

# Needs-Centric Searching and Ranking Based on Customer Reviews

Thomas Lee, Simon Li, and Ran Wei  
*Operations and Information Management Department*  
*The Wharton School, University of Pennsylvania*  
{thomasy, simonx, ranwei}@wharton.upenn.edu

## Abstract

*Online retailers have associated the introduction of user-generated product reviews with increased customer sales and decreased product returns. For all of the perceived value conveyed by customer reviews, however, little effort has been directed towards leveraging user-generated content beyond a product-centric focus: customers first select a product in order to read from prior users of that product. In this paper, we integrate traditional information retrieval relevance ranking with database aggregation to model the knowledge within online product reviews and product descriptions. Customers search the knowledgebase of reviews by querying on specific needs or interests. The result is a customized ranking of products, a recommendation list, that is based upon and explained by the text of reviews written by prior users expressing similar needs or interests. We evaluate the approach using a knowledgebase of online reviews from Epinions.com and compare results to expert recommendations from Consumer Reports.*

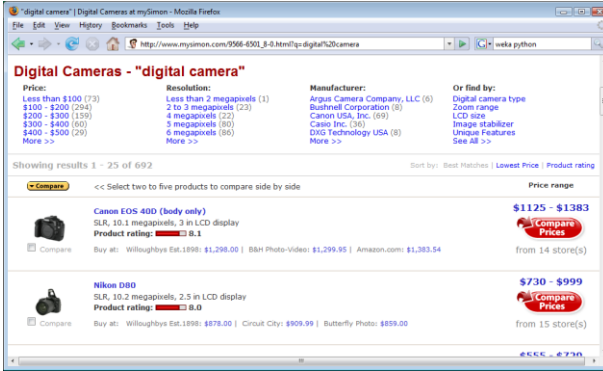
## 1. Introduction

In the classic approach to marketing and retailing, a customer approaches the shopping exercise seeking to satisfy a particular need [8]. For customers who already know what specific make and model will best meet their need, online shopping is an attractive option. Online retailers provide wider product variety, larger inventories, and lower search costs [3]. For customers with no specific product in mind, however, online retailers have found it difficult to replace the role of a live, knowledgeable sales clerk to help customers relate needs to products. Indeed many online retailers have installed instant messaging and online chat software to enable customers to interact directly with knowledgeable staff.

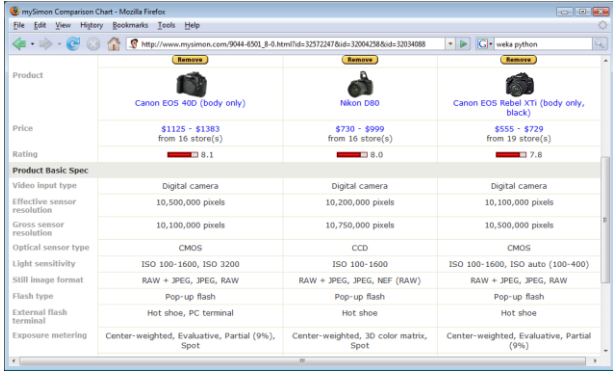
As depicted in a typical online retailing interface (see Figure 1a), customers have the opportunity to browse and rank the available inventory by a myriad of options including price, brand, physical dimensions, etc. In some online retailers and for certain products, customers are even able to perform side-by-side product comparisons by physical attribute (see Figure 1b). For customers who may have no prior purchase experience in a particular product category, however, attribute-by-attribute product comparisons or multiple database-enabled views of inventory do not help. Separating microwaves by wattage or digital cameras by resolution makes no difference for customers have no idea what wattage or resolution even are. However, customers do know what their "needs" are. The challenge for online retailers is to enable customers to search and sort the product-space in a need-centric rather than a product-centric manner.

Whether it is a knowledge sales clerk or a well-informed friend, the knowledge, of what products are most appropriate for a particular need, derives from experience. User-generated content in the form of customer reviews constitute an archive of such experience. Many retailers do enable customers to sort products by average customer rating; but sorting strictly by rating does not account for the many different needs that might motivate that rating. On a scale of 1 to 5, the same product could receive a low rating of 1 and a high rating of 5 simply because two different users evaluated the same product from the perspective of two different needs. A microwave oven that is perfect for a single student living in an apartment might prove ill-suited for a family of five living in a four bedroom house.

Perhaps in light of this inability to distinguish needs, retailers today typically provide only product-centric access to customer reviews. Customers first select a product. After selecting a product, customers can read the associated reviews for that one product to help make a purchase decision. Simply put, product-centric reviews allow users to learn about one product from the prior users of that product. By combining the



a. Product-centric browsing



b. Attribute-by-attribute comparisons

Figure 1. Online price comparison shopping from MySimon.com

reviews from many different products at once, customers and retailers can learn category-level characteristics, such as which products better match specific needs.

