InterestMap: Harvesting Social Network Profiles for Recommendations

Hugo Liu
MIT Media Laboratory
20 Ames St., Cambridge, MA, USA
hugo@media.mit.edu

Pattie Maes
MIT Media Laboratory
20 Ames St., Cambridge, MA, USA
pattie@media.mit.edu

ABSTRACT

While most recommender systems continue to gather detailed models of their "users" within their particular application domain, they are, for the most part, oblivious to the larger context of the lives of their users outside of the application. What are they passionate about as individuals, and how do they identify themselves culturally? As recommender systems become more central to people's lives, we must start modeling the person, rather than the user.

In this paper, we explore how we can build models of people outside of narrow application domains, by capturing the traces they leave on the Web, and inferring their everyday interests from this. In particular, for this work, we harvested 100,000 social network profiles, in which people describe themselves using a rich vocabulary of their passions and interests. By automatically analyzing patterns of correlation between various interests and cultural identities (*e.g.* "Raver," "Dog Lover," "Intellectual"), we built InterestMap, a network-style view of the space of interconnecting interests and identities. Through evaluation and discussion, we suggest that recommendations made in this network space are not only accurate, but also highly visually intelligible – each lone interest contextualized by the larger cultural milieu of the network in which it rests.

Keywords

User modeling, person modeling, recommender systems, item-item recommendation, social networks, collaborative filtering, cultural visualization.

1. INTRODUCTION

Recommender systems (*cf.* Resnick & Varian, 1997) have thus far enjoyed remarkable practical and commercial success. They have become a mainstay of e-commerce sites such as Amazon and Ebay for product recommendation; and recommenders have also been deployed to cater to subject domains such as books, music, tutoring, movies, research papers, and web pages. Most recommenders operate within a single application domain, and are powered by domain-specific data – either through explicitly given user profiles, or through implicitly gathered models of user behavior within the application framework. But why should recommenders be restricted to data gathered within the context of the application?

Enter the Web. Web-based communities are quite social and dynamic places – there are online chat forums, blogging and journaling sites, "rings" of personal web pages, and social network communities. In all of these communities, recommendations are happening "in the wild," all of the time. With natural language processing and a bit of finesse, we might hope to harvest information from these sources and use them to construct richer models of people, of communities and their cultures, and to power new kinds of recommender systems whose recommendations are sourced from online trends and *word-of-mouth*. The idea that recommendations could be sourced from traces of social activity follows from Terveen & Hill (2001), who refer to their approach as *social data mining*. They have looked at mining web page recommendations from Usenet messages, and through the structural analysis of web pages.

In this work, we turn to web-based social networks such as Friendster[1], Orkut[2], and MySpace[3] as a source of recommendations for a broad range of interests, *e.g.* books, music, television shows, movies, sports, foods, and more. On web-based social networks, people not only specify their friends and acquaintances, but they also maintain an explicit, self-crafted run-down of their interests and passions, inputted through free-form natural language. Having harvested 100,000 of these social network profiles, we apply natural language processing to ground interests into vast ontologies of books, music, movies, etc. We also mine out and map a category of special interests, called "passions," into the space of social and cultural identities (e.g. "Book Lover," "Raver," "Rock Musician"). By analyzing patterns of how these interests and identities co-occur, we automatically generated a network-style "map" of the affinities between different interests and identities, which we call an InterestMap. By spreading activation over the network (Collins & Loftus, 1975), InterestMap can be applied directly to make interest recommendations; due to InterestMap's unique network topology, we show that recommendations produced by this method incorporates factors of identity and taste. Outside of recommendation, we are also exploring other applications for InterestMap, such as marketing and matchmaking.

