Combining Content-Based and Collaborative Filters in an Online Newspaper

Mark Claypool, Anuja Gokhale, Tim Miranda,
Pavel Murnikov, Dmitry Netes and Matthew Sartin
Computer Science Department, Worcester Polytechnic Institute
Worcester, Massachusetts, USA *

Abstract

The explosive growth of mailing lists, Web sites and Usenet news demands effective filtering solutions. Collaborative filtering combines the informed opinions of humans to make personalized, accurate predictions. Content-based filtering uses the speed of computers to make complete, fast predictions. In this work, we present a new filtering approach that combines the coverage and speed of content-filters with the depth of collaborative filtering. We apply our research approach to an online newspaper, an as yet untapped opportunity for filters useful to the wide-spread news reading populace. We present the design of our filtering system and describe the results from preliminary experiments that suggest merits to our approach.

1 Introduction

That we are in the age of information is evident quite clearly in newspapers as an information source. Nearly everywhere in North America you can have 1/2 dozen newspapers delivered to your doorstep, each with hundreds of new articles each day. Nearly everywhere in the world via the World Wide Web, you can access more than 2500 daily newspapers through their online Web sites [27], providing tens of thousands of news articles of potential interest. We need information filters to help us prioritize news articles so that we may spend more of our time reading articles of interest.

Newspaper filters present the additional opportunity for personalization, which has been shown to have a strong appeal to newspaper readers [5]. Practical considerations have prevented hard-copy newspapers from obtaining any degree of personal customization, but online newspapers are not subject to the same constraints as printed matter. Online newspaper presentation can be personalized in terms of contents, layout, media (text only, text with pictures, text with video, etc.), advertisements and more. While there have been attempts at customization of newspapers and filtering of newspapers [16, 25], these attempts have not completely countered the weaknesses of human and computer filters.

Both humans and computers need help in filtering information. Much of our everyday filtering is done by human critics. We read book reviews before we decide what books to read. We listen to movie critics before we decide what movies to watch. We count the number of rating stars before deciding at what restaurant to eat. However, while humans are generally smart at deciding what information is good and why, we are slow compared to the amount of information out there that requires filtering.

The power and connectivity of computers has allowed computers to be applied to the problem of filtering information. Mechanical filters such as keyword searching and artificial intelligence filters such as natural language processing have attempted to apply the power of computers to the problem of prioritizing information. However, while computers are fast at processing information, they are generally stupid relative to humans when it comes to making meaningful decisions about the information content.

Collaborative filtering applies the speed of computers with the intelligence of humans. Collaborative filtering is the technique of using peer opinions to predict the interest of others. Users indicate their opinions in the form of ratings on various pieces of information, and the collaborative filter correlates the ratings with those of other users to determine how to make future predictions for the rater. In addition, the collaborative filter shares the ratings with other users so they can use them in making their own predictions. Early on, Tapestry [13] and GroupLens [21] applied collaborative filtering to email and Usenet news, respectively. Ringo [22] and Video Recommender [15] used collaborative filtering to recommend music and movies, respectively. Today, collaborative filtering systems recommend everything from books to restaurants, and there is even a collaborative filtering system for jokes [14].

However, collaborative filtering alone can prove ineffective for several reasons:

- *Early rater problem.* Pure collaborative filtering can not provide a prediction for an item when it first appears since there are no users ratings on which to base the predictions. Moreover, early predictions for the item will often be inaccurate because there are few ratings on which to base the predictions [12]. Similarly, even an established system will provide poor predictions for each and every new user that enters the system. As extreme case of the early rater problem, when a collaborative filtering system first begins every user

*This paper appears in *ACM SIGIR Workshop on Recommender Systems – Implementation and Evaluation*, August 19, 1999 in Berkeley CA, USA.

suffers from the early rater problem for every item.

- *Sparsity problem.* In many information domains, the number of items far exceeds what any individual can hope to absorb, thus matrices containing the ratings of all items for all users are very sparse. Relatively dense information filtering domains will often still be 98-99% sparse, making it hard to find documents that have been rated by enough people on which to base collaborative filtering predictions.

- *Gray sheep.* In a small or even medium community of users, there are individuals who would not benefit from pure collaborative filtering systems because their opinions do not consistently agree or disagree with any group of people. These individuals will rarely, if ever, receive accurate collaborative filtering predictions, even after the initial start up phase for the user and system.

