Mobile Recommender Systems

Francesco Ricci
Faculty of Computer Science
Free University of Bolzano, Italy
fricci@unibz.it

January 11, 2010

Abstract

Mobile phones are becoming a primary platform for information access and when coupled with recommender systems technologies they can become key tools for mobile users both for leisure and business applications. Recommendation techniques can increase the usability of mobile systems providing personalized and more focussed content, hence limiting the negative effects of information overload. In this paper we review the major issues and opportunities that the mobile scenario opens to the application of recommender systems especially in the area of travel and tourism. We overview major techniques that have been proposed in the last years and we illustrate the supported functions. We also illustrate specific computational models that have been proposed for mobile recommender systems and we close the paper by presenting some possible future developments and extension in this area.

1 Introduction

Mobile phones are becoming a primary platform for information access. More and more people use these communication and information access tools, and the functionalities and the challenges provided by these devices are growing [Turban et al., 2008] [Bertelé and Rangone, 2007]. Tourism is surely a primary application area for mobile applications and an incredible number of services are now offered to support the traveller before, during and after the travel [pho, 2009] [Fesenmaier et al., 2006] [Werthner and Ricci, 2003] [Werthner, 2003]. Hence, it is important to understand the capabilities of this channel and the information access behavior of mobile users. Moreover, as the amount of information and online services increases, it becomes more and more difficult for users to find the right information that is needed to complete a particular task (e.g., choosing a movie, or planning a trip). In particular, users of e-commerce web sites often find it difficult to locate their best products and services, due to the overwhelming number of options to consider and the lack of effective system support in making decisions. Recommender systems (RSs) are information filtering and decision support tools aimed at addressing these problems, providing product and service recommendations personalized to the user's needs and preferences at each particular request.

Here, in the contest of information access through mobile devices, the information and choice overload problem becomes even harder, because of the intrinsic obstacles of mobile usage environments. At the same time, the evolution of mobile devices, e.g., personal digital assistants (PDAs) and mobile phones, the ubiquitous availability of wireless communication services (e.g., wireless LAN and GPRS/UMTS) and the development of position detection techniques (e.g., RFID or Wi-Fi beacon-based and GPS) have fostered the development and commercialization of new and sophisticated mobile services, e.g., location-based information services suited to the needs and constraints of mobile users. Because of the development of these technologies and the incredible appeal of mobile devices and services there has been also much research and development work trying to apply recommendation technologies to this market.

But recommendation approaches, which proved to be successful for PC users, cannot be straightforwardly applied for mobile users. On the one hand, mobile RSs have to overcome the obstacles typically present in mobile usage environments: the limitations of mobile devices, the limitations of wireless networks, the impacts from the external environment, and the behavioral characteristics of mobile users. On the other hand, mobile RSs can exploit two peculiar characteristics of mobile information services. The first exclusive property is "*location-awareness*", i.e., the knowledge of the user's physical position at a particular time that can be exploited as an important source of information to adapt the information delivered by the system. The second exclusive property is "*ubiquity*", i.e., the ability to deliver the information and services to mobile users wherever they are, and whenever they need.

In the remaining of this paper we will provide a survey of the current research and development of mobile RSs with a particular focus on services that are potentially useful to travellers. We will first briefly introduce the general notion of a Recommender system (Section 2) and then we will discuss (in Section 3) some factors that have facilitated and motivated the development of mobile RSs: mobile services, wireless communication technologies, mobile application frameworks, and mobile devices. In a third part (Sections 4, 5, and 6), we will provide the principal contribution of this paper, i.e., we will present an overview of the state of the art of mobile RSs, providing a description of the methodologies and techniques that have been developed in this emerging and challenging research field. The material is structured in three main sections: Section 4 is presenting some basic issues related to the design of effective user interfaces and interactions through mobile devices; Section 5 focuses on some special user tasks, such as tourist guidance, or news access, that have motivated a number of mobile applications and specific technologies; finally in Section 6 we will survey three recommendation architectures that arose mainly in the mobile scenario. At the end of this paper, we will elaborate our conclusions and mention some technical difficulties, challenges, and possible extensions for future research and development of mobile RSs. It is worth noting that this topic would deserve much more space than a single paper, and the reader will easily spot topics that should have been addressed and are here omitted. We beg pardon to the reader since now.

2 Recommender Systems

In this section we will provide a very short introduction to the general topic of Recommender Systems. The reader that is familiar with this subject can skip this section.

Recommender Systems (RS) are information search tools that have been recently proposed to cope with the "information overload" problem, i.e., the typical state of a web user, of having too much information to make a decision or remain informed about a topic. In fact, users who are approaching an E-commerce web site (e.g., Amazon) or a content web site (e.g., cnet.com or visitfinland.com) for collecting information about a product or service, or simply a topic (e.g., Lapland) could be overwhelmed by the quantity of the relevant pages and ultimately the information displayed in these web sites. In order to address this problem Recommender Systems have been proposed [Resnick and Varian, 1997]. These are intelligent personalized applications that suggest products or services, or more generally speaking information "items", that best suit the user's needs and preferences, in a given situation and context [Anand and Mobasher, 2005] [Adomavicius and Tuzhilin, 2005] [Burke, 2007] [Goy et al., 2007].

The core computational task of a RS is to predict the subjective evaluation a user will give to an item. This prediction is computed using a number of predictive models that have a common characteristic, i.e., they exploit the evaluations/ratings provided by user(s) for previously viewed or purchased items. Based on the particular prediction technique being employed, recommender systems have been classified into the following four main categories [Burke, 2007]: collaborative-based, content-based, knowledge-based and hybrid.

The simplest collaborative-based systems compute correlations between users; they predict product ratings for the current user based on the ratings provided by other users, who have preferences highly correlated to the current user [Herlocker et al., 1999]. Newer and more sophisticated approaches are based on matrix decomposition techniques, they try to approximate the user-item matrix, i.e., the two-dimensional matrix with entry in position (i, j) equal to the rating provided by user i to item j, as the product of two smaller matrices [Koren, 2008]. Most of the entries in the original matrix are actually unknown and with this factorization a prediction is computed for all the missing values.

Content-based systems use only the preferences of the current user; they predict ratings for an unseen item based on how much its description (content) is similar to items which the user has highly rated in the past [Pazzani and Billsus, 2007]. These approaches are based on information retrieval techniques [Manning, 2008] since the item description is usually a text, and a vector (feature based) representation is derived by identifying the most relevant keywords appearing in the text. But in content-based RSs there is not any equivalent of what is a query for an IR system. In other words, the ranking produced by the system for a user is static and it represents the best (predicted) ordering of the items with respect to the relevance of the items for the user.

Knowledge-based systems use a knowledge structure to make inferences about the user needs and preferences. An important knowledge-based technique that is exploited within recommender systems is case-based reasoning (CBR) [Bridge et al., 2006], which is a prediction technique that retrieves similar (previously-stored) recommendation sessions, or products, from a case-base and reuses the information stored in these cases in order to identify the recommended product set.

Finally, hybrid systems combine two or more techniques in order to gain better performance with fewer limitations of each approach [Burke, 2007]. Many hybrid systems have been applied to travel and tourism applications. For instance [Ricci et al., 2006] illustrates a travel planning recommender system that is case-based, hence is knowledge-based, but also collaborative-based since it recommends travel services that have been evaluated positively by other users with similar travel related preferences. Another example is presented in [Ricci and Nguyen, 2007]; it exploits the "critiquing" methodology, i.e., a conversational hybrid approach where the recommendation is revised iteratively by the system acquiring user feedbacks on the system suggestion (e.g., "the proposed restaurant is fine but a bit too expensive, do you have something cheaper?").

More general information on RSs can be found in [Ricci et al., 2010] [Adomavicius and Tuzhilin, 2005].

3 Mobile Computing and Mobile Services

3.1 Mobility

In this section we will review some general facts and definitions about mobile services. Mobile information systems, and in particular recommender systems, can be characterized by positioning them along three fundamental dimensions: user mobility, device portability and wireless connectivity [Shiller, 2003].

- **User mobility** refers to the fact that the user can access a mobile information system in different locations. For instance a traveller, landed at Vienna airport, can use a kiosk to access Tripadvisor.com recommender system, for deciding where to book a room in Vienna. In this example, the user is mobile, he can access the same hotel recommender system everywhere, for instance, before or during the travel, and by means of several kinds of devices, e.g., through a kiosk, or with his laptop. A true mobile information system should be designed in such a way that this user mobility is supported and exploited, i.e., by assigning to him a unique logical application session, so even if the system is accessed with different devices and in different usage contexts the user will get coherent replies that take into account the previous user/system interactions. Moreover, the RS should also adapt its recommendations to the context of the user, hence, for instance, suggesting some good restaurant when the user is arriving to the selected hotel.
- **Device portability** refers to the fact that the device used to access the information system is mobile. For instance our user may access the hotel recommender system with a laptop or a smart phone, or a PDA (Personal Digital Assistant). These portable devices can move together with the user and can deliver the recommendations by connecting to other devices, e.g., a host running a web portal with a RS. Alternatively, the mobile device can host a stand alone recommendation application, e.g., a tourist guide sold on a memory card, and installed on the device. Device portability is the dimension that has been mainly studied in previous researches, because of the impact of the physical characteristics of the mobile devices, i.e., limited screen size, limited computation power and data storage, on the human/computer interaction.
- **Wireless connectivity** refers to the fact that the device used to access the recommender system is networked by means of a wireless technology such as Wifi, or UMTS, or Bluetooth. The network is used to access some components of the recommender system that are not residing on the device, and without the need of any wire. Actually this is probably the most important technological development, which really caused the mobile revolution, but it has not been extensively exploited in the recommender system literature. In fact a wireless connectivity, in addition to adding the convenience of providing the standard networking services without the need of a wire, can open new peer to peer connection scenarios via ad hoc networking that have not been extensively exploited yet [Shiller, 2003] (see also Section 6.2) .

The above mentioned dimensions are independent as there are examples for all the possible combinations of these dimensions [Shiller, 2003], and each one has its own impact on system usage and functionality. The complete scenario, exploiting all the three, is depicted by a mobile user accessing the system with a mobile device with wireless connectivity, e.g., a UMTS phone with Internet connectivity. But other scenarios are possible and interesting, for instance a mobile user accessing a fixed device such as a kiosk, or a user accessing a recommender system with a phone while staying at home, i.e., supporting a limited form of mobility, inside the house. A related classification of the different forms of mobility has been proposed by [Pernici, 2006]. The author classifies mobility in three categories: fixed, where no mobility is involved; nomadic, referring the situation where users access a system from different places with different devices, but the location does not change during the interaction; and mobile, where users are also in movement while using a service. In some sense, these are sub-conditions of the "user mobility" dimension, as it was defined above.

Mobile RSs development has been driven but also constrained by these three dimensions: new mobile devices, with wireless communication capabilities, have been considered for supporting the recommendation needs of mobile users. We will discuss the impact of these dimensions on the research and development of RSs in the next sections. We anticipate here that the information system could be aware (or not) of these dimensions and therefore could be adapted accordingly (or not). For instance the above mentioned hotel recommender system could (erroneously) deliver always the same hotel recommendations, in the same way, to a mobile user before and during the travel, i.e., just considering the explicitly stated user preferences. Or, using a better approach, it could exploit the additional knowledge about the user context to further personalize the service. For instance, the current time and location of the user could be used to decide the number of recommended items and their spatial position. Or furthermore, the user interface could be adapted to the device, i.e., in particular its screen size, and to the connection technology limitations, to improve the usability and quality of the delivered service.

