Constructing Recommendation Systems for Effective Health Messages Using Content, Collaborative, and Hybrid Algorithms

By
JOSEPH N. CAPPELLA,
SIJIA YANG,
and
SUNGKYOUNG LEE

Theoretical and empirical approaches to the design of effective messages to increase healthy and reduce risky behavior have shown only incremental progress. This article explores approaches to the development of a "recommendation system" for archives of public health messages. Recommendation systems are algorithms operating on dense data involving both individual preferences and objective message features. Their goal is to predict ratings for items (i.e., messages) not previously seen by the user on content similarity, prior preference patterns, or their combination. Standard approaches to message testing and research, while making progress, suffer from very slow accumulation of knowledge. This article seeks to leapfrog conventional models of message research, taking advantage of modeling developments in recommendation systems from the commercial arena. After sketching key components in developing recommendation algorithms, this article concludes with reflections on the implications of these approaches in both theory development and application.

Keywords: recommendation algorithms; content-based approaches; collaborative filtering; hybrid approaches; health message design

The success of Amazon and Netflix has drawn wide attention to the underlying recommendation algorithms deployed to generate user-specific predictions regarding other products the user would like. The capacity of these recommendation systems to automatically sift through product records and select a short list that meets a target user's needs, while doing so simultaneously for millions of others, makes the approach worthy of investigation by communication scholars interested in message tailoring.

Joseph N. Cappella is the Gerald R. Miller Professor of Communication at the Annenberg School for Communication at the University of Pennsylvania. His research focuses on health and political communication and the ways that effective and ineffective messages function in these arenas.

DOI: 10.1177/0002716215570573

Online recommendation systems encompass a wide variety of approaches to providing evaluative information about products and services. These currently include open-ended commentaries, product reviews, simple likes and dislikes, star ratings, preferences, behavioral selections (such as views of YouTube videos), and even ratings of the utility of the recommendations themselves (McAuley and Leskovec 2013; Schafer, Konstan, and Riedl 2001). The presence of recommendations accompanying products, services, and informational entities fundamentally changes the way the public information environment must be studied (Bennett and Iyengar 2008; Jenkins 2006; Kim et al. 2013). For example, a set of news stories prepared and published by institutional sources now exists as news items plus recommendations so that their effects on audiences can no longer be equated to the effects of the news stories alone.

In this article, we focus on only one type of recommendation system, the often hidden algorithms that use a person's prior preferences and choices in a limited domain of cases to predict their choices—that is, making recommendations—for other items to which they have not yet been exposed. Two factors differentiate algorithmic recommendation systems from other forms of prediction using standard social scientific methods: behavioral data and individualized predictions. In both the Amazon and Netflix use of recommendation algorithms, the issue is not what people prefer and like or even what segments of people (men, women) like. Rather, based on what a specific person has selected previously (that is, viewed, moved to the shopping cart, or purchased), what will that same person likely select in the future from a range of possibilities not yet examined? Note that past behavior by the target person is required, and a prediction for that person on entities as yet unseen is the result. This goal is a lofty one indeed, but methods seeking to achieve it have been deployed and are being invented in an emerging media environment where preference data are extensive and widely available. In what follows, we present current practices in the process of developing recommendation algorithms illustrating their application to informational products such as public health messages and news stories.

Standard approaches to message testing and research, while making progress, suffer from very slow accumulation of knowledge, weak theory that fails to guide choices in a comprehensive way, and an infinite number of features that demand evaluation. Tailoring messages to individuals' needs is often treated as the gold standard for effectiveness (Rimer and Kreuter 2006). Yet the number of message features—from the trivial to the profound—on which tailoring can occur is overwhelming. This article seeks to leapfrog conventional models of message

Sijia Yang is a doctoral student at the Annenberg School for Communication at the University of Pennsylvania. He studies how the dynamics between message production/processing and social interactions produce collective outcomes such as misinformation spreading and curbing, public opinion formation, and moralization of previously moral-neutral entities in both political and health domains.

Sungkyoung Lee is an assistant professor in the School of Journalism at University of Missouri, Columbia. Her research focuses on understanding mediated message processing and designing effective health communication messages.

research by taking advantage of extensive developments in recommendation systems.

Although a fully functional recommendation system inevitably consists of various components operating together (e.g., a data management system, the design of the webpage layout, incorporation of user feedback such as ratings and reviews, etc.), it is the recommendation algorithms that we focus on here.

The work on recommendation algorithms in commercial applications falls into three categories: content of the items being recommended, collaborative recommendations based on patterns of user ratings (also known as social filtering), and their hybrid using both types of data (Adomavicius and Tuzhilin 2005; Almazro et al. 2010). This article will sketch key components in developing the algorithm for each category, drawing the implications of these approaches in both theory development and application.

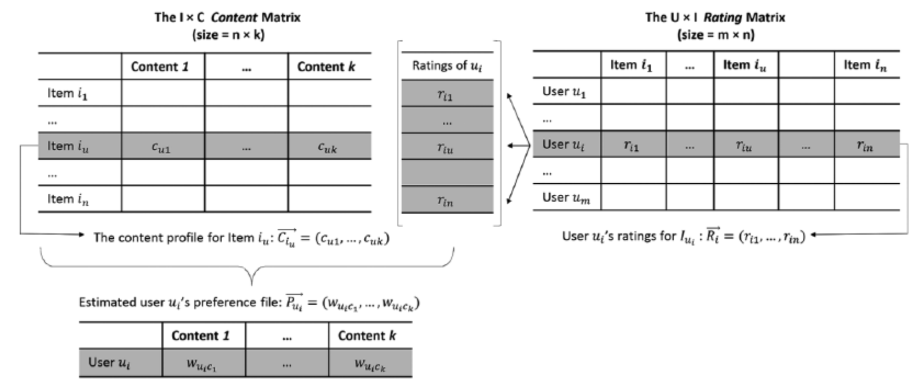
Approaches to Building Recommendation Systems

Two essential data matrices

The goal for any recommendation system is to estimate a target user's evaluation of an unseen item from a pool of items that have yet been seen by the target person. The system recommends to the user the top item(s) on the list of estimated rankings. To accomplish this task, the recommendation system typically utilizes two sets of information: descriptions of and evaluation from users on items in the system.

