Under consideration for publication in Knowledge and Information Systems

Estimating Feature Ratings through an Effective Review Selection Approach

C. Long[1], J. Zhang[2], M. Huang[3], X. Zhu[3,5], M. Li[4], B. Ma[4]

[1]Yahoo! Labs, Beijing, China (This work was done when he was a PhD student in Tsinghua University)
[2]School of Computer Engineering, Nanyang Technological University, Singapore
[3]State Key Laboratory of Intelligent Technology and Systems
Tsinghua National Laboratory for Information Science and Technology
Department of Computer Science and Technology, Tsinghua University, China
[4]School of Computer Science, University of Waterloo, Canada
[5]Corresponding Author: zxy-dcs@tsinghua.edu.cn

Abstract. Most participatory websites collect overall ratings (e.g. five stars) of products from their customers, reflecting the overall assessment of the products. However, it is more useful to present ratings of product features (such as price, battery, screen and lens of digital cameras) to help customers make effective purchase decisions. Unfortunately, only a very few websites have collected feature ratings. In this paper, we propose a novel approach to accurately estimate feature ratings of products. This approach selects user reviews that extensively discuss specific features of the products (called specialized reviews), using information distance of reviews on the features. Experiments on both annotated and real data show that overall ratings of the specialized reviews can be used to represent their feature ratings. The average of these overall ratings can be used by recommender systems to provide feature-specific recommendations that can better help users make purchasing decisions.

Keywords: Data Mining; Text Mining; Kolmogorov Complexity; Information Distance; Feature Rating Estimation

1. Introduction

With the rapid development of Web2.0 and e-commerce that emphasizes the participation of users, websites such as Amazon (www.amazon.com) encourage users to express opinions on products by posting overall ratings and textual

Received Nov 06, 2009
Revised Dec 26, 2011
Accepted Jan 20, 2012

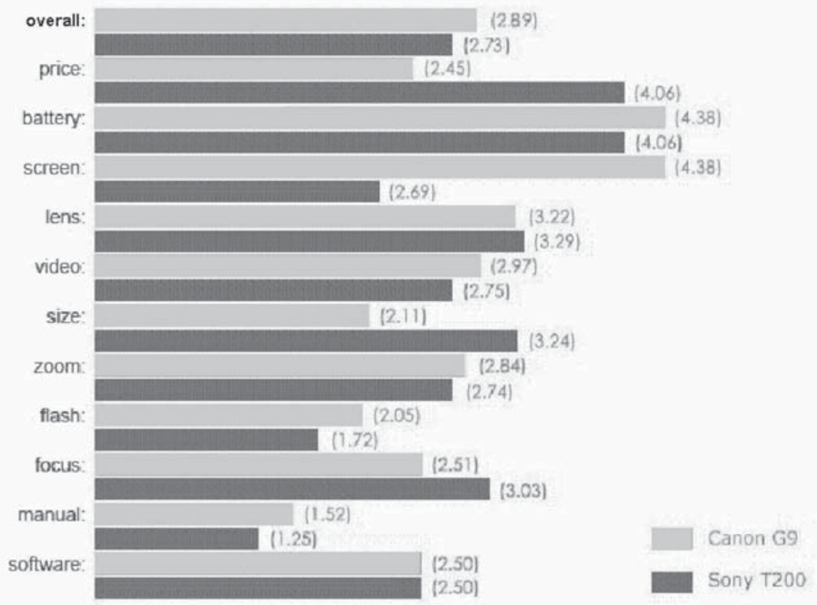


Fig. 1. Overall and Feature Ratings

reviews [1]. These ratings are often used by recommender systems to recommend highly rated products, assisting users in decision making [2]. However, overall ratings of a product tend to be generic and a user may be more interested in some particular feature of the product. Feature ratings, on the other hand, can provide users with an accurate and comprehensive understanding of a product's performance on each feature. Figure 1 shows an example generated from real world data that compares two types of cameras based on the average of overall ratings and 12 important feature ratings. Although these two cameras' overall ratings are close to each other, the differences between many of their feature ratings are remarkable. These detailed feature ratings can then be used by a personalized recommender system to provide feature-specific recommendations. However, most websites do not ask for feature ratings, simply because it may cost users too much effort to provide detailed feature ratings. Even for the websites that do collect feature ratings such as TripAdvisor (www.tripadvisor.com), a large portion (approximately 43%) of users do not provide feature ratings. When feature ratings are not available, we can try to predict them.

Most existing work predicts feature ratings through sentiment orientation classification. For example, [3] is the earliest work of automatic sentiment classification at the document level, using several machine learning approaches with common text features to classify movie reviews. Hu and Liu [4] proposed to use word attributes, including occurrence frequency, part-of-speech, and synset in WordNet. However, these methods make use of only textual reviews for sentiment classification. They generally require a lot of annotation work from experts. They also do not provide a method to obtain a final feature rating that accurately reflects the general opinion of users about the feature.

Distinguished from the methods of semantic orientation classification, we propose a novel approach to estimate feature ratings by making use of users' textual reviews as well as their overall ratings. We utilize the correlation between feature ratings and overall ratings. More specifically, we believe that overall ratings of products would be largely affected by users' feelings about a certain feature if the users extensively discuss this feature in their reviews (called specialized reviews), and their ratings for the feature would be similar to their overall ratings. We select specialized reviews using the information distance measure based on Kolmogorov complexity. Reviews are ranked according to their information distance specialized on one feature. The overall ratings of the ranked top reviews are then used to represent the ratings for that feature.

We carry out experiments to justify our approach on two sets of real data, where one set is collected from a popular travel website TripAdvisor and the second is collected from Amazon's customer reviews on digital cameras. Compared to the method of Talwar et al. [5] and the TF*IDF method, our specialized review selection approach is able to choose specialized reviews more effectively, and requires much less manual work from experts. The method of Talwar et al. needs experts to manually identify approximately 50 words for each core feature. In contrast, we only need to manually choose 4.5 or less core feature words on average, to represent the features whose ratings we would like to estimate. For selected specialized reviews, the corresponding overall ratings are carefully verified to be more similar to their feature ratings. This result becomes the basis of our proposed feature rating estimation approach. Evidence from the dataset also indicates that feature ratings and overall ratings of selected specialized reviews are more similar to each other, respectively. We are then able to use the average of these overall ratings as the estimation of an average feature rating. This average feature rating actually represents an overall opinion of a set of users who care more about the feature and are more likely to be knowledgable (or "experts") on this feature. It can be used by recommender systems to recommend highly rated products based on this feature.

This paper is an extension of our previous work [6]. In comparison to the previous work, we define more clearly the problem of feature estimation, along with a more detailed overview of related studies. More importantly, we extensively evaluate our review selection approach and compare it with the other two methods (the method of Talwar et al. [5] and the TF*IDF method) on the two sets of real data. Some hypotheses we make for this work are additionally verified using the dataset of reviews on digital cameras.

The rest of this paper is organized as follows. We first introduce related studies in Section 2 and define the problem of feature estimation in Section 3. Then, we present the hypotheses that support our proposed approach for feature rating estimation in Section 4. After that, we present our specialized review selection approach in Section 5, along with the part about the information distance theory. We then carry out experiments to verify the hypotheses and evaluate our approach in Section 6. Finally, we conclude the current work and propose some future work in Section 7.

2. Related Work

Our work is aimed at estimating feature ratings of a product based on overall ratings and textual reviews. It is related to the following topics in the review mining area: sentiment classification (overall rating estimation), feature rating estimation, review helpfulness prediction and opinion summarization.

2.1. Sentiment Classification (Overall Rating Estimation)

The task of sentiment classification is to determine the sentiment levels of a textual unit. Usually there are two levels (positive and negative), three levels (positive, neutral and negative), or five levels (strong positive, positive, neutral, negative and strong negative). Most websites use five stars to represent those five sentiment levels.

Researchers mainly focus on two different granularities of a sentiment unit: word level and document level. Most of the early work on sentiment classification used words as the processing unit. For example, the word "good" is positive and "bad" is negative. In 1997, Hatzivassiloglou and McKeown investigated the sentiment orientations of adjectives [7] by utilizing the linguistic constraints on the semantic orientations of adjectives in conjunctions. In 2002, Kamps and Marx proposed a WordNet (http://wordnet.princeton.edu) based approach [8], using distance from a word to good and bad in WordNet as the classification criterion. OPINE system uses relaxation labeling for finding the semantic orientation of words [9].

[3] is the earliest work of automatic sentiment classification at the document level, using several machine learning approaches with common text features to classify movie reviews. Dave et al. [10] designed a classifier based on information retrieval techniques for feature extraction and scoring. Mullen and Collier [11] integrated PMI values, Osgood semantic factors and some syntactic relations into the features of SVM. Pang and Lee [12] proposed another machine learning method based on subjectivity detection and minimum-cut in graph.

The task of overall rating estimation can be viewed as multi-scale document sentiment classification at document level. Pang and Lee [13] extended their document level work in [12] to determine an overall multi-scale rating for a textual review. More specifically, they studied the "rating-inference problem" and proposed an SVM regression model to predict an overall rating to each review which can fit the reviewer's rating as closely as possible.

However, the above work makes use of only textual reviews for sentiment classification. While doing the evaluation, it requires a lot of annotation work from experts. Moreover, given the reviews or sentences which have already been classified as "positive" or "negative", these researchers did not provide a method to obtain a final feature rating that accurately reflects the general opinion of users. Our feature rating estimation method instead makes use of the information of both textual reviews and overall ratings (overall sentiment levels). It is able to produce the general opinion of a special set of expert/knowledgeable users.