In this paper, we aim to enable online customers to sort and rank products based upon needs. We integrate information retrieval (IR) and database (DB) methods to query the experiential knowledge base of user-generated content in online customer reviews. A customer inputs a free-text query IR query to find reviews written by others expressing a similar interest. Focusing only on those products with at least one relevant review, results are grouped by product. For each product, the review ratings are weighted and aggregated to reflect how well a product matches the initial search terms. The resulting product rankings are aligned with the specific needs as mentioned by the customer. Furthermore, customer can read the underlying rankings as a means for explaining the specific rankings. This paper contains two explicit contributions. First, we present needs-centric searching of customer reviews as a means of producing customized product rankings that are explained by the text of the underlying reviews. Second, we propose relevance-weighted ratings aggregation as an approach to ranking in DB+IR integration.

Section 2.0 defines our model and Section 3.0 is our approach to querying and ranking. Section 4.0 is a description of initial experiments to evaluate the approach. Section 5.0 is a discussion of the experimental results. Section 6.0 reviews the related literature before the Section 7.0 conclusions.

## 2. Model

Colloquially, we may think of a product review as a numerical rating that summarizes one previous customer's assessment of how well the reviewed product satisfies the customer's needs as expressed in

the free-text portion of the review. More formally, let  $\mathcal{R}$  be the set of reviews for a product category  $\mathcal{C} = \langle \wp, S \rangle$ .  $\wp$  is the set of products in the category and  $S$  is the schema of product attributes that define the category. The product  $P \in \wp$  is defined by a disjoint subset of reviews  $R \subseteq \mathcal{R}$  and a set of values  $A$  that is drawn from the relation defined by the corresponding product schema  $S$ . Finally, a review  $r \in R$  includes a scalar rating  $s$  (commonly on a scale of 1 to 5), the identifier for a reviewer  $u$ , and a vector of text strings (words)  $v$  representing the bag of words that constitute the textual review.

For example, the product category of microwave ovens consists of all available microwave oven makes and models. Every microwave oven is defined by a set of physical attributes including brand, price, color, external dimensions, internal volume, and cooking power. A specific make or model is defined by the product of its specific attribute values.

Although more sophisticated vector indices are possible [4], for simplicity, we assume that an individual review vector  $r.v$  is encoded using the familiar, TF-IDF (Term-Frequency Inverse-Document-Frequency) word weighting scheme. The weight of a word  $w$  in  $r.v$  follows the general framework:

$$\log(1 + x_{wr}) * \log\left(\frac{|\mathcal{R}|}{\sum_{r \in \mathcal{R}} I(x_{wr} > 0)}\right)$$

The TF element is defined in terms of  $x_{wr}$ , the frequency with which word  $w$  appears in  $r$ . Intuitively, if the word  $w$  appears more frequently in  $r$ , there is a greater likelihood that a person interested in  $w$  will be interested in  $r$ . The term is log transformed to reflect the diminishing returns to relevance as word occurrences increase.

IDF represents the fraction of all reviews in collection  $\mathcal{R}$  containing at least one occurrence of the word  $w$ . Intuitively, words appearing in a large fraction of documents are less useful for distinguishing between documents.  $I(x_{wr} > 0)$  is a function that

takes the value 1 when word  $w$  appears at least once in document  $r$ .  $\sum_{r \in \mathcal{R}} I(x_{wr} > 0)$  counts the number of reviews in the corpus that include word  $w$ .

### 3. Querying the model

Every review is a document. Customers can search the document knowledgebase by using keywords that return relevant reviews. However, customers searching the review space ultimately seek to learn about products that match their needs and interests; a traditional IR search orders results by relevance, intermingling reviews from all products. What we desire is to rank products (rather than rank reviews) according to their relevance to a customer's needs or interests.

#### 3.1. Ranking by average rating

Most online retailers already support inventory browsing according to average customer rating. Each product review is summarized by a numerical product rating. Grouping reviews by product, we can rank products by the average rating of the corresponding reviews. However, simple grouping and aggregation overlooks the reality that individual reviews with a particular set of needs or interests in mind. Moreover, simple averaging is biased towards fewer, high-scoring reviews. A product with one high rating would rank above a product with ten highly-rated reviews and one low rating.

**3.1.1. Relevant reviews.** Intuitively, users with a specific need are only interested in reviews expressing a similar need. In the traditional vector-space model (VSM) of information retrieval, a query  $Q$  is modeled as a vector of weighted word counts to parallel the vector representation of reviews. While there are many alternatives for comparing review vectors, for simplicity, we use the basic cosine measure [4] of the angular distance between two vectors. Vectors are normalized to unit length.