The content of items in the system can be described in a myriad of ways, an infinite number really. For instance, Netflix movies are categorized by production year; actors; director; and a variety of genres, ranging from general tags like "romantic dramas" to very specific niches like "violent thrillers about cats for ages 8 to 10" (Madrigal 2014). Content features stored in a well-structured format considerably speed up the process of identifying items similar to those that have already received positive ratings from the target user, and in turn, the system can recommend ones most closely mimicking the target user's past selections. This calculus, namely, to recommend items similar to the ones the user preferred in the past, is the fundamental principle of *content-based* recommendation systems (Lops, de Gemmis, and Semeraro 2011; Pazzani and Billsus 2007). With that said, items of interest to communication scientists—such as news articles, health-related public service announcements (PSAs), video clips, and so on—typically come unstructured and require either manual or computer-assisted coding to transform them into suitable data formats for mathematical calculation. The result is an item profile where each feature serves as a unique "variable" to describe one aspect of the item. In the item (row) by content feature (column) matrix (see Figure 1), each item is treated as a separate case, which is mathematically equivalent to a vector located in a high-dimensional feature space.

FIGURE 1
Two Essential Matrices and the Estimation of User Preference Profile



The second matrix (see Figure 1) crucial to recommendation systems is the user (row) by item (column) record of users' evaluations of items as they accumulate over time. Evaluations can take many forms including purchase records, ratings on various scales, "save for later," and so forth, all presumed to signal users' true preferences. More implicit feedback such as clicks to view or download, viewing times, and re-transmissions via social media can also be included. The user by item matrix can have a variety of entries from individual evaluations to more summary versions for each item.

Depending on the number of users' evaluations of items stored in this single matrix, a matrix of dense data would allow grouping users by the similarity of their evaluations of the same set of items (for more details, see the sections "Item Clustering" and "Hybrid Recommendation Systems"). Assuming common evaluations and stability over time, a well-designed recommendation system can predict the value of a new item to a target user based on evaluations assigned to that item by other users who are "similar" in taste to the target user, based on co-rating history of other items (Cacheda et al. 2011; Koren and Bell 2011). This *collaborative filtering* (CF) approach to recommendation does not utilize content features in its algorithms at all and solely operates on the user by item rating matrix, although various hybrid recommendation models take advantage of both content features and user ratings (Barragáns-Martínez et al. 2010; Burke 2007).

Here we introduce the notation system used from here forward. The set of users in the system is denoted as $U = \{u_1, u_2, \ldots, u_m\}$, the set of items as $I = \{i_1, i_2, \ldots, i_n\}$, the set of content features characterizing items as $C = \{c_1, c_2, \ldots, c_k\}$, and the rating for user $u_i \in U$ on item $i_u \in I$ as r_{iu}. Each item has a unique content profile in the system, *Content* $I_u = \overrightarrow{C_{i_u}} = (c_{u1}, c_{u2}, \ldots, c_{uk})$. Similarly, with users forming the rows and items forming the columns, the $U \times I$ *rating* matrix characterizes users on the basis of their ratings. In addition, for any item $i_u \in I$, there is a subset of users who have given their ratings on i_u, and we denote this subset of users as U_{i_u}. Likewise, for any user $u_i \in U$, I_{u_i} denotes the subset of items u_i has evaluated. Finally, the subset of items that have been co-rated by any two users u_i and u_j is denoted as $I_{u_i u_j} = I_{u_i} \cap I_{u_j}$. In a similar fashion, $U_{i_u i_v}$ denotes the subset of users who have given ratings to both item i_u and item i_v. The overlap in items rated by users is crucial to the applicability of any preference algorithm. When the overlap is small the data are said to be sparse and so strategies for dealing with sparse data are required. The next section explains how the *content-based* approach and *collaborative filtering* each operates on the $I \times C$ *content* matrix and the $U \times I$ *rating* matrix to generate recommendations.

Content-based recommendation

Building a content profile for items. Communication scholars are familiar with using content analysis to characterize news articles, social media posts, user comments, health-related PSAs and so forth along theoretically important dimensions to explain or predict individual-level or aggregated outcomes. Content-based recommendation starts with building a content profile for each item in the system using a common set of features and that represents any item I_u as a feature vector $\overline{C_{i_u}} = (c_{u1}, c_{u2}, \ldots, c_{uk})$. The selection of what features to include should be guided by domain knowledge (Lops, de Gemmis, and Semeraro 2011). For example, in the domain of tobacco control PSAs, structural features related to message sensation value (MSV) such as formal visual features (e.g., cuts, edits, special effects), formal audio features (e.g., sound effects, voiceover, etc.), and content format features (e.g., the use of narrative, a surprise ending, etc.) are known to affect viewers' sensory, affective, and cognitive responses (Harrington et al. 2003; Lang 2006; Morgan et al. 2003; Strasser et al. 2009) and therefore are prime candidates as dimensions defining the feature space. For text-based items, keywords, topics, and themes can be treated as attributes that form the feature space (Pazzani and Billsus 2007).

A variety of approaches to content-based recommendations have been deployed in the literature but they typically require dense data in the $U \times I$ matrix. When such data are sparse, item clustering allows the number of items needed to be reduced so that the evaluations per item-cluster by users are denser than the evaluations per item. Content-based recommendation makes predictions for user preferences for new items based solely on content similarity to previously rated items. However, the number of previously rated items must be dense for content filtering to work.

Item clustering. Because the density of data in the $U \times I$ matrix for some communication research will be sparse, we first treat the issue of item clustering of the transpose of the $I \times C$ matrix as a prerequisite for the application of some of the methods described in content-based and collaborative recommendation described below.