2.2. Feature Rating Estimation

Talwar et al. [5] proposed a method to compute the weight of a feature in a review. Firstly, a large number (approximately 50 on average) of words related to a feature are manually selected. Then, a feature's weight in a review is measured by the number of corresponding feature words appeared in this review. Finally, the speciality of a review on one feature depends on its weight. We, however, minimize manual work by automatically generating a collection of associated feature words. Our approach reduces the number of manually selected feature words from approximately 50 to 4. Besides, our method is based on the well established information distance theory instead of the empirical measurements used in [5].

Lu et al. [14] used PLSA (Probabilistic Latent Semantic Analysis) to summarize reviews with comments and ratings on different features. Wang et al. [15] extended the work of Lu et al. using a rating regression approach. However, they did not predict which reviews are useful for readers.

2.3. Review Helpfulness Prediction

Our method tends to select the reviews which are useful for rating estimation, and is thus related to the work of "review helpfulness prediction". Helpfulness prediction is often based on the user behavior study. Kim et al. [16] first used SVM regression to predict other users' evaluation on reviews. Their reviews were collected from Amazon with the pattern like "264 of 322 people found the following review helpful". Liu et al. [17] studied the important factors of helpful reviews and proposed a nonlinear regression model. Their experiments were carried out based on the data collected from the IMDB website (www.imdb.com). Cristian et al. [18] deeply analyzed Amazon data and made some interesting conclusions related to review helpfulness prediction.

Our task is different from the above mentioned work in two aspects. First, their "helpful" reviews do not focus on a certain feature, while our method is feature-specific. Second, their selected helpful reviews are not used for rating estimation. We study user behavior in providing feature ratings through the analysis of the correlations between overall ratings and feature ratings provided by review writers.

2.4. Opinion Summarization

The task of opinion (review) summarization is to generate a concise and comprehensive summary of typical opinions expressed in the textual reviews. The early research on review mining focused on extracting and summarizing sentiment phrases containing feature-opinion pairs such as "good price","the room is very clean", and so on. As the pioneering work, Hu and Liu [4] proposed to use word attributes, including occurrence frequency, part-of-speech, and synset in WordNet. Zhuang et al. [1] focused on movie reviews and introduced a multi-knowledge based approach to integrate WordNet, statistical analysis, and movie knowledge. Liu et al. [19] summarized reviews with low qualities. Li et al. [20] used the random walk method to summarize opinionated answers to a question for question answering. Different from their work, we focus on estimating users' feature ratings instead of a set of summarized sentences or phrases.

3. Problem Definitions

For a set of reviews, we provide five definitions that relate to our problem:

Definition 1. An **overall rating** of a review is a numerical rating that indicates the overall sentiment level of this review. In most websites (like TripAdvisor), an overall rating is an integer ranging from one to five, which means one to five stars.

Definition 2. Features of an entity (e.g. a product, a hotel) broadly mean this entity's aspects that have been commented on. In [4], product features are defined as product attributes and functions that have been commented on in reviews. For example, "price", "screen", "lens", ... , are all important features of a digital camera. Hotel features include "price", "room", "service", and so on.

Definition 3. A **feature rating** of a review on a feature is a numerical measure with respect to this feature, showing the degree of satisfaction demonstrated in the comments toward this feature. Usually, feature ratings keep the same range of overall ratings.

In most situations, people generally have similar criteria about the feature rating of a product. For example, for the "cleanliness" of a hotel room, it is unlikely that a half of people say that it is very clean but another half complain that it is too dirty. However, there are some exceptions because people may have different preferences for a feature. For example, a review says "good color" because its writer prefers red color, and it is possible that this suggestion is not suitable for others who prefer different colors. In order to deal with this problem, user behavior analysis should be performed to classify review writers into different groups of preferences for this feature. In our current work, we consider only features that are more objective, and we leave the research of dealing with subjective features for future work. Based on this assumption, we now can define our feature rating estimation.

Definition 4. Our **feature rating estimation** is defined as follows: given a set of textual reviews and their overall ratings about a product, the task is to estimate a general (i.e. averaged) feature rating that reflects the performance of this product on each feature.

Note that a review is unlikely to cover all the features of a product. It usually focuses only on a small number of features that are most interesting to its review writer. It is obvious that feature ratings can only be estimated from the reviews that have mentioned this feature. Moreover, if some reviews focus on only one feature, the writers of these reviews are likely the experts of this feature; and their opinions are more helpful to other users who are also interested in this feature. We call this type of reviews "specialized reviews", which will be formally defined next.

Definition 5. Specialized reviews are the reviews that extensively discuss only one feature of a product. With this definition, a review is judged to be specialized because of the following two aspects: one is that the review should only focus on one feature, and the other one is that the review should amply discuss this feature.

4. Hypotheses for Feature Rating Estimation

A textual review written by a user for a product normally reveals how much the user cares about certain features of the product. Extensive discussion of a feature reveals that the user cares more about the feature. It is also intuitive that users' overall ratings of a product will be more heavily affected by the features that the users care about the most. We therefore believe that users who extensively discuss a certain feature in their textual reviews are likely to provide feature ratings that are close to overall ratings. These reviews are referred to as specialized reviews on the feature. For example, on the TripAdvisor website, in her review a user strongly criticized rooms of a hotel:

Example 1. "PLEASE DO NOT STAY HERE!!" ... I reserved a non-smoking room, and was placed in a smoking room. The halls, even in non-smoking, stink of stale smoke. There was no hot water(!) in my bathroom. The beds have 1 sheet and no mattress pad. The pillows are stained and have 1 pillow case, no cover. The bathroom has mold all over the tub lining ...

This user gave "1" (in a range of 1-5) to both the feature "Rooms" rating and the overall rating. As in Example 2, the user really liked the services of a hotel, and she gave 5 to both the feature "Service" rating and the overall rating:

Example 2. We had stayed here, ten years ago, and this time enjoyed it just as much. The staff are intelligent and friendly without being obsequious, leaving you alone but being aware if you need them. No request is a surprise and all are happy to help find an answer ...

We formally state our argument in Hypothesis 1:

Hypothesis 1. Compared to the users' ratings for other features, the ratings for a feature provided by these users who extensively discuss this feature in their reviews are more similar to the overall ratings provided by them.

This hypothesis indicates that, for a user who writes a specialized review on a feature, we can use the overall rating to estimate the rating for the feature if absent.

Statistical analysis by Talwar et al. [5] on real data shows that users who extensively discuss a certain feature are more likely to agree on a common rating for that feature. This evidence is formally stated in Hypothesis 2 as follows:

Hypothesis 2. The ratings for a feature that correspond to specialized reviews on this feature are more similar to each other than to the whole collection of ratings for the feature.

Together with Hypothesis 1, Hypothesis 2 can then be extended as follows:

Hypothesis 3. The overall ratings for reviews that are specialized on a feature are more similar to each other than to the entire collection of overall ratings.

Hypothesis 3 indicates that the users who amply discuss a certain feature of a product tend to converge to a common opinion about the product.

When users amply discuss a certain feature of a product, the feature is obviously important to the users. Since people tend to be more knowledgable in the aspects they consider important, these users' ratings will contain a subset of "expert" ratings for the feature. For the specialized reviews written by these users, their feature ratings are more similar to overall ratings (Hypothesis 1). Also, according to Hypotheses 2 and 3, feature ratings and overall ratings for selected specialized reviews are more similar to each other, respectively. The average of these overall ratings should also be closer to the average of these feature ratings. We are then able to use the average of the overall ratings to estimate an average feature rating, representing the overall opinion from more knowledgable users about the feature.

In summary, we have the feature rating estimation approach that works as follows. We select specialized reviews on a feature using the specialized review selection method that will be described in Section 5. The overall ratings for these reviews will be averaged to represent an average feature rating. We verify the three hypotheses and evaluate our approach in Section 6.

5. Specialized Review Selection

In this section, we first give an overview of our system. We then introduce our theory in Section 5.2 for defining the information distance measure. The word extraction method will be discussed in Section 5.3. Finally, the information distance will be approximated and the specialized reviews will be selected according to the formulas presented in Section 5.4.

5.1. The Framework of Our System

Figure 2 shows the framework of our system. Our review corpus is collected from websites. It is used for extracting feature words and expanded words, and for computing the document frequencies of words. The reviews that discuss one type of product and have overall ratings are parsed, and dependent words are extracted according to these extracted feature words and expanded words. Therefore, for each feature, a "related word set" consists of three types of words: feature words, expanded words and dependent words. From these related word sets we can calculate the information distances, based on which specialized reviews are selected. The average feature rating of a product can then be computed based on the overall ratings of its specialized reviews. Next, we describe each step in greater detail.

5.2. Theory

Specialized reviews extensively discuss only one feature, according to Definition 5 in Section 3. Thus, a review's speciality depends on the amount of information discussed on a feature. We use Kolmogorov complexity and information distance to measure the amount of information. Kolmogorov complexity was introduced almost half a century ago by Solomonoff et al. [21]. It is now widely accepted as an information theory for individual objects, parallel to that of Shannon's information theory which is defined on an ensemble of objects.