Enter pure content-based filtering. A pure content-based filter recommends items based solely on a profile built up by analyzing the content of items that a user has rated. Pure content-based filters for newspapers include [7], [8] and [19]. A content-based filter analyzes items rated by an individual user and uses the content of the items as well as the provided ratings to build a profile to compare to other non-rated items to recommend additional items of interest. Content-based filters are less affected by the above problems of pure collaborative filters because they use techniques that apply across all documents. For example, a filter that predicts high interest for articles with the word "Clinton" in them can give the prediction before anyone has read the article.

Despite these strengths, content-based filters alone can prove ineffective. Unlike humans, content-based techniques have difficulty in distinguishing between high-quality and low-quality information that is on the same topic. And as the number of items grows, the number of items in the same content-based category increases, further decreasing the effectiveness of content-based approaches.

Experiments have shown collaborative filtering systems can be enhanced by adding content-based filters [1, 2]. By using a combination of content-based and collaborative filters we can realize the benefits of content-based filters which include early predictions that cover all items and users, while gaining the benefits of accurate collaborative filtering predictions as the number of users and ratings increases.

There have been several filtering approaches which have utilized combined content-based and collaborative filters:

GroupLens implements a hybrid collaborative filtering system for Usenet news that supports content-based filters as users [18]. These *filterbots* evaluate new articles as soon as they are published and enter ratings for those documents. The collaborative filtering system treats a filterbot as another ordinary user that enters many ratings. The filterbot author writes a filterbot just like a content-based agent that responds whenever a new article arrives and returns a rating to the system. Filterbots help with the problem of sparsity since they are able to rate many articles quickly, but since the GroupLens predictions still use pure collaborative filtering, new users, and hence new filterbots, still suffer from the early rater start-up problem. Ironically, at start-up, even a filterbot and its author still have no correlation even though the author likely has a good indication of correlation with the bot. In addition, really excellent filterbots may not be weighed heavily enough if there are many news readers with medium to high correlations.

Fab implements a hybrid content-based collaborative system for recommending Web pages [2]. In Fab, user profiles based on the pages a user liked are maintained by using content-based techniques. The profiles are directly compared to determine similarity between users in order to make collaborative filtering predictions. In order to be effective, the Fab approach mandates that the content-based techniques to build the user profile be extremely accurate. Inaccurate profiles result in inaccurate correlations with other users, greatly diminishing the strength of the collaborative filtering predictions.

Basu, Hirsh and Cohen applied an inductive learning approach using ratings and artifact information to predict user movie preferences [3]. They fed movie data and training data into Ripper, a machine learning tool, in an attempt to produce better recommendations than either collaborative or content-based recommendations alone. In order to achieve results that were clearly better than content-based and collaborative predictions alone they required a human-engineering effort to produce additional hybrid features.

ProfBuilder recommends web pages using both content-based and collaborative filters [24]. Users are provided a single interface to two lists of recommended web sites, one list generated by a collaborative filter the other list by a content-based filter. However, the two lists are not combined into a single list with a combined prediction, nor are the relative strengths of each prediction given so as to allow the user themselves to globally choose the best sites from both lists.

1.1 Our Approach

We provide a unique approach to combining content-based and collaborative filtering by basing a prediction on a weighted average of the content-based prediction and the collaborative prediction. Our approach fully realizes the strengths of content-based filters, mitigating the effects of the sparsity and the early rater problems. Moreover, the weights of the content-based and collaborative predictions are determined on a per-user basis, allowing the system to determine the optimum mix of content-based and collaborative recommendation for each user, helping to solve the gray sheep problem.

In addition, our approach allows for the content-based and collaborative weights to be computed on a per-item basis. The weights are adjusted based on the "strength" of each prediction. As the number of users and ratings for the item increase, the collaborative filter is (usually) weighted more heavily, increasing the overall accuracy of the prediction. This allows us to take advantage of the in-depth human understanding of the material as ratings enter the system.

Our approach is not a hybrid approach, meaning that the basis for the content-based and collaborative predictions are kept separate. This allows us to benefit from individual advances made to either collaborative or content-based filters since there is no inter-dependency between the two content components. For example, individual improvements to pure collaborative filtering algorithms [6] [12] can be fully realized. As advances in each component are incorporated into a working system, our weighted average approach will adapt to provide the best combination of the individual predictions.

Our approach is extensible to additional filtering methods by allowing each method to be added as a separate weight in the weighted average. For example, predictions based on demographics [20] could be readily included into our prediction scheme. In addition, our approach is hierarchical in that within each individual component, subcomponents that make up the prediction for that component can be given different percentages of the weight. For example, our content-based filter uses a weighted average to combine its keyword and section components.

We have built a collaborative filtering system for an online newspaper, the Worcester Telegram and Gazette Online (Tango) [26], in order to test our approach. Preliminary user studies suggest merits to our approach, with more extensive user studies currently ongoing.