Item clustering operates on the transpose of the $I \times C$ matrix to produce groupings of items much smaller than the original number of items much the same as is done with questionnaire items to reduce their dimensionality. A wide variety of distance metrics among pairs of items and a variety of clustering techniques are available to yield a set of item groups that would redefine item space. Lesot, Rifqi, and Benhadda (2009) offered a detailed survey on different families of distance/similarity metrics such as Jaccard, Euclidean, cosine and Mahalanobis distances and distance-converted similarity scores. Readers interested in the state of art of existing approaches and algorithms to clustering—including but not limited to k-means/medoids, spectral clustering, and density-/model-/graph-based techniques—are invited to consult Aggarwal and Reddy's (2014) recent book on this subject.

The purpose of item clustering is to optimize between two criteria: increase the number of user evaluations per cluster (so a small number of clusters is desired) and maximize the similarity of items within a cluster (so a larger number of clusters is desired). Optimizing between these two criteria depends on the density of the user ratings before clustering and the number of items in the original set. Recent simulation data indicates that no a priori clustering criterion will provide a single, well-defined clustering of items (Kleinberg 2002). Instead judgment by the researchers and pertinence to the research questions are required.

In what follows on content-based and collaborative filtering, we describe users, content features, and individual items. However, in the presence of sparse data the reader can insert "item clusters" in the place of "items" to allow the applicability of the approach in the presence of sparse data.

The user preference profile: User content weights. After constructing the item profile, a user preference profile based on the $U \times I$ rating matrix is developed. The preference profile $\overrightarrow{P_{u_i}}$ for user u_i is defined on the same $I \times C$ feature space that describes items, and takes the form of a series of weights reflecting the user's preference for each element of item content, that is, $\overrightarrow{P_{u_i}} = (w_{u_i c_1}, w_{u_i c_2}, \ldots, w_{u_i c_k})$. Of specific note is the fact that each user has a unique preference profile, derived from his or her preference data and the content features associated with preferred and disliked items. Conceptually the analytic approach is analogous to a random-effects model, allowing random intercepts and random slopes to vary for each individual and in the process estimating "personalized" weights attached to each content feature (see Figure 1 for illustration). This approach is only feasible when individual-level data are dense. However, even when selection data are dense, the wide range of potential feature dimensions necessary to characterize the item set could potentially dwarf the selection data. As a result, heuristics to estimate the user preference vector $\overrightarrow{P_{u_i}}$ are common among content-based recommendation systems for their simplicity and accuracy given reasonably rich user-level data, although relying exclusively on heuristics typically reduces estimation precision and stability.

For example, for user u_i, given the set of rated items I_{u_i} and content profiles $\overrightarrow{C_{i_u}}(i_u \in I_{u_i})$ associated with items in this set, it makes sense to heuristically estimate parameters in $\overrightarrow{P_{u_i}}$ by the formula: $\overrightarrow{P_{u_i}} = \frac{\sum_{i_u \in I_{u_i}} (r_{iu} - \overline{r_i})\overrightarrow{C_{i_u}}}{\left|I_{u_i}\right|}$. Suppose user

u_i on average gives a rating of 3, three out of four antismoking PSAs she had rated all had the theme "smoking can cause lung cancer" (i.e., the feature variable) but the fourth PSA did not, and these four PSAs were rated as 2, 4, 5, and 5 on a 5-point Likert scale. Then in the user profile, the preference for this theme would be the average of (2−3)·1, (4−3)·1, (5−3)·1, and (5−3)·0, which is 1/2.

After obtaining the user profile, the recommendation problem is reduced to simply searching for the item in the database that matches up closest to the user profile. "Closeness" between the item profile $\overrightarrow{C_{i_u}}$ and the user profile $\overrightarrow{P_{u_i}}$ is usually formalized as a cosine similarity measure (Lops, de Gemmis, and Semeraro 2011) especially when the algorithm is applied to textual items:

$$sim\left(\overrightarrow{P_{u_i}},\overrightarrow{C_{i_u}}\right)=\cos\left(\overrightarrow{P_{u_i}},\overrightarrow{C_{i_u}}\right)=\frac{\overrightarrow{P_{u_i}}\cdot\overrightarrow{C_{i_u}}}{\|\overrightarrow{P_{u_i}}\|_2\times\|\overrightarrow{C_{i_u}}\|_2}=\frac{\sum_{k=1}^{K}w_{u_ic_k}c_{uk}}{\sqrt{\sum_{k=1}^{K}w_{u_ic_k}^2}\sqrt{\sum_{k=1}^{K}c_{uk}^2}}\ (eq.1),$$

where K is the total number of features characterizing both the item profile and the user profile. After being compared with the target user's preference profile, the item or items receiving the highest cosine score(s) will be recommended to user u_i.

More advanced learning algorithms. Other than simply relying on the heuristic of taking averages, more advanced algorithms from the machine learning literature can be leveraged to boost the accuracy of estimating the user profile $\overrightarrow{P_{u_i}}$. One particularly well-performing technique that originated from information retrieval is the Support Vector Machine (SVM). SVM treats the training set (items already rated by user) as labeled instances (e.g., "liked" or "disliked" items) in the high-dimensional feature space and attempts to find a separating hyperplane (i.e., formed by the dot product of $\overrightarrow{P_{u_i}}\cdot\overrightarrow{C_{i_u}}$) that maximizes the distances between the decision boundary and the closest training instances (i.e., the so-called support vectors). Since SVM estimates parameters in $\overrightarrow{P_{u_i}}$ by maximizing margin of separation and by considering combinations of features, it is robust to the problem of overfitting, thus resulting in robust performance in a variety of empirical experiments on benchmark datasets (Pazzani and Billsus 2007). A detailed description of SVM is beyond the scope of this article; interested readers are invited to consult Aggarwal and Zhai (2012, 163–222) for a general introduction to this topic and Hastie, Tibshirani, and Friedman (2009, 417–55) for a more mathematical treatment.