3.2 Mobile Services

M-Commerce or M-Business has been classically defined as "any business conducted over wireless telecommunications network" [Turban et al., 2008]. This definition stresses the impact of the "wireless communication" dimension that we mentioned above. In this paper we want to address the pragmatic aspects of the mobility scenario and consider technology as a tool for implementing useful services. Therefore we would simply define Mobile Services as information services for mobile users, hence emphasizing the mobility dimension rather than the wireless one.

There are several drivers for the development and diffusion of mobile services. The wide availability and convenience of the wireless communication technologies and wireless devices is just one motivation. According to Reuters, in June 2007, the total number of cellular connections in the world reached 3.25 billion. This means that half of the world population has a mobile phone, whereas the worldwide number of PC in use is estimated to be around 1 billion. Wireless communication technologies are improving in terms of price/performance and quality (e.g. bandwidth). Consider for instance the wide diffusion of free Wi-Fi connectivity in several European cities, or the availability of flat rates data communication services over 3G networks. These technologies support effective bandwidth of some megabits, so for instance making possible and convenient streaming audio and video.

In fact, the mobile communication network operators and the manufacturers of mobile phones are pushing (advertising) new technologies, products and services to the customers [Bulander et al., 2005, Vatanparast, 2007]. This push is particularly effective on the youth; they represent a key target market for this industry. Their understanding and utilization of mobile technology and the Internet is fundamentally different from that of their parents, e.g., with respect to social and privacy issues related to mobile services [Choi and Choi, 2007]. Meeting their demands will be a key for the success for many companies over the next years.

Another important factor determining the diffusion of mobile services is the increased number of workers that are mobile most of their time. Salespersons are typical and well known examples of mobile workers, but we can also mention: field service workers, such as, journalists, care givers, equipment maintenance operators, or those dealing with transportation and delivery of goods. All these workers are exploiting several kinds of mobile services. Mobile services have been therefore developed in several application areas: sales force automation; field force automation; warehouse and stock management; asset management; wireless operations; fleet management; customer relationships; mobile and wireless office; machine-to-machine [Bertelé and Rangone, 2007]. We observe that most of these application areas have not been addressed by RSs yet. RSs have mostly been applied to customer relationship and in particular to mobile shopping, advertising and content provisioning [Turban et al., 2008]. An increasing number of online vendors allow customers to book or shop with mobile devices such as mobile phones. For instance, users can bid on Ebay using their mobile phones or buy a book using a mobile version of Amazon. Consumers in many cities can pay the taxi or the parking using their mobile phones. Mobile portals, i.e., the equivalent of standard web portals (such as Yahoo, as in Figure 3.2 or MSN) adapted for access through mobile phones, are normally ran by the mobile communication network operators, and provide a huge selection of information services and content of various types and on various subjects.

3.3 Mobile Technologies

In this section we will very briefly mention the most important mobile communication technologies that made possible the extraordinary development of mobile services. We refer the reader to [Shiller, 2003] for a comprehensive description of these technologies. For the aim of this survey, the two most important technology areas are wireless communications and application frameworks for mobile services.

Wireless communication technologies can be classified according to the type of supported networks. PAN (Personal Area Network) is a computer network (CN) used for communication among computer devices (including telephones and personal digital assistants) close to one person, in the range of some meters. The leading technologies for such networks are: IrDA and Bluetooth. LAN (Local Area Network) is a CN covering a small geographic area, like a home, an office, or a group of buildings. The leading communication technology for these networks is Wi-Fi. MAN (Metropolitan Area Networks) are large CNs usually spanning a city, and the typical technology used in these CNs is

Figure 1: Yahoo mobile portal.

WiMAX. Finally WAN (Wide Area Network) is a CN that covers a broad area, e.g., cross metropolitan, regional, or national boundaries, and it is typically implemented with: UMTS, HSDPA, EDGE, GPRS, GSM.

With respect to the application frameworks, the leading solutions for developing mobile services are: WAP, J2ME, Windows CE .NET with Embedded Visual C++ 4.0, and now iPhone and Android SDK. Mobile Internet requires special application frameworks because the main Internet protocol, i.e., HTTP (Hypertext Transfer Protocol) and the HTML language (Hypertext Markup Language) have not been designed for mobile applications and mobile devices. In fact, a single http request may fire several other requests: static and dynamic content may be requested, the interaction with servers via forms, content transformation, automatic loading and reloading, redirecting. These operations may not be easily supported by a wireless communication network with limited bandwidth and without TCP/IP support. For these reasons, the WAP (Wireless Application Protocol) was designed to deliver Internet content, such as standard HTML pages, and enhanced services to mobile devices and users (mobile phones, PDAs): independently from wireless network standards, and with protocols open for everyone to participate. The protocol specifications have been proposed and standardized initially by the WAP Forum (www.wapforum.org), co-founded by Ericsson, Motorola, Nokia, Unwired Planet, and now by Open Mobile Alliance (www.openmobilealliance.org). WAP protocol was designed to scale well beyond current transport media and device types and to be applicable to future developments. The current version of WAP is 2.0, and it uses XHTML MP (XHTML Mobile Profile) as markup language. XHTML MP is a subset of XHTML (XHTML Basic + specific add-ons), and in addition to the older WAP 1.2 stack, offers a stack based on common Internet stack including TCP, TLS and HTTP (with Wireless Profile). WAP was initially a real "failure", a very small minority of the users with WAP-enabled phones were using it. The problem was not actually WAP itself but the cost and convenience of mobile internet. The high cost of the pay per byte model and the limited power and small display of the first phones killed the service. Nowadays, modern phones, such as the iPhone, or Android Hero, or Nokia N97, and the large diffusion of rather inexpensive and flat-rate subscriptions have revitalized WAP. More and more users access the web with these phones and this is surely now the application model supported by the largest number of devices

J2ME stands for "Java 2 Micro Edition" and it is basically a customized version of the Java programming language targeted at devices that have limited processing power and storage capabilities, and intermittent or fairly low-bandwidth network connections. These include mobile phones, pagers, wireless devices and set-top boxes among others. J2ME support the development and deployment on mobile devices of MIDlets. These are applications running on the devices as applets are Java applications running on the standard Web browser. MIDlets run in a protected sandbox - the KVM - but unlike applets, they much more limited, e.g., a reduced set of graphical widgets are available. Nevertheless, MIDLets enable the development of applications that can just occasionally use the network (when needed) and can execute most of their functionality without being connected. For instance a MIDLet museum guide can store locally (in the mobile device) various information on the exhibits and present them to the user in the right context. J2ME applications run on several operating systems but not easily on Window CE platforms and Android, and not at all on iPhone. In these environments similar, but proprietary, application framework (e.g., .NET) are provided, enabling the development of complex and rich connected applications.

3.4 Mobile Devices

Nowadays there are plenty of mobile devices, and many new types are introduced in the market every day. It is not easy to make a classification or simply to illustrate the main classes of devices. It is out of scope to provide here a full description of this subject but we want to make some examples of devices that have been used in developing advanced information search application sand RSs. The reader is referred to [Gansemer et al., 2007] for an extensive description and classification of mobile devices. We will focus here on three classes of devices: sensor and radio frequency identification devices; mobile phones and personal digital assistant; and laptop computers. We will briefly describe just the first two types of devices as laptop computers does not deserve any description. It is only important to note that all these distinctions are rather fuzzy. For instance small and cheap laptops weighting less than one kilo and with screen size of 8/10 inches are now rather popular. Or even laptop computers without a keyboard with small touch screen (7 inches) represent transition points between laptops and personal digital assistants.

- **Sensor and Radio Frequency Identification** These are very simple wireless devices devoted to accomplish a few simple functions [Shiller, 2003]. Their exploitation in RSs is still at an early stage, but there is a good potential for their application. Sensors can detect the room temperature and humidity and transmit them to another mobile device (e.g., a laptop or a meteo station) that will collect and process the data and adapt, for instance, the sports activity recommendations accordingly. Sensors can detect various human body biometric data that can be exploited to adapt the recommender output. For instance, heart rate or skin conductance sensors can measure the arousal parameter while brain scanners could probe multiple emotional dimensions. RFID (Radio Frequency Identification) technology enables objects or humans identification from distance [Want, 2006]. Unlike earlier bar-code technology it does not require line of sight (LOS). RFID tags now support a large set of unique IDs and can incorporate additional data (e.g., manufacturer, product type). An RFID reader can be connected to a PDA or a mobile phone and can detect many different tags located in the same general area [Cena et al., 2008]. RFID technologies are more and more used as convenient localization techniques inside buildings [Tesoriero et al., 2008] [Barrat et al., 2008] [Lamber et al., 2009]. RFID tags can now be manufactured at low prices and therefore can be widely used to tag even inexpensive items.
- **Mobile phones and personal digital assistant** This is the type of devices that received most of the attention in the recommender systems community, and many applications running on these devices will be illustrated later on. Current mobile phones exploit 3G communication technologies (UMTS) and are offering to users a wide range of advanced services: Wireless Application Protocol (WAP2.0) for content browsing, Multimedia Messaging (MMS), large bandwidth Internet connection, etc. Top of the line phones such as Nokia N97 or Apple iPhone, weight approximately 100 grams, offer advanced web browsing support and sophisticated touch-screen interfaces with 640x360 (3.5 inches) and 320x480 (3.5 inches) resolution respectively. These phones can take and display photos and videos, play music downloaded from Internet at several megabits per second, route the user using GPS data and maps, store data in several Gigabytes of internal memory, send email, browse the web, connect to other devices (such a laptop or a TV) via Bluetooth, or Wi-Fi, or USB, or a video cable. They run an operating system (Symbian OS and iPhone OS respectively), enabling them to execute several kinds of applications provided with the phone and they enable the user to install and run other applications, e.g., those based on Java Micro Edition (see section 3.3). In the title of this paragraph we mentioned personal digital assistant devices, since many earlier mobile RSs were developed for these types of handset. But nowadays their market has almost completely disappeared due to the enriched form and functionality of mobile phones, so now does not make sense to distinguish the two anymore.

The importance of the device development for the penetration of mobile services is unquestionable. For instance Net Applications in their monthly survey for January 2009 states that the iPhone accounts for 0.48% of all Internet traffic (http://marketshare.hitslink.com/os-market-share.aspx?qprid=9). This is a remarkable result given that Windows Mobile, Googles G1, Symbian and BlackBerry all together account for 0.45% of Internet traffic. It is clear that this success is due to the particular design and convenience of the device.

It is unclear what will be the evolution of the mobile devices in the next years. The current trend is for computers more and more integrated, small, cheap, portable, and replaceable. The technology is projected to go in the background and the computer will be aware of its environment and will adapt to it ("location awareness"). Advances in technology are providing: more computing power in smaller devices; flat lightweight displays with low power consumption (e.g., digital ink); new user interfaces; more bandwidth per cubic meter; multiple wireless interfaces (wireless LANs, wireless WANs, regional wireless telecommunication networks).

4 User Interfaces for Mobile Recommender Systems

In this section we will illustrate the major problems for RSs that are arising from the particular characteristics of the mobile devices and in particular those related to the limitations of the user interface. We will illustrate some of the technical solutions that have been developed to tackle with these issues. In this section we will focus essentially on general usability issues and their solution when porting standard recommendation techniques to the mobile scenario. In section 5 we will focus on new tasks and functions, originating in the mobile scenario, and the user interfaces that have been developed to support these tasks.

