

Article

An Extended HITS Algorithm on Bipartite Network for Features Extraction of Online Customer Reviews

Chen Liu ¹, Li Tang ¹ and Wei Shan ^{2,*}

¹ Business School, University of Shanghai for Science and Technology, Shanghai 200093, China; liuchen.chn@hotmail.com (C.L.); tangli13847346936@163.com (L.T.)

² School of Economics and Management, Beihang University, Beijing 100191, China

* Correspondence: shanwei@buaa.edu.cn

Received: 8 April 2018; Accepted: 27 April 2018; Published: 4 May 2018



Abstract: How to acquire useful information intelligently in the age of information explosion has become an important issue. In this context, sentiment analysis emerges with the growth of the need of information extraction. One of the most important tasks of sentiment analysis is feature extraction of entities in consumer reviews. This paper first constitutes a directed bipartite feature-sentiment relation network with a set of candidate features-sentiment pairs that is extracted by dependency syntax analysis from consumer reviews. Then, a novel method called MHITS which combines PMI with weighted HITS algorithm is proposed to rank these candidate product features to find out real product features. Empirical experiments indicate the effectiveness of our approach across different kinds and various data sizes of product. In addition, the effect of the proposed algorithm is not the same for the corpus with different proportions of the word pair that includes the “bad”, “good”, “poor”, “pretty good”, “not bad” these general collocation words.

Keywords: opinion mining; feature extraction; bipartite network; extended HITS algorithm

1. Introduction

With the growth of e-commerce, online shopping has become emerged as more convenient. At the same time, online shopping has gradually replaced the conventional brick-and-mortar store stores, become an indispensable shopping means of individuals’ mundane lives. In order to enhance the customer satisfaction associated with the shopping experience, almost every e-commerce transaction platform has designed the function of user evaluation. These reviews are termed as quite critical for the consumers, businesses, and manufacturers [1], since they not only impact the consumers’ shopping decisions [2,3] or word-of-mouth intention [4] and merchants’ purchasing strategies but also impact the design and improvement of the products [1,5]. Therefore, consumer reviews are a substantial information resource [4–6]. Nonetheless, there are three major significant barriers when it comes to fully exploiting the product reviews. One of barriers possibly encountered is that the enormous quantity of user reviews results in an overload of information. Secondly, the consumer reviews that contain abundant user behavior information, for instance, user’s opinions, attitudes [7] and preferences [8], are hardly useable. Furthermore, comments are also subjective, since the user’s evaluation of the current product will be impacted by the previous purchasing experience. For the above reasons, it is deemed as quite hard for the consumers to fetch the satisfactory and essential information from the product reviews [3]. Therefore, sentiment analysis becomes a serviceable tool to address this kind of information extraction issue.

Sentiment analysis, also called opinion mining, is a subfield of natural language processing [2]. Generally, there are three levels of sentiment analysis: document-level, sentence-level, and feature-level [9]. Among them, feature-level sentiment analysis draws a lot of attention,

which refers to opinions or sentiments expressed on dissimilar features of the products and services [10]. Its task emphasizes the identification of the targets of user's opinions or sentiments from online reviews [10], which is called feature extraction.

There are two main approaches to feature extraction in previous works: frequency-based method and syntax-based approach. Both of these approaches have their own limitations. Frequency-based method generally does not consider the low-frequency features. Together with that, it always takes all the frequent nouns as features [11]. However, for instance, the nouns "wife" and "gift" usually appearing in the reviews are all not features. Syntax-based approach has three key limitations. Firstly, this method crucially hinges on the grammatical accuracy of the sentences [12]. However, the grammatical structure of a sentence is usually erroneous owing to the fact that the comment text is close to the spoken language. Accordingly, it is capable of lowering the accuracy of this method. In addition to that, too long or too short sentence also exerts negative impact on this method [13]. Eventually, this method is hard to achieve high precision as well as high recall. Speaking otherwise, a particularly useful set of rules is hardly observable [14].

This study extends previous work that has attempted the HITS (Hyperlink-Induced Topic Search) algorithm on feature extraction [15], and aims to provide a better feature extraction algorithm based on feature sentiment networks. In particular, this study includes more types of products datasets, a feature-sentiment network with weighted edges, and an improved weighted HITS algorithm [16] for feature extraction called Mutual Information-based Hyperlink-Induced Topic Search (MHITS) which combines words pairs' co-occurrence relation and link structure. The method extract feature based on their importance which consists of two factors: one is relations between the product feature and the expressions of opinion [17], whereas, the other one is the weight between product features and sentiment words. The basic idea of feature ranking is that if a candidate feature frequently embellished by sentiment words, it should be considered to a high ranking. Besides, if a candidate sentiment frequently embellishes a large number of feature words, it should be listed as high ranking. The weight between product features and sentiment words is their co-occurrence frequency in the corpus. To be specific, the proposed method pre-processes online consumer reviews with the use of an analysis tool that can be employed for processing the review text for obtaining the feature-sentiment pairs and their frequency. Then they are utilized to construct a directed bipartite network by taking each word as the node and pointwise mutual information between nodes as the edge weight. Eventually, the MHITS algorithm is employed to filter the candidate features and sentiment words. We verify our approach based on the four consumer datasets collected from jd.com. Experimental results illustrate that the result of MHITS algorithm is better in comparison with the benchmark algorithms.

The remainder of the paper is organized as follows. We provide an overview of the related researches in Section 2. Section 3 presents the proposed product feature extraction and MHITS algorithms. The dataset and the experimental results are discussed in Section 4. Finally, we conclude in Section 5.

2. Related Work

Feature extraction has recently received considerable attention. There are many studies about feature extraction and some of them have achieved pretty good results. We introduce some related works in this section.