Another classification algorithm that assumes a probabilistic generative model is the Naïve Bayesian Classifier (NBC). This approach estimates the probability that item i_u belongs to a category G_p (e.g., "liked" or "disliked") given the set of content features $\overrightarrow{C_{i_u}}: Prob(G_p \mid c_{u1}\ \&\ \ldots\ \&\ c_{uk})$. The NBC bypasses the step of learning the user preference profile and proceeds directly to model the probabilistic relationship between item content features and categories of user evaluation. In this way, the NBC does not tailor item feature weights to individual raters but instead calculates $Prob(G_p \mid c_{u1} \,\&\, \ldots \,\&\, c_{uk})$ for each item and each category, and i_u is assigned the class label of the category associated with the highest probability (for details see McCallum and Nigam 1998; Pazzani and Billsus 2007). The algorithm recommends to the target user items falling into a predefined category (e.g., the "liked" class). Despite the popularity of the NBC in content-based recommendation systems, SVM has been found to outperform this method in several applications (Lops, Gemmis, and Semeraro 2011).

Content-based recommendation systems are useful for capturing the nuances in objective features of items themselves, and are most desirable when there are only a limited number of users in the system. In addition, since content features can be coded and processed offline, it is feasible to prepare item profiles before the system is even open to users. This feature renders content-based recommendation systems scalable to a large number of users as half of the computation can be done offline. On the other hand, one of the most serious drawbacks that content-based systems face is overspecialization due to the simple principle of recommending items similar to those preferred by the user in the past. The method is prone to producing "boring" recommendations. Recommendation systems based on CF, on the contrary, can mitigate overspecialization by taking advantage of "peers" with whom the target user never interacts.

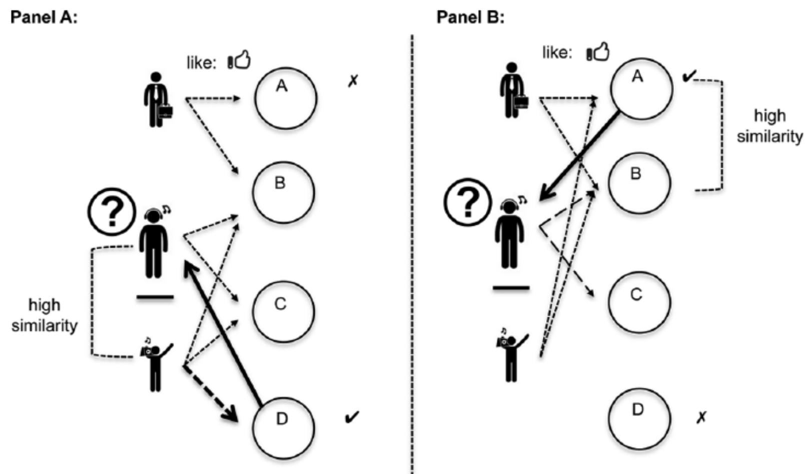
Recommendation Based on CF

Classic CF recommendation systems only utilize the $U \times I$ rating matrix, leaving information about item content attributes in the $I \times C$ content matrix unexploited. Two major subcategories of CF have been widely adopted: the *memory-based* k-nearest neighbor (k-NN) approach and the *model-based* matrix factorization approach (e.g., Singular Value Decomposition or SVD).

Memory-based neighborhood models. k-NN operates by identifying a prespecified number ($n = k$) of items most similar to the ones that have received ratings from the target user (i.e., item-based CF, hereafter referred to as IBCF; see Figure 2 for illustration); or alternatively, by identifying other users in the system that share similar tastes with the target user (i.e., user-based CF, hereafter referred to as UBCF). Though conceptually similar, IBCF is distinct from content-based recommendation as the degree of similarity between any pair of items i_u and i_v is not assessed from comparing content profiles $\overrightarrow{C_{i_u}}$ and $\overrightarrow{C_{i_v}}$, but rather from patterns of co-ratings provided by users in $U_{i_u i_v}$. UBCF appeared first (Herlocker et al. 1999) and later the analogous IBCF was developed (Sarwar et al. 2001), with its enhanced scalability and accuracy. Hybrid CF methods combine these two approaches (Wang, de Vries, and Reinders 2006). Amazon has successfully incorporated memory-based neighborhood models into its recommendation system.

The key component in k-NN is the similarity function that defines the pairwise item-item relationship in IBCF, and analogously the pairwise user-user relationship in UBCF. In IBCF, the most popular choice is cosine-based similarity with

FIGURE 2
The Idea Behind User- and Item-Based Collaborative Filtering



NOTE: Panel A illustrates the logic behind user-based collaborative filtering (UBCF) and Panel B illustrates how item-based collaborative filtering (IBCF) operates.

FIGURE 3
Calculation of Pairwise Item-Item (in IBCF) and User-User (in UBCF) Similarities

	Item i_1	...	Item i_u	Item i_v	...	Item i_n
User u_1			r_{1u}	r_{1v}		
...			...	...		
User u_i	r_{i1}	...	r_{iu}	r_{iv}	...	r_{in}
User u_j	r_{j1}	...	—	r_{jv}	...	r_{jn}
...			...	...		
User u_m			r_{mu}	r_{mv}		

$Sim_{u_i,u_j} =?$

$Sim_{i_u,i_v} =?$

NOTE: In this illustration, user u_j is excluded from the calculation of the similarity between item i_u and item i_v because the rating r_{ju} is missing. Similarly, item i_u is excluded from the calculation of the similarity between user u_i and user u_j.

modifications. Figure 3 illustrates how the similarity between i_u and i_v is determined.