Fix a universal Turing machine U. The Kolmogorov complexity [21] of a binary string x condition to another binary string y, $K_U(x|y)$, is the length of the shortest (prefix-free) program for U that outputs x with input y. It can be shown that for a different universal Turing machine U', for all x, y

$$K_U(x|y) = K_{U'}(x|y) + C, \tag{1}$$

where the constant C depends only on U'. Thus $K_U(x|y)$ can be simply written as $K(x|y)$. $K(x|\epsilon)$, where ϵ is the empty string, can be written as $K(x)$. For

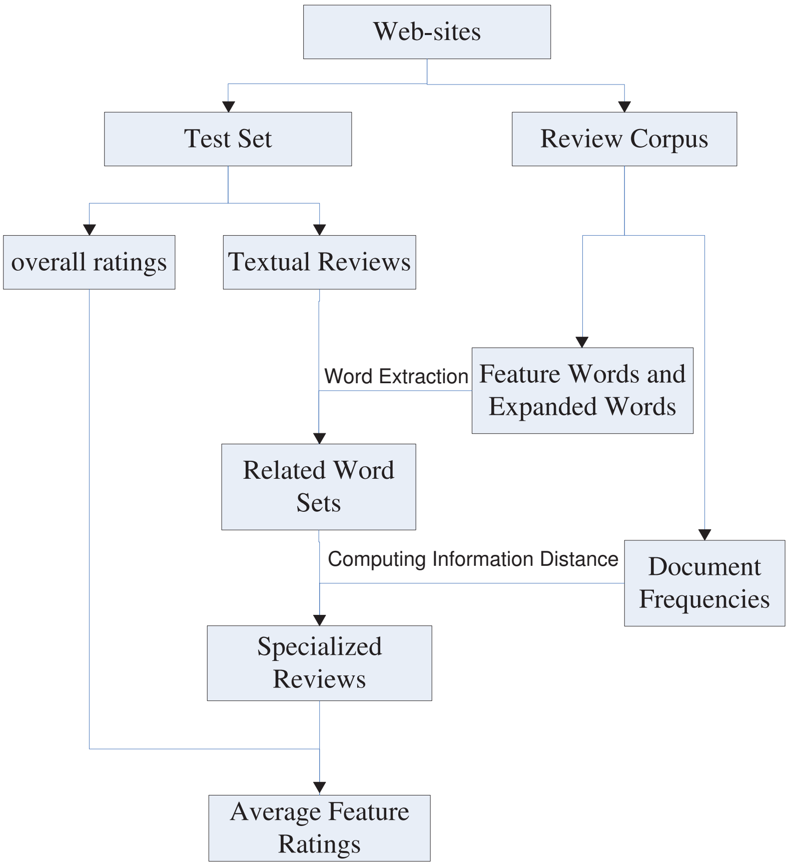


Fig. 2. The Framework of Our System.

example, a string x "111111..." with length N has a relatively small complexity because we can simply print it using a "for" loop. The pseudo-code is as follows:

```
for (i = 1 to N)
  print "1";
```

The length of this program mainly relies on the coding length of number N in the computer, which is $O(\log N)$. However, for a "completely random" string x with length N, we have to print it directly,

```
print x[0];
print x[1];
...
print x[N];
```

The length of this program mainly relies on the length of string x, which is

$O(N)$. For a comprehensive study of Kolmogorov complexity and its applications, see [21].

In the classical Newton's world, "distance" is measured uniquely. This has not been the case for distance in cyber space. A good information distance metric should not only be application independent but also provably better than other "reasonable" definitions. Traditional distances such as the Euclidean distance or the Hamming distance fail for even trivial examples. Tan et al. [22] have demonstrated that none of the 21 metrics used in data mining community is universal, practically. In fact, for any computable distance, they can always find counterexamples.

What would be a good departure point for defining an "information distance" between two objects? To answer this question, in the early 1990's, Bennett el al. [23] have studied the energy cost of conversion between two strings x and y. John von Neumann hypothesized that performing 1 bit of information processing costs $1KT$ of energy, where K is the Boltzmann's constant and T is the room temperature. Observing that reversible computations can be done for free, in the early 1960's Rolf Landauer revised von Neumann's proposal to hold only for irreversible computations. It is proposed in [23] to use the minimum energy needed to convert between x and y to define their distance, as it is an objective measure. Thus, if one wishes to erase string x, she can reversibly convert it to x^*, x's shortest effective description, and then erase x^*. Only the process of erasing $|x^*|$ bits is irreversible computation. Carrying on from this line of thinking, it has been defined in [23] that the energy to convert between x and y is the smallest number of bits needed to convert from x to y and vice versa. That is, with respect to the universal Turing machine U, the cost of conversion between x and y is:

$$E(x,y) = \min\{U(x,p) = y,\ U(y,p) = x\} \tag{2}$$

It is clear that $E(x,y) \leq K(x|y) + K(y|x)$. From this observation, and some other concerns, the sum distance has been defined in [23]:

$$D_{\mathrm{sum}}(x,y) = K(x|y) + K(y|x). \tag{3}$$

However, the following theorem proved in [23] was a surprise:

Theorem 1. $E(x,y) = \max\{K(x|y), K(y|x)\}$.

Thus, the maximum distance was defined in [23]:

$$D_{\max}(x,y) = \max\{K(x|y), K(y|x)\}. \tag{4}$$

This distance is shown to satisfy the basic distance requirements such as positivity, symmetricity and triangle inequality [23]. It was further shown that $D_{\max}$ and D_{sum} minorize (up to constant factors) all other distances that are computable and satisfy some reasonable density condition that within distance k to any string x, there are at most 2^k strings. Formally, if D is a distance,

$$\sum_y 2^{-D(x,y)} \leq 1. \tag{5}$$

$D_{\max}(x,y)$ satisfies the above requirement because of Kraft's Inequality (with the prefix-free version of Kolmogorov complexity). It was proven in [23] that for any computable distance D, there is a constant c, for all x, y,

$$D_{\max}(x,y) \leq D(x,y) + c. \tag{6}$$

Putting it bluntly, if any such distance D discovers some similarity between x and y, so will $D_{\max}$.

Then, after normalization [24, 25],

$$d_{\max}(x,y) = \frac{\max\{K(x|y), K(y|x)\}}{\max\{K(x), K(y)\}}. \tag{7}$$

This theory has been initially applied to alignment free whole genome phylogeny [24], chain letter history [26], question and answering system [27], and many other applications.

Here for an object x, we can measure its information by Kolmogorov complexity $K(x)$; for two objects x and y, their shared information can be measured by information distance $D(x,y)$. In [28], the authors generalize the theory of information distance to more than two objects. Similar to Equation (2), given strings $x_1, \ldots, x_n$, they define the minimum amount of thermodynamic energy needed to convert from any x_i to any x_j as:

$$E_m(x_1, \ldots, x_n) = \min\{|p| : U(x_i, p, j) = x_j\} \tag{8}$$

Then, it is proven in [28] that:

Theorem 2. Modulo to an $O(\log n)$ additive factor,

$$E_m(x_1, \ldots, x_n) \leq \min_i \sum_{k \neq i} D_{\max}(x_i, x_k) \tag{9}$$

Given n objects, this equation may be interpreted as the most specialized object that is similar to all of the others. In the next section, we will use this equation to guide our practical work.

However, one problem is that neither the Kolmogorov complexity $K(\cdot,\cdot)$ nor $D_{max}(\cdot,\cdot)$ is computable. Therefore, we find a way to "approximate" these two measures. The most useful information in a review article is the English words that are related to the features. If we can extract all of these related words from the review articles, the size of the word set can be regarded as an estimation of information content (or Kolmogorov complexity) of the review articles.

Our method for selecting specialized reviews is outlined as follows. First, for each type of product or service (such as a hotel), a small set of core feature words (such as price and room) is generated statistically. Then this set of core feature words is used to generate the expanded words. Thirdly, a parser is used to find the dependent words associated to the occurrences of the core feature words and expanded words in a review. For each review-feature pair, the union of the core feature words, expanded words and dependent words in the review defines the related word set of the review on the feature. Lastly, information distance is used to select the most specialized reviews.

5.3. Word Extraction

5.3.1. Features and Core Feature Words

We have already defined "features" as product features (or attributes) and functions that have been commented on in textual reviews. Feature words are the most direct and frequent words describing a feature, for example, price, room or service of a hotel. Given a feature, the core feature words are the very few most common English words that are used to refer to that feature. For example, both "value" and "price" are used to refer to the same feature of a hotel. In [4], the authors indicate that when customers comment on product features, the words they use converge. If we remove the feature words with frequency lower than 1% of the total frequency of all feature words, the remaining words, which are just core feature words, can still cover more than 90% occurrences. So firstly we extract those words through statistics; then some of those with the same meaning (such as "value" and "price") are grouped into one feature. They are the "core feature words". Note that during the word extraction process, only the grouping process needs little manual work. The other processes of selecting core feature words and extracting expanded words and dependent words are all automatic.

In our data set collected from TripAdvisor website, seven features are provided for rating: value, room, service, cleanliness, location, check/front desk and business service. The first four features: value, room, service and cleanliness are the most frequently rated by review writers. Therefore, our experiments in Section 6 will focus on analyzing the results on these four features. After statistics and grouping, on average, each of the four feature has 4.5 or less core feature words on average(see the Appendix for details). We also list the core feature words for the features of digital cameras.

