Volume 3, Issue 4, April 2013 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

Scoreboard Approach towards Products Ranking for e-Commerce Platforms

Namit Dhingra
Research Scholars, Department of CSE
Bharati Vidyapeeth University, College of Engineering,
Pune, Maharashtra, India

J. Naveen Kumar
Research Scholars, Department of CSE
Bharati Vidyapeeth University, College of Engineering,
Pune, Maharashtra, India

Abstract – Online Product Ranking is complicated by conflicting attributes. To promote the e-commerce services, it is necessary to upgrade existing online shopping networks to benefit not only small buyers but also volume buyers. However, the ranking systems used in existing e-commerce system fail to address volume sellers and buyers requirements. Current systems such as eBay or Amazon don't satisfactorily support buyers who want to order a large number of product items from different sellers at the same time. Earlier a product was ranked on the basis of product rating and product recommendations which mainly takes into account ratings of the user on a product. Now the rating of a product is done via Online Social Rating Networks (SRNs), these ratings and recommendations of users in the market, collaborated with Product Ranking Algorithms like "Rainbow Product Ranking Algorithms" help in upgrading the e-commerce and providing the user with better and essential information about the product.

Keywords - Product, Product Ranking, e-commerce services, Buyers, Sellers, Product Rating, Product Recommendations, Online Social Rating Networks (SRNs), Product Ranking Algorithms, Rainbow Product Ranking Algorithm.

I. Introduction

Product ranking is being done since e-commerce has been developed, previously it was based on product rating and product recommendations done by a user on a product. But now in the time of Social Networking being at its top the rating and recommendation on a product is done via "*Online Social Rating Networks (SRNs), such as Epinions and Flixter*", it allows users to form several "implicit" social networks, through their daily interactions like co – commenting on the same products, or similarly co – rating products. In SRNs user can also built their explicit social network by adding each other as friends. The existing e – commerce services need to be upgraded or modified according to the current market trends and also according to buyers and sellers. The current systems only keep their attention towards buyers and sellers who want items in bulk. In addition the buyers can be hindered by many conflicting issues, such as lower prices, shorter time of delivery, and higher transaction reputation. To address these complex issues, an intelligent ranking system must be generated for all available product items to help buyers make better decisions and sellers get big volume orders and thus higher profits. In online shopping, the system calculates a ranking score for each seller by combining multiple attributes with proper weight values. A good example is the **Best Match System** (Used basically buy online shopping corporation named "eBay.com"). Most previous work on Product Ranking concentrates on weighting multiple attributes. Obtaining weight distribution that satisfies all users is almost impossible. For example, the eBay system's attribute and weight distribution is hidden from buyers and could be biased toward sellers, leading buyers to make misinformed decisions. [3], [4]. To solve this problem, various Algorithms were developed to rank the sellers as well as the products sold by them in the market. One of such algorithms is *"Rainbow Ranking Algorithm"* which ranks the sellers and the products sold by them based on attributes such as sale price, quantity ordered, delivery time, sellers trust, and product quality and so on. In section II, the, analysis of *"Rainbow Ranking Algorithm"* is performed. In section III, the experiment and results of the *"Rainbow Ranking Algorithms"*, with different datasets and different Algorithms are discussed. In section IV, performances of *"Rainbow Ranking Algorithm"* is computed, followed by the references used in drafting of this paper.

II. Analysis If Rainbow Ranking Algorithms

The Rainbow ranking system, ranks sellers based on attributes such as sale price, quantity ordered, delivery time, seller trust, and product quality. Rainbow extends the single-level skyline approach to a multilevel optimization scheme that lets buyers' select product items at any quantity more systematically, with small computational overhead. The Rainbow system also extends eBay's one-to-one shopping paradigm to a many-to-many paradigm, widening both retailing and wholesales in any quantity. It eliminates the painful task of determining attribute weight distribution and expands online purchase services. Our experiments with eBay trace data prove the Rainbow system's efficiency in satisfying buyers' preferences. [1][2][6]

A. Online Shopping Networks

Figure 1 presents the architecture of an online shopping network. We can apply peer-to-peer (P2P) technology to extend this system from a *one-to-one* to a *many-to-many* paradigm. The system's core comprises the database subsystem, which keeps all item properties and purchase records. The *transaction manager* performs auctioning, ranking, authorization, selection, closure, and so on. The *reputation system* evaluates peer performance in all transactions. These components might be centralized as with eBay, or distributed depending on the network's implementation and operational cost limitations.

