Improving the Quality of the Personalized Electronic Program Guide [†] [‡]

Derry O' Sullivan, Barry Smyth and David C. Wilson
({dermot.osullivan, barry.smyth, david.wilson}@ucd.ie)
Smart Media Institute, University College Dublin, Dublin 4, Ireland.

Kieran McDonald and Alan Smeaton
({kmcdon, asmeaton}@computing.dcu.ie)
Centre for Digital Video Processing, Dublin City University, Dublin 9, Ireland.

February 28th, 2002.

Abstract. As Digital TV subscribers are offered more and more channels, it is becoming increasingly difficult for them to locate the right programme information at the right time. The personalized Electronic Programme Guide (pEPG) is one solution to this problem; it leverages artificial intelligence and user profiling techniques to learn about the viewing preferences of individual users in order to compile personalized viewing guides that fit their individual preferences. Very often the limited availability of profiling information is a key limiting factor in such personalized recommender systems. For example, it is well known that collaborative filtering approaches suffer significantly from the sparsity problem, which exists because the expected item-overlap between profiles is usually very low. In this article we address the sparsity problem in the Digital TV domain. We propose the use of data mining techniques as a way of supplementing meagre ratings-based profile knowledge with additional item-similarity knowledge that can be automatically discovered by mining user profiles. We argue that this new similarity knowledge can significantly enhance the performance of a recommender system in even the sparsest of profile spaces. Moreover, we provide an extensive evaluation of our approach using two large-scale, state-of-the-art online systems—PTVPlus, a personalized TV listings portal and Físchlár, an online digital video library system.

Keywords: Personalization, Data Mining, Digital TV, Collaborative Filtering, Similarity Maintenance, Case-based Reasoning.

1. Introduction

Recent years have seen dramatic changes in the TV sector on a number of fronts. The advent of Digital TV (DTV) services has offered consumers a greater range of channels and programming content in addition to a host of new interactive services. In parallel, new breeds

[†] This article has not been submitted elsewhere in identical or similar form, nor will it be during the first three months after its submission to UMUAI.

[‡] The support of the Informatics Research Initiative of Enterprise Ireland is gratefully acknowledged.

of TV-based consumer devices have emerged, such as personal video recorder (PVR) technologies, such as TiVo (http://www.tivo.com/) and WinTV (http://www.hauppauge.com/). As a result, consumers are faced with a new challenge, namely how to search and browse for relevant video content—collected by a new range of consumer devices and made available through a variety of online services—in an intuitive and efficient way. The personalized electronic programme guide (pEPG) [8] and digital video library [4] comprise part of an overall answer to this challenge, and together they can provide users with a direct interface to programme content that is tailored to their needs. Such systems employ user profiling and information filtering techniques to learn about the viewing preferences of individual users in order to pro-actively promote relevant programmes.

As part of our ongoing research programme into personalized TV services, we have been developing a range of personalization and recommendation techniques that are well suited to the TV domain. In this article we describe recent work on the application of data mining methods to extract new programme metadata from user profiles, which can significantly augment knowledge about programme similarity and relevance in order to address the *sparsity problem* normally associated with collaborative filtering (CF) recommendation techniques. The sparsity problem results from the fact that, on average, two random users are unlikely to have rated many of the same items resulting in little or no direct overlap between their profiles. While we recognize the importance of issues in collaborative filtering such as the *first-rater problem* (new items added to system not being available in recommendation until they are rated by enough users) [13, 6, 26] and the *cold-start problem* (new users have not rated items causing difficulties in making valid recommendations to them) [9, 28, 30], the present work in improving recommendations is concerned solely with the sparsity issue. We evaluate our data mining approach to show that it delivers superior personalization accuracy across a range of experimental conditions. Our earlier work in this area has already shown great promise for explicit ratings-based user profiles as the central source of preference information [2], and in this article we focus on new results from the collection of implicit behavioural profile data by monitoring the use of an online server-based PVR system called Físchlár (www.fischlar.dcu.ie) [4]. In particular, we investigate the key issue of how accurate these implicit profiles are in relation to the explicit ratings-based profiles.

The remainder of this article is organised as follows. In the next section we describe the backdrop to our research, providing a background on recommendation technologies and describing our two evaluation platforms, PTVPlus and Físchlár. In Section 3 we describe our approach to improving the quality and effectiveness of recommender systems by using data mining techniques to discover new types of similarity knowledge. We go on in Section 4 to discuss how this knowledge can be exploited during recommendation. In Section 5 we describe a comprehensive evaluation of our new recommendation approach with a particular emphasis on comparing the use of explicit and implicit profiles. Before concluding and highlighting possible future avenues for research, we discuss the use of implicit and explicit data, commenting on the advantages of using implicit data for prediction of future user activities; we also look at the factors governing the potential of our technique and comment on the effect of dataset density in recommendation.

2. Background

In order to ground our discussion on recommendation, we briefly overview the key techniques that have been used to drive recommender systems, emphasizing opportunities that exist for improving these techniques. We go on to provide detailed information on the state-of-the-art PTV-Plus [8] (pEPG) and Físchlár [4] (PVR) systems that serve as testbeds for our research.

2.1. Existing Recommender Techniques

The area of research known as recommender systems combines techniques from user modelling and information filtering in order to build search systems that are better able to respond to the preferences of individual users during their search for a particular item or product. Collaborative filtering (CF) techniques generate recommendations for a target user by leveraging the preferences of a group of like-minded individuals—individuals whose profiles display a significant degree of similarity to the target user's profile [5–8, 11–13].

The success of collaborative filtering depends critically on the ability to successfully identify a suitable set of similar users to the target. Typically, collaborative filtering techniques employ simple notions of profile similarity that exploit direct overlaps or correlations between matching profile elements and their ratings. In recommending books, for example, two users who have each positively rated both *Harry Potter and the Philosopher's Stone* (by J.K. Rowling) and *The Colour of Magic* (by Terry Pratchett) can be recognised as similar on the basis of overlapping book titles alone. However, such simple measures of similarity can only operate when there is an exact alignment between profile elements (book titles in this case), and this alignment requirement greatly limits the potential to detect strong indirect similarities between profiles. For example, a third user who has positively rated Rowling's *Harry Potter and the Goblet of Fire* and Pratchett's *Wyrd Sisters* would not typically be viewed as similar to either of the previous users, even though this third user has enjoyed very similar books.

This lack of flexibility in measuring similarity gives rise to the so-called *sparsity problem* in collaborative filtering; that is, individual users typically rate only a small portion of the available items and so the expected item overlap between two random users tends to be low. In other words, the user-item ratings matrix is sparsely populated. As a consequence relevant users may be missed, and in extreme situations this may result in the failure to select any like-minded individuals as recommendation partners for a given target user.

Case-based reasoning (CBR) and, more generally, content-based approaches to recommendation can be viewed as complimentary to collaborative techniques [24, 19]. In the case-based approach, a feature-based representation of the current user's interests is used as a retrieval probe or query into a library of item descriptions and the best matching descriptions (according to some similarity metric) are retrieved. In some ways, case-based approaches to recommendation are strongly related to collaborative filtering techniques—one can usefully treat collaborative profiles as cases, and the identification of like-minded individuals can be thought of as a form of case retrieval [24]. Interestingly, case-based systems usually adopt more sophisticated models of similarity that go beyond the computation of direct overlaps between case features. Case-based systems allow for the fact that two cases may be similar even though they contain none of the same features [24, 18].

A key insight in our research is that by extending collaborative filtering systems to exploit more sophisticated models of case-based similarity, the sparsity problem may be significantly ameliorated by supporting the detection of a wider range of profile similarities. Our previous work has demonstrated the validity of using data mining techniques with CF and CBR for recommendation and its effect in combating the sparsity problem [1]. Data mining techniques have also been used in other recommendation research [14]; our approach differs with respect to use of the knowledge generated. We use mined association rules to augment standard collaborative filtering algorithms, rather than using these rules directly to provide recommendations. We will return to this issue in more detail in Sections 3 & 4.

2.2. PTVPlus - Personalized Electronic Programme Guide

PTVPlus (www.ptvplus.com) is an established online recommender system deployed in the television listings domain. It is operated commercially by ChangingWorlds (www.changingworlds.com). PTVPlus uses its recommendation engine to generate a set of TV programme recommendations for a target user, based on their profiled interests, and it presents these recommendations in the form of a personalized programme guide (see Figure 1). The uniqueness of PTVPlus stems from its combination of complementary recommendation results from separate collaborative and case-based recommendation strategies. The key to PTVPlus's personalization facility is an accurate database of interactively acquired user preference profiles that contain collaborative filtering style ratings lists. These are employed directly in the collaborative filtering component and by transformation to a content summary

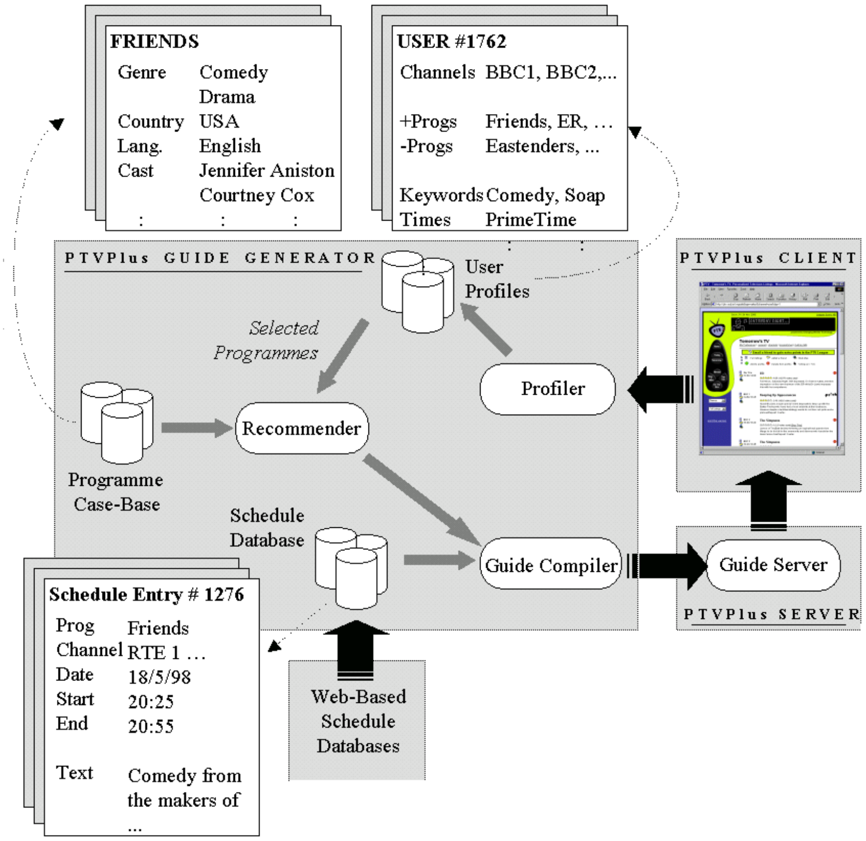


Figure 1. Structure of the PTVPlus system.

profile schema for matching in the case-based component. While this means that each set of recommendations can help to make up for the shortfalls of the other set's personalization strategy, each set still faces the limitations of its own strategy and the information that it has at its disposal.