4.1 General Issues

In this section we highlight some of the issues that must be considered when designing RSs for mobile users. Most of these issues refer to mobile devices in the class of mobile phones and PDAs (see section 3.4).

First of all, recommendation sessions on small screen devices can be difficult and frustrating for end-users. It is known that users can actually read and understand information offered by small interfaces, but the size of the display can impact on users performance [Jones and Marsden, 2005]. For example, on a small screen the user may be forced to carry out extensive scrolling while browsing a web page, and the more a user has to scroll down, the smaller the chances of an item being clicked. In addition, a user on a small screen is less effective in completing an assigned task when compared to users of large screens.

In addition to this, mobile devices offer limited input and interaction capabilities. Most existing mobile phones incorporate only a standard 12-key numeric keypad thus making quite complicated any text-input, even if predictive text techniques as T9 can help. More advanced, expensive, and bigger devices include a miniature QWERTY keyboard, but these devices are not (yet) very popular and are mostly used by business users. Mobile devices have a small number of control keys assisting users with navigation and scrolling tasks. The simpler devices include only two softkeys which are located adjacent to the screen (below) and have variable functions. The more advanced provide a navigation joystick which typically provide 4-direction movement and scrolling support. As we mentioned above more recent mobile phones such as the Apple iPhone and Nokia N97 support more sophisticated input and interactions capabilities via a touch-screen interface that reacts to user gestures.

Browsing the Mobile Internet poses serious problems because of the above mentioned issues, and for instance, whereas PC users can surf the Web for hours, mobile phone users have much shorter browsing sessions, i.e., in the range of some minutes. Moreover, in the Mobile Internet there is a lack of a (de facto) standardization of the browsing tools. In fact, Internet Explorer and Firefox currently dominate the PC Web browsing market, but, in contrast, the browsers provided in different operating systems environments could be rather diverse. So, to match mobile phone capabilities, web servers need to identify the browser type and adapt the content to the requesting client [Laakko and Hiltunen, 2005, Reynolds, 2008]. Another important issue is related to the cost of the interaction. Nowadays we access the Web with ADSL connections with flat rates. These same connections could be used by advanced mobile devices equipped with Wi-Fi cards, but when we are really moving outside our home or office, we need a fast 3G data connection that is becoming reasonably priced (and flat rate) only recently.

Because of the above limitations, not only browsing, but especially information search, and in particular item recommendations, is problematic on mobile devices. Entering queries, e.g., based on keywords or on preferred product attribute values, is too time consuming and complex. Moreover it is quite difficult for users to process the result lists returned. [Church et al., 2007, Church et al., 2008] analyzed the key differences between browsing and search behavior on the mobile internet compared to the Web. They show that browsing continues to dominate mobile information access, but they also indicate that search is becoming an increasingly popular alternative to information access, especially in relation to certain types of mobile handsets and information needs.

For these reasons, RSs interfaces for mobile devices have basically tried to support better the browsing and direct manipulation approach to information access, rather than the search model, much more popular on the PC. Notwithstanding that, increasing usability of mobile devices is likely to push more and more users to exploit the search model.

4.2 Intelligent User Interfaces Approaches

In this section we will illustrate some techniques that have been exploited in mobile RSs to address the issues mentioned in the previous section.

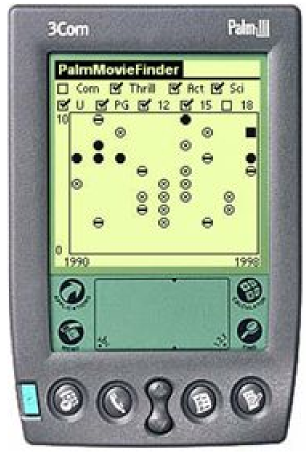

Figure 2: Starfield visualization on a PDA

4.2.1 Starfield Displays

Starfield displays are two-dimensional scatterplots used to structure result sets enabling zooming to reduce clutter [Dunlop et al., 2004, Schneiderman, 1994]. A starfield display is basically a mapping of selected attributes of a multidimensional information space (i.e., a collection of objects represented in a multidimensional space), onto a two-dimensional representation. X and Y coordinates are used to plot attribute values in such a way that an overview of a large information space can be obtained and used to explore specific areas of interest for the user. A key goal of starfield displays is to take advantage of the human visual information processing capabilities when presented with interfaces that combine starfield displays and dynamic query filters. Figure 2 shows an example of a starfield display on a small palmtop mobile device [Dunlop et al., 2004]. It shows a Movie Finder system with different movie categories such as (comedy, thriller, action or sci-fi), categorized by rating certificate (using the UK scheme: U, PG, 12, 15, 18) and the movies' year of release and current popularity rating were used as X- and Y-axes. The figure shows a sample screen with all non-18 certificate non-comedies selected.

Starfield displays have been proved to be suitable and successful access methods for mobile device information systems. By adjusting buttons, check boxes and other control widgets on starfield displays, users feel being more in control of the system than by using traditional approaches with menus [Schneiderman, 1994].

4.2.2 Displaying Similar Searches

[Church and Smyth, 2008] proposes a novel interface to support multi-dimensional, context-sensitive mobile search, combining context features such as location, time, and community preferences to offer a unique search experience that is well-adapted to the needs of mobile users. The system integrates the user context, in the form of temporal and location data, with preference information derived from the queries of mobile searchers with similar interests and presents a view of the evolving search activities of this community of users (see Figure 3. In this way, a user can browse through community search experiences and manipulate the searches of others, learn from these searches, and initiate his own. The main idea is that instead of recommending information explicitly requested by the user the system becomes proactive by presenting information, i.e., searches performed by other users in similar contexts. This will support a more exploratory approach to information search [Marchionini, 2006] [Marchionini and White, 2009].

When the user enters the application a map centered at his current location is shown. The map shows all recent queries and result-selections submitted by other users in that specific location. The user can manipulate the display with two sliders. The first slider represents the time and the user can adjust this to display queries submitted at different times (e.g., now or earlier). The second slider models how close the queries shown on the map, which were made by like-minded users, are to the past queries of the user. The default value for this slider is broad. Again users can manipulate it to adjust the set of prime queries shown on the interface. The authors use a simple measure of query overlap (also known as query similarity) from [Balfe and Smyth, 2005]. Manipulating the preference slider modifies the similarity threshold and thus relaxes or tightens the community aspect of the interface. It is clear the similarity of this approach with the Starfield displays illustrated earlier.

Figure 3: Mobile searches by a community of users

4.2.3 Critique- and Preference-based Interfaces

Critiquing is a powerful recommendation technique where the user is required to express her preferences by criticizing items that the system recommended [Burke, 2002] [McGinty and Smyth, 2006] [Chen and Pu, 2009]. This is in contrast with standard preference elicitation techniques that ask upfront for user preferences. These are entered as preferred values for product attributes, and then are used to build a query and retrieve some recommendations. In critiquing, the user, instead of being required to formulate a precise search query at the beginning of the interaction, is involved in a dialogue where the system's product suggestions interleave with the user's critiques to the recommended products. A user's critique is a comment (judgment) on one of the displayed products. A critique points out an unsatisfied preference (e.g., "I would like a cheaper restaurant"), or confirms the importance of a product feature for the user ("I'd like to have dinner in a restaurant with a garden terrace"). This critique-based user preferences elicitation procedure enables the system building a better understanding the user's needs and preferences, and hence ultimately, in constantly improving the recommendations [Chen and Pu, 2009].

In [Ricci and Nguyen, 2007] it is shown that critique-based recommendation methodology is particularly effective in supporting mobile users in product selection decisions. When making a critique in that system (MobyRek), the user is also supported by the GUI on the mobile device to express the strength of the preference implied in that critique, i.e., as a must or a wish condition. This helps the system to correctly exploit the user's critique, i.e., whether to focus on a certain part of the product repository (by using the must conditions) or to refine the products' ranking (by using the wish conditions).

Eliciting user preferences through critiques has some general advantages, and some that are particularly suited to the mobile scenario. Firstly, the preferences are explicitly stated by the user, and hence, are more reliable than those implicitly collected, for instance, by mining the user's interaction behavior, or those expressed on the whole item (as in Collaborative Filtering ratings). Secondly, the user effort to make a critique is low, as compared to other methods using interviews or early ratings. In practice, a critique to a recommended product is done simply by a few button clicks (see Figure 4). Finally, compared to the explicit request to specify preferences, the request to criticize a real product is more convincing because the system first provides some immediate benefit to the user by showing some recommendations, and then requests and motivates the user to reveal additional preferences in the form of critiques.

Preference-based is an approach similar to critiquing. In preference-based the user feedbacks are expressed on the whole item rather than on single attributes of the item [Burigat et al., 2005] [McGinty and Smyth, 2006]. [Kim et al., 2004] presents a preference-based system called *Viscors*. It recommends wallpaper images to mobile users. In fact, that system uses a hybrid approach, combining collaborative and content-based filtering. Here we focus on the content-based method that requires a specific user interface to enter the search keywords and to express the ratings on presented images. The Collaborative Filtering (CF) module produces the initial list of recommended images. Then, the customer selects an image and after viewing the image, the customer might decide to purchase it, or use it as a query for a content-based search of similar images, or go back to the CF generated recommendation

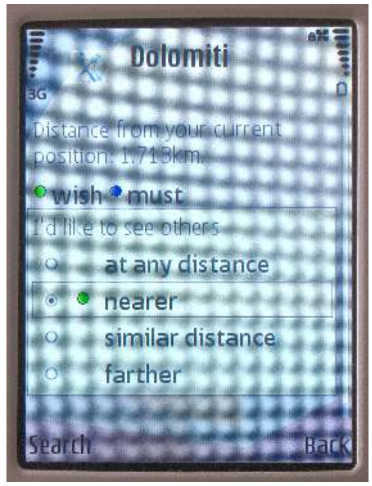

Figure 4: A critique in MobyRek

list. If the customer decides to use the viewed image as a query, then the Content Based (CB) module computes the distances from the query to all the other images in the database, and generates the list of the most similar images. The user can then express her preferences for the displayed images, declaring which one she prefers. After this step the CB module updates the preference information using the preferred set and applies this information to refine the query and to update the distance function. It then uses the refined query and updated distance function to compute a new set of recommendations. The system is designed to run on mobile phones. The authors have shown that this method can increase significantly the efficiency of the recommendation, measured as a decrease of the number of page views per single purchase, compared to a pure collaborative filtering approach.

4.2.4 Query Rewriting

The idea of helping the user to rewrite a failing query to a recommender system has been largely used in PC based recommender systems [Mirzadeh et al., 2005] [Jannach, 2006] [McSherry, 2004] [Bridge et al., 2006] [Mirzadeh and Ricci, 2007]. In these approaches, when a user query is over-constrained and no item in the data base satisfies the query conditions, then one or more relaxed queries are offered to the user. In these new queries some constraints present in the original query are either removed or just relaxed.

In mobile recommender systems as in [Tung and Soo, 2004] [Ricci and Nguyen, 2007] the relaxed version of the user query is computed automatically by the system to simplify the human-computer interaction. In [Tung and Soo, 2004] the recommender suggests restaurants to the user and it represents a user's query with a set of constraints. The system exploits the user's past selections for repairing a failing query with constraint relaxation. For example, given a failing query of (cost less than 20 USD, serving Chinese food, and non-smoking) the system realizes that the user used to spend 50 USD for his lunch. Hence, it proposes a relaxation increasing the constraint on cost to 30 USD which results in some products. Having the system's relaxation proposals, the user then has full control on which one he might want to follow. The prototype system is designed to run on pocket PCs.

