{t,d}$ is the number of times that the term (t) occurs in the document (d), the term frequency. N is the total number of documents in the corpus and n_t is the number of documents in which the term (t) occurs.

By applying the tf-idf weight to the raw frequencies of words, the meanings of sentences can be represented in a more accurate way. This weighting is originally calculated with documents, however, in this study, we extract information from separate sentences, so the tf-idf weight is calculated with the assumption of treating each sentence as a document.

3.2.1.2 Latent Dirichlet Allocation

The problem with representing the information of a sentence with the frequencies of words is that short sentences have 0 values for the most word dimensions. When checking if two sentences bear similar information or meaning, there is no way to find the relationship between these sentences if there are no overlapping words. Thus, we converted frequency word vectors to topic vectors; in

turn each sentence can have the same topic dimensions regardless of the length of the sentence. For this conversion, the Latent Dirichlet Allocation (LDA) (Blei, Ng, & Jordan, 2003) is applied to word frequency vectors by using the Gensim library (Sojka, 2010).

LDA enables us to convert the meaning of a sentence from a bag-of-words or tfidf-weighted space into a latent space of a lower dimensionality, which is a collection of a pre-defined number of topics. The LDA model assumes that each document is a mixture of topics and that each word in the document in chosen from one of the topics, each of which is composed of words. This is commonly referred to as a generative model in that writers enumerate words out of topics to write a document. Due to the generative assumption, it becomes possible to find the relatedness between documents that do not have words in common. Similarity can be found if two different words belong to the same topic.

The whole of the collected reviews for each product domain are used to train each LDA model. The result, based on LDA conversion, can be compared with that of a simple tfidf-weighted representation of the sentences.

3.2.1.3 The Sentence Range

Though the conversion from simple frequency to topic dimension can improve the quality of extracting information from sentences, LDA still relies on the words that occur in the sentences. It is reasonable to assume that the information of the current sentence relies on the previous or following sentences, that is, the contextual information. In fact, without context, it can be difficult to understand the meaning of the current sentence. Thus, the previous and following sentences can be

added to the current sentence, expanding and enriching the information of the current sentence. Finding the best representation approach will be shown in next chapter through experiments.

3.2.2 Clustering Similar Information-bearing Sentences

Within each information type of sentences, the tfidf-weighted or LDA topic-based sentence vectors are clustered to find more specific meaning groups. Though a sentence is classified into the core information type, the helpfulness of the sentence varies depending on the specific target of the information. For instance, consumers want to know about the readability of the e-book reader more than the appearance of the frame material. Among various clustering methods, DBSCAN (Ester, Kriegel, Sander, & Xu, 1996) and K-means (Arthur & Vassilvitskii, 2007) clustering algorithms are tested to find sentence clusters that are considered to deliver similar information for the different characteristics of each algorithm.

The K-means algorithm begins with the k initial “means”, k data points, randomly chosen among the data points. Then every other observed data point’s cluster is decided by finding their nearest mean. Within each cluster, the “means” are moved to the centroid of each cluster. The steps are repeated until convergence has been reached.

The distance metric of KMean is the Euclidean distance, measuring the distance between points. Though the similarity between vectors is often calculated with the cosine similarity, which measures the directional difference between two normalized vectors, K-means finds sentence clusters based on the Euclidean distance between sentences. The number of clusters is experimentally decided,

assuming the best clustering model results in the best review ranking result. The experiment used to decide the cluster number will be introduced in later experiment chapter.

Different from K-means, the Density-Based Spatial Clustering of Applications with Noise (DBSCAN) treats clusters as high-density areas separated by areas of low density (Ester, Kriegel et al. 1996). This algorithm makes a cluster of points that are closely located on the vector space, at the same time making outlier points that are in low-density regions.

DBSCAN needs ε(eps) and the minimum number of points (minPts) to form a dense region. It begins with a random data point and searches its neighbors within the distance of eps. If the number of discovered points is above the minPts, a cluster is generated. Otherwise, the point is treated as noise. If a point is in a dense area of a cluster, its ε–neighborhood also belongs to the cluster. Therefore, all data points with the ε–neighborhood are treated as the same cluster. Then another unvisited point is chosen and used to build another cluster or noise. The process repeats until all data points are visited and their clusters are found or labeled noise. Due to the method of finding clusters, the shape of clusters can be any shape. Also, different from K-means, it is robust to outliers.

DBSCAN algorithm can choose different distance metrics depending on the algorithm to decide the nearest neighbors. The brute-force algorithm is chosen by using the cosine distance metric. The eps and minPts are also experimentally decided.

3.3 The Summary of the Extracting Procedure

The whole procedure of extracting information from sentences is illustrated in Figure 2. It begins with review documents. The information classifier or clustering algorithm categorizes each sentence into a different information type following RITs. Then each sentence is converted into a word frequency vector or topic space vector. Within each sentence information type, the transformed sentences are clustered to find the similar information-bearing groups. These trained groups are used later to extract the information from review sentences in the test data.

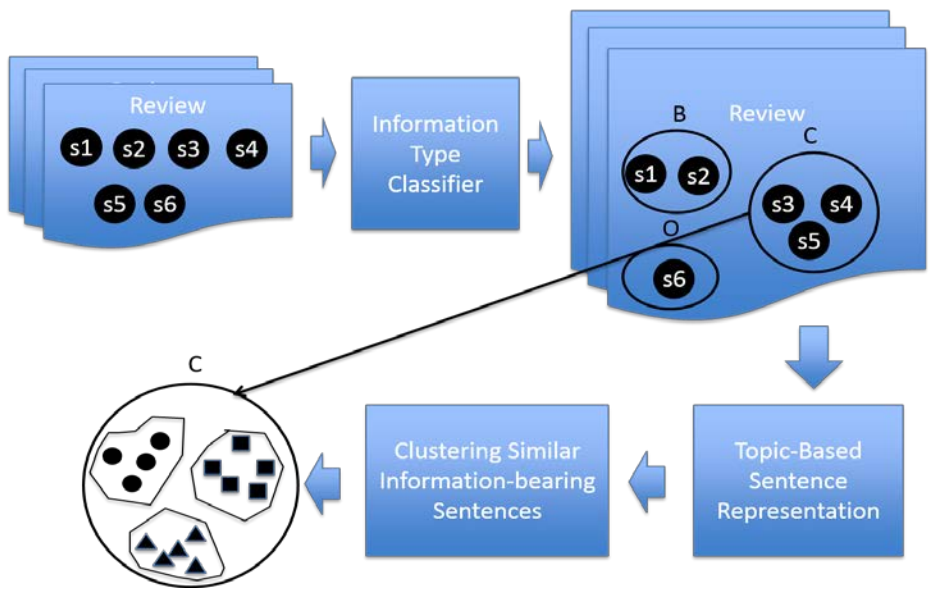

Figure 2. The Illustration of Extracting Information from Review Sentences

∗B, C, O respectively stands for Background Core, and Overall information.

Similar-information holding clusters for each information type is used to label what information each sentence holds with the cluster number. The following example shows the process of extracting what information a review delivers.

(19) The example of extracting what information is delivered in a review

This model is my 4th or 5th Kindle. (B-15)

I thought it would be superb but I am disappointed with it. (O-36)

For this two reasons:- The battery drains faster than previous models I had (even with the screen light AS LOW AS POSSIBLE). (C-38)

The advertising said the battery duration was especially good despite of the screen light. (C-38)

The "touch" feature to change pages is not good for me. (C-35)

The slightest accidental touch of the screen while reading, changes the pages (one or more, back or forward) and bothers a lot the reading experience. (C-35)

And/or changes the font size. (C-40)

I am stating this after having read three books, so I am not in the "learning curve". (B-19)

Regarding the rest, Kindle is a wonderful product which I couldn't live without it. Also its quality is very good. (O-4)

Horacio Venturino (N)

The information of the review:

(B-15 : 1), (B-19 : 1), (C-38 : 2), (C-35 : 2), (C-40 : 1), (O-4 : 1), (O-36 : 1)

The example is an e-book reader review. The “C” in “C-35” indicates what information type each sentence belongs to, the core information, and “35” refers to

what similar-information bearing cluster the sentence belongs to. Each sentence is classified according to what information type each sentence belongs to determined by the trained review information type classification model. Then it is labeled its similar-information-bearing-cluster number within each information type so that the whole review can be represented as the amount each review information type appears in the review and specifically what information the review gives.

4 Preparing Product Reviews

4.1 Collecting Data

The collected products reviews are used throughout a series of experiments in this study. We collected 19,153, 7,834 and 9,175 reviews on e-book readers, outdoor tents and jeans, respectively. To crawl the data from Amazon.com, a crawler was used[2]. The product domains were chosen to ensure the robustness of the experiment results. The summary of the collected reviews is in Table 1. For the kinds of products, see Appendix I.

The reviews were collected with some usable meta-data as follows.

- A counter of the reviews extracted so far (to be used as a unique ID for the dataset).
- Date of the review in YYYYMMDD format. (Note: In non-English domains, this feature won't work without edit the script to set the names of the months in the desired language).
- Date of the review in human readable format (in the language used by the specified domain).
- ID of the reviewed product.
- ID of the author of the review.
- Star rating assigned by the reviewer.

[2] Available at https://github.com/aesuli/Amazon-downloader

- Count of "yes" helpfulness votes.
- Count of total helpfulness votes ("yes" + "no").
- Title of the review.
- Content of the review.

Among these, the proportion of the count of “yes” helpfulness votes to the count of total helpfulness votes is used to calculate the helpfulness score (h_v).

$$helpfulness\ vote\ score = \frac{the\ count\ of\ \text{"yes" helpfulness votes}}{the\ count\ of\ total\ helpfulness\ votes\ (yes + no)}$$

EQUATION 2. HELPFULNESS VOTE SCORE (H)

For each product domain, 200 reviews were chosen for training and testing for the experiments. All the reviews for each domain were used to pre-train the LDA model to convert review sentences into LDA-based topic vectors.

Table 1. Collected Data

Product Domain	e-book Reader	Outdoor Tent	Jeans
Total Number of Reviews	36140	7834	9175
Total Number of Products	2	10	10
Number of Annotated Reviews	200	200	200
Number of Products in 200 Annotated Reviews	1	8	5

4.2 The Review Helpfulness Vote Score

The helpfulness vote score (h_v) is convenient to use as the gold score to train models for estimating the review helpfulness. In fact, many studies have used the score (Ghose & Ipeirotis, 2007, 2011; Kim et al., 2006; Korfiatis et al., 2012; Y. Liu et al., 2008; Zhang & Varadarajan, 2006). Though each review has the helpfulness vote score (h_v), the score is reported to be biased in many ways (J. Liu et al., 2007). The study reported three types of biases: imbalance vote bias, winner circle bias and early bird bias.

The imbalance vote bias is where Amazon users tend to judge the helpfulness of the review positively rather than negatively. In Figure 3, among 23,141 reviews, half of them have more than 90% helpful votes. It also reports there are reviews with a helpfulness score of 1.0, but are not, in fact, helpful.

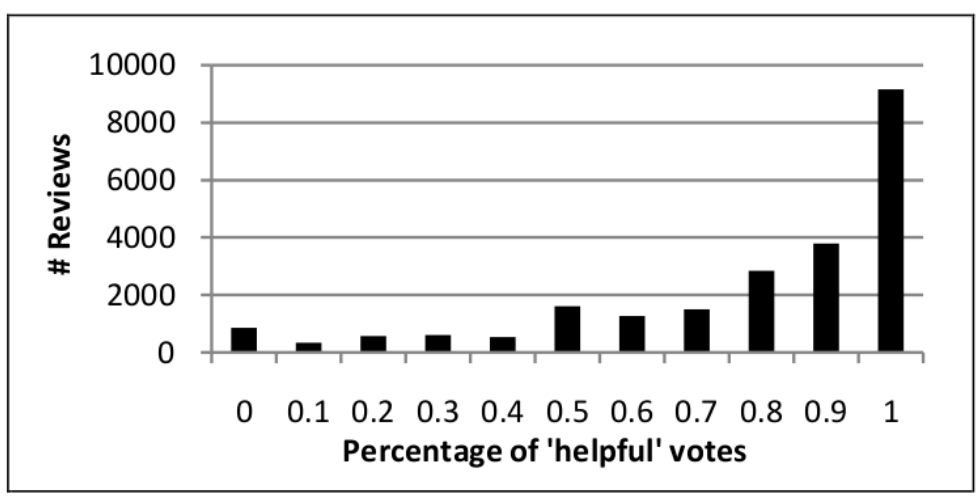

Figure 3. The Reviews' Percentage Score (J. Liu et al., 2007)

The winner circle bias is where reviews with more votes are likely to appear to users more often than others so that they get even more votes, influencing the objectivity of readers' votes. Figure 4 describes the votes held by the top 50 reviews from 127 digital cameras. The top 2 reviews held 250 and 140 reviews respectively. The number of votes for other reviews rapidly decreases. Among 19153 e-book reviews for one product, only 561 reviews have more than 10 helpfulness votes, and more than 30% of reviews do not have any votes.

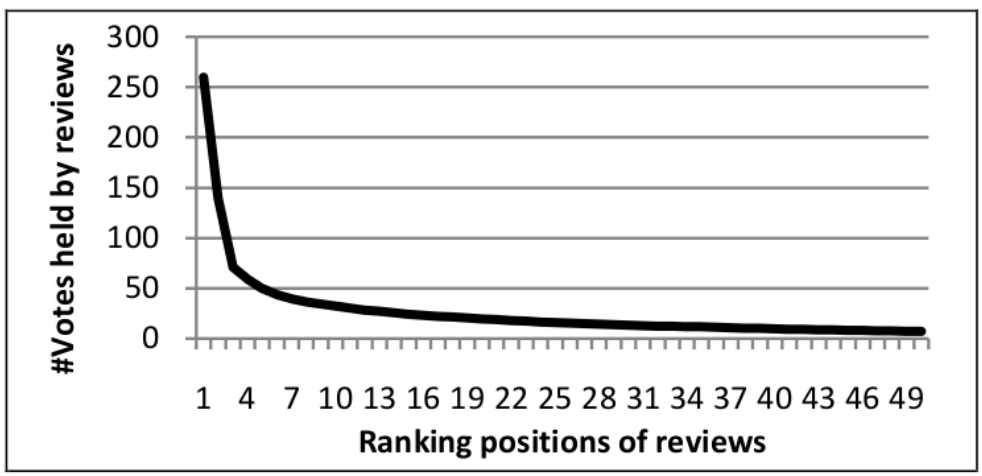

Figure 4. Votes of the Top-50 Ranked Reviews (J. Liu et al., 2007)

The last bias is early-bird bias, where reviews written earlier are likely to get more votes than other reviews. Figure 5 shows the *n*th month after the product is released and the number of votes held by the reviews. As seen in the figure, early reviews get more votes because the early ones are exposed to readers for a longer time.

To avoid the bias problems of helpfulness votes, some studies have tried to manually annotate the scores of reviews. Liu et al. (2007) turned the problem of the biased helpfulness score into building four review quality categories: *best*, *good*, *fair* and *bad*. Chen and Tseng (2011) also raises the problem of the gold standard and manually categorized reviews into *high-quality*, *medium-quality*, *low-quality*, *duplicate* and *spam*. In turn, the review ranking problem is turned into a multi-class classification problem.

However, multi-class classification models are difficult to apply practically to real-world data since the number of reviews for each category is so large that users cannot collectively see only helpful reviews. For this reason, this study builds manual helpfulness scores for 600 reviews: 200 e-book reader reviews, 200 tent reviews and 200 jeans reviews.

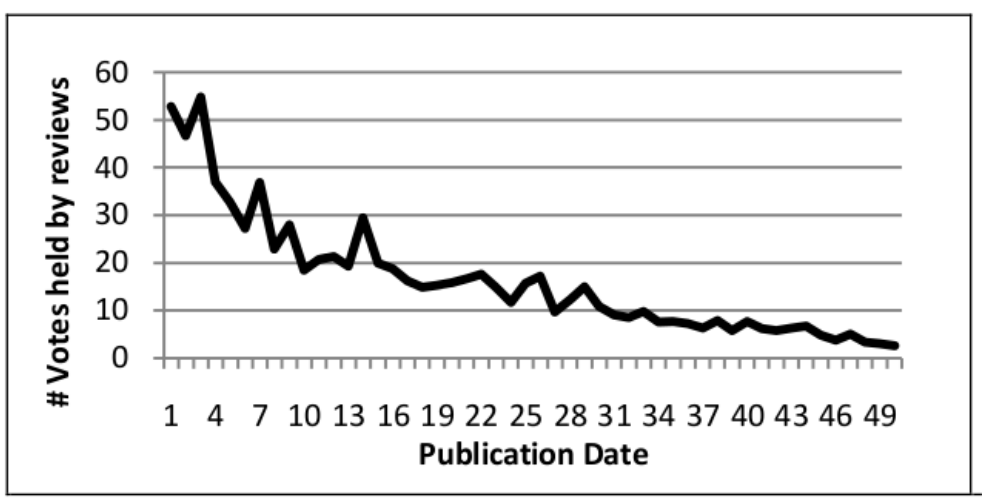

Figure 5. Dependency on Publication Date

4.3 Building Review Helpfulness Manual Score

4.3.1 Annotating Manual Helpfulness Score

Among the two e-book readers, 200 reviews from only one e-book reader were chosen to be manually annotated for their gold scores. For the outdoor tent and jeans reviews, 200 reviews for each domain were randomly chosen based only on the length of the reviews regardless of products, which were originally collected for 10 different products for each domain. Thus, the number of products for the randomly chosen 200 reviews is less than 10 products as in Table 1; reviews from some products are not included. The reason for choosing 200 reviews from only one product for only the e-book reader domain was to compare the manually annotated score (h_m) with the helpfulness vote score (h_v) from Amazon.com. The helpfulness vote scores from different products, though they are in the same product domain, cannot be considered to have the same value. From the comparison of two different scores, we can show the need for building a manual helpfulness score.

The ideal way of building manual review helpfulness scores is to give annotators pairs of reviews and instruct them to indicate which reviews are better. From the annotation, votes for a better review can be converted to a score: dividing the number of winning votes by all votes. However, as mentioned above, the number of all possible pairs is too large for annotators to work on. For example, if there are 200 reviews, 19,900 review pairs will need to be annotated. It is practically impossible, because in real-world data, the number of training and testing reviews

goes far beyond 200. The alternative way of building manual gold data was to have annotators score reviews and use the average scores.

Six annotators were instructed to follow the annotation guideline provided below.

(20) Annotation Procedure

- Step 1. Search the product description page with product ID. (If you have already read the product description page, you don't have to repeat this step.)
- Step 2. Read the product description page before reading the review. You can find the product description pages by searching with the given IDs.
- Step 3. Read an example of a good review and a bad review.
- Step 4. Read the reviews (50 each).
- Step 5. Give the 3 scores (1~7) for each review: Completeness, Effectiveness, and Persuasiveness.

The separation of the score into three distinct dimensions is to avoid inconsistency among annotators when scoring the reviews. Further explanation about the score dimensions were given to annotators as seen in (22) - (24). In addition, since the 7-score scale scoring can be different for each annotator, more specific guidelines for each scale point was given as in (21).

(21) The Specific Guideline for 7-score Scale

Completeness: Very Incomplete (1), Quite Incomplete (2), Somewhat Incomplete (3), Neither incomplete nor complete (4), Somewhat Complete (5), Quite Complete (6), Very Complete (7)

Effectiveness: Very ineffective (1), Quite ineffective (2), Somewhat ineffective (3), Neither ineffective nor effective (4), Somewhat effective (5), Quite effective (6), Very effective (7)

Persuasiveness: Very unpersuasive (1), Quite unpersuasive (2), Somewhat unpersuasive (3), Neither unpersuasive nor persuasive (4), Somewhat persuasive (5), Quite persuasive (6), Very persuasive (7)

(22) Completeness

A good review has to sufficiently contain personalized experiences and thoughts about the product for all various aspects, from the appearance to the specific aspect of the product.