First, ratings by users in $U_{i_u i_v}$ form the elements of vectors $\overrightarrow{i_u}$ and $\overrightarrow{i_v}$ that represent each of the two items, with their dimensions equaling the number of users in $U_{i_u i_v}$, i.e. $\dim\left(\overrightarrow{i_u}\right) = \dim\left(\overrightarrow{i_v}\right) = |U_{i_u i_v}|$. Second, to remove user biases in the tendency to rate items either higher or lower than others, the average rating for each user $u_i \in U_{i_u i_v}$ across all the items u_i has rated, $\overline{r_i}$, should be subtracted from r_{iu} and r_{iv}. Then the adjusted pairwise item-item cosine similarity can be calculated using Equation (2):

$$S_{uv} = sim_{i_u, i_v}\left(cosine\right) = \frac{\sum_{u_i \in U_{i_u i_v}} (r_{iu} - \overline{r_i})(r_{iv} - \overline{r_i})}{\sqrt{\sum_{u_i \in U_{i_u i_v}} \left(r_{iu} - \overline{r_i}\right)^2} \sqrt{\sum_{u_i \in U_{i_u i_v}} \left(r_{iv} - \overline{r_i}\right)^2}} \left(eq.2\right).$$

Third, once the item-item similarity matrix is set, the system continues to predict the target user u_i's rating $\widehat{r_{iu}}$ on an unseen item i_u. If user u_i has a rich history of rating items, a "neighborhood" of item i_u, $N(i_u; u_i)$, will be first selected by comparing i_u to each of the items in I_{u_i}. The k most similar items will be retained in the neighborhood, i.e. $\left|N\left(i_u; u_i\right)\right| = k, k < |I_{u_i}|$. The predicted rating $\widehat{r_{iu}}$ will then be a weighted sum of observed ratings in $N(i_u; u_i)$, with weights being similarity scores and the weighted sum normalized by adding up the absolute values of weights:

$$\widehat{r_{iu}} = \frac{\sum_{i_j \in N\left(i_u; u_i\right)} S_{uj} \times r_{ij}}{\sum_{i_j \in N\left(i_u; u_i\right)} | S_{uj} |} \left(eq.3\right).$$

In practice, the predictive accuracy will be greatly enhanced if we take into consideration the target user's rating tendencies and the target item's overall popularity across all the users in the system (Bell and Koren 2007; Koren and Bell 2011; Rajaraman and Ullman 2012). Equivalently, this is to build a baseline prediction model:

$$b_{iu} = \mu + b_{u_i} + b_{i_u} \, (eq.4),$$

where b_{iu} is the baseline prediction, μ is the global average rating across all the users and all the items in the system, b_{u_i} refers to the rating deviation of user u_i from the global average (i.e., $b_{u_i} = \overline{r_i} - \mu$), and b_{i_u} refers to rating deviation of item i_u from μ (i.e., $b_{i_u} = \bar{r_u} - \mu$). For example, if all the antismoking PSAs in the system on average receive a rating of 3, and user u_i is a generous rater who on average rates 4, and PSA i_u is an unpopular one that on average receives only a 1, then the baseline predictor for PSA i_u will be 2. After taking into consideration the baseline prediction model, Equation (3) can be transformed to the following:

$$\widehat{r_{iu}} = b_{iu} + \frac{\sum_{i_j \in N(i_u;u_i)} S_{uj} \times (r_{ij} - b_{ij})}{\sum_{i_j \in N(i_u;u_i)} |S_{uj}|} \, (eq.5).$$

Last, the system simply searches through all the predicted ratings and recommends the item(s) that have the highest score(s).

In UBCF, the prediction $\widehat{r_{iu}}$ can be obtained following similar steps: first, rather than selecting a "neighborhood" for item i_u, the system identifies a "neighborhood" for the target user u_i, $N(u_i\,;\,i_u)$, consisting of k "peer" users who 1) are most similar to user u_i based on rating histories of $i_v \in I_{u_i u_j} (v \neq u)$ and 2) have rated item i_u; second, the system computes a weighted sum of neighbors' ratings of item i_u with the weights being pairwise user-user similarities S_{ij}. In the UBCF literature, the Pearson's correlation is often used to calculate user-user similarity:

$$S_{ij} = sim_{u_i,u_j}(Pearson) = \frac{\sum_{i_v \in I_{u_i u_j}} (r_{iv} - \overline{r_i})(r_{jv} - \overline{r_j})}{\sqrt{\sum_{i_v \in I_{u_i u_j}} \left(r_{iv} - \overline{r_i}\right)^2} \sqrt{\sum_{i_v \in I_{u_i u_j}} \left(r_{jv} - \overline{r_j}\right)^2}} \, (eq.6).$$

Incorporating the same baseline model in IBCF, the predictive model can be specified as follows:

$$\widehat{r_{iu}} = b_{iu} + \frac{\sum_{u_j \in N(u_i;i_u)} S_{ij} \times (r_{ju} - b_{ju})}{\sum_{N(u_i;i_u)} |S_{ij}|} \, (eq.7).$$

In both IBCF and UBCF, the value of k unfortunately cannot be deduced from theoretical principles but is usually guided by empirical availability of rating data. The general rule of thumb would suggest, though, that a small number of high-confidence neighbors will provide a better ratio of signal to noise than a large number of neighbors whose similarity weights are less accurate or unreliable (Desrosiers and Karypis 2011).

Readers might find IBCF and UBCF to be two sides of the same coin. They are both motivated by the same conceptual framework and follow very similar steps in generating recommendations. In practice, which one is chosen depends largely on the ratio between the number of users and items. IBCF can save considerable computing resources when the number of users outnumbers items (Desrosiers and Karypis 2011). Researchers have also proposed integrative methods to fuse these two (Wang, de Vries, and Reinders 2006).