The rest of this paper proceeds as follows: Section 2 describes our content-based and collaborative filters in detail; Section 3 introduces the system we use to test our approach; Section 4 provides analysis from our preliminary experiments; Section 5 summarizes our conclusions; and Section 6 presents areas of future work.

2 Research Approach

In this section, we describe the pure collaborative filtering prediction, the pure content-based prediction and the means by which they are combined into a single prediction.

2.1 Collaborative Filter

We build upon the work of the pure collaborative filtering algorithms published that compute similarities between users using a *Pearson correlation coefficient* [4, 21, 9]. Predictions for an item are then computed as the weighted average of the ratings for the items from those users which are similar, where the weight is the computed coefficient. The general formula for a prediction for an item for user u is:

$$prediction = \overline{u} + \frac{\sum_{i=1}^{n}(corr_i) \times (rating_i - \overline{i})}{\sum_{i=1}^{n}(corr_i)}$$

where $\overline{u}$ is the mean rating for the user in question, $corr_i$ is the Pearson's correlation coefficient of user i with the user for whom the prediction is being computed, $rating_i$ represents the rating submitted by user i for the article for which the prediction is being computed, $\overline{i}$ is the average rating (the average of the user ratings for the articles in common) for user i, and n is the total number of users in the system that have some correlation with the user and have rated the item.

2.2 Content-Based Filter

Our content-based filtering algorithms match article keywords to keywords in the user profile. We first briefly describe the format of the user's profile required for the calculation of the content-based prediction, then give details on keyword generation and lastly describe our matching function.

Each user profile is divided into *sections* corresponding to the Tango newspaper, such as "Business" or "Sports." For other information sources these sections could be mapped to Usenet news groups or Web server volumes. Users can explicitly indicate preference for articles in these sections by marking the checkboxes for each particular section. In addition, users can specify *explicit keywords* for each section. For example, a user may select the section "Sports" and enter the keyword "Broncos" to indicate interest in sports articles in general and high interest in Denver Bronco sports articles. Each profile section also contains a list of *implicit keywords* that is populated by appending the keywords of the articles that the user has given a high rating (currently, the top quartile of their range of ratings) to the current list of implicit keywords.

The explicit interest indicators (newspaper section and explicit keywords) enable the content-based filter to use direct learning in predicting article interest. Direct learning provides predictable behavior for the user and is often precise in defining user interest, but requires a lot of manual effort in order to achieve accurate predictions.

Users are not required to enter either newspaper section or explicit keywords in order to utilize the content-based filter. The implicit interest indicator (implicit keywords) enables the content-based filter to use indirect learning in predicting interest in the absence of (or despite) the explicit interest indicators. Indirect learning does not require extensive user effort, but may miss some potentially useful content information and capture some non-useful content information.

To generate keywords, for each article we remove stop words [10] and then perform word stemming [11]. Keywords are selected based on a frequency count of words, assuming that the occurrence of words in an article furnishes a useful measurement of word significance [17].

To compute the degree of match between the article keywords and the keywords in the user's profile, we use the *Overlap Coefficient* given in the following formula:

$$M = \frac{2|D \cap Q|}{min(|D|, |Q|)}$$

where D is the set of keywords extracted from the article and Q is the set of keywords in the user's profile. The coefficient, M, is not influenced by the sizes of D and Q, which is desirable as the number of article keywords could be much larger than the keywords in the user's explicit keyword list or much smaller than the keywords in a user's implicit keyword list.

Predictions from the three interest indicators, explicit keywords, implicit keywords and newspaper section, are combined via matching functions with the article keywords and section. Until we determine appropriate weights for each content-match through extensive user studies, we give each an equal weight of 1/3 and combine them to produce a single content-based prediction. Also, a weight of only 1/3 for an individual component can allow a high indication of interest by the other two components to override it. For example, if a user has checked only "Sports" as a section of interest, then when there are no strong implicit or explicit keyword matches in other sections, sports articles will be given preference. However, a business article with a strong implicit and explicit keyword match would receive preference over a general sports article.

2.3 Combination Filter

Vogt et al showed that a simple linear combination of scores returned by different Information Retrieval agents can improve the performance of those individual systems on new documents, achieving a better performance than any individual agent [23]. We build upon their work by combining our collaborative filtering prediction with our content-based prediction using a weighted average.