(23) Effectiveness

It is important to describe the product aspects in detail, but it should be written effectively to be a good review. Long reviews that do not contain much helpful information are inefficient, repeating the same information through too many sentences or offering information that users are not interested in. This score dimension is not related with the grammaticality or style, but only the effectiveness of delivering information with regard to its length.

(24) Persuasiveness

A good review persuades readers to buy or not buy the product. If the reader follows the reviewer's opinion and decision, the review is persuasive. The persuasiveness can be affected by the reviewer's writing style (word choice, sentence length, structure, attitude towards the product or company, etc), language proficiency or his previous experience with other related products.

Separating the score dimensions not only ensures the scoring consistency between annotators but also allows the inclusion of all possible factors to estimate the review helpfulness. In addition, separating score dimensions helps to increase the total score for each review so that the distribution of review helpfulness scores becomes continuous. Since the 7-score scale dimension is ordinal, summing the all score dimensions does not guarantee the continuity of the score, as in the case of height or temperature. Nonetheless, as the number of annotators increases, the average of helpfulness scores across all annotators can become approximately continuous and suitable for regression modeling.

4.3.2 Evaluation of Review Helpfulness Manual Score

To evaluate the annotated scores for reviews, the ranks of the reviews based on the annotated scores are compared using Spearman’s rank correlation coefficient (ρ: the Greek letter *rho*), which is a non-parametric measure of statistical dependency between two continuous or ordinal variables.

$$\rho_{rg_X, rg_Y} = \frac{cov(rg_X, rg_Y)}{\sigma_{rg_X} \sigma_{rg_Y}}$$

EQUATION 3. THE SPEARMAN'S RANK CORRELATION COEFFICIENT

Where

- ρ denotes the Pearson correlation coefficient, applied to the rank variables.
- $cov(rg_X, rg_Y)$ is the covariance of the rank variables.
- σ_{rg_X} σ_{rg_Y} are the standard deviations of the rank variables.

This metric uses the difference between the ranks of the same review from two annotators. The score ranges from -1 to 1: -1 (100% negative association, or perfect inversion) to +1 (100% positive association, or perfect agreement). In addition, the score is regarded as *very weak* (.00-.19), *weak* (.20-.39), *moderate* (.40-.59), *strong* (.60-.79), and *very strong* (.80-1.0).

By manually annotating gold review scores, we discovered that Amazon's helpfulness vote score (h_v) differs quite significantly from what score a person might actually give a review. The comparison between h_v and h_m is only possible for e-book reader reviews since the h_v scores are reliable only if they are from reviews on the same product. The *rho* score between the ranks of e-book reader reviews based on the helpfulness score (h_v) and the manual score (h_m) was 0.481 (p-value: 5.135e-13). Additionally, the Kendall's tau-b score (Agresti, 2010; Kendall, 1938), which measures how much the order of all review pairs correspond between two rankings, was 0.326. These evaluation metrics show that the ranks of reviews based on h_v and h_m are significantly different, which can result in a quite

different review ranking model. This supports the need to build a new manual helpfulness score for product reviews.

However, there appeared to be difficulties and problems related to building manual helpfulness scores for reviews. First, it is important to note that 200 reviews cannot have 200 distinct ranks. It is intuitively reasonable to suppose that there exist same-ranked reviews. That is, there should be reviews with tied scores. This actuality can be found in both the Amazon helpfulness vote score (h_v) and also in our manually built scores (h_m): 123 ranks with h_v and 97 ranks with h_m among 200 reviews.

Moreover, the criteria for scoring the helpfulness of reviews can vary depending on the annotator. As seen from the evaluation results in Table 2, the Spearman correlation score of review ranks between annotators for e-book reader reviews is 0.7 (highest), 0.46 (mean: *moderate* grade) and 0.158 (lowest), which are bolded in the table. It seems that Annotator 6 in particular shows lower agreement compared to the other annotators. However, it is difficult to set a threshold between the usable or unusable annotated data with this score because there is no definite correct ranking for the reviews. Also, in terms of Spearman's score grade, the 0.46 mean score between all annotator pairs is *moderate*, thus we decided to include all annotation result for e-book reader reviews.

For tent and jeans reviews in Table 3 and Table 4, the average Spearman's correlation was 0.270 and 0.311, both of which are *weak* grade. Annotator 2 (A2), colored in bold, was consistently showing a low correlation with other annotators. Thus, the annotations from A2 are excluded from calculations of the manual score. By removing these annotations, the average Spearman score for the product review

of both domains are raised to 0.35 (weak) and 0.40 (moderate), respectively, for tent and jeans reviews.

The reason for the inconsistency of scoring comes as an inherit problem with the scoring task. To begin with, annotators forget their own criteria as they score 200 reviews. Even though an annotator considers a review (r1) more helpful than another (r2), r2 can be given a higher score than r1 because of the distance (the number of intervening reviews) between r1 and r2.

Table 2. The Spearman Rank Correlation between Annotators (e-book reader)

	A2	**A3**	**A4**	**A5**	**A6**	**Avg**[a]
A1	0.605	0.450	**0.700**	0.623	0.439	0.563
A2		0.506	0.634	0.425	0.307	0.495
A3			0.459	0.438	**0.158**	0.402
A4				0.562	0.286	0.528
A5					0.318	0.473
A6						0.301
					Total Average:	**0.460**

[a] The score in the average column is an average for all scores related to the annotator of the row

Table 3. The Spearman Rank Correlation between Annotators (tent)

	A2	**A3**	**A4**	**A5**	**A6**	**Avg**
A1	**-0.039**	0.43	0.242	0.201	0.417	0.265
A2		**0.088**	**0.165**	**0.174**	**0.077**	0.108
A3			0.363	0.305	0.304	0.298
A4				0.557	0.382	0.341
A5					0.319	0.311
A6						0.299
			Average without A2:	0.352	Total Average:	**0.270**

Table 4. The Spearman Rank Correlation between Annotators (jeans)

	A2	A3	A4	A5	A6	Avg
A1	**0.002**	0.441	0.42	0.447	0.328	0.327
A2		**0.212**	**0.094**	**0.198**	**0.128**	0.126
A3			0.38	0.382	0.412	0.365
A4				0.395	0.483	0.354
A5					0.352	0.354
A6						0.340
			Average without A2:	0.404	Total Average:	0.311

Another reason for the inconsistency in scoring is that everyone has different criteria for what makes a helpful review. This is seen in another evaluation. We made four sets of 10 reviews that have continuous ordered ranks (rank 1-10, 20-29, 30-39, 40-49) in the e-book reader domain. Each set was given to three annotators, who were told to rank each review in the set from 1 to 10. We expected it to be possible to give an accurate ranking for only 10 reviews and expected a remarkably high agreement between annotators, but we found the average tau-b score for all pairs of 10-review sets was just 0.35, the highest being 0.73, and the lowest was 0.02, as shown in Table 5. Though the overall agreement between annotators from the direct 10-review ranking is higher than the one from the manual helpfulness score (h_m), there exists the low tau-b scores between annotators.

Table 5. The Agreement on the Ranks of 10 Reviews between Annotators

	1~10	20~29	30~39	40~49	avg
a1-a2	0.02	0.73	0.73	0.60	0.52
a1-a3	0.06	0.20	0.51	0.28	0.263
a2-a3	0.24	0.11	0.51	0.24	0.275
avg	0.106	0.346	0.583	0.373	

Through a series of evaluations of manual scores for reviews, we found that the judgment of helpful reviews can largely vary depending on individuals, and the agreement on the ranks based on the manual helpfulness score (h_m) between annotators is not low, compared to the results of direct ranking evaluations.

Finally, it is important to point out, despite the various issues and difficulties, building gold scores for review helpfulness ranking will lead to building a more accurate review ranking model than the available Amazon vote scores (h_v). The two different scores (h_v and h_m) are examined with experiments to see the different effects of features due to the score difference.

4.4 Annotation of Information Types

Additionally, the information types of review sentences need to be annotated to be used for training a model to recognize the review information type of sentences. The proportion of information types in Table 6 indicates that core information is the most dominant information class. The proportion of background information is similar with overall information, and peripheral information has the smallest proportion. None type (non-relevant) sentences takes the smallest proportion among product types. Since the number of none types is too small to include as one category, it is excluded from the information types used for the experiments. Recognizing the non-relevant text from product reviews is different from dividing other information types of product reviews. Also, the error type sentences are errors that are automatically tokenized as sentences by the sentence tokenizer from NLTK (Bird et al. 2009). These errors are usually just one of repeating exclamation

markers or some text particles that have nothing to do with review helpfulness, so they are also ignored for this task.

One interesting property of the information types in Table 6 is that there is a tendency that the proportion of each information type is constant through all product domains. It indicates people write product reviews keeping the same ratio of information types.

Table 6. The Number of Information Types

	e-book Reader	**Tent**	**Jeans**
Background	979	480	351
Core	3666	3999	1028
Peripheral	767	249	132
Overall	1060	518	380
None	88	23	5
Error	87	17	1
Total	6647	5286	1897

The data was annotated by one experienced annotator and the results were compared with another annotator to validate the annotated data. To evaluate the agreement between annotators on judging review information types, 30 sentences are randomly chosen for each information type, thus 120 sentences are selected for each product domain. The results of annotation agreement are shown below. The Kohen's Kappa agreement score for each product domain was 0.633, 0.678 and 0.589 respectively for e-book reader, tent and jeans reviews, indicating the degree of agreement on annotating review information types is "substantial" for e-book reader and tent reviews and "moderate" for jeans reviews, following Landis and

Koch (1977). The agreement results below show that *overall* and *peripheral information* types are generally more miscategorized than *background* and *core information* type sentences.

Table 7. Agreement for judging information types of e-book reader reviews

	background	**core**	**overall**	**peripheral**
background	26	2	1	1
core	5	25	0	0
overall	5	5	18	2
peripheral	5	7	0	18

Table 8. Agreement for judging information types of tent reviews

	background	**core**	**overall**	**peripheral**
background	26	2	2	0
core	1	28	1	0
overall	2	4	24	0
peripheral	10	4	3	13

Table 9. Agreement for judging information types of jeans reviews

	background	**core**	**overall**	**peripheral**
background	26	3	1	0
core	7	21	2	0
overall	11	3	15	1
peripheral	7	1	1	21

4.5 Summary

This chapter reports the process of collecting product review data and annotating review helpfulness score and review information types. We have crawled product reviews on three different product domains from Amazon.com: e-book reader, outdoor tent and jeans. Among all the reviews, only 200 reviews are chosen for

each domain for annotating the review helpfulness and their sentence information type.

The annotation of review helpfulness score for 200 reviews for each product domain is conducted by six annotators. They were guided to give a 1~7 scale score for each dimension of completeness, effectiveness and persuasiveness. The reason for dividing the score dimensions is to assign continuous helpfulness scores to reviews and to ensure more consistent judgment on review helpfulness.

With the evaluation of helpfulness scores between annotators, we could achieve a satisfactory correlation between review ranks from annotations for e-book reader and jeans reviews, but found a weak average correlation score for tent reviews. Additionally, the evaluation of direct ranking of only 10 reviews, which was in fact expected to show high agreement, turns out to be as low as the scoring judgment result, showing the difficulty of achieving high correlation score for scoring review helpfulness. However, this kind of problem can be resolved as the number of annotators increases.

Moreover, the review information types were annotated by annotators and their agreement was evaluated to find to what extent it is possible to have a consensus on annotating review information types of sentences. The results reveal that the review information types can be annotated with substantial agreement, indicating necessity of dividing the review information types.

5 A Preliminary Study: Introducing Background Information Type for Product Review Helpfulness

5.1 Task Description

Before examining all types of review information types, this preliminary study was conduct to see how much extracting background information from product reviews based on the topic-based approach can help to estimate the review helpfulness, compared to other studies and other possible naïve approaches.

The task at hand is to rank the reviews according to some automatically estimated score or criteria in terms of their helpfulness. For the helpfulness score, h_m was used. This can be done by two different approaches: by performing a pair-wise comparison of two reviews or by ordering all reviews by a score. The former transforms the matter of ranking to a classification problem. It is maybe more accurate to decide which review is more helpful than the other. However, it is practically impossible to calculate the pair-wise classification whenever the new review is written. Additionally, building the gold standard for this classification problem requires too much time and human resources due to the huge amount of reviews. Therefore, ranking should be trained and tested using regression models. With a usual regression model, the independent variables are restricted to continuous models, which in turn restricts the types of features used to train the model. For this reason, Support Vector Regression (SVR) model (Vapnik, 1995) was chosen for training and testing to use various variable types of features from

sentences. The SVR model will be introduced in more detail in a later chapter. All the clustering and regression models, Scikit-learn, were also used (Pedregosa et al., 2011).

5.2 Data Collection

The data for this preliminary study is restricted to only product reviews from one e-book reader due to the limitations of the comparable approaches using pattern matching. Since this part of the data was introduced in Chapter 4, it will simply be re-introduced in this chapter. When we examined the effect of background information, only e-book reader data had been collected. The collected reviews totaled 36140, including two e-book readers. Among these reviews on two products, 200 reviews were chosen from only one product review. The summary of used review data is shown in Table 10.

Table 10. Product reviews used for this preliminary study

Product Domain	e-book Reader
Total Number of Reviews	36140
Total Number of Products	2
Number of Annotated Reviews	200
Number of Products in Annotated Reviews	1

5.3 Extracting Background Information

This experiment examined three different ways of extracting information from sentences, from the most naïve pattern recognition to a topic-based approach. From the product reviews of an e-book reader, it was noticeable that the kinds of

background information that a reviewer offers to other consumers is restricted. As noted from the examples of review information types in Chapter 3, reviewers are likely to give information about his previous experience on other related products, how long he has experienced the product, or his own particular circumstances, all of which can give more credibility to reviews. These kinds of information can be captured in various ways: pattern matching, a seed-based similarity approach, and a topic-based approach.

5.3.1 Pattern Matching for Background Information

Despite its inefficient design, pre-coded regular patterns can be used to extract background knowledge quite naively from sentences bearing background information. In total, 25 regular expressions were built to find background information in reviews and the findings were encoded as features, as in Table 11. Though this method can recognize the background information with high precision, the recall of recognition would depend on the regular expressions. For this pattern matching method, the information types of the sentences are not given, and solely depend on pattern matching. The quality of recognition is too dependent on the builder of the patterns. Furthermore, it is nearly impossible to build these patterns for all product domains. The entire list of regular expressions used for pattern matching can be found in Appendix II.

Table 11. Summary of Background Knowledge with Regular Patterns

Type	Summary
Previous Experience	The experience on the use of previous versions (f1), competitor products (f2), or similar products (f3)
Degree of Product Experience	A short (f4) or long (f5)[a] period of use, understanding on the product (f6), and whether or not the reviewer returned the product (f7)
Intention of Purchase	Whether or not a reviewer has purchased the product as a gift (f8) or bought it for themselves (f9) or someone else (f10) are encoded as features. In addition, a feature that indicates whether or not the reviewer recommends the product is encoded as (f11).
Helpfulness	The number of helpful (f12) or unhelpful features (f13) is counted by summing all helpful or unhelpful features, respectively.

[a] A short period of time (f4) is defined as less than a month.

5.3.2 Seed-based Information Extraction

The seed-based information extraction method is a mixture of pattern matching and a topic-based approach. The seed sentences, which respectively represent each type of background knowledge, are the collection of sentences matched by pattern matching. The length of seed sentences was limited to 30 words.

The seed sentences and sentences in the training data are converted into Latent Dirichlet Allocation (LDA) topic vectors with a pre-trained LDA model on the product domain. Calculating the cosine similarity between a seed sentence and every review sentence lead to similar background information sentence groups. The similarity cutoff threshold was set to the 99.5th percentile of all similarity scores. This approach enables us to find intuitively similar information-bearing sentence

groups. These groups are directly used to find what information each background sentence holds.

5.3.3 Topic-based Information Extraction

The topic-based information extraction approach is the one introduced in Chapter 3. Given the information type of each sentence, the background sentences in the training data are converted to LDA topic vectors. Then these vectors are grouped into 400 similar information-bearing clusters by the K-means clustering algorithm. These clusters from the training data were used to predict the information of background sentences in reviews from test data.

5.3.4 Features

5.3.4.1 Baseline

The most naïve assumption is to use the length of reviews as a feature. It is attested by many studies (Cao et al., 2011; Chen & Tseng, 2011; Kim et al., 2006; Korfiatis et al., 2012; Y. Liu et al., 2008; O'Mahony & Smyth, 2009; Zhang & Varadarajan, 2006) that the review length is the most intuitive indicator of review helpfulness. The number of words and sentences are encoded as features (**BASE**).

5.3.4.2 Bag of Words

Using the frequency of all word types is commonly referred as the bag-of-word feature (BOW). This feature is one of the most basic features for most natural

language processing tasks. Since this includes all word type frequencies, using this feature exhibits a strong performance in various tasks. At the same time, however, it also has so much noise; the same words can be used both in a helpful review and in an unhelpful review.

Kim et al. (2006) included unigram and bigram tf-idf-weighted frequency as features and through a review ranking experiment, they reported that the unigram frequency is one of the most effective features.

5.3.4.3 Features from Previous Studies

TF-IDF Frequency and Sentence types

These features come from the effective features reported in Kim et al. (2006). The tf-idf-weighted frequency of words for review documents are used to represent the meaning of sentences. In addition, the proportion of question or exclamation sentences were measured and used as features (**BOW_SENTTYPE**).

Subjectivity and Content

As mentioned in Chapter 2, among the effective features based on the information quality framework (Chen & Tseng, 2011), the Objectivity and the Appropriate Amount of Information dimensions are used and compared with this study (**SUBJ_CONTENT**). The objectivity dimension includes:

- the number of opinion sentences, positive sentences, negative sentences, and neutral sentences
- the percentage of opinion sentences, positive sentences, negative sentences, and neutral sentences

- the percentage of positive and negative sentences in all opinion sentences.

The appropriate amount of information dimension includes:

- the number of product features, opinion-bearing words, words, and sentences in a review
- the average frequency of product features in a review
- the number of sentences that mention product features in a review.

5.3.4.4 Features from Comparable Approaches of this Study

As previously explained, the approaches to extract background information vary from the simple pattern matching method to the topic-based approach. For the pattern matching method, f1 – f11 features in Table 11 extracted from background information are binary features and f12 and f13 are continuous features counting how many helpful and unhelpful binary features each review includes. The f1 – f11 features are categorized into helpful or unhelpful features depending on their values.

For instance, the experience on the use of previous versions (f1), competitor products (f2), or similar products (f3) are considered helpful if the values of the features are true, but unhelpful otherwise. In the same way, other features can be categorized into helpful or unhelpful features. These features from this pattern matching approach are referred as BK_REGEX in this study.

For the seed-based information extraction approach, since each seed sentence represents each type of background knowledge, when a review contains sentences that are considered to be similar to the seed sentences, the review is assumed to

have the background information. Thus, seed-based background information features are binary features corresponding to whether or not a review contains certain background information. This feature group is referred to as BK_SEED.

The feature group extracting background knowledge based on LDA has 400 feature dimensions, which is the number of clusters for k-means clustering. For each type of background information, the frequency of sentences that are predicted to be the same cluster is encoded as a feature. This feature group is referred to as BK_LDA.

5.4 Experiments and Analysis

5.4.1 Experiment Setting

For the review ranking experiments, the 200 e-book reader reviews for each product domain are randomly shuffled three times, but with a fixed sequence to ensure the same data set between experiments. Since this is a ranking experiment, the training and test data can be varied with each random shuffling. For each shuffling, the experiments are 5-fold cross-validated, thus there is a total of 15 trials.