In [Ricci and Nguyen, 2007] the approach is different as the constraints that cannot be attained are removed from the query, which is a conjunction of strict constraints (logical conditions), and are moved to a similarity-based query. So the system first applies the query, based on logical constraints, to find the restaurants that satisfy the must-have conditions which are actually attainable in the given database of restaurants. Then, the system ranks the resulting items using the similarity based query containing others conditions that are not logically attainable, and are derived either from the user direct input or mining past cases. So for instance if there are no restaurants with price in the range of 10-20 Euro, the system will score higher a restaurant with average cost of 22 euro than a restaurant with average cost of 26, as the first is closer to the original request of the user.

4.2.5 Map-based Interfaces

In many mobile recommender systems and mobile search tools (as discussed in Section 4.2.2), maps and map-based interfaces are used as primary access method to visualize the recommended items, e.g., points of interests

Figure 5: MapMobyRek display of the effects of a "must" critique

(restaurants, museums, or hotels), their spatial relations, and various kinds of information related to these points (e.g., menus, opening hours, or in-room services). So, with that respect, map-based interfaces help to address some relevant information access problems in mobile devices.

However, map-based interfaces pose also new problems. In fact, in order to be effective and readable, the display should be kept free from irrelevant information. And this is particularly true in the mobile usage environment. Because of the various limitations of mobile devices, referred to in Section 4.1, displaying on an electronic map a large number of objects and their related information is computationally expensive and usually not effective. Hence, RSs and filtering mechanisms, provide a very good solution and simplify the usage of map-based interfaces for mobile travel services.

There are many applications of mobile recommender systems that have used maps as primary item access method, and some will be illustrated in Section 5. Here we mention [Averjanova et al., 2008], where the authors extend the MobyRek critique-based system [Ricci and Nguyen, 2007] with a map interface. MobyRek employs a text-based interface for recommendation visualization, where the recommendations are presented to the user in a ranked list, as in many recommender systems and in standard search engines like Google. On the contrary, MapMobyRek uses maps as the main user interface for information display and access, adding some new decision-support functions based on the map. For instance, to recognize immediately the differences between good and weak recommendations, colored icons are used and the effect of a new critique is shown as a progressive change of the color or shape of an item. For instance in Figure 5 green icons (with a smiley on the bar at the top of the screen) represent top recommended restaurants and the icons with smaller size are disappearing from the map because the user previously entered a must critique that filters out the corresponding items.

Another notable example of map-based interface for recommendation visualization is presented in [Burigat et al., 2005]. Also this system is designed to recommend POIs (i.e., hotels and restaurants) for city visitors. Here the system builds the user-query representation by asking the user to indicate her constraints on the item (i.e., hotel or restaurant) attributes. However, the system does not employ the constraint-filtering approach or a multi-attribute utility function. Instead, the system constructs the recommendation list by ranking the items according to their satisfaction score. An item's satisfaction score is measured by the number of constraints (indicated in the user's query) that are satisfied by the item. Each recommended item is visualized by an icon superimposed on the map of the geographic area, augmented by a "filled-in" vertical bar representing how much the item satisfies the user's query. The system is designed to run on pocket PCs.

4.2.6 Visualizing query results

Apart from interfaces based on graphical representations (e.g., starfield display or maps) recommendations are typically displayed as a ranked list of information items. The format is very similar to that used by a search engine to display the retrieved hyperlinks. To address the limitations due to the small screen size, several techniques have been proposed to convey as much information as possible on the presented item optimizing the display usage. The typical approach in mobile search consists of using snippet texts, i.e., short descriptions of the hyperlink content. Conversely, recommender systems, which exploit a structured internal representation of the items, display a subset of the item features that are considered as more important [Ricci and Nguyen, 2007].

This issue of how to help users to understand the value of recommended results in a mobile search interface was addressed by [Jones et al., 2004]. In their approach, instead of using standard snippet text approaches, which require the extraction of a block of document text related to the query, they use a set of key phrases, automatically extracted from result pages. The resulting key phrases provide for a more economic use of screen space and are at least as effective and informative as using long result titles.

[Church et al., 2006] presents and evaluates an alternative approach to search result gisting that replaces result snippets with a much shorter text representation made up of the terms of related queries that have led to the selection of a particular result in the past. This approach relies on data collected by a community-based personalized meta-search engine (I-SPY [Smyth et al., 2005]) that records the queries and search results of different communities of users. Hence, if for example, the query "Java" offers as first search result, a hyperlink with anchor "Java Sun Technology", then the associated queries "j2sdk1.5" and "java tutorials", are shown to illustrate the content of the hyperlinked page "Java Sun Technology". These related queries help to inform the user about other contexts in which this result was selected. The authors show that related queries are as informative as snippet texts and offer the potential for a significant space saving.

5 Mobile Recommendation Tasks and Functions

In this section we will focus on new recommendation tasks and the corresponding support functions that have been developed in the mobile scenario. The tasks illustrated are not intended to be exhaustive, but provide a reasonable coverage of those most often considered in current recommender systems.

5.1 Tourist Guides

This is the application area that received the largest attention. The functions supported by tourist guides are related to finding relevant attractions and services, or supporting the exploration of an area.

5.1.1 Finding relevant attractions

The city attractions (e.g., museums, art galleries, churches, etc.) are the recommended items of these recommender systems, and they are visualized either in a traditional list-based interface, or as a set of recommendations on a map [Dunlop et al., 2004], or as an itinerary [Kramer et al., 2006] [Dunlop et al., 2007]. The attractions are computed on the base of session-specific and long-term preferences stored in the user model. The position of the user is often used to personalize the results.

[Cena et al., 2006] present an interesting tourist guide that focusses on intelligent adaptation as key tool for the design. Their system (UbiquiTO), is a tourist guide which integrates different forms of adaptation: to the device type, to the user characteristics and preferences, to the context of interaction (user location and the time of the day). UbiquiTO adapts the content of the provided recommendation, such as the amount/type of information/features associated with each recommendation.

[Dunlop et al., 2007] presents an application for recommending skiing routes (i.e., pistes). The system asks the user to indicate her ski-run preferences (e.g., route difficulty level), and then uses the indicated preferences to computes a list of recommended routes. The system visualizes on the map the recommended routes and their suitability for the user. The system is designed to run on mobile phones.

5.1.2 Finding relevant services

This is a very similar functionality to that mentioned above. Here the user will typically receive information about travel services such as restaurant, hotel, transportation services, information offices, etc. [Burigat et al., 2005] [Park et al., 2007].

The system presented in [Park et al., 2007] is aimed at recommending restaurants. The system computes personalized recommendations using a Bayesian network that models the probabilistic influences of the input parameters (i.e., the user's personal information and contextual information) on the restaurant attribute values. A restaurant is represented by three discrete-valued attributes: class (e.g., Korean or Italian restaurant), price (e.g., low or medium), and mood (e.g., romantic or tidy). The Bayesian network is defined by an expert, and is learned using a training dataset. At the first use of the system, the user is asked to provide some personal information (e.g., age, gender, income, preferred restaurant class, etc.). The user's contextual information is automatically collected (detected) by the system, including season (e.g., spring), time in day (e.g., breakfast), position, weather (e.g., sunny), and temperature (e.g., warm). When a user requests for a restaurant recommendation, the system computes the recommendation score for all the restaurants in the database. A restaurant's recommendation score is a weighted sum of the conditional probabilities of the restaurant's attribute values, where these conditional probabilities are derived (inferred) from the learned Bayesian network. The system visualizes on the map a small set of the restaurants, i.e., those achieving the highest recommendation scores. The system is designed to run on PDAs.

[Dunlop et al., 2004] presents *Taeneb City Guide* a restaurant recommender system for city visitors. The system uses the constraint-based filtering approach to control which restaurants are shown on the map. The user specifies constraints on restaurant cuisine and price. The system retrieves from the database the restaurants that satisfy the user's indicated constraints, and then ranks them according to their match to the preferences stored in the user's profile. The system builds and updates the user's profile, which maintains her long-term preferences, by mining and interpreting the user's actions (e.g., writing a restaurant review, reading a review, viewing a restaurant's details, etc.) and collecting the user's ratings to the restaurants. Though the system ranks the recommended restaurants and shows them on the map, it does not visualize how close each recommended restaurant matches the user's preferences (i.e., the recommendation level). The system is designed to run on Palm devices. Other restaurant recommender systems are presented in [Yap et al., 2005] [Yap et al., 2006] [Yap et al., 2007].

5.1.3 Exploring a city

In this task the recommender system is not driven by queries, i.e., by explicit requests of the user, as in the previous cases. The recommender is more aimed at discovering, even unexpected, attractions or services. An interesting example of these types of applications is George Square system [Brown et al., 2005], running on a handheld tablet PCs. It is designed for supporting visitors to explore a city as well as share their visit experiences. As the user moves in the city and visits the attractions, her positions and activities are automatically recorded (logged) by the system. The user's activities include the attractions she encounters, the web pages (i.e., weblogs created by previous visitors) she browses, and the photographs she takes. To build the recommendation list, the system makes use of patterns of co-occurrence of positions and activities, and employs collaborative filtering technology. In particular, the system uses the user's recent activities (i.e., in the last few minutes) to define the user's current context, then finds in the previous visitors' log data those periods of time with similar contexts, and finally includes in the recommendation list the activities done in those periods of time by these previous visitors. In addition, when showing the recommended items (i.e., attractions, web pages, photographs) on the map, the system displays the positions of nearby visitors and supports their leisure collaboration. For instance, the system lets the user to collaboratively produce pictures of the attractions, or to have (voice-over-IP) discussions on co-interested attractions and objects.

A similar idea of supporting the exploration of a place with context-dependent information services is presented in [Hinze and Buchanan, 2006], a mobile infrastructure for cooperating information services using an event-based communication layer to support continually changing information (TIP). The TIP system prioritizes the display of sights of interest relevant to the user interests expressed in his profile. TIP's information delivery is based on the user's context: her location, a personal profile describing her interest in sight groups and topics, and the user's travel history. The system also considers a sight's context, such as its location and its membership in predefined semantic groups of sights.

5.2 Route Recommendation

Transportation and navigation services offer to mobile users directions relative to their current geographical location using the ability of a network infrastructure (typically GPS, but also Wi-Fi or RFID) to locate the exact position of the mobile user and give him detailed directions to reach a desired destination. Optimal route computation in road networks, i.e., typically the shortest route between two points, is their common function. By default these systems focus on road safety and comfort while offering a good compromise between time and distance. Additionally the user can specify his preferences by selecting what they call "recommendation" of the quickest, the shortest or the most economical path. In these portable, GPS based navigation systems, the user is also supported in following the route by voice indications and annotated maps interfaces.

A relatively new network routing problem is defined when the user is allowed to use several transport means, i.e, not just the car, but also busses, or walking. Such type of networks, when a user can swap the transport means, is called transit network. For instance the cities of Geneva and Rome offer mobile services to support this task. A system developed for the Public Transports of Geneva (TPG) uses a mobile phone connected to a Bluetooth GPS receiver to display a map of the closest bus stops, as well as time tables and information about the next departures and how to reach the chosen bus stop by foot. The mobile service for Rome Atac Mobile provides as well different information on timetables of the public transport (busses, metro), traffic situation, accidents, and it also offers a route finding function. The optimal route is computed given two arbitrary end points as input to the system (street addresses). The output is a detailed textual description of the trip, involving the walking distances, the bus lines to be taken, their arrival frequency, the places where to change the transport mean as well as the estimated travel time.

