# Islands of Interest: Mining Concentrations of User Search Intent over e-Commerce Product Categories

Neeraj Pradhan*, Vinay Deolalikar†, and Deependra Singh‡

Search and Data Mining
Groupon
Palo Alto, CA 94306
E-mail: *prad.neeraj@groupon.com, †deolalikar.academic@gmail.com, ‡deeps@seas.upenn.edu

***Abstract*—e-Commerce has many core problems which can benefit from data mining—constructing recommendations for users, designing product taxonomies in a way that the user finds easy to navigate to, facility allocation for inventory to minimize shipping costs—to name a few.**

**A big component of e-commerce data comprises of search activity. Search as a share of traffic has overtaken direct browsing, and most e-commerce sites generate more search data than browse data. Search session data is also more voluminous than any user aggregated data since it includes anonymous sessions. However, search data is inherently *local* to a query. Therefore, it is not immediately obvious whether it can be used to build *global* (i.e., where no query is involved) knowledge to address many of these problems that are of interest.**

**In this paper, we introduce a global structure, namely *islands of interest*, that is mined from local search data. We show that these concentrations of user search intent are highly relevant to each of the e-commerce problems mentioned earlier. We introduce two algorithms—one based on community detection, and the other on clustering—that can identify islands of interest. We build a framework that can compare the characteristics of the islands identified using these two approaches. We believe that in addition to providing insights into user behavior, islands of interest can be important in tackling lesser researched problems such as the design of product taxonomies.**

## I. Introduction

There are many core technical challenges in e-commerce which can benefit, and in many cases only be viably addressed at massive scales, using big data mining. Some of these problems, such as building user recommendations, have received widespread research attention, while others have not been as extensively studied.

One such example that has received limited attention is the design of the *customer-facing taxonomy*[1], which we simply refer to as the *taxonomy*. This is central to product layout in e-commerce. The taxonomy is usually of the form of a multi-level directed acyclic graph (Fig. 1) and mediates the user experience in two important ways—by directly providing a navigational channel to browse the products within each category, or indirectly through recommendations such as the "related items", and "you may also like" widgets that rank other items that are close to the given product in the taxonomy using a metric (such as popularity).

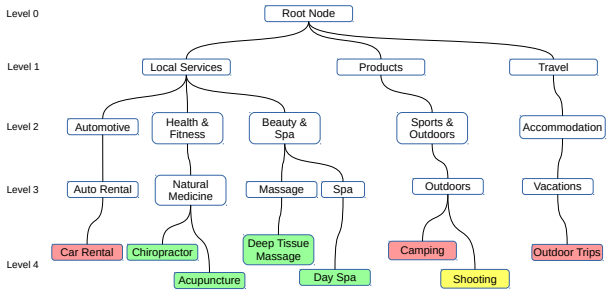



Fig. 1: An example of a taxonomy showing color-coded concentration of user intent across nodes that are far apart in the taxonomy tree.

For the most part, such taxonomies are hand-crafted based on human intuition, and evolve very slowly due to the technical costs involved. This is a recurring investment of time and money, and underscores the importance of understanding what constitutes a well-designed taxonomy. Another example of a relatively lesser researched problem is that of inventory placement in warehouses.

Users interact with e-commerce sites primarily through two means: search and browse. Search involves specification of an intent, expressed through a query. The e-commerce site then responds by showing products that match that specific intent. Browse is typically more free-form, and a single browse session might incorporate diverse product views. At Groupon, a major e-commerce vendor, search has been the fastest growing component of traffic, and has long surpassed browse in overall traffic generation. We believe that similar statistics hold true at other e-commerce majors, particularly those with a large inventory.

Search data is also typically much larger than any user aggregated data because it incorporates all search activity, including those from anonymous traffic sources, e.g. first time

* Work done while authors were at Groupon Technologies, Inc.

[1]This is the publicly available taxonomy that the user can view, and navigate. E-commerce sites typically also have an internal taxonomy that is often finer than the customer-facing taxonomy.

users and users who are not logged in. However, since a search session is specific in intent, conventional wisdom is that search data has limited applicability when it comes to modeling situations that require global knowledge, which is the case for all the problems mentioned earlier. For instance, most of the existing literature uses user aggregated browse data, when designing recommender systems.