5.4.2 Model

As explained earlier, the general regression model is restricted to only continuous independent variables, so this study used the support vector regression (SVR) model (Vapnik, 1995), which has the advantage of choosing independent

variable types. The support vector machine is commonly used for classification problems. It finds a hyperplane that maximizes the margins from the separating hyperplane to the closest positive and negative samples (support vectors) in the training data. The SVR model is a variant of the support vector machine for applying it to a regression problem. The SVR model uses a new type of loss function, referred as the ε-sensitive loss function. At the same time, it reduces the distance, the Euclidean norm $||w||^2$, from support vectors to the hyperplane as a svm model. In other words, the model does not care about errors as long as they are less than ε and minimize $||w||^2$. It is formally stated as follows.

$$f(x) = wx + b$$

$$\text{minimize } \frac{||w||^2}{2}$$

$$\text{subject to } \begin{cases} y_i - wx_i - b \leq \varepsilon \\ wx_i + b - y_i \leq \varepsilon \end{cases}$$

EQUATION 4. THE LINEAR FUNCTION AND CONDITIONS FOR E-SENSITIVE SVR

This model introduces slack variables (ξ_i, ξ_i^*), to measure the deviation of training samples outside the ε-insensitive zone. Thus Equation 4 arrives at the formulation stated in (Vapnik, 1995) as follows.

$$\text{minimize } \frac{1}{2}||w||^2 + C\sum_{i=1}^{\ell}(\xi_i + \xi_i^*)$$

$$\text{subject to } \begin{cases} y_i - wx_i - b \leq \varepsilon + \xi_i \\ wx_i + b - y_i \leq \varepsilon + \xi_i^* \\ \xi_i, \xi_i^* \geq 0 \end{cases}$$

EQUATION 5. OPTIMIZATION PROBLEM WITH SLACK VARIABLES (Ξ_i, Ξ_i^*)

This formulation can be graphically shown with the linear model in Figure 6 below.

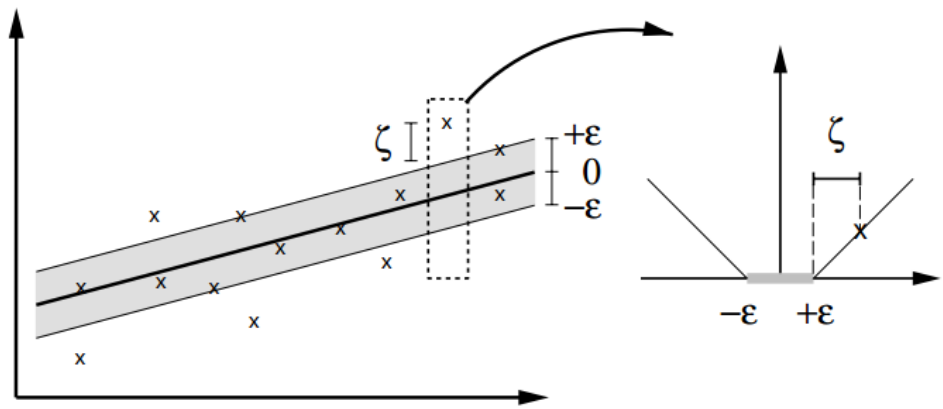

Figure 6. The soft margin loss setting corresponds to the linear SV machine (Smola & Schölkopf, 2004)

For the SVR model of this experiment, the ε and C parameters, for the loss sensitivity and the penalty constant for training data out of the insensitive band, are respectively set to 0.1 and 1. Additionally, the rbf kernel was chosen.

5.4.3 The Evaluation Metrics

The evaluation metrics used throughout all regression or order comparing tasks in this study are introduced below. This study adopts one of the most commonly used metrics, Kendall's tau, and proposes its own metrics designed for this specific task.

5.4.3.1 Kendall's tau

The pair-wise review order agreement was measured by Kendall's tau-b score (Agresti, 2010), which is a deviation of the tau coefficient, commonly used for comparing ranks with tied ranks. This score is also used for evaluating the review ranking result. The tau-b is calculated to show the degree of concordance of review ranks, ranging from -1 to 1. The formula for calculating tau-b is as below.

$$\boldsymbol{\tau_B = \frac{n_c - n_d}{\sqrt{(n_0 - n_1)(n_0 - n_2)}}}$$ [3]

EQUATION 6. THE KENDALL'S TAU-B

Where

$$n_0 = n(n-1)/2$$

$$n_1 = \sum_i t_i(t_i - 1)/2$$

$$n_2 = \sum_j u_j(u_j - 1)/2$$ [4]

5.4.3.2 Ranking Distance

This study proposes a new metric to measure the distance of ranking change, named ranking distance (rd). This measures the average distance between the original rank of a review and its estimated rank, normalized to range from 0 to 1

[3] n_c = Number of concordant pairs
n_d = Number of discordant pairs

[4] t_i = Number of tied values in the i^{th} group of ties for the first quantity
u_j = Number of tied values in the j^{th} group of ties for the second quantity

for all reviews. The motivation of proposing the ranking distance metric is that the tau-b metric can only measure whether the rank order between a pair of reviews is still the same with the estimated rank order, not the degree of the rank change. In the following equation, ranking distance is explained in more detail.

$$\boldsymbol{dist_{diff}} = |\boldsymbol{rank_g}(\boldsymbol{r}) - \boldsymbol{rank_e}(\boldsymbol{r})|$$
$$\boldsymbol{dist_{max}} = \boldsymbol{max}\begin{cases}\left|\boldsymbol{length}(\boldsymbol{R_e}) - \boldsymbol{rank_g}(\boldsymbol{r})\right| \\ \boldsymbol{rank_g}(\boldsymbol{r})\end{cases}$$
$$\mathbf{rd} = \sum_{\boldsymbol{r}\in\boldsymbol{R}} \frac{\boldsymbol{dist_{diff}}}{\boldsymbol{dist_{max}}} \times \frac{\mathbf{1}}{\boldsymbol{length}(\boldsymbol{R})}$$

EQUATION 7. THE EQUATION TO CALCULATE THE RANKING DISTANCE

In Equation 7, the distance ($dist_{diff}$) is the distance between the original rank and the estimated rank of the review. Here, a review (r) is a member of the original ranked review set (R): $r \in R_g$. Since reviews can be tie-ranked, R_g is a set of tie-ranked reviews. Additionally, R_e is the set of estimated reviews. The $rank_g()$ is the original rank function and $rank_e()$ the estimating rank function. The $dist_{diff}$ is normalized by the possible maximum distance ($dist_{max}$) that the rank of the review can be changed. This normalizing factor is the maximum value between the original rank and the distance from the estimated rank to the lowest rank of estimated ranks. Then the degree of ranking change for each review is calculated by dividing $dist_{diff}$ by $dist_{max}$. This value for each review is averaged for all reviews. This ranking distance (rd) can be measured by averaging the degree of the rank change for all reviews. The specific algorithm of the ranking distance is described with pseudo-code in Figure 7.

```
function ranking_distance (predicted_review_ranking)

    for each review R in predicted_review_ranking:
        compute the maximum distance (md) that each review(R) can move
        diff = the number of predicted ranks – the original rank of R
        if diff > the original rank of R then
            md = diff
        else
            md = the original rank of R
        the degree of ranking change for R =
            | the predicted rank of R – the original rank of R | / md
    Average the degree of ranking change for R
    Return Average the degree of ranking change for R
```

Figure 7. The Algorithm to Calculate the Ranking Distance

5.4.3.3 Top-n

Though the ranking distance can calculate the degree of ranking change, most consumers do not care about all the reviews, but only the top-n most helpful reviews. Thus, the estimated ranks of the originally high ranked reviews are more important than the others for better customer experience. To measure the top-n metric, the number of top-10 reviews remaining in the estimated top-10 ranks is calculated.

5.4.4 Results and Analysis

To see the separate effect of feature groups, we combine each feature group with baseline features (BASE), the length of reviews. As explained about the scores for helpfulness of reviews, the helpfulness vote score (h_v) and helpfulness manual score (hm) are two different helpfulness scores. The score that suit the purpose of this study is the helpfulness manual score (h_m). However, to see the different experiment results trained on the h_v or h_m score, the examination of separate feature groups was conducted with both scores.

Here, it is important to point out that from the experiment results, one should not conclude that one helpfulness score is superior to the others based on the performance scores for the experiments. Rather, it is proper to say that the two helpfulness scores are different and lead into different models. The question of which helpfulness score is better than the others can be answered by directly comparing the order of reviews by the scores. Chapter 4 discussed the different helpfulness scores in depth.

The results in Table 12 show the evaluation scores of ranking reviews trained on the helpfulness vote score (h_v). Comparing this result to the one with the helpfulness manual score (h_m) in Table 13, the effect of feature groups (BOW_SENTTYPE and SUBJ_CONTENT) from previous studies is opposite; BOW_SENTTYPE seems to be more effective than SUBJ_CONTENT in Table 12 in terms of tau-b score, but in Table 13 it is reversed. In addition, the best feature group in Table 12 is the BOW feature groups in tau-b, rd and top-10 scores. On the contrary, the best feature groups in the h_m score results is BK_LDA and SUBJ_CONTENT groups. These two groups show a little difference in tau-b and

rd score. For the top-10 score with h_m, BK_LDA is superior to SUBJ_CONTENT. Comparing the different effect of features depending on the helpfulness scores (h_v and h_m) shows the necessity of building manual helpfulness scores.

Table 12. Separate Feature Examination with helpfulness vote score (h_v)

	Tau-b	rd	Top-10
BASE	-0.016	0.441	0.240
BASE+BOW	0.168	0.383	0.393
BASE+BOW_SENTTYPE	0.122	0.406	0.380
BASE+SUBJ_CONTENT	0.000	0.444	0.193
BASE+INFO_TYPE	0.028	0.435	0.186
BASE+BK_REGEX	0.011	0.447	0.206
BASE+BK_SEED	0.022	0.446	0.206
BASE+BK_LDA	0.116	0.419	0.320

Table 13. Separate Feature Examination with helpfulness manual score (h_m)

	Tau-b	rd	Top-10
BASE	0.020	0.420	0.220
BASE+BOW	0.080	0.414	0.346
BASE+BOW_SENTTYPE	0.036	0.420	0.340
BASE+SUBJ_CONTENT	0.108	0.403	0.240
BASE+INFO_TYPE	0.064	0.425	0.233
BASE+BK_REGEX	0.062	0.415	0.220
BASE+BK_SEED	0.069	0.417	0.220
BASE+BK_LDA	0.100	0.408	0.313

Table 14. The best combination of feature groups with h_m

	Tau-b	rd	Top-10
BASE	0.026	0.420	0.226
BASE+INFO_TYPE+BK_LDA	0.190*[a]	0.388*	0.313*
BASE+INFO_TYPE+BK_LDA+BOW	0.221*	0.379*	0.346*
BASE+INFO_TYPE+BK_LDA+ TFIDF_SENT_TYPE	0.154*	0.395	0.373*
BASE+INFO_TYPE+BK_LDA+SUBJ_CONTENT	0.265*	0.367*	0.413*

[a] The star(*) indicates the average score is better than the baseline average score. The statistical significance of the results are tested by t-test ($p < 0.05$).

Based on the results of the separate feature examination, various feature combinations are examined. Though the separate feature groups related with background information does not seem to be significantly more effective than the baseline performance, combining it with other feature groups shows a significant effect on estimating the review helpfulness. Among the background information features, the BK_LDA was chosen and added to the INFO_TYPE features. Combining BASE, BK_LDA and INFO_TYPE indicates that the use of background information and the review information type can help to predict the review helpfulness. Adding BOW and SUBJ_CONTENT features, which are one of the basic factors and a sentiment related factor, improved the overall performance for this task even more.

5.5 Summary

This chapter introduced a preliminary experiment to examine the possibility of using separable information types in review sentences, especially the background information, for estimating the review helpfulness.

To extract background information from sentences, this study compared three different approaches: a simple regular pattern search approach based on pre-built regular expressions, a hybrid approach of pattern matching and grouping with a cosine similarity method based on seed sentences, each of which is a collection of sentences found by pattern matching, and lastly a topic-based approach that uses LDA sentence representations and finds clusters based on LDA representations, consequently finding similar information-bearing sentences to extract background information. Additionally, this study uses the review length features as a baseline and compares features from other studies.

Through review ranking experiments, it was found that though background information sentences take a small portion among all sentences composing a review, extracting specifically what background information a review bears can help to evaluate the review helpfulness. In addition, among the approaches for extracting background information, features based on the topic-based approach were the most effective in estimating review helpfulness. Furthermore, the counts of each review information type and features based on the Information Quality Framework (SUBJ_CONTENT) (Chen & Tseng, 2011), as well as background information features, appear to be effective. Combining the background features with these effective feature groups showed noticeable improvement in review ranking performance.

6 Recognition of Review Information Types

As the preliminary study indicates extracting specific background information from sentences can help to predict review helpfulness, it is reasonable to assume that other review information types are as effective as background information. In addition, it is expected that each extracted information type shows a different degree of effectiveness in estimating review helpfulness. To estimate the helpfulness of product reviews, this study proposes to divide the review sentences into different review information types and then extract specific information from each sentence within each information type group. Thus, the experiments are composed of two phases: recognizing review sentence information types and estimating the review helpfulness based on recognized review information types. This chapter explains the experiment of automatically identifying review sentence information types.

6.1 Task Description

Following the RIT (Review Information Type), sentences of reviews need to be categorized into their types. If there are overt clues that can differentiate one type from another, recognizing the information type of review sentences might be an easy task. However, due to the difficulties of categorizing the information types, as discussed in Chapter 5, there are obstacles to overcome when classifying the information types of sentences. We will introduce experiments with both

supervised and unsupervised models to recognize the information types and compare their results.

6.2 Models

Recognizing the information types of review sentences can be resolved by either supervised or unsupervised machine learning methods. The unsupervised models, such as clusters methods, have the advantage of not requiring any annotated data. For the case of recognizing the review information types of sentences, it means that, with clustering methods, annotating information types for review sentences is not necessary. In fact, depending on the product category, the sentences of each information type can significantly vary. For instance, the core type sentences in e-book readers are likely to include words such as touch screen, battery life, sensitivity, etc. On the other hand, core sentences in outdoor tent reviews would contain words such as rainfly, pole, window, etc. This indicates that to train a supervised classification model for review information types, it is required to annotate information types of enough sentences for every domain of every product that shares similar product aspects. Therefore, from the point of view of the cost of annotating data, clustering methods are more efficient.

However, clustering methods have a weakness in determining the members of each cluster. Though the number of clusters can be predefined to be the same as the number of information types, the members of the clusters are not decided depending on the information types. For the clustering approaches, k-means and DBSCAN were applied, as introduced in an earlier chapter. On the contrary,

supervised classification models can ensure the classification of each type of information with annotated training data.

6.2.1 Unsupervised Clustering Methods

The k-means and DBSCAN algorithms are also tested to categorize the information types. Since the number of information types is decided by the nature of the suggested information kinds, k-means clustering is chosen to partition n observations into k clusters. For k-means, the number of clusters was set to 4, the number of times the k-means will be run with different centroid seeds was 10, and the way of selecting initial clusters was conducted with the 'k-mean++' method to reduce the convergence time by initializing centroids to be distant from each other, implemented in scikit-learn (Pedregosa et al., 2011).

The parameters for DBSCAN, the eps and minPts, were decided to find the number of clusters that correspond to the number of information types and to guarantee as the smallest noise possible.

6.2.2 Supervised Learning Models

Support Vector Machine

One of the widely used classification models among supervised learning models is support vector model (Vapnik, 1995). This model constructs a hyperplane or set of hyperplanes that separate training-data points by keeping the margin between points and the planes as wide as possible. The larger the margin, the lower the

generalization error of the support vector model. With the decided hyperplane, other data points from test data can be classified to one or the other.

From Figure 8, w is normal to the hyperplane, |b| / ||w|| is the perpendicular distance from the hyperplane to the origin, and ||w|| is the Euclidean norm of w. The points that lie on the separating hyperplane satisfy $w \cdot x + b = 0$. When the margins from the separating hyperplane to the closest positive and negative samples are said to be d_+ and d_-, the support vector machine looks for the separating hyperplane with the largest d_+ and d_-.

When the data points cannot be linearly separable, the svm model maps the original finite-dimensional space into a much higher-dimensional space, making the separation easier in the higher-dimensional space. This is achieved by using various kernel functions, such as the linear function, the Radial Basis Function and the polynomial kernel.

The svm model is widely used due to the advantages in effectiveness in high dimensional spaces and in cases where the number of dimensions is greater than the number of samples. Also, the model is memory efficient because it only uses a subset of training points (support vectors) in the decision process.

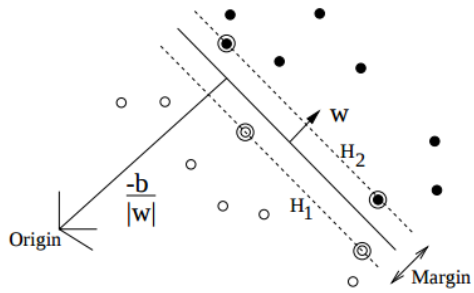

Figure 8. Linear separating hyperplanes for the separable case. The support vectors are circled. (Burges, 1998)

For the classification of information types, the number of classes is 4, hence multi-class classification models should be used. The multi-class classification models are divided into “one-against-one” (Knerr, Personnaz, & Dreyfus, 1990) and “one-vs-the-rest” approaches. The number of classification models differs depending on the approach. For the “one-against-one” approach, if n is the number of classes, n * (n-1) / 2 is the number of models to be constructed. For the “one-vs-the-rest” approach, the number of models corresponds to the number of classes, in turn significantly reducing the runtime. For this reason, in this study, “one-vs-the-rest” is used, implemented in the scikit-learn machine learning library (Pedregosa et al., 2011).

Conditional Random Fields

Recognizing the information types of review sentences is involved with finding the semantic meaning of sentences and categorizing it. However, it is often difficult to decide the information type of a sentence without any contextual information,

lacking enough lexical information to correctly infer the meaning of the sentence. Thus, the conditional random fields (CRFs) classification model (Lafferty et al., 2001) which considers the contextual information to predict the class label is examined.

The CRFs is a popular probabilistic method for structured prediction, or a sequence prediction, different from models predicting the label for one sample. It is a combination of undirected graphical modeling which enables the compact modelling of many interdependent input variables, and a discriminative classification method to make predictions using large sets of input features.

Discriminative probabilistic models are models that describe directly how to take a feature vector **x** and assign it a label **y**, contrary to the generative models which describe how a label vector **y** can probabilistically generate a feature vector **x**. The way to predict a single discrete class variable **y** given a vector of features **x** is to assume that all the features are independent, as the naive Bayes classifier, one of the generative models. The generative model is a family of joint distributions, $p(\mathbf{y},\mathbf{x})$. On the contrary, a discriminative model, such as CRFs, is a family of conditional distributions: $p(\mathbf{y}|\mathbf{x})$. The discriminative model makes conditional independence assumptions among **y**, but does not make conditional independence assumptions among **x**, which consequently allows interdependency among input features.

So far, the unsupervised and supervised models to recognize the information types of sentences have been introduced. From the next subsection, the features used for each models will be introduced.

6.3 Features for Recognizing Information Types

This subsection introduces the features used to recognize the difference in information types. The difficulty in extracting features for this task arises from the word overlapping problem between information types. As mentioned earlier, for instance, the words from the core information type can also occur in overall information sentences. It is expected that using the bag-of-words approach does not guarantee differentiation of the information types.

For the basic features, the tf-idf-weighted word frequencies (TFIDF) are used as features. Despite the word overlapping problem, these features are the first to be tested. For this feature dimension, not only the unigram frequencies of words are included as features, but bigram frequencies were also calculated. Additionally, tf-idf-weighted frequencies can be obtained based on the words and their part-of-speech tags (TFIDF_POS). The parts-of-speech of each token are analyzed by the Stanford part-of-speech tagger (Vapnik, 1995).