Recently there has been an opportunity to couple PTVPlus technology with the Físchlár PVR system by integrating the PTVPlus-ClixSmart personalization engine into Físchlár. The enhanced Físchlár system maintains a distinct user-base from PTVPlus (and thereby datasets in our evaluation), but it helps to address a number of issues that both systems have experienced on their own. For example, users of the Físchlár system were hampered by a rudimentary static electronic programme guide as the primary recording and playback interface. The result was the need to browse through many pages of listings content in order to locate a programme to record. By integrating PTVPlus' personalized TV listings service it is now possible to offer users a far more effective personalized EPG interface.

Moreover, one of the key problems with the PTVPlus system has been the fact that it has always been disconnected from the TV domain in the sense that users could not access the PTVPlus listings through their TV set and so the tasks of locating programming content and viewing programming content were separated. One implication of this is that user preferences could only be gathered by asking users to provide explicit preference information, either in the form of programme lists or programme ratings. By integrating PTVPlus and Físchlár technologies users can now access TV listings and view programming content from a single location and interface and this leads to new types of profiling possibilities as we shall see.

The implementation of PTVPlus has been described in detail within the recommendation literature [9, 10]. Here we take the opportunity to present details on the implementation of Físchlár.

2.3. The Físchlár Digital Video Library System

Físchlár is a video library system which allows users to record, browse, search and watch television programmes online using their web browser. Users browse personalized television schedules, provided by the PTVPlus-ClixSmart personalization engine [8], to select programmes to be recorded by the system. These programmes are captured digitally and then analysed to support browsing and interactive searching of their content. While browsing a programme, a user can decide to play all or part of that programme. The following sections describe the Físchlár system architecture and user interface.

2.3.1. *System Architecture*
The architecture of Físchlár is shown in Figure 2. The television schedules are provided by the PTVPlus-ClixSmart personalization engine, which also provides users with recommendations of programmes to record and browse from the library of recorded programmes. Each programme has a title, description, channel and themes associated with it. Programmes are recorded from any of the eight terrestrial free-to-air channels in Ireland. The programmes are categorised into themes—Comedy, Drama, Soaps, Movies, Science-Fiction, Music, Sports, Kids, Teens, Food & Drink, Science & Nature, Home & Garden and Games & Quizzes.

TV programmes are captured at sub-VHS quality in MPEG-1 format. Each video capture device is a PC with a television tuner card and a video encoder card. A queue of programmes to be captured by each recording device is maintained by the system. When a programme that is to be captured begins transmission, the tuner switches the tuned signal to the programme's channel and the encoder starts encoding the signal to an MPEG-1 file. After the programme is captured, the capture device waits until there is another programme to be recorded. While results in this article represent an earlier deployment with only a single capture device available, there are currently three capture devices running thus allowing capture of three programmes from different channels simultaneously. The teletext closed captions for each of the 8 channels are continuously captured. These are broadcast for people with hearing

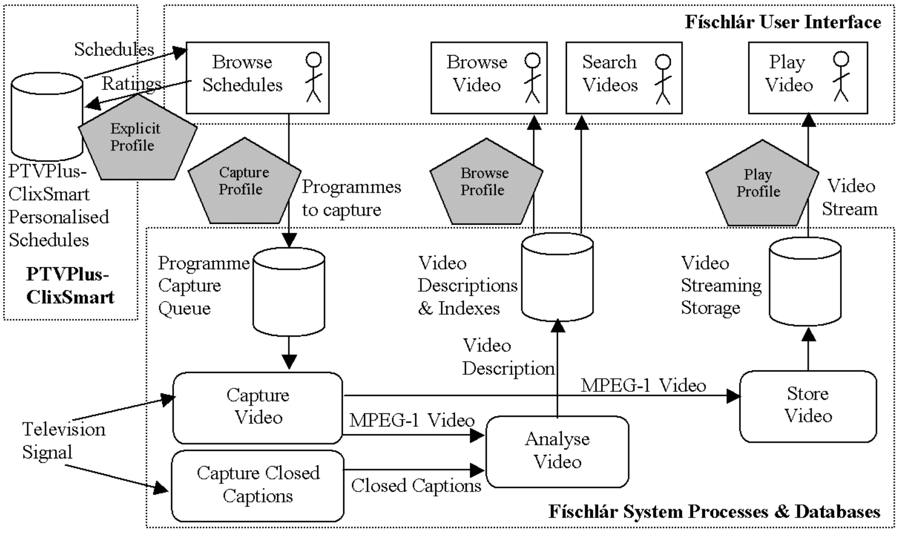


Figure 2. Structure of the Físchlár system.

difficulties and they are considered a close transcription of what is said during the programme. These closed captions are integrated into the programme's description when a captured video is being analysed. Each closed caption capture device consists of a PC with multiple teletext tuner cards each tuned to a fixed channel.

A programme's video content is analysed after it is completely captured. The analysis consists of shot-boundary detection and shot keyframe identification and extraction. A shot is a continuous sequence of frames taken from a single camera and is considered the physical editing structure for composing a video programme. Físchlár identifies shot boundaries by first calculating the YUV histogram difference between adjacent frames in the video and then applying a dynamic threshold to these values [15]. A keyframe is a frame (image) chosen judiciously from the shot in order to best represent the shot's contents. Físchlár selects the frame with maximum similarity to the average histogram of the frames in the shot as the keyframe. A visual overview of the programme is also generated by selecting a fixed number of keyframes from equally spaced temporal locations across the video.

The description of each video therefore consists of programme title, description, associated themes and broadcast information (channel and timing) supplied by the PTVPlus-ClixSmart system, as well as the closed captions, shot boundaries, shot keyframes and overview keyframes generated from an analysis of the recorded broadcast. This description is stored and retrieved using the MPEG-7 standard for describing audio-visual material [16]. When the description is completed text information retrieval indices are updated to index the text from the video's description. This index supports text-based searching of the captured programme's closed captions.

After the video is analysed the MPEG-1 video file is stored on a video streaming server in order to support efficient access to any point within the video. The storage capacity of the video server is limited to 400 hours and therefore whenever a video is added, space must be made available by deleting existing content. Currently, the strategy is simply to delete the oldest recorded content in order to make space available for new content. The architecture of the web-based system supports the separation of application logic and presentation logic (see Figure 3). Each web request that executes application logic produces its response at the server side as XML and this XML response is transformed at the server side into HTML using XSL stylesheets. The stylesheet is chosen based upon the web request and the client device—either a PC or a PDA with a wireless LAN connection. The profiles analysed in this work are taken from the PC version of the system.

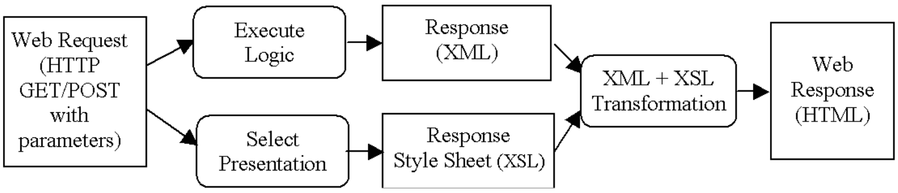


Figure 3. Web architecture for handing a request and producing HTML output.

2.3.2. *User Interface*

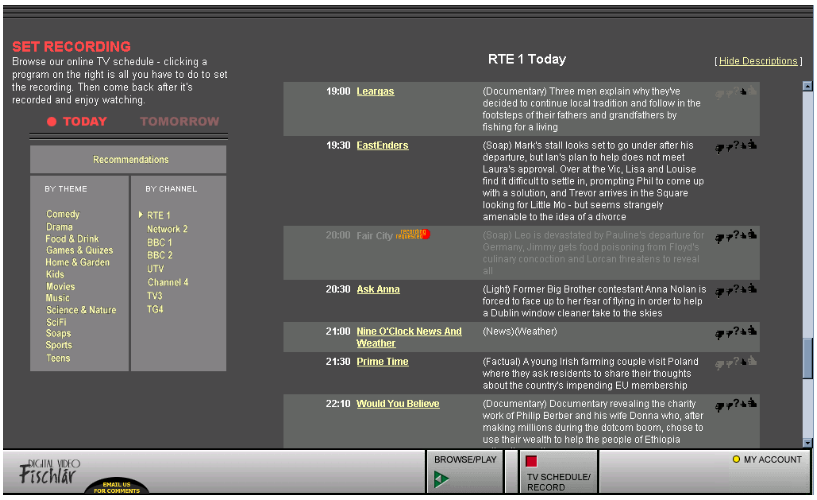
In this section we briefly present the user interface for recording programmes and the interface for browsing, searching and playing from recorded programmes.

