Addressing Overchoice: Automatically Generating Meaningful Filters from Hotel Reviews

ISTVÁN VARGA, Megagon Labs, Tokyo, Japan, Recruit Co., Ltd., Japan
YUTA HAYASHIBE, Megagon Labs, Tokyo, Japan, Recruit Co., Ltd., Japan

In this paper we present a hotel filter recommendation method designed to address the cognitive load users face in an overchoice scenario. As online products and services are continuously diversifying, user needs are also becoming increasingly sophisticated. However, with more items to choose from, grasping the entire choice set and differentiating among all matching options becomes increasingly difficult, leading to sub-optimal outcomes. Conventional hotel reservation platforms provide with a limited set of additional filters, but these can not accommodate all intricate user needs. Employing natural language processing and machine learning techniques, we provide a simple framework that identifies meaningful filters from customer reviews. We define criteria and scoring methods to acquire relevant and interesting filters that may help customers refine their needs or even identify hidden, previously unknown ones. Our simulated user experiments show that our proposal is capable of identifying intricate and useful filters, leading to increased customer satisfaction.

CCS Concepts: • **Computing methodologies** → **Machine learning**; • **Information systems** → **Content ranking**; **Recommender systems**; **Rank aggregation**; **Similarity measures**.

Additional Key Words and Phrases: overchoice, clustering, filter recommendation

1 INTRODUCTION

Online services have become not only ubiquitous, but indispensable in almost every aspect of our life. Nearly every imaginable product or service is available through e-commerce transactions, including online shopping, restaurant or hotel reservation to matchmaking. In a conventional hotel reservation service the customer is provided with an interface that facilitates search using some of the most crucial criteria, typically objective queries that are meant to reduce the choice set to a manageable size (e.g., number of visitors, length of stay, location, etc.).

The emergence of e-commerce systems or online reservation services brought forward the advantage of having an increased selection at the convenience of only a few clicks away. Both classic economics and psychology emphasize the benefits of a larger number of choices [34, 41, 42]. However, it also raised a number of important challenges as well. One such challenge is that the size of the choice set can be a cognitive load in the decision making process. Overchoice, or having too many choices, can be detrimental, leading to anxiety or depression [25, 45, 48]. Even though a larger number of choices is initially appealing, the consumers may feel less satisfied or convinced that they actually made the best decision available [26]. Recent studies even suggest an inverted U-shaped relationship between customer commitment and the number of available choices [46], with customers being more likely to find an item to their liking with the growing number of choices, but starting to have difficulties when multiple items fit their needs.

Furthermore, with continuous product diversification, user queries are also becoming even more refined, contributing to customer satisfaction and self-satisfaction being increasingly difficult to achieve [16, 44]. To address the customers' refined expectations, hotel reservation services provide faceted search functions, e.g., additional *filters* (e.g., *free breakfast* or *late check-out*), sets of objective options that serve as potential additional queries to reduce the choice set. Such

Authors' addresses: István Varga, istvan@megagon.ai, Megagon Labs, Tokyo, Japan, Recruit Co., Ltd., 7-3-5 Ginza Hulic GINZA7 Bld 3F Chuo-ku, Tokyo, Japan, 104-8227; Yuta Hayashibe, hayashibe@megagon.ai, Megagon Labs, Tokyo, Japan, Recruit Co., Ltd., 7-3-5 Ginza Hulic GINZA7 Bld 3F Chuo-ku, Tokyo, Japan, 104-8227.

CEUR-WS.org/Vol-2974/paper4.pdf

Copyright © 2021 for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

filters range from being just a handful of pre-defined, static options to sometimes even thousands of carefully curated ones over the course of several years [3]. However, continuously updating such a set of filters as a response to product diversification and customer expectation can be extremely costly. Moreover, navigating through a large set of filters can even become a burden, defeating its very own purpose [3].

In our work we provide a simple framework for automatically acquiring filters related to the hotels that match the customer's initial query, by identifying useful *mentions* from customer reviews. We especially focus on customer experiences that are potentially both relevant and interesting for other customers, while also having the capability of reducing the choice set in an intuitive and natural way.

The main contributions of this paper are the following:

(1) In order to address the overchoice problem, we present a simple clustering based approach to identify useful filters in a dynamic manner, from customer reviews.
(2) We define key concepts and strategies in scoring and ranking filters that are meaningful and natural for the customer.
(3) We present simple but efficient methods to implement filter scoring and ranking.
(4) We validate our proposal through a series of user experiments. We found that subjective, experience based filters that express quality judgements were especially useful for potential users to narrow down the search space.

The paper is organized as follows: in Section 2 we discuss the related work, followed by the definition of key concents of our approach in Section 3, data description in Sections 4 and details of our proposal in Section 5. We describe our experiments in Section 6, followed by discussions with future directions in Section 7 and the concluding remarks in Section 8.

2 RELATED WORK

Automatic facet generation is a closely related field to our task. Faceted search augments traditional search by presenting a set of attributes or filters that are grouped into facets, allowing customers to narrow down the search results [8, 10, 22, 37]. Manual curation and continuous updating of facets can be extremely costly [3], thus automatic methods to identify and rank filters have been proposed [18, 27]. Our work differs in three main aspects from automatic facet generation. Firstly, facet generation methods employ knowledge bases to maintain a well organized structure of facets [18, 27]. Our method does not employ structured knowledge bases, instead, we rely only on customer reviews. Secondly, faceted search typically targets objective filters to populate facets. Our work, besides objective filters, identifies subjective filters as well, crucial in expressing unique experiences that might be of value for new potential customers. Thirdly, compared to faceted search, our method puts an emphasys on addressing overchoice. The explanatory search nature of faceted search does address overchoice, but sometimes navigation through a large set of facets becomes a burden in itself, defeating its very own purpose [3]. Our method has the option of providing only a handful of potentially meaningful and unique filters that can reduce the choice set, without putting extra burden on the customer.

As another method to reduce information overload, customer review summarization is also related to our field [9, 11, 23, 38]. Our work mainly differs from review summarization in that we attempt to identify filters that are common across multiple items, whereas review summarization mainly focuses on identifying main characteristics of individual items.