The LDA-based representation of sentences is also tested to recognize the information types. The LDA model is trained with the assumption of treating each sentence as a separate document with 300 predefined topics.

Additionally, it was observed from the data that there is a noticeable tendency to choose the subject, auxiliary verbs, and main verbs depending on the information type. For background information, people are likely to use the present perfect tense and personal pronouns, as seen in examples (10) to (13) in Chapter 3. Compared to background information, core information sentences are prone to begin with the product aspect-related words as subjects and use the present tense. Thus, the

syntactic dependency structure of sentences was analyzed with the Stanford coreNLP toolkit (Smola & Schölkopf, 2004). Additionally, the form and part-of-speech (pos) tag for subject, main verb and auxiliary verb are extracted as features (GRAMMAR).

Lastly, there was a difference in sentence positions depending on their information types. Background sentences are usually located at the beginning of reviews. Core information sentences are placed in the middle. Meanwhile, overall information sentences are written either at the beginning or the end of reviews. The position of the sentence is normalized by the length of the review to be used as a feature (POSITION).

6.4 Recognition of Review Information Types

6.4.1 The Results with Clustering Models

The reason for applying unsupervised clustering models to recognize the review information types of sentences is that building the annotated corpus for review information type is expensive. For this study, a total of 600 reviews were annotated to be used as the gold standard for the information type recognition task.

Firstly, the DBSCAN algorithm was applied to recognize the information types. The eps and the minPts were adjusted to make the number of clusters the same as the number of information types. The distance between points was calculated based on the cosine similarity with the brute-force algorithm to find the nearest neighbors, which is a way of calculating the similarity between vectors. The TFIDF features

for the clustering algorithm included only weighted counts for unigrams. The GRAMMAR features were not used to make sentence clusters due to the inappropriateness of the categorical features for clustering algorithms. In Table 15, the number of clusters were set to 4, but the number of noise sentences is so many that this algorithm cannot label the majority of sentences properly. The amount of noise t can be reduced by adjusting the value of the eps. However, this results in the reduction of the number of cluster, and thus is not able to guarantee the number of clusters is 4. Thus, sentences in other product domains were not further examined.

Table 15. The Clustering Results of Review Sentences for an E-reader with DBSCAN

	1	**2**	**3**	**4**	**noise**
TFIDF[a]	4078	8	13	5	1736
LDA[b]	1380	24	31	24	4381

[a] eps: 0.6, minPts: 20

[b] eps: 0.45, minPts: 9

Secondly, k-means was examined to cluster review sentences. Clustering methods only categorize the sentences into different clusters and do not tell what the property or criteria for making clusters is. From the results in Table 16 and Table 17, it can be noticed that the clusters do not correspond to the information types; rather, the different information type sentences are divided into four clusters. For every cluster, the core sentences take the highest proportion among all information types, while the proportion of other information types varies depending on the cluster. This experiment indicates that clustering methods cannot

distinguish the information types of sentences based on the word frequencies or topic distributions. Though the clustering algorithm does not cluster depending on the information types, it is worth examining how the results of k-means clustering for information type recognition affects the overall performance in the review helpfulness evaluation task.

Table 16. The Clustering Results for sentences with k-means (e-book reader reviews)

	TFIDF				LDA			
Clusters	0	1	2	3	0	1	2	3
Background	52	342	33	552	802	91	65	21
Core	407	1876	237	1143	2833	319	402	109
Peripheral	39	450	37	262	632	56	81	19
Overall	35	544	35	445	848	48	87	76

Table 17. The Clustering Results for sentences with k-means (tent reviews)

	TFIDF				LDA			
Clusters	0	1	2	3	0	1	2	3
Background	15	206	16	243	16	188	31	245
Core	357	2137	574	925	561	2173	331	928
Peripheral	11	131	17	90	20	105	16	108
Overall	36	276	20	186	13	286	25	194

6.4.2 The Results with SVM model

The recognition of review information types is expected to show improved performance with supervised classifiers. The results in Table 18 show that among the three different sentence representations (TFIDF, TFIDF_POS, LDA), the TFIDF_POS feature group appears to have the best performance considering the f1 scores in every information type. Surprisingly, the LDA shows considerably

inferior results to recognize the sentence information types compared to TFIDF and TFIDF_POS features. The difference in the number of sentences among information types is remarkable due to the nature of the information type in that the classification result is affected by this imbalance of information type. As earlier noted, the core information shows the most accurate precision and recall results. On the other hand, the results with the background, peripheral and overall information sentences need to be improved. When considering the number of classes, randomly guessing sentence information types should result in 25% accuracy for each class. Fortunately, the result with TFIDF_POS features is far above this baseline.

Adding the sentence position in a review as a feature (POSITION) to TFIDF_POS helps to increase the precision and recall for background and overall information. This indicates that the sentence position is an important factor to recognize the information type. In most cases, the background information appears at the beginning of product reviews and the overall information appears at the beginning or the end of reviews.

Furthermore, adding the form and part-of-speech tag for main subject, verb, and auxiliary verb (GRAMMAR) to TFIDF_POS features led to a considerable improvement in background information and a small increase of recall in peripheral and overall information. This indicates that the subjects and verbs in background information have a certain tendency compared to other information types, as explained earlier. The addition of POSITION and GRAMMAR to TFIDF_POS features set raises the performance in recall and precision for background information type. Thus, the key to train a model for recognizing

information types should be focused on the performance enhancement of background, peripheral and overall information types. This result has to be compared to the results in other product domains.

Table 18. The Result of Recognizing Information Types with SVM (e-book reader)

Feature	B (979) [a]	C (3666)	P (767)	O (1060)	Total
TFIDF	0.50 / 0.34 / 0.40[b]	0.67 / 0.87 / 0.76	0.57 / 0.23 / 0.32	0.52 / 0.38 / 0.43	0.61 / 0.63 / 0.60
TFIDF_POS	0.48 / 0.38 / 0.43	0.71 / 0.84 / 0.77	0.56 / 0.35 / 0.42	0.51 / 0.43 / 0.46	0.62 / 0.64 / 0.62
LDA	0.47 / 0.20 / 0.27	0.63 / 0.92 / 0.74	0.43 / 0.16 / 0.23	0.47 / 0.21 / 0.28	0.56 / 0.60 / 0.54
TFIDF_POS+ POSITION	0.52 / 0.42 / 0.46	0.72 / 0.84 / 0.77	0.57 / 0.36 / 0.43	0.54 / 0.48 / 0.51	0.64 / 0.66 / 0.64
TFIDF_POS+ GRAMMAR	0.51 / 0.44 / 0.48	0.71 / 0.83 / 0.76	0.56 / 0.37 / 0.44	0.51 / 0.44 / 0.47	0.63 / 0.65 / 0.63
TFIDF_POS+ POSITION+ GRAMMAR	0.56 / 0.47 / 0.51	0.72 / 0.83 / 0.77	0.56 / 0.37 / 0.43	0.53 / 0.48 / 0.50	0.65 / 0.66 / 0.65

[a] B, C, P and O respectively refers to Background, Core, Peripheral and Overall information. The number in parentheses is the total number of sentences for each category.

[b] The numbers are enumerated in the order of precision / recall / f1 score.

The experiment with tent reviews in Table 19 shows a slightly different result from the one with e-book reader reviews above. The TFIDF_POS features outperform TFIDF and LDA features, and the addition of the sentence position features (POSITION) or the GRAMMAR feature set to TFIDF_POS shows an improved result compared to only the TFIDF_POS feature set. However, adding POSITION and GRAMMAR together to TFIDF_POS does not enhance the result, compared to the result of TFIDF_POS + POSITION.

Additionally, the core information sentences are more accurately recognized, but there is a decrease in recall for peripheral and overall information with tent reviews

when compared with the results for e-book reader reviews. Especially, the low recall for peripheral information recognition was due to the imbalance of sentence samples in training and testing, since the number of core sentences is more than 15 times higher than that of peripheral sentences.

Table 19. The Result of Recognizing Information Types with SVM (Tent)

Feature	B (480)[a]	C (3999)	P (249)	O (518)	Total
TFIDF	0.65 / 0.28 / 0.39 [b]	0.82 / 0.97 / 0.88	0.46 / 0.06 / 0.10	0.58 / 0.30 / 0.39	0.76 / 0.80 / 0.75
TFIDF_POS	0.57 / 0.35 / 0.43	0.83 / 0.95 / 0.89	0.49 / 0.17 / 0.24	0.55 / 0.32 / 0.40	0.76 / 0.80 / 0.77
LDA	0.55 / 0.20 / 0.29	0.80 / 0.98 / 0.88	0.19 / 0.02 / 0.04	0.59 / 0.15 / 0.24	0.72 / 0.78 / 0.72
TFIDF_POS+ POSITION	0.64 / 0.42 / 0.50	0.84 / 0.95 / 0.89	0.55 / 0.19 / 0.26	0.58 / 0.36 / 0.44	0.78 / 0.81 / 0.78
TFIDF_POS+ GRAMMAR	0.59 / 0.36 / 0.45	0.83 / 0.95 / 0.89	0.53 / 0.19 / 0.26	0.55 / 0.34 / 0.41	0.77 / 0.80 / 0.77
TFIDF_POS+ POSITION+ GRAMMAR	0.60 / 0.37 / 0.45	0.84 / 0.95 / 0.89	0.52 / 0.19 / 0.26	0.55 / 0.34 / 0.41	0.77 / 0.80 / 0.77

[a] B, C, P and O respectively refer to Background, Core, Peripheral and Overall information.

The number in parentheses is the total number of sentences for each category.

[b] The numbers are enumerated in the order of precision / recall / f1 score.

Lastly, the same experiment was conducted with jeans reviews. The TFIDF_POS feature set was the best sentence representation, corresponding to the results with other products. For peripheral information sentences, it shows the lowest precision and recall. Additionally, other information types are not recognized as accurately as the other two product reviews. It is perhaps because the number of sentences for each information type, used to train the SVM model, is generally lower than that of other reviews. It can be seen from the results of the three different products that the

precision and recall are highly related with the number of sentences for the information type.

Table 20. The Result of Recognizing Information Types with SVM (Jeans)

Feature	B (351)[a]	C (1028)	P (132)	O (380)	Total
TFIDF	0.47 / 0.23 / 0.30	0.61 / 0.85 / 0.71	0.16 / 0.04 / 0.06	0.38 / 0.29 / 0.32	0.51 / 0.57 / 0.51
TFIDF_POS	0.48 / 0.34 / 0.38	0.65 / 0.80 / 0.71	0.21 / 0.09 / 0.12	0.42 / 0.40 / 0.41	0.54 / 0.58 / 0.55
LDA	0.51 / 0.29 / 0.36	0.60 / 0.87 / 0.70	0.03 / 0.01 / 0.01	0.44 / 0.23 / 0.29	0.51 / 0.57 / 0.51
TFIDF_POS+ POSITION	0.50 / 0.38 / 0.42	0.66 / 0.79 / 0.72	0.29 / 0.08 / 0.11	0.46 / 0.46 / 0.46	0.57 / 0.60 / 0.57
TFIDF_POS+ GRAMMAR	0.53 / 0.39 / 0.43	0.66 / 0.80 / 0.72	0.35 / 0.11 / 0.15	0.44 / 0.42 / 0.42	0.57 / 0.60 / 0.57
TFIDF_POS+ POSITION+ GRAMMAR	0.52 / 0.39 / 0.43	0.66 / 0.79 / 0.72	0.36 / 0.10 / 0.15	0.44 / 0.44 / 0.43	0.57 / 0.60 / 0.57

[a] B, C, P and O respectively refers to Background, Core, Peripheral and Overall information. The number in parentheses is the total number of sentences for each category.

[b] The numbers are enumerated in the order of precision / recall / f1 score.

Overall, the results for recognizing information types for the three different products show that the combination of TFIDF_POS features with the sentence position feature (POSITION) and the subject, verb, and auxiliary features (GRAMMAR) guarantees the strong performance in recall and precision, though the GRAMMAR feature for outdoor tent products seems to not cooperate with the POSITION feature.

The difficulty in recognizing background, peripheral and overall information is due to a comparably lower data size than core information sentences. It is expected that by increasing the number of sentences for each information type, the accuracy would increase as high as core type information.

From the experiment results, it is also difficult to recognize the sentence information type, since sentences often do not contain enough information in themselves, rather they rely on contextual information to be accurately interpreted. Thus, another learning model which uses contextual information was examined.

6.4.3 The Results with CRF model

Previously, it was pointed out that the conditional random field (CRF) model might be necessary to recognize the sentence information type which depends on the contextual information. Different from the experiment with the SVM model, the TFIDF feature is not included in this experiment. In addition, the feature dimensions in Table 21 seem to be the same as the feature dimensions in Table 18, Table 19 and Table 20, but each dimension also includes the same features from the i-1 and i-2 previous sentences in order to recognize the sentence information type in terms of previous information.

The first noticeable change in Table 21 from the results of the svm model experiments is that the recall of background and peripheral information is remarkably increased and the precision of core and peripheral information also improved. This result indicates that the information type of the current sentence is partially dependent on the previous sentences, and possibly the information types of previous sentences. The feature combination of TFIDF_POS with POSITION and GRAMMAR features also help to increase the overall performance. The POSITION and GRAMMAR features also include the same features from previous sentences.

Table 21. The Result of Recognizing Information Types with CRF (e-book Reader)

Feature	B (979) [a]	C (3666)	P (767)	O (1060)	Total
TFIDF_POS	0.50 / 0.41 / 0.44	0.73 / 0.85 / 0.78	0.61 / 0.48 / 0.54	0.52 / 0.37 / 0.43	0.64 / 0.66 / 0.64
LDA	0.49 / 0.22 / 0.30	0.66 / 0.89 / 0.76	0.51 / 0.43 / 0.44	0.58 / 0.24 / 0.33	0.60 / 0.63 / 0.58
TFIDF_POS+ POSITION	0.53 / 0.44 / 0.48	0.73 / 0.86 / 0.79	0.63 / 0.51 / 0.57	0.57 / 0.40 / 0.46	0.66 / 0.68 / 0.66
TFIDF_POS+ GRAMMAR	0.54 / 0.44 / 0.48	0.73 / 0.86 / 0.79	0.64 / 0.52 / 0.57	0.54 / 0.40 / 0.45	0.66 / 0.68 / 0.66
TFIDF_POS+ POSITION+ GRAMMAR	0.56 / 0.46 / 0.50	0.74 / 0.86 / 0.80	0.64 / 0.52 / 0.57	0.58 / 0.44 / 0.50	0.68 / 0.69 / 0.68

[a] B, C, P and O respectively refers to Background, Core, Peripheral and Overall information.

[b] The numbers are enumerated in the order of precision / recall and f1 score.

The results with the CRF model for tent reviews is different from the results of the SVM model in that there was a noticeable improvement in recognizing the background type and overall information sentences. Unfortunately, the precision and recall for peripheral information sentences are decreased. This is possibly due to the data size for training. The number of peripheral sentences is considerably lower than other information type sentences for tent reviews. The addition of POSITION and GRAMMAR features to TFIDF_POS reveals the best performance for all information types.

Table 22. The Result of Recognizing Information Types with CRF (Tent)

Feature	B[a]	C	P	O	Total
TFIDF_POS	0.66 / 0.52 / 0.57[b]	0.85 / 0.95 / 0.90	0.30 / 0.13 / 0.16	0.65 / 0.33 / 0.43	0.79 / 0.81 / 0.79
LDA	0.74 / 0.43 / 0.53	0.82 / 0.97 / 0.89	0.02 / 0.02 / 0.02	0.80 / 0.24 / 0.36	0.77 / 0.81 / 0.76
TFIDF_POS+ POSITION	0.69 / 0.55 / 0.60	0.86 / 0.95 / 0.90	0.33 / 0.12 / 0.15	0.66 / 0.43 / 0.51	0.80 / 0.83 / 0.80
TFIDF_POS+ GRAMMAR	0.69 / 0.54 / 0.59	0.86 / 0.96 / 0.90	0.37 / 0.18 / 0.21	0.67 / 0.38 / 0.48	0.80 / 0.82 / 0.80
TFIDF_POS+ POSITION+ GRAMMAR	0.74 / 0.56 / 0.62	0.87 / 0.96 / 0.91	0.38 / 0.16 / 0.20	0.66 / 0.44 / 0.52	0.81 / 0.83 / 0.81

[a] B, C, P and O respectively refers to Background, Core, Peripheral and Overall information.

[b] The numbers are enumerated in the order of precision / recall / f1 score.

Lastly, the CRF model is applied to jeans reviews. For this product domain, the CRF model outperforms the SVM model in every information type. In addition, the POSITION and GRAMMAR features help to increase the overall performance for all information types.

Table 23. The Result of Recognizing Information Types with CRF (Jeans)

Feature	B[a]	C	P	O	Total
TFIDF_POS	0.50 / 0.38 / 0.42[b]	0.65 / 0.84 / 0.73	0.31 / 0.17 / 0.20	0.47 / 0.29 / 0.35	0.56 / 0.60 / 0.56
LDA	0.69 / 0.28 / 0.38	0.60 / 0.93 / 0.73	0.07 / 0.05 / 0.06	0.64 / 0.16 / 0.25	0.59 / 0.59 / 0.52
TFIDF_POS+ POSITION	0.52 / 0.41 / 0.45	0.67 / 0.83 / 0.74	0.34 / 0.17 / 0.21	0.52 / 0.37 / 0.43	0.59 / 0.61 / 0.58
TFIDF_POS+ GRAMMAR	0.52 / 0.40 / 0.43	0.67 / 0.83 / 0.73	0.32 / 0.18 / 0.21	0.51 / 0.36 / 0.41	0.58 / 0.61 / 0.58
TFIDF_POS+ POSITION+ GRAMMAR	0.56 / 0.42 / 0.46	0.68 / 0.82 / 0.74	0.39 / 0.18 / 0.21	0.54 / 0.44 / 0.47	0.61 / 0.63 / 0.60

[a] B, C, P and O respectively refers to Background, Core, Peripheral and Overall information.

[b] The numbers are enumerated in the order of precision / recall / f1 score.

From these experiments with the CRF model, it is expected that if the information type of the previous sentence is known, recognition of the current sentence's information type could have been easier. Thus, another experiment was conducted with the same environment but with the information types of the previous sentences given. The results in Table 24 proves that knowing the information types of previous sentences remarkably improves the performance when recognizing the information type of the current sentence. Furthermore, it indicates that with the CRF model and features to expect the information types of the previous sentences, the model can be improved to be practically usable for the review ranking experiment.

Table 24. The Result of Recognizing Information Types with CRF (given previous information types)

Feature	B[a]	C	P	O	Total
e-book reader	0.70 / 0.62 / 0.66[b]	0.82 / 0.90 / 0.86	0.78 / 0.70 / 0.73	0.69 / 0.56 / 0.62	0.78 / 0.78 / 0.78
tent	0.78 / 0.72 / 0.74	0.92 / 0.97 / 0.94	0.71 / 0.48 / 0.56	0.79 / 0.62 / 0.69	0.88 / 0.89 / 0.88
jeans	0.64 / 0.56 / 0.59	0.75 / 0.88 / 0.81	0.54 / 0.43 / 0.45	0.66 / 0.50 / 0.56	0.70 / 0.71 / 0.69

[a] B, C, P and O respectively refers to Background, Core, Peripheral and Overall information.

[b] The numbers are enumerated in the order of precision / recall / f1 score.

6.5 The Summary of Recognizing Information Types

Overall, this chapter introduced the review information types which divide the sentences depending on the target of the information. To recognize the information

type for each sentence, supervised and unsupervised learning algorithms were applied.

The unsupervised algorithms DBSCAN and k-means were examined, but these approaches cannot correctly categorize the information types.

For supervised approaches, SVM and CRF models were applied. From the experiment with the SVM model, effective feature combinations could be discovered, but the accuracy of recognizing information types was not sufficient enough to be used. The effective feature combination was the tf-idf weighted unigram and bigram frequencies with part-of-speech tagged words (TFIDF_POS), the sentence position (POSITION), and the form and part-of-speech tag of the main subject, verb and auxiliary (GRAMMAR).