The trick is to come up with the weights that result in the most accurate prediction. The collaborative filtering

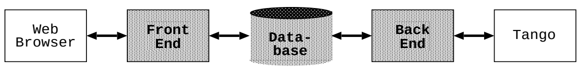

Figure 1: **The P-Tango System Architecture.** P-Tango consists of a front end, database and back end. The user accesses the front end through a web browser. The back end downloads articles from the Worcester Telegram and Gazette Online (Tango) and makes predictions.

predictions are more inaccurate in cases where the the number, agreement or history between users is low [12]. Similarly, there are cases where the content-based predictions are more inaccurate, such as when a users has not specified explicit keywords, or has rated too few articles highly to generate implicit keywords.

Both the collaborative filtering and content-based scores are important but the extent of their importance towards the aggregate score (or prediction) is very user-specific. We therefore implement per-user, per-article weights for the content-based and collaborative predictions. These weights can also change over time to reflect the change in user tastes.

We start by giving an equal weight to both the collaborative filtering and the content-based scores for all the users. As users make ratings, we compute the absolute error from the content-based predictions and the collaborative predictions and adjust the weights so as to minimize past error. The weights adjust quickly at first, but slow as the number of ratings and predictions increase.

3 System

We have designed and developed a filtering system for the Worcester Telegram and Gazette Online (Tango [26]) to use as a test-bed for research approaches to collaborative filtering. Our system, Personalized Tango or *P-Tango*, provides a personalized, customizable, Web-based interface to Tango. The P-Tango system has three components: the *front end*, the *database* and the *back end*, as depicted in Figure 1.

3.1 Front End

The front end allows users to login, modify their user profiles, and browse, read and rate newspaper articles. The front end consists of the web pages accessible through a Web browser, run by the user on their workstation or PC, and Active Server Pages, run on a Windows NT workstation at WPI. P-Tango supports any standard Web browser that supports frames, such as recent versions of Netscape and Internet Explorer, allowing users to access P-Tango without need for additional custom client software.

When first using P-Tango, users must register a pseudonym and select a password. Login is then accomplished with a simple name and password, with the option of using automatic login via cookies for subsequent visits. Upon their first login, users are requested to set up their personal profile.

The user profile page lists the available newspaper sections and provides checkboxes for indication of interest in each section as depicted in Figure 2. All selected sections are added to a user drop down listbox that appears as a navigation aid at the left of the P-Tango window. Each user can also indicate the desired section to appear as their "front page" when re-visiting P-Tango. In addition, users can specify multi-word explicit keywords within each section. The current newspaper sections supported are: Archives, Breaking News, Business, City Edition, Community Pages, Editorials, Food, Health, Movies, Nation/World, Obituaries, People, Regional, Sports, Time Out, Travel and Your Top Ten List. Additional sections include the regular Telegram and Gazette columnists and North, South, East and West regional sections for localities around Worcester.

P-Tango divides the browser window into frames as depicted in Figure 3. The top frame contains the title, allows modification of user settings and access to the user profile. The left frame contains newspaper navigation aids, including a list of all possible newspaper sections, a personalized drop-down listbox of those sections which were indicated of interest in the user profile and a text-box for searches. The middle frame initially shows a list of the articles in the current section. This article index lists each article with its title and byline (location, first sentence(s) of the article and sometimes the newspaper reporter). Upon selecting an article (as in Figure 3), the middle frame text is replaced by the full article text and a frame for entering ratings appears on the right-hand side of the browser.

Upon reading an article, users enter ratings via a color-coded ratings bar on the right hand side of the page. The bar is not numerically labeled, but rather the top of the bar is red and labeled "More", while the bottom of the bar is blue and labeled "Less." The entire bar is then wrapped in the phrase "I would like to see *(bar)* of this type of article." The user selection along the bar is mapped to an integer between 1 (less) to 10 (more).

Predictions of interest are given in one of two ways, selectable by the user. The default way is to have the front end show the title and byline of articles of high interest (upper quartile of a user's prediction range) with a blue background, while the title and byline of articles of low interest (lower three quartiles of a user's prediction range) are shown with a white background. Alternatively, the front end can be configured to show predicted interest with 1 to 5 "stars" appearing to the left of the byline of each unread article.

When displaying article indices to a user, the front end restructures access to the newspaper articles in two ways. First, it orders the articles in decreasing order of predicted interest. This allows users to easily access those articles that are most likely to be of interest. Second, it provides a "top ten" list containing the 10 unread articles with the highest predicted interest across all sections, which are accessible via a separate article index page.

3.2 Database

The database stores the article text, user profiles, ratings and predictions. P-Tango runs on an Oracle 8 database running on Windows NT 4.0. The back end and the front end connect to the database via ODBC.

3.3 Back End

The back end is the heart of the P-Tango system. It imports articles from the Tango site, computes correlation scores between users, generates article keywords, and calculates content-based, collaborative and combined predictions.