This paper is structured as follows. First, we discuss the source and nature of the corpus of social network profiles used to build

Workshop: Beyond Personalization 2005
IUI'05, January 9, 2005, San Diego, California, USA
http://www.cs.umn.edu/Research/GroupLens/beyond2005

[1] http://www.friendster.com

[2] http://www.orkut.com

[3] http://www.myspace.com

InterestMap. Second, we outline the approach and implementation of InterestMap. Third, we describe how InterestMap may be used for recommendations, and we present an evaluation of the system's performance under this task. Fourth, we give further discussion for some lingering issues – the tradeoffs involved in using social network profiles to drive recommendations; and the implications of InterestMap's network-style representation for explainability and trust. We conclude with a greater vision for our work.

2. SOCIAL NETWORK PROFILES

The recent emergence and popularity of web-based social network software (*cf.* boyd, 2004; Donath & boyd, 2004) such as Friendster, Orkut, and MySpace can be seen as a tremendous new source of subject domain-independent user models, which might be more appropriately termed, *person models* to reflect their generality. To be sure, well over a million self-descriptive personal profiles are available across different web-based social networks.

While each social network's profile has an idiosyncratic representation, the common denominator across all the major web-based social networks we have examined is a *category-based representation* of a person's broad interests, with the most common categories being music, books, movies, television shows, sports, and foods. Within each interest category, users are generally unrestricted in their input, but typically enumerate lists of items, given as fragments of natural language. Even within a particular category, these items may refer to different things; for example, under "books," items may be an author's last name, a book's title, or some genre of books like "mystery novels," so there may be some inference necessary to map these natural language fragments into a normalized ontology of items. Figure 1 shows the structure and contents of a typical profile on the Orkut social network.

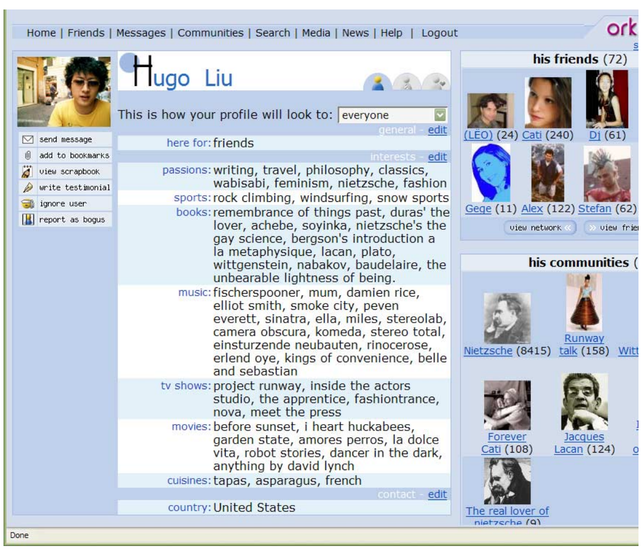

Figure 1. A screenshot of a typical profile taken from the Orkut social network. The interest categories shown here are typical to most web-based social network profile templates.

Note also in Figure 1, that there is a special category of interests called "passions." Among the social network profile templates we have examined, *all* of them have this special category, variously called "general interests," "hobbies & interests," or "passions." Furthermore, this special category always appears above the more specific interest categories, as it does in Figure 1, perhaps to encourage the thinking that these passions are more general to a person than other sorts of interests, and is more central to one's own self-concept and self-identification.

When mining social network profiles, we distinguish passions from other categories of more specific interests. With the hypothesis that passions speak directly to a person's social and cultural identity, we map the natural language items which appear under this category into an ontology of *identity descriptors*. For example, "dogs" maps into "Dog Lover," "reading" maps into "Book Lover", "deconstruction" maps into "Intellectual." Items in the other categories are mapped into their respective ontologies of *interest descriptors*.

In the following section, we describe how these profiles were harvested, normalized and correlated to build InterestMap.