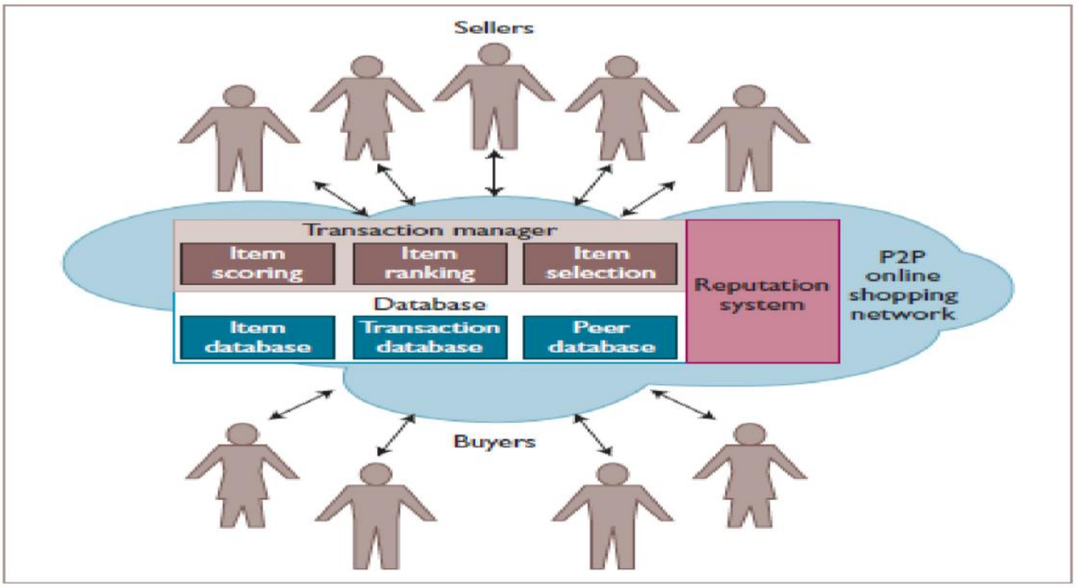

Figure 1. Peer-to-peer shopping network architecture for competitive e-commerce. Such an architecture expands online sale services to benefit both the retailing and wholesale of products.

Four system parameters $\{n, m, q, and\ k\}$ specify a P2P shopping network. There are n available items in response to a buyer's query. Each product item is characterized by m attributes. When a user chooses his or her favourite item, the network makes a one-to-one link between the buyer and the seller. We extend this paradigm by introducing the two parameters $\{q, k\}$. The Rainbow system, applies k skyline levels to help buyers make decisions on purchasing q items at once. The system thus links the buyer to q sellers at once. These parameters' range is based on online market distribution, product availability, purchase patterns, and user behaviours. In eBay's case, the typical values are in the range $1 \leq n \leq$ 10,000, $1 \leq m \leq 10$, $1 \leq q \leq$ 1,000, and $1 \leq k \leq 50$.

B. The Skyline Approach

The skyline method was developed as an operator for fast database search. To search over m attributes, a m-dimensional search space can be visualized. Each product item is represented by a point in this space. An item falls on the skyline if its attribute values aren't worse than any other points along all attribute dimensions. The skyline is formed by the best items that satisfy the shopping criterion.

Let's look at an example to illustrate this selection. Consider the 10 items in Table 1, labelled $I1$, $I2$…., $I10$, which appeared in response to a query about iPhones in eBay. Each item is described by two attributes: its cost, including the delivery charge, and the time left until the auction deadline. Buyers care not only about price but also about the time they have left to make a decision and win the bidding process. [10]

Table 1. Ten iPhone items listed on eBay.