In this paper, we address this gap by building a framework that allows us to mine search data, which until now has mostly been restricted to problems in retrieval, to address some of these core e-commerce challenges. Our framework combines a combination of techniques from data mining—one-mode projections [15], [7], random walks, community detection [9], spherical k-means clustering [5]—to mine global structure from local search session data. Our work is being put into production at Groupon.

## II. Algorithms

### A. Background and Preliminaries

*1) Customer Facing Taxonomy:* The customer-facing taxonomy is designed to provide an intuitive categorization for users to navigate the products in an e-commerce application. Fig. 1 shows a subset of the product taxonomy. The *depth* of the taxonomy $\Delta$ is the number of levels that exist below the root node in $\mathcal{T}$. In Fig. 1, $\Delta$ is 4.

A *leaf node* in a taxonomy is one that has no children. Distinct products assigned to the same leaf node are usually very similar. Products assigned to sister leaf nodes, i.e. those sharing the same parent node, are somewhat less so. Let us denote our customer-facing taxonomy by $\mathcal{T}$, and the set of $n$ leaf nodes by $\mathcal{C} = \{c_1, c_2, .., c_n\}$.

A *taxonomy partition* $\mathcal{P}$ is defined as the partitioning of $\mathcal{C}$ into $K$ components, i.e. $\mathcal{P} = \{P_1, P_2.., P_K\}$, where $K \geq 1$ is referred to as the *size* of the partition. Each leaf node falls into exactly one component. The *implicit partition* of the taxonomy $\mathcal{T}$ is defined as the partitioning of $\mathcal{C}$ such that each component consists of all leaf nodes that are children of the same parent node in $\mathcal{T}$.

*2) Query-Click Graph:* Product search engines on e-commerce sites aim to match a user's intent with the inventory of products indexed in the database and rank them by their relevance to the given query. The clicks generated for a given query term over the product inventory, when aggregated over all users, can be an important relevance signal. In this study, instead of looking at click volume from queries to individual products, we aggregate it over the different leaf nodes of the taxonomy, which can negate issues related to sparsity of click volume over individual products [14].

Let us denote the set of $m$ queries in our model by $\mathcal{Q} = \{q_1, q_2, .., q_m\}$. We can connect a query $q_i$ with a category $c_j$ by an edge, whose weight $w_{i,j}$ is given by the click volume going from $q_i$ to $c_j$. We represent the edge weights by the weight matrix $\mathbf{W} : \mathbb{R}^{m \times n}$, where $w_{i,j}$ is the $(i, j)$ entry in $\mathbf{W}$. The *query signature* for the category $c_i$ is defined as the click volume flowing from $\mathcal{Q}$ to $c_i$, and is denoted by the column vector $\mathbf{w}_i$,

$$\mathbf{W} = (\mathbf{w}_1, \mathbf{w}_2, .., \mathbf{w}_n)\,.$$

For certain algorithms, we look at the transpose of the weight matrix, namely $\mathbf{W}^T$, where each row corresponds to a leaf category's query signature.

We will now discuss the two algorithms that we use in this paper to mine islands of interest. The corresponding schematic is provided in Fig. 2.

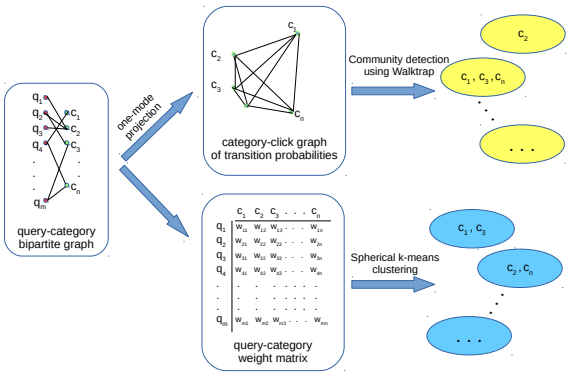



Fig. 2: Schematic of algorithm IslandComms (top) and algorithm IslandClusts (bottom)

### B. IslandComms Algorithm

*1) One Mode Projection. :* Many real-world networks can be modeled as bipartite graphs where edges capture the interaction between two *modes* or distinct sets of entities (in this case, $\mathcal{Q}$ and $\mathcal{C}$) [8], [13]. In many cases, we are primarily interested in modeling the relationship between nodes in one of these modes. A common approach is to use *one-mode projection* [15], [7] which models edge weights between nodes in one of the modes by incorporating information about their connectivity in the original bipartite graph.