$$\cos(r_1, r_2) = \frac{r_1 \cdot r_2}{|r_1| |r_2|}$$

Individual review vectors  $r \in \mathcal{R}$  are now comparable with respect to their distance from the query vector  $Q$ . Therefore, we can limit the ratings aggregation to only those reviews satisfying a minimum distance (relevance) threshold  $t$  where the threshold reduces to 0 in the case of Boolean querying. If  $r.s$  is the rating and  $r.v$  the vector for review  $r$ , we can write this in algebraic terms [5] as:

$$Y_{P,AVG(r.s)} \left( \sigma_{\cos(Q,r.v) > t}(\mathcal{R}) \right)$$

**3.1.2. Bayesian updating.** Limiting the average to only relevant reviews does not account for the problem of a bias against products with more reviews. Intuitively, rankings should reflect the preponderance of evidence. Products with more highly rated reviews should outrank products with fewer. Alternatively, the weight of one negative (positive) review should be scaled relative to the total number of relevant reviews. Consider the product  $P$  with  $n$  relevant reviews  $P.R = \{r_1 \dots r_n\}$  that have respective ratings  $r_i.s$ . Borrowing from the IMDB (Internet Movie Data Base) formula for Bayesian updating of movie ratings, we begin by assuming that product ratings are normally distributed with known variance. The updated product rating is then modeled as calculating the mean of the posterior distribution for an unknown variable distributed normally with mean  $\theta$  [6].

$$p(\theta | r_1.s \dots r_n.s) = p(\theta | \bar{r}.s) = N(\theta | \mu_n, \tau_n^2)$$

As with IMDB, if we assume that unrated products are judged relative to a neutral rating (e.g. 3 on a scale of 1-poor to 5-excellent), then the prior distribution  $N(\mu_0, \tau_0^2)$  is the neutral rating with  $\mu_0 = 3$ . If we further assume a constant standard deviation  $\tau_0^2 = \sigma^2 = 1$ , then the prior distribution is simply treated as an additional posterior observation; we can calculate the posterior distribution parameter directly as [6]:

$$\mu_n = \frac{\mu_0 + n\bar{r}.s}{1 + n}$$

Intuitively, aggregated ratings are calculated as deviations from the neutral rating. The accumulation of reviews draws the aggregate away from the neutral towards the true rating. As a baseline for needs-centric product rankings, we use averages that update a neutral prior where each review is an additional observation. The average is Bayes adjusted using only reviews that exceed a minimum relevance to the customer's needs.

#### 3.2. Ranking by weighted average

The baseline measure for average rating considers only relevant reviews. However, not all relevant reviews are the same. We model a review rating as a numerical summary of the corresponding text. However, as circled in Figure 2, a review might mention several different needs or it might mention only one. If a review mentions only one need, then the corresponding product rating is more likely to fully reflect the product's performance with respect to that need. By contrast, for a review that mentions several needs, the overall rating is more difficult to interpret. The challenge is to therefore determine how much a particular need contributes to the overall product rating. While there have been some attempts to

automatically identify user needs within the text of reviews [10], the problem is not yet solved.

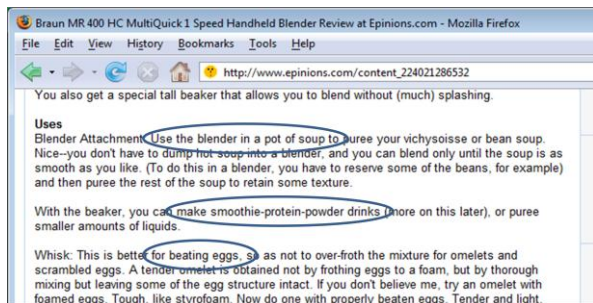


Figure 2. Uses in the text of an Epinions.com review

Intuitively, reviews with higher relevance are more useful to the user; the ratings of more relevant reviews should correspondingly carry more weight. Relevance, as defined by the cosine distance metric with TF\*IDF weighting, is based upon term frequency. Because the measure is normalized for document length, it follows that relevance is a rough indicator of how important the query (a customer need) is within each corresponding review.

More significantly, absolute measures of relevance are unimportant. Our objective is to aggregate ratings from the reviews for one product. We are therefore only interested in the *relative* importance of the query between reviews for the same product. For example, the IDF factor discounts query terms that are common across many reviews; but the IDF contribution is constant for all reviews. To emphasize the relative importance, for a single product  $P$  with  $n$  relevant reviews, we first normalize relevance.

$$m = \sum_n \cos(Q, r_i)$$

We may then generate a weighted average as follows:

$$\mu_n = \frac{\mu_0 + \frac{1}{m} \sum_n \cos(Q, r_i) r_i \cdot s}{1 + n}$$

Notice that the baseline average rating is simply a special case of the weighted average using a Boolean measure of relevance.

## 4. Evaluation

To evaluate our approach, we first consider the need for a needs-centric approach. If the product-centric approach is adequate for current needs, navigating the knowledgebase of reviews becomes unimportant. Second, we attempt to validate DB+IR ratings aggregation as an approach to needs-centric searching and querying of reviews.