Internally, the back end consists of a master control thread and a slave thread for each of the modules. The modules in the back end include the following:

- *DBImport* imports articles from the Tango web site into the database. DBImport is run once per day at

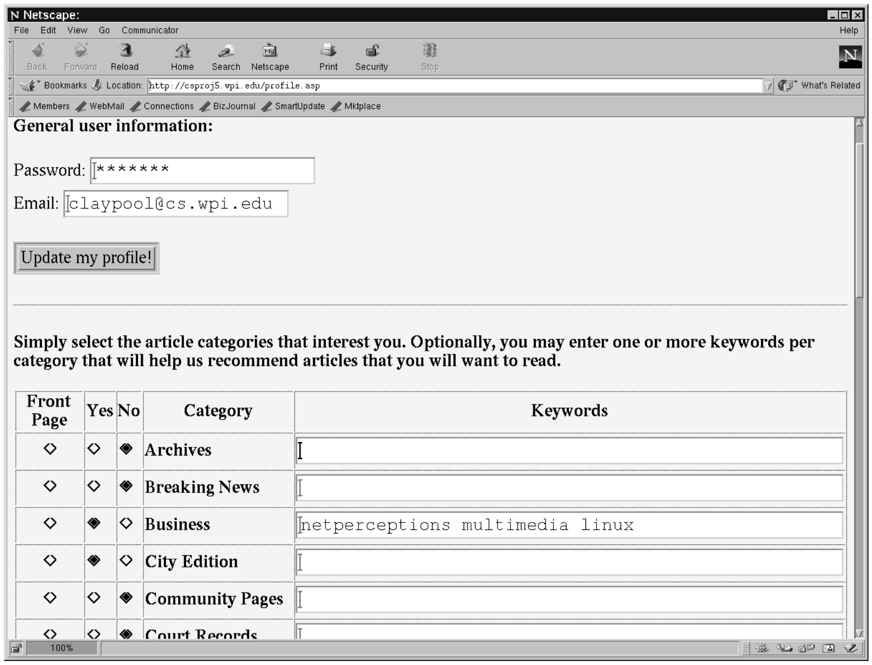

Figure 2: **The P-Tango Online Newspaper User Profile.** Users can change passwords and email, choose newspaper sections and keywords of interest and determine which section is the "front page."

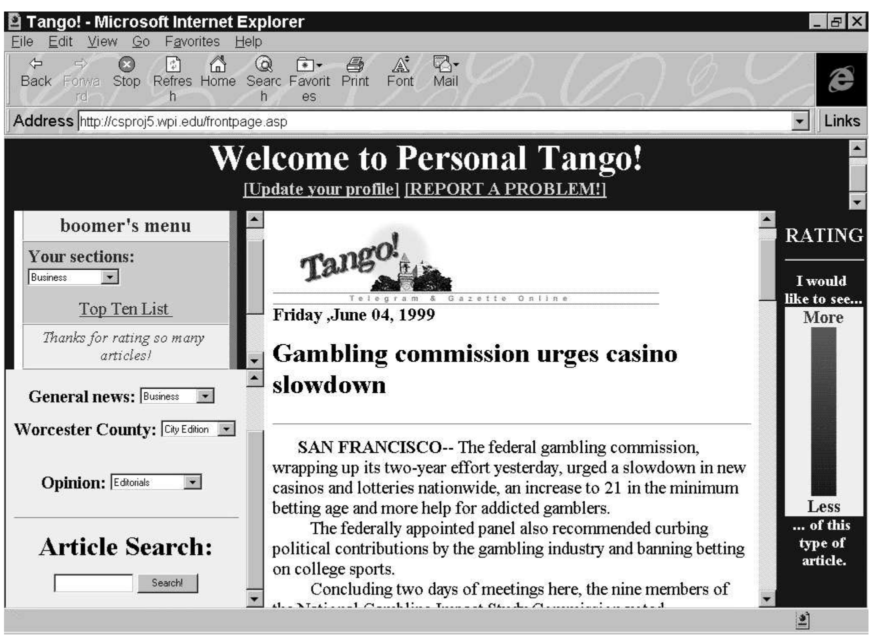

Figure 3: **The P-Tango Online Newspaper.** The left frame contains navigation aids, the middle frame the newspaper text, the right frame a bar for users to enter ratings and the top frame a title and access to profile information.

5am, shortly after the Tango Web site is updated with the day's articles.

- *KeyGen* generates keywords for new articles. KeyGen is invoked upon completion of DBImport, when it then processes all newly imported articles.