ITEM ID	ITEM COST ($)	TIME LEFT (HRS.)
$I1$	227	23
$I2$	160	19
$I3$	100	27
$I4$	237	11
$I5$	250	40
$I6$	315	14
$I7$	138	12
$I8$	329	30
$I9$	69	16
$I10$	25	40

Table 2 shows the partial-ordering relationships among all pairs of items based on the attribute values in Table 1. In the diagonal, items are equal to themselves — for example $I9 = I9$. However, $I9 < I3$ because item $I9$ has cost = $69 and time left = 16 hours; both are lower than the $100 and 27 hours for item *I*3. Similarly, the > relationship is exemplified by *I*1 > *I*2. If one item is better than another in one attribute dimension but inferior in a different dimension, however, we consider them incomparable — for example, *I*2 ? *I*4 because Table 1 shows that *I*2 is lower in price but higher in time left than *I*4.

In Table 2, we see that *I*4, *I*7, *I*9, and *I*10 are skyline items because their corresponding rows have only <, =, and? Entries. Figure 2a shows a single-level skyline solution, with items represented by red squares and linked with a red line. All items above the skyline have either higher cost or longer time to expire than the skyline items. Essentially, the skyline divides all items into two categories: acceptable (*I*4, *I*7, *I*9, *I*10) versus rejected items (*I*1, *I*2, *I*3, *I*5, *I*6, *and I8*). *Algorithm 1* specifies the procedures for skyline levelling.

Table 2. Partial-ordering relationships among the 10 product items from Table 1.*

ITEMS	***I*1**	***I*2**	***I*3**	***I*4**	***I*5**	***I*6**	***I*7**	***I*8**	***I*9**	***I*10**
*I*1	=	>	?	?	<	?	>	<	>	?
*I*2	<	=	?	?	<	?	>	<	>	?
*I*3	?	?	=	?	<	?	?	<	>	?
*I*4	?	?	?	=	<	<	?	<	?	?
*I*5	>	>	>	>	=	?	>	?	>	>
*I*6	?	?	?	>	?	=	>	<	?	?
*I*7	<	<	?	?	<	<	=	<	?	?
*I*8	>	>	>	>	?	>	>	=	>	?
*I*9	<	<	<	?	<	?	?	<	=	?
*I*10	?	?	?	?	<	?	?	?	?	=

*= represents two items' equality in all attribute values > means the right-hand item is better than the left-hand one

< indicates that the left-hand item is better than the right-hand one ? marks pairs that aren't comparable

Algorithm 1: Skyline selection

Input: Set I of responding items to a buyer's query

Output: A subset S of I (skyline items) as the finalist to purchase from

Procedure:

1. **Forall** items **x ε I**
2. **Set** the Flag = true
3. **Forall** item **y ε I**
4. **If x > y**, set the Flag = false
5. **EndForall**
6. If Flag = true, then add item x as a skyline item in set S
7. **EndForall**

Table 3. Partial-ordering relationships among six items after removing level 1 items

ITEM	***I*1**	***I*2**	***I*3**	***I*5**	***I*6**	***I*8**
*I*1	=	>	?	<	?	<
*I*2	<	=	?	<	?	<
*I*3	?	?	=	<	?	<
*I*5	>	>	>	=	?	?
*I*6	?	?	?	?	=	<
*I*8	>	>	>	?	>	=

C. Rainbow Spectrum

Single-level skyline items aren't different than any of the remaining unselected items, but this quantity might fall short of a buyer's demand. We developed the Rainbow approach to solve this problem. Figure 2b shows the basic concepts for generating a Rainbow spectrum over 10 items in Table 1. We first select four level-1 skyline items: *I*4, *I*7, *I*9, and *I*10. This leaves us with six unselected items. Table 3 shows their partially ordered relationships. We apply the same skyline *Algorithm 1* to these six remaining items. Using Table 3, we produce a level-2 skyline, which consists of three additional items, *I*3, *I*2, and *I*6.

For example, if users want to select many items at once, the system can first display the items at level 1. If this order doesn't provide enough items, the system will display the items in level 2 and so on, until the users are fully satisfied with the volume order. So, the first two levels will let users buy up to seven items. To satisfy online wholesales, however, we need a more flexible product-ranking and item-selection method. Thus, Rainbow method is used to build multiple skyline levels across its spectrum. We use *Algorithm 1* as a kernel computation in the Rainbow spectrum-generation procedure (see *Algorithm 2*), which in turn will be called by multi-attribute ranking procedure in *Algorithm 3*.