People often use similar vocabulary to describe the details of a product and express their opinions. Thus, those frequent nouns and adjectives are usually genuine features and sentiments. Based on this idea, Hu and Liu [18] utilize association rule mining to find out high-frequency nouns as features in customer reviews. Following their initial work, Popescu and Etzioni [19] use PMI (Pointwise mutual information) between noun phrases and a set of meronymy discriminators associated with the product categories to extract product features. Following this study, Li et al. [20] propose an extended PMI-IR (Pointwise Mutual Information and Information Retrieval) which is used to measure the semantic similarity between feature candidates and product entities to rank product feature. In Long et al. [21],

they firstly extracted high-frequency nouns as core feature words. Then they select other features by calculating information distance between the other word and the core feature word. The work of Scaffidi et al. [22] is based on an idea that the frequency of some words emerging in consumer reviews is higher than in ordinary English articles with the same length. These words are most likely to be the features. Thus, Scaffidi et al. [22] establish two corpora. One is a corpus of customer reviews, and the other is a corpus of spoken and written conversational English. This method identified product features by comparing the frequency of the nouns and noun phrases extracted from the prior corpus and the frequency of the same words in the later corpus. Frequency-based methods only need to use the information of Part-of-Speech tagging, and the accuracy of this method is high. Nevertheless, this approach usually ignores features that are seldom mentioned. Although these features are less frequent, they are important. For example, the phone's features "power" and "sound quality". Therefore, it is not enough to merely use the frequency-based method, and some rules require adding for finding low-frequency feature.

Zhuang et al. [23] select four most frequent dependency relations which are subject-predicate relations, adjectival modifying relations, relative clause modifying relations, and verb-object relations, respectively to extract features in the movie review dataset. Wu et al. [24] is carried out another dependency relation-based work. In his research, all nouns and verbs in reviews are taken into account as candidate features, and the adjectives near the candidate features are taken as candidate sentiment words. A tree kernel of the phrase dependency tree is defined, and it is incorporated within an SVM (support vector machine) to extract relations between opinion expression and product feature. Qiu et al. [25] consider that there are direct and indirect relations between words. Direct relation means that one word depends on the other word directly or they both depend on a third word directly. In contrast, indirect relation means that one word depends on another word through other words or they both depend on a third word indirectly. So Qiu et al. [25] propose an algorithm called DP (double propagation) built on direct relations. Several empirical rules were designed to extract sentiment words and features when given known sentiment words. Qiu et al. [26] use a seed opinion lexicon to refine the DP algorithm. The same is based on the DP algorithm, Zhang and Zhou [27] propose an extended DP algorithm with indirect dependency and comparative constructions to extract subjective feature. As a further study, Zhang et al. [15] propose that there are part-whole relations between the words in a sentence. Part-whole relations are felled into three categories, namely phrase pattern, sentence pattern, and "no" pattern. Both direct relation and part-whole relations are utilized to extract candidate features and feature indicators. In contrast, Su and Lynn [28] just utilize the phrase patterns of opinion words/phrases to extract features. Rana and Cheah [2] propose a two-fold rules-based model (TF-RBM) which extracts features associated with domain independent opinions and domain dependent opinions respectively. In addition, dependency trees are also used for feature extraction. Poria et al. [29] use dependency parse tree-based rules to extract features. They are combined sentence dependency trees with common-sense knowledge to detect features [30]. The most prominent advantage of syntax-based approach is that it can extract low-frequency feature and sentiment words. Another benefit is that this method can be applicable to large-corpus. However, the main shortcoming of this method is that it is difficult to find a set of proper rules. Nevertheless, the choice of rule set will significantly affect the result of feature extraction. Therefore, in order to be able to find the most effective set of rules, Liu et al. [14] employ a greedy algorithm and simulated annealing to select rules. In addition, Hai et al. [31] use association rules to mine potential rules that exist between features and sentiment words. They use association rule mining based on the co-occurrence matrix to mine the relationship between opinions and explicit feature words and form more robust rules by clustering explicit feature words. Although there are some researches on the choice of rules, it still cannot meet the needs of practical applications. A set of strict extraction rules is leading to high precision but low recall, while a general set of extraction rules is leading to high recall but low precision [11]. Thus, the results with high recall cannot be used directly as final features and sentiment words, and some additional methods need to be used to filter candidate features and sentiment words.

Consumers may be concerned about a variety of product features. However, each feature has different impacts on the process of consumer's purchase decision making [7]. Therefore, it is necessary to find a way to rank features automatically.

Hu and Liu [18] first use frequency to rank candidate features. However, ranking features using only frequency is not enough. Eirinaki et al. [17] propose a HAC(High Adjective Count) algorithm, in which the main idea is that nouns referring to more sentiment words are most likely to be actual product feature. On the basis of this idea, the HAC algorithm takes the number of adjectives for a noun to be its opinion score. Then candidate features are ranked in accordance with the sentiment score. Yan et al. [32] and Zhang et al. [15] use HITS and PageRank respectively to rank the candidate features. Among them, the main difference between our proposed algorithm and HITS is taking into account the frequency of word pairs. Additionally, Zha et al. [33] propose a probabilistic feature ranking algorithm based on SVM to calculate the importance of various features by combining feature frequency and relation between overall opinion and opinions on specific features. Wang et al. [34] propose a probabilistic rating regression model. They employ a bootstrapping-based algorithm to identify the major aspects, then propose a generative LRR (Latent Rating Regression) model to inferring aspect ratings and weights. Hai et al. [35] propose a feature selection method based on Intrinsic and Extrinsic Domain Relevance. They extract candidate feature by estimating IDR(intrinsic-domain relevance) and EDR(extrinsic-domain relevance) scores on the domain-dependent and domain-independent corpora to select features with larger IDR scores and smaller EDR scores. Later this method was used for feature ranking. Snyder and Barzilay [36] propose multiple aspect ranking using good grief algorithm. Zhou et al. [37] consider that there is a mutually reinforcing relationship between feature and microblog sentences. They first mainly considered three different similarity measures: character similarity, contextual similarity, and semantic similarity, which is used to cluster candidate features. Then, they propose an unsupervised label propagation algorithm based on the assumption that similar messages may focus on similar features to collectively rank the opinion target candidates of all sentences in a topic. Liu et al. [38] exploit a random walk-based co-ranking algorithm to estimate the confidence of each candidate features and sentiment words.

Frequency-based methods, syntax-based methods, and feature ranking are all important feature extraction methods. However, existing researches have rarely considered bipartite network structure constructed by the relationship of co-occurrence frequency between candidate feature and sentiment words to assist feature extraction. In addition, existing research data are generally electronic products such as computers and mobile phones. Other categories of products are rarely used, such as daily necessities and sporting goods. Therefore, a new algorithm that not only considered network structure and word frequency but also can be applied to multiple types of products is needed.