2.3.2.1. *User Interface for Recording Programmes*

To record programmes, users browse today's and tomorrow's television schedules organised by recommendations, by themes and by channels. The recommendations are a union of the programmes the user has previously indicated they like and the programmes the PTVPlus-ClixSmart system suggests they may like. When users browse the TV schedules they can indicate which programmes they like or dislike by using the thumb icons beside each programme's description (see Figure 4). The PTVPlus-ClixSmart system uses a five point scale—love, like, no comment, dislike, and hate—for the user to rate a programme. A programme is recorded by simply clicking on its title. When the capture devices cannot record a programme because it is past transmission time or too many programmes are simultaneously being captured its title is greyed out. Currently, each user is limited to recording a maximum of 5 programmes per day.

2.3.2.2. *User Interface to Browse, Search, and Play Recorded Programmes*

All programmes captured and analysed by Físchlár are available for all users to browse, search and play. Programmes can be browsed by date, by recommendation, by theme and by channel. Users can also search for programmes using a text query matched against the programmes' closed captions. When users select a programme from a category listing or search results its content is displayed and can be browsed at two levels—overview and detail view. The overview is the default presentation view and its purpose is to give the users an at-a-glance summary of the programme's content (see Figure 5). The purpose of the detail view is to allow more detailed shot based browsing of the content. There are four keyframe browsers in Físchlár—Slide



SET RECORDING

Browse our online TV schedule - clicking a program on the right is all you have to do to set the recording. Then come back after it's recorded and enjoy watching.

● TODAY TOMORROW

Recommendations

BY THEME

Comedy
Drama
Food & Drink
Games & Quizes
Home & Garden
Kids
Movies
Music
Science & Nature
SciFi
Soaps
Sports
Teens

BY CHANNEL

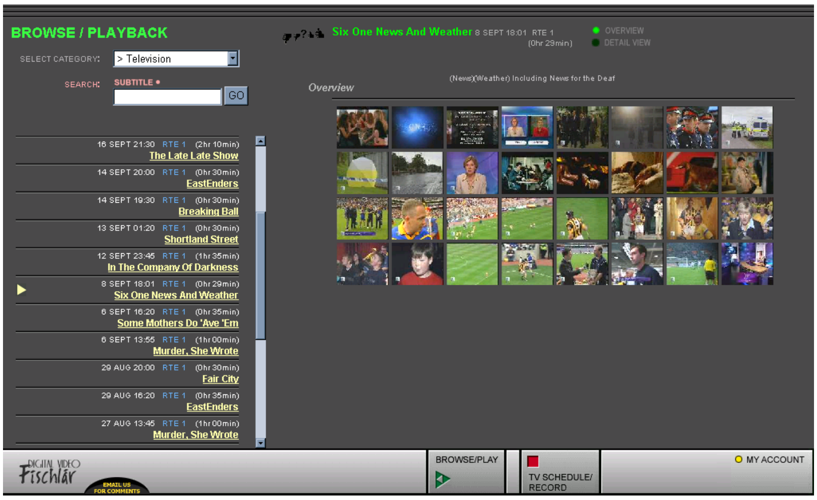
RTE 1
Network 2
BBC 1
BBC 2
UTV
Channel 4
TV3
TG4

RTE 1 Today [Hide Descriptions]

19:00	Leargas	(Documentary) Three men explain why they've decided to continue local tradition and follow in the footsteps of their fathers and grandfathers by fishing for a living
19:30	EastEnders	(Soap) Mark's stall looks set to go under after his departure, but Ian's plan to help does not meet Laura's approval. Over at the Vic, Lisa and Louise find it difficult to settle in, prompting Phil to come up with a solution, and Trevor arrives in the Square looking for Little Mo - but seems strangely amenable to the idea of a divorce
20:00	Fair City recording requested	(Soap) Leo is devastated by Pauline's departure for Germany, Jimmy gets food poisoning from Floyd's culinary concoction and Lorcan threatens to reveal all
20:30	Ask Anna	(Light) Former Big Brother contestant Anna Nolan is forced to face up to her fear of flying in order to help a Dublin window cleaner take to the skies
21:00	Nine O'Clock News And Weather	(News)(Weather)
21:30	Prime Time	(Factual) A young Irish farming couple visit Poland where they ask residents to share their thoughts about the country's impending EU membership
22:10	Would You Believe	(Documentary) Documentary revealing the charity work of Philip Berber and his wife Donna who, after making millions during the dotcom boom, chose to use their wealth to help the people of Ethiopia

Figure 4. Interface to select programmes to be recorded on Físchlár.

show, Scrollbar, Timeline and Hierarchical—and these are described in [17]. Each displays keyframes and closed captions in a different way—temporally, spatially, segmented spatially, and hierarchically. Clicking on a keyframe in any of these views starts playback from the start of the corresponding shot.



BROWSE / PLAYBACK

SELECT CATEGORY: > Television

SEARCH: SUBTITLE ● GO

16 SEPT 21:30 RTE 1 (2hr 10min)
The Late Late Show

14 SEPT 20:00 RTE 1 (0hr 30min)
EastEnders

14 SEPT 19:30 RTE 1 (0hr 30min)
Breaking Ball

13 SEPT 01:20 RTE 1 (0hr 30min)
Shortland Street

12 SEPT 23:45 RTE 1 (1hr 35min)
In The Company Of Darkness

8 SEPT 18:01 RTE 1 (0hr 29min)
Six One News And Weather

6 SEPT 16:20 RTE 1 (0hr 35min)
Some Mothers Do 'Ave 'Em

6 SEPT 13:55 RTE 1 (1hr 00min)
Murder, She Wrote

29 AUG 20:00 RTE 1 (0hr 30min)
Fair City

29 AUG 16:20 RTE 1 (0hr 35min)
EastEnders

27 AUG 13:45 RTE 1 (1hr 00min)
Murder, She Wrote

Six One News And Weather 8 SEPT 18:01 RTE 1 (0hr 29min)

● OVERVIEW
● DETAIL VIEW

(News)(Weather) Including News for the Deaf

Overview

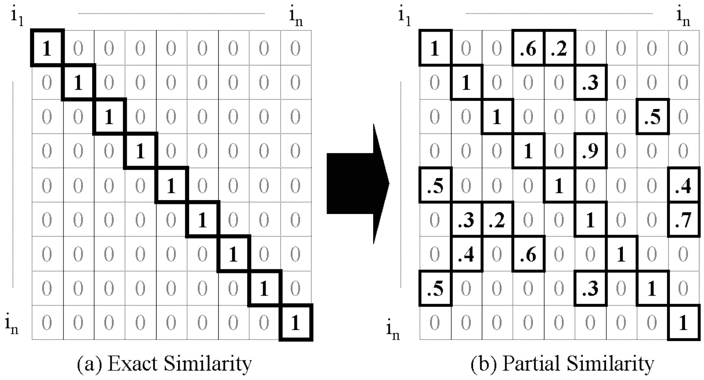
Figure 5. Browse a programme's content, overview level is shown.

3. Mining Similarity Knowledge

A central objective in our research is to explore ways of improving the quality and scope of recommender systems by looking for opportunities to solve critical problems that exist with current approaches. In this regard, a central contribution of this article is our description of a novel recommendation technique that attempts to solve the sparsity problem associated with collaborative filtering in two distinct stages:

- We use association rule mining methods to discover new similarity knowledge among user profiles;
- We harness this newly generated knowledge using case-based approaches in collaborative recommendation.

3.1. Similarity Knowledge in Collaborative Recommendation



1	0	0	0	0	0	0	0	0
0	1	0	0	0	0	0	0	0
0	0	1	0	0	0	0	0	0
0	0	0	1	0	0	0	0	0
0	0	0	0	1	0	0	0	0
0	0	0	0	0	1	0	0	0
0	0	0	0	0	0	1	0	0
0	0	0	0	0	0	0	1	0
0	0	0	0	0	0	0	0	1

(a) Exact Similarity

1	0	0	.6	.2	0	0	0	0
0	1	0	0	0	.3	0	0	0
0	0	1	0	0	0	0	.5	0
0	0	0	1	0	.9	0	0	0
.5	0	0	0	1	0	0	0	.4
0	.3	.2	0	0	1	0	0	.7
0	.4	0	.6	0	0	1	0	0
.5	0	0	0	0	.3	0	1	0
0	0	0	0	0	0	0	0	1

(b) Partial Similarity

Figure 6. The item-item similarity knowledge available in standard CF approaches (a) can be extended by data mining methods to support partial similarities (b).

In Section 2 we mentioned that conventional collaborative filtering approaches tend to rely on very limited models of profile similarity by exploiting simple overlap or correlation metrics during similarity assessment. In other words, profiles can only be compared to the extent to which they contain the same profile items. From a CBR perspective, treating profiles as cases and (rated) profile items as features, this is equivalent to exploiting an item-item similarity matrix in which all entries are 0 (zero similarity) except for the diagonal entries, which are 1 (corresponding to a perfect exact match), as shown in Figure 6(a).

Clearly the limited availability of similarity knowledge significantly reduces the opportunity to detect and exploit the profiles of similar users. Moreover, the assumption that non-identical items have a similarity of zero is clearly flawed; non-identical items may indeed be very similar. Our key objective in this work is to seek out ways of supplementing this meagre similarity knowledge with additional knowledge so that non-exact item matches can be accommodated where appropriate. Moreover, we believe that this additional knowledge can be derived directly from the user profiles by looking for frequently co-occurring pairs (or even groups) of items. For example, suppose that 75% of the time that users prefer item i, they also prefer item j. This would suggest that i and j are similar in some unknown way, and one could use 0.75 as a proxy for their degree of similarity. Thus, by detecting frequent co-occurrences in this fashion we may be able to significantly extend the non-zero entries in the item-item similarity matrix, as shown in Figure 6(b).

In the following sections we detail how efficient data mining techniques can be used to discover this type of similarity knowledge and how this knowledge can be usefully exploited to generate high-quality recommendations. We should emphasize that data mining is one of many possible approaches to generating additional similarity knowledge; we have simply chosen data mining as a reasonable initial technique to demonstrate the feasibility of our new recommendation strategy. Other possibilities that we hope to investigate in future work include Singular Value Decomposition (SVD) [29, 37, 31], Latent Semantic Indexing (LSI) [32, 33], and Principal Component Analysis (PCA) [34, 36, 35]. Indeed, it is worth highlighting that SVD and PCA methods have already been used in the past to address the sparsity problem [34, 36], although they have not been used to generate similarity knowledge. We would also like to compare our full recommendation strategy to techniques involving the usage content-based techniques, whether hybrid or independent, such as in [38, 39].