Algorithm 2: Generating the Rainbow spectrum

Input: Set *I* of *n* responding items to a buyer's query
Output: Divide the *n* items into *k* skyline levels *Li* for *i* = 1, 2 …k
Procedure:

1. Start with *i* = 1
2. **While** *I* is not empty
3. **Call** Algorithm 1 to generate skyline level *Li* of *ni* items
4. Remove level *Li* items from set *I*
5. Increment index *i* = *i* + 1 //Go to next skyline level//
6. **End While**

Algorithm 3: Rainbow ranking of product items

Input:
I: set of responding items,
q: number of ordered items,
A: attribute used in product ranking at selected skyline levels
Output: An ordered list *R* of *q* items in *k* skyline levels
Procedure:

1. Call Algorithm 2 to generate the Rainbow spectrum in *k* levels
2. For *i* = 1 to *k*
3. Generate *Ri* by sorting items in *Li* based on attribute *A*
4. Stop when |*R*1|+ |*R*2| + … + |*R*i| ≥ *q*
5. R is formed by concatenation *R* = *R*1, *R*2 … *R*i

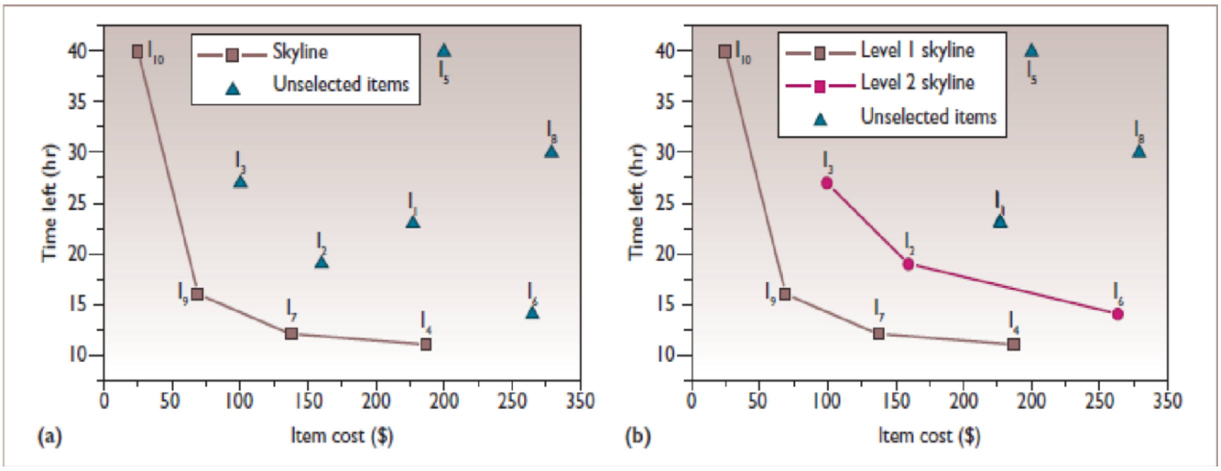

Figure 2. A skyline solution and a Rainbow spectrum over the 10 product items in Table 1. We can see (a) a single-level skyline solution and (b) a Rainbow spectrum consisting of two skyline levels

1) *Rainbow Ranking Process*: Consider a case in which a buyer wants to purchase *q* product items. After submitting a request to the system, the buyer receives *n* > *q* candidate items in response. The problem is selecting *q* items out of those *n* items. One solution is to choose the *q* items all from the level 1 skyline (see Figure 2a), which works only if the ordered quantity *q* is satisfied at level 1. However, this approach has its limits when serving large e-commerce customers. Thus, Rainbow extends the skyline method to multiple levels (see Figure 2b),
2) *The Selection Process***:** Figure 3 illustrates the Rainbow item selection process. The example generates three primitive attributes, *A*1, *A*2, and *A*3; *A*4 is a compound attribute from eBay. Users can further generate their own attributes by merging primitive ones to create more complex ones, such as *A*5, *A*6, and *A*7. Rainbow can use these attributes in three ways. First, some are used as filter attributes (*A*6 and *A*7, for example) to generate a subset of items that satisfies user preferences. Second, some are used as levelling attributes (*A*1, *A*2, and *A*5, for example) to create skyline levels across the Rainbow spectrum. Finally, one attribute (*A*4, for example) is used to rank the items within each level of the

spectrum. We apply item levelling in Rainbow with a macroscopic view of buyer behaviour. Rainbow levelling can be combined with any other ranking algorithm, We specify the Rainbow levelling process with a 3-tuple, $<A, I, L>$, where