Copyright © 2021 for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

Customer reviews have also been the target of sentiment analysis [4], aspect based opinion mining [43, 54], feature based ranking [53]. Similarly with review summarization, these methods focus on the reviews of single items, as opposed to identifying common, but meaningful characteristics across multiple items.

Also, customer reviews can be employed to generate recommendations [17, 47]. These methods rely on customer logs and information extracted from reviews to recommend items that are similar to previously liked ones. Our work does not imply the existence of previous customer logs.

Published work on query suggestion and recommendation has been prominently focused on the web domain [2, 24, 31], with recent focus on e-commerce product search [19] or news related content [12]. Typically these works employ knowledge bases [19, 24] or customer action logs [2, 24, 31] to suggest queries that are relevant to the original user query. Our method differs in two key aspects. First, our target is not to suggest similar queries or filters, instead, we attempt to provide useful filters that are not restricted to being related to the original user query. Second, we only utilize customer reviews, without the employment of knowledge bases or customer action logs.

Related to query recommendation is the field of query rewriting, the task which aims to reformulate customer queries into well-formed ones, in order to improve customer experience [50, 52]. It differs from our work in that query rewriting does not attempt to recommend new filters or queries to the customer.

Interestingness or uniqueness discovery, key concepts in our work, is another related field, with special focus on news articles [14, 28, 36], but definitions of uniqueness are often contain heavily domain dependent elements, such as article freshness [14, 28] or differences in events that occur before and after publication [36], not applicable in our domain. A more robust method is presented in [39], where authors define interestingness of articles as a combination of multiple features, such as topic relevancy, source reputation, writing style or freshness. The main difference from our work is that our target for uniqueness are simple sentences, rather than full articles.

On a note, the field of anomaly detection [6, 7] is also related to the concept of interestingness. However, unique or interesting in our context does not go as far as being abnormal, as in Hawkins's [20] definition of outlier[1].

3 KEY CONCEPTS

Our goal is to automatically identify filters that are characterized by: (1) being appealing to the customer; (2) having the potential of addressing the overchoice problem by reducing the choice set in an intuitive and natural way. We define a filter "appealing" as having the quality of being both *relevant* and *unique*. Also, we define a filter set "appealing" as being *diversified*, without too much emphasis on a single topic or aspect. Furthermore, to perform choice set reduction in an intuitive way, we introduce *size control* policies.

Relevance, uniqueness, diversity and size control are key concepts of our proposal. We employ size control policies and diversity rules as hard constraints to identify possible filters, while using relevance and uniqueness scores to determine the final filter ranking.

3.1 Relevance

Filters are required to hold enough decision power in order to be viable expressions of user intent. Pre-defined static filters of conventional hotel reservation platforms are good examples of high relevance (e.g., *breakfast included*, *late check-out*). We attempt to assign relevance scores to all possible filters. While relevance is highly subjective, we can

[1] "an observation that deviates so significantly from other observations as to arouse suspicion that it was generated by a different mechanism"

Copyright © 2021 for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

argue that all else being equal, certain filters satisfy a larger audience than others (e.g., *close to the city center* versus *bright pink curtains*). Detailed information about relevance scoring can be found in Section 5.2.1.

3.2 Uniqueness

Filters are also required to be representative of the choice set that matches the customer's original query, capturing characteristics that are unique within the search results. The motivation behind uniqueness is to identify options that are especially appealing within the hotels that already match the user query (e.g., *next to the city aquarium*), with the added potential to offer choices previously unknown by the customer (e.g., *private hot-spring*). Section 5.2.2 offers detailed description on uniqueness scoring.

3.3 Diversity

The importance of diversity and serendipidity is well recognized in the context of recommender systems [5, 30, 33]. Studies also point out that decision making factors are sometimes not even part of the original query [1]. As a result, we argue that, especially in cold start situations, a diversified set of filters that covers a wide range of topics is more suitable to accommodate customer needs, than filters biased towards one or more topics. More information about our approach in acquiring a diversified set of filters can be found in Section 5.1.

3.4 Size control

By definition, filters are designed to address overchoice and reduce the choice set, i.e., the number of matching hotels. We argue that the degree of the size reduction is also important. Providing highly appealing, but too generic or too specific filters might result in a too drastic or too shallow choice set reduction, leading to customer dissatisfaction. Instead, our strategy is to identify and provide only the filters that are guaranteed to result in a "just right" window of matching hotels, compared to the original number of matching hotels. Naturally, this implies that our filters are based on *availability*, i.e., filters that obey size control rules are guaranteed to reduce the choice set.

Intuitively, in practice this should provide a natural way in reducing the choice set, balancing between relevance and uniqueness. However, more often than not, relevance and uniqueness work against each other. Highly relevant filters are often not very unique (e.g., *free continental breakfast*), while highly unique filters may not be relevant to a large audience (e.g., *stay at a buddhist temple*). When the choice set is large, arguably it is more natural to select from more generic, thus high relevance, low uniqueness filters, with the preference shifting towards high uniqueness, low relevance filters with a decreasing choice set. With a large choice set, size control policies rule out filters that are not frequent enough, thus disregarding long-tail, but unique filters, with higher relevance ones gaining more prominence. With a decreasing choice set, long-tail, unique filters should gain more exposure at the expense of more generic, relevant filters. More information about size control policies can be found in Section 5.1.2.

4 HOTEL REVIEWS AS DATA SOURCE

As our data source we use over 20 million sentences extracted from hotel reviews, collected from one of the largest hotel booking sites in Japan[2]. The hotel review corpus contains the customer review texts and the location data associated to each hotel.