Model-based CF using matrix factorization methods. Unlike memory-based k-NN approaches, model-based CF assumes the observed rating matrix is generated from an underlying linear model: $r_{iu} = \overrightarrow{p_{u_i}} \cdot \overrightarrow{c_{i_u}}$, where vector $\overrightarrow{p_{u_i}}$ represents user u_i's content preferences and vector $\overrightarrow{c_{i_u}}$ is item i_u's content profile on the same dimensions. The reader will quickly recognize that conceptually, this dot product model is no different from the basic logic underlying various content-based recommendation algorithms. However, the distinction lies in the fact that in model-based CF, the exact dimensions along which item content could be characterized (hence user preferences as well) need to be inferred directly from the $U \times I$ rating matrix, without the aid of prior human or computer-assisted content coding. Matrix factorization techniques, such as SVD, are commonly adopted to uncover the underlying latent factor space (i.e., the content feature space) where both users and items can be simultaneously projected (Koren and Bell 2011). Conceptually, this is analogous to conducting a factor analysis on the $U \times I$ rating matrix to extract latent factor structures, with each factor corresponding to a content feature. The details of this procedure along with corrections to improve fit and deal with sparse data can be found in other sources (Bell and Koren 2007; Koren, Bell, and Volunsky 2009).

Although memory-based and model-based CF allow "innovative" selections different from the ones the target user has already seen, CF is vulnerable to the problem of sparse data. Memory-based CF will be unstable if there are little data on the target user's past ratings, or given past ratings, there is nevertheless little overlap among raters. Moreover, CF cannot make recommendations for newly added items until they are rated (Adomavicius and Tuzhilin 2005). In light of these drawbacks, hybrid recommendation methods that combine the merits of both content-based and CF methods hold the potential to be fruitful in overcoming the weaknesses of both.

Hybrid Recommendation Systems

Hybrid recommendation methods have been proposed to address the problem of sparse ratings in a CF system, especially when there are few co-ratings for pairs of items. In this case, content similarity from comparing the items' content profiles can be imported to improve the calculation of item-item similarity scores, hence enhancing the accuracy of IBCF (Pazzani 1999). Alternatively, predicted ratings based on content similarity can be used to impute missing values in the $U \times I$ rating matrix before performing CF (Melville, Mooney, and Nagarajan 2002; Eckhardt 2012). Some researchers combine content similarity and rating similarity to develop a unified item-item similarity matrix (Mobasher, Jin, and Zhou 2004). Still another technique supplements user-user similarity by incorporating extra information on users, such as demographic data, or other domain-specific individual attributes relevant to user similarity (e.g., smoking status for a system recommending tobacco-control messages) (Burke 2007). As with the constructions of many small groups of items based on content clustering described earlier, a similar approach with users can help with the sparse data issue by treating clusters of users as interchangeable.

Different recommendation methods can be fused to achieve better accuracy. One of the best-known examples in this regard is the Koren and Bell algorithm (Koren, Bell, and Volinsky 2009) that won the million-dollar Netflix prize. Despite the complexity in the algorithm that addresses temporal dynamics in user-item interactions, what sits at its core is the combination of factorization methods (e.g., SVD) operating at the regional level and neighborhood methods (e.g., k-NN) at the local level (see Bell, Koren, and Volinsky [2007] for details about the unified model).

Although there are no general guidelines regarding what type of hybrid methods would achieve better results, it is our belief that in the domain of public health message recommendations, some hybrid forms of content-based and CF methods should outperform using either of them alone. The good news is we can experimentally test the performances of various recommendation algorithms once the system is set up.

Applying Recommendation Systems

The success of any public health campaign depends in large measure on how effectively information is communicated to the targeted audience through the use of messages. Message tailoring selects the most appropriate set of message features based on an individual's unique needs. Meta-analytical reviews have confirmed that compared with nontailored interventions, both print and web-delivered tailored interventions are more effective in stimulating and sustaining desirable behavioral changes in domains of smoking cessation, healthy diet maintenance, physical activity, and regular mammography screening (Noar, Benac, and Harris 2007; Krebs, Prochaska, and Rossi 2010; Lustria et al. 2013). Despite the consensus among public health scholars and practitioners that tailoring works, the scholarly community has not reached an agreement on what core set of message features should be tailored to achieve optimal effectiveness in a given behavioral domain (e.g., smoking cessation) let alone across domains (Rimer and Kreuter 2006).

Conventional approaches to testing effective messages are often experimental and manipulate a few features at a time. This paradigm has produced important but very slow accumulation of knowledge and has shown only incremental progress in providing guidance to the practice of health message tailoring.

What recommendation systems can provide is message tailoring without the a priori specification of the content features. Instead, content features can be chosen before, during, or after the fact of item selection and the weights of individual content features for a given user assessed. The weights for features by user represent the extent to which a given set of features works for a given user. Users can be grouped on the basis of similarity in feature weights or items can be grouped in terms of those having similarities in feature weights across users ex post facto. In the recommendation approach, tailoring becomes a set of feature weights for a given user or for a new user similar in his or her choice architecture to a past user.

Recommendation systems have come out of commercial applications and so their roots are primarily predictive, rather than theoretical and explanatory. However, as we suggested above, recommendation systems can provide outcomes from prediction models that are unique and unanticipated parameter values. It is these values themselves that can become the objects of explanation. For example, to have a particular set of feature weights be predictive of a set of item ratings for a subgroup of users is an outcome from collaborative and content-based recommendation algorithms but is also an object of explanation. The prediction approach of recommendation systems asks why these features work for these users in this context. Why do other groups rely on other features for their positive and negative assessments? How much diversity (that is, uncertainty) exists in the feature weights overall? Given the diversity in feature weights, is tailoring to individual users worth the effort? To answer these types of questions is to offer explanations and construct predictive mechanisms—in short to model and build theory. So although recommendation systems are prediction-oriented at their heart, they can provide outputs in the form of parameters, which themselves become the objects of explanation.