- $A = \{A1, A2\ldots Am\}$ is a set of attributes describing the desired product items. The smaller the attribute's value, the better the product item.
- $I = \{I1, I2\ldots In\}$ is the set of available items responding to a query search.
- $L = \{L1, L2\ldots Lk\}$ is a set of k skyline levels (subsets) in the Rainbow spectrum. Each level Li is a subset of available items from the set I.

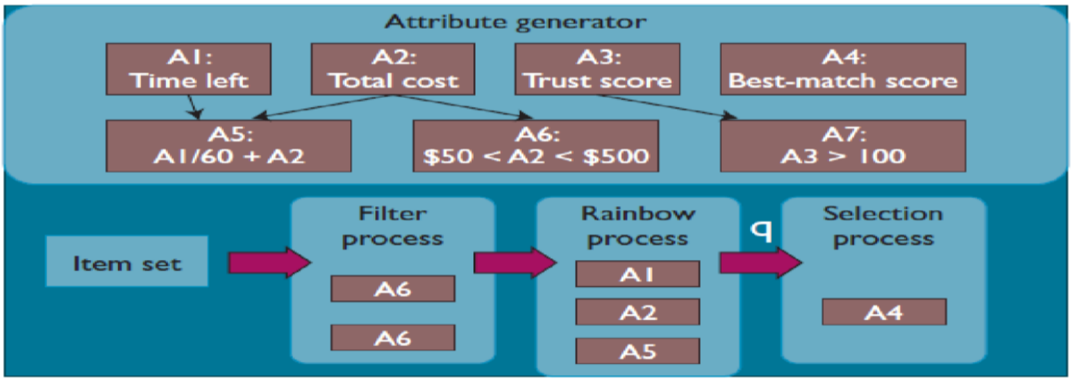

Figure 3. A representative Rainbow ranking process. The process consists of attribute generation, item filtering, ranking and levelling, and final selection of q product items.

3)*Rainbow Spectrum Generation*: Once Rainbow decides on the number k of skyline levels, *Algorithm 2* generates the Rainbow spectrum in k iterations. The aim is to generate k skyline levels to satisfy the purchase order from buyers. Items in successive skyline levels on the Rainbow spectrum must satisfy two properties:

- When $i < j$, any item in level Li must be better than or incomparable with any item in level Lj.
- Each item must be located at its lowest Rainbow level in the spectrum.

We specify the Rainbow ranking scheme in *Algorithm 3*, which calls *Algorithm 2* repeatedly. The idea is to subdivide the product items into multiple skyline levels via relaxation over the attribute values. Assume ni items in skyline level Li. We must apply *Algorithm 1* repeatedly k times to generate the k skyline levels. Because the buyer has requested q items, we decide the value of k as the minimum integer that satisfies the following condition:

$\Sigma\, ni \geq q$ for $i = 1$ to k. [1]

We *Algorithm 3* can be illustrated via a simple example, where I = {10 items in Table 1}, $q = 10$, and A = price. Let's look at this example in five steps:

1. Our system divides the 10 items into four levels: $L1 = \{I4, I7, I9, I10\}$, $L2 = \{I6, I2, I3\}$, $L3 = \{I1\}$, and $L4 = \{I8, I5\}$, so $k = 4$.
2. The index i runs from 1 to 4.
3. The system generate $R1 = \{I10, I9, I7, I4\}$ by ranking the items in $L1$ under the price attribute. Similarly, we generate $R2 = \{I3, I2, I6\}$; $R3 = \{I1\}$, and $R4 = \{I5, I8\}$.
4. Because $|L1| = 4$, $|L2| = 3$, $|L3| = 1$, $|L4| = 2$, their sum is $10 \geq q$.
5. We generate the ranking order R by concatenating $R1\ R2\ R3\ R4$ as follows: $\{I10, I9, I7, I4, I3, I2, I6, I1, I5, I8\}$, which column 4 in Table 4 shows.