[2] jalan.net

Copyright © 2021 for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

Table 1. Predicate argument structures and their extracted cores

Core	Original predicate-argument structure
delicious food	very delicious food, really delicious food, all food is delicious, food is of course delicious, delicious food as advertised, more than delicious food
close to the station	hotel is close to the station, really close to the station, close to the station as mentioned, closest to the station, pretty close to the station
clean rooms	extremely clean rooms, very clean rooms, rooms clean as always, rooms are of course clean, thoroughly cleaned rooms, rooms cleaned to the last detail

4.1 Filter units

An underlying assumption of our method is that a user friendly filter extracted from customer reviews can be represented by a simple predicate-argument structure. To this end, we extracted over 20 million predicate-argument structures from our corpus by using JUMAN++ (v2.0.0-rc3), a Japanese morphological analyzer [49] and KNP++ (v0.9-21cc58c), a Japanese dependency and case structure analyzer [29]. We modified the case structure analyzer in order to retain only the core arguments of the predicates, discarding subtle nuances (e.g., modifiers, adverbs, adjectives, adverbial or adjective phrases, etc.) that are not relevant in the context of user friendly filters. To this end, we retained the arguments that mark the most essential Japanese grammatical cases: nominative, accusative, dative, instrumental, and the Japanese topic marker[3]. Table 1 illustrates some examples of core predicate argument structures together with their original form before the discarding process.

Some of the resulting predicate-argument structures were unrelated to hotels or had negative polarities, unsuitable for our filter policies. As a result, we employed a filtering method based on the automatic classification results of two BERT-based classifiers fine-tuned with an annotated corpus[4] [21] to identify non-negative predicate-argument structures relevant to the hotel or its services. As pre-trained model we used a BERT model trained on our hotel review corpus. For more information about our BERT model refer to Section 4.2. Finally, we retain core predicate-argument structures whose frequency is at least 5 in our corpus. As a result of the above processes, we retained 167,886 unique non-negative core predicate-argument structures.

4.2 Filter representation

To represent filters, we pre-trained a BERT [15] model on our review corpus. Here we followed the methodology described in [21]. The authors in [21] employ SentencePiece [32], an unsupervised text tokenizer which learns sentence units for a predetermined vocabulary size. We set the vocabulary size to 32,000. To train the BERT model, we used the parameter values officially distributed with BERTBase. We set the batch size to 512, the number of attention heads to 12, the number of layers to 12, and the number of hidden layers to 12. We trained the BERT model for 1,500,000 steps using TPUs.

To improve on BERT's embeddings, we employed the sentence embedding framework described in [40], using the triplet loss function to fine-tune our pre-trained model. The triplet-loss function requires a triplet of (anchor, positive, negative) sentences where the (anchor, positive) tuple is a positive pair, while the (anchor, negative) tuple is a negative pair. As input triplets for fine-tuning our pre-trained model, we employed a simple tf-idf based word2vec sentence

[3]We retained the arguments that were marked by the Japanese particles *ga*, *wo*, *ni*, *de* and *ha*.
[4]https://github.com/megagonlabs/jrte-corpus

Copyright © 2021 for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

representation described in [35] for each filter, randomly selecting 30,000 triplets whose (anchor, positive) pair had a cosine similarity larger than 0.85, and whose (anchor, negative) cosine similarity was smaller than 0.20.

5 PROPOSED METHOD

We developed machine learning based methods to identify and rank filters. Given a set of hotels that match an initial set of original user queries, we automatically extract the non-negative core predicate-argument structures described in Section 4.1 from the customer reviews of the matching hotels. These core predicate-argument structures will act as potential filters. First, using the sentence embedding representations of the filters, we apply a 2 staged hierarchical clustering method to group them into semantically similar clusters. In this step we employ policies to identify clusters that follow *size control* restrictions. Next, we score each cluster for *relevance* and *uniqueness* to determine the final filter class ranking. In this step we employ *diversity* policies. Also in this step we label the top ranked filter clusters. Below is a detailed description of each step.

5.1 2-stage clustering

5.1.1 Stage 1: main topic identification. In the first stage of clustering we attempt to group filters into main latent topics, e.g., `food`, `location`, `hot spring`, etc. The purpose is to serve *diversity* by identifying such latent topics, with the assumption being that filters from different clusters after stage 1 will roughly have different topics[5].

In order to identify the main topics, we employ Ward's agglomerative clustering method [51] with complete linkage and cosine similarity as metric. As feature representation for the filters, we used the sentence embeddings described in Section 4.2. Empirical results showed that a similarity threshold of 0.5 resulted in latent topic clusters with a good trade-off between inter-clusters homogeneity and intra-cluster variance.

We recognize that a carefully curated knowledge-driven approach may have the advantage in accurately associating pre-defined topics to filters. However, besides cost issues, our data-driven approach has the advantage of recognizing a potentially infinite number of intrinsic topics that would be difficult to manually acquire.

Also note that the purpose of the first stage is solely to associate filters with latent topics, thus this step can be performed beforehand, independently of user queries.

5.1.2 Stage 2: filter identification with size control. In the second clustering stage we identify filter clusters from each main topic from Section 5.1.1 whose size obey *size control* rules. The size of a filter cluster is defined as the total number of hotels that the members of the cluster are linked to. Size control is governed by two parameters, $lower\_bound$ and $upper\_bound$ that represent the lower bound percentage and upper bound percentage, respectively, in respect to the size of the original choice set.

To achieve this, for each topic output by the main topic identification step described in Section 5.1, we parse the hierarchical subtree of each topic by incrementally moving up in the cluster hierarchy. During this process we retain clusters that obey size control rules and stop where the linkage drops below a certain similarity threshold, empirically set to 0.7. Empirically, we set $lower\_bound$ and $upper\_bound$ to 30% and 70%, respectively.

5.2 Filter scoring

We score and rank filter clusters retrieved in the 2-stage clustering step based on their *relevance* and *uniqueness*. After ranking, we apply *diversity* rules and label the top K filter clusters as described below.

[5]Note that we do not attempt to label the resulting topics. Instead, we only attempt to identify filter groups that belong to the same latent topic.

Copyright © 2021 for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

Table 2. Relevance examples in the training data

Filter	Relevance score
rentable private open-air bath	5
delicious dinner	4
great view	3
television available in the rooms	2
good water pressure	1

5.2.1 Relevance score. In Section 3.1 we stressed the importance of discovering filters that are crucial enough in the decision making process. Determining relevance is a non-trivial task, since people's preferences are obviously not uniform. A filter that may be highly relevant for one customer, may be less relevant for another one (e.g., *rich choice of baby formula* or *free pair ticket to the city aquarium*), depending not only on personal preferences, but also on situations or even purpose of visit.