We have gathered data for preliminary analysis on two categories of consumer electronic appliances,

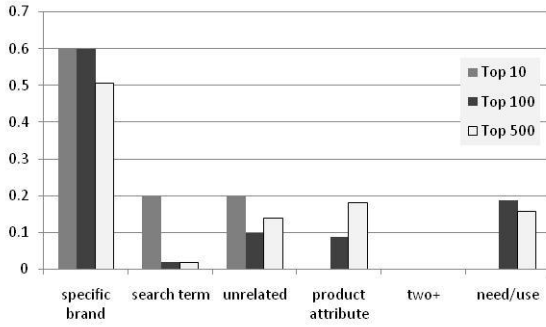
blenders and microwave ovens. The knowledgebase consists of 1000+ customer reviews covering 195 blenders and 1000+ customer reviews covering 259 microwave ovens from Epinions.com. Each review is stored as an individual tuple in a MySQL database. ReviewerID and productID form a composite key; for simplicity, word weights are also generated stored as relations to facilitate comparisons between weighted and un-weighted aggregations.

Shopping.com provides traditional product rankings based on average customer-rating (using reviews from Epinions.com) and by popularity. To contrast the product-centric approach, we use needs-centric rankings produced by third-party expert reviewers. We collect the 100 most frequently-used search terms in each product category as reported by MySimon.com. From the list of frequent searches, we select terms that reflect customer needs or interests. We then find third-party expert rankings that correspond to one or more of these terms. While needs-based expert rankings are not common, we did find Consumer Reports rankings in two consumer appliances that corresponded to needs-based searches. For blenders, we found Consumer Reports rankings that matched the search "frozen icy drinks" as reported in 2004 and 2007. For microwave ovens, we used Consumer Reports rankings from 2006 and 2007 that explicitly rated "over the range" microwaves.

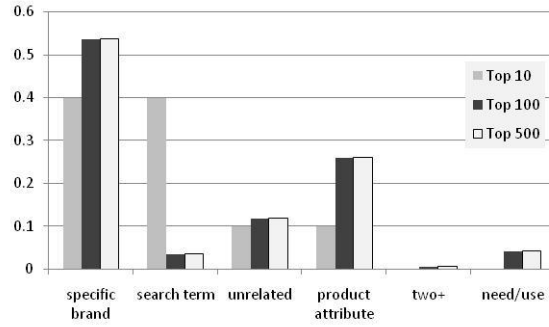
### 4.1. The need for needs

To consider the need for needs-centric approaches, two different coders independently classified the top search strings used by customers searching MySimon.com within the Blender category. Search strings classified as a "specific brand" meant that the customer arrived at the shopping site with a specific manufacturer make and/or model in mind. Search strings classified as "search term" meant that customers simply used some variant of the product category including singular or plural forms of the category name as well as misspellings. "Unrelated" search terms are random search strings. Customers could also search on an explicit product "attribute" such as color or style, could search on "two or more" other classes, or on a "need/use" for the product. The same process was repeated for Microwave ovens. To understand the relative importance of needs-centric approaches, the distribution of search terms was considered for the top 10, top 100, and top 500 search terms in each category. Results are reported in Figure 3.

### 4.2. Ranking products



a. "blender" product category



b. "microwave" product category

Figure 3. Classifying different search terms by product category

Our approach reduces the problem of needs-centric navigation to one of generating needs-centric product rankings. We compare product rankings that use a relevance-based average rating (RA) and relevance-weighted-average rating (RWA) to product rankings based upon other, more traditional measures. For example, we expect that needs-centric product rankings will differ from the traditional product rankings, which are based upon overall average customer rating or product sales. If RA or RWA rankings always match traditional rankings, then the needs-based approach is unnecessary.

Conversely, to the degree that we believe that the review knowledgebase reflects the wisdom of consensus opinion, we expect the needs-centric rankings to more closely align with third-party expert rankings. Specifically, that RA or RWA would align with Consumer Reports buying guides (CR) that rank products for the same customer needs or interests.

We used MySQL to calculate a relevance score based upon the cosine distance for every review, group by product, and rank by the average rating (RA) or weighted average (RWA). As a refinement to the basic TF-IDF vector indices, we used a variant of pivoted document length normalization as implemented in MySQL's full-text retrieval rather than the standard L2 normalization [7]. Rankings are compared using Spearman's Rank Correlation Coefficient ( $\rho$ ). For two different rankings over the same set of data,  $\rho$  is based upon the differences in rank for each product. If  $n$  is the number of items ranked and  $i$  indexes over each item,  $d_i$  measures the difference between the rankings for item  $i$ . Spearman's Rank Correlation Coefficient is then defined as:

$$\rho = 1 - \frac{6 \sum_i d_i^2}{n(n^2 - 1)}$$

$\rho$  varies between -1 and 1 where a coefficient of 1 indicates perfect correlation and -1 indicates a perfect inverse correlation. Because it is only possible to compare rankings between two sources if the products

being ranked are the same, we first identify the intersection, the set of products ranked by both sources. Thus, we calculate  $\rho$  for the relative ranking of products in the intersection between two approaches. It is possible for products to have identical weighted ratings and there are possible adjustments for ties. However, because our context is in consumer search and consumers are known to read in list order, we use the physical list order as a total order on products. Ties are sorted first by descending number of reviews and second in alphabetical order by product name.