With the assumption that the recognition of information types could depend more on the information from previous sentences, the CRF model was examined and showed improved accuracy in general. However, the f1 score with the CRF model for background, peripheral and overall information types seems to be not sufficient for the next phase's input. These results will be used as information types of review sentences for the review ranking experiment.

7 Estimation of Review Helpfulness

7.1 Task Description and Restriction

The task at hand is to rank the reviews according to some automatically estimated score or criteria in terms of their helpfulness. This can be done by two different approaches: by performing a pair-wise comparison of two reviews or by ordering all reviews by a score. The former transforms the matter of ranking to a classification problem. It is maybe more accurate to decide which review is more helpful than the other. However, it is practically impossible to calculate the pair-wise classification whenever the new review is written. Additionally, building the gold standard for this classification problem requires too much time and human resources due to the huge amount of reviews. Therefore, ranking should be trained and tested using regression models. With the usual regression model, the independent variables are restricted to continuous models, which in turn restricts the types of features used to train the model. For this reason, Support Vector Regression (SVR) model (Basak, Pal, & Patranabis, 2007) was chosen for training and testing to use various variable types of features from sentences. All the clustering and regression models, Scikit-learn, were also used (Pedregosa et al., 2011). Since the SVR model is introduced in depth in Chapter 5, a further explanation about the model is omitted here.

7.2 Data Collection

The data for this study are the reviews of all three product domains, earlier explained in Chapter 4. The data will simply be re-introduced in this chapter. The summary of collected reviews is repeated here in Table 25 below.

Table 25. Collected Data

Product Domain	e-book Reader	Outdoor Tent	Jeans
Total Number of Reviews	36140	7834	9175
Total Number of Products	2	10	10
Number of Annotated Reviews	200	200	200
Number of Products in 200 Annotated Reviews	1	8	5

7.3 Features for Estimating the Review Helpfulness

7.3.1 Baseline (BASE)[5]

The most naïve assumption is to use the length of reviews as a feature. It is attested by many studies (Cao et al., 2011; Chen & Tseng, 2011; Kim et al., 2006; Korfiatis et al., 2012; Y. Liu et al., 2008; O'Mahony & Smyth, 2009; Zhang & Varadarajan, 2006) that the review length is the most intuitive and strongest indicator of review helpfulness. The number of words and sentences are encoded as features.

[5] The "(CAPITAL LETTERS)" notation is used to refer to feature groups in the experiment results.

7.3.2 Features from Previous Studies

7.3.2.1 TF-IDF Frequency and Sentence types

These features come from the effective features reported in Kim et al. (2006). The tf-idf-weighted frequency of words (**BOW**) for review documents is used to represent the meaning of sentences, which is one of the basic features for natural language processing. In addition, the proportion of question or exclamation sentences were measured and used as features, which is a sentiment factor for estimating review helpfulness (**SENT_TYPE**).

7.3.2.2 Sentiment

Among the factors that are related with how information is delivered, the sentiment of product reviews was of the most widely used features in previous studies.

As mentioned in Chapter 2, among the effective features based on the information quality framework (Chen & Tseng, 2011), the sentiment-related factors, referred to as Objectivity in the study, are implemented and compared with this study.

The objectivity dimension includes:

- the number of opinion sentences, positive sentences, negative sentences, and neutral sentences
- the percentage of opinion sentences, positive sentences, negative sentences, and neutral sentences

- the percentage of positive and negative sentences in all opinion sentences.

Whether or not a sentence is an opinion sentence or a positive or negative polarity sentence follows the approach in Hu and Liu (2004) that Chen and Tseng (2011) used to extract the number of opinion sentences, positive sentences and negative sentences. They defined the opinion sentence as “a sentence [that] contains one or more product features and one or more opinion words.” We used the opinion words used in Hu and Liu (2004) and calculated the polarity of sentences as in Figure 9.

```
Procedure SentenceOrientation( )
begin
   for each opinion sentence si
   begin
      orientation = 0;
      for each opinion word op in si
         orientation += wordOrientation(op, si );
         /*Positive = 1, Negative = -1, Neutral = 0*/
      if (orientation > 0) si's orientation = Positive;
      else if (orientation < 0) si's orientation = Negative;
      else {
         for each feature f in si
            orientation +=
            wordOrientation(f's effective opinion, si );
         if (orientation > 0)
            si's orientation = Positive;
         else if (orientation < 0)
            si's orientation = Negative;
         else
            si's orientation = s i-1 's orientation;
      }
   endfor
end

Procedure wordOrientation(word, sentence)
begin
      orientation = orientation of word in seed_list;
      If (NEGATION_WORD appears closely around word in sentence)
         orientation = Opposite(orientation);
end
```

Figure 9. Predicting the orientation of opinion sentences (Hu & Liu, 2004)

7.3.2.3 Readability

As mentioned in the background studies, Ghose and Ipeirotis (2007, 2011) measured how easy or difficult a text is to read with various readability metrics from DuBay (2004) in order to estimate the helpfulness of reviews. Other studies (Cao et al., 2011; Chen & Tseng, 2011) measured the readability factor in different

ways, but attempt to use a similar aspect of reviews, the number of letters, words and sentences. For the readability factors, the traditional measures described in DuBay (2004) are Automated Readability Index (ARI), Coleman-Liau Index (CLI), Flesch Reading Ease (FRES), Flesch-Kincaid Grade Level (F-K), Gunning fog index, and SMOG, as in Equation 8. These factors are extracted as features to estimate the review helpfulness.[6]

ARI = 0.50 (words per sentence) + 4.71 (strokes per word) – 21.43

CLI = 0.0588L - 0.296S - 15.8

Where

L is the average number of letters per 100 words

S is the average number of sentences per 100 words.

$$\text{FRES} = 206.835 - 1.015\left(\frac{total\ words}{total\ sentences}\right) - 84.6\left(\frac{total\ syllables}{total\ words}\right)$$

$$\text{F-K} = 0.39\left(\frac{total\ words}{total\ sentences}\right) - 1.015\left(\frac{total\ words}{total\ sentences}\right) - 84.6\left(\frac{total\ syllables}{total\ words}\right)$$

$$\text{Gunning-Fog} = 0.4\left(\frac{total\ words}{total\ sentences} + 100\ \frac{total\ syllables}{total\ words}\right)$$

$$\text{SMOG} = 1.0430\sqrt{total\ polysyllables \times \frac{30}{number\ of\ sentences}} + 3.1291$$

EQUATION 8. FORMULAS OF READABILITY MEASURES

[6] https://pypi.python.org/pypi/textstat/.

7.3.3 Product Aspect Keyword-based Features (ASPECT)

The aspect-related keywords can be used to find the topic of each sentence. Despite the inefficiency of building product aspect keywords, the effect of this approach on extracting information from sentences needs to be tested and compared with other approaches.

The keywords are not just built as a series of possible keywords in a product domain. This study built groups of keywords, of which each group corresponds to one aspect of the product. For the e-book reader data, a total of 53 groups of 265 keywords are manually extracted from the reviews. See Table 26 for examples of keyword groups. The complete list of product features for e-book readers can be found in Appendix III.

Table 26. Example of Product Aspect Keywords

Keyword Group	Keywords
Backlight	back-light, backlight, lighting, bright, brightness, backlit, back-lit, background light, no glare, no glaring, too dim, dim, the light, light setting, lighting setting, light set, white screen, read at night, built-in screen light, light adjustment, unevenness, lighting technology, whiter, the new screen, the screen quality, the day light, uneven, shadow, glare, light condition, read outside, the sun, led light
Weight	Weight, heavy, heavier, lighter, lightweight, light weight, is light, lightness, a feather
Screen Defect	dead pixel, speck, spot, dust, blotch

With the manually built keywords, it is possible to recognize what aspect the sentence is about, though there are still difficulties in using the keywords. The simplest way of extracting information from sentences is to assume that each sentence contains a description of only one aspect. With this assumption, we can count the frequency of keywords from each sentence and decide the topic (aspect) of the sentence based on the most frequent keyword group. Then, the number of words used to describe each aspect is used as a feature to evaluate the helpfulness of a product review.

When it is assumed that each sentence contains more than one aspect, the measurement of how long each feature is dealt with becomes more complicated. In fact, recognizing the exact range for each product aspect in a sentence is difficult. Therefore, we approximately calculated the range of mentioned aspects in sentences in the following step.

(25) Steps to Calculate the Range of Aspects in a Sentence

i) Find the aspect groups in sentences by pattern matching using the keywords.

ii) Ignore the matched aspect, unless there is one of the conjunctions: *and*, *or*, *as well as*, *but*, and a comma (,).

iii) The ranges of the aspects are calculated by dividing the number of words by the number of recognized aspect groups.

Through the steps above, the range or length of product aspects can be encoded as features. These features correspond to how much each product aspect is dealt with in the review. This can be an alternative way of extracting what core

information a sentence hold. These product aspect keyword-based approaches will be examined for only the core information of an e-book reader and compared with other features.

7.3.4 The Proportion of Information Types (INFO_TYPE)

This feature group is one of the most crucial feature sets that can be expected to have a distinctive effect on the review helpfulness. This includes a count of sentences for each information type and the proportion of each information type in all sentences of a review, which is a normalized frequency for each information type.

7.3.5 The Semantics of Sentence Information

The semantic meaning of sentence information is extracted by clustering sentences holding similar information within each information type. Here, the meaning of sentences is represented in various ways. The feature sets can be made by combining three conditions: the way of representing sentences, information types and the extension of the sentence ranges.

7.3.5.1 Sentence Representation

For this feature dimension, we choose how each sentence is represented, tf-idf weighted bag-of-words vector (**TFIDF**) or LDA space conversion (**LDA**). The inverse document frequency and the LDA model were calculated with not just training data, but all the reviews collected for each product domain in Table 1.

Utilizing a large enough amount data enables us to obtain a more accurate topic model and weighting. Each sentence is represented by either a tf-idf weighted word frequency or LDA topic vector. Then, within each information type, sentences in training data reviews are clustered by the k-means or DB-scan algorithm. Clustering models are generated for each information type: core, background, peripheral, and overall. In the testing phase, each sentence in the test reviews is clustered, given its information type, with its corresponding clustering model. The frequencies of sentences for each cluster for every information type are used as input features. For instance, if there are set be 100 clusters for each information type, the dimension of all the features for the sentence information representation is 400. The choice of sentence representation is always combined with the way of extending the range of sentences.

7.3.5.2 Extracted Information Type

Each sentence can be converted into the word frequency or topic dimension, as noted above, and the information of the sentence can be divided depending on the Review Information Types: background (**BK**), core (**CORE**), peripheral (**PERI**), overall (**OVERALL**) or all types (**ALL_IF**). Sentences for each information type can be independently used to see the effect of each information type. The *none*-type sentences are not included for the experiments because the total number of sentences that are non-relevant to the product is so low that it does not have an effect on the results: 54 out of 3995 sentences of 200 reviews.

7.3.5.3 The Extension of the Sentence range

As previously mentioned in Chapter 3, a sentence often contains too small a number of words or is only composed of pronouns and non-content words. In these cases, the meaning or information of the sentence cannot be extracted by itself. Cai and Li (2011) experimented with various ways of finding similar sentences and reported that including context sentences in the representation of the current sentence helps to make better sentence cluster quality. Thus, the range of a sentence can be extended. Basically, using only the words from the current sentence is the most naïve range assumption (**CURRENT**). Then each sentence can includes the previous and next sentence to represent the meaning of the sentence (**SURR**). The first and the last sentence of a review contain only the next and previous sentence, respectively. Since the meaning of sentence can rely on the only the previous sentences and not the next ones, it can include two previous sentences (**PREV_PREV**).

From the three conditions of composing sentence meaning, the feature set can be made with the format of Information_Representation_Range, such as **BK_LDA_CURRENT, CORE_TFIDF_PREV_PREV**, etc. These feature groups, introduced so far, will be tested through the following experiments.

7.4 Experimental Setting

The experiments to rank reviews in terms of review helpfulness are conducted with various settings. Some preliminary experiments to decide the parameters of models and learning models are introduced here.

For the review ranking experiments, the 200 reviews for each product domain are randomly shuffled 3 times, but with a fixed sequence to ensure the same data set between experiments. Since this is a ranking experiment, the training and test data can vary with each random shuffling. For each shuffling, the experiments are 5-fold cross-validated, thus totaling 15 trials. In the following sub-sections, some experiments to decide the clustering model and the number of clusters for the chosen clustering algorithm will be introduced.

7.4.1 Evaluating Clustering Algorithms

The k-means and DBSCAN are tested for their clustering methods because of their advantages, discussed earlier.

Table 27 shows how clusters are generated with the DBSCAN algorithm with core information sentences with LDA representation. The total number of core information sentences in the training data was 1818. The DBSCAN algorithm has an advantage in that the number of clusters does not need to be predefined. To use the cosine metric, the brute force algorithm is used to compute the pointwise distances and find nearest neighbors. However, it is possible the number of clusters is hard to control; the number of clusters drops too rapidly as the minimum number of samples increases. From the condition of two minimum samples for a cluster, the noise sentences, which are not assigned any cluster label, are too high to be used as a clustering model to find the information from sentences. When the minimum number of samples is one, the noise is one, indicating most sentences can have their information label from the clustering model. However, the distribution of sentences among clusters is too skewed; most sentences are labeled as the same

cluster. For instance, with 0.3 eps and 1 min sample, 672 out of 1818 sentences have the same clustering label, and 1111 clusters have only one sentence as a member. Therefore, DBSCAN is not appropriate to be used to cluster the same information-holding sentences.

Table 27. The Clustering Results of Core Information Sentences with DBSCAN

Min samples	eps							
	0.1		**0.2**		**0.3**		**0.4**	
	clusters	**noise**	**clusters**	**noise**	**clusters**	**noise**	**clusters**	**noise**
1	1697	1	1528	1	1116	1	556	1
2	26	1672	40	1489	20	1097	22	535
5	5	1722	14	1584	4	1159	1	631
10	3	1734	4	1670	3	1206	2	719

On the contrary, k-means does not have the same problem as the DBSCAN algorithm, but the k-means algorithm needs a pre-defined number of clusters. The number of clusters for k-means needs to be experimentally decided.

To examine the effect of the number of clusters, the same feature set was used except with a different number of clusters. Among the features, BASE, BK_LDA_CURRENT, CORE_LDA_CURRENT, PERI_LDA_CURRENT and OVERALL_LDA_CURRENT were used. Except the information type on examination, the number of clusters for other information types were set to 100. The experiment results with varying cluster numbers for each information type can be seen from Table 28 to Table 39.

In this paper, the cluster numbers for e-book reader reviews promising the best review ranking performances were 400, 800, 250, and 300 for background, core, peripheral and overall information clusters, respectively. The optimal cluster numbers for each information type is different depending on the product reviews due to the different number of sentences for each information type and their different semantic properties. The best performing cluster numbers for tent reviews were 50, 50, 50, and 50 for background, core, peripheral and overall information clusters. Contrary to the e-book reader reviews, the performance of review helpfulness estimation becomes worse as the number of clusters increases, as seen in Table 32 to Table 35. Lastly, for the jeans reviews, as the cluster numbers increase, the review ranking result shows performance improvement, as seen in Table 36 to Table 39. Thus, the optimal cluster numbers are set to 200, 700, 50, and 250.

Also, the number of samples in clusters seems to be distributed to capture the same information clusters. Among 1818 sentences, only 70 sentences were 1-sample clusters. The number of samples in a cluster ranges from 1 to 109. 252 out of 300 clusters are composed of from 1 to 10 sentences. For this reason, k-means is chosen to cluster the similar information sentences.

Table 28. Varying cluster numbers in Background sentences for e-book reader reviews

	100	**150**	**200**	**250**	**300**	**400**
Tau-b	0.187	0.19	0.193	0.197	0.199	0.204
rd	0.705	0.702	0.7	0.701	0.7	0.418
Top-n	0.379	0.378	0.368	0.358	0.357	0.326

Table 29. Varying cluster numbers in Core sentences for e-book reader reviews

	100	200	300	400	500	600	700	800
Tau-b	0.187	0.192	0.196	0.2	0.202	0.203	0.205	0.21
rd	0.705	0.701	0.702	0.702	0.701	0.702	0.704	0.702
Top-n	0.379	0.383	0.352	0.357	0.352	0.351	0.361	0.362

Table 30. Varying cluster numbers in Peripheral sentences for e-book reader reviews

	50	100	150	200	250
Tau-b	0.197	0.187	0.191	0.191	0.199
rd	0.699	0.705	0.705	0.419	0.417
Top-n	0.383	0.379	0.378	0.326	0.333

Table 31. Varying cluster numbers in Overall sentences for e-book reader reviews

	50	100	150	200	250	300
Tau-b	0.188	0.187	0.187	0.195	0.198	0.199
rd	0.704	0.705	0.705	0.419	0.417	0.418
Top-n	0.374	0.379	0.378	0.34	0.333	0.326

Table 32. Varying cluster numbers in Background sentences for tent reviews

	50	100	200	250
Tau-b	0.185	0.183	0.174	0.157
rd	0.438	0.437	0.437	0.437
Top-n	0.353	0.36	0.34	0.353

Table 33. Varying cluster numbers in Core sentences of tent reviews

	50	100	200	300	400	500
Tau-b	0.185	0.183	0.17	0.155	0.141	0.132
rd	0.438	0.437	0.436	0.436	0.435	0.437
Top-n	0.353	0.36	0.333	0.346	0.333	0.33

Table 34. Varying cluster numbers in Peripheral sentences of tent reviews

	50	100	150
Tau-b	0.185	0.183	0.173
rd	0.438	0.437	0.435
Top-n	0.353	0.36	0.373

Table 35. Varying cluster numbers in Overall sentences of tent reviews

	50	100	150	200	300
Tau-b	0.185	0.183	0.177	0.175	0.156
rd	0.437	0.437	0.436	0.436	0.436
Top-n	0.366	0.36	0.366	0.346	0.353

Table 36. Varying cluster numbers in Background sentences of jeans reviews

	50	70	100	150	200
Tau-b	0.149	0.149	0.153	0.159	0.161
rd	0.427	0.428	0.428	0.427	0.428
Top-n	0.293	0.286	0.32	0.306	0.313

Table 37. Varying cluster numbers in Core sentences for jeans reviews

	50	100	200	300	400	500	600	700
Tau-b	0.149	0.149	0.161	0.161	0.157	0.162	0.162	0.166
rd	0.427	0.428	0.427	0.427	0.429	0.43	0.43	0.429
Top-n	0.293	0.293	0.306	0.313	0.3	0.306	0.313	0.313

Table 38. Varying cluster numbers in Peripheral sentences for jeans reviews

	50
Tau-b	0.149
rd	0.427
Top-n	0.293

Table 39. Varying cluster numbers in Overall sentences jeans reviews

	50	100	150	200	250
Tau-b	0.185	0.183	0.177	0.175	0.16
rd	0.437	0.437	0.436	0.436	0.429
Top-n	0.366	0.36	0.366	0.346	0.3

7.5 Experiment Results

The first experiment was to decide between the two clustering models, k-means or DBSCAN. With the selected clustering model, then, the necessity for the manual

helpfulness score will be demonstrated through an experiment that can show the difference in the effects of features depending on the gold score.

With the decided clustering method and the gold helpfulness score, the various ways of representing sentence information are tested. Then, experiments to find the best combination of features are conducted. Moreover, experiments that could answer whether or not extracting information from each sentence shows better performance than the document unit based method on this review helpfulness ranking task.

7.5.1 Gold Standard Ranking Validation

As discussed earlier, the need to manually build review helpfulness scores can be shown by comparing different effects of features depending on the review helpfulness scores: helpfulness vote score (h_v) and helpfulness manual score (h_m).