The density of data required by recommendation approaches is a challenge for communication scientists, but it is also a challenge in commercial applications. The use of sparse data workarounds helps to deal with the sparseness issue. These include creating optimal subgroupings of items or subgroupings of users so that the density per subgroup is improved under the assumption of parallelism within subgroup. Much research is being done to propose and evaluate various solutions to the sparse data problem but this should not stop us from deploying relatively straightforward solutions that de facto are variants of missing data options through clustering of items and clustering of users.

The idea that the number of items available for rating is too small to be a serious set ignores the fact that the number of news articles generated daily by the press is gigantic; the number of antismoking PSAs in the CDC's archive is in the thousands; the number of health-based or political videos on YouTube is huge; as are the number of public comments, blog, and micro blog comments on any topic daily. So the items and reactions to them provide an extensive database for cultivation both as they exist online and as objects of more intensive study using crowd sourced samples in Amazon's Mechanical Turk or other more representative sets of raters.

Although many of the pertinent developments in methodology and statistics for recommendation systems are coming from engineering, computer science, and even physics, the core analyses are at heart modifications of techniques reasonably well known to social scientists. The identification of individual weights for content features in predicting item ratings is a standard outcome from certain classes of multilevel models. The clustering techniques for items and for users employ measures of distance and clustering that are variants of procedures used commonly in the social sciences. Although data manipulation and management are challenges within the recommendation environment, the analytic tools are translations and extensions of existing and more or less well-known techniques.

The effectiveness of any recommendation algorithm can be assessed by comparing the algorithm's predictions for a target person to that of a simpler standard. For example, if a content-based recommendation does less well in predicting a user's rating of unexposed messages than does the mean rating of all other users, then the algorithm is certainly ineffective. If the algorithm outperforms the simple mean, then it can be compared to other more conservative standards such as other algorithms, computationally simpler ones, ones based on smaller training sets and so on.

Recommendation systems offer a way to envision prediction of unseen items by users. They identify item features as one basis for predicting future choices but also incorporate individual variation in what features matter more and for whom. They also incorporate behavioral similarity between users as a basis for predicting what will be preferred in future exposures. Although these two components are in principle obvious elements in prediction of future choices, the methods and techniques being employed in the data environment of emerging media marks recommendation approaches as candidates both for predictive possibilities and theoretical advancement. It would be a mistake for theoretically inclined social and communication scientists to ignore recommendation systems and their algorithms or label them as counter to the goals of the scientific enterprise. After all, every science requires a fact base from which to build explanations and theories. Recommendation systems seek to build exactly that—prediction models both for the sake of pure prediction to solve a problem but also to provide truly predictive (versus "retro-dictive") models that then become the objects of explanatory accounts. Researchers employing recommendation modeling do not need to think of themselves as the unwanted progeny of commercial vendors just because they focus on predictive models. Such models are or at least can be the basis for explanatory theory.

References

Adomavicius, Gediminas, and Alexander Tuzhilin. 2005. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE Transactions on Knowledge and Data Engineering* 17 (6): 734–49.

Aggarwal, Charu, and Chandan Reddy. 2014. *Data clustering: Algorithms and applications*. Boca Raton, FL: CRC Press.

Aggarwal, Charu, and ChengXiang Zhai. 2012. A survey of text classification algorithms. In *Mining text data*, eds. Charu Aggarwal and ChengXiang Zhai, 163–222. New York, NY: Springer.

Almazro, Dhoha, Ghadeer Shahatah, Lamia Albdulkarim, Mona Kherees, Romy Martinez, William Nzoukou, and Dhoha Almazro 2010. A survey paper on recommender systems. *arXiv preprint arXiv:1006.5278*. Available from http://arxiv.org/abs/1006.5278.

Barragáns-Martínez, Ana Belén, Enrique Costa-Montenegro, Juan C. Burguillo, Marta Rey-López, Fernando A. Mikic-Fonte, and Ana Peleteiro. 2010. A hybrid content-based and item-based collaborative filtering approach to recommend TV programs enhanced with singular value decomposition. *Information Sciences* 180 (22): 4290–4311.

Bell, Robert M., and Yehuda Koren. 2007. Lessons from the Netflix prize challenge. *ACM SIGKDD Explorations Newsletter* 9 (2): 75–79.

Bell, Robert, Yehuda Koren, and Chris Volinsky. 2007. Modeling relationships at multiple scales to improve accuracy of large recommender systems. Paper presented at the 13th ACM SIGKDD international conference on Knowledge Discovery and Data Mining.

Bennett, Lance, and Shanto Iyengar. 2008. A new era of minimal effects? The changing foundations of political communication. *Journal of Communication* 58 (4): 707–31.

Burke, Robin. 2007. Hybrid web recommender systems. In *The adaptive web: Methods and strategies of web personalization*, eds. Peter Brusilovsky, Alfred Kobsa, and Wolfgang Nejdl, 377–408. New York, NY: Springer.

Cacheda, Fidel, Víctor Carneiro, Diego Fernández, and Vreixo Formoso. 2011. Comparison of collaborative filtering algorithms: Limitations of current techniques and proposals for scalable, high-performance recommender systems. *ACM Transactions on the Web (TWEB)* 5 (1): 1–33.

Desrosiers, Christian, and George Karypis. 2011. A comprehensive survey of neighborhood-based recommendation methods. In *Recommender systems handbook*, eds. Francesco Ricci, Lior Rokach, Bracha Shapira, and Paul Kantor, 107–44. New York, NY: Springer.

Eckhardt, Alan. 2012. Similarity of users'(content-based) preference models for collaborative filtering in few ratings scenario. *Expert Systems with Applications* 39 (14): 11511–16.

Harrington, Nancy Grant, Derek R. Lane, Lewis Donohew, Rick S. Zimmerman, Gretchen R. Norling, Jeong-Hyun An, Wai Hsien Cheah, Leola McClure, Tim Buckingham, and Elizabeth Garofalo. 2003. Persuasive strategies for effective anti-drug messages. *Communication Monographs* 70 (1): 16–38.