3. The Proposed Method

In order to reconcile the frequency-based with the syntax-based approaches, we propose a novel feature extraction method based on an extended HITS algorithm which is called MHITS. This method mainly consists of two steps: feature-sentiment pairs extraction and feature ranking. In the first step, the dependency relations between features and sentiment words are selected after dependency syntax analysis on partly customer reviews. Subsequently, they are used to extract candidate features-sentiment pairs of the entire corpus by using known dependency relation. In the second step, MHITS will calculate the authority values of candidate feature to find out the actual feature.

3.1. Analysis of Dependency Syntax

Dependency grammar is used to describe the relation between words in sentence. There are many types of dependency relations in a sentence such as "attribute", "adjunct", "head", "subject-verb", "adverbial" and so on. This paper uses LTP (HIT-SCIR, LTP [Language Technology Platform], <http://www.ltp-cloud.com/>) to automatically do POS tagging and analyse dependency relation of sentences. The LTP tool divides the dependency relation into 15 types, respectively "ATT", "RAD",

“VOB”, etc. Taking sentence “the appearance of the mobile phone is very good-looking” as an example, it can be observed that there are five kinds of dependency relations, which are “ATT (attribute)”, “RAD (adjunct)”, “HED (head)”, “SBV (subject-verb)”, and “ADV (adverbial)” respectively. Among them, the relationship between “mobile phone” and the word “appearance” is an attribute relation “ATT”; “appearance” and “good-looking” are a subject-verb relation “SBV”; “mobile phone” and “of” are an adjunct relation “RAD”; “very” and “good-looking” are an adverbial relation “ADV”, and the entire sentence and “good-looking” constitute a head relation “HED”. However, only part of these dependency relations is helpful for identifying product features and sentiment words. In this research, candidate product features and its corresponding sentiment words are identified by LTP. Specifically, through the dependency relation analysis of the sentence, we find that the relation between feature and sentiment is mainly subject-predicate relation, namely the “SBV” relation. However, the “ATT” relation is also considered to improve the recall rate of features. They are used in previous feature extraction studies [35,39]. So, we manually select two types of dependency relation, “SBV” and “ATT”.

“SBV” relation represents the subject-verb relation of words in a sentence. Two specific “SBV” models are as follows.

- **Adj+Noun:** This pattern represents a word pair that meets the “SBV” relation, in which the subject must be a noun and the predicate is an adjective. As shown in Figure 1, “quality” is a noun and “good” is an adjective. Moreover, “quality” is modified by “good”. It can be found in this sentence that “quality” is a product feature and “good” is an opinion word on it. Therefore, “Adj+Noun” means that noun is modified by a single adjective.
- **COO+SBV:** “COO” represents a parallel relation of words in a sentence. “COO+SBV” pattern represents the fact that there are multiple adjective modifiers of a noun to form multiple “SBV” relations, and these “SBV” relations constitute “COO” relations. For sentence “this screen is clear and large”, “screen” is the feature word and “large” and “clear” are opinion words. After parsing, it can find that “large” and “clear” depend on “screen”. That is, “screen” and “clear”, “large” are all “SBV” relation, and the relationship between these two “SBV” relations is “COO”. In other words, “COO+SBV” means that both “large” and “clear” modifies “screen”.

“ATT” relation represents an attribute relationship in a sentence. Three specific “ATT” models are as follows.

- **Noun+Adj:** This pattern represents a word pair that satisfies the “ATT” relation, in which attributive is an adjective and the attributive modifier is a noun. In the example of Figure 1, “exquisite” is not only an adjective but also an attributive of “design”.
- **ATT+COO:** This pattern represents two nouns are modified by the same adjective. For example, in sentence “Reasons I bought it is its clear picture quality and sound quality”, it can be found that the adjective “clear” modifies both two features of “picture quality” and “sound quality”.
- **COO+ATT:** This pattern means that multiple adjectives modify one noun in an “ATT” relation. For example, in sentence “I love this large and clear screen”, both words “large” and “clear” are not only adjectives but also attributes of the “screen”.



Figure 1. Example of dependency relations in Chinese.

3.2. Feature Filtering

Although these dependency relations cover almost all product features in reviews, not all nouns in these relations are features. Therefore, some simple rules should be utilized to eliminate manifest non-feature nouns. We find that only nouns referring to specific properties of a product are features. Otherwise, they are generally not features. For example, “quality” and “appearance” are features, but “thing” does not. In addition, brand names of products will be eliminated. For instance, phone’s brand names “Apple” and “Samsung”. Nevertheless, there is a special feature of a product, which is itself as a whole. For example, in sentence “this phone is perfect”, the word “phone” can be taken as a feature of the product. So the noun that refers to the product itself will not be eliminated. In this study, if a candidate feature belongs to one of the following types of noun or noun phrases listed in Table 1, the feature-sentiment pair will be removed from the candidate pairs set generated by the previous steps.

Table 1. Examples of non-feature words.

Types	Examples
Brand names	Apple, Huawei, Leshi
Non-specific words	Things, Products, Machines

3.3. The MHITS Algorithm

HITS is a link analysis algorithm which was proposed by Kleinberg in 1999 to evaluate the importance of web pages [40]. The main idea of the HITS algorithm is that the number of web pages referenced and the number of its links are used to calculate its authority and hub value respectively [41]. Specifically, HITS algorithm can be described as follows. Firstly, it collects k highest ranked pages that are assumed to be highly correlated with a broad search query q in search engine. This set R with k pages is called root set. Then, other nodes that point to nodes of R will constantly become nodes of R . Finally, R becomes a base set S when there is not any more node point to node in it. The (directed) link graph of S is denoted by $G = (V, E)$, in which V is the set of nodes and E is the set of directed edges. HITS works on G and assigns to each node an authority score and a hub score. The authority score estimates the value of the content of the page, and the hub score estimates the value of its links to other pages. The relationship of them is represented as follows:

$$\begin{aligned}
 A(i) &= \sum_{(j,i) \in E} H(j) \\
 H(i) &= \sum_{(i,j) \in E} A(j),
 \end{aligned}
 \tag{1}$$

where $A(i)$ is the authority score of the node i , and $H(i)$ is the hub score of node i .

From Equation (1), it can be found that the node authority is the sum of the hub score of all nodes which it links to, and the hub score of a node is the sum of the authority score of nodes which it points to. The node with larger authority score is called authority node, and the node with larger hub score is called hub node. If a node is linked by many hub nodes, it will have a high authority score. Similarly, if a node is connected by many authority nodes, it will have a high hub score. Thus, authority nodes and hub nodes have a mutual reinforcement relationship [15].