3.2. Association Rule Mining

The Apriori algorithm [21] is a well-known data mining technique that can be used to efficiently discover similarity knowledge from profile cases by finding frequently occurring associations between profile items (TV programmes), and by assigning confidence scores to the associations. These association rules indicate which non-identical items can be considered to be similar, and their associated confidences can be used as a proxy for the level of similarity. In turn, these direct rules can be chained together to learn additional associations and similarities in order to further elaborate the item-item similarity matrix.

Association rules are of the form $A \Rightarrow B$, where A and B are sets of items (television programmes). In data mining terms, whenever a transaction (case) T contains a certain itemset (set of programmes) A, then the transaction probably contains another itemset B.

The probability that a given rule holds, *rule confidence*, is the percentage of transactions containing B given that A occurs:

$$P(B \subseteq T | A \subseteq T) \tag{1}$$

The *support* of an itemset A is defined as the fraction of transactions supporting A with respect to the entire database. The support of a rule $A \Rightarrow B$, then, is the probability that both itemsets occur together in a transaction:

$$\mathit{support}(A \Rightarrow B) \ = \ P((A \cup B) \subseteq T) \tag{2}$$

The measure of rule confidence is related to support, and can be computed as follows:

$$confidence(A \Rightarrow B) \ = \ \frac{\mathit{support}(A \cup B)}{\mathit{support}(A)} \tag{3}$$

When mining association rules, the confidence and support values are used to constrain exponentially large candidate rule sets by setting thresholds.

The Apriori algorithm is designed to efficiently process a database of transactions to discover well-supported association rules. In brief it operates by finding the set of frequently occurring items and developing patterns between these.

3.3. Direct Item-Item Similarities

By treating PTVPlus user profiles as transactions and the rated programmes therein as itemsets, the Apriori algorithm can be used to derive a set of programme-programme association rules and confidence values, as shown in Table I. The confidence values are taken as similarity scores and used to fill in the corresponding entry in the item-item similarity matrix, as shown in Table II.

Table I. Sample association rules.

Rule	Support	Confidence
Friends ⇒ Frasier	13%	24%
Friends ⇒ ER	12%	20%
Frasier ⇒ ER	7%	23%

Table II. A simple item-item similarity matrix.

	Friends	*Frasier*	*ER*
Friends	1	.24	.20
Frasier	-	1	.23
ER	-	-	1

3.4. Indirect Item-Item Similarities

The method described in Section 3.3 discovers *direct* similarity knowledge, and while it has the capacity to significantly extend the coverage of the item-item similarity matrix, in all likelihood many similarity entries will still remain zero. Two extensions to the above approach suggest themselves as a means of further extending similarity knowledge coverage.

One possibility is to exploit similarity symmetries that might exist to potentially double the number of item-item similarities. In other words, learning an association $A \Rightarrow B$ may indicate that the inverse association could also hold, that $B \Rightarrow A$—the question remains as to how to calculate the confidence associated with this inverse association. Early experiments, however, showed that the confidence levels necessary to provide good symmetric correlation incurred a high cost in terms of programme similarity matrix coverage [1]. Thus we have chosen to ignore this aspect at present, noting it instead for future work.

Alternatively, association rules may be chained to learn indirect similarities. For example, discovering rules $A \Rightarrow B$ and $B \Rightarrow C$ may indicate that A and C are also related. We explore this strategy here along with different methods for combining the confidence of individual rules in the chain to derive a confidence value for the new indirect association (e.g., computing the maximum, minimum or mean confidences along the chain).

Building and using this type of item-item similarity knowledge to address the sparsity problem is similar in spirit to *item-based collab-*

orative techniques [22]. The item-based collaborative approach uses rating overlaps to build item-item similarities, and suggested items are retrieved in direct comparison to the elements that comprise a user profile. We believe that the social mechanism of first recommending users and subsequently deriving recommended items from those users is fundamental to preserving the natural diversity of collaborative approaches. A lack of diversity in recommendation is a problem faced by content-based approaches; in directly recommending items, the item-based approach may also incur a similar diversity cost. We expect to investigate comparisons with item-based techniques in future work.

4. Similarity Based Recommendation

The availability of item-item similarity knowledge facilitates a new type of similarity-based recommendation strategy that combines elements from case-based and collaborative techniques. It allows the use of more sophisticated CBR-like similarity metrics on ratings-based profile data, which in turn makes it possible to leverage indirect similarities between profile cases, and so generate improved recommendation lists. This new recommendation strategy consists of two basic steps:

1. The target profile, t is compared to each profile case, $c \epsilon C$, to select the k most similar cases.

2. The items contained within these selected cases (but absent in the target profile) are ranked according to their relevance to the target, and the r most similar items are returned as recommendations.

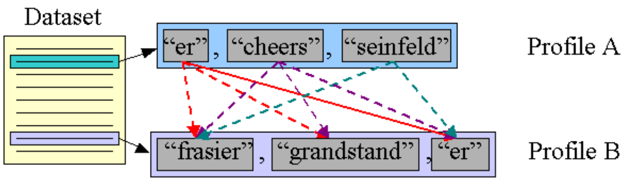
4.1. Profile Comparison

The profile similarity metric is presented in Equation 4 as the sum of the similarities between the items in the target and source profile cases. In the situation where there is a direct correspondence between an item in the source, c_i, and the target, t_j, the maximal similarity is assumed (Equation 5). However, the nature of ratings-based profile cases is such that these direct correspondences are rare and in such situations the similarity value of the source profile item is computed as the mean similarity between this item and the n most similar items in the target profile case ($t_1 ... t_n$) (Equation 6).

$$PSim(t, c, n) = \sum_{c_i \epsilon c} ISim(t, c_i, n) \qquad (4)$$

$$\begin{aligned} ISim(t,c_i,n) &= 1,\ if\ \exists\, t_j\, \epsilon\, t:\ t_j = c_i \qquad &(5)\\ &= \frac{\sum_{j=1..n} sim(t_j,c_i)}{n},\ otherwise \qquad &(6) \end{aligned}$$

Future work will look at alternatives to Equation 6 in order to find the best method for calculating the similarity between an item and a profile; one possibility would be to take the maximal item-item similarity rather than the mean. Figure 7 shows how the above equations work on given profiles, with a solid arc showing item overlap and dashed arcs representing knowledge gained through association rules.



ISim(B, “er”, 3)	=		**1**	Identical Correspondence
“er”	=>	“er”	1	
ISim(B, “cheers” , 3)	=		**0.5**	Rule-based Correspondence
“cheers”	=>	“frasier”	0.8	
“cheers”	=>	“grandstand”	0.4	
“cheers”	=>	“er”	0.3	
ISim(B, “seinfeld” , 2)	=		**0.6**	
“seinfeld”	=>	“frasier”	0.9	
“seinfeld”	=>	“er”	0.3	
“seinfeld”	=>	*“grandstand”*		No Correspondence Available
PSim(B, A, 3)	=		**2.1**	

Figure 7. An example of how profile A can be matched against profile B and their similarity computed.

4.2. Recommendation Ranking

Once the k most similar profile cases, $\hat{C}$, to the target have been identified, a set of ranked item recommendations is produced. There are three factors to consider when ranking these recommendations. First, we want to give priority to those items that have a high similarity to the target profile case. Second, items that occur in many of the retrieved profile cases should be preferred to those that occur in few profile cases. Finally, items recommended by profiles similar to the target should be preferred to items recommended by less similar profiles. Accordingly, we compute the *relevance* of an item, c_i, from a retrieved profile case,

c, with respect to the target profile, t, as shown in Equation 7; where $C' \subseteq \hat{C}$ is the set of retrieved profile cases that contain c_i.

$$Rel(c_i, t, \hat{C}) = ISim(c_i, t, k) \cdot \frac{|C'|}{|\hat{C}|} \cdot \sum_{c \epsilon C'} PSim(c, t) \tag{7}$$

In order to test our recommendation ranking function against pure CF methods, we also implement a CF-adapted version of our recommendation ranking method (*CFRR*). For each unique item in our list of recommended profiles c_i, where $C' \subseteq C$ is the set of retrieved profiles that contain c_i, and a target t:

$$Rel(c_i, t) = \frac{|C'|}{|C|} \cdot \sum_{c \epsilon C'} CFSim(c, t) \tag{8}$$

where $CFSim(c, t)$ is the standard collaborative filtering similarity found by calculating the percentage of overlapping programmes between profiles c and t.

5. Experimental Results

We are focused on addressing the sparsity problem, which limits the ability of collaborative filtering recommender systems to recognise vital profile similarities unless there is a direct overlap between their rated items. Our proposed solution has two parts. First, association-rule mining techniques are used to discover item-item similarity knowledge from ratings-based user profiles of the sort normally found in collaborative filtering recommender systems. Second, a similarity-based recommendation strategy takes advantage of this new similarity knowledge to measure the similarity of user profiles even when there is no direct profile overlap, and generates ranked recommendations accordingly.

Of course the success of this new approach depends critically on its ability to generate accurate recommendations and so in this section we fully evaluate its recommendation accuracy characteristics across a range of data-sets from the DTV domain. In addition the availability of profile data from the DTV domain, which combines explicit and implicit user ratings, provides an interesting opportunity to evaluate the relative utility of explicit and implicit ratings with respect to recommendation accuracy.

Finally the data-sets used in this evaluation have different density characteristics and this makes it possible to gain some concrete insight into the precise impact of the sparsity problem on recommendation performance. We will reflect on this issue in the next section when we examine the relationship between recommendation accuracy and dataset density in more detail.

5.1. Experimental Setup

The present evaluation uses a range of different profile data-sets from the DTV domain to test the recommendation accuracy of a number of different recommendation algorithms including standard collaborative filtering and a number of variations of our association-rule mining strategy.