Each feature set is added to the BASE features, which is the length of the reviews. Finding the best performing features is not the concern of this experiment, so the number of clusters were uniformly set to be 100 for each information type to reduce the computational complexity. The 200 reviews were randomly shuffled 3 times and divided into 5 folds: 4 folds (150 reviews) for training and 1 fold (50 reviews) for testing, a total of 15 cross-validations. The task is a ranking task, so every time the reviews are randomly shuffled, a different training and testing data set can be made. The order of shuffled reviews were kept the same for every shuffling to maintain the same data set.

In Table 40 and Table 41, it is important to note that the results with h_v and h_m should not be compared each other to see which one is higher than other. Since the

tau-b, rd and top-10 scores in each table are obtained from different gold standards, higher scores do not mean that one helpfulness standard score is better than the other. Since we observed that the helpfulness vote score (h_v) is biased in many ways, we need to see how the effect of the feature set is different depending on the helpfulness scores. From the tau-b scores with h_v score, we can see the highest score was obtained by the BASE + BOW feature combination. On the other hand, the feature set that shows the highest tau-b score from the results with h_m in Table 41 is the combination of BASE and ASPECT.

The lower the ranking distance score (*rd*) a review has, the closer the predicted rank of the review is found to its original rank. The BASE+BOW feature combination shows the lowest *rd* result with the h_v score, but the BASE+ASPECT combination is the best with the h_m. The best feature set in *rd* is consistent with tau-b score.

From the top-10 precision score, the best feature combination was BASE+BOW with h_v and with h_m. The BASE+ASPECT feature set shows a similar level of top-10 precision score, not significantly different, with BASE+BOW in the case of the h_m score.

As seen so far, the effect of feature sets on the review helpfulness ranking task is quite different depending on the review helpfulness score. That said, manual scoring for helpfulness of reviews is necessary to build a more accurate review helpfulness estimation model. All the following experiments are conducted with the manual helpfulness score.

Table 40. The different effects of features depending on the helpfulness vote score (h_v)

Combinations	**h_v**
BASE	-0.016 / 0.441 / 0.24[a]
BASE+BOW	0.171 / 0.383 / 0.386
BASE+SENT_TYPE	-0.034 / 0.435 / 0.246
BASE+SENTIMENT	0.030 / 0.432 / 0.206
BASE+READABILITY	-0.020 / 0.453 / 0.24
BASE+INFO_TYPE	0.028 / 0.435 / 0.186
BASE+ASPECT	-0.026 / 0.454 / 0.186
BASE+BK_LDA_CURRENT	0.058 / 0.442 / 0.266
BASE+CORE_LDA_CURRENT	0.088 / 0.427 / 0.286
BASE+PERI_LDA_CURRENT	0.038 / 0.446 / 0.286
BASE+OVERALL_LDA_CURRENT	0.056 / 0.444 / 0.266

[a] Score Order: Tau-b / ranking distance / top-10 precision

[b] Underlined Scores: the best result for each score column

Table 41. The different effects of features depending on the manual helpfulness score with all three score dimensions (h_m)

Combinations	**h_m**
BASE	0.026 / 0.42 / 0.226
BASE+BOW	0.081 / 0.415 / 0.353
BASE+SENT_TYPE	0.015 / 0.413 / 0.273
BASE+SENTIMENT	0.083 / 0.407 / 0.253
BASE+READABILITY	0.059 / 0.415 / 0.26
BASE+INFO_TYPE	0.064 / 0.425 / 0.233
BASE+ASPECT	0.143 / 0.391 / 0.3
BASE+BK_LDA_CURRENT	0.086 / 0.426 / 0.246
BASE+CORE_LDA_CURRENT	0.108 / 0.421 / 0.26
BASE+PERI_LDA_CURRENT	0.081 / 0.429 / 0.24
BASE+OVERALL_LDA_CURRENT	0.088 / 0.426 / 0.226

7.5.2 Sentence Representations

The representation of a sentence information can varied in six different ways: The TFIDF weighted frequency vector and the LDA topic vector, each separately combined with the three different sentence ranges (CURRENT, SURR and PREV_PREV).

The experiment setup is the same as the previous gold score validation experiment except for the number of clusters. For TFIDF word frequencies, the dimension of each sentence is the number of all word types from the corpus; most of the values are 0. Due to the high dimensionality of sentences, the computational complexity raises rapidly as the number of information sentence clusters increases. The number of clusters for each information type was 300, 300, 200 and 200 for background, core, peripheral, and overall information, respectively. The number of clusters were reduced from the best performing number of clusters (400, 800, 250, 300) to enable the computations to be practically possible, while keeping a similar ranking performance for each information type. With the dimension of all word types for each sentence and 800 clusters to be checked to find neighbors, the computational complexity becomes impossible to test. The cluster number of each information type for the LDA representation was set to the most promising cluster numbers.

Table 42. Different Effect of Sentence Information Representations

combinations	Tau-b	rd	Top-10
BASE+INFOTYPE+ALL_IF_TFIDF_CURRENT	0.217	0.379	0.373
BASE+INFOTYPE+ALL_IF_TFIDF+SURR	0.216	0.379	0.353
BASE+INFOTYPE+ALL_IF_TFIDF+PREV_PREV	0.217	0.379	0.366
BASE+INFOTYPE+ALL_IF_LDA+CURRENT	0.23	0.417	0.386
BASE+INFOTYPE+ALL_IF_LDA+SURR	0.228	0.377	0.386
BASE+INFOTYPE+ALL_IF_LDA+PREV_PREV	0.228	0.376	0.38

From the experiment results in Table 42, LDA representations show a higher score in tau-b and top-10. Though the difference between the two presentations is not remarkable, the LDA representation has more advantages in computational complexity and also ranking performance. If the data size increases, it is more likely that the computational complexity increases more with the TFIDF representation compared to the LDA representation, because the topic dimension is fixed in the LDA, unlike the TFIDF. Also, the increase in data size brings a positive effect on building more accurate clustering models, which consequently could result in an improved performance on this review ranking task.

We expected expanding the range of the sentences to include the surrounding or two previous sentences could help to extract accurate information from the sentences. However, from this experiment, we found that there was no advantage in including context sentences with current sentences. It might be because including contextual words from surrounding sentences causes the consecutive sentences that share the common words to be clustered as one cluster. It was found from the distribution of sentence clusters that the number of clusters that are composed of 2 or 3 sentences remarkably increases with SURR and PREV_PREV conditions.

7.5.3 The Best Feature Combinations

This experiment was to see the separate effects of each feature set and to prove the assumption that what information is delivered is more effective than how information is delivered when estimating review helpfulness. Moreover, it suggests the best performing feature combinations for this task. Including features from previous works (Chen & Tseng, 2011; Kim et al., 2006), various combinations of feature sets were examined.

From the representations of sentence information, LDA was chosen and only current sentence tokens are used to represent the sentence information based on the experiment results in Table 42. The semantic features based on LDA were separately tested depending on their information type. The number of clusters for each information type were optimized depending on the product following the previous experiment results in Table 28 to Table 39.

The experiment results are divided into 4 parts: (1) - (4). The results in (1) show the effects of the basic features (BASE, BOW), the length of reviews and bag-of-word features, and the results of combining these basic features with features related to how information is delivered in reviews: SENT_TYPE, SENTIMENT and READBILITY. The following results in (2) are all about how information is delivered, which is the effect of features that this study proposes. In (3), the separate feature sets of (2) were combined to find the effect as a whole and the difference between the effects of different information types. By comparing the results of (3) and (1), we can see the different effects of two feature dimensions that are posited from the beginning of this study: how information is delivered and

what information is delivered in product reviews. Lastly, the results in (4) show the best performing feature combination for the review helpfulness estimation task.

We added feature sets, one by one, to BASE features, the length of reviews. In terms of tau-b score, only the SENT_TYPE feature set shows a smaller tau-b score than the BASE feature set, though the differences in tau-b scores between feature sets are not statistically significant. In terms of rd and top-10 respectively, BOW and READABILITY are the least effective and SENTIMENT shows the lowest performance. Combining BASE+SENT_TYPE+SENTIMENT + READABILITY, all of which are related to how information is delivered, shows less effective performance in tau-b compared with the sole effect of SENTIMENT or BOW, possibly due to the negative effect of the SENT_TYPE features. Thus, we excluded the SENT_TYPE and combined the SENTIMENT and READABILITY features with the BASE to obtain the best performance in tau-b and rd, in bold. However, this feature combination does not ensure the best result in top-10. Since the measure top-10 is not as reliable as the other metrics as far as the nature of the metric, it should only be used as a complement. The result of this feature combination is compared with feature sets related to what information is delivered based on t-test ($p<0.05$). Adding BOW to the feature combination above shows higher performance, but the comparable feature combination should be only related to how information is delivered.

In (2), the features of what information is delivered are separately examined by adding each feature set to BASE. The CORE features, which extract what information is delivered from core sentences, and the ASPECT features, which use the groups of product aspect-indicating words to extract how much each product

aspect is dealt with in a review, are the top most effective features sets in terms of both tau-b and rd. Among the information types, BK and CORE seem to be more effective than PERI and OVERALL, which can be intuitively expected.

In (3), the information type-related features are combined to see the effects of feature combinations, and then to compare them with the results in (1), which are about how information is delivered. The feature combination of BASE + ASPECT + INFO_TYPE + BK + CORE + PERI + OVERALL is proved to be significantly more effective than the feature combination of BASE + SENTIMENT + READABILITY, which directly shows that extracting what information is delivered is more effective than how information is delivered. The significant difference in tau-b, rd and top-10 is still valid even in the combination without the ASPECT feature set, which requires building a list of product aspect-related words.

Strictly speaking, since the BASE feature is the length of the reviews, it is not one of the features related to what information is delivered. Thus, the combinations of information type-related features without the BASE feature are examined. It turns out that excluding the BASE feature improves the overall performance for all score domains. This indicates that the length of the review is a not effective cue to estimate the review helpfulness. This is perhaps due to the distribution of review length; reviews were collected to have a similar length as much as possible: for e-book reader reviews, sentence length ranges were 10~19, 20~29, 30~39, 40~49, and 50~59, and the number of reviews for each range were 125, 43, 17, 12 and 3, respectively. For the tent and jeans reviews, the review length was set to have as similar a length as possible from all the data. The distribution of reviews length is

10~19 (61), 20~29 (55), 30~39 (59), 40~49 (19), and 50~59 (6) for tent reviews and 1~9 (128), 10~19 (72) for jeans reviews.

By comparing various feature combinations in (4), the best feature combination that extracts both what and how information is delivered was BASE + BOW + SENTIMENT + READABILITY + INFO_TYPE + ASPECT + BK + CORE + PERI + OVERALL. The BASE feature is included to compare with other results, though the combination without BASE feature shows improved ranking scores.

This series of examinations on feature combinations is required to be examined in other product domains: tent and jeans in this study. In Table 44, the separate examination of feature sets related with how information is delivered corresponds to the results with e-book reader reviews. Thus, the BASE + SENTIMENT + READABILITY was chosen to be compared with the features related to what information is delivered.

Comparing the best feature combinations of what information is delivered with the feature combinations of how information is delivered shows a similar tendency, but there is a small difference between the two feature combinations, which results in only a significant differences in of tau-b scores. It is interesting to notice that adding PERI and OVERALL to BK and CORE does not improve the overall performance, which indicates extracting information from peripheral and overall type sentences does not help to estimate the review helpfulness depending on the product domain.

Combining the features of how information is delivered with those of what information is delivered does not improve the overall performance compared to

using only the features BASE + ASPECT + INFO_TYPE + BK + CORE + PERI + OVERALL.

Table 43. The Best Performing Feature Combination for e-book reader Reviews

	combinations	e-book reader		
		Tau-b	rd	Top-10
(1)	BASE	0.02	0.42	0.226
	BASE+BOW	0.081	0.415	0.353
	BASE+SENT_TYPE	0.015	0.413	0.273
	BASE+SENTIMENT	0.083	0.407	0.253
	BASE+READABILITY	0.059	0.415	0.26
	BASE+SENT_TYPE+SENTIMENT+READABILITY	0.025	0.425	0.24
	BASE+SENTIMENT+READABILITY	**0.094**	**0.404**	**0.273**
	BASE+BOW+SENTIMENT+READABILITY	0.202	0.375	0.386
(2)	BASE+ASPECT	0.143	0.391	0.3
	BASE+INFO_TYPE	0.064	0.425	0.233
	BASE+BK[a]	0.09	0.413	0.273
	BASE+CORE	0.194	0.381	0.34
	BASE+PERI	0.129	0.409	0.3
	BASE+OVERALL	0.082	0.416	0.266
(3)	BASE+INFO_TYPE+BK+CORE	0.233*[b]	0.371*	0.346
	BASE+INFO_TYPE+PERI+OVERALL	0.203*	0.387	0.333
	BASE+INFO_TYPE+BK+CORE+PERI+OVERALL	0.248*	0.368*	0.38*
	BASE+ASPECT+INFO_TYPE+BK+CORE+PERI+OVERALL	0.299*	0.351*	0.4*
	INFO_TYPE+BK+CORE+PERI+OVERALL	0.404*	0.316*	0.493*
	ASPECT+INFO_TYPE+BK+CORE+PERI+OVERALL	0.512*	0.277*	0.586*
(4)	BASE+SENTIMENT+INFO_TYPE+BK+CORE	0.283*	0.349*	0.386*
	BASE+SENTIMENT+INFO_TYPE+PERI+OVERALL	0.242*	0.368*	0.406*
	BASE+SENTIMENT+INFO_TYPE+BK+CORE+PERI+OVERALL	0.289*	0.351*	0.4*
	BASE+BOW+SENTIMENT+INFO_TYPE+BK+CORE+PERI+OVERALL	0.291*	0.350*	0.433*
	BASE+BOW+SENTIMENT+READABILITY+INFO_TYPE+BK+CORE+PERI+OVERALL	0.298*	0.347*	0.4*
	BASE+BOW+SENTIMENT+READABILITY+INFO_TYPE+ASPECT+BK+CORE+PERI+OVERALL	0.318*	0.339*	0.44*

[a] The representation and the range of sentence conditions are not specified on this table, they are all LDA topic vectors with the CURRENT sentence range.

[b] The star(*) indicates the average score is better than the average score of BASE+SENTIMENT+READABILITY combination. The results are tested their statistical significance by t-test ($p < 0.05$).

Table 44. The Best Performing Feature Combination for Tent Reviews

	combinations	Tent		
		Tau-b	rd	Top-10
(1)	BASE	0.078	0.375	0.32
	BASE+BOW	0.119	0.415	0.253
	BASE+SENT_TYPE	0.029	0.378	0.32
	BASE+SENTIMENT	0.123	0.393	0.313
	BASE+READABILITY	0.124	0.399	0.306
	BASE+SENT_TYPE+SENTIMENT+READABILITY	0.141	0.382	0.38
	BASE+SENTIMENT+READABILITY	**0.174**	**0.383**	**0.373**
	BASE+BOW+SENTIMENT+READABILITY	0.186	0.400	0.3
(2)	BASE+ASPECT	0.186	0.385	0.333
	BASE+INFO_TYPE	0.12	0.415	0.3
	BASE+BK	0.153	0.406	0.313
	BASE+CORE	0.186	0.395	0.333
	BASE+PERI	0.136	0.413	0.3
	BASE+OVERALL	0.12	0.413	0.306
(3)	BASE+INFO_TYPE+BK+CORE	0.207 [a]	0.394	0.353
	BASE+INFO_TYPE+PERI+OVERALL	0.156	0.405	0.306
	BASE+INFO_TYPE+BK+CORE+PERI+OVERALL	0.192	0.398	0.32
	BASE+ASPECT+INFO_TYPE+BK+CORE+PERI+OVERALL	0.231* [a]	0.385	0.346
	INFO_TYPE+BK+CORE+PERI+OVERALL	0.273*	0.377	0.306
	ASPECT+INFO_TYPE+BK+CORE+PERI+OVERALL	0.277*	0.372	0.293
(4)	BASE+SENTIMENT+INFO_TYPE+BK+CORE	0.206	0.388	0.4
	BASE+SENTIMENT+INFO_TYPE+PERI+OVERALL	0.183	0.399	0.373
	BASE+SENTIMENT+INFO_TYPE+BK+CORE+PERI+OVERALL	0.223	0.385	0.38
	BASE+BOW+SENTIMENT+INFO_TYPE+BK+CORE+PERI+OVERALL	0.213	0.390	0.333
	BASE+BOW+SENTIMENT+INFO_TYPE+BK+CORE+PERI+OVERALL+READABILITY	0.217	0.388	0.333
	BASE+SENTIMENT+INFO_TYPE+BK+CORE+PERI+OVERALL+READABILITY	0.206	0.393	0.36
	BASE+SENTIMENT+INFO_TYPE+ASPECT+BK+CORE+PERI+OVERALL+READABILITY	0.215	0.385	0.34

[a] The star(*) indicates the average score is better than the average score of the BASE+SENTIMENT+READABILITY combination. The results are tested their statistical significance by t-test ($p < 0.05$).

Table 45 The Best Performing Feature Combination for jeans reviews

	combinations	Jeans		
		Tau-b	rd	Top-10
(1)	BASE	0.028	0.445	0.2
	BASE+BOW	0.080	0.436	0.286
	BASE+SENT_TYPE	-0.059	0.457	0.22
	BASE+SENTIMENT	**0.061**	**0.432**	**0.193**
	BASE+READABILITY	-0.003	0.446	0.2
	BASE+SENT_TYPE+SENTIMENT+READABILITY	0.021	0.440	0.24
	BASE+BOW+SENTIMENT	0.168	0.394	0.406
(2)	BASE+ASPECT	0.1	0.425	0.233
	BASE+INFO_TYPE	0.108	0.415	0.28
	BASE+BK	0.071	0.428	0.226
	BASE+CORE	0.161	0.41	0.286
	BASE+PERI	0.005	0.46	0.2
	BASE+OVERALL	0.057	0.432	0.246
(3)	BASE+INFO_TYPE+BK+CORE	0.19* [a]	0.399*	0.32*
	BASE+INFO_TYPE+PERI+OVERALL	0.144*	0.403*	0.32*
	BASE+INFO_TYPE+BK+CORE+PERI+OVERALL	0.195*	0.398*	0.34*
	BASE+ASPECT+INFO_TYPE+BK+CORE+PERI+OVERALL	0.246*	0.377*	0.333*
	INFO_TYPE+BK+CORE+PERI+OVERALL	0.297*	0.372*	0.493*
	ASPECT+INFO_TYPE+BK+CORE+PERI+OVERALL	0.353*	0.348*	0.46*
(4)	BASE+SENTIMENT+INFO_TYPE+BK+CORE	0.232*	0.380*	0.393*
	BASE+SENTIMENT+INFO_TYPE+PERI+OVERALL	0.188*	0.391*	0.393*
	BASE+SENTIMENT+INFO_TYPE+BK+CORE+PERI+OVERALL	0.240*	0.374*	0.406*
	BASE+BOW+SENTIMENT+INFO_TYPE+BK+CORE+PERI+OVERALL	0.243*	0.375*	0.446*
	BASE+BOW+SENTIMENT+INFO_TYPE+ASPECT+BK+CORE+PERI+OVERALL	0.275*	0.366*	0.453*

[a] The star(*) indicates the average score is better than the average score of BASE+SENTIMENT. The results are tested their statistical significance by t-test ($p < 0.05$).

In Table 45, the feature combinations were examined with jeans products. In (1), only the SENTIMENT feature set was used to ensure the best score in tau-b and rd. The result of this combination is compared with the best feature combination in (3). The comparison supports the idea that considering what information is delivered is more important than how information is delivered in estimating review helpfulness.

The best performance feature combination was shown to be BASE + BOW + SENTIMENT + INFO_TYPE + ASPECT + BK + CORE + PERI + OVERALL. The result of the feature combination effect with jeans products corresponds to that of e-book reader reviews, except the most effective feature combination in (1) block.