5.3.2. Expanded Words

Apart from core feature words, many other less-frequently used words that are connected to the feature also contribute to the information content of the feature. For example, "price" is an important feature of a hotel, but the word "price" is usually dropped from a sentence. Instead, words such as "$", "dollars", "USD", and "CAD" are used. We use information distance $d(.,.)$ based on Google to expand feature words. In [29], the Google code of length $G(x)$ represents the shortest expected prefix-code word length of the associated Google event x (i.e. the number of pages returned by querying the word x in Google). $G(x, y)$ represents Google event on both x and y. Then the Google distribution can be used as a compressor for the Google semantics associated with the search terms. Normalized Google distance (NGD) is defined as follows:

$$\begin{aligned} NGD(x,y) &= \frac{G(x,y) - \min(G(x),G(y))}{\max(G(x),G(y))} \\ &= \frac{\max\{\log f(x), \log f(y)\} - \log f(x,y)}{\log N - \min\{\log f(x), \log f(y)\}} \end{aligned} \tag{10}$$

where N is the total number of pages, $f(x)$ denotes the number of pages containing x, and $f(x, y)$ denotes the number of pages containing both x and y, as reported by Google.

In our work, the distance d between words is defined according to NGD based on the words' frequencies and co-occurrence frequencies in the training corpus. Let α be a feature and $\mathcal{A}$ be the set of its core feature words. The distance between a word w and the feature α is then defined to be

$$d(w, \alpha) = \min_{v \in \mathcal{A}} d(w, v) \tag{11}$$

where $d(w, v)$ is approximated by $NGD(w, v)$, and the candidate words (w) are obtained from the words co-existing with the core feature words in the training data. A distance threshold (of 0.01 in our experiments in Section 6) is then used to determine which words should be in the set of expanded words for a given feature. Note that we start from the set of core feature words to find the expanded words. Because the number of core feature words is very small, our system can obtain the expanded words in a very short time.

5.3.3. Dependent Words

If a core feature word or an expanded word is found in a sentence, the words which have grammatical dependent relationship with it are called the dependent words [30]. For example, in sentence “It has a small, but beautiful room”, the words “small” and “beautiful” are both dependent words of the core feature word “room”. All these words also contribute to the reviews and are important to determine the reviewer’s attitude towards a feature. Different from the core feature words and expanded words, dependent words are the ones grammatically dependent with either core feature words or expanded words in reviews.

The Stanford Parser [30] is used to parse each review. For review i and feature j, the core feature words and expanded words in the review are first computed. Then the parsing result is examined to find all the dependent words for the core feature words and expanded words, all of which are called “related words”.

5.4. Computing Information Distance

Let S and T be two sets of words. Then the Kolmogorov complexity can be intuitively estimated by

$$K(S) = \sum_{w \in S} K(w),\ K(S|T) = \sum_{v \in S \setminus T} K(v) \tag{12}$$

Here w and v are the words in S and $S \setminus T$, respectively. Each element (word) of a set contains certain amount of information. We can use frequency count and the Shannon-Fano code (Page 67, Example 1.11.2 in [21]) to encode a phrase which occurs in probability p in approximately $-\log p$ bits to obtain a short description. Therefore, a related word w’s complexity can be estimated by

$$K(w) = -\log P(w|u) = -\log P(w,u) + \log P(u) \tag{13}$$

where w is in feature u’s related word set, $P(w,u)$ can be approximated by the frequency of w in a corpus, and $P(u)$ can be approximated by feature u’s frequency.

Such intuition of estimating $K(S|T)$ can be extended to vectors of sets. For two vectors of sets $S_i = (S_{i1}, S_{i2}, \ldots, S_{in})$, $i \in \{1, 2\}$, define

$$S_1S_2 = S_1 \cup S_2 = (S_{11} \cup S_{21}, \ldots, S_{1n} \cup S_{2n}) \tag{14}$$

$$K(S_i) = \sum_{j=1}^{n} K(S_{ij}),\quad K(S_1|S_2) = \sum_{j=1}^{n} K(S_{1j}|S_{2j}) \tag{15}$$

Then, $D_{max}(S_1, S_2) = \max\{K(S_1|S_2), K(S_2|S_1)\}$. Thus, we are able to use Equation (9) for the selection of specialized reviews.

If there are m reviews $(x_1, x_2, \ldots, x_m)$ and n features $(u_1, u_2, \ldots, u_n)$, the selection of the most specialized reviews needs some minor changes to Equation (9). Without modification, the most specialized review i for a feature j

Table 1. Summary of the Hotel Data Set

Location	# Hotels	# Feedback	# Feedback with feature ratings
Boston	57	3949	2096
Sydney	47	1370	879
Vegas	40	5588	3144

would be such that

$$i = \arg\min_i \sum_{k \neq i} D_{max}(S_{ij}, S_{kj}). \tag{16}$$

However, for specialized review selection, we want that the selected review focuses on the given feature only. The above formula should be modified to

$$i = \arg\min_i \sum_{S_{kj} \neq \emptyset} D_{max}(S_i, S_{kj}). \tag{17}$$

Here

$$D_{max}(S_i, S_{kj}) = D_{max}(S_i, \{\emptyset, \ldots, S_{kj}, \ldots, \emptyset\}) \tag{18}$$

where S_{kj} is in the position of j. More specifically, S_{ij} is changed to S_i to penalize the content of review i that is not related to feature j; and the reviews with an empty word set on feature j are excluded from the selection. In the next section, Equation (17) is used to select specialized reviews.

6. Experimental Results

In this section, we present a set of experimental results to justify our work. First, the performance of our specialized review selection approach is evaluated by manually annotated data, and compared with the method of Talwar et al. [5] and the classic TF*IDF method. Results show that using our approach, a great part of specialized reviews is able to be selected, but requires much less manual work than the competing approaches. Three hypotheses are then verified to support our proposal. Finally, we validate our approach in estimating feature ratings. In general, our approach is able to accurately estimate feature ratings. Rankings of hotels based on estimated feature ratings are only slightly different from those using real feature ratings.

6.1. The Data Set

Our first data set is collected from the popular travel website TripAdvisor. This website indexes hotels from cities across the world. It collects feedback from travelers. Feedback of each traveler consists of an overall rating (from 1, lowest, to 5, highest), a textual review written by the traveler, and numerical ratings for different features of hotels (e.g., value, service, rooms).

We crawled this website to collect travelers' feedback for hotels in three cities: Boston, Sydney and Las Vegas. During this crawling process, we carefully removed information about travelers and hotels to protect their privacy. Their

Table 2. Summary of the Camera Data Set

Type	# Feedback
Canon PowerShot SD600	339
Fujifilm Finepix E550	95
Olympus C-3000	90
Sony DSCF717	77
All	601

Table 3. Consistency of the Three Annotations

Annotation No.	2	3	Averaged
1	96.5%	96.8%	97.6%
2		97.7%	98.4%
3			99.2%

names were replaced by randomly generated unique numbers. For users' feedback, we recorded overall ratings, textual reviews, and numerical ratings for four features: Value/Price (V), Rooms (R), Service (S) and Cleanliness (C). These features are rated by a significant number of users. Table 1 summarizes our data set. For each city, this table contains information about the number of hotels, the total amount of feedback and the amount of feedback with feature ratings. In general, each hotel has sufficient feedback with feature ratings for us to evaluate our work.

The second dataset is collected from Amazon's customer reviews on digital cameras. 138,985 reviews in total with 28 million words are used as the corpus for extracting features and computing information distances between feature words and other words. With these distances, expanded words are extracted [28]. Then 601 reviews about four types of digital cameras are collected for annotation, as shown in Table 2. They have textual reviews and overall ratings but no feature ratings. We have three annotators decide feature ratings by reading textual reviews. First, we select four of the most frequently discussed features: Value (V), Battery (B), Screen/LCD (S) and Lens (L). If a review contains the related words of one of the features, each annotator is then asked to read the textual review and separately give one of the five sentiment levels: 5 (great), 4 (good), 3 (fair), 2 (bad) and 1 (terrible). After that, we have 601 reviews with a set of overall ratings given by review writers, and three groups of feature ratings given by annotators. Let R be the set of all feature ratings, and i and j be two different groups of annotations. $C(i, j)$ is the measurement of the consistency between these two groups. If the bias of one level is allowed between two different annotations, the consistency between annotations i and j can be calculated as follows:

$$C(i, j) = \frac{|R^{'}|}{|R|}, \quad R^{'} = \{r | r \in R, |r^i - r^j| \leq 1\} \tag{19}$$

Here $|R|$ is the number of feature ratings in R. r^i and r^j are the feature ratings given by two annotators i and j on the same feature of the same review. R' contains feature ratings with the biases that are less than or equal to one level between two annotators.

Table 3 shows the consistencies between each two of the three annotations. All

Table 4. The Best Specialized Reviews (Boston)

Top #	**V**	**R**	**S**	**C**
1	SV	SR	SS	SC
2	SV	SR	SS	SC
3	SV	N	SS	SC
4	SV	SR	SS	SC
5	N	SR	N	SC
⋮	⋮	⋮	⋮	⋮

of them are higher than 96%. The ratings of the three groups are also averaged: $r = \frac{r^1+r^2+r^3}{3}$. The consistency between the averaged annotation and any of the three groups is higher than 97.5%. Therefore, the averaged ratings are used as feature ratings in our following experiments. Compared with the first dataset that has reviews of 144 hotels, there are only annotated reviews for four types of digital cameras. But it is still enough for us to evaluate the performance of specialized review selection method and verify Hypothesis 1.

6.2. Specialized Review Selection

We first evaluate the performance of our specialized review selection approach using two sets of manually annotated data. In the first dataset described in Section 6.1, 415 reviews for Boston hotels, 161 for Sydney hotels, and 420 for Las Vegas hotels (996 reviews in total) are selected for manual annotation. Two annotators look over each review and agree on whether the review is specialized or not. The reviews are annotated as "specialized" only when both of these two annotators believe that they are specialized. After that, each review has one of the following five labels: Specialized on feature Value (SV), Specialized on feature Room (SR), Specialized on feature Service (SS), Specialized on feature Cleanliness (SC), and Not Specialized (N).