5.1.1. *Test Data*

The PTVPlus and Físchlár systems are used to provide a range of different profile data-sets collected from real user interactions with both systems over the past two years: PTVPlus has been fully deployed and freely accessible since 1999 while Físchlár has been available to staff and students at the Dublin City University campus since 2000. The user profiles available from both systems are made up of explicit programme ratings as described in Section 2; although it is worth noting that for the purpose of this evaluation only positive ratings are used in the PTVPlus and Físchlár profiles. Físchlár has the added advantage of providing implicit profiles because the recording, playback and browsing actions of Físchlár users are tracked and can be translated directly into simple implicit ratings. As a result the following individual profile datasets are used:

- **PTVPlus**: 622 profiles containing explicit programme ratings from PTVPlus users.
- **Físchlár**: 650 Físchlár profiles containing explicit programme ratings equivalent to those found in the PTVPlus data.
- **Record**: 650 Físchlár profiles containing only those programmes that a user has selected for recording.
- **Browse**: 650 Físchlár profiles containing only those programmes that a user has browsed.
- **Play**: 650 Físchlár profiles containing programmes that the user has played back.
- **Combined**: 650 Físchlár profiles made up of programmes that the user has recorded, browsed or played.

The above profile datasets are composed of profiles selected randomly from the PTVPlus and Físchlár user populations. Also, a very simple approach was taken when translating Físchlár record, playback and browse behaviours into implicit ratings. For example, each Play profile contains all programmes selected for playback by a particular user, even though some of these playback actions may be less reliable preference indicators. For instance, a more reliable alternative might be to impose a minimum playback time before a programme is considered relevant to a user to eliminate spurious playbacks; this is left for future work.

5.1.2. *Test Algorithms*

Two basic collaborative filtering algorithms are used as benchmarks against which to evaluate the success of our family of similarity-based recommendation methods. We test a number of variations on our approach: one relies on *direct* association rules only as the basis for similarity knowledge, while a set of variations seek to exploit *indirect* similarity knowledge based on rule chaining, each differing in the way that the individual rule confidences are combined. In summary, the following test algorithms are evaluated:

- **CF**: A basic collaborative filtering algorithm.
- **CFRR**: CF adapted to including the ranking technique described in Section 4.2.
- **DR**: The so-called direct approach to similarity-based recommendation in which only those association rules discovered by Apriori are used for similarity knowledge (see Section 3.3).
- **Indirect**: A range of so-called indirect approaches to similarity recommendation that utilise additional similarity knowledge by chaining the mined association rules. **Max, Min, Add, Mult, Av** variations are implemented; these differ depending on how the final similarity value is computed from the confidence values of a set of chained rules. For example, Max simply takes the maximum of the chained confidence values, while Av takes an average (mean) and Mult computes the product.

5.1.3. *Methodology & Metrics*

Before describing our basic evaluation methodology and metrics it is worth highlighting an important issue that arises out of the use of Apriori as our rule generation technique. The Apriori algorithm is not without its parameters and it requires confidence and support thresholds to be set in order to constrain rule generation. A complete discussion of this issue is beyond the scope of this article but the interested reader is referred to [1] for a detailed treatment. The tuning policy described in this previous work was used to select a confidence threshold of 10% for all datasets, and support thresholds of 5% for the explicit datasets and 10% for the implicit datasets.

In order to evaluate recommendation accuracy we need some way to judge whether the recommendations made by a particular algorithm are relevant to a target user. Ideally, of course, we would like to avail of the services of the 600+ users that make up the PTVPlus and Físchlár profiles, but unfortunately, for very practical reasons this is not possible. Instead, we adopted a standard evaluation methodology in which each profile set is divided into a *training set*, containing 40% of the profiles, and a *test set*, containing the remaining 60% of the profiles. The training profiles are used as the basis for collaborative and similarity-based recommendation by the various test algorithms. In other words, only the ratings in the training profiles are used as the basis for profile comparisons prior to recommendation and only the training profiles are mined to discover similarity knowledge for similarity-based recommendation.

In our experiments we vary recommendation set size from 10 to 90 items and once a set of recommendations is generated for a specific user with a given recommendation algorithm, their accuracy is evaluated with reference to the ratings contained in the test profile for that user; that is, those recommendations that we know to be relevant.

In this evaluation our primary accuracy metric measures the percentage of test profile items that are present in a user's recommended set; this is equivalent to the standard recall metric used in information retrieval. So, for example, if all of the items in a user's test profile are contained within their recommended set a maximum recall of 100% is achieved.

Recall: The proportion of items in the user's test profile that are recommended, averaged over all users.

In general recall is a strong measure of recommendation accuracy and it should be noted that in our evaluation it serves as a *lower-bound* on real recommendation accuracy. This is because the only way that we can judge a recommendation to be relevant is if it exists in the user's test profile, which of course represents only a limited subset of those recommendations that are truly relevant to the user. Many of the recommendations made may well be relevant to a user, but go unnoticed simply because at the time the profiles were harvested the user had not yet rated, recorded, played, or browsed these relevant programmes.

With this in mind we also introduce a weaker notion of recommendation accuracy, which we call *hit rate*. The basic idea is that a given set of recommendations has at least some measurable value of usefulness to a user, when it contains at least one recommendation from the user's test profile. A maximum hit rate of 100% indicates that a given algorithm always makes at least one relevant recommendation (present within the user's test profile) per recommendation session.

Hit Rate: The proportion of users for which at least one item from the user's test profile is recommended.

5.2. Recommendation Accuracy

In the following experiments we measure the recall and hit rate statistics across all profile datasets and test algorithms. In addition, we distinguish between two types of accuracy experiment depending on whether the training profiles and the test profiles are taken from the same original profile set. For example, in the intra-dataset experiment the training profiles and the test profiles come from the same dataset; that is, if the recommender contains training profiles from the Play dataset then its recommendations are evaluated with respect to the corresponding test profiles in the Play dataset. In contrast, in the inter-dataset experiment a recommender that contains training profiles from the Play dataset might be evaluated with respect to test profiles from another dataset. By comparing respective results, it is possible to evaluate the degree to which explicit ratings are useful predictors of real user behaviours such as recording and playback.

5.2.1. *Intra-Dataset Accuracy*

Figures 8, 9, 10, 11, 12 and 13 present the recall and hit rate scores for the various recommendation algorithms across the six datasets. For example, the recall results for the PTVPlus dataset (see Figure 8) indicates that the DR algorithm, which exploits similarity knowledge derived directly from the Apriori association rules, achieves a recall of 20%. In other words, on average, 20% of test profile items are present in the recommendations made by DR. In fact DR outperforms all of the algorithm variations that exploit indirect rules generated by rule-chaining. These indirect algorithms achieve a maximum recall of 18%. In turn the basic collaborative filtering algorithms, CFRR and CF, that do not avail of the mined similarity knowledge, achieve significantly lower recall levels of 7% and 8%, respectively. A similar ordering of algorithms is found in the hit rate values for the PTVPlus dataset. The difference between indirect and CF algorithms as well as direct

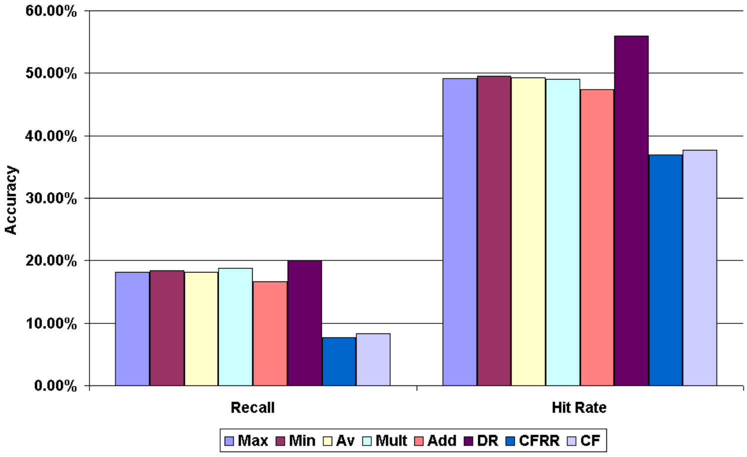


Figure 8. PTVPlus recommendation accuracy.

and CF algorithms was found to be significant at the 99% level. These results highlight the usefulness of the mined similarity knowledge during recommendation; in the PTVPlus dataset the DR recall value, with its direct similarity knowledge, is 2.5 times greater than the standard CF recall results. However, the indirect recall and hit rate values are less than those for DR. This suggests that the additional similarity knowledge available to the indirect algorithms is of lower quality that the knowledge available to DR.

The remaining datasets follow a similar trend to that found for PTVPlus, with direct methods outperforming indirect, and indirect outperforming standard collaborative filtering (Figures 9, 10, 11, 12 and 13). Again, we found that the difference between algorithms using our association rule approach and standard CF algorithms was significant at the 99% level. This trend is also summarised in Figures 14 and 15 as graphs of recall and hit rate versus datasets for each type of recommendation technique; note that instead of graphing the individual recall and hit rate values for each indirect technique we simply take the average recall and hit rate value over all indirect methods. From these summaries the dominance of DR over indirect over CF techniques should be clear—there is a clear and consistent separation between the DR and indirect recall (or hit rate) values and the CF recall (or hit rate) values. Overall these summary results indicate that there is considerable added-value to be derived from the newly generated similarity knowledge when it comes to recommendation accuracy. On average, across all datasets, the DR recall value is 3.4 times that of the CF recall value, and the DR hit rate value is 1.6 times the CF hit

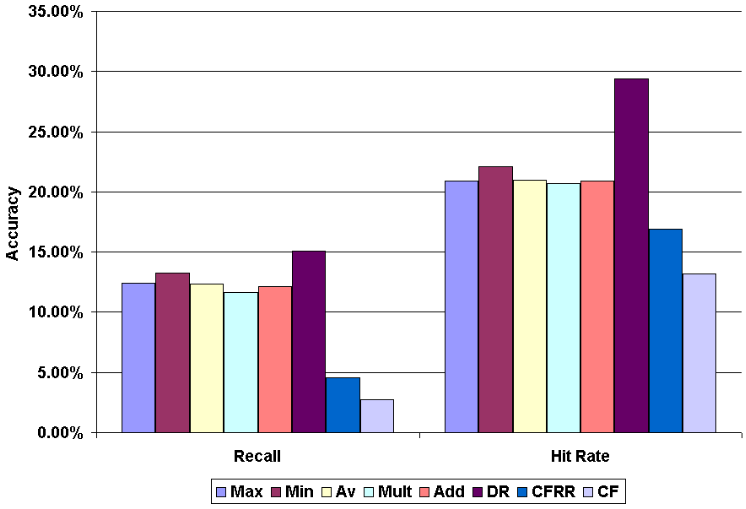


Figure 9. Físchlár recommendation accuracy.