3. THE INTERESTMAP APPROACH

The general approach we took to build InterestMap consists of four steps: 1) mine social network profiles; 2) extract out a normalized representation by mapping casually-stated keywords and phrases into a formal ontology of *interest descriptors* and *identity descriptors*; 3) augment the normalized profile with metadata to facilitate connection-making (*e.g.* "War and Peace" also causes "Leo Tolstoy," "Classical Literature," and other metadata to be included in the profile, at a discounted value of 0.5, for example); and 4) apply a machine learning technique to learn the semantic relatedness weights between every pair of *descriptors*. What results is a gigantic semantic network whose nodes are identity and interest descriptors, and whose numerically weighted edges represent strengths of semantic relatedness. Below, we give an implementation-level account of this process.

3.1 Building a Normalized Representation

Between January and July of 2004, we mined 100,000 personal profiles from two web-based social network sites, recording only the contents of the "passions" category and common interest categories, as only these are relevant to InterestMap. We chose two social networks rather than one, to attempt to compensate for the demographic and usability biases of each. One social network has its membership primarily in the United States, while the other has a fairly international membership. One cost to mining multiple social networks is that there is bound to be some overlap in their memberships (by our estimates, this is about 15%), so these twice-profiled members may have disproportionately greater influence on the produced InterestMap.

To normalize the representation of each profile, we implemented 2,000 lines of natural language processing code in Python. First, for each informally-stated list of interests, the particular style of delimitation had to be heuristically recognized. Common delimiters were commas, semicolons, character sequences (*e.g.* " \…/"), new lines, commas in conjunction with the word "and," and so on. A very small percentage of these "lists" of interests were not lists at all, so these were discarded.

The newly segmented lists contained casually-stated keyphrases referring to a variety of things. They refer variously to authorship like a book author, a musical artist, or a movie director; to genre like "romance novels," "hip-hop," "comedies," "French cuisine"; to titles like a book's name, an album or song, a television show, the name of a sport, a type of food; or to any combination thereof, *e.g.* "Lynch's Twin Peaks," or "Romance like Danielle Steele."

To further complicate matters, sometimes only part of an author's name or a title is given, *e.g.* "Bach," "James," "Miles," "LOTR," "The Matrix trilogy." Then of course, the items appearing under "passions," can be quite literally anything.

For a useful InterestMap, it is not necessary to be able to recognize every item, although the greater the recognition capability, the more useful will be the resulting InterestMap. To recognize the maximal number and variety of items, we created a vast formal ontology of 21,000 interest descriptors and 1,000 identity descriptors compiled from various comprehensive ontologies on the web for music, sports, movies, television shows, and cuisines, including The Open Directory Project[4], the Internet Movie Database[5], TV Tome[6], Wikipedia[7], All Music Guide[8], and AllRecipes[9]. The ontology of 1,000 identity descriptors required the most intensive effort to assemble together, as we wanted them to reflect the types of passions talked about in our corpus of profiles; this ontology was taken mostly from The Open Directory Project's hierarchy of subcultures and hobbies, and finished off with some hand editing. To facilitate the classification of a "passions" item into the appropriate identity descriptor, each identity descriptor is annotated with a bag of keywords which were also mined out, so for example, the "Book Lover" identity descriptor is associated with, *inter alia,* "books," "reading," "novels," and "literature." To assist in the normalization of interest descriptors, we gathered aliases for each interest descriptor, and statistics on the popularity of certain items (most readily available in The Open Directory Project) which could be used for disambiguation (*e.g.* "Bach" → "JS Bach" or → "CPE Bach"?).

Using this crafted ontology of 21,000 interest descriptors and 1,000 identity descriptors, the heuristic normalization process successfully recognized 68% of all tokens across the 100,000 personal profiles, committing 8% false positives across a random checked sample of 1,000 mappings. We suggest that this is a good result considering the difficulties of working with free text input, and enormous space of potential interests and passions. Once a profile has been normalized into the vocabulary of descriptors, they are expanded using metadata assembled along with the formal ontology. For example, a book implies its author, and a band implies its musical genre. Descriptors generated through metadata-association are included in the profile, but at a discount of 0.5 (read: they only count half as much). The purpose of doing this is to increase the chances that the learning algorithm will discover latent semantic connections.