However, the original HITS algorithm has apparent shortcomings when it was employed in search engines. It completely excludes text and content of the web page. Due to this algorithm treats all hyperlinks identical and lack semantic analysis, it is prone to the phenomenon of “Topic-Drift” which will result in mismatch of query results and consumer requirements [16]. Thus, in order to take into account the text and subject of web pages, Jon Kleinberg and David Gibson [16] introduced a weight to the original HITS algorithm. The idea is assigning a positive numerical weight $w(p, q)$

to each edge from node p to node q . This weight measures the authority on the topic, which is the number of matching terminology of topic description in two web pages.

In this paper, we introduce Pointwise mutual information into HITS algorithm to identify actual features. Before illustrating how mutual information can be applied to HITS, let us first give a brief introduction to mutual information. Mutual information (MI) is a measure of the information overlap between two random variables [42]. The MI between random variables X and Y is defined as:

$$I(X;Y) = \sum_{x,y} p(x,y) \log \frac{p(x,y)}{p(x)p(y)}, \quad (2)$$

where $p(x)$ and $p(y)$ are marginal probabilities of X and Y respectively, and $p(x,y)$ is joint probability. MI is also taken as the expected or average value of pointwise mutual information which is used to measure the correlation between two individual events $p(x)p(y)$ [43].

$$PMI(x;y) = \log \frac{p(x,y)}{p(x)p(y)} \quad (3)$$

However, PMI is sensitive to low frequency, which will result in relatively high score for low frequency events. It is more accurately described as a lack of sensitivity to high frequency [43]. Therefore, we use the normalized PMI called PMI^2 which is used by Bouma [43] in collocation extraction to avoid this bias. The normalized PMI is defined as:

$$PMI^2(x;y) = \log \left(\frac{\frac{p(x,y)}{p(x)p(y)}}{\frac{1}{p(x,y)}} \right) = \log \left(\frac{p(x,y)^2}{p(x)p(y)} \right) \quad (4)$$

Normalized PMI has less biased towards low frequency collocations. So it has a moderate but positive effect on the effectiveness in measure the co-occurrence frequency of words [43].

Normalized PMI reflects tightness between two variables. The higher the normalized PMI value is, the higher the correlation between X and Y is. In other words, if the co-occurrence frequency of a feature-sentiment pair is very high, the relation between feature and its sentiment of this word pair is tight. Therefore, the pair with high normalized PMI is more likely to express the consumer's attention to this product feature with corresponding sentiment.

People often use nouns to denote product features and use adjectives to express their sentiments toward specific product features [44]. In the context of feature extraction, a noun or noun phrase is more likely to be a feature if it is modified by quite a few adjectives [17]. Similarly, an adjective is more likely to be a sentiment word if it is the modifier of many product features. In this paper, we treat candidate feature and sentiment word as different kinds of nodes in a network, which is called authority node and hub node respectively. Modifying relations between candidate features and sentiment words is taken as edges of the network. Based on the above analysis, we define such a network as a directed bipartite feature-sentiment relation network.

The MHITS algorithm measures the importance of network nodes based on the link structure of a network and co-occurrence relation between nodes. In this directed bipartite feature-sentiment network, the authority value of a candidate feature (authority node) can be calculated by the sum of hub value of candidate sentiment words (hub node) that link to it. If a candidate feature is modified by many candidate sentiment words with high hub value, this feature will have a high authority value. On the other hand, the hub value of a candidate sentiment word can be calculated by the sum of authority value of candidate features that link to it. If a candidate sentiment word modifies many feature words with high authority values, it will have a high hub value. Finally, the edge weight is calculated by the PMI^2 between authority nodes and hub nodes. If the PMI^2 value of a feature-sentiment word pair is high, the edge of corresponding network has a high weight. Thus, if the hub or authority value of a node is high, this node is more likely to be an actual sentiment or feature.

Assuming that N represents the set of dependency word pairs; F represents the set of candidate feature in W ; S is the set of candidate sentiment in N . Nodes in this directed bipartite feature-sentiment network are divided into two categories, namely authority node and hub node. Authority node is comprised of candidate feature words, and hub node is composed of candidate sentiment words. If an authority node f_i and a hub node s_j are a pair of candidate feature and sentiment word, an edge e_{ji} of the network is directed by s_j towards f_i . The weight of edge w_{ji} is measured by the normalized pointwise mutual information between s_j and f_i . Figure 2 is an example of this network.

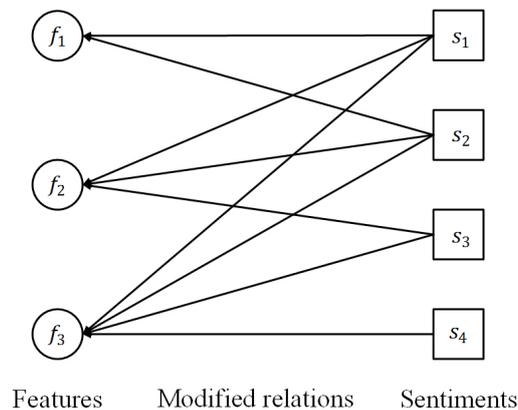


Figure 2. Relations between sentiment and feature.

Considering that a candidate feature is more likely to be an actual feature if it is modified by more hub adjectives. We define the authority value $p(f_i)$ of an authority node f_i as:

$$p(f_i(0)) = \frac{\sum_{j \in T} w_{ji}(0)}{\sum_{i \in F} \sum_{j \in S} w_{ji}(0)}, \quad (5)$$

where $p(f_i)$ is the value of the authority node; T is the set of hub nodes linked to f_i ; F is the set of authority nodes in the network; S is the set of hub nodes in the network; w_{ji} is the weight of the edge from hub node s_j to authority node f_i ; w_{ji} is the weight of the edge from s_j to f_i as follows:

$$w_{ji}(k) = \log \frac{p(f_i, s_j)^2}{p(f_i)p(s_j)} = \log \frac{w_{ji}(k-1)^2}{p(f_i)p(s_j)}, \quad (6)$$

where $p(f_i, s_j)$ is the value of feature-sentiment word pairs, which is the pointwise mutual information of a hub node s_j and an authority node f_i . Thus, it can also be represented by the weight of edge of s_j pointing to f_i .