The reviews for hotels in each city are ranked according to their information distances on each feature. For example, the most specialized review on feature "Value", which has the minimal information distance (see Equation 17) to this feature, is ranked No.1. Table 4 shows the annotated reviews for Boston hotels that are ranked on top five on each feature. Note that the rankings for a review on different features are independent. For example, the review ranked No.1 on feature "Value" may not be ranked No.1 on feature "Service". It can be seen that most of these top reviews are labeled as specialized reviews on respective features. Our specialized review selection approach generally performs well.

To clearly present the performance of our specialized review selection approach, we use the measures of precision, recall and f-score. Precision represents the ratio of correctly classified reviews among all the reviews that are classified as specialized. Recall represents the ratio of correctly classified reviews among all the specialized reviews. The measure f-score is a single value that can represent the result of our evaluation. It is the harmonic mean of precision and recall. These three measures are formally defined as follows. Suppose there are N reviews in total. Let p_{jk} $(1 \leq k \leq N)$ be the review ranked the kth specialized on

Table 5. Evaluation and Comparison of Specialized Review Selection on Hotel Dataset

City		Talwar			TF*IDF			Our Results		
		P	R	F	P	R	F	P	R	F
Boston	V	0.83	0.67	**0.74**	0.57	0.53	**0.55**	0.83	0.67	**0.74**
	R	0.69	0.84	**0.76**	0.64	0.71	**0.67**	0.65	0.74	**0.69**
	S	0.86	0.74	**0.79**	0.74	0.79	**0.76**	0.76	0.84	**0.80**
	C	0.69	0.75	**0.72**	0.50	0.67	**0.57**	0.80	0.67	**0.73**
Sydney	V	0.67	1.00	**0.80**	0.67	1.00	**0.80**	1.00	1.00	**1.00**
	R	0.82	0.73	**0.77**	0.71	0.63	**0.67**	0.74	0.72	**0.73**
	S	0.88	0.75	**0.81**	0.48	0.75	**0.59**	0.68	0.65	**0.67**
Vegas	V	0.59	0.68	**0.63**	0.33	1.00	**0.50**	0.73	0.64	**0.68**
	R	0.75	0.67	**0.71**	0.68	0.66	**0.67**	0.69	0.70	**0.70**
	S	0.80	0.54	**0.64**	0.69	0.71	**0.70**	0.70	0.75	**0.72**
	C	0.41	0.60	**0.49**	0.65	0.65	**0.65**	0.81	0.65	**0.72**

Table 6. Number of Feature Words

Feature	Talwar	TF*IDF	Our Results
V	34	5	5
R	37	7	7
S	59	4	4
C	49	2	2

feature j. Define

$$z_{jk} = \begin{cases} 1 & p_{jk} \text{ labeled specialized on feature } j; \\ 0 & \text{otherwise.} \end{cases} \tag{20}$$

The precision (P), recall (R), and f-score (F) of top k reviews specialized on feature j are formalized as follows:

$$P_{jk} = \frac{\sum_{l=1}^{k} z_{jl}}{k}, \quad R_{jk} = \frac{\sum_{l=1}^{k} z_{jl}}{\sum_{l=1}^{N} z_{jl}}, \quad F_{jk} = \frac{2P_{jk}\ R_{jk}}{P_{jk} + R_{jk}} \tag{21}$$

For each ranked review set on a feature, the maximum f-score and its associated precision and recall are listed in the last three columns of Table 5. It can be seen that for the best f-score, the precision and recall values are mostly larger than 70%, that is, a great part of reviews labeled as specialized receive top rankings by using our specialized review selection. Note that there are no selected reviews specialized on feature "Cleanliness" for the selected hotel reviews in Sydney, so there are no results for this row. Also note that only very few reviews for hotels in Sydney are labeled specialized on the feature "Value" and they are all ranked on the top, therefore the precision, recall and f-score are high as 1.0. Table 6 lists the number of core feature words used by our method.

We compare our approach with the method of Talwar et al. [5] and the TF*IDF method. As mentioned in Section 2, the method of Talwar et al. computes the weight of a feature in a review based on the number of corresponding feature words appeared in this review. The speciality of a review on one feature depends on the weight. The TF*IDF method weights a feature based on the TF*IDF scores of the related words for the feature. Here, a word's "TF" (Term Frequency) is computed from its document. And, its "IDF" (Inverse Document

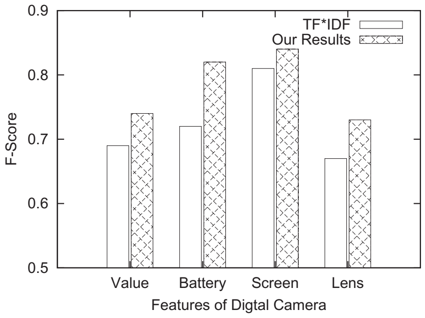


Fig. 3. Evaluation and Comparison of Specialized Review Selection on Camera Dataset

Frequency) comes from the corpus. Although our method also uses the document frequency to approximate the complexity of a word, there are a lot of differences between our method and TF*IDF. The major one is that our method is well supported by the theory and it applies Equation 17 to measure the relationship between different documents within a product review set.

The results of the method of Talwar et al. and the TF*IDF method are listed in Table 5, and the numbers of core feature words used by these two methods are also listed in Table 6. Although the method of Talwar et al. produces generally similar results as our approach, a greater number of manually selected core feature words and weights are required by Talwar et al.'s method. Moreover, due to the bias of manual selection, the method of Talwar et al. performs poorly for the feature "Cleanliness" on the Vegas review set. Our approach produces more stable and reliable results. Our approach also outperforms the TF*IDF method when using the same number of core feature words.

We also evaluate our specialized review selection approach on the camera dataset using the same way as evaluating it on the hotel dataset. The results (f-score) of the TF*IDF method and our method are shown in Figure 3. We can see from this table that our method also outperforms the TF*IDF method on the camera dataset. Note that the method of Talwar et al. requires a great deal of manual work to select feature words and decide their weights. This is the reason why we do not evaluate this method on the camera dataset. Also note that the comparison between our method and the TF*IDF method on both the hotel dataset and the camera dataset produces the similar results. We will see that the verification of Hypothesis 1 on both the two sets of data also gives the similar results. These evidences indicate that the evaluation results on the camera dataset are also generalizable.

6.3. Verification of Hypotheses

In the previous section, we evaluated our review selection method. It performs better than the TF*IDF method. Also, compared to the method of Talwar et al., our approach obtains similar but more stable performance with substantially less feature words. The main purpose of our work is to make use of this method in selecting specialized reviews for feature rating estimation. This idea is supported by the three hypotheses presented in Section 4. Here, we carefully verify these hypotheses.

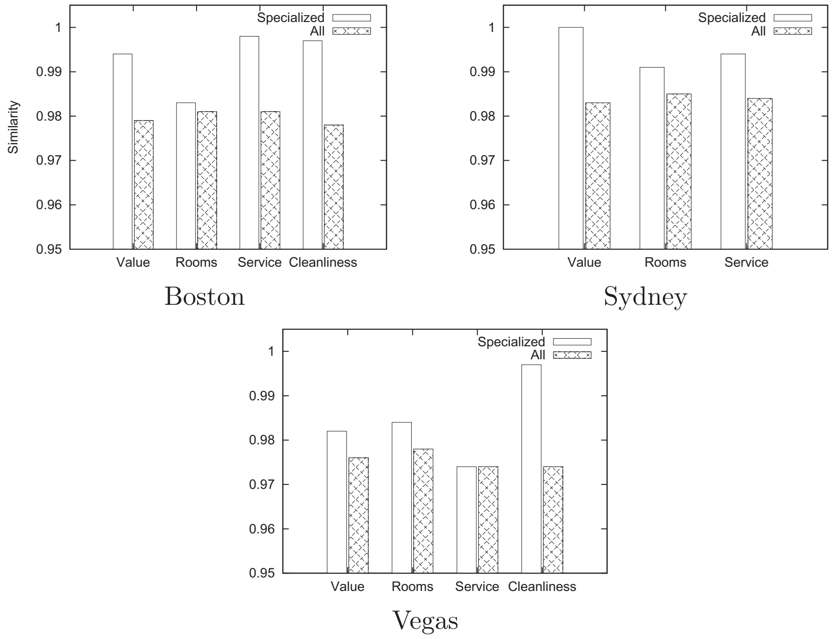


Fig. 4. Similarity between Overall and Feature Ratings on Hotels (Using Manually Labeled Data)

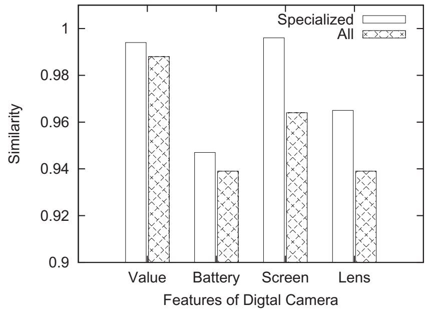


Fig. 5. Similarity between Overall and Feature Ratings on Cameras (Using Manually Labeled Data)

6.3.1. Hypothesis 1

Hypothesis 1. The feature ratings provided by reviewers who provide specialized reviews on this feature are more similar to their corresponding overall ratings.