We first retrieved Epinions blender reviews using the phrase "frozen icy drinks." We deliberately selected this "need" because it best matches existing Consumer Reports blender rankings from 2004 and 2007. We recorded  $\rho$  for the pairwise comparisons of blender rankings based upon the RA and RWA to each of the respective Consumer Reports "expert rankings" (CR04 and CR07). In addition, we considered two traditional product-centric rankings. Shopping.com ranks blenders based upon popularity (Pop) and based upon average customer rating (Rating). All pairwise comparisons are included in Table 1.

Likewise, we retrieved Epinions microwave oven reviews with the phrase "over the range" to match explicit Consumer Reports rankings for "over the range" microwaves in 2006 and 2007. Table 2 reports  $\rho$  for the pairwise comparisons of microwave ovens. Again, RA and RWA aggregated ratings are analyzed relative to each of the respective Consumer Reports "expert rankings" (CR06 and CR07) as well as to the Shopping.com product-centric rankings based upon popularity (Pop) and rating (Rating). The numbers in parentheses indicate the number of elements  $n$  in the ranking or intersection between two rankings. \* and \*\* indicate significance at .05 and .005 respectively using the Pearson Type II approximation [26].

**Table 1.  $\rho(n)$  for Blender Comparisons**

	Pop	Rat	CR04	CR07
Pop(58)	1			
Rating(88)	0.12(46)	1		
CR04(16)	.26(12)	.35(10)	1	
CR07(5)	.5(5)	.5(2)	.5(3)	1
RA(136)	.12(46)	.62(88)**	-.42(13)	.3(5)
RWA(136)	.08(47)	.62(68)**	-.3(13)	.3(5)
Baby(102)	.02(36)	.60(50)**	-.29(7)	1.0(3)

**Table 2.  $\rho(n)$  for Microwave Comparisons**

	Pop	Rat	CR06	CR07
Pop(90)	1			
Rating(90)	.18(43)	1		
CR06(26)	.38(8)	.15(12)	1	
CR07(32)	.39(11)	.15(10)	.63(17)**	1
RA(105)	-.15(21)	.61(31)**	.33(38)*	.04(10)
RWA(105)	-.24(19)	.65(27)**	.36(35)*	-.36(10)
Dorm(34)	-.07(7)	.5(5)	.5(5)	(0)

## 5. Discussion

Our evaluation is rooted in the comparison of different methods for ranking products. Because two lists might rank tens to hundreds of items while ranking only a few items in common, we compute the correlation only for products ranked in both lists; but there is no way of knowing whether products that were unranked by experts would have reinforced or contradicted the observed trends. Moreover, calculating the intersection is imprecise. If only one instance of a model appears in each list, we equate the two instances even if they differ in one or more attributes that are subjectively deemed irrelevant (e.g. same model but different color). Where multiple versions of a model appear, we use the closest match. In addition, because the intersections are sometimes quite small (2 or 3 products in the extreme), generating statistically significant results is challenging. Sample size is further complicated by the need for search terms that both reflect customer needs and have a comparative expert ranking (e.g. a Consumer Reports guide). As a consequence, although rank statistics can indicate general trends, actual user studies are needed to evaluate the utility of a needs-based approach.

### 5.1. The need for needs

Based upon the search terms collected by MySimon.com, we might classify anywhere from 5 to 15% of the searches as need or interest-based. Moreover, classifying search terms can prove ambiguous. For example, needs or uses that customers might have for a product (e.g. making frozen drinks) might also be captured in product attributes (e.g. a brand name like "Margaritaville" blenders). To that end, product-centric browsing and needs-centric browsing may overlap. Additionally, the comparatively low percentage of need and interest-based terms could simply reflect selection bias. For example, online consumers in general, or price comparison shoppers in particular, may be conditioned by experience to think and search in a product-centric way.

### 5.2. Rankings validation

There is anecdotal evidence that, after introducing reviews, sales of highly rated items increase while sales of lower rated items decrease[13]. However, in comparing the rankings, the first observation is that, perhaps surprisingly, popularity (by sales) and ratings are uncorrelated. Conversely, as we might have expected, there is strong agreement between different years of Consumer Reports (expert) rankings of microwave ovens. Lack of a similar correlation in Consumer Reports blender rankings may be due to the small intersection among products rated by the experts between 2004 and 2007 (CR04 and CR07). The lack of any clear correlation between the expert (CR) rankings and the shopping.com general rankings (pop, rating) is consistent with our initial intuitions: general purpose rankings do not necessarily reflect a product's fitness for distinct customer needs. Finally, we see that both the RA and RWA rankings differ significantly from the rankings based upon popularity (pop).