Some advantages in analyzing the one-mode projection of the original graph is availability of many existing network analysis algorithms typically designed for one-mode graphs and, in some cases, the computational benefits of running these algorithms on the reduced size graph. In our case, the second benefit is particularly relevant as $|\mathcal{C}| \ll |\mathcal{Q}|$.

Projecting the query-category graph on to $\mathcal{C}$ results in a graph that connects different leaf nodes within $\mathcal{C}$. The projected weight matrix $\mathbf{W}^{\mathcal{C}} : \mathbb{R}^{n \times n}$ models the strength of the connection between the different leaves, with $w^{\mathcal{C}}_{i,j}$ denoting the edge weight between category $c_i$ and category $c_j$ in the projected graph. We shall refer to this projected graph as the *category-click graph*.

There is limited literature on the problem of one-mode projection of a weighted bipartite graph. To address this issue, we use a *cosine similarity projection* that models the strength of the connection between $c_i$ and $c_j$ by the cosine similarity of the click distribution coming from $\mathcal{Q}$ given by their query signatures, $sim_{cos} = \frac{\mathbf{w}_i \cdot \mathbf{w}_j}{\|\mathbf{w}_i\| \|\mathbf{w}_j\|}$. We apply a threshold $\theta$ to

remove the long tail of noisy edges connecting dissimilar categories, so that the projected weight is given by

$$w^{\mathcal{C}}_{i,j} = \begin{cases} sim_{cos}(c_i, c_j) & \text{, if } sim_{cos}(c_i, c_j) \geq \theta \\ 0 & \text{, if } sim_{cos}(c_i, c_j) < \theta. \end{cases}$$

*2) Island detection using Walktrap communities.:* We use a community detection algorithm, *Walktrap*, on the projected graph that utilizes short random walks on the graph to find communities of like nodes. The idea behind the Walktrap algorithm is that short random walks starting at any given node tend to get trapped within communities [9]. The *probability transition* for a node is the probability of reaching any other node starting from the given node in $t$ random hops. The probability transition vectors of two nodes that have dense inter-connections is likely to be similar. Walktrap defines a distance metric over the space of probability transitions and uses this metric to do a hierarchical agglomerative clustering.

*3) Parameters in* ISLANDCOMMS.*:* There are two tuning parameters — the number of steps in the random walk $t$, and the threshold $\theta$ used in the cosine similarity projection (Sec. II-B1).

### C. ISLANDCLUSTS *Algorithm*

As our baseline, we discover category communities directly from the query-click data over $\mathcal{C}$ by clustering the query signatures of different categories given by $\mathbf{W}^T$ (Sec. II-A2). Standard k-means with euclidean distance does not give good clustering results on the category-click data due to the high dimensionality of the query signatures as $|\mathcal{C}| \ll |\mathcal{Q}|$. As a solution, we can treat $\mathbf{W}^T$ as a document-term matrix, and use spherical k-means [5], an efficient document clustering algorithm that uses the cosine distance metric.

*1) Parameters in* ISLANDCLUSTS.*:* Note that there is only one tuning parameter — number of resultant islands $K$.

In Sec. IV, we will be using the framework discussed above to identify islands of interest in the leaf nodes of the taxonomy, and study their properties.

## III. RELATED WORK

The bipartite graph of query-click data has been well studied to provide query suggestions [6] and rank retrieved documents [3]. The authors in [6] use hitting time from random walks to rank queries, and in [3], the authors use forward and backward walks to rank images associated with a given query. Besides the bipartite graph, short random walks on derived graphs like the query-flow graph, a unipartite directed graph containing query transition probabilities, has also been studied for query suggestions [1]. Our motivation for doing a random walk on the query-category graph is to quantify the similarity between different nodes in the taxonomy, and uncover the community structure that corresponds to a concentration of user intent flow. Note that we do not directly do a random walk on the query-category graph, but rather its one-mode projection on to the set of leaf node categories.

Our work is relevant to *collaborative filtering* (CF) methods for recommendations that infer user's preferences from remaining users having similar tastes [11], [12]. In many cases, an explicit ratings matrix is not available, and more noisy implicit signals like click volume need to be taken into account [4]. In the case of Groupon, deals have a high churn rate, which makes implicit signals important. Furthermore, the islands of interest provide a lower dimensional attribute space over which user activity can be projected, and can be used as building blocks for CF based recommender systems. In [14], the authors also use information from the taxonomy to develop a hybrid filtering method that addresses the issue of sparsity. While all of these CF based approaches use data aggregated over users, we mine search session data that is often larger because it includes sessions from users who are not logged in.