Overall, through a series of experiments examining feature combinations in three product domains, we found extracting what information is delivered in the way that this study proposes is more effective than extracting how information is delivered in terms of the sentiment, sentence type and readability of reviews to automatically estimate the review helpfulness.

To see the difference in estimating review helpfulness based on how information is delivered and what information is delivered, we present an example of two reviews with extracted features and their helpfulness estimation difference. For the features of how information is delivered, the length (BASE) + SENTIMENT + READABILITY feature combination was chosen, which is the best performing result for e-book reader reviews. For the features of what information is delivered, the feature combination of ASPECT+INFO_TYPE+ALL_IF_LDA_CURRENT was used. In Figure 10, two product reviews (#11683 and #6305) are shown with features on how information is delivered. As seen, the estimated helpfulness scores, the order of the review rank between two, #11683 > #6305, changes to #11683 < #6305 with the estimated helpfulness value when considering only how information is delivered.

On the other hand, Figure 11 shows the feature extraction of what information is delivered and the estimated helpfulness values. The results show that the feature

extraction based on what information is delivered can correctly estimate the rank of helpfulness scores of the two reviews. These real data examples are presented to help understand the way feature extraction is implemented for this study.

#11683 After having lost a Kindle Keyboard due to a broken screen, decided to go for the 2nd gen Paperwhite. NOTE - make sure to check the screen for imperfections. I had to send two back, one with a bright white spot when the light was on, a second with a dark spot that didn't go away. Both were small and I probably could have lived with them, but thought they might becoming distracting so sent them back. Amazon customer support was great and my 3rd device was perfect. I'd dock it 1/2 star if I could, but enjoy using it too much to say it's a 4 star device, and the one I have now is worth 5. I thought I might miss the hardware buttons for both page turning and typing, but this new screen is so responsive I really haven't missed them. The touch screen is much better then I anticipated. The light was what I was most excited about when I first heard rumors of the Paperwhite. I previously used the Kindle Lighted cover for my Kindle Keyboards (which I am convinced contributed to one cracked screen), so love having it built in to the device and being able to adjust the brightness. It really is the perfect device for reading: great size, price, display, battery, and light. I hope e-ink readers don't disappear in favor of tablets.

Features of How Information is delivered

<BASE-LENGTH>
"the number of words": 231
" the number of sentences": 12
<READABILITY>
"FLESCH_KINCAID": 7.2,
"COLEMAN_LIAU": 7.54
"ARI": 8.5
"GUNNING_FOG": 9.60
"FLESCH_READING": 77.57
"SMOG": 3.1

<SENTIMENT>
"% of positive sentences": 0.583
"% of negative sentences ": 0.416
" the number of positive sentences": 7
"the number of negative sentences": 5
"the number of opinion sentences": 10
"the % of positive sentences in all opinion sentences": 0.7
"the % of negative sentences to opinion sentences": 0.5
"% of opinion sentences ": 0.833

<Helpfulness Scores>
Helpfulness Score: 4.777, Estimated Helpfulness Score: 3.774

#6305 The size and weight are good for a reader. If you only use this reader for books you will be pleased. AFTER using these past few days it locks up and you have to turn it on/off which is annoying. It is not a great choice if you like to read magazines. I also think it should have come with the adapters in the same box. The battery life based on 30 min. a day seems silly. I easily read 2 hours or more a day. It is better than my old Sony Reader that was heavy and you had to load books from your computer. I had a lighted background and the battery did not last long while reading. It is lighter than my Nook Color. Of course it is smaller and lighter than my Eepad which makes it easier to carry in my purse.

<Features of How Information is delivered>

BASE-LENGTH
"the number of words": 146
" the number of sentences": 13

READABILITY
"FLESCH_KINCAID": 2.9,
"COLEMAN_LIAU": 4.04
"ARI": 2.5
"GUNNING_FOG": 6.4
"FLESCH_READING": 94.15
"SMOG": 3.1

SENTIMENT
"% of positive sentences": 0.923
"% of negative sentences ": 0.076
" the number of positive sentences": 12
"the number of negative sentences": 1
"the number of opinion sentences": 13
"the % of positive sentences in all opinion sentences": 0.923
"the % of negative sentences to opinion sentences": 0.0769
"% of opinion sentences ": 1.0

<Helpfulness Scores>
Helpfulness Score: 4.166, Estimated Helpfulness Score: 3.933

Figure 10. An example of feature extractions for how information is delivered and helpfulness estimation

Features of What Information is delivered for a review(#11683)

The number of each information types
"core_cnt": 8
"bk_sent_cnt": 1
"peri_cnt": 1
"overall_cnt": 2
"nonrel_cnt": 0

The % of each information types
"core_cnt_norm": 0.666
"bk_sent_cnt_norm": 0.0833
"peri_cnt_norm": 0.0833
"overall_cnt_norm": 0.166
"nonrel_cnt_norm": 0.0

The number of specific topics for each information type
"c_33": 1, "c_70": 1, "c_80": 1, "b_81": 1, "p_6": 1, "o_3": 1 …

The % of specific topics for each information type
"c_33_normed_all": 0.0833
"b_81_normed_all": 0.0833
…

The features using product aspect-related expressions (ASPECT)
"The number of a different aspects in a review": 8
"The number of a different aspects in a review normalized its sentence counts": 0.666
"The number of a different aspects in a review normalized its word counts": 0.0346
"The averaged length (# of words) of each aspects": 18.125
"The 0^{th} , 25^{th}, 50^{th}, 75^{th}, 100^{th} percentile of aspect length (# of words) ": 25_perc_len_asp": 0.0259 …"100_perc_len_asp": 0.259

<Helpfulness Scores>

Helpfulness Score: 4.777
Estimated Helpfulness Score: 4.047

Features of What Information is delivered for a review(#6305)		
The number of each information types "core_cnt": 11 "bk_sent_cnt": 1 "peri_cnt": 1 "overall_cnt": 0 "nonrel_cnt": 0 **The % of each information types** "core_cnt_norm": 0.846 "bk_sent_cnt_norm": 0.0769 "peri_cnt_norm": 0.0769 "overall_cnt_norm": 0.0 "nonrel_cnt_norm": 0.0	**The number of specific topics for each information type** "c_37": 1, "c_55": 1, "c_103": 1, … "b_33": 1, "p_39": 1 … **The % of specific topics for each information type** "c_37_norm": 0.076, "c_418_norm": 0.076 …	**The features using product aspect-related expressions (ASPECT)** "The number of a different aspects in a review": 3 "The number of a different aspects in a review normalized its sentence counts": 0.230 "The number of a different aspects in a review normalized its word counts": 0.0205 "The averaged length (# of words) of each aspects": 23.66 "The 0^{th} , 25^{th}, 50^{th}, 75^{th}, 100^{th} percentile of aspect length (# of words) "25_perc_len_asp": 0.130 … "100_perc_len_asp": 0.267
<Helpfulness Scores> Helpfulness Score: 4.166 Estimated Helpfulness Score: 3.802		

Figure 11. An example of feature extractions of what information is delivered and helpfulness estimation

7.5.4 Whole Document vs Separate Sentences

This experiment compared the approach of extracting LDA vectors from whole reviews to our approach, which makes every LDA vector from separate sentences.

To extract information from the whole review, all sentences in a review have to be converted into one LDA vector. Since it is not about extracting what information each sentence contains, the topic numbers and the values of LDA converted review documents are used as features to train the SVR model. The experiment results in Table 46 prove that it is more worthwhile and possible to extract information from separate sentences to find the information of a review rather than using one vector from the whole review.

Table 46. The Comparison of Document-based with Sentence-based Approach

	e-book reader			**tent**			**jeans**		
	Tau-b	rd	Top-10	Tau-b	rd	Top-10	Tau-b	rd	Top-10
BASE	0.021	0.42	0.226	0.074	0.396	0.313	0.061	0.44	0.213
DOC	0.092	0.419	0.286	0.005	0.473	0.22	-0.0007	0.465	0.26
SENT[a]	0.248*[b]	0.368*	0.38*	0.192*	0.398*	0.32*	0.195*	0.398*	0.34*

[a] SENT is the combination of INFO_TYPE + BK + CORE + PERI + OVERALL.

[b] The star(*) indicates the difference in score is significant compared to the DOC results based on t-test ($p < 0.05$).

7.5.5 No Distinction on Information types

Another experiment was conducted to answer if extracting semantic information based on review information types is more effective than extracting information without any information types.

The approach of not considering any information types is to unify the all information types into one and extract information in the same way as our approach. The number of clusters was set to 1700, 200 and 1000 respectively for e-

book reader, tent and jeans reviews, accumulating the cluster numbers for each information type. The experiment results in Table 47 confirms our assumption that extracting the semantics of sentences within the same type of information is more effective in this review ranking task.

Table 47. The Comparison of results with no information type distinction

	e-book reader			**tent**			**Jeans**		
	Tau-b	rd	Top 10	Tau-b	rd	Top 10	Tau-b	rd	Top 10
NO INFO	0.098	0.407	0.34	0.149	0.408	0.293	0.079	0.436	0.286
INFO[a]	0.248 *[b]	0.368 *	0.38	0.192 *	0.398	0.32	0.195 *	0.398 *	0.34

[a] INFO is the combination of INFO_TYPE + BK + CORE + PERI + OVERALL.

[b] The star(*) indicates the difference in score is significant compared to the NO INFO results based on t-test ($p < 0.05$).

7.5.6 Review Helpfulness Evaluation with Predicted Sentence Information Types

In the previous experiment results, the information type for review sentences is given to estimate the review helpfulness. However, to practically apply this approach to real-world data, the estimation of review helpfulness should still be possible with the predicted information type of review sentences. To predict the information type of sentences, the CRF classification model with the best performance feature combination (TFIDF_POS, POSITION, GRAMMAR) was used. Then, the predicted information types of sentences are given to estimate the review helpfulness. In Table 48, the feature combination with given information

types (GIVEN_INFOTYPE) was BASE + INFO_TYPE + BK + CORE + PERI + OVERALL, which is a feature combination of only information type-related features. This same feature combination is also used for the trial with predicted information types (PREDICTED_INFOTYPE). Though the prediction of sentence information types makes inevitable errors in the information type classification results with the CRF model in Table 21 to Table 23, the estimation of review helpfulness with automatically recognized information types still shows a performance as accurate as the review helpfulness estimation results with given sentence information types.

Moreover, compared to the results when extracting only how information is delivered based on previous studies (SENT_TYPE, SENTIMENT, and READABILITY), the PREDICTED_INFOTYPE results in tau-b score is still superior to e-book reader and jeans reviews in all scores. However, using only BASE + INFO_TYPE + BK + CORE + PERI + OVERALL does not make a significant difference with the features of how information is delivered. We could not find the exact cause of this, but it is possibly due to the effect of comparably low correlation of helpfulness scores between annotators for tent reviews.

Table 48. Comparison of results with previous studies and given information type and results with predicted information types

	e-book			**tent**			**jeans**		
	Tau-b	rd	Top-10	Tau-b	rd	Top-10	Tau-b	rd	Top-10
GIVEN_INFOTYPE	0.248*	0.368*	0.38*	0.192	0.398	0.32	0.195*	0.398*	0.34*
PREDICTED_INFOTYPE	0.222*	0.378*	0.366*	0.159	0.409	0.3	0.154*	0.412*	0.333*

7.5.7 The Product Domain Adaptation

To adopt the approach of this study, the annotation of each sentence information type needs to be avoided as much as possible due to the inefficiency of building annotated data for each product domain. Therefore, we conducted an experiment to show how much it is possible to train on reviews from one product domain and test on another product domain.

In Table 49, for different-domain training and testing environments, 200 reviews were used for training and 200 reviews were tested for 1 fold. On the other hand, for the same-domain training and testing environments, the 15-fold cross validation was performed with 160 and 40 reviews for training and testing, respectively. To apply the trained model from one domain to another domain, the features related with fixed word lists are not used. The feature combination used for these experiments was INFO_TYPE + BK + CORE + PERI + OVERALL.

Table 49. The result of product domain adaptation

Train \ Test	e-book Reader	Tent	Jeans
e-book Reader	0.404 / 0.316[a]	0.293 / 0.429	0.296 / 0.430
Tent	0.386 / 0.407	0.273 / 0.377	0.262 / 0.428
Jeans	0.403 / 0.414	0.285 / 0.433	0.297 / 0.372

[a] The scores are ordered as tau-b / ranking distance. The top-n score is not included because the number of testing reviews is different between the same-domain and different-domain experiments.

The anticipated result was to see the best result for the same-domain training and testing experiments. This is true for only e-book reader reviews in both tau-b and ranking distance scores. For the tent and jeans product domains, testing on e-book

reader reviews records the best results in tau-b score. In the ranking distance scores, the same-domain training and testing experiments show the best performance. It is difficult to find the reason for the unexpected results. It may be due to the manual helpfulness agreements between annotators; the e-book reader reviews were annotated with the highest agreement in review scores between annotators among all three-product domains. From the results of this experiment, it is important to notice that the domain adaptation across different product domains seems to still be effective compared with the features on how information is delivered in previous experiments. However, the cross-domain adaptation experiments seem to be conducted with a larger number of reviews and domains to conclude the applicability because of the unexpectedly high performance on different-domain training and testing results in tau-b score.

7.6 Summary

This entire chapter was dedicated to showing the experiment results of estimating review helpfulness based on review information types. The experiments wre conducted with all three product domain reviews to show the results are independent of product domain. The features used to estimate the review helpfulness were introduced: baseline features, features from previous studies that are related to *how* review information is delivered, and features that reveal *what* information a review delivers.

Through a series of experiments, we first showed the differences in effectiveness of feature sets depending on the helpfulness scores: vote (h_v) and manual (h_m). The

results pointed out that the effect of each feature is different with the two helpfulness scores. Thus, h_m was chosen for all the following experiments.

Secondly, the ways of representing sentence meaning were examined with the review helpfulness estimation experiment. The results showed that the LDA based approach with only tokens from the current sentence is more effective than the tfidf-based approach or expanding the range of sentences.

We also established a process showing the different effects of features on how and what information is delivered on estimating the review helpfulness, and then showed the best performing feature combinations in estimating the review helpfulness. According to the results, we found that the combination of the proportion of each review information type, the specific meaning of each sentence within each information type, and the number of product aspect-indicating words for each group of product aspect words outperform the features of how information is delivered based on previous studies. Additionally, we found the best performing feature combination for estimating review helpfulness in different product domains.

Further, since we attempted to extract the meaning of each sentence and aggregate the extracted meaning to estimate the review helpfulness, this result had to be compared with the approach of extracting the information from the entire review document. We showed our approach outperforms the document unit-based approach.

Our study subsequently proposed to categorize the review information types and use them to more accurately extract the information that each sentence holds. This also had to be compared with an approach that does not have information type distinction. The results sufficiently showed that dividing the review information

types is necessary to extract the meaning of sentences and further to evaluate the reviews.

Additionally, all the previous experiments of estimating review helpfulness were conducted with given review information types. To automate the entire process of the review helpfulness estimation, automatically recognized information types should be used. Though there was a bit of decrease in performance of review estimation, the performance with recognized information types still showed promise to estimate the review helpfulness.

Lastly, cross-product domain experiments were conducted to see the possibility of adopting a model trained on one domain to another. Though the results showed an abnormality on the experiment results, that is, for tent and jeans reviews, the cross-domain results performed better than the same-domain results in tau-b score, the experiments indicated the possible applicability between different product domain data.

8 Conclusions and Future Directions

Though the task of this study was restricted to automatic evaluation of product review helpfulness, it opens the possibility of evaluating texts based on what information the text delivers, which resembles the cognition of human beings. Since the purpose of product reviews is to share valuable information related with personal experience of products, we focused on finding what information people seek in product reviews and used it to estimate the helpfulness of product reviews. To find the information people look for in product reviews, we categorized the review information types depending on the target of the information and the effectiveness or usefulness of the information. We assumed that dividing the information types of sentences can help to more accurately extract the meaning or information of each sentence, thus resulting in finding what information people consider useful when reading product reviews. To estimate the helpfulness of product reviews, the helpfulness score was manually annotated. Firstly, to determine the review information types of review sentences, linguistically motivated features are employed to examine the various supervised and unsupervised models. By a series of experiments, we found a best performing information type classification model, which takes contextual information into account. Secondly, the estimation of review helpfulness was conducted with the determined review information types of review sentences. We began by examining the features of previous studies and then looked at the features of our approach, which was to convert the meaning or information of sentences to topic vectors and make clusters of similar information-holding sentences to find the information of

review sentences through review helpfulness estimation experiments. Then, we trained a computational model that estimated the helpfulness of product reviews with the features and the manually-built helpfulness scores.

8.1 Summary of Contribution and Results

8.1.1 Categorization of Information Types

In this paper we proposed the categorization of information types of review sentences depending on the target of the information and the effectiveness of the information. The information types are divided into background information (about the reviewer's previous experience or expertise that can help to raise the credibility of the review), core information (about the product itself and its aspects), peripheral type (about something not directly related with the product itself) and overall information (summary and final judgment, decision or recommendation). These information types are used to help to extract the information of each sentence and find more or less useful information sentences for readers.

8.1.2 Review Helpfulness Annotation

The helpfulness of product reviews is automatically obtainable from highly developed online review systems, such as Amazon.com, by using the total helpful / unhelpful votes. However, the votes are biased in many ways (J. Liu et al., 2007). Thus, our study proposed a method of annotating helpfulness scores and carried out such an annotation. We evaluated annotated scores in various ways and clarified

the difficulty and problems of annotating helpfulness scores of product reviews. The helpfulness scores were used to train a model to estimate the review helpfulness.

8.1.3 Features for Recognizing Review Information Types

To recognize review information types, various features were examined. Basically, with an assumption that there are words that only appear for certain review information types, the tf-idf weighted frequency of words and words with part-of-speech tags were tested. Additionally, an alternative way of representing sentences, the LDA-based approach, was examined and compared with the tf-idf weighted frequency. Moreover, the sentence position, the subject, auxiliary verb and main verbs were found to be effective in recognizeing the review information types.

8.1.4 Computational Modeling of Information Type Recognition

Recognition of review information types can be achieved by various unsupervised and supervised models. For the unsupervised models, DBSCAN and K-means clustering methods were examined and evaluated by how they can categorize the review information types. Though these unsupervised models have the advantage of not requiring manual annotation of review information types, they can not accurately cluster the same information type sentences. For supervised models, the SVM classification model was applied to classify review information types. Though this model showed an improvement in accuracy, it still was not

effective enough to specifically find peripheral and overall information sentences. With the observation that the information types of sentences cannot identified by only through the sentences themselves, we examined the conditional random field model that predicts the sequence of labels, considering the contextual information from surrounding sentences. With the crf model, we obtained the highest accuracy for recognizing each information type sentence.

8.1.5 Features and Computational modeling for Estimating Review Helpfulness

Review helpfulness can be estimated through various factors. Those factors are divided into two dimensions: how information is delivered and what information is delivered. Most previous studies are focused on how reviews deliver information, such as the sentence types, sentiment or the readability of product reviews. Our study assumed that the review helpfulness is more dependent on what information reviews deliver, thus, we attempted to identify and extract the information each sentence holds. The proportion of review information types in reviews was tested as a factor. More specifically, to extract what information each sentence holds, we proposed converting review sentences to LDA-based topic vectors and learning a cluster model to find similar information-holding sentence clusters with the topic vectors for each information type. These clusters then are used to extract what specific information a sentence contains and this was used to learn a support vector regression model that estimated the review helpfulness. Through various experiments, we showed that our proposed approach of using what information a

review delivers is more effective than the factors related with how information is delivered in reviews to estimate review helpfulness.