Counter to our expectations, however, the needs-based rankings (RA and RWA) are significantly correlated with rankings based strictly upon overall averages (rating). There are at least three possible interpretations. First, it may be possible that review

ratings, aggregated by product, are too homogeneous to distinguish between needs. If the variance among ratings within individual products is consistently low, sorting by needs may be unable to elicit clear distinctions between the alignment of products and needs. However, prior research on reviews in other domains does suggest a bi-modal distribution to ratings across products [3]. For microwave ovens, of the 221 products (total of 360) that were reviewed by more than one consumer (so variance is not zero), the variance in ratings that vary from 0 to 5 is 1.29. 129 out of 202 blenders were reviewed by more than one customer; of those 129, the variance was 1.30. This suggests that ratings are not too homogeneous.

Second, "needs" may simply be a poor way of differentiating between some types of products or services. A product might legitimately have only one (or a few) needs which are universal (e.g. shared by all users). Alternatively several needs may share similar attributes or latent characteristics. To informally test universality, we see that among 202 blenders, 136 (or more than half) have at least one review matching our blender "need". The same ratio holds for microwave ovens: 105 out of 190 different microwave ovens have at least one review matching our microwave "need." However, 489 of 1173 total blender reviews matched our search string but only 190 out of 1340 total microwave reviews matched our search terms. If the total number of relevant reviews and the number of products with relevant reviews is sufficiently large, searching by need reduces to searching on all products to begin with. Our blender "need" appears more universal than our microwave "need."

As a second test of universality, we generated a second needs-centric ranking using RWA on a different "need." For blenders, we searched on "baby food" (baby) and for microwave ovens, we searched on "dorm or apartment" (dorm). Comparisons are reported in the last row of Table 1 and 2 respectively. Although there are no expert rankings for direct comparison, we can at least establish the difference between needs-based rankings and ranking by overall rating averages. Consistent with our informal indicators of universality, rankings for "baby food" and for "frozen icy drinks" are similar. Intuitively, based upon underlying product attributes, blenders that are good for "frozen icy drinks" seem similar to "baby food." By contrast, the "dorm or apartment" rankings compare quite differently to the search for "over the range" microwaves.

Granularity is a third possible explanation for the high correlation between need-based rankings and shopping.com's rankings based on overall rating. Correlation coefficients may be too coarse as a metric for detailing differences in rankings. Evidence from

the search community suggests that users tend to look only at the first few search results. Consequently, a correlation among long lists of products may not reflect the true voice of the customer. Differences in ranking among the first ten products may be significant from the perspective of customer needs.

Finally, we consider the comparisons between RA and RWA rankings and the Consumer Reports expert rankings. Among blenders, there is very little relationship, counter to our expectations. However, given the strong correlation between our relevance-based rankings and the shopping.com rankings based upon overall average rating, perhaps the difference between relevance-based rankings and expert rankings is explained by transitivity. For microwave ovens, there is a statistically significant correlation between both RA and RWA and the CR06 rankings. However, there is virtually no correlation whatsoever between relevance-based rankings and CR07. The correlation in CR06 is calculated over nearly four times as many products as in CR07. Calculating the coefficients using only products in both CR06 and CR07 may account for the difference in the comparisons.

More generally, the timeliness of reviews relative to the expert rankings may account for some of the difference. In addition, it may also be the case that the expert reviewers simply do not accurately reflect the "voice of the consumer." In this respect, a more comprehensive evaluation will ultimately require both a larger number of reviews and actual user studies.

Relying upon reviews to capture the "voice of the consumer" raises several obvious objections. As already noted, our current product categories have only a limited number of reviews; the sensitivity of the approach to review counts must be assessed. We are currently in the process of gathering reviews for products with larger review counts while searching for complementary needs-based expert rankings to provide a point of comparison.

The approach is also vulnerable to false, biased, and negative reviews. Conversely, precisely because this approach is review based, users can immediately view the reviews upon which a ranking is based and decide for themselves. Second, more sophisticated search techniques can account for negation in text such as "X is not Y." Third, although we use Bayesian weighting to update reviews, the technique is still sensitive to outliers e.g. products with one review versus products with multiple reviews. We can weight the neutral rating by simulating an additional number ( $m$ ) standard reviews with neutral ratings to prevent any one review from over-influencing products with more reviews.

## 6. Related work

Product ranking is similar in spirit to the idea behind product recommendations. Recommending based upon needs is not new. At a high level, Orman [18] defines Customer Support Systems (CSS) in the same way that O’Keefe and McEachern [17] use the term Web-based Customer Decision Support Systems (CDSSs). The common principle is to adopt the vocabulary and perspective of the consumer. Where traditional recommender systems are based upon some combination of prior purchase histories and product attributes [2], context-based recommenders [1] express similar elements to needs-based navigation and ranking. For example, the search for a movie might be qualified by a "context" or need such as occasion (a romantic date versus a family outing) or time (matinee versus evening). However, while prior work proscribes how needs-based systems should perform, it does not address how or where to acquire the necessary data. In this work, we propose to exploit the text within customer reviews to discover the knowledge relating needs to products.