3.2 Learning the Map of Interests and Identities

From these normalized profiles, we wish to learn the overall strength of the semantic relatedness of every pair of descriptors, across all profiles, and use this data to build InterestMap's network graph. Our choice to focus on the similarities between descriptors rather than user profiles reflects an item-based recommendation approach such as that taken by Sarwar *et al.* (2001).

Technique-wise, the idea of analyzing a corpus of profiles to discover a stable network topology for the interrelatedness of interests is similar to how *latent semantic analysis* (Landauer, Foltz & Laham, 1998) is used to discover the interrelationships between words in the document classification problem. For our task domain though, we chose to apply an information-theoretic machine learning technique called *pointwise mutual information* (Church *et al.*, 1991), or PMI, over the corpus of normalized profiles. For any two descriptors f_1 and f_2, their PMI is given in equation (1).

$$PMI(f_1, f_2) = \log_2\left(\frac{\Pr(f_1 f_2)}{\Pr(f_1)\Pr(f_2)}\right) \qquad \textbf{(1)}$$

Looking at each normalized profile, the learning program judges each possible pair of descriptors in the profile as having a correlation, and updates that pair's PMI.

What results is a 22,000 x 22,000 matrix of PMIs. After filtering out descriptors which have a completely zeroed column of PMIs, and applying thresholds for minimum connection strength, we arrive at a 12,000 x 12,000 matrix (of the 12,000 descriptors, 600 are identity descriptors), and this is the complete form of the InterestMap. Of course, this is too dense to be visualized as a semantic network, but we have built less dense semantic networks from the complete form of the InterestMap by applying higher thresholds for minimum connection strength. Figure 2 is a visualization of a simplified InterestMap.

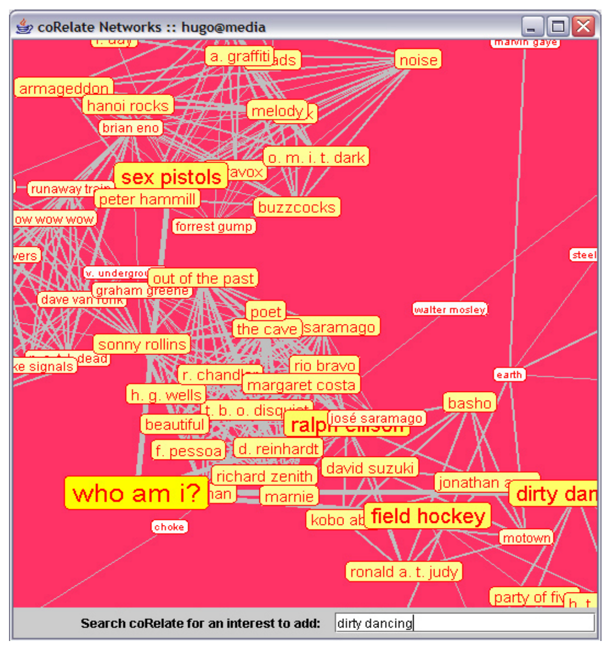

Figure 2. A screenshot of an interactive visualization program, running over a simplified version of InterestMap (weak edges are discarded, and edge strengths are omitted). The "who am i?" node is an indexical node around which a person is "constructed." As interests are attached to the indexical, correlated interests and identity descriptors are pulled into the visual neighborhood.

[4] http://www.dmoz.org

[5] http://www.imdb.com

[6] http://tvtome.com

[7] http://www.wikipedia.org

[8] http://www.allmusic.com

[9] http://allrecipes.com

3.3 Network Topology

Far from being uniform, the resultant InterestMap has a particular topology, characterized by two confluence features: *identity hubs,* and *taste cliques.*

Identity hubs are *identity descriptor nodes* which behave as "hubs" in the network, being more strongly related to more nodes than the typical *interest descriptor node.* They exist because the ontology of identity descriptors is smaller and less sparse than the ontology of interest descriptors; each identity descriptor occurs in the corpus on the average of 18 times more frequently than the typical interest descriptor. In InterestMap, identity hubs serve an *indexical* function. They give organization to the forest of interests, allow interests to cluster around identities. What kinds of interests do "Dog Lovers" have? This type of information is represented explicitly by identity hubs.