As we've discussed, users can specify the attribute used to rank the items within the skyline levels. This leads to the alternating choices Table 4 illustrates. [1]

III. Experiments And Results

A.Alternate Ranking Schemes

In this approach, users can choose from multiple ranking schemes. *Dimensional ranking* is based on sorting items by only one attribute at each skyline level. It appeals to users who prefer specific attributes to dominate. *Mixed ranking* is based on sorting items by combing other complex ranking algorithms, such as the Best Match methods in eBay. The idea is to generate the Rainbow spectrum first, then apply the weighted sum score calculated via eBay's Best Match scheme to rank all items in the same level.

Table 4 shows four alternative ways to rank 10 product items in our running example. The choice of ranking scheme depends on the buyer's preference. For example, for those who are primarily concerned with price, skyline items in the same level should be ranked by item cost. We could also rank items in the same level using other attributes, such as auction deadline, trust scores, or time to delivery. We can use the order information from all four ranking methods to make a final choice. We do this by finding the intersection of different ranking orders up to the top q selection. When q becomes very large, the ranking order might differ very little. Say, for example, we want to select four items for purchase. Using Rainbow ranking based on either auction deadline or item cost, we have the same subset of selected items. However, the top four choices differ by 50 percent when Rainbow is aided by the Best Match method. When the order quantity becomes very large, this difference increases rapidly. The last column in Table 4 shows rankings aided by Best Match. With alternating choices, the Rainbow rankings let buyers have more personalized choices to place their orders [2] [3]

B.Testing Results on eBay Trace Data

To test our Rainbow system, we conducted trace experiments using eBay trace data. We checked the sales records of 3,123 iPhone items in 120 hours on eBay during July 2008. Each item is characterized by an eBay trust score, item cost, time left, and so on. The original item rankings were determined based on eBay's Best Match system. The item cost includes the sales price and shipping charges, and the time left counts up to the end of the auction process. The trust score summarizes the seller's reputation. We report our eBay testing results looking at the item cost and auction deadline attributes. Depending on the purchase quantity q, buyers choose items at lower levels more often than those chosen at higher skyline levels. [2]

C. Product Ranking Results

In our experiments, we bundle every 10 skyline levels together into a spectrum band. Figure 4 shows 163, 280, and 360 items in three total spectrum bands. When the order quantity is low, items on the lower band can satisfy the buyers. Otherwise, the system will choose more skyline levels from higher bands. The remaining 2,320 dots in Figure 4 correspond to unselected items (each dot is a single item in the 2D attribute space). Most top-ranking items are located in the lower left corner of the graph. The lower the auction deadline and item cost values, the better deal buyers can get. [2]

Table 4. Three Rainbow ranking orders compared with eBay's Best Match system.

ITEM ID	EBAY'S BEST MATCH	RAINBOW BY AUCTION DEADLINE	RAINBOW BY ITEM COST	RAINBOW AIDED BY BEST MATCH
*I*1	1	8	8	8
*I*2	2	6	6	5
*I*3	3	7	5	6
*I*4	4	1	4	1
*I*5	5	10	9	9
*I*6	6	5	7	7
*I*7	7	2	3	2
*I*8	8	9	10	10
*I*9	9	3	2	3
*I*10	10	4	1	4