Hastie, Trevor, Robert Tibshirani, and Jerome Friedman. 2009. *The elements of statistical learning*. New York, NY: Springer.

Herlocker, Jonathan L., Joseph A. Konstan, Al Borchers, and John Riedl. 1999. An algorithmic framework for performing collaborative filtering. Paper presented at the 22nd annual international ACM SIGIR conference on Research and Development in Information Retrieval.

Jenkins, Henry. 2006. *Convergence culture: Where old and new media collide*. New York, NY: NYU Press.

Kim, Hyun Suk, Sungkyoung Lee, Joseph N. Cappella, Lisa Vera, and Sherry Emery. 2013. Content characteristics driving the diffusion of antismoking messages: Implications for cancer prevention in the emerging public communication environment. *JNCI Monographs* 2013 (47): 182–87.

Kleinberg, Jon. 2002. An impossibility theorem for clustering. Paper presented at Advances in Neural Information Processing Systems.

Koren, Yehuda, and Robert Bell. 2011. Advances in collaborative filtering. In *Recommender systems handbook*, eds. Francesco Ricci, Lior Rokach, Bracha Shapira, and Paul Kantor, 145–86. New York, NY: Springer.

Koren, Yehuda, Robert Bell, and Chris Volinsky. 2009. Matrix factorization techniques for recommender systems. *Computer* 42 (8): 30–37.

Krebs, Paul, James O. Prochaska, and Joseph S. Rossi. 2010. A meta-analysis of computer-tailored interventions for health behavior change. *Preventive Medicine* 51 (3): 214–21.

Lang, Annie. 2006. Using the limited capacity model of motivated mediated message processing to design effective cancer communication messages. *Journal of Communication* 56 (s1): S57–S80.

Lesot, Marie-Jeanne, Maria Rifqi, and H. Benhadda. 2009. Similarity measures for binary and numerical data: A survey. *International Journal of Knowledge Engineering and Soft Data Paradigms* 1 (1): 63–84.

Lops, Pasquale, Marco de Gemmis, and Giovanni Semeraro. 2011. Content-based recommender systems: State of the art and trends. In *Recommender systems handbook*, eds. Francesco Ricci, Lior Rokach, Bracha Shapira, and Paul Kantor, 73–105. New York, NY: Springer.

Lustria, Mia Liza A., Seth M. Noar, Juliann Cortese, Stephanie K. Van Stee, Robert L. Glueckauf, and Junga Lee. 2013. A meta-analysis of web-delivered tailored health behavior change interventions. *Journal of Health Communication* 18 (9): 1039–69.

Madrigal, Alexis. 2 January 2014. How Netflix reverse engineered Hollywood. *The Atlantic*.

McAuley, Julian, and Jure Leskovec. 2013. Hidden factors and hidden topics: Understanding rating dimensions with review text. Paper presented at the 7th ACM Conference on Recommender Systems.

McCallum, Andrew, and Kamal Nigam. 1998. A comparison of event models for naive Bayes text classification. Paper presented at AAAI-98 Workshop on Learning for Text Categorization.

Melville, Prem, Raymond J. Mooney, and Ramadass Nagarajan. 2002. Content-boosted collaborative filtering for improved recommendations. Paper presented at the Eighteenth National Conference on Artificial Intelligence at Edmonton, Canada.

Mobasher, Bamshad, Xin Jin, and Yanzan Zhou. 2004. Semantically enhanced collaborative filtering on the web. In *Web mining: From web to semantic web*, eds. Bettina Berendt, Andreas Hotho, Dunja Mladenič, Maarten van Someren, Myra Spiliopoulou, and Gerd Stumme, 57–76. Lecture Notes in Computer Science. Berlin: Springer Heidelberg.

Morgan, Susan E., Philip Palmgreen, Michael T. Stephenson, Rick H. Hoyle, and Elizabeth P. Lorch. 2003. Associations between message features and subjective evaluations of the sensation value of antidrug public service announcements. *Journal of Communication* 53 (3): 512–26.

Noar, Seth M., Christina N. Benac, and Melissa S. Harris. 2007. Does tailoring matter? Meta-analytic review of tailored print health behavior change interventions. *Psychological Bulletin* 133 (4): 673–93.

Pazzani, Michael J. 1999. A framework for collaborative, content-based and demographic filtering. *Artificial Intelligence Review* 13 (5–6): 393–408.

Pazzani, Michael J., and Daniel Billsus. 2007. Content-based recommendation systems. In *The adaptive web: Methods and strategies of web personalization*, eds. Peter Brusilovsky, Alfred Kobsa, and Wolfgang Nejdl, 325–41. New York, NY: Springer.

Rajaraman, Anand, and Jeffrey David Ullman. 2012. *Mining of massive datasets*. Cambridge: Cambridge University Press.

Rimer, Barbara K., and Matthew W. Kreuter. 2006. Advancing tailored health communication: A persuasion and message effects perspective. *Journal of Communication* 56 (s1): S184–S201.

Sarwar, Badrul, George Karypis, Joseph Konstan, and John Riedl. 2001. Item-based collaborative filtering recommendation algorithms. Paper presented at the 10th International Conference on World Wide Web.

Schafer, J. Ben, Joseph A. Konstan, and John Riedl. 2001. E-commerce recommendation applications. *Applications of Data Mining to Electronic Commerce* 5 (1/2): 115–53.

Strasser, Andrew A., Joseph N. Cappella, Christopher Jepson, Martin Fishbein, Kathy Z. Tang, Eugene Han, and Caryn Lerman. 2009. Experimental evaluation of antitobacco PSAs: Effects of message content and format on physiological and behavioral outcomes. *Nicotine & Tobacco Research* 11 (3): 293–302.

Wang, Jun, Arjen P. De Vries, and Marcel J. T. Reinders. 2006. Unifying user-based and item-based collaborative filtering approaches by similarity fusion. Paper presented at the 29th annual international ACM SIGIR conference on Research and Development in Information Retrieval.