The text-mining literature treats customer reviews as a knowledgebase for ranking products [12, 23]. However, these review-based rankings are product-centric, based upon product attributes that are automatically learned from reviews [12, 16]. These product-centric rankings do not adjust for consumers who may have different needs or interests. Acknowledging the needs-centric view, there are a few efforts to learn needs [10] as well as attributes. While this may help users learn about a product category in the language of the consumer, there is no subsequent attempt to rank products [9].

In a few domain-specific cases, researchers have used utility-based models to relate user needs to attributes and attributes to products. Urban and Hauser [25] implemented AutoChoiceAdvisor as a tool for the auto industry. Their data came from customer purchase data, product manufacturing details, and survey results from a third-party, automotive marketing services company. Randall et al. [20] developed a needs-based PC configuration tool that draws on expert knowledge to relate attributes to products via multi-attribute utility models. Stolze and Stroble [24] develop a similar tool to recommend digital cameras.

Our work treats customer reviews as a knowledge base about product categories. We propose a DB+IR integration strategy for supporting user search of the knowledgebase. Queries, written in the vocabulary of the consumer, are matched to reviews written by other customers. Search results are aggregated into product

rankings. Customers can thus rank based upon their own vocabulary for describing needs and interests.

While this paper develops a DB+IR approach for reviews, DB+IR integration has a long history. The naïve approach applies IR indexing to DB attributes stored as text. DB selection conditions restrict the text entries that are evaluated for IR relevance [14]. Advanced techniques treat IR relevance as uncertainty and adapt probabilistic DB frameworks to DB+IR integration. A subsequent body of literature develops functions to calculate "aggregates" and their corresponding uncertainty or uncertainty distributions [15, 19, 21, 22]. "Grouping" of uncertain data is treated as a data clustering problem over fuzzy data [11].

The approach in this paper groups product reviews by product name (e.g. DB attributes) rather than clustering on fuzzy data (e.g. IR relevance). Our objective for grouping is to rank the resulting groups. Thus, we extend a traditional vector-space search model to facilitate grouping and weighted-aggregation of product review scores. We compare the ranked results of different aggregation approaches rather than develop a probabilistically consistent uncertainty distribution over aggregated groups.

## 7. Conclusions and future work

In addition to expanding the experiment to include products with larger quantities of reviews, we are also extending the current research in several directions. First, we would like to study the impact of the reviewer. By including reviews from new sources (e.g. a general purpose retailer like Amazon or a specialty retailer like Adorama), we might study whether different review sources reflect different underlying user communities. Review-source is only one of several variables we are using to study reviews as a knowledgebase. Review length and reviewer characteristics such as "total number of products reviewed" or "distance to the user in a social network" may also affect the informativeness of a review from a needs-centric perspective.

Second, we are studying the sensitivity of needs-centric rankings to the number of reviews. To increase the number of reviews within a single site, we are experimenting with Query Expansion and Product Expansion. Query Expansion addresses the question of finding related terms. Customers who search for blenders to make "frozen icy drinks" might also benefit from reviews relating to "margaritas." Likewise, Product Expansion extends the review knowledgebase to products with only a limited number of reviews. By discovering the common attributes of highly ranked

products that match a customer's needs and interests, we can also point to unranked products in the same category (or a different category) with similar attribute characteristics. We use decision-tree classifiers to elicit product attributes and attribute values useful for characterizing particular "needs." To decrease the number of reviews used to generate rankings, we are evaluating several different approaches to pruning review sets including random selection, discounting the influence of older reviews, and pruning reviews based upon reviewer characteristics.

Finally, we are designing user-studies to determine the impact of needs-centric v. product-centric searching on subjective variables such as time-to-search, confidence-in-selection, and intention-to-purchase. In a traditional product-centric presentation, each review is a tuple with a composite key of product and reviewer. Tuples are grouped by product. Users navigate the review space by ordering (ranking) products based upon one or more product attributes e.g. by manufacturer, by zoom or order by aggregation on scalar attributes (average rating) or lowest/highest price. By contrast, needs-centric search/navigation of the review space generates a needs-centric ranking of products (and their associated reviews). Needs-centric searching may increase confidence and decrease time-to-search because, in part, it offers evidence-based support for a specific product ranking. Customers can always refer to the underlying recommendations to explain and justify a particular product's ranking.

This work treats customer reviews as a knowledgebase about product categories. We propose a DB+IR integration strategy for supporting search of reviews. IR strategies match queries that are written in the vocabulary of the consumer, to reviews that are written by other consumers. Specifically, we introduce relevance-weighting to aggregate product review ratings and rank products based upon explicit customer needs. Not only does this enable customers to access the knowledge within reviews, but it also explains those rankings using the underlying reviews.

## Acknowledgements

Support for this work was provided by the Wharton eBusiness Initiative and the Fishman-Davidson Center for Service and Operations Management.