The more candidate features modified by an adjective are, the more likely this adjective is an actual sentiment word. We define the hub value of node s_j as:

$$p(s_j(0)) = \frac{\sum_{i \in U} w_{ji}(0)}{\sum_{i \in F} \sum_{j \in S} w_{ji}(0)}, \quad (7)$$

where U is the set of authority nodes linked to s_j .

The MHITS algorithm can be described as follows:

- Initial step: let the value of authority node in the network be $p(f_i(0)); i = 1, 2, \dots, n$. The value of the hub node be $p(s_j(0)); j = 1, 2, \dots, m$. The edge $w_{ji}(0)$ is the co-occurrence frequency of the pair of feature-sentiment words represented by nodes.
- Iterative process: The following three operations are carried out at step $k(k \geq 1)$:
 - Authority value adjustment rule: the value of each authority node is adjusted to the sum of the edge weights of the hub nodes.

$$p(f_i(k)) = \frac{\sum_{j \in T} w_{ji}(k-1)}{\sum_{i \in F} \sum_{j \in S} w_{ji}(k-1)} \quad (8)$$

- Hub value adjustment rule: the value of each hub node is adjusted to the sum of the edge weights of the authority nodes.

$$p(s_j(k)) = \frac{\sum_{i \in U} w_{ji}(k-1)}{\sum_{i \in F} \sum_{j \in S} w_{ji}(k-1)} \quad (9)$$

- The weights of edge adjustment rule:

$$w_{ji}(k) = \log \frac{w_{ji}(k-1)^2}{p(f_i(k))p(s_j(k))} \quad (10)$$

This problem is addressed by using a power iteration method. The MHITS algorithm computes iteratively until it converges. Finally, the value of node and edge tends to be stable in the network.

4. Experiments

In order to verify MHITS algorithm, we first collect Chinese online customer reviews of four different products from Jingdong (www.jd.com), then use MHITS algorithm to rank product features, and finally compare results with two baselines HITS [15] and EXPRS [32].

4.1. Data Sets

We crawl online all the Chinese consumer reviews of four products up to 25 April 2017 from <http://www.jd.com>. These four products include two mobile phones, one facial cleanser, and one racket, with 3224, 1888, 1559 and 2316 reviews respectively. Among them, mobile phone reviews are typically used in previous studies [32]. Facial cleansers are daily necessities, and rackets are sporting goods. They are rarely employed in previous studies.

In this paper, we are preprocessing the dataset by eliminated following types of irrelevant texts. The first type is texts which are referring to neither product nor consumer opinion, such as advertising information, review replies, etc. The second type is reviews which does not fit with language expression habit, such as the sentence has no punctuation or has too many repetitive characters. In other words, if the same word repeated many times in a review, it is likely to be comment spam. So only a review that deleted repetition word more than seven words, it will be retained. Last, short reviews less than 10 words will be eliminated.

We employ eight graduate students to manually annotate the dataset. First of all, they discuss together to reach a consensus on fuzzy feature-sentiment pairs words. Then, each student is assigned to the task according to the amount of data. The word pairs that more than two individuals labeled will be the outcome of the final marking. Finally, the annotators are unanimous in 76% of labels. In addition, the results of annotation are sorted out. Numbers of finally agreed features of four products are shown in Table 2.

Table 2. Details of the four product data after preprocessing.

Product Name	Number of Reviews	Number of Reviews Words	Number of Manually Labeled Features
Facial cleanser	1006	23,845	20
Racket	1486	37,481	25
Leshi mobile	2141	64,918	56
Huawei mobile	1359	35,149	67

4.2. Evaluation Metrics

We use precision (P), recall (R), and F-measure (F), which are often used in natural language processing, to assess the effectiveness of the proposed MHITS algorithm. Besides, we also use precision@N [45] which is the proportion of the actual features in the top N results of in a ranked list.

4.3. Benchmarks

In order to validate MHITS algorithm, we adopt two similar algorithms as benchmarks, which are an algorithm based on the HITS proposed by Zhang et al. [15] and an algorithm based on the PageRank called EXPRS algorithm proposed by Yan et al. [32]. Both of them are used to feature extraction based on the idea of node ranking on a network. Zhang et al. [15] assumed that if a candidate feature is modified by many adjectives, this candidate feature is highly possible to be a real feature. Therefore, they use an improved double propagation to extract candidate features and feature indicators which form a directed bipartite graph. Then the HITS algorithm is used to calculate feature relevance. The main idea of EXPRS algorithm [32] is that a network node is important if it is linked by many other important nodes. So they use feature-sentiment word pairs extracted by dependency relations to constitute a network, in which the Cartesian product of candidate features and sentiment words is taken as nodes in the network. Then an extended PageRank is used to assess the importance of nodes and derive product feature.

5. Results

In this part, we apply the MHITS algorithm to our dataset of reviews of four products. At the same time, we compare results of our method's with benchmarks on Precision, Recall and F-measure. The results of the four datasets are shown in Tables 3–5, respectively. Besides, the results of precision@N will be discussed separately.

Table 3. The precision of HITS, EXPRS, and MHITS on four kinds of review data.

	Facial Cleanser	Racket	Leshi Mobile	Huawei Mobile
HITS	0.92	0.86	0.51	0.69
EXPRS	0.75	0.73	0.54	0.38
MHITS	0.86	0.78	0.75	0.75

Table 4. The recall of HITS, EXPRS, and MHITS on four kinds of review data.

	Facial Cleanser	Racket	Leshi Mobile	Huawei Mobile
HITS	0.55	0.72	0.63	0.4
EXPRS	0.58	0.52	0.49	0.54
MHITS	0.60	0.84	0.54	0.4

Table 5. The F-measure of HITS, EXPRS, and MHITS on four kinds of review data.