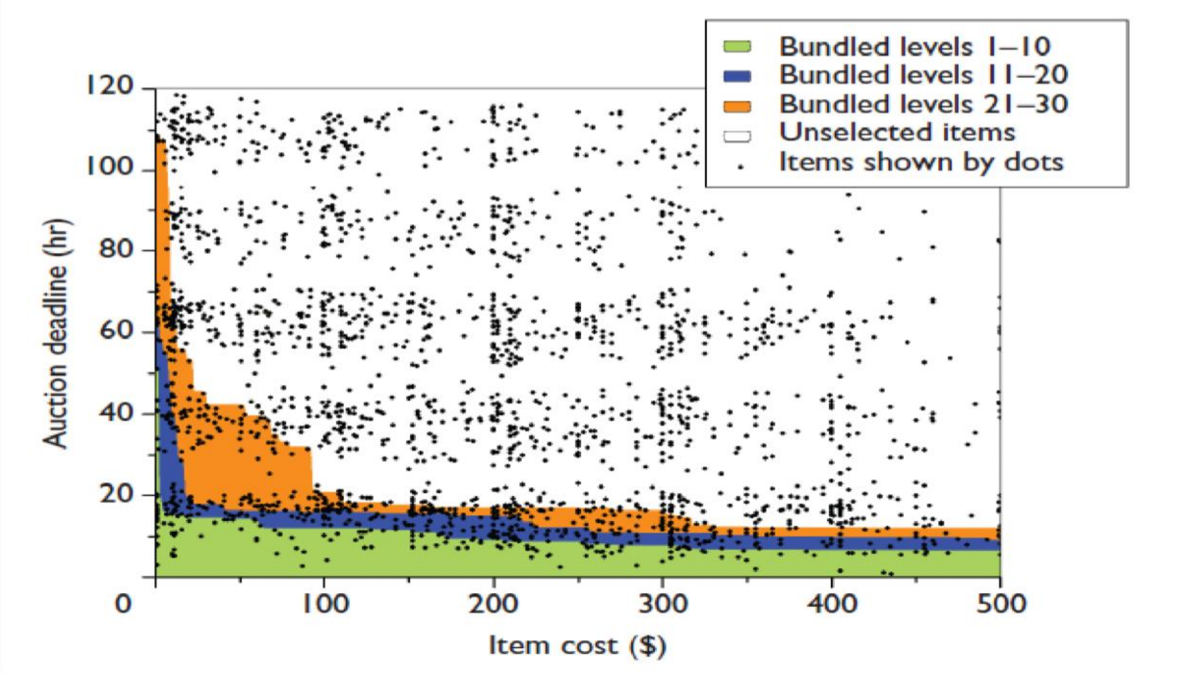

Figure 4. Rainbow ranking of 3,123 iPhone items in an eBay experiment. Each dot represents a single item. Every 10 skyline levels are bundled together into three spectrum bands, consisting of 163, 280, and 360 items, respectively. The remaining 2,320 items are unselected. Top-ranking items have lower cost and shorter auction deadlines

IV. Conclusion

The Rainbow system combines the advantage of the skyline method and eBay's Best Match method. The Rainbow system is easy to implement and user friendly to appeal to both buyers and sellers. The buyer receives more versatile choices among preferred product items and the seller receives better profit returns due to volume sales. It also optimizes the decisions for all parties involved in an e-transaction and can greatly improve QoS – Quality of Service, as evidenced by the 50 to 70 percent performance gain Rainbow had compared with the skyline and Best Match methods.When the number of ranking attributes becomes very large, we face a degeneration problem due to conflicting goals. Many merchandise pairs might become incomparable. On the surface, this implies that Rainbow scheme might not perform well under degeneration. However, from the trace data we crawled on eBay, we find that first-level items won't increase much when more attributes are added. This is due to the fact that a few dominating attributes dictate first-level item selection. The eBay database taught us that trusted users, reasonable pricing, and shorter auction deadlines predominate user choices.

The remaining attributes impact only the higher-level or lower-priority choices. This situation is quite close to most customer behaviours. Thus, Rainbow prevails under degeneration cases. For an example with 10 attributes, we first use four dominating ones to level the items. We then use minor attributes to achieve further levelling. Because we use fewer attributes at successive levels, this converts many incomparable pairs to comparable ones with <, >, or = relationships, greatly alleviating the degeneration problem.