We first verify Hypothesis 1 using the manually annotated data. As mentioned in Section 6.2, we manually labeled for randomly selected reviews as specialized or not on features. For specialized reviews on each feature, the similarity between their feature ratings and overall ratings is calculated and compared with the similarity between feature and overall ratings for all reviews. More formally, if there are m reviews labeled as specialized on a feature, their feature ratings are formed as a vector X and the overall ratings are formed as Y. Cosine similarity [31] is used to measure the similarity between them as follows:

$$Sim(X,Y) = \frac{X \bullet Y}{\|X\| \, \|Y\|}$$

The similarity between overall ratings and feature ratings of all reviews for hotels in each city is also calculated. Figure 4 compares these similarities for each feature of hotels in each city (Boston, Sydney and Vegas). It is clear that the similarities between overall ratings and feature ratings for specialized reviews are higher than those for all reviews. We do not have results for the feature "Cleanliness" of the hotels in Sydney because there are no reviews annotated as specialized on this feature among those randomly selected reviews for hotels in this city. The similarities on the camera dataset are shown in Figure 5. From this figure, we have the same conclusion.

Then we verify Hypothesis 1 using specialized reviews selected by our review selection approach. All reviews are ranked based on each feature for hotels in each city. Because of the space limitation we show here only the results for the city of Boston in Figure 6. Each diagram in this figure shows the similarities between overall and feature ratings for top 5, top 10, top 15, ..., top 995 and top 1000 reviews selected by our review selection method. From this figure, we can see that the overall and feature ratings of more specialized reviews tend to have higher similarities. Therefore, overall ratings of the most specialized reviews can be used to represent their feature ratings.

6.3.2. Hypothesis 2

Hypothesis 2. The feature ratings corresponding to specialized reviews are more similar to each other than to the overall collection of feature ratings.

In this experiment, we use our review selection approach to select specialized reviews to verify Hypothesis 2. More specifically, for each city, hotels that receive at least 10 reviews with feature ratings are selected. We use our specialized review selection approach to select top 20% and 50% specialized reviews on each feature for hotels in each city. We calculate the standard deviation of their feature ratings, as well as that of all feature ratings, for each hotel in a city. We then average these standard deviations over all the hotels in the same city. The average values are listed in Table 7. The feature ratings of specialized reviews have smaller average standard deviations. Standard T-test is used to measure the significance of the results between top 20% specialized reviews and all reviews, city by city and feature by feature. Their p-values are shown in the braces, and they are significant at the level of 0.05. Although for some items there does not seem to be a significant difference, the results are significant for the entire data set. Therefore, when these travelers write reviews that focus on one feature, their feature ratings tend to converge. These feature ratings can be averaged to represent a specific opinion of these travelers on that feature.

6.3.3. Hypothesis 3

Hypothesis 3. The overall ratings corresponding to specialized reviews on a feature are more similar to each other than to the collection of entire overall ratings.

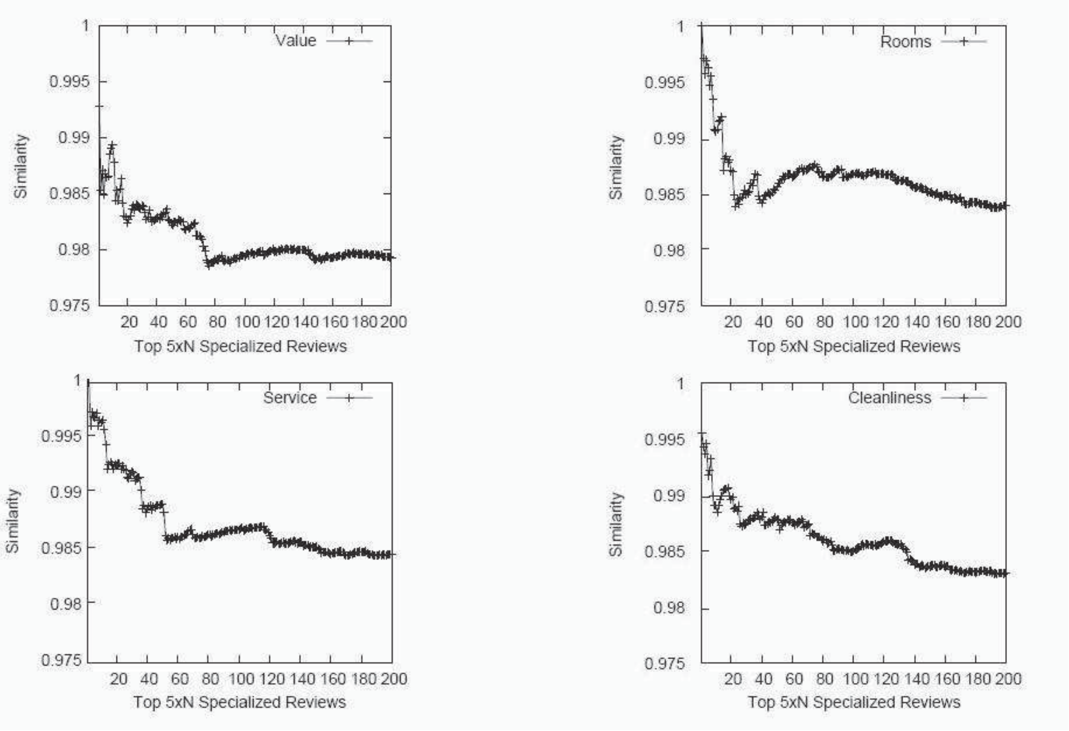


Fig. 6. Similarity between Overall and Feature Ratings (Specialized Review Selection Method)

Table 7. Deviation of Feature Ratings

City	#Hotel	Feature	20%	50%	All
Boston	52	V	0.921 (0.0003)	1.040	1.136
		R	0.953 (0.1851)	1.028	1.013
		S	0.936 (0.0057)	1.070	1.144
		C	0.756 (0.0009)	0.866	0.949
Sydney	34	V	0.925 (0.0670)	0.991	1.054
		R	0.677 (0.0030)	0.879	0.945
		S	0.815 (0.0023)	1.006	1.115
		C	0.660 (0.0056)	0.737	0.907
Vegas	33	V	1.116 (0.0324)	1.210	1.291
		R	0.885 (0.0058)	1.161	1.175
		S	1.165 (0.1130)	1.246	1.269
		C	1.056 (0.1477)	1.058	1.158

Similar to our verification of Hypothesis 2, we verify Hypothesis 3 by first selecting the top 20% and 50% specialized reviews on each feature. We then calculate the average standard deviations of the overall ratings for the specialized reviews, as well as those for all reviews, as listed in Table 8. The p-values are also calculated and shown in the braces. It is clear that, for any feature, overall ratings for specialized reviews have smaller standard deviation. Travelers who wrote these reviews tend to agree on their overall ratings for hotels.

Table 8. Deviation of Overall Ratings

City		V	R	S	C
Boston	20 %	0.882 (0.0005)	0.982 (0.011)	0.930 (0.0026)	0.866 (0.0002)
	50%	1.016	1.123	1.061	1.005
	All	1.128			
Sydney	20 %	0.874 (0.050)	0.726 (0.0015)	0.751 (0.0024)	0.831 (0.017)
	50%	0.947	0.963	0.953	0.930
	All	1.058			
Vegas	20 %	1.075 (0.011)	0.928 (0.0005)	1.185 (0.111)	1.103 (0.026)
	50%	1.237	1.264	1.258	1.224
	All	1.299			

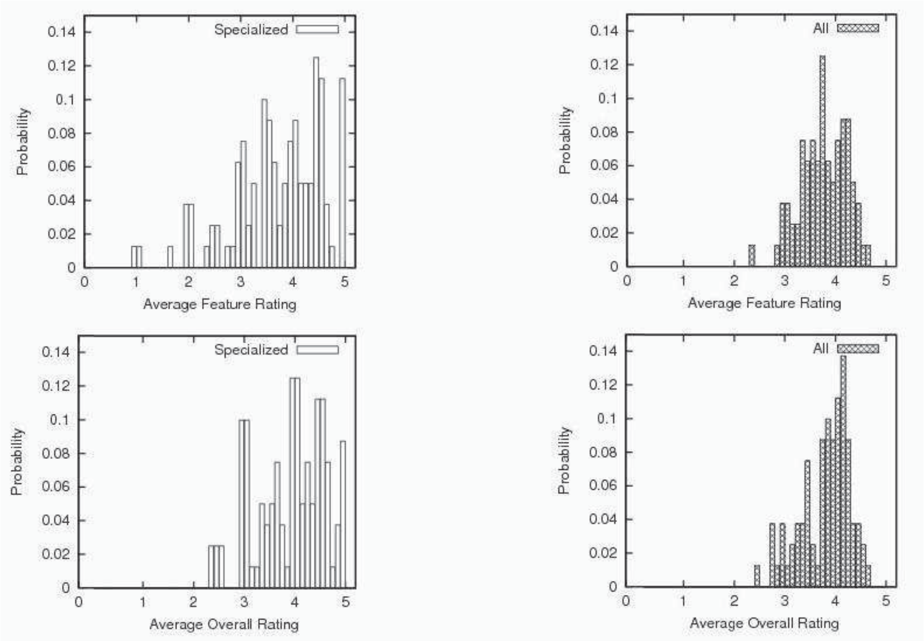


Fig. 7. Comparison between Average Rating Distribution on Specialized Reviews and that on All Reviews

6.4. Estimating Average Feature Rating

Supported by the three hypotheses verified in the previous section, we can then use the average of overall ratings for specialized reviews on a feature to estimate an average rating for this feature. This single rating of the feature can represent a general opinion of users who are more knowledgable about this feature. We use an average of the feature ratings for top 20% specialized reviews to reflect a general opinion of knowledgeable/expert users. In other words, the average feature rating will be estimated using the overall ratings for the top 20% specialized reviews.