	Facial Cleanser	Racket	Leshi Mobile	Huawei Mobile
HITS	0.69	0.78	0.56	0.51
EXPRS	0.65	0.60	0.45	0.45
MHITS	0.71	0.81	0.63	0.52

As can be seen from Table 2, facial cleansers and rackets have fewer product features with only 20–30 features, but Huawei and Leshi mobile have 50–70 product features. Combining Table 2 with the results, it can be seen that our approach has a higher recall than other algorithms for products with fewer features, and has a higher precision than other algorithms for products with a lot of features. Moreover, our method outperforms other methods on F-measure. The main reason causing such a

result is the effect of general collocation words. General collocation words are high frequency sentiment words that can modify most of feature word, and they are very common in customer reviews. There are five most commonly used general collocation words, “bad”, “good”, “poor”, “pretty good”, “not bad”. They can be used to modify the majority of the features of different products. The proportion of word pairs that including general collocation words in all word pairs of facial cleansers, racket, Leshi mobile and Huawei mobile is 0.3, 0.41, 0.26, and 0.32, respectively. Additionally, the proportion of real word pairs that including general collocation words in all real word pairs is 0.4, 0.59, 0.36 and 0.37, respectively. In the real general collocation word pairs, the proportion of feature words matching with two or more general collocation words is 0.79, 0.66, 0.5 and 0.38 respectively. The general collocation word has the characteristics of high frequency, and links with many candidate feature words in the network. That means it has a high hub value. Besides, because the authority nodes and hub nodes have mutually reinforcing relationship, those candidate features that linked with general collocation word also have higher authority values and ranking. Therefore, the larger the number of features linking with general collocation words is, the higher the recall of product features is. Moreover, the more features that link with multiple general collocation words are, the higher the precision of product features is. In addition, considering pointwise mutual information utilizes words frequency, the effect of general collocation words will be strengthened.

Next, in order to show that the results of our algorithm improve the ranking of most features, we use the precision@N metric to compare the precision of three algorithm ranking results on four products. The ranking comparison is shown in Tables 6–9, which gave the precision of top 10 to 70 results respectively.

Table 6. The precision of top 10 to 70 result of HITS, EXPRS, and MHITS on facial cleanser dataset.

	HITS	EXPRS	MHITS
10	0.90	0.86	0.90
20	0.60	0.61	0.65
30	0.43	0.41	0.47
40	0.43	0.32	0.38
50	0.36	0.27	0.34
60	0.30	0.23	0.28
70	0.26	0.21	0.24

Table 7. The precision of top 10 to 70 result of HITS, EXPRS, and MHITS on racket dataset.

	HITS	EXPRS	MHITS
10	0.90	0.85	0.90
20	0.85	0.66	0.80
30	0.60	0.47	0.70
40	0.45	0.36	0.55
50	0.36	0.33	0.44
60	0.33	0.29	0.37
70	0.29	0.27	0.31

Table 8. The precision of top 10 to 70 result of HITS, EXPRS, and MHITS on Huawei mobile dataset.

	HITS	EXPRS	MHITS
10	0.80	0.88	0.80
20	0.85	0.71	0.85
30	0.73	0.55	0.77
40	0.68	0.46	0.68
50	0.56	0.45	0.58
60	0.47	0.42	0.53
70	0.46	0.41	0.49

Table 9. The precision of top 10 to 70 result of HITS, EXPRS, and MHITS on Leshi mobile dataset.

	HITS	EXPRS	MHITS
10	0.90	0.90	1.00
20	0.90	0.72	0.95
30	0.77	0.57	0.83
40	0.65	0.45	0.75
50	0.54	0.40	0.62
60	0.48	0.38	0.55
70	0.50	0.37	0.50

As can be observed in Tables 6–9, the proposed method outperforms other benchmarks in all datasets except for facial cleansers. However, the precision of the top 10–30 results of the algorithm is higher than other benchmarks, and the recall rate of the top 30 results of this algorithm reaches 71 percent in facial cleanser dataset. Therefore, this result is acceptable. In addition, since the features of facial cleansers and rackets are only about 20, N takes 40, 50, 60 is a bit large for facial cleansers and rackets when calculating the precision@ N . So, there are some lower values for the two products. We calculate them to unify the expression of the four products' result. In summary, it can be seen from the results that our proposed algorithm does improve the ranking of most features in the results.

Finally, we measure the processing time of the proposed algorithm and the two benchmarks by using 400 and 500 word pairs. All tests are performed on the same device. The execution time of 400 and 500 word pairs by HITS, EXPRS, and MHITS is shown in Table 10.

Table 10. The execution time of 400 and 500 word pairs by HITS, EXPRS, and MHITS.

	HITS	EXPRS	MHITS
400	1.51	21.56	2.04
500	1.85	125.80	2.27

From Table 10, it can be seen that the time-consuming of both the MHITS algorithm and the HITS algorithm is similar. Moreover, by comparing the performance of the algorithm in 400 and 500 word pairs, it can also be found that the time-consuming of MHITS algorithm does not grow much after adding 100 word pairs. Therefore, it can be concludes that the weighted HITS use PMI to measure the co-occurrence relation of authority nodes and hub nodes, which does not affect the overall performance of it too much.

6. Discussion

This paper proposed a new extended HITS algorithm, namely MHITS, to discover actual product features by both considering the relations between product feature and expressions of opinions and the fixed degree between product features and sentiment words. This method first uses dependency relations to extract candidate feature-sentiment pairs. It then ranks the extracted candidate feature by MHITS algorithm. Finally, the effectiveness of the proposed algorithm is evaluated by using real data and two benchmark methods. Experimental results illustrate that MHITS is more effective than two baseline methods in F-measure across all three different types of four products.

There are three-fold theoretical and practical significance of this study. First, since the MHITS algorithm is a weighted HITS algorithm based on the co-occurrence frequency between nodes, it can be used not only for feature extraction but also for identification of important nodes. Secondly, the MHITS algorithm not only needs to build a small feature-emotion network but also requires less computing resources. Therefore, the weighted HITS algorithm requires less time for operation and is easy to use in real life. Finally, the proposed method is able to help mitigate the information overloads. For the consumers, this method can help them quickly figure out the information of product feature that they

are interested in. Accordingly, it does not just minimize the time required for the consumer to collect the product information but also helps them make better purchase decisions. Conversely, it is capable of supporting the e-commerce platforms, in addition to providing personalized information screening and product recommendations for the consumers. Since each user's product feature preference is different, e-commerce platform can predict this user's product feature preferences on the basis of the consumer's query to the product features. Accordingly, the e-commerce platform is not only capable of screening out the product feature reviews that consumers require beforehand but also delivering the products, which are consistent with their feature preferences. As regards the manufacturers, the proposed approach is likely to help them manage their product's reviews in real-time. The manufacturer can understand the pros and cons of its product in the market through the analysis of the consumers' comments on specific product features, so as to improve the product. Furthermore, the manufacturers can develop the strategy for controlling the public opinion or marketing through the analysis of the impacts of public opinion or advertisement on the changes in the product features.