## 8. References

[1] Adomavicius, G., Sankaranarayanan, R., Sen, S. and Tuzhilin, A. Incorporating Contextual Information in Recommender Systems Using a Multidimensional Approach. *ACM TOIS*, 23, 1 (January 2005), 103-145.  
[2] Adomavicius, G. and Tuzhilin, A. Towards the Next Generation of Recommender Systems: A Survey of the

State-of-the-art and Possible Extensions. *IEEE Transactions of Knowledge and Data Engineering*, 17, 6 (2005), 734-749.  
[3] Chevalier, J. A. and Mayzlin, D. The Effect of Word of Mouth on Sales: Online Book Reviews. *Journal of Marketing Research*, 43, 3 (August 2006), 345-354.  
[4] Fang, H., Tao, T. and Zhai, C. A Formal Study of Information Retrieval Heuristics. *SIGIR* (Sheffield, UK, July 25-29, 2004).  
[5] Garcia-Molina, H., Ullman, J. D. and Widom, J. *Database Systems: The Complete Book*. Prentice Hall, Upper Saddle River, NJ, 2002.  
[6] Gelman, A., Carlin, J. B., Stern, H. S. and Rubin, D. B. *Bayesian Data Analysis*. Chapman & Hall/CRC, New York, 2000.  
[7] Gulutzan, P. MySQL's Full-Text Formulas. *Database Journal*(15 June 2005).  
[8] Kotler, P. *Marketing Management*. Prentice Hall, 2003.  
[9] Lee, T. Use-centric mining of customer reviews. *WITS* (Washington, D.C., 11-12 December, 2004).  
[10] Lee, T. Needs-Based Analysis of Online Customer Reviews. *ICEC* (Minneapolis, MN, 19-22 August, 2007).  
[11] Li, C., Wang, M., Lim, L., Wang, H. and Chang, K. C.-C. Supporting Ranking and Clustering as Generalized Order-By and Group-By. *SIGMOD* (Beijing, China, 11-14 June, 2007).  
[12] Liu, B., Hu, M. and Cheng, J. Opinion Observer: Analyzing and Comparing Opinions on the Web. *WWW* (Chiba, Japan, 10 - 14 May, 2005).  
[13] Mangalindan, M. *Web Stores Tap Product Reviews*. Dow Jones & Co., City, 2007.  
[14] McCabe, M. C., Lee, J., Chowdhury, A., Grossman, D. and Frieder, O. On the Design and Evaluation of a Multi-dimensional Approach to Information Retrieval. *SIGIR* (July, 2000).  
[15] Murthy, R. and Widom, J. Making Aggregation Work in Uncertain and Probabilistic Databases. *Stanford CS-TR 2007-7*, 2007.  
[16] Nasukawa, T. and Yi, J. Sentiment Analysis: Capturing Favorability Using Natural Language Processing. *K-CAP'03* (Sanibel Island, Florida, 23-25 October, 2003).  
[17] O'Keefe, R. H. and McEachern, T. Web-based Customer Decision Support Systems. *Communications of the ACM*, 41, 3 (March 1998), 71-78.  
[18] Orman, L. Consumer Support Systems. *Communications of the ACM*, 50, 4 (April 2007), 49-54.  
[19] Perez, J. M., Berlanga, R., Aramburu, M. J. and Pedersen, T. B. A Relevance-Extended Multi-dimensional Model for a Data Warehouse Contextualized with Documents. *DOLAP* (Bremen, Germany, 4-5 Nov, 2005).  
[20] Randall, T., Terwiesch, C. and Ulrich, K. T. User Design of Customized Products. *Marketing Science*, 26, 2 (March-April 2007), 268-280.  
[21] Ross, R., Subrahmanian, V. S. and Grant, J. Aggregate Operators in Probabilistic Databases. *Journal of the ACM*, 52, 1 (January 2005), 54-101.  
[22] Rundensteiner, E. A. and Bic, L. Evaluating Aggregates in Probabilistic Relational Databases. *Data & Knowledge Engineering*, 7, 3 (February 1992), 239 - 267.  
[23] Scaffidi, C., Bierhoff, K., Chang, E., Felker, M., Ng, H. and Jin, C. Red Opal: Product-Feature Scoring from Reviews. *ACM EC* (San Diego, CA, June 11-15, 2007).

[24] Stolze, M. and Strobel, M. Recommending as Personalized Teaching. in (eds. C.-M. Karat, J. O. Blom and J. KaratBook) *Designing Personalized User Experiences in eCommerce*. Springer, Netherlands, 2004.

[25] Urban, G. L. and Hauser, J. R. "Listening In" to Find Unmet Customer Needs and Solutions. eBusiness@MIT no. 156, MIT, Boston, MA, 2003.

[26] Zar, J. H. Significance Testing of the Spearman Rank Correlation Coefficient. *Journal of the American Statistical Association*, 67, 339 (September 1972), 578-580.