The problem of mining association rules, and more generally correlations, from market basket data is an old one [2]. Such correlation data can be used to inform inventory placement in warehouses. Our method provides an arguably simpler and slower changing recipe, as it can mine similarity in the space of taxonomy leaf nodes into which items are already catalogued. This, therefore, also addresses the issue of cold start and sparsity of more direct user signals like purchases.

## IV. PROPERTIES OF ISLANDS

We introduce four quantitative measures that allow us to characterize partitions of a taxonomy. The first two measures were introduced in [10] and we extend their definition to an arbitrary taxonomy partition, $\mathcal{P}$, to characterize the partitions mined by the islands of interest. The latter two allow us to quantify how different, in terms of relative location in the taxonomy, an island of interest is to the implicit partition.

*1) Locality.:* A property that we would intuitively associate with an island of interest is that its constituent leaves are *similar*. We have already defined a notion of similarity between two leaves $c_i$ and $c_j$ in terms of the cosine similarity of their *query signatures*, $sim_{cos}(c_i, c_j)$. Given a taxonomy $\mathcal{T}$ and its partition $\mathcal{P}$, we can define locality as the weighted average cosine similarity of the categories within the same component,

$$\lambda^{\mathcal{P}} = \frac{\sum_{k=1}^{K} \sum_{i,j \in P_k; j>i} sim_{cos}(c_i, c_j)}{\sum_{k=1}^{K} \binom{|P_k|}{2}}.$$

Note that high locality favors higher number of partitions $K$ with each partition $P_k$ containing leaf nodes with high similarity.

*2) Interest Flow.:* In order to measure the concentration of user interest with respect to a query over any partition of the taxonomy, we define the notion of *flow of interest*, or simply *flow*.

We define the click volume from a query $q_i$ to a specific $P_k \in P$ as the aggregated clicks to all leaves belonging to $P_k$, $V_{i,k} = \sum_{j \in P_k} w_{i,j}$. Similarly, the click volume of a query $q_i$, denoted by $V_i$, can be defined as the aggregated clicks over $\mathcal{C}$, $V_i = \sum_{j=1}^{n} w_{i,j}$. Given a taxonomy $\mathcal{T}$, its partition $\mathcal{P}$, and a

query $q_i$, the interest flow $\xi_i^{\mathcal{P}}$ for $q_i$ is defined as the entropy of the click distribution of $q_i$ over $\mathcal{P}$, $\xi_i^{\mathcal{P}} = \sum_{k=1}^{K} \frac{V_{i,k}}{V_i} \ln\left(\frac{V_{i,k}}{V_i}\right)$.

We can now define the interest flow for a taxonomy partition $\mathcal{P}$ as the aggregate flow of interest over all queries $q \in \mathcal{Q}$ weighted by the click volume for the query,

$$\xi^{\mathcal{P}} = \frac{\sum_{q=1}^{m} V_q \xi_q^{\mathcal{P}}}{\sum_{q=1}^{m} V_q} = \frac{\sum_{q=1}^{m}\sum_{k=1}^{K} V_{q,k} \ln\left(\frac{V_{q,k}}{V_q}\right)}{\sum_{q=1}^{m} V_q}.$$

Note that low flow tends to favor smaller sized partitions which leads to a natural concentration of query click volume. One way to think of flow is in terms of separation of items in the taxonomy which are closely related in intent. Therefore, while high locality is a measure of *precision*, low flow can be considered a *recall* measure. [2]

While locality and flow give us a framework to compare the quality of partitions, we would further like to know how they differ with respect to the implicit partition. To develop a measure of distance of a partition from the taxonomy, we introduce the notion of *girth* and *expansion coefficient* of a component of a partition.