- *ConPredGen* generates content-based predictions based on the keyword matches. ConPredGen is run once per day, after the completion of KeyGen.

- *CorrGen* generates correlations between pairs of users. CorrGen is run once per day, at 2am, so as to minimize load on the database during regular online news reading hours.

- *CollPredGen* generates collaborative filtering predictions. CollPredGen is invoked after the completion of CorrGen and at 10 minute intervals throughout the day. CollPredGen polls the database for articles that have new ratings, then re-computes any predictions that are may be affected.

- *ComPredGen* generates the combined content-based and collaborative predictions. ComPredGen is invoked for each article that has a new content-based or collaborative prediction.

The back end has a main control dialog box for system administration, allowing stopping and starting P-Tango, starting and stopping individual components, and configuration options such as the time each component waits between polling the database and execution times for components that run daily. The control panel also displays vital information about each component, such as its current status and recent performance information.

4 Experiments

We have begun preliminary experiments to evaluate the effectiveness of our research approach in the P-Tango system. Our experiments thus far have been for a short amount of time and with a small number of users[1]. We had 18 users, mostly computer science students, both graduate and undergraduate, who used P-Tango on a semi-regular basis for about 3 weeks. During this time, P-Tango imported about 50 articles per day. Although some users were able to read nearly all the articles on some days, overall, there were only about 0.5% of the possible ratings entered.

In order to evaluate the filtering performance of P-Tango, we computed the *mean absolute error* between the numerical predictions provided by P-Tango and the numerical ratings entered by the user for the same articles. The mean absolute error is a measure of the deviation of predictions from their user-specified ratings. The lower the error, the more accurately the P-Tango system predicts user ratings. We call the mean absolute error the *inaccuracy* of the predictions.

Figure 4 depicts the inaccuracy of the P-Tango predictions over the three week trial. There are three data sets plotted: the inaccuracy of the content-based predictions, the inaccuracy of the collaborative predictions and the inaccuracy of the combination predictions. For all points, inaccuracy is plotted as the average over all users over all items rated. For the collaborative predictions, inaccuracy scores are only given for those in which a collaborative prediction was possible (*ie*- no first rater problem).

For the first week, the content-based predictions are consistently more accurate than the collaborative predictions. During this time, we hypothesize the base of user correlations and ratings is still being established, causing collaborative filtering predictions to be inaccurate. During the second week, it is not clear which is more inaccurate, the content-based predictions or the collaborative predictions. By the third week, the collaborative predictions seem to be somewhat more accurate than the content-based predictions. The collaborative predictions show a slight downward trend in inaccuracy over the three week testing period. Throughout the three weeks, the combined predictions are somewhat more accurate than either the content-based predictions or the collaborative predictions alone.

Not reflected in this data are the articles where there would be no predictions available if pure collaborative filtering was used. In these cases, there is still great value to the user in using a combined or content-based prediction, but it is difficult to measure this benefit numerically.

Lastly, we point out that there is a lot of variation among the predictions from day to day. Thus, we believe that the true value of the P-Tango system will only be revealed once our current set of ongoing user experiments are completed, as we indicate in Section 6.

5 Conclusion

The explosive growth of online information demands new techniques for prioritizing and presenting items of potential interest to users. Collaborative filtering combines the strengths of human intelligence in understanding information content with the speed of computers in information processing. Unfortunately, collaborative filtering techniques alone can be ineffective when users have not rated an item, for new users of the filtering system, or for users who do not generally benefit from the opinions of others. Content-based filtering techniques can be combined with collaborative filtering techniques to mitigate all these short-comings.

In this work, we present a filtering approach that combines pure content-based predictions with pure collaborative filtering predictions. Our approach fully realizes the benefits of the content-based approach to filtering while adapting to the ever strengthening collaborative filtering predictions.

The growing domain of online newspapers presents a rich area which can benefit immensely from personalized filtering approaches. We have designed and implemented a collaborative filtering test-bed, called *P-Tango* for our filtering research that provides personalized filtering of an online newspaper. After incorporating our approach into P-Tango, our preliminary results suggest merit to our approach.

In summary, the contributions of this work include:

- A unique approach to integrating content-based and collaborative filters.

- The first application of collaborative filtering to an online newspaper.

- A content-based filter for online newspapers that uses both direct and indirect learning.

- Preliminary experiments evaluating our approach.

[1] The P-Tango system has been released to a limited group for testing purposes. However, it is our intent to make it publicly available for general use at the end of these tests.