Another confluence feature is a *taste clique*. Visible in Figure 2, for example, we can see that "Sonny Rollins," is straddling two cliques with strong internal cohesion. While the identity descriptors are easy to articulate and can be expected to be given in the special interests category of the profile, tastes are often a fuzzy matter of aesthetics and may be harder to articulate using words. For example, a person in a Western European taste-echelon may fancy the band "Stereolab" and the philosopher "Jacques Derrida," yet there may be no convenient keyword articulation to express this. However, when the InterestMap is learned, cliques of interests seemingly governed by nothing other than taste clearly emerge on the network. One clique for example, seems to demonstrate a Latin aesthetic: "Manu Chao," "Jorge Luis Borges," "Tapas," "Soccer," "Bebel Gilberto," "Samba Music." Because the cohesion of a clique is strong, *taste cliques* tend to behave much like a singular identity hub, in its impact on network flow.

In the following section, we discuss how InterestMap may be used for recommendations, and evaluate the impact that identity hubs and taste cliques have on the recommendation process.

4. USING INTERESTMAP FOR RECOMMENDATIONS

InterestMap can be applied in a simple manner to accomplish several tasks, such as identity classification, interest recommendation, and interest-based matchmaking. A unique feature of InterestMap recommendations over straight interest-item to interest-item recommendations is the way in which identity and tastes are allowed to exert influence over the recommendation process. The tail end of this section describes an evaluation which demonstrates that identity and taste factors can improve performance in an interest recommendation task.

4.1 Finding Recommendations by Spreading Activation

Given a seed profile which represents a new user, the profile is normalized into the ontology of interest descriptors and identity descriptors, as described in Section 3.1. The normalized profile is then mapped onto the nodes of the InterestMap, leading to a certain activation pattern of the network.

With InterestMap, we view interest recommendation as a *semantic context* problem. By spreading activation (Collins & Loftus, 1975) outward from these seed nodes, a surrounding neighborhood of nodes which are connected strongly to the seed nodes emerges. As the distance away from the seed nodes increases (in the number of hops away), activation potential decays according to some discount (we having been using a discount of 0.75). The semantic neighborhood defined by the top *N* most related interest descriptor nodes corresponds with the top *N* interest recommendations produced by the InterestMap recommender.

Another straightforward application of InterestMap is identity classification. A subset of the semantic neighborhood of nodes resulting from spreading activation will be identity descriptor nodes, so the most proximal and strongly activated of these can be thought of as recognized identities. Identity classification with InterestMap can be useful in marketing applications because it allows a distributed interest-based representation of a person to be summarized into a more concise demographic or psychographic grouping.

Finally, we are experimenting with InterestMap for interest-based matchmaking, which may be useful for making social introduction. To calculate the affinity between two people, two seed profiles lead to two sets of network activations, and the strength of the contextual overlap between these two activations can be used as a coarse measure of how much two people have in common.

4.2 Evaluation

We evaluated the performance of spreading activation over InterestMap in the interest recommendation task. In this evaluation, we introduced three controls to assess two particular features: 1) the impact that identity hubs and taste cliques have on the quality of recommendations; and 2) the effect of using spreading activation rather than a simple tally of PMI scores. In the first control, identity descriptor nodes are simply removed from the network, and spreading activation proceeds as usual. In the second control, identity descriptor nodes are removed, and *n*-cliques[10] where $n>3$ are weakened[11]. The third control does not do any spreading activation, but rather, computes a simple tally of the PMI scores generated by each seed profile descriptor for each of the 11,000 or so interest descriptors. We believe that this successfully emulates the mechanism of a typical non-spreading activation item-item recommender because it works as a pure information-theoretic measure.