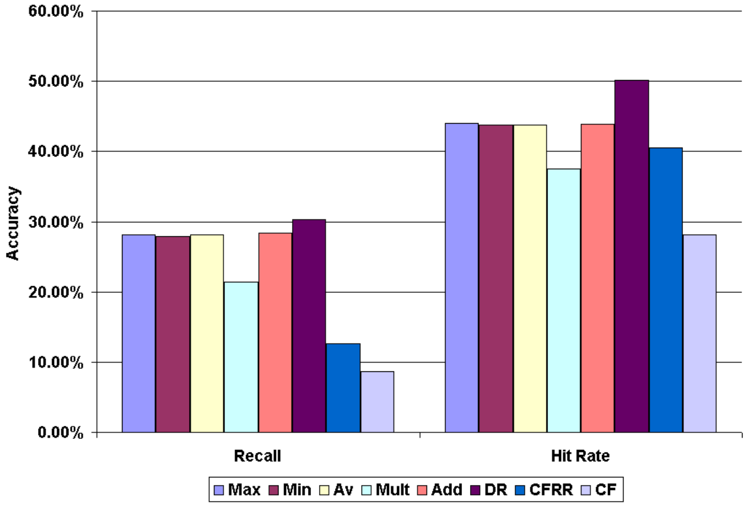


Figure 10. Play behaviour recommendation accuracy.

rate value. The fact that the direct similarity knowledge outperforms the indirect knowledge perhaps should not be so surprising. It indicates that the chaining procedure has a tendency to produce lower-quality similarity knowledge—this may be expected given the nature of the chaining and confidence combination process.

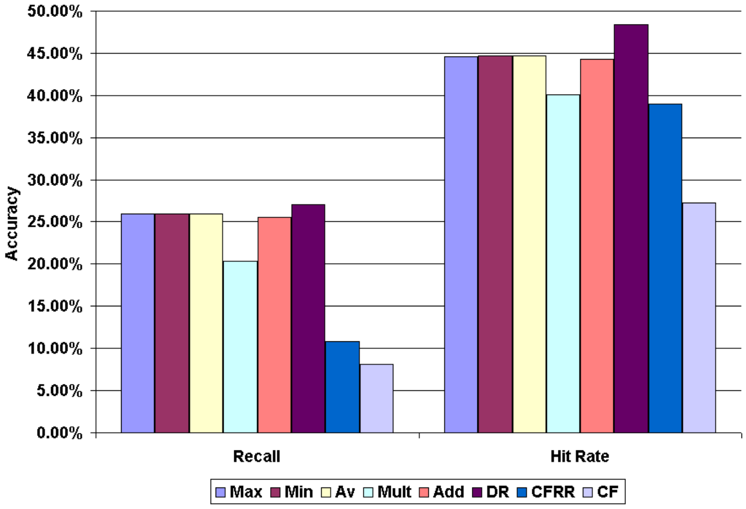


Figure 11. Record behaviour recommendation accuracy.

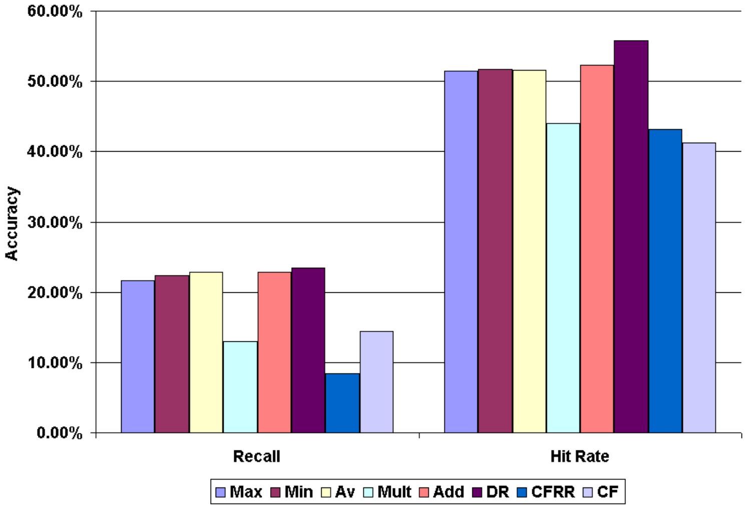


Figure 12. Browse behaviour recommendation accuracy.

5.2.2. *Inter-Dataset Accuracy*

In the previous section we tested and evaluated recommendation accuracy by drawing training and test profiles from the same dataset. Because of the relationship that exists between the five Físchlár datasets (Físchlár, Play, Record, Browse, Combined) it is also possible, and informative, to examine the impact of using various combinations of

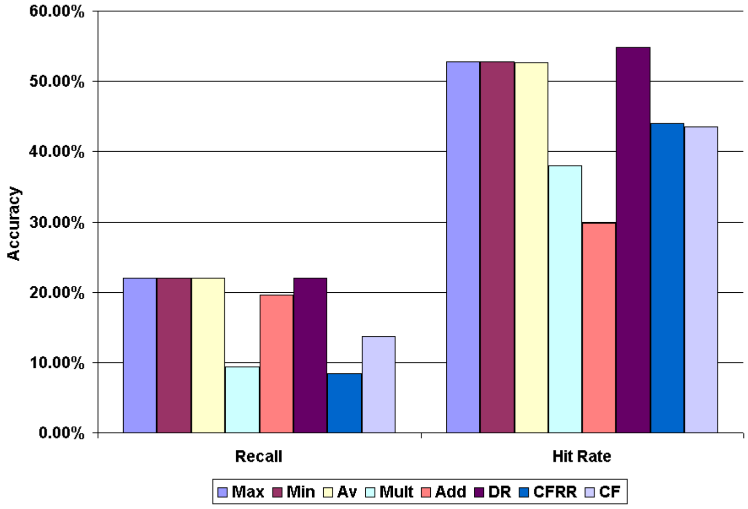


Figure 13. Combined behaviours recommendation accuracy.

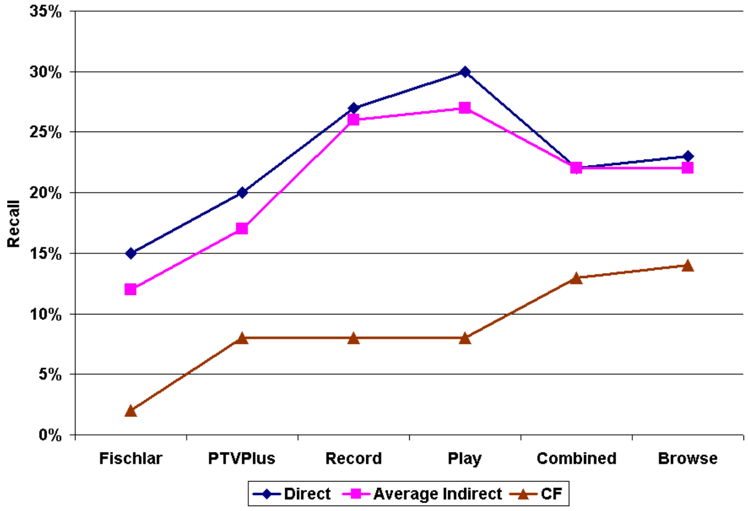


Figure 14. A summary of recall values for each technique across all datasets.

explicit and implicit (behavioural) profiles during the training and testing parts of the evaluation.

For example, Figure 9 presented the accuracy results when using explicit Físchlár profiles as the basis for recommendation and also as the basis for testing. In Figure 16, the recall results are displayed for

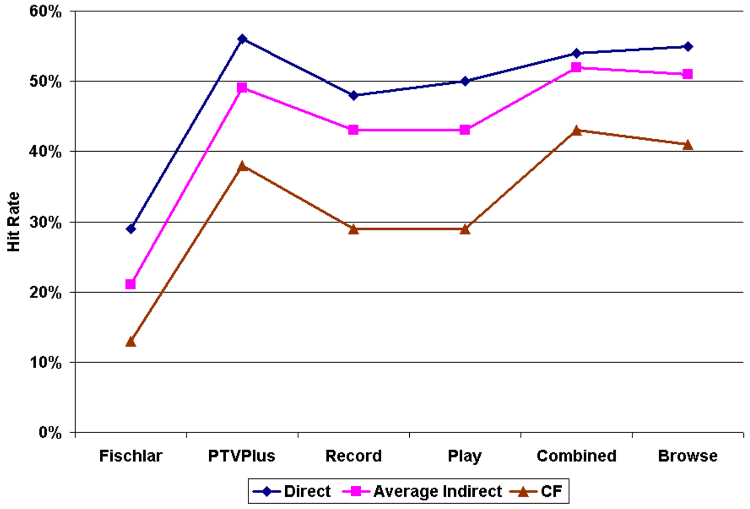


Figure 15. A summary of hit rate values for each technique across all datasets.

recommender systems that are trained using explicit Físchlár profiles, but tested using implicit (behavioural) profiles.

The results are interesting. They show, for example, that even though the recommender is generating recommendations from explicit, ratings-based profiles, recall is greater when the test profiles are from an implicit dataset. For example, we see that the recall of DR, using Físchlár training profiles, but tested with respect to Físchlár Play profiles, is about 26%. Compare this to a 15% recall value when the test profiles are explicit ratings-based profiles (see Figure 9). The same result is found across all of the behavioural profiles, indicating that, in general, explicit ratings-based profiles are just as capable of predicting real behaviours (recording, playback and browsing) as they are at predicting future ratings. Of course it is this prediction of real behaviours that is ultimately the most important consideration in a system like Físchlár; after all, it is more important for Físchlár to recommend programmes that a user will ultimately record, play, or browse than programmes they will simply rate positively.