Although this last system does compute the shortest path in a transit based network, the user involvement in selecting their preferences and hence obtaining personalized suggestions is totally missing: she is allowed to select just the start and the destination points of the trip and the route is not personalized. So, for instance, if the user is a visitor and she is at the station and must go to the Coliseum, and have plenty of time to reach a destination, the system will suggest her the same route that will be given to a citizen that is moving to the destination for work motivations (i.e., using the underground), whereas a good combination of underground and walking would better serve the Rome visitor.

As this example shows, in real life situations users are not always interested in recommendations for the quickest way to travel from point to point, and other aspects could be more important when deciding on a specific trip in a city. For example, the price of the ticket or the fare (e.g., for a taxi) of the transport mean might influence the user's decisions. Moreover especially the visitors of a city are rarely interested in reaching a specific place in the city as soon as possible, and a longer route leading through the most famous POIs of the city might be more convenient, even if, for instance, requiring some more walking.

A recent research project addressed these issues taking into account more diverse user's preferences and needs, and developing a personalized solution [Tumas and Ricci, 2009]. PECITAS (Personalised City Transport Advisory System), is a system prototype implemented for the city of Bolzano, Italy. It uses a knowledge-based approach [Burke, 2007] and provides to the citizens and city guests of Bolzano user recommendations for the best route between two arbitrary points in the city, using several possible transport means, such as bus, taxi and walking. PECITAS generates multiple routes (travel profiles) for a given departure and arrival points using some additional route constraints. The constraints were chosen such that the generated travel profiles would be as diverse as possible, and at least one of these can match the preferences of the user (user model). So for instance fast routes are generated but also routes that pass close to important points of interest or mostly based on walking. The recommendation, i.e., the ranking of the generated routes, is based on the user model. To build the user model PECITAS poses explicit optional questions to the user on four criteria: walking preferences, number of bus changes, arrival time at the destination, sightseeing.

Another route recommender system has been illustrated in [Kramer et al., 2006]. The system here recommend city tours, i.e., itineraries starting and ending in the same point and touching a good selection of points of interest that match the user preferences and that can be visited in the user given amount of time. Given a user's request, the system computes a personalized tour in the city by asking the user her interests and the time she would like to spend for the visit. A tour is composed by a set of POIs (i.e., restaurants, attractions and events). The system then visualizes the recommended tour on the map and provides some audio guide information. The system computes and visualizes just one tour per each user request, not a list of recommended tours. The system is designed to run on pocket PCs.

5.3 Information Recommendation for mobile users

Information recommendation has been a very popular topic for RSs, and especially news and web pages recommendations. RSs in this area have been normally developed with content-based technologies, or hybridization of content-based with collaborative-based approaches [Balabanovic and Shoham, 1997] [Billsus and Pazzani, 2007]. We now illustrate some examples of information and news recommender systems explicitly developed for the mobile scenario.

5.3.1 News Recommendations

In [Billsus et al., 2000] and [Billsus and Pazzani, 2000] the authors present *Daily Learner*, a system running on Palm PDA, that addresses the problem of recommending daily news stories. The user's preferences are represented by a vector of weights, representing the importance of some selected keywords. To address the cold-start problem, the first time the system is used, the user is asked to select one among nine news channels, i.e., some pre-defined categories. Then, during the interactions with the user, the system collects her implicit feedback observing her reading behavior. To produce the recommendations, the system first tries to use a Nearest Neighbor classifier using the user's short-term model. If a story cannot be classified by the Nearest Neighbor classifier, because it cannot find some very similar neighbors, or no neighbors at all, then the system uses a naive Bayesian classifier on the user's long-term model.

A very different model for web page recommendations has been proposed in [Brunato and Battiti, 2003]. The *PILGRIM* system applies data mining techniques and exploit the users position as relevant information to select and rank a small number of links that are probably of interest to the user. Here a middleware layer, the location broker, collects a historic database where user locations and links explored in the past are mined to develop models relating resources to their spatial usage pattern. The models are used to calculate a preference metric when the current user is asking for resources of interest. The system is designed to run on PDAs.

The work described in [Lee and Park, 2007] stresses some important factors for news recommendation in a mobile scenario. First, on the mobile web, news services focus on the distribution of current news (content) rather than on past or related news articles. It is therefore important, when personalizing news, to consider the recency and importance of news articles. Secondly, even though it is possible to learn about preferences for past news articles, it is difficult to apply standard collaborative filtering techniques for new items when no or few users have read them. The third point relates to the social behavior of users. Users can be classified into a user group (segment) by article usage pattern and demographic information. User segments with similar content or similar usage patterns can be used to make recommendations by selecting articles preferred by the segment. Hence, the proposed method provides personalized recommendations using: the overall importance of an article; its recency; the preferences of the user (and the user's segment) for the categories the item belongs to; and a measure of the ratio of users, in the segment of the user, who have read that item.

5.3.2 Multimedia Content Recommendations

In [Smyth and Cotter, 2000] [Smyth and Cotter, 2001] the *PTV* recommender system is presented. It is aimed at recommending TV programmes and presents them on a WAP-based mobile phone. In this system the user's query is represented as a favorite features vector. The products recommended to a user are presented as a ranked list. The system uses a content-based and collaborative based approach. The favorite features vector, representing the user's preferences on TV programmes, is used by the system to produce content-based recommendations. Whereas the ratings vector, i.e., the user's ratings on the recommended TV programmes are exploited to build collaborative-based recommendations. The system maintains for each user a profile schema which contains the user's indicated preferences on the TV programmes.

Another more recent example of a mobile recommender system for a digital multimedia broadcasting service has been presented in [Park et al., 2006]. Here the authors illustrate a recommendation algorithm using users' transitions data for the channels. This system, as usual in the mobile scenario, makes a large usage of implicit data, such as the start time and end time of a users content viewing, automatically logged by the client application (a J2ME Midlet).

5.3.3 Wap Portal Adaptation

Mobile portals attempt to reproduce the success of portal services on the Internet, but through mobile handsets and PDAs (see also Section 3.2). Mobile portals have not been very successful in the past, essentially because of limited usability, i.e., poor portal design and limited device functionality. They are all based on a menu hierarchically organizing the content presented in the portal. But this means that users are spending a significant amount of their time navigating to content (navigation time) and limited time interacting with content (content time). The solution proposed in [Smyth and Cotter, 2004] is based on the idea of automatically adapting the structure of a mobile portal to the needs of individual users.

The time a user takes to locate and access a specific content item is a measure of navigation effort. The effort for locating a specific content depends on the location of that item within the portal structure, and it can also be measured as the number of navigation steps required in order to locate and access the item from a given starting position. In fact, the authors found a near-perfect correlation between the time spent to locate a specific content and the number of navigation steps, also called click-distance. Hence they developed a solution based on the idea of reducing the click-distance of the content items, which a given user is likely to be interested in, by "promoting" these items to higher positions within the portal menu structure. They used a probabilistic model of user navigation to predict the likelihood that some menu option will be selected by a user, given that he is currently in a particular menu, based on his past navigation history. Hence, when a user arrives at a menu page, not necessarily the default options are displayed, instead the system computes the options that are most likely to be accessed by the user from that position.

As we mentioned above the WAP scenario is rapidly changing due to the current availability of better devices (larger screens and more computing power) and better wireless connections (larger data transfer rate). Still these results are important since a better usability of the portal would always play a major impact on the service penetration.

6 Computing Models for Mobile recommender Systems

In this section we will delve into some original recommendation computation models that have been explicitly developed for mobile RSs. In this section we will emphasize the aspects related to the distributed computing models for data storage, access and recommendation prediction that have been introduced to deal with and exploit the characteristics of mobile devices and wireless communication technologies.

6.1 Context-Dependent Recommendations

Mobile RSs, as we stressed above, can largely benefit from the exploitation of information relative to the users' current context [Abowd et al., 1997] [Chen and Kotz, 2000] [Dey, 2001] [Dourish, 2004]. "Context is any information that can be used to characterize the situation of an entity" [Dey, 2001]. Here, an entity is a person, place, or an object that is considered relevant to any phase of the recommendation process. Contextual information could be, e.g., in a tourist guide application, the companion, the weather, the temperature or the location. For example, in this application, a flea market can be recommended for a user on a sunny day with low traffic, but not during a rainy day. Context-aware computing is becoming a wide research area and recently is gaining more and more attention in recommender systems [Adomavicius et al., 2005] [Palmisano et al., 2008]. We already illustrated some mobile recommender systems that exploit contextual information [Cena et al., 2006] [Church and Smyth, 2008] [Ricci and Nguyen, 2007] [Averjanova et al., 2008] [Brown et al., 2005] [Brunato and Battiti, 2003]. Here want to extend the description focussing on the recommendation technique and how it is driven by the context model.

[Ahn et al., 2006] present an approach to mobile context-dependent recommendations that extends the classical collaborative filtering (CF) method by using information about the user and item location, the time of the recommendation and the type of the user needs: either hedonic, or neutral or utilitarian. The recommendation process starts by collecting the user position, time and needs and filtering out the items that are not located close to the user position. Then, in order to apply CF, it searches for similar users, using a particular similarity metric. This metrics combines (by multiplication) the standard adjusted cosine metric [Sarwar et al., 2001] between the active user and a neighbor user, and a measure of the similarity of the current time, position and needs between the two users. The authors collected their own rating data about shopping places in Korea and compared several versions of the CF algorithm with their proposed model showing a slightly better performance (in mean absolute error [Herlocker et al., 2004]) of their approach. The idea of using the location of the user to tune the user-to-user similarity function has also been exploited by [Horozov et al., 2006]. In their restaurant recommender system (Geowhiz) they assume that people who live in the same neighborhood are likely to visit the same nearby places. Hence, since people can be correlated in CF only if they have co-rated items, they infer that there is a higher probability of correlating people who live close to each other than correlating people who live further apart.

[van Setten et al., 2004] illustrates COMPASS, a COntext-aware Mobile Personal ASSistant, serving a tourist with information and services needed in her specific context, including the user's information goals. For example, a tourist expressing an interest in history and architecture is presented with information about nearby monuments built before 1890. The user can browse a map indicating her current location and a selection of nearby buildings, buddies, and other relevant objects for her user profile. The map and the objects shown are updated when the user moves (context changes) or when her profile or goal changes. The application accomplishes this functionality by discovering services delivering objects matching the hard criteria of the users context and goal. The retrieved objects are then sent to the recommendation engine that scores each object using soft criteria, such as the users interests and other contextual factors like the last time an object was visited. It is worth noting that in this system the user can deliberately specify the contextual conditions that matters.

As the previous example illustrates, a major issue in context-dependent reasoning is the assessment of which contextual data and conditions should really influence the recommendation procedure. For instance in [Adomavicius et al., 2005] a search procedure is performed to identify the segments of contextually tagged ratings that must be considered when a collaborative-filtering based prediction is computed for a particular context-dependent target situation. The algorithm search, in the space of all possible segments, for those segments containing ratings that determine a different output recommendation from that computed by a standard non-contextual dependent CF algorithm. [Yap et al., 2007] explores another approach to tackle the same problem, basing on Bayesian networks (BNs), it identifies the minimal set of parameters that are important for a user, hence minimizing the cost of the context acquisition phase. Their learning procedure iteratively discards parameters that are not connected to the user rating variable in the learned BN. The remaining parameters constitute the minimal context for that user. Moreover, in order to cope with possibly missing and erroneous context data, they exploit the causal dependencies among context parameters. They capture causal dependencies with a two-tiered context model and they show that the BN learned on this context model is able to compensate for those context inputs that are left unspecified during the prediction.