8.2 Future Directions and Open Problems

8.2.1 Extraction of Sentence Information

The meaning of sentences can be represented in various ways. Though we only tested using the surrounding sentences to extract the information of current sentences in a naive way, the degree of dependency on contextual information to understand the meaning of current sentences can vary depending on the sentence. For sentences that are composed mostly of pronouns, such that the meaning of the sentences depends on previous sentences, previous sentences should be consider in information extraction. On the other hand, for sentences that can be sufficiently understood with context, extraction of sentence meaning may solely depend on that sentence itself.

Moreover, the meaning of sentences might be more precisely represented with a more sophisticated learning model, such as a deep neural network, which could results in finding more accurate similar information-bearing clusters. With the growing interest on the deep neural networks, it is necessary to examine the method of representing sentence meaning for the same task.

8.2.2 Topic based Clustering

The ideal way to extract what information a review delivers is to find what product aspects each sentence is about. We have seen from the experiment results that the using groups of product feature keywords to extract specifically what product aspects each sentence is about can be used as one of the most effective features to estimate review helpfulness. If we could train a cluster model that can gather sentences, not just depending on the distance between topic vectors, but depending on the target of the information, the cluster model could be more effective in estimating the review helpfulness.

8.2.3 Remaining Practical Issues

To apply this approach to real world data, there are some obstacles that have to be conquered. Firstly, the helpfulness score was manually annotated by trained experts in this study. However, as current online review systems only use helpful / unhelpful votes, it could be more difficult to encourage people give the 5 or 7 scale score for review helpfulness. In addition, for reviews on forums or blogs, it is impossible to make a uniform scoring system. Therefore, another way of calculating the helpfulness of reviews should be proposed.

Secondly, in this study, the information types were manually annotated. It is practically impossible to make annotated data for sentence information types of all product domains. Ideally, the recognition of review information types should be performed in an unsupervised way..

8.2.4 Expandability of Review Information Types

In this study, only the helpfulness of product reviews was examined by extracting what information a review was about. There is a possibility of expanding this approach to other tasks that still need to find what information the texts are about. Persuasive texts, for instance, are also commonly studied in an attempt to automatically determine what stance the writer holds. However, there is also a need to estimate which persuasive text is more well-written and supported by the most readers. In addition, for academic essay rating systems, it is important to find not only grammatical mistakes or the use of vocabulary, but what the essay is arguing and what support the author has given for her idea. We expect that this approach will be expanded to deal with these problems and others that are related with finding what information texts offer.

REFERENCES

Agresti, A. (2010). *Analysis of ordinal categorical data* (Vol. 656): John Wiley & Sons.

Arthur, D., & Vassilvitskii, S. (2007). *k-means++: The advantages of careful seeding.* Paper presented at the Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms.

Basak, D., Pal, S., & Patranabis, D. C. (2007). Support vector regression. *Neural Information Processing-Letters and Reviews, 11*(10), 203-224.

Blei, D. M., Ng, A. Y., & Jordan, M. I. (2003). Latent dirichlet allocation. *the Journal of machine Learning research, 3*, 993-1022.

Burges, C. J. (1998). A tutorial on support vector machines for pattern recognition. *Data mining and knowledge discovery, 2*(2), 121-167.

Cai, X., & Li, W. (2011). Enhancing sentence-level clustering with integrated and interactive frameworks for theme-based summarization. *Journal of the American Society for Information Science and Technology, 62*(10), 2067-2082.

Cao, Q., Duan, W., & Gan, Q. (2011). Exploring determinants of voting for the "helpfulness" of online user reviews: A text mining approach. *Decision Support Systems, 50*(2), 511-521.

Chen, C. C., & Tseng, Y.-D. (2011). Quality evaluation of product reviews using an information quality framework. *Decision Support Systems, 50*(4), 755-768.

Deerwester, S. (1988). Improving information retrieval with latent semantic indexing.

DuBay, W. H. (2004). The Principles of Readability. *Online Submission*.

Ester, M., Kriegel, H.-P., Sander, J., & Xu, X. (1996). *A density-based algorithm for discovering clusters in large spatial databases with noise.* Paper presented at the Kdd.

Ghose, A., & Ipeirotis, P. G. (2007). *Designing novel review ranking systems: predicting the usefulness and impact of reviews.* Paper presented at the Proceedings of the ninth international conference on Electronic commerce.

Ghose, A., & Ipeirotis, P. G. (2011). Estimating the helpfulness and economic impact of product reviews: Mining text and reviewer characteristics. *Knowledge and Data Engineering, IEEE Transactions on, 23*(10), 1498-1512.

Hatzivassiloglou, V., & Wiebe, J. M. (2000). *Effects of adjective orientation and gradability on sentence subjectivity.* Paper presented at the Proceedings of the 18th conference on Computational linguistics-Volume 1.

Hu, M., & Liu, B. (2004). *Mining and summarizing customer reviews.* Paper presented at the Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining.

Kendall, M. G. (1938). A new measure of rank correlation. *Biometrika, 30*(1/2), 81-93.

Kim, S.-M., Pantel, P., Chklovski, T., & Pennacchiotti, M. (2006). *Automatically assessing review helpfulness.* Paper presented at the Proceedings of the 2006 Conference on empirical methods in natural language processing.

Knerr, S., Personnaz, L., & Dreyfus, G. (1990). Single-layer learning revisited: a stepwise procedure for building and training a neural network *Neurocomputing* (pp. 41-50): Springer.

Korfiatis, N., García-Bariocanal, E., & Sánchez-Alonso, S. (2012). Evaluating content quality and helpfulness of online product reviews: The interplay of review helpfulness vs. review content. *Electronic Commerce Research and Applications, 11*(3), 205-217.

Landis, J. R., & Koch, G. G. (1977). The measurement of observer agreement for categorical data. *biometrics*, 159-174.

Liu, J., Cao, Y., Lin, C.-Y., Huang, Y., & Zhou, M. (2007). *Low-Quality Product Review Detection in Opinion Summarization.* Paper presented at the EMNLP-CoNLL.

Liu, Y., Huang, X., An, A., & Yu, X. (2008). *Modeling and predicting the helpfulness of online reviews.* Paper presented at the Data Mining, 2008. ICDM'08. Eighth IEEE International Conference on.

Myers, J. L., Well, A., & Lorch, R. F. (2010). *Research design and statistical analysis*: Routledge.

O'Mahony, M. P., & Smyth, B. (2009). *Learning to recommend helpful hotel reviews.* Paper presented at the Proceedings of the third ACM conference on Recommender systems.

Pang, B., & Lee, L. (2004). *A Sentimental Education: Sentiment Analysis Using Subjectivity Summarization Based on Minimum Cuts.* Paper presented at the Proceeding in 42nd Annual Meeting of the Association for Compuational Linguistics.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., . . . Dubourg, V. (2011). Scikit-learn: Machine learning in Python. *the Journal of machine Learning research, 12*, 2825-2830.

Smola, A. J., & Schölkopf, B. (2004). A tutorial on support vector regression. *Statistics and computing, 14*(3), 199-222.

Sojka, P. (2010). *Software framework for topic modelling with large corpora.* Paper presented at the In Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks.

Thelen, M., & Riloff, E. (2002). *A bootstrapping method for learning semantic lexicons using extraction pattern contexts.* Paper presented at the Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10.

Vapnik, V. (1995). *The Nature of Statistical Learning Theory*: Springer.

Wiebe, J. (2000). *Learning subjective adjectives from corpora.* Paper presented at the AAAI/IAAI.

Zhang, Z., & Varadarajan, B. (2006). *Utility scoring of product reviews.* Paper presented at the Proceedings of the 15th ACM international conference on Information and knowledge management.

Appendix I. Product lists and Ids from Amazon.com

<e-book Readers>

- Fire HD 6, 6" HD Display, Wi-Fi, 8 GB (id: B00KC6I06S)
- Kindle Paperwhite, 6" High Resolution Display (212 ppi) with Built-in Light (id: B00JG8GOWU)

<Sports & Outdoors-Tent>

- Coleman Sundome Tent (id: B004J2KDHK)
- Coleman 8-Person Instant Tent (id: B003QUT9OE)
- Coleman 8-Person Red Canyon Tent (id: B000W7BHJY)
- Coleman WeatherMaster 6-Person Screened Tent (id: B001TS6WWC)
- Coleman Evanston Screened Tent (id: B004E4AW1K)
- Wenzel Alpine 3 Person Tent (id: B002PAT60S)
- Wenzel Klondike 8 Person Family Tent (id: B002PB2HPS)
- Coleman Hooligan Tent (id: B001TSABLA)
- Coleman Montana 8 Tent (id: B001TSCF96)
- Eureka! Solitaire - Tent (id: B000EQCVNY)

<Clothing-Jeans>

- Levi's Men's 505 Regular Fit Jean (id: B0018OFKJS)
- Levi's Men's Jeans 501 Original Fit (id: B0018OOVPC)
- Levi's Men's 511 Slim Fit Jean (id: B008YNIFJI)

- Wrangler Men's Rugged Wear Relaxed Fit Jean (id: B0007CKMGS)
- Wrangler Men's Cowboy Cut Original Fit Jean (id: B0006U68IC)
- Levi's Men's 501 Shrink To Fit Jean (id: B0018OMPQE)
- Carhartt Men's Washed Duck Work Dungaree Utility Pant (id: B001LRJE7G)
- Lee Men's Relaxed Fit Straight Leg Jean(id: B0008EOQHQ)
- Carhartt Men's Relaxed Straight Denim Five Pocket Jean (id: B004TQHM1G)
- Levi's Men's 514 Straight Jean(id: B003ZJH6T6)

Appendix II. Regular patterns for finding background information of e-book reader reviews

- Returning a product
  - "return(ed|ing)? [a-z]+"
  - "(it|this|)[^,]* going back"
  - "turn(ing|ed) in"
  - r"send [^,]*back"
- Obtaining as a gift
  - "(received|got) .*(gift|present)"
  - "(gift|present) .*for me"
  - "ordered me"
- Giving as a gift
  - "i .*bought .*(gift|present)"
  - "(gift|present) .*for [a-z][^e]"
  - "ordered .*for (my|her|him)"
  - "ordered (her|him)"
- Not understanding a product functionality
  - "i .*can('t|not) figure"
  - "i .*?(n't|not) understand.*how"
  - "(instruction|manual)"
- Using a short period of time

- "(got|received|purchased|bought|arrived|ordered|switched to|came).*(yesterday|today)"
- "(got|received|purchased|bought|arrived|ordered|switched to|came).*(weeks|days) ago"

- Being an avid reader
  - "(avid|heavy) reader"
  - "reads? a ?lot"
- Previous experience
  - "this is (just)? my first kindle"
  - "second kindle"
  - "previous kindle"
  - "(lived | with | purchased | have | loved | got | had | use | used | owned | owning | bought)[^,;:\"\']*?(" + COMPETITORS + ")"
  - COMPETITORS: a list of comparable product names
  - "(lived | with | purchased | have | loved | got | had | use | used | owned | owning | bought)[^,;:\"\']*?(" + OLD_VERSIONS + ")"
  - OLD_VERSIONS: a list of old version names
  - "(lived | with | purchased | have | loved | got | had | use | used | owned | owning | bought)[^,;:\"\']*?(" + SIMILAR_PRODUCTS + ")"
  - SIMILAR_PRODUCTS: a list of similar product names
- Not recommending
  - "(do|would) not (buy|recommend)"

Appendix III. Groups of product features for each product domain

<e-book Reader>

1	back-light | backlight | lighting | bright | brightness | backlit | back-lit | background light | no glare | no glaring | too dim | dim | the light | light setting | lighting setting | light set | white screen | read at night | built-in screen light | light adjustment | unevenness | lighting technology | whiter | the new screen | the screen quality | the day light | uneven | shadow | glare | light condition | read outside | the sun | led light
2	weight | heavy | heavier | lighter | lightweight | light weight | is light | lightness | a feather
4	battery life | charge | recharge | recharged | charged | charging | the battery | battery | energy hog | power | sucks it | sucks power | unnecessary power
5	readability | e-ink | clarity | easy to read | resolution | ease of reading | sharpness | read well | sharper | darker | clearer | crisp | contrast | clear | crystal clear | clearest | readable
6	screen size | small | large | compact | larger | smaller | size screen | wide | the frame | edge | size
7	dictionary | smart lookup | lookup | look up | dictionaries
8	buttons | button
9	wifi | wi-fi | built-in wifi
10	bookmarks
11	memory | memory size | storage | \dgb?

12	customer | service
13	broke
14	freezed | stoped | stopped | software | fast | restart | os | freezes | failed to function | failed to work | locked up | responsive | lag | frozen | slower | screen failure | not working | notworking | did not word | didn't work | doesn't work | the screens | performance | repair | seamless | be reset
15	user-friendly | easy to use | how to operate | hard to use | complexity | so many options| interface
16	automatic adjustment
17	design | looking
18	turn a page | turning the pages | turning a page | page-turning | page turning | page turns | page-back key | flashing | easy to navigate | change pages | change the pages | turns of the page | skip pages | turning the page | swipe per chapter | suddenly advanced | scroll | turn pages | touch-screen | touchscreen | touch screen | touches | fingers | sensitivity | sensitive | touch sensitivity | soft-touch | " touch" screen | finger-brush navigation | page flip | finger | chapter-skip gesture
19	auto wake
20	dead pixel | speck | spot | dust | blotch
21	audio outlets | headphone jack | speaker | music | sound
22	landscape mode
23	playing games
24	vocabulary builder | vocab cards | vocabulary | dictionary builder
25	X-Ray
26	Goodreads
27	organize | organizing | categories | catergory

28 whispersync

29 freetime | free time

30 improvement | software upgrade | upgrade firmware

31 screen defect | pin sized holes | pin hole | pin holes | pinhole | dot

32 airplane

33 eye | eye fatigue | eyes

34 screen savers | screen saver | book cover | screensaver

35 text to audio | tts | text to speech | text-to-speech | audio book

36 hand | handy | hands

37 highlighting | highlight

38 page number

39 typing | editing text

40 margins | margin | the amount of white space

41 ads pop up

42 library compatibility | format | library | e-books | pdf

43 fragile | damage | crack

44 unable to register | register

45 account | syncs | shared | sharing | cloud collection | syncing | collection | share | collections | synchronized

46 tutorial

47 the font size | change the font | enlarge | shrink | change fonts

48 ads | advertising | advertisements | recommendations

49 upload quotes

50 quality control

51 enlarge pictures

52 onscreen keyboard

54 24 clock

<Outdoor Tents>

1 instruction

2 room | space | store | spacious | roomy | storage

3 rain | leak | dry | seal | wet | water | waterproof | drop

4 ventilation | ventilate | circulation | airy | mesh | venting

5 warm

6 heavy | light | weight

7 putting it back | dissemble | dissembly | takedown | take down | folding

8 bag | carry case | case

9 door

10 accessibility

11 stake | peg

12 easy to set up | easy to put up | assemble | assembly

13 fly | rainfly | tarp

14 wind

15 large | small | bigger | big

16 awning | screen

17 holding up | hold up

18 pole

19 pocket

20 zipper | zip

21 price

22 material

23	height
24	tight

<Jeans>

1	straight cut | fit | sit lower | style | sits at waist | cut | inseam
2	rommier | roomy | tight | tighter | snug | uncomforable | comfortable | loose
3	last long | wear out | withstand | durable | long wearing | last
4	fade | dye
5	knees
6	undersize
7	button | button hole
8	belt loop | waist band | loop
9	pocket
10	thin | thinner | elastic | lightweight | light | flimsy | light weight | stiff | tough | strong | armor | thick | heavy
11	cheap
12	material | fabric
13	sizing | size | length | longer | long | smaller | small
14	manufacture | manufacturing | construction | made in | quality | build quality
15	tag
16	crease | wrinkle
17	tag | label
18	stitching
19	weave

20 shrink | shrinking | soak | soaking | shrunk

국문초록

리뷰 정보 유형에 기반한 상품평 유용성 평가

김문형

언어학과

서울대학교 대학원

온라인 상품평의 수가 비약적으로 많은 경우 소비자들이 구매에 도움이 되는 유용한 상품평을 선별하는 것이 어려워 진다. 이를 위해 상품평의 유용성을 자동으로 평가하는 연구가 필요하다. 많은 연구자들이 상품평의 유용성을 평가하기 위한 다양한 방법을 연구해왔지만 그 동안의 연구들은 주로 상품평의 길이, 감정, 가독성 등과 같은 상품평이 어떻게 정보를 전달하는지에 관련된 특성을 이용하여 상품평의 유용성을 평가해왔다. 본 연구는 상품평이 어떤 정보를 전달하는 지를 이용하는 것이 상품평의 유용성을 평가에 더 효과적일 것이라는 가정에서 시작한다.

따라서 이 연구는 상품평의 유용성을 평가하기 위하여 상품평이 어떤 정보를 제공하는 지를 추출하여 이용하는 방법을 연구하는 것을 목표로 한다.

이를 위하여 상품평을 구성하는 각 문장들을 리뷰 정보 유형에 따라 먼저 분류한다. 각 문장이 전달하는 정보의 대상에 따라 그 정보가

구별될 수 있는데, 리뷰어의 개인적인 경험이나 전문성에 관련된 배경 정보(Background Information), 상품 자체의 특성이나 기능에 대한 정보인 핵심 정보(Core Information), 배송이나 AS와 같이 주변 정보(Peripheral Information), 상품의 구매에 대한 마지막 의사 결정이나, 추천 혹은 상품평을 요약하는 종합 정보(Overall Information), 상품과 관련이 없는 비관련 정보(Non-relevant Information)가 이 연구에서 제안하는 리뷰 정보 유형이다.

각 문장이 리뷰 정보 유형에 따라 분류되면, 각 정보 유형에 속한 문장들을 잠재 디리클레 할당(Latent Dirichlet Allocation)을 이용하여 토픽들의 벡터로 변환하고 문장들의 토픽 벡터들 사이에 군집화 모델을 통해 유사한 의미를 갖는 클러스터들을 생성한다. 각 정보 유형마다 생성된 클러스터 모델은 테스트 리뷰의 각 문장이 어떤 정보를 제공하는지 자동으로 추출하는데 사용된다.

상품평 분석을 위해 아마존(amazon.com)에서 전자책, 아웃도어 텐트, 청바지 영역에서 각 도메인 마다 200개의 리뷰를 선정했고, 이를 상품평의 유용성 평가를 위한 학습 데이터로 사용하였다. 이를 위해 각 상품평의 유용성과 상품평의 문장의 정보 유형을 수동으로 주석하였다.

먼저 상품평의 각 문장이 속하는 정보 유형을 얼마나 정확하게 예측할 수 있는지 분류 실험을 진행했다. 이를 위해 문장에 속한 단어와 빈도 정보, 리뷰에서 그 문장이 나타나는 위치 정보, 문장의 주어, 동사,

보조사와 그 품사 정보를 이용하여 상품평에 속한 문장의 정보 유형을 분류하는 실험을 진행했다.

다음으로 이렇게 정보 유형을 나누는 것이 상품평 유용성 평가에 사용될 수 있을 지를 판단하기 위하여 예비 실험을 진행하였다. 이 실험은 배경 정보(Background Information) 만을 이용하여 본 연구에서 제시하는 방식으로 정보를 추출하고 이를 상품평의 유용성 평가에 활용하여 그 결과가 기존 연구들의 연구 결과와 비교하여 동등하게 효과적인 것을 보여주었다.

마지막 실험으로 상품평이 어떤 정보를 제공하는지 모든 정보 유형에 대하여 추출하여 상품평의 유용성의 평가하는 방법과 기존연구에서 상품평이 어떻게 정보를 전달하는지를 이용하여 유용성을 평가하는 방법의 비교를 통해서 본 연구에서 제안하는 방법이 상품평의 유용성을 더 정확히 평가할 수 있음을 증명하였다.

키워드: 리뷰 유용성 평가, 리뷰 정보 유형, 잠재 디리클레 할당, 토픽 기반 방식, 상품평 평가

학번: 2010-30873