When we compare the recall results in Figure 16 to corresponding recall results in Figures 10, 11, 12, 13, we find only marginal differences. For example, as mentioned above the recall of DR, using explicit training profiles, but tested with Play profiles, is 26%. In Figure 10 we see that the same DR technique achieves a precision value of 30% when it uses Play profiles for training and testing. In other words, the explicit ratings-based profiles are almost as good predictors of real behaviour as the behavioural profiles themselves.

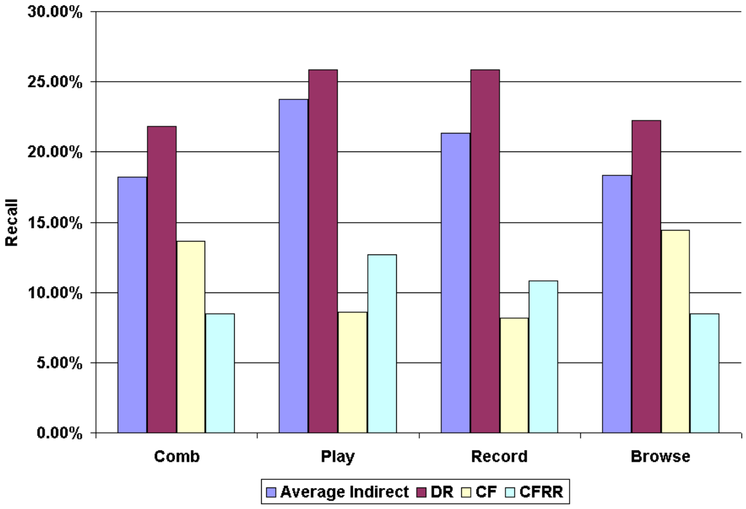


Figure 16. Comparison of recall results for recommenders trained on explicit, ratings-based Físchlár profiles, but tested on implicit, behavioural profiles.

6. Discussion

The techniques developed in this article and their application in the Digital TV space have highlighted a number of important issues, not just for personalized EPGs but for recommender systems in general.

6.1. Explicit vs Implicit Feedback - Ratings vs Behaviour

One of the most important observations to note from the recommendation accuracy results concerns the recall and hit rate characteristics of the implicit (behavioural) Físchlár profiles compared to the recall and hit rate characteristics of the explicit Físchlár profiles. Specifically, recall and hit rate are better for implicit profiles than for explicit profiles. In other words, in Físchlár at least, genuine user behaviours such as playing or recording a programme appear to be more accurate indicators of what programmes a user will tend to record or play back in the future. Thus there is less pressure for Físchlár (and other similar systems) to incorporate an explicit feedback mechanism in order to profile user preferences. Instead, tracking the right sort of natural behaviours (record, play, browse) provides a more accurate profile representation.

This result is analogous to related work on the implicit profiling of Web users in more traditional Web applications [27, 6, 25, 23]. For example, work in [23] investigates the use of implicit behavioural data such as read-time and click-data as a basis for profiling. The current results are particularly significant given the degree of success found with the implicit profiles as analogous results from the Web space are often less compelling [27, 6, 25, 23].

So although there is growing evidence that in certain applications the availability of implicit ratings may obviate the need to collect explicit user ratings, due care should be taken before choosing to ignore explicit ratings. Just because it is possible to translate some natural user behaviour, such as a mouse click or a page view, into a rating does not mean that this rating will serve as a reliable interest indicator. Some behaviours are likely to carry more meaning than others and as such are likely to be more reliable indicators of interest. This is particularly true of the type of behaviours available in Físchlár—recording and playback behaviours are likely to be far more indicative of a user's interest in a TV programme than simple clicks might be of a user's interest in say a news article. It would perhaps, be more accurate to compare these record and playback behaviours to behaviours in the online world such as the purchase of a product, a much stronger indicator of interest than a simple request for information.

6.2. Density vs Accuracy

At the beginning of this article we explained that one of our primary motivations in this work is to look for ways to overcome the sparsity problems associated with typical profile spaces. Any set of ratings-based profiles can be translated into a ratings matrix, with each row corresponding to an individual profile and each column corresponding to an individual item, so that user ratings are represented as values with the appropriate cells in the matrix. The sparsity problem occurs because most of the cells in a typical ratings matrix are empty—most users have rated very few items; conversely most items are rated by very few users.

Sarwar et al. [22] suggest measuring the density of such a matrix for a particular dataset as shown in Equation 9; where the number of total entries is calculated by multiplying the number of users by the number of items that have been rated at least once and the number of nonzero entries is the total number of ratings overall in the dataset. Now we can rank-order different datasets according to the degree to which they suffer from the sparsity problem. For example, Table III shows the density values for the different datasets used in this work.

Percentage increases in density relative to the least dense dataset, the Físchlár dataset, are also shown; for example, the PTVPlus dataset has a density of 0.00575, indicating that only about 5 out of 1000 cells in the ratings matrix are filled in; moreover, the PTVPlus dataset is 60% more dense than the Físchlár dataset.

$$Density(Dataset) \; = \; \frac{Number\ of\ nonzero\ entries}{Number\ of\ total\ entries} \tag{9}$$

Table III. Dataset Density.

Dataset	**Density**	**Relative Density**
Físchlár	0.00358	-
PTVPlus	0.00575	60%
Record	0.00809	126%
Play	0.00843	135%
Combined	0.01182	230%
Browse	0.01191	233%

Interestingly there appears to be a strong correlation between the success of our new similarity-based techniques (relative to the standard collaborative technique) and the density of the ratings matrix in a given dataset. For example, in Section 5.2.1 the most pronounced increase in DR recall (relative to CF) is achieved in the Físchlár dataset, which happens to be the least dense dataset. Conversely, the lowest relative increase in recall for DR, compared to CF, is noted in the Browse and Combined datasets, which happen to have the highest density of profile entries. This can be seen more clearly in Figure 17, which graphs dataset density against the increase in DR recall relative to CF recall for the Físchlár explicit and implicit datasets. For example, DR recall is more than 7 times that of CF recall in the explicit Físchlár dataset, with its density of 0.00358. A similar graph can be obtained for the indirect recommendation techniques, as well as for the hit rate metric. Moreover, all of these graphs indicate a strong correlation between dataset density and the increase in recommendation accuracy (recall or hit rate) for our new similarity-based techniques relative to the standard CF method. For example, the correlation between the DR increase for the dataset density is -0.98.

This suggests that datasets with high degrees of sparsity (low density) are likely to offer the greatest opportunity for improvements on collaborative recommendation by exploiting our similarity-based tech-

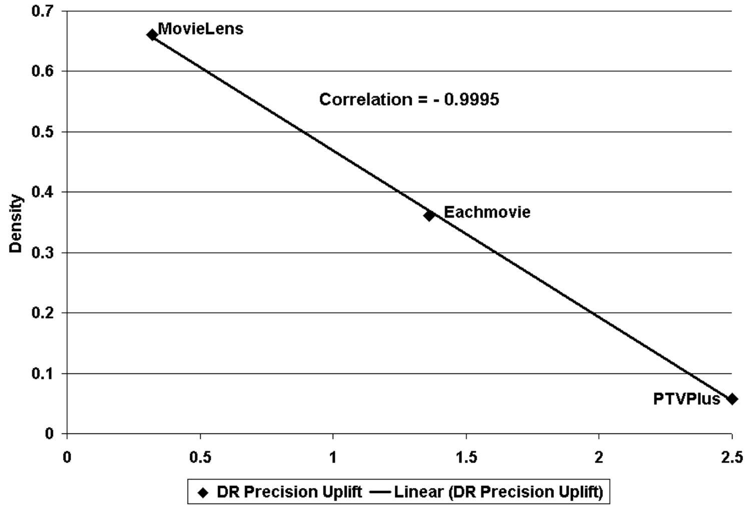


Figure 17. Dataset Density vs Precision Uplift.

niques. Clearly, this particular result is of key importance for recommender systems in general if the relationship holds outside of the TV domain. Of course there is nothing special about the TV related profile datasets used here that would lead us to suspect that this density relationship would not hold in other domains. In fact, in recent work we have demonstrated a similar relationship for movie recommenders [1].

7. Conclusions

The Digital TV domain is a fertile ground for recommender systems research. Digital TV subscribers are faced with an important information overload problem: selecting relevant programming content from a sea of possibilities. DTV subscribers are routinely faced with 10's or even 100's of pages of daily programme listings across up to 200 channels. Moreover, if this problem is not relieved then subscribers are unlikely to benefit fully from the Digital TV revolution, ultimately limiting the uptake and growth of Digital TV services. This genuine business need serves as a strong motivator for the development of personalization solutions that attempt to respond better to the personal preferences of individual users and recommender systems are an important component of these solutions.

In this article we have made a number of important contributions. First, we have presented a novel solution to the sparsity problem that plagues many collaborative recommendation methods. This solution uses data mining techniques to discover similarity knowledge from collaborative filtering (ratings-based) user profiles and uses CBR related methods to exploit this new knowledge during the recommendation process; thus we are using collaborative filtering knowledge to drive content-based recommendation. Second, we have evaluated the effectiveness of this solution in the Digital TV domain in the context of two commercial systems, PTVPlus and Físchlár. We have demonstrated considerable recommendation accuracy improvements over standard collaborative techniques and shown how these improvements are particularly significant in sparse datasets. These results have been validated statistically and are found to be significant at the 99% level. Third, we have compared the use of explicit (ratings-based) and implicit (behavioural) profiles during the recommendation process using data collected from Físchlár. These results indicate that, in Físchlár at least, behavioural information such as a user selecting to record a TV programme, or playback an already recorded programme, is an accurate predictor of the TV preferences of the user, and of future record and playback actions.

As part of our ongoing research agenda we plan to continue to develop and evaluate our recommendation approach in a variety of datasets within, and outside of, the TV domain. We also plan to make comparative evaluations to other techniques, both for making overall recommendations, such as the item-based collaborative approach [22], and for discovering new similarity knowledge, such as Singular Value Decomposition [29] and Latent Semantic Indexing [32].