Since this is a cold start scenario, personalized relevance estimators based on customer action logs are not feasible. Instead, we define the relevance of a filter independently from the original user query, as the average of multiple subjective relevance scores.

We pre-computed filter relevance scores using a simple k-nearest neighbor classifier. For each filter we took the top $k = 5$ similar filters from our training data, and computed their average relevance, weighted by the similarity score, as shown in the below formula, where x denotes the target filters, x_i denote filters of the training data, $\text{relevance}_{\text{gold}}$ denotes gold relevance scores of the training data.

$$\text{relevance}(x) = \frac{\sum_{i=1}^{k} \text{cossim}(x, x_i) \times \text{relevance}_{\text{gold}}(x_i)}{\sum_{i=1}^{k} \text{cossim}(x, x_i)} \tag{1}$$

As similarity score we employed cosine similarity, computed on the sentence embeddings described in Section 4.2. We normalized the relevance score by scaling it to between 0 and 1.

As training data we randomly selected 8000 filters and asked 5 crowd workers to label their degree of relevance from 5 to 1[6]. The most relevant was labeled with 5, the least relevant being 1. Ungrammatical or semantically unsound filters were labeled with 0. We calculated pairwise inter-annotator agreement using Weighted Cohen's kappa [13]. Kappa values were between 0.24 and 0.56, representing fair to moderate agreement, underlying the highly subjective nature of the task.

For our classifier we used the filters that were judged as grammatically correct by at least 4 out of the 5 workers. For the grammatically correct filters we averaged the individual worker scores. We preferred to use truncated mean (e.g., ignoring the lowest and highest scores of the 5 workers) in order to counter for highly subjective relevance scores (e.g., *rich choice of baby formula*). Table 2 shows an excerpt of the filters and their respective averaged relevance scores.

We evaluated our relevance classifier on a held-out data of 1000 samples by calculating the precision on increasing error ranges. We achieved a precision of 60.40% when the error range between the estimated relevance and reference relevance was less or equal than 0.1 points, and 85.50% precision at 0.2 points error range, as shown in Figure 1.

[6]We manually selected the crowd workers based on their demographic information (i.e., gender, age range) to ensure diversity.

Copyright © 2021 for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

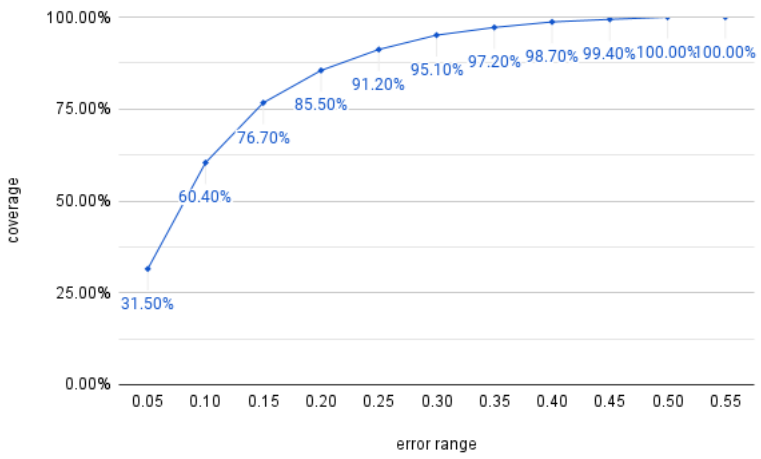

Fig. 1. Relevance evaluation: cumulative error range.

We compute relevance scores for filter clusters as the weighted average relevance of its member filters as shown in the below formula, where C_x denotes filter clusters, relevance(x) denotes filter relevance and freq(x) denotes the frequency of filter x in the target choice set.

$$\text{relevance}(C_x) = \frac{\sum_{x \in C_x} \text{freq}(x) \times \text{relevance}(x)}{\sum_{x \in C_x} \text{freq}(x)} \tag{2}$$

5.2.2 Uniqueness score. We define the uniqueness of a filter as the property of being important within a selected group of hotels. We employed term frequency–inverse document frequency (tf-idf) as uniqueness of each filter, where x denotes the filter, d denotes the reviews of a specific hotel.

$$\text{uniqueness}(x, d) = \text{tf}(x, d) \times \text{idf}(x) \tag{3}$$

Intuitively, a unique characteristic of a subset is more dominant in the subset than within the entire population. The sparse nature of the filters makes it unfeasible to handle them individually, thus for the purpose of computing tf-idf, we employed Ward's agglomerative clustering method [51] with complete linkage and cosine similarity as metric, with a similarity threshold of 0.7 to group together filters of similar semantic properties. As a result, we clustered the filters into 5178 clusters and we computed the tf-idf scores on the resulting clusters. Cluster members inherited the tf-idf values of their parent cluster.

Similarly to relevance scores, we computed filter cluster uniqueness score as the weighted average uniqueness of its member filters as shown in the below formula, where C_x denotes filter clusters, uniqueness(x, d) denotes filter uniqueness for filter x in hotel review set d, freq(x) denotes the frequency of filter x in the target choice set.

$$\text{uniqueness}(C_x, d) = \frac{\sum_{x \in C_x} \text{freq}(x) \times \text{uniqueness}(x, d)}{\sum_{x \in C_x} \text{freq}(x)} \tag{4}$$

5.2.3 Filter ranking and employing diversity rules. Filter cluster ranking is determined by multiplying the filter cluster's relevance and uniqueness scores, weighted by their respective weights (i.e., α and β for relevance and uniqueness,

Copyright © 2021 for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

respectively), as shown in the below formula. During preliminary empirical evaluations, we found that a reasonable value for α and β were 1 and 2, respectively.

$$\text{rank}(C_x) = (\alpha + \text{relevance}(C_x)) \times (\beta + \text{uniqueness}(C_x)) \tag{5}$$

To produce the final ranking, only the top $k_{diversity}$ filter clusters are retained for each main topic described in Section 5.1. We set $k_{diversity}$ to 1, e.g., we retain only the top filter from each topic[7].

5.2.4 Labeling filters. Finally we label the top K filter clusters that will be presented for the customer. We perform cluster labeling by choosing the most representative member, i.e., the member closest to the cluster centroid. We utilize cosine similarity to determine the representative member that will act as the label of the filter cluster.