Table 9. Result of Estimating Average Feature Rating

Feature	V	R	S	C	AVG
$\|F_{20} - O_{20}\|$	0.316	0.191	0.231	0.300	0.260
$\|Fr_{20} - Or_{20}\|$	0.396	0.284	0.300	0.389	0.342
$\|F_{20} - F_{100}\|$	0.458	0.429	0.429	0.348	0.416
$\|F_{20} - Or_{20}\|$	0.654	0.635	0.624	0.625	0.634
$\|F_{20} - O_{100}\|$	0.455	0.519	0.430	0.522	0.481

Note that if we use a large number of specialized reviews, the error of estimating feature ratings from the overall ratings for these reviews may be large. On another hand, if we use a small number of specialized reviews, the average feature rating estimated based on the overall ratings for these reviews may not be representative. Also note that 20% of specialized reviews may not be the best choice among all possible percentages. However, finding the best percentage is not an easy task due to the following reasons: 1) The numbers of specialized reviews on different features may vary from each other. Most users tend to focus on a small range of important features. So these important features will have a larger number of specialized reviews; 2) The numbers of reviews for different products are normally different. If a product has only a few reviews, a small percentage of specialized reviews could not be representative; 3) Different products have different types of customers, and customers have their own writing styles. The percentage of specialized reviews may be affected by the writing styles of consumers. In this paper, we choose the three percentages (20%, 50% and 100%) that provide the clear trend where 20% is better than 50%, and 50% is better than 100%. The exploration of the best percentage is left for future work.

In the following, we will first provide statistical analysis to show the rationale of ranking hotels using an average of feature/overall ratings for specialized reviews. We then directly evaluate the performance of this process. Given a set of reviews for one hotel, let F_{20} be the mean of the feature ratings for top 20% specialized reviews, and O_{20} be the mean of the overall ratings for these reviews. Similarly, the mean values of feature and overall ratings for all reviews are referred to as F_{100} and O_{100} respectively. We plot in Figure 7 the distribution of these average feature ratings (F_{20} and F_{100} in the two left most figures) and average overall ratings (O_{20} and O_{100} in the two right most figures) of all hotels. We can see that the average values of feature or overall ratings for all reviews are mostly located around the rating 4. Ranking hotels based on these average ratings is not easily distinguishable. This may be caused by several reasons. The most obvious one is noise [32] (for example, caused by users' subjectivity) in the data that reduces rating difference among hotels. Comparably, the average of ratings for specialized reviews is spread out because more knowledgable opinions tend to contain less noise. It is thus better to use the average of these ratings to rank hotels and to provide feature-specific recommendations.

We evaluate the performance of using overall ratings for specialized reviews to estimate an average feature rating. In this experiment, we randomly choose 20% reviews from the set of reviews for each hotel. We repeat this process for five times. In each time, we calculate the mean of feature and overall ratings. We then obtain the average of the mean values, referred to as Fr_{20} and Or_{20} respectively.

For all hotels that receive no less than ten reviews with feature ratings, we

Table 10. Error Range of Hotel Ranking

Top	Range	V	R	S	C	AVG
O_{20}	Max	0.157	0.105	0.118	0.192	0.131
	Min	0.084	0.044	0.050	0.098	0.063
	AVG	0.120	0.074	0.084	0.145	0.097
O_{100}	Max	0.245	0.203	0.191	0.310	0.237
	Min	0.126	0.117	0.099	0.152	0.124
	AVG	0.186	0.160	0.145	0.231	0.181

calculate the average absolute differences between F_{20} and O_{20}, written as $|F_{20} - O_{20}|$ in the second row of Table 9. Compared with the average absolute difference between Fr_{20} and Or_{20} (written as $|Fr_{20} - Or_{20}|$ in the third row), $|F_{20} - O_{20}|$ is 23.98% lower. This result directly confirms our idea of using overall ratings for specialized reviews to estimate an average feature rating, resulting from the three verified hypotheses.

Second, we calculate the average difference between F_{20} and F_{100} ($|F_{20} - F_{100}|$) for all these hotels. The result shows that the average of feature ratings for specialized reviews is different from the average of all feature ratings, that is, the general opinion of all users is not the same as that of more knowledgable users. This inequality indicates that we cannot use the former to replace the later. Finally, the last two rows, $|F_{20} - Or_{20}|$ and $|F_{20} - O_{100}|$, show that neither Or_{20} nor O_{100} can be used to estimate F_{20}.

To evaluate the performance of our approach, we also show how much the estimation error $|F_{20} - O_{20}|$ will affect the result of ranking hotels based on their average feature ratings. This directly reflects how well our approach can assist a feature-specific recommender system in recommending hotels. We present both maximum and minimum errors in ranking hotels. We also compare the result with that of $|F_{20} - O_{100}|$ (the baseline) which uses the average of all reviews' overall ratings.

Suppose that m hotels in one city are ranked according to their F_{20}, each of which is written as a_i ($1 \leq i \leq m$). Let $g(a_i)$ be the ranking of an average feature rating a_i, and $e = |F_{20} - O_{20}|$ (or $e = |F_{100} - O_{100}|$). The maximum ranking error is formalized as follows:

$$\overline{rd} = \frac{1}{m^2} \sum_{i=1}^{m} \max(|g(a_i + e) - g(a_i)|, |g(a_i - e) - g(a_i)|) \tag{22}$$

The minimum ranking error is defined as follows:

$$\underline{rd} = \frac{1}{m^2} \sum_{i=1}^{m} \min(|g(a_i + e) - g(a_i)|, |g(a_i - e) - g(a_i)|) \tag{23}$$

The average ranking error is the mean of the maximum ranking error $\overline{rd}$ and the minimum ranking error $\underline{rd}$. The results are summarized in Table 10. From this table we can see that the average ranking error range for O_{20} is 9.7%, which is much lower than that for the baseline (18.1%). The error range of O_{20} is within 3-5 hotel ranks for a city (while x-x hotel ranks for O_{100}), according to the number of hotels in each city listed in Table 7. Our feature rating estimation method provides sufficiently good performance.

7. Conclusion and Future Work

We developed a novel approach to accurately estimate feature ratings of products, taking advantage of specialized reviews extracted by a review selection method that is based on the information distance theory. Our review selection method has been extensively evaluated and compared with the other two methods on real data collected from TripAdvisor and Amazon. Our approach performs better than the TF*IDF method. It is also comparable to the method of Talwar et al. [5], but requires much less manual work from experts. Three hypotheses have also been verified through experiments. Based on these hypotheses, a general feature rating can be estimated for a feature of a product from the overall ratings of specialized reviews for this product. This general feature rating can then be used by recommender systems to provide feature-specific recommendations. Our work is therefore a novel first attempt to accurately estimate feature ratings using users' overall ratings and their textual reviews without requiring much manual annotation.

Our approach works well when a sufficient number of specialized reviews exist for each product. This might not be the case for some products, which may not have many reviews. Or, they may have features that are not easily distinguishable. For example, the hotel features of "Rooms" and "Cleanliness" are often both mentioned in many reviews because one important aspect of rooms is their cleanliness. For future work, we may make use of reviews that are not specialized. We plan to use syntax and semantic information and machine learning methods to train the frequently used words and phrases in specialized reviews for describing a feature. These words and phrases will be used for estimating feature ratings of other reviews which are not specialized.

After having sufficient feature ratings, we plan to develop a recommender system that makes use of these ratings to provide feature specific recommendations. Users often have different preferences [33]. They may prefer one feature of a product over another feature. Our system will therefore be aimed at providing personalized feature-specific recommendations, by taking into account users' frequent behaviors [34] on different items [35], in order to better help them make effective purchasing decisions.

Acknowledgment

The work was supported by NSFC under grant No.60973104 and the National Basic Research Program (973 project in China) under grant No.2012CB316301 The work was also supported by IRCI project (104519-006) from the International Development Research Center, Canada.