[Yu et al., 2006] in addition to adapting to the more classical categories of user preference and situation context (e.g., location and time) introduces the "capability context", i.e., device and network capability, as input for both content and presentation recommendations. Their goal is similar to that described in Section 4.1 apropos of content adaptation. In order to deal with all three context categories, they use a hybrid recommendation approach exploiting content-based methods, a Bayesian classifier, and rule-based methods. Their context-aware media recommendation platform is called CoMeR and supports media recommendation, adaptation, and delivery for smart phones. It is worth noting that they used an ontology-based context model for context representation. This model adopts OWL as representation language to enable expressive context description and data interoperability with third-party services and applications. They also adopt the multidimensional model proposed by [Adomavicius et al., 2005], so for instance their recommendation output is not limited to a standard user-adapted rating prediction, but it can generate a more specific rating prediction for each type of user mobile device or type of output for the same item recommendation (e.g., image vs. text description of the item).

6.2 Distributed Models

In this section we will survey a couple of computational models that are particularly suited for mobile recommender systems since they exploit the distributed data storage of a network of mobile devices and they exploit specific communication paradigms for mobile users.

Standard Web-based RSs are based on a client/server distributed computing model. The browser, either running on a mobile device or on a standard PC, connects to the web server hosting the recommender and asks/retrieves recommendations. The major limitations of this approach are related to the bottleneck generated by the server that must be always running and accessible by the clients to return the requested recommendations.

Peer-to-Peer (P2P) computing refers to a subclass of distributed computing, where system's functionality is achieved in a fully decentralized way by using a set of distributed resources [Androutsellis-Theotokis and Spinellis, 2004]. P2P systems usually lack a dedicated centrally managed infrastructure, depending rather on a voluntary contribution of resources (e.g., computing power, data, and network traffic) by the connected peers. As a result, P2P systems provide a purely distributed communication middleware with theoretically unlimited storage, communication, and processing capabilities. Hence, P2P systems are characterized by the following advantages: cost sharing and reduction, improved scalability, reliability and robustness, resource aggregation and operability, increased autonomy, dynamism, and high levels of anonymity and privacy.

P2P architectures offer a convenient platform for developing mobile recommender systems, they can make them more portable and more reliable. These issues have been investigated in [Miller et al., 2003] [Miller et al., 2004]. The authors advocate the importance of truly personal recommenders for delivering high quality recommendations to mobile devices, even when disconnected from the Internet. It is also stressed that these architectures can protect the users privacy by storing personal information locally, or by sharing it in encrypted form. PocketLens is a collaborative filtering algorithm that can use five peer-to-peer architectures for finding neighbors. The authors are able to evaluate the architectures and algorithms in an off-line experiment, and they show that PocketLens can run on connected servers, on usually connected workstations, or on occasionally connected portable devices, and produce recommendations that are as good as the best published algorithms to date. It is worth noting that the recommendation algorithm in this work is a standard item-to-item collaborative filtering [Sarwar et al., 2001], but the user ratings are maintained in a distributed way by the P2P infrastructure, hence users keep their ratings on their mobile devices and when a recommendation is needed P2P lookup methods are used to find and retrieve relevant ratings.

In [Schifanella et al., 2008] the authors develop a similar approach. They observe that mobile devices may be unable to access the Internet or a remote server for several reasons (cost, failure of wireless network, etc.). Therefore, the architectural model should allow the recommender system to operate even without that kind of connectivity. But, a user device can also be able to connect to other mobile devices, which are in the proximity, via ad-hoc connections, hence relying on a very limited portion of the users' community and just on a subset of all the available data. The authors model the relationships between users with a similarity graph that links users to each other by using a configurable affinity threshold. The proposed system, MobHinter, then allows a mobile device to identify the affinity network neighbors from random ad-hoc communications. The collected information is then used to incrementally refine locally calculated predictions, with no need of interacting with a remote server or accessing the Internet. The system maintains a queue of "neighbos's neighbors", where the rationale is that if v is among the best neighbors of a user u, then it is likely that v neighbors are good for u as well. Differently from the approach presented in [Miller et al., 2004], the data describing the neighbors are here locally maintained. If some of the "neighbor's neighbors" are off line while building the model, they are substituted with a pool of quality neighbors.

6.3 Proactive Recommendations

As we mentioned in Section 3.3 recent developments in wireless communication technologies are transforming many kinds of appliances, e.g. electric household, into computerized and networked machines. When the surrounding environment will recognize us and our activities, by using sensors, then novel proactive services, which exploit this knowledge, will be possible. This idea was initially proposed by Weiser and defined as “ubiquitous computing” [Weiser, 1994].

In [Sae-Ueng et al., 2008] the authors focus on personalized shopping assistance by using personal behavior log analysis on ubiquitous shop space. They classify the possible user behaviors in five classes: “standing”, the state in which a consumer stops by at a certain distance in front of a certain object; “viewing”, where a consumer looks at a certain object; “touching”, where a consumer touches a certain object; “carrying”, where the consumer picks up a certain object from a shelf; and “fitting”, where a consumer uses a mirror to match a certain object with her appearance. They use RFID sensors and a digital camera to detect these behaviors. The system presents recommendations using three channels: a display next to the product, a robot interacting with the user and with music and sounds illustrating products. In this way, the system offers an augmented reality environment.

The proposed recommendation algorithm first checks the position of the user. Then it accumulates data on the consumer’s actions, discovers commonalities among the items that were interesting for the user, and finally recommends personalized item information to the user, based on his behavior and current situation. After the recommendation it collects the user’s feedback for the recommended items to improve the recommendation service. In their approach, user actions are interpreted and converted into ratings. So, for instance, if the user touches a product this is interpreted as a mild form of interest in the product and therefore in a small/medium rating. The authors evaluated the capability of the system to detect the actual user behavior and found a precision around 93%. They also run controlled experiments where users where provided recommendations with the three described channels. They found that the robot is the best attention drawer. The display method is suitable for delivering the description and position of a product. But, if the robot and display method are combined together effectively, they will create an effective consumer-friendly environment that helps the user to make better decisions.

[Bulander et al., 2005] presents the MoMa-system, where end users can create orders according to a given catalogue. This catalogue is a hierarchically ordered set of possible product and service offers that are described by appropriate attributes. Each category is specified by certain attributes, for instance, in a restaurant scenario this could be “price level” and “style”. When creating an order the client application will automatically fill in the appropriate context and profile parameters, e.g., “location” and “weather”: so that, for example, the facility shouldn’t be too far away from the current location of the user and beer gardens should not be recommended if it is raining. On the other side the advertisers put offers into the MoMa-system. These offers are also formulated according to the catalogue. When the system detects a pair of a matching order and offer, the end user is notified the way she specified (e.g. SMS, e-mail). At this point the user can decide whether she likes to contact the advertiser to accept that offer or not. [Mahmoud, 2006] describes another context-dependent proactive mobile advertising system.

There are many other systems and techniques that exploit contextual information to proactively personalize the interaction experience on mobile devices. We want to conclude this section with [Yoon et al., 2008], that explores a novel area, context-aware photo selection. Their algorithm takes into account the mobile phone contexts, such as the current location and the recent calls, to decide whether and when to offer selected pictures to the user. For instance, the photo selection algorithm for the mBackground service, i.e., a service that offers a picture to be used for the screen device background, selects a photo that was taken in the context most similar to the current one, i.e., considering the similarity of the date, time, GPS coordinates, and call logs information attached to the two photos. Or, the photo selection algorithm of mReminder, which is another function of the system, selects a photo that is relevant to the alarm event, including the event name, location, and the people involved.

7 Future Directions and Conclusions

It is probably clear now, at the end of this paper, that the research area of mobile RSs is not clearly shaped nor it is likely to become much clearer in the next future. The main determinant of this uncertainty is that it is not clear what will be the characteristics of the new successful mobile communication devices, i.e., those that will get the largest market share.

Consider that the first mobile recommender systems have been developed for PDA devices. These were rather popular around the first 2000’s and they were estimated to obtain a larger and larger success. But this did not occur and they have been confined mainly to professional usage or hosting custom applications (e.g., for ticket emission in a train or to collect order in a restaurant). In practice modern advanced mobile phones have today better specifications than the "old" PDAs (see Section 3.4), with respect to computation power, networking capability and graphic displays. Or in other words, the PDAs and mobile phone segments have merged (convergence). But also new devices have come and are coming to the fore, such as subnotebooks and MID (Mobile Internet Devices). Subnotebooks are light (1 Kilo) and (sometime) inexpensive laptops that can be easily transported and can run all the standard PC applications. They have a keyboard and a screen size around 8-10 Inches. MID devices (e.g., Samsungs Q1, Asus R2H, Founders Mini-Note, TabletKiosk eo* i7210) have been launched nowadays (2009), they are in the middle between phones and subnotebooks. They are multimedia-capable handheld computers providing wireless Internet access, and are designed to provide entertainment, information and location-based services for the consumer market. MID devices do not have a keyboard but they are light (300 grams) and still with a screen size (approximately 7 inches) that allows to browse the Web effectively. Digital ink based devices are also very promising especially for book reading. Digital ink screens are light, foldable and very parsimonious in term of battery consumption. Currently they support only black and white displays but new technologies are now studied and it il likely that in the next years light, rather large, and portable color displays could offer ubiquitous access to complex interfaces.

In fact, as we noted in this paper, a consistent number of new methodologies and techniques have been developed to address the limitations of the currently available mobile devices, i.e., limited screen size, limited or intermittent networking, limited computational power. All these limitations may quickly disappear making obsolete or at least less relevant some researches. It is also very likely that in the future we will not even consider "mobile services" as a new or distinct type of information services because the large majority of our devices will be mobile and wireless. But at the same time new issues will surely be raised by the limitations of these new devices, with respect the even growing expectations of the market.

Notwithstanding, or because of that, still some research topics are clearly central in this area and will deserve more and more attention. We will close this paper by listing some of them.

- Elicitation of the user's preferences. Fast and accurate acquisition of user's preferences is still an open research problem. Mobile recommenders can need even more user preferences, e.g., those related to the context, and have not flexible user interfaces for supporting this task. Inferring (implicit) preferences from user's behavior sounds as the most obvious solution, but new interfaces, e.g., based on speech recognition, could provide a more effective channel. Other approaches could be related to the video and imaging capabilities of new devices, such are those that are at the base of augmented reality applications. Using these techniques the user input can be driven by the interaction with the space (.e.g. gestures or pointing) and can overcome some of the limitations of more classical interactions (keyboard).
- Proactive and sensor based recommendations. Most of the RSs wait for a user's request before delivering any recommendation. The query is possibly very simple, as in standard CF systems, e.g., just detect that the user is accessing the system. So minimal input is asked to the user, but none of the reviewed systems is capable to proactively interrupt the user activity with unsolicited but relevant recommendations. There is too much risk to bother the user with irrelevant information that nobody has really tried this approach. Besides, new generation of inexpensive and reliable biometric sensors can make this type of recommendations possible and convenient. This can revolutionize the role of RSs from topic oriented information seeking and decision making tools to information discovery and entertaining companions.
- Explanations of recommendations. This is topic that received very few studies and it is clearly very important for the mobile scenario. Explanation is based on item descriptions and this is problematic with small screens and with the typical "noise" of mobile environments. New approaches for the explanation of the recommendations are needed. A stricter integration of the recommender with other information services could be an approach, but the core problem of limiting the amount of descriptive information associated to the recommendations is still an open issue.
- Security and privacy and the user memory. Mobile users will always room from systems to systems and from networks to networks. A mobile user must be open to discover new services and at the same time must be prepared to shade her privacy and security from the potential dangers coming from unknown and hostile programs. Related to the privacy issue is a functionality that we expect to grow considerably in the future, i.e., supporting personal memories; helping the user to remember personal facts and tasks, and helping her to make the best usage of this information.
- Portable recommender systems. This last point refers to the capability to bring and move a RS through different platforms and devices. The user in practice should be able to operate an information services using a variety of devices (his phone, laptop and digital camera) and the activity performed, and the service obtained, with these devices and interfaces should be integrated. The user must be recognized and served appropriately whatever

devices is used, and activities performed with one device should be part of the whole service personalization process.