6 EXPERIMENTS

We conducted a number of user experiments to evaluate: (1) the top $k_{filters}$ overall output and (2) the top $k_{filters}$ individual filter outputs of our `proposed` method, described in Section 5. Particularly, we compared the filters output by our `proposed` method against manually acquired filters. Also, we assessed the effect of uniqueness, relevance and diversity policies. To this end, we performed pairwise comparison against the following baseline models:

- `human`: a manually compiled filter list described below in Section 6.1.
- `relevant`: `proposed` without the uniqueness score, i.e., filter ranking is determined only by relevance.
- `unique`: `proposed` without relevance score, i.e., filter ranking is determined only by uniqueness.
- `non-diverse`: `proposed` without diversity policies, i.e., output is not restricted to the top $k_{diversity}$ filters for each main topic.

6.1 Manually compiled filters

To manually acquire filters in a simulated overchoice scenario, we randomly selected 10 *<original request, location>* query tuples with the following conditions:

- the resulting hotel hit count is at least 30 in our hotel review corpus;
- the total number of corresponding reviews[8]is at least 3000 in our hotel review corpus.

Table 3 shows the query tuples used in this process. From the resulting reviews we randomly selected 1000 reviews for each query tuple. Next, using the selected reviews, we asked 3 crowd workers to extract all simple short phrases which in their opinion contain meaningful information in further filtering the choice set. Such phrases were manually grouped by each worker into clusters that share the same meaning. Finally, all clusters were aggregated by a fourth crowd worker, registering the number of contributing workers and the number of hotels each cluster links to. As the final output, we considered filters that had a number of majority contributors (at least 2 out of 3), ranked in descending order by the number of corresponding hotels. Table 4 shows an example of manually acquired filters.

[7]One important note is that before applying diversity policies, first we remove filters that are semantically too similar to the original user queries. We perform this by using cosine similarity on their sentence embeddings described in Section 4.2. Empirically we set this similarity threshold to 0.8.
[8]We performed exact text match in retrieving hotels reviews that mention an original request.

Copyright © 2021 for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

Table 3. Query tuples used with manually compiled filters. (All examples are translations from Japanese.)

Query tuples
Delicious dinner. @Okinawa
Relaxing atmosphere. @Nagano
Very helpful staff. @Gunma
Atmosphere that makes you feel at home. @Iwate
Suitable for sightseeing. @Kyoto
Near downtown. @Aichi
Close to the sea. @Shizuoka
Fashionable rooms. @Hokkaido
Child friendly. @Chiba
Hotel with good access. @Akita

Table 4. Manually selected filters for *Very helpful staff.* @Gunma.

Filter	Hotel count	Worker count
Very satisfying food.	13	2
Clean rooms.	11	3
Open-air bath available.	11	3
Large rooms.	10	3
Suitable for families.	9	3
Delicious breakfast.	8	3
Suitable for couples.	8	2
Buffet style breakfast.	7	3
Cheap price.	6	2

6.2 Overall filter list evaluation

In the first set of experiments we performed pairwise evaluation against the target models described above. We used the same sets of 1000 random reviews of the query tuples utilized during the manual filter acquiring process, described in Section 6.1. For each query, we considered the top $k_{filters} = 5$ filters from each method's output.

We crowdsourced the pairwise evaluation, asking 300 workers using Yahoo!Japan's crowdsourcing service[9] to choose the filter list they find more suitable in further narrowing down the choice set. In randomized order, we showed the filter lists of the two methods (named lists A and B, respectively), asking the workers to choose exactly one of four choices:

- list A is more useful than list B
- list B is more useful than list A
- list A and list B are both equally useful
- neither of the lists are useful

We also asked the workers to motivate their choice for each filter list pair. After basic data quality check (i.e., removing workers that (1) did not provide any explanation for their choices, (2) always choose the same option, or (3) working

[9]https://crowdsourcing.yahoo.co.jp/

Copyright © 2021 for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

Table 5. Filter list evaluation: vote share difference in points for our proposal against the baseline models for each evaluation query tuple (statistically significant differences in **boldface**).

	Evaluation query tuple	vs human	vs non-diverse	vs unique	vs relevant
1	Close to the sea. @Shizuoka	**+92.38**	**+76.27**	-7.80	+2.29
2	Relaxing atmosphere. @Nagano	**+13.73**	**+74.00**	**+25.22**	-1.37
3	Very helpful staff. @Gunma	-8.94	**+67.16**	**+42.03**	**+31.65**
4	Atmosphere that makes you feel at home. @Iwate	**-18.00**	**+56.99**	**+36.73**	**-43.48**
5	Suitable for sightseeing. @Kyoto	**+65.88**	**+14.00**	**+15.15**	**+76.53**
6	Near downtown. @Aichi	**+29.76**	**+83.18**	**+20.34**	**+69.59**
7	Delicious dinner. @Okinawa	**-19.64**	**-22.08**	-5.00	**-18.18**
8	Fashionable rooms. @Hokkaido	**+67.90**	**+43.95**	-2.90	**+77.40**
9	Child friendly. @Chiba	-6.09	+9.83	**+13.13**	**-35.48**
10	Hotel with good access. @Akita	+6.85	**-16.14**	**-19.35**	**+40.64**

time was too short), we retained the results of 224 workers. Table 5 shows the overall list comparison results for each query tuple.

Against the manually generated human filters, proposed was considered to be the significantly better[10]overall choice with 5 out of the 10 evaluation query tuples. In 2 out of 10 cases the human output was considered to be significantly superior to the output of proposed. Overall, proposed was found to have a significant advantage over human with over 30 points difference.

Analysing the workers' comments, we observed that the overall output of proposed was overwhelmingly preferred over human when the filters offered very specific choices or experiences (e.g., *the parking lot is large, thus easy to park the car*, *delicious food with local ingredients*). At the same time, human was preferred by workers who value filters which proposed considered as relevant, but not unique enough to rank high (e.g., *large room*, *clean hotel*). It is also worth mentioning that when human outperformed proposed, the number of votes counted for either of the methods was actually smaller than in average, both lists being equally preferred or unpreferred by a large number of workers. We can also note that proposed was voted as the better overall choice by an overwhelming majority with a number of query tuples (e.g., *Close to the sea* @Shizuoka). The reason for this vote difference is that proposed managed to identify filters that are highly specific to the initial query tuple, and at the same time are also quite appealing to potential customers (e.g., *the splendid alphonsino was very delicious*), while human failed to identify such filters with a high enough frequency.