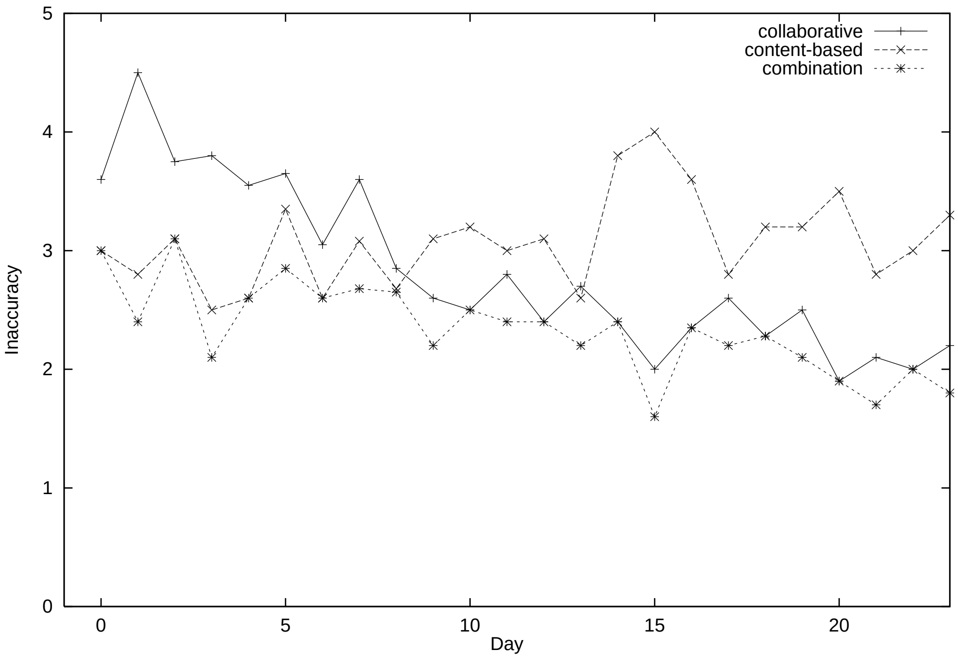

Figure 4: **Inaccuracy versus Time.** The horizontal axis is the days since the beginning of the study. The vertical axis is the average inaccuracy. The three lines depict the inaccuracy of the content-based filter, the collaborative filter and the combined filter.

6 Future Work

The areas of both content-based and collaborative filtering, coupled with a test-bed supporting real users, is rich with future work possibilities.

Our ongoing effort continues by extending our experiments to a more extensive user study of P-Tango. In these experiment, we will use 'real' newspaper readers who read Tango on a regular basis, dividing them into a control group and a test group, and gather data over the course of many months. Additional measures of performance to those presented here will be gathered, including mean time reading online newspaper, mean time per article, overall opinions on newspaper layout and utility, and more.

In [20], Pazzani introduces a means of filtering using demographic information. Our approach can be extended to include a pure demographic filter, which can be combined with the content-based and collaborative filters using our weighted average approach. Future work includes incorporating such a filter into P-Tango and evaluating its effectiveness.

The accuracy of the combined predictions in our approach largely depends upon a measure of the "strength" of the collaborative filtering prediction. The prediction strength is used to determine the relative weights of the content-based and collaborative predictions. To our knowledge, a computation of the strength of a collaborative filtering prediction has not been adequately treated in the literature.

Online newspapers hold great promise for restructuring newspaper layout according to individual preferences. Thus far, our top-10 list and article re-ordering have only scratched the surface of the problem. Rich possibilities include personalizing the number of articles per page, the inclusion and size of pictures, even the shape and depth of the newspaper "tree." With accurate predictions on a user's level of interest in unread articles, P-Tango will seek to deliver a personalized front-page, containing only the articles of highest interest, individually created everyday, for each user that accesses the P-Tango site.

Research in pure collaborative filtering and pure content-based filters continues to better the methods of identifying items of user interest. Future work includes incorporating ongoing breakthroughs in these filtering technologies into our P-Tango filtering system.

References

[1] Joshua Alspector, Aleksander Kolcz, and Nachimuthu Karunanithi. Comparing Feature-Based and Clique-Based User Models for Movie Selection. In *Proceedings of the Third ACM Conference on Digital Libraries*, pages 11 – 18, 1998.

[2] Marko Balabanovic and Yoav Shoham. Content-based, collaborative recommendation. *Communications of the ACM*, 40(3), March 1997.

[3] Chumki Basu, Haym Hirsh, and William Cohen. Recommendation as Classification: Using Social and Content-Based Information in Recommendation. In *Proceedings of the American Association for Artificial Intelligence*, pages 714 – 720, 1998.

[4] D. Billsus and M. Pazzani. Learning Collaborative Information Filters. In *Machine Learning: Proceedings of the 15th International Conference*, 1998.