This list is by far not exhaustive, but we hope to have provided at least some illustrative examples of the future developments that we can now foresee in this new and fast developing application field.

References

[pho, 2009] (2009). Phocuswright's mobile: The next platform for travel. PhoCusWright, www.phocuswright.com.

[Abowd et al., 1997] Abowd, G. D., Atkeson, C. G., Hong, J., Long, S., Kooper, R., and Pinkerton, M. (1997). Cyberguide: A mobile context-aware tour guide. *Wireless Networks*, 3(5):421–433.

[Adomavicius et al., 2005] Adomavicius, G., Sankaranarayanan, R., Sen, S., and Tuzhilin, A. (2005). Incorporating contextual information in recommender systems using a multidimensional approach. *ACM Trans. Inf. Syst.*, 23(1):103–145.

[Adomavicius and Tuzhilin, 2005] Adomavicius, G. and Tuzhilin, A. (2005). Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE Transactions on Knowledge and Data Engineering*, 17(6):734–749.

[Ahn et al., 2006] Ahn, H., Kim, K.-J., and Han, I. (2006). Mobile advertisement recommender system using collaborative filtering: Mar-cf. In *Proceedings of the 2006 Conference of the Korea Society of Management Information Systems*, pages 709–715.

[Anand and Mobasher, 2005] Anand, S. S. and Mobasher, B. (2005). Intelligent techniques for web personalization. In *Intelligent Techniques for Web Personalization*, pages 1–36. Springer.

[Androutsellis-Theotokis and Spinellis, 2004] Androutsellis-Theotokis, S. and Spinellis, D. (2004). A survey of peer-to-peer content distribution technologies. *ACM Comput. Surv.*, 36(4):335–371.

[Averjanova et al., 2008] Averjanova, O., Ricci, F., and Nguyen, Q. N. (2008). Map-based interaction with a conversational mobile recommender system. In *Mobile Ubiquitous Computing, Systems, Services and Technologies, 2008. UBICOMM '08. The Second International Conference on*, pages 212–218.

[Balabanovic and Shoham, 1997] Balabanovic, M. and Shoham, Y. (1997). Content-based, collaborative recommendation. *Communication of ACM*, 40(3):66–72.

[Balfe and Smyth, 2005] Balfe, E. and Smyth, B. (2005). An analysis of query similarity in collaborative web search. In Losada, D. E. and Fernández-Luna, J. M., editors, *ECIR*, volume 3408 of *Lecture Notes in Computer Science*, pages 330–344. Springer.

[Barrat et al., 2008] Barrat, A., Cattuto, C., Colizza, V., Pinton, J.-F., den Broeck, W. V., and Vespignani, A. (2008). High resolution dynamical mapping of social interactions with active rfid. *Computing Research Repository*, abs/0811.4170.

[Bertelé and Rangone, 2007] Bertelé, U. and Rangone, A. (2007). Rapporto mobile and wireless business. Politecnico di MIlano.

[Billsus and Pazzani, 2000] Billsus, D. and Pazzani, M. J. (2000). User modeling for adaptive news access. *User Modeling and User-Adapted Interaction*, 10(2-3):147–180.

[Billsus and Pazzani, 2007] Billsus, D. and Pazzani, M. J. (2007). Adaptive news access. In *The Adaptive Web*, pages 550–570. Springer Berlin / Heidelberg.

[Billsus et al., 2000] Billsus, D., Pazzani, M. J., and Chen, J. (2000). A learning agent for wireless news access. In Riecken, D., Benyon, D., and Lieberman, H., editors, *Proceedings of the 5th International Conference on Intelligent User Interfaces, IUI'00*, pages 33–36, New Orleans, LA, USA. ACM Press.

[Bridge et al., 2006] Bridge, D., Göker, M., McGinty, L., and Smyth, B. (2006). Case-based recommender systems. *The Knowledge Engineering review*, 20(3):315–320.

[Brown et al., 2005] Brown, B., Chalmers, M., Bell, M., MacColl, I., Hall, M., and Rudman, P. (2005). Sharing the square: Collaborative leisure in the city streets. In Gellersen, H., Schmidt, K., Beaudouin-Lafon, M., and Mackay, W. E., editors, *ECSCW*, pages 427–447. Springer.

[Brunato and Battiti, 2003] Brunato, M. and Battiti, R. (2003). Pilgrim: A location broker and mobility-aware recommendation system. In *PerCom*, pages 265–272. IEEE Computer Society.

[Brusilovsky et al., 2007] Brusilovsky, P., Kobsa, A., and Nejdl, W., editors (2007). *The Adaptive Web, Methods and Strategies of Web Personalization*, volume 4321 of *Lecture Notes in Computer Science*. Springer.

[Bulander et al., 2005] Bulander, R., Decker, M., Schiefer, G., and Kolmel, B. (2005). Comparison of different approaches for mobile advertising. *Mobile Commerce and Services, 2005. WMCS '05. The Second IEEE International Workshop on*, pages 174–182.

[Burigat et al., 2005] Burigat, S., Chittaro, L., and Marco, L. D. (2005). Bringing dynamic queries to mobile devices: A visual preference-based search tool for tourist decision support. In Costabile, M. F. and Paternò, F., editors, *INTERACT*, volume 3585 of *Lecture Notes in Computer Science*, pages 213–226. Springer.

[Burke, 2002] Burke, R. (2002). Interactive critiquing for catalog navigation in e-commerce. *Artificial Intelligence Review*, 18(3-4):245–267.

[Burke, 2007] Burke, R. (2007). Hybrid web recommender systems. In *The Adaptive Web*, pages 377–408. Springer Berlin / Heidelberg.

[Cena et al., 2008] Cena, F., Carmagnola, F., Cortassa, O., Gena, C., Wang, Y., Stash, N., and Aroyo, L. (2008). Tag interoperability in cultural web-based applications. In Brusilovsky, P. and Davis, H. C., editors, *Hypertext*, pages 221–222. ACM.

[Cena et al., 2006] Cena, F., Console, L., Gena, C., Goy, A., Levi, G., Modeo, S., and Torre, I. (2006). Integrating heterogeneous adaptation techniques to build a flexible and usable mobile tourist guide. *AI Commun.*, 19(4):369–384.

[Chen and Kotz, 2000] Chen, G. and Kotz, D. (2000). A survey of context-aware mobile computing research. Technical Report TR2000-381, Dartmouth Computer Science.

[Chen and Pu, 2009] Chen, L. and Pu, P. (2009). Interaction design guidelines on critiquing-based recommender systems. *User Model. User-Adapt. Interact.*, 19(3):167–206.

[Choi and Choi, 2007] Choi, S. S. and Choi, M.-K. (2007). Consumer's privacy concerns and willingness to provide personal information in location-based services. *Advanced Communication Technology, The 9th International Conference on*, 3:2196–2199.

[Church and Smyth, 2008] Church, K. and Smyth, B. (2008). Who, what, where & when: a new approach to mobile search. In Bradshaw, J. M., Lieberman, H., and Staab, S., editors, *Intelligent User Interfaces*, pages 309–312. ACM.

[Church et al., 2008] Church, K., Smyth, B., Bradley, K., and Cotter, P. (2008). A large scale study of european mobile search behaviour. In ter Hofte, G. H., Mulder, I., and de Ruyter, B. E. R., editors, *Mobile HCI*, ACM International Conference Proceeding Series, pages 13–22. ACM.

[Church et al., 2007] Church, K., Smyth, B., Cotter, P., and Bradley, K. (2007). Mobile information access: A study of emerging search behavior on the mobile internet. *ACM Trans. Web*, 1(1):4.

[Church et al., 2006] Church, K., Smyth, B., and Keane, M. T. (2006). Evaluating interfaces for intelligent mobile search. In *W4A: Proceedings of the 2006 international cross-disciplinary workshop on Web accessibility (W4A)*, pages 69–78, New York, NY, USA. ACM.

[Dey, 2001] Dey, A. K. (2001). Understanding and using context. *Personal and Ubiquitous Computing*, 5(1):4–7.

[Dourish, 2004] Dourish, P. (2004). What we talk about when we talk about context. *Personal and Ubiquitous Computing*, 8(1):19–30.

[Dunlop et al., 2004] Dunlop, M., Morrison, A., McCallum, S., Ptaskinski, P., Risbey, C., and Stewart, F. (2004). Focussed palmtop information access through starfield displays and profile matching. In *Proceedings of the Workshop on Mobile and Ubiquitous Information Access*, pages 79–89, Glasgow, Scotland.

[Dunlop et al., 2007] Dunlop, M. D., Elsey, B., and Masters, M. M. (2007). Dynamic visualisation of ski data: a context aware mobile piste map. In Cheok, A. D. and Chittaro, L., editors, *Mobile HCI*, volume 309 of *ACM International Conference Proceeding Series*, pages 375–378. ACM.

[Fesenmaier et al., 2006] Fesenmaier, D. R., Werthner, H., and Woeber, K. (2006). *Destination Recommendation Systems: Behavioural Foundations and Applications*. CABI Publishing.

[Gansemer et al., 2007] Gansemer, S., Groner, U., and Maus, M. (2007). Database classification of mobile devices. *Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications, 2007. IDAACS 2007. 4th IEEE Workshop on*, pages 699–703.

[Goy et al., 2007] Goy, A., Ardissono, L., and Petrone, G. (2007). Personalization in e-commerce applications. In [Brusilovsky et al., 2007], pages 485–520.

[Herlocker et al., 1999] Herlocker, J. L., Konstan, J. A., Borchers, A., and Riedl, J. (1999). An algorithmic framework for performing collaborative filtering. In *SIGIR '99: Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, August 15-19, 1999, Berkeley, CA, USA*, pages 230–237.

[Herlocker et al., 2004] Herlocker, J. L., Konstan, J. A., Terveen, L. G., and Riedl, J. T. (2004). Evaluating collaborative filtering recommender systems. *ACM Transaction on Information Systems*, 22(1):5–53.

[Hinze and Buchanan, 2006] Hinze, A. and Buchanan, G. (2006). The challenge of creating cooperating mobile services: experiences and lessons learned. In *ACSC '06: Proceedings of the 29th Australasian Computer Science Conference*, pages 207–215, Darlinghurst, Australia, Australia. Australian Computer Society, Inc.

[Horozov et al., 2006] Horozov, T., Narasimhan, N., and Vasudevan, V. (2006). Using location for personalized POI recommendations in mobile environments. In *Proc. Int'l Sym. Applications on Internet*, pages 124–129. EEE Computer Society.

[Jannach, 2006] Jannach, D. (2006). Finding preferred query relaxations in content-based recommenders. In *3rd International IEEE Conference on Intelligent Systems*, pages 355–360.

[Jones and Marsden, 2005] Jones, M. and Marsden, G. (2005). *Mobile Interaction Design*. John Wiley and Sons.