*3) Girth:* The *taxonomic distance* $d(c_i, c_j)$ between two nodes $c_i$ and $c_j$ is the maximum of the height from the lowest common ancestor of $c_i$ and $c_j$ to either of $c_i$ or $c_j$ in $\mathcal{T}$. Note that $1 \leq d(c_i, c_j) \leq \Delta \; \forall c_i, c_j \in \mathcal{C}$. The taxonomic distance is 1 when $c_i$ and $c_j$ are leaf sister nodes, whereas it is maximum ($\Delta$) if $c_i$ and $c_j$ have the root as the lowest common ancestor. We then define the girth of a component $P_k \in P$ as the average taxonomic distance between any two nodes in $P_k$,

$$\rho(P_k) = \frac{\sum_{i,j \in P_k; j>i} d(c_i, c_j)}{\binom{|P_k|}{2}}.$$

*4) Expansion Coefficient:* Given a set of leaf nodes in a component $P_k$, we define the *closure* $cl(P_k)$ as the superset that is closed under the operation of taking sister nodes. The expansion coefficient of a component is then defined as the ratio of the size of its closure and the component's size,

$$\eta\left(P_k\right) = \frac{\left|cl\left(P_k\right)\right|}{\left|P_k\right|}.$$

For instance, if a component's expansion coefficient is 2, it implies that around half the leaf nodes from related components in the implicit partition were picked up to form this component. [3] For a partition $\mathcal{P}$, we define the *median girth* and *median expansion coefficient* as the median of the respective measures for all the components in $\mathcal{P}$.

## V. Dataset and Parameter Settings

Our experiments have been performed on search session data and taxonomy data from Groupon. The query-category bipartite graph (Sec. II-A2) as depicted in Fig. 2 is constructed from the query-click logs over $\mathcal{C}$, collected over a set of $55,000$ popular queries that had the highest query volume over a 6 month duration (Jul-Dec 2015). The taxonomy portion of the data is publicly available. The number of leaf nodes in the taxonomy for the period under study, $|\mathcal{C}|$, was 986. Each leaf node in the customer-facing taxonomy is a child of one of the 149 parent nodes which form the implicit partition.

We study the islands of interest generated by IslandComms and IslandClusts. In both cases, we vary the number of islands, $K$, in the range 90–280. We use the implicit partition as a baseline. We try to characterize the islands produced by our two algorithms in terms of the four measures—locality, flow, girth, and expansion coefficient—introduced in Sec. IV.

Value of $K$ is experiment specific. For IslandComms, $K$ is specified implicitly by specifying $\theta$ and we use $t = 4$ for all our experiments, without loss of generality[4] . For IslandClusts, $K$ is specified explicitly.

## VI. Empirical Findings

### A. Specificity and similarity in islands of interest

Table I shows a sample of islands of interest having the highest click volume that are mined by IslandComms and IslandClusts. We have chosen $K = 156$ (directly for IslandClusts and indirectly by choosing $\theta$ for IslandComms). This is done in order to make the number of islands mined algorithmically close to the number of islands present in the implicit partition. We outline our observations below.

1) Islands from IslandComms tend to have large variation in sizes, whereas those from IslandClusts tend to have more even sizes.
2) Islands mined by IslandComms have a tendency to club together categories which are somewhat distinct but may have high similarity in user intent space.
3) Relatedly, Islands from IslandClusts may bifurcate islands discovered by IslandComms. e.g. Island 2 from IslandComms is a general "beauty" related island. IslandClusts represents these categories in terms of two beauty islands (3, 4)— one centered exclusively on massage and natural therapy, whereas the other is centered on hair treatment and skin cosmetic procedures. Similarly, some of the leaf nodes from the general "home improvement" island (1) from IslandComms seem to have been decomposed into a "service contractor" island (2) and a "floor cleaning/repair" (1).
4) We find that, in general, the girth of the islands discovered by IslandComms is larger than those discovered by IslandClusts, which is indicative of more diversity in the leaf nodes. Interestingly however, the expansion coefficient of the islands from IslandClusts is higher in many cases, which indicates that it selects fewer sister nodes from the implicit partition.

[2]We can have high locality with low flow; for instance, when multiple similar leaf nodes in $\mathcal{C}$ are partitioned into separate components.

[3]Note that the girth and expansion coefficient of all components in the implicit partition is one.

[4]The authors in [9] recommend $t$ to be in the range 3–8, and we observed our results to be stable with respect to the choice of $t$ in the range 4–16.

TABLE I: Examples of high click volume islands of interest mined using ISLANDCOMMS ($\theta = 0.1$) and ISLANDCLUSTS ($K = 156$) along with their respective girth ($\rho$) and expansion coefficient ($\eta$). Note the similarity between island 1 from ISLANDCOMMS with islands 1 and 2 from ISLANDCLUSTS; likewise, islands 3 and 4 with island 2 from ISLANDCOMMS.