We performed five-fold cross validation to determine the accuracy of InterestMap in recommending interests, versus each of the three control systems. The corpus of 100,000 normalized and metadata-expanded profiles was randomly divided into five segments. One-by-one, each segment was held out as a test corpus and the other four used to either train an InterestMap using PMI.

Within each normalized profile in the test corpus, a random half of the descriptors were used as the "situation set" and the remaining half as the "target set." Each of the four test systems uses the situation set to compute a *complete recommendation*— a rank-ordered list of all interest descriptors; to test the success of this recommendation, we calculate, for each interest descriptor in the target set, its percentile ranking within the complete recommendation list. The overall accuracy of recommendation is the arithme-

---

[10] a qualifying clique edge is defined here as an edge whose strength is in the 80th percentile, or greater, of all edges

[11] by discounting a random 50% subset of the clique's edges by a Gaussian factor (0.5 *mu,* 0.2 *sigma*).

tic mean of the percentile scores generated for each interest descriptor of the target set.

We opted to score the accuracy of a recommendation on a sliding scale, rather than requiring that descriptors of the target set be guessed exactly within *n* tries because the size of the target set is so small with respect to the space of possible guesses that accuracies will be too low and standard errors too high for a good performance assessment. For the InterestMap test system and control test systems #1 (Identity OFF) and #2 (Identity OFF and Taste WEAKENED), the spreading activation discount was set to 0.75). The results of five-fold cross validation are reported in Figure 3.

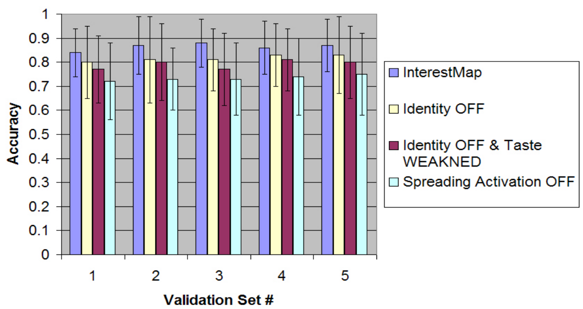

Figure 3. Results of five-fold cross-validation of InterestMap and three control systems on a graded interest recommendation task.

The results demonstrate that on average, the original InterestMap recommended with an accuracy of 0.86. In control #1, removing identity descriptors from the network not only reduced the accuracy to 0.81, but also increased the standard error by 38%. In control #2, removing identity descriptors and weakening cliques further deteriorated accuracy slightly, though insignificantly, to 0.79. When spreading activation was turned off, neither identity hubs nor taste cliques could have had any effect, and we believe that is reflected in the lower accuracy of 73%. However, we point out that since control #3's standard error has not worsened, its lower accuracy should be due to overall weaker performance across all cases rather than being brought down by exceptionally weak performance in a small number of cases.

We believe that the results demonstrate the advantage of spreading activation over simple one-step PMI tallies, and the improvements to recommendation yielded by identity and taste influences. Because activation flows more easily and frequently through identity hubs and taste cliques than through the typical interest descriptor node, the organizational properties of identity and taste yield proportionally greater influence on the recommendation process; this of course, is only possible when spreading activation is employed.

5. DISCUSSION

In this section, we discuss some of the advantages and disadvantages of using social network profiles to drive recommendations, and implications of InterestMap's network-style view of a space for trust and explainability.

5.1 Tradeoffs in Using Social Network Profiles to Drive Recommendations

The harvesting of social network profiles for recommendations involves several important tradeoffs to be considered.