[Jones et al., 2004] Jones, S., Jones, M., and Deo, S. (2004). Using keyphrases as search result surrogates on small screen devices. *Personal and Ubiquitous Computing*, 8(1):55–68.

[Kim et al., 2004] Kim, C. Y., Lee, J. K., Cho, Y. H., and Kim, D. H. (2004). Viscors: A visual-content recommender for the mobile web. *IEEE Intelligent Systems*, 19(6):32–39.

[Koren, 2008] Koren, Y. (2008). Factorization meets the neighborhood: a multifaceted collaborative filtering model. In Li, Y., Liu, B., and Sarawagi, S., editors, *KDD*, pages 426–434. ACM.

[Kramer et al., 2006] Kramer, R., Modsching, M., and ten Hagen, K. (2006). Field study on methods for elicitation of preferences using a mobile digital assistant for a dynamic tour guide. In *SAC '06: Proceedings of the 2006 ACM symposium on Applied computing*, pages 997–1001, New York, NY, USA. ACM Press.

[Laakko and Hiltunen, 2005] Laakko, T. and Hiltunen, T. (2005). Adapting web content to mobile user agents. *Internet Computing, IEEE*, 9(2):46–53.

[Lamber et al., 2009] Lamber, P., Girardello, A., Ricci, F., and Mitterer, M. (2009). Mobiday: a personalized context-aware mobile service for day hospital workflow support. In Grasso, F. and Paris, C., editors, *Proceedings of the AIME09 International Workshop on: Personalization for e-Health*, pages 15–19, July 19, 2009. Verona, Italy.

[Lee and Park, 2007] Lee, H. and Park, S. J. (2007). Moners: A news recommender for the mobile web. *Expert Systems with Applications*, 32(1):143 – 150.

[Mahmoud, 2006] Mahmoud, Q. (2006). Provisioning context-aware advertisements to wireless mobile users. *Multimedia and Expo, 2006 IEEE International Conference on*, pages 669–672.

[Manning, 2008] Manning, C. (2008). *Introduction to Information Retrieval*. Cambridge University Press, Cambridge.

[Marchionini, 2006] Marchionini, G. (2006). Exploratory search: from finding to understanding. *Commun. ACM*, 49(4):41–46.

[Marchionini and White, 2009] Marchionini, G. and White, R. W. (2009). Information-seeking support systems [guest editors' introduction]. *IEEE Computer*, 42(3):30–32.

[McGinty and Smyth, 2006] McGinty, L. and Smyth, B. (2006). Adaptive selection: An analysis of critiquing and preference-based feedback in conversational recommender systems. *International Journal of Electronic Commerce*, 11(2):35–57.

[McSherry, 2004] McSherry, D. (2004). Incremental relaxation of unsuccessful queries. In Funk, P. and Calero, P. A. G., editors, *ECCBR 2004, the 7th European Conference on Case-Based Reasoning*, pages 331–345, Madrid, Spain.

[Miller et al., 2003] Miller, B. N., Albert, I., Lam, S. K., Konstan, J. A., and Riedl, J. (2003). Movielens unplugged: Experiences with an occasionally connected recommender system. In *Proceedings of the 7th International Conference on Intelligent User Interfaces, IUI'03*, pages 263–266, Miami, Florida, USA. ACM Press.

[Miller et al., 2004] Miller, B. N., Konstan, J. A., and Riedl, J. (2004). Pocketlens: Toward a personal recommender system. *ACM Transaction on Information Systems*, 22(3):437–476.

[Mirzadeh and Ricci, 2007] Mirzadeh, N. and Ricci, F. (2007). Cooperative query rewriting for decision making support and recommender systems. *Applied Artificial Intelligence*, 21:1–38.

[Mirzadeh et al., 2005] Mirzadeh, N., Ricci, F., and Bansal, M. (2005). Feature selection methods for conversational recommender systems. In *Proceedings of the IEEE International Conference on e-Technology, e-Commerce and e-Services*, Hong Kong. IEEE Press.

[Palmisano et al., 2008] Palmisano, C., Tuzhilin, A., and Gorgoglione, M. (2008). Using context to improve predictive modeling of customers in personalization applications. *Knowledge and Data Engineering, IEEE Transactions on*, 20(11):1535–1549.

[Park et al., 2007] Park, M.-H., Hong, J.-H., and Cho, S.-B. (2007). Location-based recommendation system using bayesian user's preference model in mobile devices. In Indulska, J., Ma, J., Yang, L. T., Ungerer, T., and Cao, J., editors, *UIC*, volume 4611 of *Lecture Notes in Computer Science*, pages 1130–1139. Springer.

[Park et al., 2006] Park, S., Kang, S., and Kim, Y.-K. (2006). A channel recommendation system in mobile environment. *Consumer Electronics, IEEE Transactions on*, 52(1):33–39.

[Pazzani and Billsus, 2007] Pazzani, M. J. and Billsus, D. (2007). Content-based recommendation systems. In [Brusilovsky et al., 2007], pages 325–341.

[Pernici, 2006] Pernici, B. (2006). *Mobile Information Systems*. Springer, Berlin.

[Resnick and Varian, 1997] Resnick, P. and Varian, H. R. (1997). Recommender systems. *Communications of the ACM*, 40(3):56–58.

[Reynolds, 2008] Reynolds, F. (2008). Adapting content. *Pervasive Computing, IEEE*, 7(4):6–8.

[Ricci et al., 2006] Ricci, F., Cavada, D., Mirzadeh, N., and Venturini, A. (2006). Case-based travel recommendations. In Fesenmaier, D. R., Woeber, K., and Werthner, H., editors, *Destination Recommendation Systems: Behavioural Foundations and Applications*, pages 67–93. CABI.

[Ricci and Nguyen, 2007] Ricci, F. and Nguyen, Q. N. (2007). Acquiring and revising preferences in a critique-based mobile recommender system. *IEEE Intelligent Systems*, 22(3):22–29.

[Ricci et al., 2010] Ricci, F., Rokach, L., and Shapira, B., editors (2010). *Recommender Systems Handbook*. Springer.

[Sae-Ueng et al., 2008] Sae-Ueng, S., Pinyapong, S., Ogino, A., and Kato, T. (2008). Personalized shopping assistance service at ubiquitous shop space. *Advanced Information Networking and Applications - Workshops, 2008. AINAW 2008. 22nd International Conference on*, pages 838–843.

[Sarwar et al., 2001] Sarwar, Karypis, G., Konstan, J., and Riedl, J. (2001). Item-based collaborative filtering recommendation algorithms. In *Proceedings of WWW10 Conference*, pages 285–295, Hong Kong. ACM.

[Schifanella et al., 2008] Schifanella, R., Panisson, A., Gena, C., and Ruffo, G. (2008). Mobhinter: epidemic collaborative filtering and self-organization in mobile ad-hoc networks. In *RecSys '08: Proceedings of the 2008 ACM conference on Recommender systems*, pages 27–34, New York, NY, USA. ACM.

[Schneiderman, 1994] Schneiderman, B. (1994). Dynamic queries for visual information seeking. *IEEE Software*, 11(6):70–77.

[Shiller, 2003] Shiller, J. H. (2003). *Mobile Communications*. Addison-Wesley.

[Smyth et al., 2005] Smyth, B., Balfe, E., Boydell, O., Bradley, K., Briggs, P., Coyle, M., and Freyne, J. (2005). A live-user evaluation of collaborative web search. In Kaelbling, L. P. and Saffiotti, A., editors, *IJCAI*, pages 1419–1424. Professional Book Center.

[Smyth and Cotter, 2000] Smyth, B. and Cotter, P. (2000). A personalised tv listings service for the digital tv age. *Knowl.-Based Syst.*, 13(2-3):53–59.

[Smyth and Cotter, 2001] Smyth, B. and Cotter, P. (2001). Personalized electronic program guides for digital tv. *AI Magazine*, 22(2):89–98.

[Smyth and Cotter, 2004] Smyth, B. and Cotter, P. (2004). Mp^3 - mobile portals, profiles and personalization. In Levene, M. and Poulovassilis, A., editors, *Web Dynamics*, pages 411–434. Springer.

[Tesoriero et al., 2008] Tesoriero, R., Gallud, J., Lozano, M., and Penichet, V. (2008). Using active and passive rfid technology to support indoor location-aware systems. *Consumer Electronics, IEEE Transactions on*, 54(2):578–583.

[Tumas and Ricci, 2009] Tumas, G. and Ricci, F. (2009). Personalized mobile city transport advisory system. In *In Information and Communication Technologies in Tourism 2009*, pages 173–184. Springer.

[Tung and Soo, 2004] Tung, H.-W. and Soo, V.-W. (2004). A personalized restaurant recommender agent for mobile e-service. In Yuan, S.-T. and Liu, J., editors, *Proceedings of the IEEE International Conference on e-Technology, e-Commerce and e-Service, EEE'04*, pages 259–262, Taipei, Taiwan. IEEE Computer Society Press.

[Turban et al., 2008] Turban, E., Lee, J. K., King, D., McKay, J., and Marshall, P. (2008). *Electronic Commerce*. Prentice Hall.

[van Setten et al., 2004] van Setten, M., Pokraev, S., and Koolwaaij, J. (2004). Context-aware recommendations in the mobile tourist application compass. In Nejdl, W. and De Bra, P., editors, *Adaptive Hypermedia 2004*, pages 235–244. Springer Verlag.

[Vatanparast, 2007] Vatanparast, R. (2007). Piercing the fog of mobile advertising. *Management of Mobile Business, 2007. ICMB 2007. International Conference on the*, pages 19–19.

[Want, 2006] Want, R. (2006). An introduction to rfid technology. *Pervasive Computing, IEEE*, 5(1):25–33.

[Weiser, 1994] Weiser, M. (1994). The world is not a desktop. *interactions*, 1(1):7–8.

[Werthner, 2003] Werthner, H. (2003). Intelligent systems in travel and tourism. In *Proceeding of the 18th International Joint Conference on Artificial Intelligence, IJCAI-03*, Acapulco, Mexico.

[Werthner and Ricci, 2003] Werthner, H. and Ricci, F. (2003). Electronic commerce and tourism. *Communication of ACM*. to appear.

[Yap et al., 2005] Yap, G.-E., Tan, A.-H., and Pang, H. (2005). Dynamically-optimized context in recommender systems. In Chrysanthis, P. K. and Samaras, G., editors, *Mobile Data Management*, pages 265–272. ACM.

[Yap et al., 2006] Yap, G.-E., Tan, A.-H., and Pang, H. (2006). Discovering causal dependencies in mobile context-aware recommenders. In *MDM*, page 4. IEEE Computer Society.

[Yap et al., 2007] Yap, G.-E., Tan, A.-H., and Pang, H. (2007). Discovering and exploiting causal dependencies for robust mobile context-aware recommenders. *IEEE Trans. Knowl. Data Eng.*, 19(7):977–992.

[Yoon et al., 2008] Yoon, Y., Ahn, Y., Lee, G., Hong, S., and Kim, M. (2008). Context-aware photo selection for promoting photo consumption on a mobile phone. In *MobileHCI '08: Proceedings of the 10th international conference on Human computer interaction with mobile devices and services*, pages 33–42, New York, NY, USA. ACM.

[Yu et al., 2006] Yu, Z., Zhou, X., Zhang, D., Chin, C.-Y., Wang, X., and Men, J. (2006). Supporting context-aware media recommendations for smart phones. *IEEE Pervasive Computing*, 5(3):68–75.