However, there are some limitations using our approach. The first one is the limitation of experimental data. The validity of the proposed algorithm is verified by the three distinct types of four products used in this paper. Although these four products have a certain representation, it is not comprehensive enough. Therefore, the future work should be on more different types of products to test the effectiveness of the proposed algorithm.

The second one is the utilization of general collocation word. Not all consumers have a rich vocabulary to describe their purchase experience and feelings. Most consumers generally use the easiest and most commonly used words to express their opinions. From the four products used in this article, we can see that consumers' vocabulary is different in the evaluation of different products. For some of relatively unfamiliar products, consumers generally use a small number of words to express opinions and usually using general collocation words. Therefore, how to use general collocation words to make it work better is the next step in research.

Author Contributions: Chen Liu and Wei Shan conceived and designed the experiments; Li Tang performed the experiments; Chen Liu, Li Tang and Wei Shan analyzed the data; Li Tang wrote the paper. All authors read and approved the final manuscript.

Acknowledgments: The authors are very grateful for the insightful comments and suggestions of the anonymous reviewers and the editor, which have helped to significantly improve this article. Furthermore, this work was supported by the National Natural Science Foundation of China (Nos. 71401107, 71371025, 71774111), and the Beijing Natural Science Foundation (No. 9182010).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Ren, G.; Hong, T. Investigating Online Destination Images Using a Topic-Based Sentiment Analysis Approach. *Sustainability* **2017**, *9*, 1765. [[CrossRef](#)]
2. Rana, T.A.; Cheah, Y.N. A Two-Fold Rule-Based Model for Aspect Extraction. *Expert Syst. Appl.* **2017**, *89*, 273–285. [[CrossRef](#)]
3. Ma, B.; Zhang, D.; Yan, Z.; Kim, T. An LDA and synonym lexicon based approach to product feature extraction from online consumer product reviews. *J. Electron. Commer. Res.* **2013**, *14*, 304–314.
4. Mikalef, P.; Giannakos, M.; Pateli, A. Shopping and Word-of-mouth Intentions on Social Media. *J. Theor. Appl. Electron. Commer. Res.* **2013**, *8*, 17–34. [[CrossRef](#)]
5. Qi, J.; Zhang, Z.; Jeon, S.; Zhou, Y. Mining Customer Requirements from Online Reviews: A Product Improvement Perspective. *Inf. Manag.* **2016**, *53*, 951–963. [[CrossRef](#)]
6. D'Addio, R.M.; Domingues, M.A.; Manzato, M.G. Exploiting feature extraction techniques on users' reviews for movies recommendation. *J. Braz. Comput. Soc.* **2017**, *23*, 7:1–7:24. [[CrossRef](#)]
7. Yang, C.; Yu, X.; Liu, Y.; Nie, Y.; Wang, Y. Collaborative filtering with weighted opinion aspects. *Neurocomputing* **2016**, *210*, 185–196. [[CrossRef](#)]
8. Sun, Y.; Ma, H.; Chan, E.H.W. A Model to Measure Tourist Preference toward Scenic Spots Based on Social Media Data: A Case of Dapeng in China. *Sustainability* **2018**, *10*, 43. [[CrossRef](#)]

9. Liu, B. Sentiment Analysis and Opinion Mining. *Synth. Lect. Hum. Lang. Technol.* **2012**, *5*, 1–167. [[CrossRef](#)]
10. García-Pablos, A.; Cuadros, M.; Rigau, G. W2VLDA: Almost unsupervised system for Aspect Based Sentiment Analysis. *Expert Syst. Appl.* **2018**, *91*, 127–137. [[CrossRef](#)]
11. Schouten, K.; Frasincar, F. Survey on Aspect-Level Sentiment Analysis. *IEEE Trans. Knowl. Data Eng.* **2016**, *28*, 813–830. [[CrossRef](#)]
12. Poria, S.; Cambria, E.; Gelbukh, A. Aspect extraction for opinion mining with a deep convolutional neural network. *Knowl.-Based Syst.* **2016**, *108*, 42–49. [[CrossRef](#)]
13. Wang, W.; Tan, G.; Wang, H. Cross-domain comparison of algorithm performance in extracting aspect-based opinions from Chinese online reviews. *Int. J. Mach. Learn. Cybern.* **2016**, *8*, 1–18. [[CrossRef](#)]
14. Liu, Q.; Gao, Z.; Liu, B.; Zhang, Y. Automated Rule Selection for Opinion Target Extraction. *Knowl.-Based Syst.* **2016**, *104*, 74–88. [[CrossRef](#)]
15. Zhang, L.; Liu, B.; Lim, S.H.; O'Brien-Strain, E. Extracting and ranking product features in opinion documents. In Proceedings of the 23rd International Conference on Computational Linguistics: Posters, Beijing, China, 23–27 August 2010; Association for Computational Linguistics: Stroudsburg, PA, USA, 2010; pp. 1462–1470.
16. Chakrabarti, S.; Dom, B.; Raghavan, P.; Rajagopalan, S.; Gibson, D.; Kleinberg, J. Automatic resource compilation by analyzing hyperlink structure and associated text. *Comput. Netw. ISDN Syst.* **1998**, *30*, 65–74. [[CrossRef](#)]
17. Eirinaki, M.; Pisal, S.; Singh, J. Feature-based opinion mining and ranking. *J. Comput. Syst. Sci.* **2012**, *78*, 1175–1184. [[CrossRef](#)]
18. Hu, M.; Liu, B. Mining and summarizing customer reviews. In Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Seattle, WA, USA, 22–25 August 2004; ACM: New York, NY, USA, 2004; pp. 168–177.
19. Popescu, A.M.; Etzioni, O. OPINE: Extracting Product Features and Opinions from Reviews. In Proceedings of the HLT/EMNLP on Interactive Demonstrations, Vancouver, BC, Canada, 7 October 2005; Association for Computational Linguistics: Stroudsburg, PA, USA, 2005; pp. 32–33.
20. Li, S.; Zhou, L.; Li, Y. Improving aspect extraction by augmenting a frequency-based method with web-based similarity measures. *Inf. Process. Manag.* **2015**, *51*, 58–67. [[CrossRef](#)]
21. Long, C.; Zhang, J.; Zhut, X. A Review Selection Approach for Accurate Feature Rating Estimation. In Proceedings of the 23rd International Conference on Computational Linguistics: Posters, Beijing, China, 23–27 August 2010; Association for Computational Linguistics: Stroudsburg, PA, USA, 2010; pp. 766–774.
22. Scaffidi, C.; Bierhoff, K.; Chang, E.; Felker, M.; Ng, H.; Jin, C. Red Opal: Product-feature scoring from reviews. In Proceedings of the 8th ACM Conference on Electronic Commerce, San Diego, CA, USA, 11–15 June 2007; ACM: New York, NY, USA, 2007; pp. 182–191.
23. Zhuang, L.; Jing, F.; Zhu, X.Y. Movie review mining and summarization. In Proceedings of the 15th ACM International Conference on Information and Knowledge Management, Arlington, VA, USA, 06–11 November 2006; ACM: New York, NY, USA, 2006; pp. 43–50.
24. Wu, Y.; Zhang, Q.; Huang, X.; Wu, L. Phrase dependency parsing for opinion mining. In Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: Volume 3-Volume 3, Singapore, 6–7 August 2009; Association for Computational Linguistics: Stroudsburg, PA, USA, 2009; pp. 1533–1541.
25. Qiu, G.; Liu, B.; Bu, J.; Chen, C. Expanding domain sentiment lexicon through double propagation. In Proceedings of the 21st International Joint Conference on Artificial Intelligence, Pasadena, CA, USA, 11–17 July 2009; Morgan Kaufmann Publishers Inc.: San Francisco, CA, USA, 2009; pp. 1199–1204.
26. Qiu, G.; Liu, B.; Bu, J.; Chen, C. Opinion word expansion and target extraction through double propagation. *Comput. Linguist.* **2011**, *37*, 9–27. [[CrossRef](#)]
27. Kang, Y.; Zhou, L. RubE: Rule-based methods for extracting product features from online consumer reviews. *Inf. Manag.* **2017**, *54*, 166–176. [[CrossRef](#)]
28. Su, S.H.; Lynn, K.T. Extracting Product Features and Opinion Words Using Pattern Knowledge in Customer Reviews. *Sci. World J.* **2013**, *2013*, 394758:1–394758:5.
29. Poria, S.; Ofek, N.; Gelbukh, A.; Hussain, A.; Rokach, L. Dependency tree-Based rules for concept-level aspect-based sentiment analysis. In *Semantic Web Evaluation Challenge*; Springer: Cham, Switzerland, 2014; pp. 41–47.