Fixed Ontology versus Open-ended Input. While domain-specific behavior-based recommenders model user behavior over a predetermined ontology of items (*e.g.* a purchase history over an e-commerce site's ontology of products; a rating history over an application's ontology of movies), items specified in a social network profile are open-ended. Granted that in the normalization of a profile, items will have to be eventually normalized into an ontology, there still remains the psychological priming effects of a user working over the artifacts of a fixed ontology as he/she is composing ratings. For example, in a movie domain, a user may choose to rate a movie because of the way the movies are browsed or organized, and may find movies to rate which the user has long forgotten and is surprised to see in the ontology. In filling out the movies category in a social network profile, there is no explicit display of a movie ontology to influence user input, and a user could definitely not input movies which he/she has long since forgotten.

This generates the following tradeoff: Recommenders based on domain-specific behaviors will be able to recommend a greater variety of items than open-ended input based recommenders, including the more obscure or not entirely memorable items, because the application's explicit display of those items will remind a user to rate them. On the other hand, open-ended input may tend to recommend items which are more memorable, more significant, or possessing greater communicative value. This is especially true for social network profiles, where users have an explicit intention to communicate who they are through each of the interests descriptors they specify. We suggest that high-communicative value adds a measure of fail-softness to recommendations. For example it might be easier to rationalize or forgive the erroneous recommendation of a more prominent item like "L.v. Beethoven's Symphony No. 5" to a non-classical-music-lover than an equally erroneous recommendation of a more obscure or arbitrary feature like "Max Bruch's Op. 28."

Socially Costly Recommendation. The social cost paid by a user in producing a "rating" can greatly affect the quality and nature of recommendations. To begin with, in some domain-specific behavior-based recommender systems, the profile of user behavior is gathered implicitly and this profile is kept completely private. Here there is no cost paid by a user in producing a "rating." In a second case, some domain-specific recommender systems make their users' ratings publicly viewable. The introduction of the publicity dimension is likely to make a user more conscious about the audience for the ratings, and more careful about the ratings that he/she produces; thus, the user is paying some social cost, and as a result, we might expect the ratings to be less arbitrary than otherwise. Employing this same logic for monetary cost, we might expect Amazon recommendations based on purchases to be less arbitrary that recommendations based on products viewed.

Thirdly, in the case of social network profiles, the greatest cost is paid by a user in listing an item in his/her profile. Not only is the profile public, but it is viewed by exactly the people whose opinions the user is likely to care about most – his/her social circle; Donath & boyd (2004) report that a person's presentation of self in profiles is in fact a strategic communication and social signalling game. Items chosen for display are not just any subset of possessed interests, but rather, are non-arbitrary items meant to be representative of the self; furthermore, users may consciously intend to socially communicate those items to their social circle.

The social cost dimension to recommendation produces another interesting tradeoff. The higher the social cost paid by the user in producing a rating, the more deliberate the ratings. So we can anticipate recommendations made using this data to be consistent in their social apropos. On the other hand, social stigma will tend to suppress the rating and recommendation of malapropos items, for instance, perhaps the cost of listing the publicly-derided but oft privately-appreciated "Britney Spears" in one's profile is prohibitively high. Of course, these social pressures also manifest in real-life social recommendations, and the thought of recommending "Britney Spears" to someone you are not very comfortable with may be just as dissuasive.

5.2 Impact of Network-Style Views on Explainability and Trust

That a user trusts the recommendations served to him by a recommender system is important if the recommender is to be useful and adopted. Among the different facilitators of trust, Wheeless & Grotz (1977) identify transparency as a prominent desirable property. When a human or system agent discloses its assumptions and reasoning process, the recipient of the recommendation is likely to feel less apprehensive toward the agent and recommendation. Also in the spirit of transparency, Herlocker *et al.* (2000) report experimental evidence to suggest that recommenders which provide explanations of its workings experience a greater user acceptance rate than otherwise.