**Islands of Interest mined by ISLANDCOMMS**

| | Leaf Categories | $\rho$ | $\eta$ |
|---|---|---|---|
| 1. | hand tools, window repair, masonry contractors, air conditioning, cabinet refinishing, bathroom design, kitchen design, cookware, freezers, ceiling fan installation, light fixture, interior painting, house painting, patio furniture, toilet installation, laundry, carpet cleaning, mattress cleaning, mattress protectors, gutter cleaning, furnace repair, power tools, water heater installation, electrician, roofing contractor, closet | 3.02 | 2.33 |
| 2. | natural medicine, tea and lemonade, weight loss, nutritionist, acupuncture, massage, reflexology, shiatsu, facial, day spa, sauna, dental, family doctor, lasik, cosmetic procedures, hair restoration, facial peel, podiatrist, nail salons, mani-pedi, hair perm, tattoo, make-up, dermatologist | 2.47 | 2.60 |
| 3. | cocktail mixers, wine coolers, wine bars, wineries, festivals, coffee and treats, ground coffee, tea, coffee capsules, cafes, karaoke bars, whale watching, concerts, pop music, pizza, sushi, thai restaurants, seafood restaurants, mexican restaurants, french restaurants, italian restaurants, cuban restaurants, hot chocolate, pubs, theater and shows, sailing, tours, travel magazines, comedy clubs, amusement parks, rental services, museums, ice skating, acting classes | 2.97 | 2.84 |
| 4. | rowing, pilates, health clubs, yoga, fitness classes, gymnastics, dance classes, crossfit, personal trainer, boxing, spinning, zumba, gyms, martial arts | 2.23 | 4.67 |
| 5. | cell phone accessories, chargers and adaptors, cell phones, phone cases | 1.00 | 1.75 |

**Islands of interest mined by ISLANDCLUSTS**

| | Leaf Categories | $\rho$ | $\eta$ |
|---|---|---|---|
| 1. | carpet cleaning, hardwood floor cleaning, tile cleaning, carpet installation, carpet repair, hardwood floor repair, hardwood floor cleaning, baseboards | 1.70 | 2.63 |
| 2. | carpenters, general contractors, handyman, electricians, drywall repair, ceiling fan installation, light fixture, outlet installation, toilet installation, bathroom remodeling, kitchen remodeling | 1.80 | 3.00 |
| 3. | couples massage, deep tissue massage, hot stone massage, pre natal massage, thai massage, reflexology, chiropractor, natural medicine, spa, bath house | 1.99 | 2.50 |
| 4. | tattoo removal, hair removal, hair perm, med spa, dermatologist, plastic surgery | 2.35 | 6.36 |
| 5. | liquor and spirits, cocktail mixers, dive bars, pubs, wine bars, brazilian restaurants, cuban restaurants, french restaurants, italian restaurants, pizza, seafood restaurants | 2.19 | 2.44 |
| 6. | health clubs, pilates, personal trainer, spinning, yoga, zumba, boot camps, gyms, fitness classes | 1.96 | 7.64 |
| 7. | cell phones, phone cases, phone batteries, phone accessories, cable chargers and adaptors, tools and equipment | 1.52 | 3.43 |

TABLE II: Comparison of algorithmically generated partitions (using ISLANDCOMMS and ISLANDCLUSTS) with the implicit partition. The size of the partitions is kept similar to facilitate fair comparison.

| | **implicit partition** | **ISLANDCOMMS** ($\theta = 0.1$) | **ISLANDCLUSTS** ($k = 156$) |
|---|---|---|---|
| size ($K$) | 149 | 156 | 156 |
| locality ($\lambda$) | 0.13 | 0.09 | 0.41 |
| flow ($\xi$) | 0.88 | 0.39 | 0.66 |
| median girth ($\rho$) | 1.00 | 2.33 | 2.33 |

### B. Locality-flow trade-off in islands of interest

To study the locality-flow trade-off between ISLANDCOMMS and ISLANDCLUSTS, we vary the number of islands, $K$. In the case of ISLANDCOMMS, $K$ is varied implicitly by varying $\theta$ in the range $[0.02, 0.5]$ (the number of islands varies monotonically with $\theta$ in this range— see Fig. 3a(a)). For ISLANDCLUSTS, we specify the number of islands explicitly.