30. Poria, S.; Cambria, E.; Ku, L.; Gui, C.; Gelbukh, A. A rule-based approach to aspect extraction from product reviews. In Proceedings of the Second Workshop on Natural Language Processing for Social Media, Dublin, Ireland, 24 August 2014; Association for Computational Linguistics: Stroudsburg, PA, USA, 2014; pp. 28–37.
31. Hai, Z.; Chang, K.; Kim, J.J. Implicit Feature Identification via Co-occurrence Association Rule Mining. In *Computational Linguistics and Intelligent Text Processing*; Springer: Berlin/Heidelberg, Germany, 2011; pp. 393–404.
32. Yan, Z.; Xing, M.; Zhang, D.; Ma, B. EXPRS: An extended pagerank method for product feature extraction from online consumer reviews. *Inf. Manag.* **2015**, *52*, 850–858. [[CrossRef](#)]
33. Zha, Z.J.; Yu, J.; Tang, J.; Wang, M.; Chua, T.S. Product Aspect Ranking and Its Applications. *IEEE Trans. Knowl. Data Eng.* **2014**, *26*, 1211–1224.
34. Wang, H.; Lu, Y.; Zhai, C. Latent aspect rating analysis on review text data: a rating regression approach. In Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Washington, DC, USA, 25–28 July 2010; ACM: New York, NY, USA, 2010; pp. 783–792.
35. Hai, Z.; Chang, K.; Kim, J.J.; Yang, C.C. Identifying Features in Opinion Mining via Intrinsic and Extrinsic Domain Relevance. *IEEE Trans. Knowl. Data Eng.* **2014**, *26*, 623–634. [[CrossRef](#)]
36. Snyder, B.; Barzilay, R. Multiple aspect ranking using the good grief algorithm. In *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics*; Association for Computational Linguistics: Stroudsburg, PA, USA, 2007; pp. 300–307.
37. Zhou, X.; Wan, X.; Xiao, J. CMiner: Opinion extraction and summarization for chinese microblogs. *IEEE Trans. Knowl. Data Eng.* **2016**, *28*, 1650–1663. [[CrossRef](#)]
38. Liu, K.; Xu, L.; Zhao, J. Co-Extracting Opinion Targets and Opinion Words from Online Reviews Based on the Word Alignment Model. *IEEE Trans. Knowl. Data Eng.* **2015**, *27*, 636–650. [[CrossRef](#)]
39. Che, W.; Zhao, Y.; Guo, H.; Su, Z.; Liu, T. Sentence compression for aspect-based sentiment analysis. *IEEE/ACM Trans. Audio Speech Lang. Process.* **2015**, *23*, 2111–2124.
40. He, X.; Zha, H.; Ding, C.H.Q.; Simon, H.D. Web document clustering using hyperlink structures. *Comput. Stat. Data Anal.* **2002**, *41*, 19–45. [[CrossRef](#)]
41. Soumen, C. Mining the Web discovering knowledge from hypertext data. *Sci. Technol. Books* **2002**, *55*, 275–276.
42. Mounts, R.B. Transmission of Information. *Bell Labs Tech. J.* **1961**, *14*, 56–58.
43. Bouma, G. Normalized (Pointwise) Mutual Information in Collocation Extraction. In Proceedings of the German Society for Computational Linguistics, Konstanz, Germany, 7–19 September 2009; Springer: Berlin/Heidelberg, Germany, 2009; pp. 31–40.
44. Wei, C.P.; Chen, Y.M.; Yang, C.S.; Yang, C.C. Understanding what concerns consumers: a semantic approach to product feature extraction from consumer reviews. *Lect. Notes Bus. Inf.* **2010**, *8*, 149–167. [[CrossRef](#)]
45. Bing, L. Web Data Mining: Exploring Hyperlinks, Contents, and Usage Data (Data-Centric Systems and Applications). *Acm Sigkdd Explor. Newsl.* **2012**, *10*, 23–25.



© 2018 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).