References

[1] Zhuang, L., Jing, F., Zhu, X.: Movie review mining and summarization. In: ACM 17th Conference on Information and Knowledge Management (CIKM). (2006) 43–50

[2] Schafer, J.B., Konstan, J., Riedi, J.: Recommender systems in e-commerce. In: 1st ACM Conference on Electronic Commerce (EC). (1999) 158–166

[3] Pang, B., Lee, L., Vaithyanathan, S.: Thumbs up? sentiment classification using machine learning techniques. In: Conference on Empirical Methods in Natural Language Processing (EMNLP). (2002) 79–86

[4] Hu, M., Liu, B.: Mining and summarizing customer reviews. In: 10th ACM International Conference on Knowledge Discovery and Data Mining (KDD). (2004) 168–177

[5] Talwar, A., Jurca, R., Faltings, B.: Understanding user behavior in online feedback reporting. In: 8th ACM Conference on Electronic Commerce (EC). (2007) 134–142

[6] Long, C., Zhang, J., Huang, M., Zhu, X., Li, M., Ma, B.: Specialized review selection for feature rating estimation. In: Proceedings of the IEEE/WIC/ACM International Conference on Web Intelligence (WI). (2009)

[7] Hatzivassiloglou, V., McKeown, K.R.: Predicting the semantic orientation of adjectives. In: Annual Meeting of the Association of Computational Linguistics (ACL). (1997) 174–181

[8] Kamps, J., Marx, M.: Words with attitude. In: the First International Conference on Global WordNet. (2002) 174–181

[9] Popescu, A.M., Etzioni, O.: Extracting product features and opinions from reviews. In: Conference on Empirical Methods in Natural Language Processing (EMNLP). (2005) 339–346

[10] Dave, K., Lawrence, S., Pennock, D.M.: Mining the peanut gallery: Opinion extraction and semantic classification of product reviews. In: International World Wide Web Conference (WWW). (2005) 519–528

[11] Mullen, T., Collier, N.: Sentiment analysis using support vector machines with diverse information sources. In: Conference on Empirical Methods in Natural Language Processing (EMNLP). (2004) 412–418

[12] Pang, B., Lee, L.: A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In: Annual Meeting of the Association of Computational Linguistics (ACL). (2004) 271–278

[13] Pang, B., Lee, L.: Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. In: Annual Meeting of the Association of Computational Linguistics (ACL). (2005) 115–124

[14] Lu, Y., Zhai, C., Sundaresan, N.: Movie review mining and summarization. In: International World Wide Web Conference (WWW). (2009) 131–140

[15] Wang, H., Lu, Y., Zhai, C.: Latent aspect rating analysis on review text data: A rating regression approach. In: ACM International Conference on Knowledge Discovery and Data Mining (KDD). (2010) 783–792

[16] Kim, S.M., Pantel, P., Chklovski, T., Pennacchiotti, M.: Automatically assessing review helpfulness. In: Conference on Empirical Methods in Natural Language Processing (EMNLP). (2006) 423–430

[17] Liu, Y., Huang, X., An, A., Yu, X.: Modeling and predicting the helpfulness of online reviews. In: IEEE International Conference on Data Mining (ICDM). (2008) 443–452

[18] Danescu-Niculescu-Mizil, C., Kossinets, G., Kleinberg, J.: How opinions are received by online communities: A case study on amazon.com helpfulness votes. In: International World Wide Web Conference (WWW). (2009) 141–150

[19] Liu, J., Cao, Y., Lin, C.Y., Huang, Y., Zhou, M.: Low-quality product review detection in opinion summarization. In: Conference on Empirical Methods in Natural Language Processing (EMNLP). (2006) 423–430

[20] Li, F., Tang, Y., Huang, M., Zhu, X.: Answering opinion questions with random walks on graphs. In: Annual Meeting of the Association of Computational Linguistics (ACL). (2004) 737–745

[21] Li, M., Vitányi, P.: An Introduction to Kolmogorov Complexity and its Applications (2nd Edition). Springer-Verlag (1997)

[22] Tan, P., Kumar, V., Srivastava, J.: Selecting the right interestingness measure for association patterns. In: The 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. (2002) 32–44

[23] Bennett, C., Gacs, P., Li, M., Vitányi, P., Zurek, W.: Information distance. IEEE Transactions on Information Theory **44**(4) (1998) 1407–1423

[24] Li, M., Badger, J., Chen, X., Kwong, S., Kearney, P., Zhang, H.: An information-based sequence distance and its application to whole mitochondrial genome phylogeny. Bioinformatics **17**(2) (2001) 149–154

[25] Li, M., Chen, X., Li, X., Ma, B., Vitányi, P.: The similarity metric. IEEE Transactions on Information Theory **50**(12) (2004) 3250–3264

[26] Bennett, C., Li, M., Ma, B.: Chain letters and evolutionary histories. Scientific American **288**(6) (2003) 76–81

[27] Zhang, X., Hao, Y., Zhu, X., Li, M.: Information distance from a question to an answer.

In: The 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. (2007)

[28]Long, C., Zhu, X., Li, M., Ma, B.: Information shared by many objects. In: ACM 17th Conference on Information and Knowledge Management (CIKM). (2008)

[29]Cilibrasi, R.L., Vitányi, P.M.: The google similarity distance. IEEE Transactions on Knowledge and Data Engineering **19**(3) (2007) 370–383

[30]Marneffe, M.C., MacCartney, B., Manning, C.D.: Generating typed dependency parses from phrase structure parses. In: The fifth international conference on Language Resources and Evaluation (LREC). (2006)

[31]Lewis, J., Ossowski, S., Hicks, J., Errami, M., Garner, H.R.: Text similarity: an alternative way to search medline. Bioinformatics **22**(18) (2006) 2298–2304

[32]Jia, Y., Zhang, J., Huan, J.: An efficient graph-mining method for complicated and noisy data with real-world applications. Knowledge and Information Systems (KAIS) **26**(2) (2011)

[33]Eirinaki, M., Vazirgiannis, M.: Web mining for web personalization. ACM Transactions on Internet Technology (TOIT) **3**(1) (2003) 1–27

[34]Saleh, B., Masseglia, F.: Discovering frequent behaviors: time is an essential element of the context. Knowledge and Information Systems (KAIS) **25**(2) (2010)

[35]Becchetti, L., Colesanti, U.M., Marchetti-Spaccamela, A., Vitaletti, A.: Recommending items in pervasive scenarios: models and experimental analysis. Knowledge and Information Systems (KAIS) **24**(3) (2010)

Appendix: Lists of Core Feature Words

Here we list the core feature words used in our approach.

1. Hotel's core feature words:

The core feature words for feature "Value": value, price, cheap, expensive, cost;

The core feature words for feature "Room": room, air, atmosphere, bathroom, bed, bedroom, space;

The core feature words for feature "Service": service, custom, staff, food;

The core feature words for feature "Cleanliness": clean, dirty.

2. Digital camera's core feature words:

The core feature words for feature "Value": value, price;

The core feature words for feature "Battery": battery;

The core feature words for feature "Screen": screen, LCD;

The core feature words for feature "Lens": lens.

Author Biographies

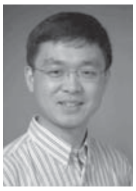
Chong Long received his B.E. and PhD degree from Tsinghua University, China in 2005 and 2010, respectively. He is now a research engineer in Yahoo! Labs, Beijing. His research interests include natural language processing, review mining, language model, text summarization and information extraction. He has published papers on CIKM, IEEE ICDM, COLING, Web Intelligence, etc. This paper was finished when he was a PhD student in Tsinghua University.

Jie Zhang received the Ph.D. degree from the University of Waterloo, Waterloo, Canada, in 2009, and was the recipient of the Alumni Gold Medal, awarded to honour the top PhD graduate. He is currently an Assistant Professor at the School of Computer Engineering, Nanyang Technological University, Singapore. His research interests include artificial intelligence and multiagent systems, trust modeling and incentive mechanisms, and machine learning and data mining. His papers have been published by top journals (i.e. Computational Intelligence) and conferences (i.e. AAAI, AAMAS, WI and UMAP). He has also won 4 best paper awards at CNSM'10, IM'09, ITCS'10 and CSWWS'06. Jie Zhang is currently also active in serving research communities as co-chair for TRUM'11, sponsorship chair for PRIMA'12, publicity chair for SASO'12, publication and publicity chair for PST'10, associate editor for ICIS'10, PC members for UMAP'12, AAMAS'11, AAAI'10, and reviewers for JAAMAS, Computational Intelligence, etc.

Minlie Huang now is a faculty member of Dept. Computer Science and Technology, Tsinghua University. He received his PhD degree in 2006 in Tsinghua University. His research interest includes natural language processing, text mining, Opinion mining and sentiment analysis, and question answering. He has published papers on IJCAI, AAAI, ACL, COLING, NA-ACL, IEEE ICDM, etc. and served as PC members for major conferences including ACL, EACL, NA-ACL, EMNLP, and so on.

Xiaoyan Zhu, professor,Deputy Head of state key lab of intelligent technology and systems, Tsinghua University. She got bachelor degree at University of Science and Technology Beijing in 1982, master degree at Kobe University in 1987. and Ph. D. degree at Nagoya Institute of Technology, Japan in 1990. She is International Research Chair holder of IDRC, Canada, from 2009. She is teaching at Tsinghua University since 1993. Her research interests include intelligent information processing, machine learning, natural language processing, quanry and answering system and Bioinformatics. She has authored more than 100 peer-reviewed articles in leading international conferences (SIGKDD, IJCAI, AAAI, ICDM, PAKDD, CIKM, APBC) and journals (Int. J. Medical Informatics, Bioinformatics, BMC Bioinformatics, Genome Biology and IEEE Trans. on SMC).

Ming Li is a Canada Research Chair in Bioinformatics and a University Professor of the University of Waterloo. He is a fellow of Royal Society of Canada, ACM, and IEEE. He is a recipient of E.W.R. Steacie Fellowship Award in 1996, and the 2001 Killam Fellowship. Together with Paul Vitanyi they have pioneered the applications of Kolmogorov complexity and co-authored the book "An introduction to Kolmogorov complexity and its applications". His research interests recently include protein structure determination and next generation internet search engine.



Bin Ma is a Professor and University Research Chair in the David R. Cheriton School of Computer Science at the University of Waterloo. He received his Ph.D. degree from Beijing University in 1999. During 2000-2008 he worked at University of Western Ontario as Assistant Professor, Associate Professor, and Canada Research Chair. He published over 100 research papers, and is the creator of several popular bioinformatics software packages including PEAKS and PatternHunter.

Correspondence and offprint requests to: Xiaoyan Zhu, Department of Computer Science and Technology, Tsinghua University, Beijing, 100084, China. Email: zxy-dcs@tsinghua.edu.cn