The results are shown in Fig. 3b(b). The results clearly indicate that while ISLANDCLUSTS trades off lower interest flow to achieve higher locality, ISLANDCOMMS tries to optimize for lower interest flow. This can also be observed in the nature of the communities (Table I) found by the two algorithms.

## VII. APPLICATIONS

### A. Taxonomy Design: How good is the implicit partition?

We compare the implicit partition with the partitions mined from ISLANDCOMMS and ISLANDCLUSTS. For a fair comparison, we choose parameter settings such that the number of islands $K$ is close to $K = 149$ from the implicit partition. Results are shown in Table. II. The implicit partition is *sub-optimal* with respect to both locality and flow, as the taxonomy partition generated by ISLANDCLUSTS not only has a lower interest flow, but more than three times the locality. The partition from ISLANDCOMMS sacrifices locality to achieve a much lower flow which is a $56\%$ reduction from the flow in the implicit partition. Finally, all the algorithmically generated partitions are quite "far" (in terms of taxonomy distance) from the implicit partition, as indicated by their median girth. [5]

### B. Facility Allocation for Inventory: Can islands of interest inform inventory placement?

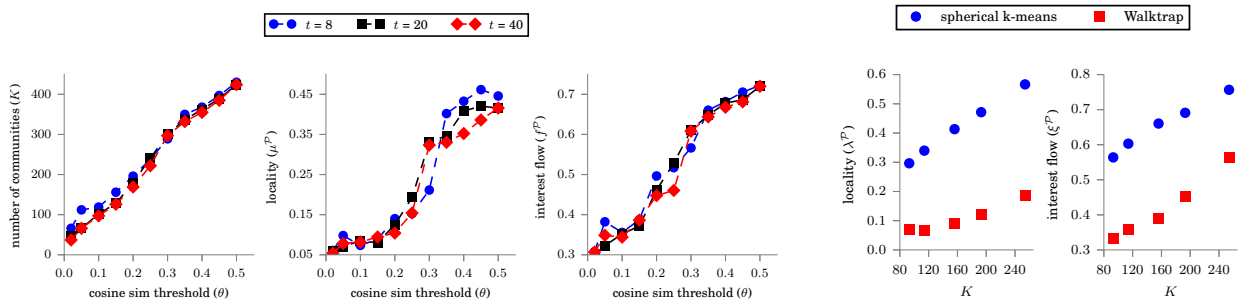
Items belonging to categories within an island correspond to similar purchase intent and have a higher likelihood of being co-purchased, and should therefore be co-located. For instance, items in categories "cell phones", "cell phone accessories", and "phone cases" (Island (5) from ISLANDCOMMS) are likely to be co-purchased together. Hence, placing these categories together will result in reduced processing and shipping time and costs.

### C. Recommendations for Browse: Can Islands of Interest inform item recommendations?

If a user has expressed interest in a category either through explicitly rating, or through implicitly browsing or purchasing items in that category, then we can recommend items from remaining categories in that island. For instance, users showing an interest in "window repair" category are also likely to be interested in "hand tools"(Island (1) from ISLANDCOMMS). Such islands of interest can be used to drive widgets such as "related items", "you may also like", "suggestions based on your last purchase" which are widely used to provide recommendations.

Another way these islands can be used to provide recommendations is through extending collaborative filtering recommender systems. User interests are projected over island space using implicit user signals such as click or purchase activities, thus generating user profile vector in an interpretable, low-dimensional and disentangled space. Then for a given user,

[5]Note that the depth of the taxonomy, $\Delta$ is four and the high value of their median girth shows that, on average for any two leaf nodes in the algorithmically generated partitions, we would need to move up more than two levels to find a common ancestor.




(a) (b)

Fig. 3: (a) Variation in number of communities, locality and interest flow as the threshold $\theta$ in the cosine similarity projection is varied keeping the number of steps in the IslandComms algorithm $t$ fixed. (b) Variation of locality and interest flow with the number of islands $K$ in IslandComms and IslandClusts.

other users with like user interest can be identified and recommendations made using standard collaborative filtering techniques.

Furthermore, we can tune the "discovery vs. specificity" of recommendations by an appropriate choice of algorithm. As noted earlier, IslandComms tends to facilitate discovery since islands identified are relatively more diverse, whereas IslandClusts tends to facilitate specificity. In addition, tuning the number of islands in each algorithm tends to make each island more specific in user intent.