Against non-diverse as well, proposed exhibited a significant vote advantage (45.4 points), validating the effect of the diversity policies. However, non-diverse did perform better with some query tuples that are related to locations especially recognized or famous in relation with a specific main topic, which was captured and over-represented by the non-diverse method (e.g., nature topic in Akita).

Proposed also outperformed unique and relevant by over 6.3 and 29.2 points vote count difference, respectively, validating that both relevance and uniqueness contribute significantly to proposed. It is worth mentioning that relevant behaved very similarly to human against proposed, suggesting that the workers employed in acquiring the manual filters may have had preference towards more relevant, rather than unique filters.

[10]We checked for significance using binomial test of significance with p set to 0.05.

Copyright © 2021 for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

Table 6. Individual filter evaluation: vote share difference in points for our proposal against the baseline models for each evaluation query tuple (statistically significant differences in **boldface**)

	Evaluation query tuple	vs human	vs unique	vs relevant
1	Close to the sea. @Shizuoka	**+80.80**	**-7.53**	**+21.51**
2	Relaxing atmosphere. @Nagano	**+8.37**	-3.00	+3.10
3	Very helpful staff. @Gunma	+0.02	**+16.74**	-4.94
4	Atmosphere that makes you feel at home. @Iwate	+7.47	**+20.00**	-3.43
5	Suitable for sightseeing. @Kyoto	**+78.33**	**+24.46**	**+54.30**
6	Near downtown. @Aichi	+5.58	**+16.49**	**+52.68**
7	Delicious dinner. @Okinawa	**-33.88**	**+15.34**	**-31.85**
8	Fashonable rooms. @Hokkaido	**+20.90**	**+25.88**	**+29.64**
9	Child friendly. @Chiba	-5.58	-3.65	-3.21
10	Hotel with good access. @Akita	**+8.12**	**+9.46**	+5.35

6.3 Individual filter evaluation

In the second set of experiments we performed pairwise comparison of the individual filters output by proposed against the outputs of the target methods[11]. Here we attempt to counter the tendency some workers may have had in rejecting certain filter lists during list based evaluation described in Section 6.2, for the reason of containing unappealing filters. We used the same filters as with list based evaluation, merging and shuffling the filters into a single list. In case of duplicates, a single occurrence was retained. We crowdsourced the evaluation asking 300 workers using Yahoo!Japan's crowdsourcing service. As opposed to filter list evaluation, here workers were asked to award individual filters, by selecting from 1 up to at most 5 filters they find appealing in further reducing the choice set. We also asked the workers to motivate their choice for each selection set.

After basic data quality check (i.e., removing workers that (1) did not conform with the rule regarding the number of selected filters, or (2) working time was too short), we retained the results of 197 workers. For each query, we counted the total number of votes each method received. Filters that were duplicates counted for both methods. The results for each query tuple are summed up in Table 6.

Against the manually acquired human method, proposed had a significantly larger vote share with 5 out of the 10 evaluation queries, while being outperformed in only one case. We also found that proposed received more votes for filters that express experiences or quality judgements, positive opinions regarding a specific service or the hotel in general (e.g., *the open-air hot spring was excellent*, *the free breakfast was delicious*), while the human filters had the tendency to have more factual filters or presence/absence indicators (e.g., *open-air hot spring was available*, *free breakfast*). Similarly to overall filter list evaluation, human received numerous votes for filters that are highly relevant, but were ruled out by size constraints by proposed (e.g., *clean rooms* being too frequent, *washing machine available* being too rare).

Also similarly to filter list evaluation, proposed outperformed both relevant and unique. However, it must be noted that both relevant and unique received numerous votes with filters that are very relevant, but less unique (e.g., *clean rooms*, *excellent service*, *free wifi*) or unique, but arguably not relevant enough (e.g., *karaoke machine is available*, *dog run attached to the hotel*). These results suggest that while relevance and uniqueness both contribute to proposed, their importance is highly subjective.

[11] Here we skip pairwise comparison against non-diverse, since the target of this experiment are filters, as opposed to filter sets.

Copyright © 2021 for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

7 DISCUSSIONS AND FUTURE DIRECTIONS

We found that `proposed` was preferred by crowd-workers in two distinct scenarios.

Firstly, as observed during both list based and individual filter evaluations, workers preferred highly specific filters as opposed to more generic ones (e.g. *the splendid alphonsino was very delicious* versus *food was delicious*, *the parking lot is large, thus easy to park the car* versus *parking lot available*). This validates our assumption that the majority of potential customers are interested in very specific details in attempting to reach a decision.

Secondly, during the individual filter evaluation, we observed the worker's tendency in preferring filters that were formulated as an experience or quality judgement, rather than as a fact or presence/absence indicator, when both options were available. (e.g., *hot-bath was great* versus *hot bath is available*; *food was delicious* versus *food available at hotel*). This tendency was weak with topics in which experience itself may not be too relevant (e.g., the experience expressing *easy to park* was not overwhelmingly preferred over the factual *parking lot available*, arguably because the fact that parking is actually available is the crucial piece of information, rather than the ease of parking), but it was very prominent with topics such as `food`, `location` or other service related ones, where previous user's experiences and reviews are more valuable than the simple availability of that specific option.

We validated this assumption with a very simple experiment. We manually selected 30 (fact, experience) filter pairs and for each filter pair we asked 20 crowd-workers to choose the filter list they find more suitable in further narrowing down the choice set. We randomized the order of the two filters (named A and B, respectively), asking the workers to choose exactly one of four choices:

- filter A is more useful than filter B
- filter B is more useful than filter A
- filter A and filter B are both equally useful
- neither of the filters are useful

After basic data quality check (i.e., (1) always choose the same option, or (2) working time was too short), we retained the results of 19 workers. In 28 out of 30 pairs the experience based filter received the higher share of votes, although both fact and experience based ones received many votes. Most of these pairs had the topic of `location`, `food`, `hot spring` or some other type of hotel service. In case of a single pair the difference was only minimally in favor of the experience based filter, namely `parking` as topic. One pair was voted as being equally helpful, having received only a few votes for either fact or experience based filters, in the topic of `hotel amenities` (*amenities are available* versus *very basic amenity*).