Unlike opaque statistical mechanisms like collaborative filtering (Shardanand & Maes, 1995), InterestMap's mechanism for recommendation can be communicated visually as a large network of interests and identities. The cliques and idiosyncratic topology of this *fabric of interests* visually represents the common tendencies of a large group of people. For example, in Figure 2, it is plain to see that "Sonny Rollins" and "Brian Eno" are each straddling two different cliques of different musical genres. The rationale for each recommendation, visually represented as the spreading of flow across the network, is easily intelligible. Thus it may be easier for a user to visually contextualize the reasons for an erroneous recommendation, *e.g.* "I guess my off-handed taste for Metallica situated me in a group of metal heads who like all this other stuff I hate."

The ability to interact with the InterestMap network space may also afford the system an opportunity to *learn* more intelligently from user feedback about erroneous recommendations. Rather than a user simply stating that she did not like a particular recommendation, she can black out or deprecate particular clusters of the network which she has diagnosed as the cause of the bad recommendation, *e.g.* "I'll black out all these taste cliques of heavy metal and this identity hub of "Metal Heads" so the system will not make *that* mistake again." Although we have not yet implemented such a capability in InterestMap, we hope to do so shortly.

6. CONCLUSION

As recommender systems play ever-larger roles in people's lives, providing serendipitous suggestions of things to do and people to meet, recommendation technology will have to be based on something other than domain-specific knowledge, which is facing a semantic interoperability crisis. To some degree, we will have to abandon *user modeling* in favor of *person modeling*, and *cultural modeling*. We hope that the work presented in this paper begins to illustrate a path in this direction. By harvesting the traces of how people behave *in the wild* on the Web and on their computers, we can build a more general model of their person. By looking at interests within the context of emergent cultural patterns, we find new bases for recommendation, driven by cultural identities, and modes of taste. And the best part of this new paradigm for recommendation is that it would be more intelligible and transparent to people, for we, as persons, are already well-equipped to understand interests in the context of cultural milieus.

7. ACKNOWLEDGMENTS

The authors would like to thank the blind reviewers for their helpful feedback, and Judith Donath for inspiring our social perspective on this work.

8. REFERENCES

[1] danah boyd: 2004, Friendster and publicly articulated social networks. *Conference on Human Factors and Computing Systems (CHI 2004).* ACM Press.

[2] K.W. Church, W. Gale, P. Hanks, and D. Hindle: 1991, Using statistics in lexical analysis. In *Uri Zernik (ed.), Lexical Acquisition: Exploiting On-Line Resources to Build a Lexicon,* pp. 115-164. New Jersey: Lawrence Erlbaum, 1991

[3] A.M. Collins, and E.F. Loftus: 1975, A spreading-activation theory of semantic processing. *Psychological Review, 82*, pp. 407-428

[4] Judith Donath and danah boyd: 2004, Public displays of connection, *BT Technology Journal 22(4):*pp. 71-82. Kluwer

[5] J. Herlocker, J. Konstan and J. Riedl: 2000, Explaining Collaborative Filtering Recommendations. *Conference on Computer Supported Cooperative Work*, pp. 241-250

[6] T.K. Landauer, P.W. Foltz, D. Laham: 1998, An introduction to Latent Semantic Analysis . *Discourse Processes, 25,* 259-284.

[7] P. Resnick and H.R. Varian: 1997, Recommender Systems. *Communications of the ACM, Vol. 40(3):*pp. 56-58.

[8] B.M. Sarwar *et al.*: 2001, Item-Based Collaborative Filtering Recommendation Algorithms. *10th Int'l World Wide Web Conference*, ACM Press, pp. 285-295.

[9] U. Shardanand and P. Maes: 1995, Social information filtering: Algorithms for automating `word of mouth'. *Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems*, pp. 210-217.

[10] L.G. Terveen and W.C. Hill: 2001, Beyond Recommender Systems: Helping People Help Each Other, in *Caroll, J. (ed.), HCI In The New Millennium*. Addison-Wesley

[11] L. Wheeless and J. Grotz: 1977, The Measurement of Trust and Its Relationship to Self-disclosure. *Communication Research 3(3)*, pp. 250-257.