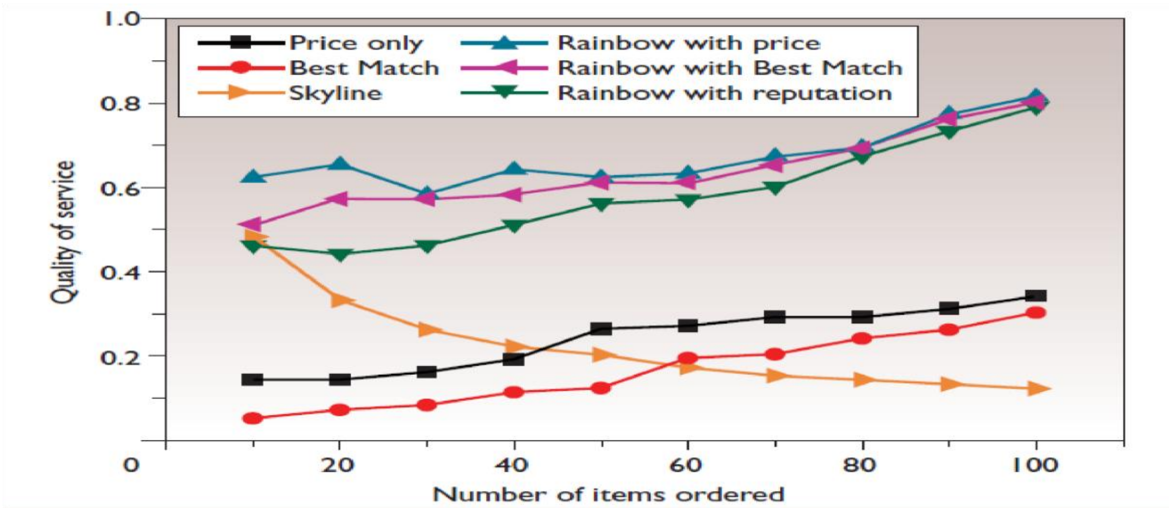

Figure 5. Quality of service (QoS) that buyers experience when order quantity increases. The top three Rainbow curves outperform the lower three QoS curves for the Skyline method and two eBay selection methods by 50 to 70 percent

Acknowledgment

The authors would thank the management of Bharati Vidyapeeth University, College of Engineering, Pune, Maharashtra for their support and the department of Computer Science and Engineering for their encouragement towards the research work.

References

[1] K. Lin, "E-Commerce Technology: Back to a Prominent Future," *IEEE Internet Computing*, vol. 12, no. 1, 2008, pp. 60– 65.

[2] J.C. Butler et al., "Enabling e-Transactions with Multi- Attribute Preference Models," *European J. perational Research*, no. 186, 2008, pp. 748–765.

[3] M. Bohanec and B.A. Zupan, "Function-Decomposition Method for Development of Hierarchical Multi-Attribute Decision Models," *J. Decision Support System*, Jan. 2004, pp. 215–233.

[4] J. Wallenius et al., "Multiple Criteria Decision Making, Multi-Attribute Utility Theory: Recent Accomplishments and What Lies Ahead," *Management Science*, vol. 54, no. 7, 2008, pp. 1336–1349.

[5] J.S. Dyer and R.A. Sarin, "Measurable Multi-Attribute Value Functions," *Operation Research*, vol. 22, 1979, pp. 810–822.

[6] J.E. Teich et al., "A Multi-Attribute e-Auction Mechanism for Procurement: Theoretical Foundations," *European J. Operational Research*, vol. 175, 2006, pp. 90–100.

[7] Z. Liang and W. Shi, "Analysis of Ratings on Trust Inference in Open Environments," *J. Performance Evaluation*, vol. 65, no. 2, 2008, pp. 99–128.

[8] D.W. Manchala, "E-Commerce Trust Metrics and Models," *IEEE Internet Computing*, vol. 4, no. 2, 2000, pp. 36–44.

[9] Y. Wang and K. Lin, "Reputation-Oriented Trustworthy Computing in E-Commerce Environments," *IEEE Internet Computing*, vol. 12, no. 4, 2008, pp. 55–59.
[10] S. Börzsönyi, D. Kossmann, and K. Stocker, "The Skyline Operator," *Proc. 17th Int'l Conf. Data Eng.*, IEEE CS Press, 2001, pp. 421–430.
[11] S. Song et al., "Trusted P2P Transactions with Fuzzy Reputation Aggregation," *IEEE Internet Computing*, vol. 9, no. 6, 2005, pp. 18–28.
[12] R. Zhou, K. Hwang, and M. Cai, "GossipTrust for Fast Reputation Aggregation in Peer-to-Peer Networks," *IEEE Trans. Knowledge and Data Engineering* (TKDE 08), IEEE CS Press, 2008, pp. 1282–1295.