This result suggests that the balance between fact and experience based filters is both subjective and possibly topic dependent. While customer reviews mainly offer intricate experience-like details, fact based filters still remain valuable. As current filters provided by conventional hotel reservation systems are largely fact based ones, undoubtedly intricate experience based filters extracted from customer reviews could add significant value. In deploying such a customer review based filter recommender, strategies need to be implemented to combine various types of filters from multiple sources of information. In the future we are planning to investigate how various sources of information can complement each other in providing meaningful filters.

8 CONCLUSIONS

In this paper we proposed a simple clustering based approach to address the overchoice problem in the hotel industry domain. We introduced size control and diversity policies, together with scoring verticals, in order to identify and score

Copyright © 2021 for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

filters that could reduce the search space in a natural and intuitive way. We validated our proposal through a series of user experiments where we also showed that the filters identified by our method were more useful than the manually acquired ones.

REFERENCES

[1] Susan Auty. 1992. Consumer choice and segmentation in the restaurant industry. *Service Industries Journal* 12, 3 (1992), 324–339.

[2] Ricardo Baeza-Yates, Carlos Hurtado, and Marcelo Mendoza. 2004. Query recommendation using query logs in search engines. In *International conference on extending database technology*. Springer, 588–596.

[3] Lucas Bernardi, Pablo Estevez, Matias Eidis, and Eqbal Osama. 2020. Recommending Accommodation Filters with Online Learning. (2020).

[4] Juergen Bross and Heiko Ehrig. 2013. Automatic construction of domain and aspect specific sentiment lexicons for customer review mining. In *Proceedings of the 22nd ACM international conference on Information & Knowledge Management*. 1077–1086.

[5] Pablo Castells, Neil J Hurley, and Saul Vargas. 2015. Novelty and diversity in recommender systems. In *Recommender systems handbook*. Springer, 881–918.

[6] Raghavendra Chalapathy and Sanjay Chawla. 2019. Deep learning for anomaly detection: A survey. *arXiv preprint arXiv:1901.03407* (2019).

[7] Varun Chandola, Arindam Banerjee, and Vipin Kumar. 2009. Anomaly detection: A survey. *ACM computing surveys (CSUR)* 41, 3 (2009), 1–58.

[8] Joseph Chee Chang, Nathan Hahn, Adam Perer, and Aniket Kittur. 2019. SearchLens: composing and capturing complex user interests for exploratory search. In *Proceedings of the 24th International Conference on Intelligent User Interfaces* (Marina del Ray, California) *(IUI '19)*. ACM, New York, NY, USA, 498–509. https://doi.org/10.1145/3301275.3302321

[9] Li Chen, Guanliang Chen, and Feng Wang. 2015. Recommender systems based on user reviews: the state of the art. *User Modeling and User-Adapted Interaction* 25, 2 (2015), 99–154. https://doi.org/10.1007/s11257-015-9155-5

[10] Li Chen and Feng Wang. 2017. Explaining recommendations based on feature sentiments in product reviews. In *Proceedings of the 22nd International Conference on Intelligent User Interfaces* (Limassol, Cyprus) *(IUI '17)*. Association for Computing Machinery, New York, NY, USA, 17–28. https://doi.org/10.1145/3025171.3025173

[11] Li Chen, Feng Wang, Luole Qi, and Fengfeng Liang. 2014. Experiment on sentiment embedded comparison interface. *Knowledge-Based Systems* 64 (2014), 44–58. https://doi.org/10.1016/j.knosys.2014.03.020

[12] Ilia Cherniavskii, Alexander Perelygin, and Russell Lee-Goldman. 2016. Suggested Keywords for Searching News-Related Content on Online Social Networks. US Patent App. 14/592,988.

[13] Jacob Cohen. 1968. Weighted kappa: nominal scale agreement with provision for scaled disagreement or partial credit. Psychology. *Bulletin, 70, 213* 220 (1968).

[14] Gianna M Del Corso, Antonio Gulli, and Francesco Romani. 2005. Ranking a stream of news. In *Proceedings of the 14th international conference on World Wide Web*. 97–106.

[15] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805* (2018).

[16] Kristin Diehl and Cait Poynor. 2010. Great expectations?! Assortment size, expectations, and satisfaction. *Journal of Marketing Research* 47, 2 (2010), 312–322.

[17] Ruihai Dong and Barry Smyth. 2016. From more-like-this to better-than-this: hotel recommendations from user generated reviews. In *Proceedings of the 2016 Conference on User Modeling Adaptation and Personalization*. 309–310.

[18] Leila Feddoul, Sirko Schindler, and Frank Löffler. 2019. Automatic facet generation and selection over knowledge graphs. In *International Conference on Semantic Systems*. Springer, Cham, 310–325.

[19] Mohammad Al Hasan, Nish Parikh, Gyanit Singh, and Neel Sundaresan. 2011. Query suggestion for e-commerce sites. In *Proceedings of the fourth ACM international conference on Web Search and Data Mining*. 765–774.

[20] Douglas M Hawkins. 1980. *Identification of outliers*. Vol. 11. Springer.

[21] Yuta Hayashibe. 2020. Japanese realistic textual entailment corpus. In *Proceedings of The 12th Language Resources and Evaluation Conference*. 6827–6834.

[22] Marti Hearst. 2006. Design recommendations for hierarchical faceted search interfaces. In *ACM SIGIR workshop on faceted search*. Seattle, WA, 1–5.

[23] Minqing Hu and Bing Liu. 2004. Mining and summarizing customer reviews. In *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*. 168–177.

[24] Zhipeng Huang, Bogdan Cautis, Reynold Cheng, Yudian Zheng, Nikos Mamoulis, and Jing Yan. 2018. Entity-based query recommendation for long-tail queries. *ACM Transactions on Knowledge Discovery from Data (TKDD)* 12, 6 (2018), 1–24.