[5] Leo Bogart. *Press and Public: Who Reads What, When, Where and Why in American Newspapers*. Lawrence Erlbaum Associates, 1989.

[6] John S. Breese, David Heckerman, and Carl Kadie. Empirical Analysis of Predictive Algorithms for Collaborative Filtering. Technical Report MSR-TR-98-12, Microsoft Research, October 1998.

[7] Vicente Luque Centeno, Carmen Fernandez Panadero, and Carlos Delgado Kloos. Personalizing Your Electronic Newspaper. In *Proceedings of the 4th Euromedia Conference (WEBTEC)*, April 26-28 1999.

[8] P. R. Chesnais, M. J. Mucklo, and J. A. Sheena. The Fishwrap Personalized News System. In *IEEE Second International Workshop on Community Networking Integrating Multimedia Services to the Home*, 1995.

[9] P. Foltz and S. T. Dumais. Personalized Information Delivery: An Analysis of Information Filtering Methods. *Communications of the ACM*, 35(12):51 – 60, 1992.

[10] C. Fox. A Stop List for General Text. In *SGIR Forum*, volume 24, pages 19 – 35, 1990.

[11] W.B. Frakes. *Term Conflation for Information Retrieval*. Cambridge University Press, 1984.

[12] Anuja Gokhale and Mark Claypool. Thresholds for More Accurate Collaborative Filtering. In *Proceedings of the IASTED International Conference on Artificial Intelligence and Soft Computing*, Honolulu, Hawaii, USA, August9-12 1999. To appear.

[13] D. Goldberg, D. Nichols, B. Oki, and D. Terry. Using Collaborative Filtering to Weave an Information Tapestry. *Communications of the ACM*, 35(12):61 – 70, 1992.

[14] Dhruv Gupta, Mark Digiovanni, Hiro Narita, and Ken Goldberg. Jester 2.0: A New Linear-Time Collaborative Filtering Algorithm Applied to Jokes. In *Proceedings of Workshop on Recommender Systems: Algorithms and Evaluation*, August 1999.

[15] W. Hill, L. Stead, M. Rosenstein, and G. Furnas. Recommending and Evaluating Choices in a Virtual Community of Use. In *Proceedings of ACM CHI'95*, pages 194 – 201, 1995.

[16] T. Kamba, K. Bharat, and M.C. Albers. The Krakatoa Chronicle – An Interactive Personalized Newspaper on the Web. In *Proceedings of the Fourth International World Wide Web Conference*, December 11-14 1995.

[17] H.P. Luhn. The Automatic Creation of Literature Abstracts. *IBM Journal of Research and Development*, 2:159–165, 1958.

[18] Badrul M.Sarwar, Joseph A.Konstan, Al Borchers, Jon Herlocker, Brad Miller, and John Riedl. Using Filtering Agents to Improve Prediction Quality in the GroupLens Research Collaborative Filtering System. In *Proceedings of the ACM Conference on Computer Supported Cooperative Work (CSCW)*, 1998.

[19] Inc. NETPressence. CRAYON: Create Your Own Newspaper - your personalized Internet news service. Internet site: `http://crayon.net/`.

[20] M. Pazzani. A Framework for Collaborative, Content-Based and Demographic Filtering. *Artificial Intelligence Review*, 1999.

[21] P. Resnick, N. Iacovou, M. Sushak, P. Bergstrom., and J. Riedl. Grouplens: An Open Architecture for Collaborative Filtering of Netnews. In *Proceedings of Computer Supported Cooperative Work Conference (CSCW)*, pages 175 – 186. ACM SIG Computer Supported Cooperative Work, 1994.

[22] U. Shardanand and P. Maes. Social Information Filtering: Algorithms for Automating 'Word of Mouth'. In *Proceedings of ACM CHI'95*, 1995.

[23] Christopher C. Vogt, Garrison W. Cottrell, Richard K. Belew, and B.T. Bartell. Using Relevance to Train a Linear Mixture of Experts. In *Proceedings of the Fifth Text Retrieval Conference*, 1996.

[24] Ahmad M. Ahmad Wasfi. Collecting User Access Patterns for Building User Profiles and Collaborative Filtering. In *Proceedings of the 1999 International Conference on Intelligent User Interfaces*, pages 57 – 64, 1999.

[25] The Los Angeles Times (Hunter). Internet site: `http://www.latimes.com/`.

[26] The Worcester Telegram and Gazette Online (TANGO). Internet site: `http://www.telegram.com/`.

[27] Facts About Newspapers, 1998. Internet Site: `http://www.naa.org/info/facts/index.html`.