Finally, in the near future we plan to integrate our recommendation techniques with the Físchlár News system, a specialist version of Físchlár designed around the delivery of personalised news video content. The aim is to develop a personalized news service that compiles a video-based personalized news magazine for individual users based on their recent news history. This system will combine a range of techniques including recommender systems, information filtering, video analysis and topic tracking in order to finely tune the selection of relevant news content for the individual user.

Acknowledgements

The authors would like to thank the anonymous reviewers for constructive suggestions and comments which aided in improving the article, as well as the Informatics Research Initiative of Enterprise Ireland for their support.

References

1. Derry O' Sullivan, David Wilson and Barry Smyth: 2002, 'Using Collaborative Filtering Data in Case-Based Recommendation', *Proceedings of the 15th International FLAIRS Conference*, pp. 121–128.
2. Derry O' Sullivan, David Wilson and Barry Smyth: 2002, 'Improving Case-Based Recommendation: A Collaborative Filtering Approach', *Proceedings of the 6th European Conference on Case Based Reasoning*, pp. 278–291.
3. Hyowon Lee, Alan F. Smeaton, Colin O' Toole, Noel E. Murphy, Sean Marlow and Noel O' Connor: 2000, 'The Fischlar Digital Video Recording, Analysis and Browsing System', *Proceedings of RIAO'2000: Content-Based Multimedia Information Access*, pp. 1390–1399.
4. Alan F. Smeaton, Noel Murphy, Noel E. O' Connor, Sean Marlow, Hyowon Lee, Kieran McDonald, Paul Browne and Jiamin Ye: 2001, 'The Físchlár Digital Video System: A Digital Library Of Broadcast TV Programmes', *Proceedings of the First ACM/IEEE-CS Joint Conference on Digital Libraries*, pp. 312–313.
5. Jon L. Herlocker, Joseph A. Konstan, Al Borchers and John Riedl: 1999, 'An Algorithmic Framework for Performing Collaborative Filtering', *Proceedings of the 1999 Conference on Research and Development in Information Retrieval: SIGIR-99*, pp. 121–128.
6. Joseph A. Konstan, Brad N. Miller, David Maltz, Jon L. Herlocker, Lee R. Gordon and John Riedl: 1997, 'GroupLens: Applying Collaborative Filtering to Usenet News', *Communications of the ACM*, **Vol. no. 40(3)**, pp. 77–87.
7. Upendra Shardanand and Patti Maes: 1995, 'Social Information Filtering: Algorithms for Automating "Word of Mouth"', *Proceedings of ACM CHI'95 Conference on Human Factors in Computing Systems*, pp. 210–217.
8. Barry Smyth and Paul Cotter: 2001, 'Personalized Electronic Programme Guides', *Artificial Intelligence Magazine*, **Vol. no. 21(2)**, pp. 210–217.
9. Barry Smyth and Paul Cotter: 1999, 'Surfing the Digital Wave: Generating Personalised Television Guides using Collaborative, Case-Based Recommendation' *In Proceedings of the 3rd International Conference on Case-based Reasoning.*
10. Paul Cotter and Barry Smyth: 2000, 'PTV: Intelligent Personalised TV Guides' *Proceedings of the 12th Innovative Applications of Artificial Intelligence (IAAI-2000) Conference.*
11. Nathaniel Good, J. Ben Schafer, Joseph A. Konstan, Al Borchers, Badrul M. Sarwar, Jonathan L. Herlocker and John Riedl: 1999, 'Combining Collaborative Filtering with Personal Agents for Better Recommendations', *Proceedings of the 1999 Conference of the American Association of Artifical Intelligence (AAAI-99)*, pp. 439–446.
12. Ian Soboroff and Charles Nicholas: 1999, 'Combining Content and Collaboration in Text Filtering', *Proceedings of the IJCAI-99 Workshop on Machine Learning for Information Filtering.*
13. Marko Balabanović and Yoav Shoham: 1997, 'Fab: Content-based, Collaborative Recommendation', *Communications of the ACM*, **Vol. no. 40(3)**, pp. 66–72.

14. Weiyang Lin, Sergio A. Alvarez and Carolina Ruiz: 2000, 'Collaborative Recommendation via Adaptive Association Rule Mining',*International Workshop on Web Mining for E-Commerce (WEBKDD'2000).*
15. Colin O' Toole, Alan F. Smeaton, Noel Murphy and Sean Marlow: 1999 'Evaluation of Automatic Shot Boundary Detection on a Large Video Test Suite.' *CIR'99 - The Challenge of Image Retrieval: 2nd UK Conference on Image Retrieval. Newcastle, UK.*
16. Shih-Fu Chang, Atul Puri, Thomas Sikora, and Hong-Jiang Zhang: 2001, 'Special Issue on MPEG-7' *IEEE Transactions on Circuits and Systems for Video Technology. June 2001*, **Vol. no. 11(3)**.
17. Hyowon Lee and Alan Smeaton: 2002, 'Designing the User-Interface for the Físchlár Digital Video Library', *Special Issue on Interactivity in Digital Libraries. Journal of Digital Information*, **Vol. no. 2(4)**.
18. Barry Smyth and Elizabeth McKenna: 1998, 'A Portrait of Case Competence: Modelling the Competence of Case-Based Reasoning Systems', *Proceedings of EWCBR-98*, pp. 208–220.
19. David C. Wilson and David B. Leake: 1998, 'Maintaining Case-Based Reasoners: Dimensions And Directions', *Computational Intelligence*, **Vol. no. 17(2)**.
20. Jochen Hipp, Ulrich Güntzer and Gholamreza Nakhaeizadeh: 2000, 'Mining Association Rules: Deriving a Superior Algorithm by Analyzing Today's Approaches', *Proceedings of the 4th European Symposium on Principles of Data Mining and Knowledge Discovery.*
21. Rakesh Agrawal, Heikki Mannila, Ramakrishnan Srikant, Hannu Toivonen and A. Inkeri Verkamo: 1996, 'Fast Discovery of Association Rules', *Advances in Knowledge Discovery and Data Mining AAAI Press*, pp. 307–328.
22. Badrul M. Sarwar, George Karypis, Joseph A. Konstan and John Riedl: 2001, 'Item-based Collaborative Filtering Recommender Algorithms', *Proceedings of the Tenth International World Wide Web Conference.*
23. Keith Bradley, Rachael Rafter and Barry Smyth: 2000, 'Case-Based User Profiling for Content Personalisation', *In Proceedings of the International Conference on Adaptive Hypermedia and Adaptive Web-based Systems* (AH2000).
24. Conor Hayes, Pádraig Cunningham and Barry Smyth: 2001, 'A Case-Based View of Automated Collaborative Filtering', *Proceedings of ICCBR-01*, pp. 234–248.
25. Mark Claypool, Phong Le, Makoto Wased and David Brown:2001, 'Implicit Interest Indicators', *Intelligent User Interfaces*, pp. 33–40.
26. Dan R. Greening: 1994, 'Collaborative Filtering for Web Marketing Efforts', *Papers from the AAAI Workshop on Recommender Systems, AAAI Press*, pp. 53–55.
27. Masahiro Morita and Yoichi Shinoda: 1994, 'Information Filtering Based on User Behaviour Analysis and Best Match Text Retrieval', *Proceedings of the 17th ACM Annual International Conference on Research and Development in Information Retrieval*, pp. 272–281.
28. Mikael Sollenborn and Peter Funk: 2002, 'Category-Based Filtering and User Stereotype Cases to Reduce the Latency Problem in Recommender Systems', *Proceedings of the 6th European Conference on Case Based Reasoning*, pp. 395–420.
29. Badrul Sarwar, George Karypis, Joseph Konstan and John Riedl: 2000, 'Application of Dimensionality Reduction in Recommender Systems–a Case Study', *Proceedings of the 2000 ACM Web KDD Workshop.*

30. Andrew I. Schein, Alexandrin Popescul, Lyle H. Ungar: 2002, 'Methods and Metrics for Cold-start Recommendations', *Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2002).*
31. Michael H. Pryor: 1998, 'The Effects of Singular Value Decomposition on Collaborative Filtering', *Dartmouth College Technical Report PCS-TR98-338.*
32. Peter W. Foltz: 1990, 'Using Latent Semantic Indexing for Information Filtering',*In Proceedings of the Conference on Office Information Systems*, pp. 40–47.
33. Mark Rosenstein and Carol Lochbaum: 2000. 'Recommending from Content: Preliminary Results from an E-Commerce Experiment' *In Proceedings of CHI'00: Conference on Human Factors in Computing, The Hague, The Netherlands: ACM.*
34. Ken Goldberg, Theresa Roeder, Dhruv Gupta, and Chris Perkins: 2001, 'Eigentaste: A Constant Time Collaborative Filtering Algorithm', *Information Retrieval Journal*, **Vol. no. 4(2)**, pp. 133–151.
35. Kim Yum: 2002, 'Comparisons of Principal Component Analysis and Singular Value Decomposition Method for Collaborative Filtering', *26th Annual Conference of the Gesellschaft fr Klassifikation (GfKl 2002).*
36. Katsuhiro Honda, Nobukazu Sugiura, Hidetomo Ichihashi and Shoichi Araki: 2001, 'Collaborative Filtering Using Principal Component Analysis and Fuzzy Clustering', *Research and Development, Lecture Notes in Artificial Intelligence 2198, Springer*, pp. 394–402.
37. Daniel Billsus and Michael J. Pazzani: 1998, 'Learning collaborative information filters', *Proceedings of the 15th International Conference on Machine Learning, Morgan Kaufmann, San Francisco, CA*, pp. 46–54.
38. Chumki Basu, Haym Hirsh and William W. Cohen: 1998, 'Recommendation as Classification: Using Social and Content-Based Information in Recommendation', *In Proceedings of the Fifteenth National Conference on Artificial Intelligence*, pp. 714–720.
39. Michael J. Pazzani: 1999, 'A Framework for Collaborative, Content-Based and Demographic Filtering', *Artificial Intelligence Review*, **Vol. no. 13(5-6)**, pp. 393–408.