[25] Sheena S Iyengar and Mark R Lepper. 2000. When choice is demotivating: Can one desire too much of a good thing? *Journal of personality and social psychology* 79, 6 (2000), 995.

[26] Sheena S Iyengar, Rachael E Wells, and Barry Schwartz. 2006. Doing better but feeling worse: Looking for the "best" job undermines satisfaction. *Psychological Science* 17, 2 (2006), 143–150.

Copyright © 2021 for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

[27] Zhengbao Jiang, Zhicheng Dou, and Ji-Rong Wen. 2017. Generating Query Facets Using Knowledge Bases. *IEEE Transactions on Knowledge and Data Engineering* 29, 2 (2017), 315–329. https://doi.org/10.1109/TKDE.2016.2623782

[28] Mozhgan Karimi, Dietmar Jannach, and Michael Jugovac. 2018. News recommender systems–Survey and roads ahead. *Information Processing & Management* 54, 6 (2018), 1203–1227.

[29] Daisuke Kawahara, Yuta Hayashibe, Hajime Morita, and Sadao Kurohashi. 2017. Automatically acquired lexical knowledge improves Japanese joint morphological and dependency analysis. In *Proceedings of the 15th International Conference on Parsing Technologies*. 1–10.

[30] Denis Kotkov, Shuaiqiang Wang, and Jari Veijalainen. 2016. A survey of serendipity in recommender systems. *Knowledge-Based Systems* 111 (2016), 180–192.

[31] Udo Kruschwitz, Deirdre Lungley, M-Dyaa Albakour, and Dawei Song. 2013. Deriving query suggestions for site search. *Journal of the American Society for Information Science and Technology* 64, 10 (2013), 1975–1994.

[32] Taku Kudo and John Richardson. 2018. Sentencepiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. *arXiv preprint arXiv:1808.06226* (2018).

[33] Matevž Kunaver and Tomaž Požrl. 2017. Diversity in recommender systems–A survey. *Knowledge-based systems* 123 (2017), 154–162.

[34] Ellen J Langer and Judith Rodin. 1976. The effects of choice and enhanced personal responsibility for the aged: A field experiment in an institutional setting. *Journal of personality and social psychology* 34, 2 (1976), 191.

[35] Joseph Lilleberg, Yun Zhu, and Yanqing Zhang. 2015. Support vector machines and Word2vec for text classification with semantic features. In *2015 IEEE 14th International Conference on Cognitive Informatics Cognitive Computing (ICCI*CC)*. 136–140. https://doi.org/10.1109/ICCI-CC.2015.7259377

[36] Sofus A Macskassy and Foster Provost. 2001. Intelligent information triage. In *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*. 318–326.

[37] Noemi Mauro, Liliana Ardissono, and Maurizio Lucenteforte. 2020. Faceted search of heterogeneous geographic information for dynamic map projection. *Information Processing & Management* 57, 4 (2020), 102257. https://doi.org/10.1016/j.ipm.2020.102257

[38] Samuel Pecar. 2018. Towards opinion summarization of customer reviews. In *Proceedings of ACL 2018, Student Research Workshop*. 1–8.

[39] Raymond K Pon, Alfonso F Cárdenas, David J Buttler, and Terence J Critchlow. 2007. iScore: Measuring the interestingness of articles in a limited user environment. In *2007 IEEE Symposium on Computational Intelligence and Data Mining*. IEEE, 354–361.

[40] Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084* (2019).

[41] Elena Reutskaja et al. 2009. *Experiments on the role of the number of alternatives in choice*. Universitat Pompeu Fabra.

[42] Richard M Ryan and Edward L Deci. 2000. Self-determination theory and the facilitation of intrinsic motivation, social development, and well-being. *American psychologist* 55, 1 (2000), 68.

[43] Amani K Samha, Yuefeng Li, and Jinglan Zhang. 2014. Aspect-based opinion extraction from customer reviews. *arXiv preprint arXiv:1404.1982* (2014).

[44] Benjamin Scheibehenne, Rainer Greifeneder, and Peter M Todd. 2010. Can there ever be too many options? A meta-analytic review of choice overload. *Journal of consumer research* 37, 3 (2010), 409–425.

[45] Barry Schwartz. 2004. The paradox of choice: Why more is less. Ecco New York.

[46] Avni M. Shah and George Wolford. 2007. Buying behavior as a function of parametric variation of number of choices. *Psychological Science -Cambridge-* 18, 5 (2007), 369.

[47] Koji Takuma, Junya Yamamoto, Sayaka Kamei, and Satoshi Fujita. 2016. A hotel recommendation system based on reviews: What do you attach importance to?. In *2016 Fourth International Symposium on Computing and Networking (CANDAR)*. IEEE, 710–712.

[48] Alvin Toffler. 1970. Future shock, 1970. *Sydney. Pan* (1970).

[49] Arseny Tolmachev, Daisuke Kawahara, and Sadao Kurohashi. 2018. Juman++: A morphological analysis toolkit for scriptio continua. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*. 54–59.

[50] Yaxuan Wang, Hanqing Lu, Yunwen Xu, Rahul Goutam, Yiwei Song, and Bing Yin. 2021. QUEEN: Neural Query Rewriting in E-commerce. (2021).

[51] Joe H Ward Jr. 1963. Hierarchical grouping to optimize an objective function. *Journal of the American statistical association* 58, 301 (1963), 236–244.

[52] Rong Xiao, Jianhui Ji, Baoliang Cui, Haihong Tang, Wenwu Ou, Yanghua Xiao, Jiwei Tan, and Xuan Ju. 2019. Weakly supervised co-training of query rewriting andsemantic matching for e-commerce. In *Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining*. 402–410.

[53] Kunpeng Zhang, Ramanathan Narayanan, and Alok N Choudhary. 2010. Voice of the Customers: Mining Online Customer Reviews for Product Feature-based Ranking. *WOSN* 10 (2010), 11–11.

[54] Jingbo Zhu, Huizhen Wang, Muhua Zhu, Benjamin K Tsou, and Matthew Ma. 2011. Aspect-based opinion polling from customer reviews. *IEEE Transactions on affective computing* 2, 1 (2011), 37–49.

Copyright © 2021 for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).