## VIII. Discussion and Conclusions

In this paper, we construct *islands of interest*—these are sets of leaf nodes of the product taxonomy in e-commerce corresponding to similar user purchase intent. In order to mine islands of interest, we have used an assortment of data mining techniques to tap search data, which is arguably the fastest growing component of e-commerce data. We provide two algorithms—IslandComms and IslandClusts—to mine islands of interest. We motivate this new data mining construct using three core e-commerce applications: taxonomy design, recommendations, inventory location.

In order to analyze and characterize islands of interest in a way that is insightful to these applications, we construct a framework of four measures: locality, flow, girth, and expansion coefficient. Then, we vary parameter values over ranges, and analyze the properties of the resulting islands from IslandComms and IslandClusts using the above four measures. We found that IslandComms and IslandClusts trade off locality and flow very differently. For the same number of islands, IslandClusts optimizes for high locality, whereas IslandComms optimizes for low interest flow.

These properties can be tied back to our motivating examples. For instance, we found algorithmically generated partitions that achieve higher locality and lower interest flow when compared to the implicit partition. Furthermore, these islands have high girth which shows that they are not minor rearrangements of the product taxonomy. We also demonstrated that islands of interest may be used to replace taxonomic closeness as the building block of recommendations. In future work, we plan to use islands of interest to model users as mixture models over islands of interest.

## References


[1] P. Boldi, F. Bonchi, C. Castillo, D. Donato, and S. Vigna. Query suggestions using query-flow graphs. In *Proceedings of the 2009 Workshop on Web Search Click Data*, WSCD '09, pages 56–63, New York, NY, USA, 2009. ACM.

[2] S. Brin, R. Motwani, and C. Silverstein. Beyond market baskets: Generalizing association rules to correlations. *SIGMOD Rec.*, 26(2):265–276, June 1997.

[3] N. Craswell and M. Szummer. Random walks on the click graph. In *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '07, pages 239–246, New York, NY, USA, 2007. ACM.

[4] A. S. Das, M. Datar, A. Garg, and S. Rajaram. Google news personalization: Scalable online collaborative filtering. In *Proceedings of the 16th International Conference on World Wide Web*, WWW '07, pages 271–280, New York, NY, USA, 2007. ACM.

[5] I. S. Dhillon and D. S. Modha. Concept decompositions for large sparse text data using clustering. *Mach. Learn.*, 42(1-2):143–175, Jan. 2001.

[6] Q. Mei, D. Zhou, and K. Church. Query suggestion using hitting time. In *Proceedings of the 17th ACM Conference on Information and Knowledge Management*, CIKM '08, pages 469–478, New York, NY, USA, 2008. ACM.

[7] D. Melamed. Community structures in bipartite networks: A dual-projection approach. *PLoS ONE*, 9(5):e97823, 05 2014.

[8] M. E. J. Newman. Scientific collaboration networks. i. network construction and fundamental results. *Phys. Rev. E*, 64:016131, Jun 2001.

[9] P. Pons and M. Latapy. Computing communities in large networks using random walks. *J. of Graph Alg. and App. bf*, 10:284–293, 2004.

[10] N. Pradhan, V. Deolalikar, and K. Li. Atypical queries in ecommerce. In *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*, CIKM '15, pages 1767–1770, New York, NY, USA, 2015. ACM.

[11] X. Su and T. M. Khoshgoftaar. A survey of collaborative filtering techniques. *Advances in artificial intelligence*, 2009:4, 2009.

[12] H. Wang, N. Wang, and D.-Y. Yeung. Collaborative deep learning for recommender systems. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '15, pages 1235–1244, New York, NY, USA, 2015. ACM.

[13] T. Zhou, J. Ren, M. c. v. Medo, and Y.-C. Zhang. Bipartite network projection and personal recommendation. *Phys. Rev. E*, 76:046115, Oct 2007.

[14] C.-N. Ziegler, G. Lausen, and L. Schmidt-Thieme. Taxonomy-driven computation of product recommendations. In *Proceedings of the Thirteenth ACM International Conference on Information and Knowledge Management*, CIKM '04, pages 406–415, New York, NY, USA, 2004. ACM.

[15] K. Zweig and M. Kaufmann. A systematic approach to the one-mode projection of bipartite graphs. *Social Network Analysis and Mining*, 1(3):187–218, 2011.