PersonalTVware: An Infrastructure to Support the Context-Aware Recommendation for Personalized Digital TV

Fábio Santos da Silva, Luiz Gustavo Pacola Alves, and Graça Bressan

***Abstract*—The coming of the Digital TV will bring a significant increase in the number TV programs offered by TV operators. Consequently, the user are facing it difficulty to find out the most interesting TV programs among the various options available. In this new scenario, the recommender systems stand out as a possible solution to the information overload problem. However, the current approaches to recommend content for Digital TV rarely considers the context during the recommendation process. Thus, this paper presents a software infrastructure – entitled PersonalTVware - to support context-aware recommendation of TV programs. To validate the PersonalTVware, a context-aware recommender system was implemented as a concept proof. In order to evaluate the quality of the recommendation, a number of experiments have been conducted. The results indicate that consider both user's profile and context can provide better recommendations.**

***Index Terms*—Context-awareness, interactive digital TV, recommender systems, ubiquitous computing.**

I. Introduction

The advent of Digital TV will bring a significant increase in the number of TV programs available to end-users, which leads to information overload problem [1]. Consequently, the users are facing this problem and having difficulties to find out the favorite TV program among the options available. The traditional tool known as Electronic Program Guide (EPG), therefore, has not efficiently responded to the user's needs.

The EPG simply displays long lists of TV programs requiring the user to spend a great deal of time looking for information on his/her favorite TV programs. In this new scenario, the recommender systems stand out as a possible solution. These systems filter relevant items according to user's preferences or group of users who have similar profiles.

However, the current generation of recommender systems for Digital TV operates in two-dimensional *User x Item* space. That is, they make their recommendations based only on the user's profile and item information without taking into consideration the contextual information [2]. In many situations, the user's interest may also depend significantly on the context. According to Dey [3], the context is any information that can be used to characterize the situation of an entity. An entity is a person, place or object that is considered relevant to the interaction between a user and an application, including the user and the application themselves.

In the case of recommender systems for Digital TV, it is also important to consider the information about the context of the entity user. Generally, the contextual information can be identified from six basic contextual known as: *where* (location), *who* (identification), *what* (action or activity), *when* (time), *why* (motivation behind the action) and *how* (a way to identify how the elements of the context are collected) [2]. According to Dey [3] a context-aware system uses the context to provide relevant information and/or services to the user, where relevancy depends on the user's task.

Thus, it becomes important to extend the traditional approaches for automatic content recommendation to support the exploration of the user's context, which may improve the quality of the generated recommendations [2]. Some questions related to the context can be exploited such as: *who* is the user and *when* he attends a particular genre of TV programs? On Sunday morning or on Monday evening and when does he come from office? *Where* and *how* the TV program will be seen? At home through a Digital TV receiver connected to full HD TV set or at school on portable TV? And *what* kind of TV program is considered relevant in such situation for the user then watching TV? Depending on his/her context, the user may have different viewing preferences and needs.

Therefore, this paper presents the PersonalTVware, a software infrastructure to support the development of context-aware recommender systems for Digital TV. The PersonalTVware provides components that allow the TV programs filtering, manage information regarding the context, user profile and TV programs, and cross-context reasoning to infer contextual preferences.

The task of inferring contextual preferences is based on data mining and machine learning techniques such as decision tree classifier, naïve Bayesian classifier, back-propagation (a neural network), and case-based reasoning technique [4]. The context-aware information filtering is based on content-based filtering technique [2]. Thus, developers of recommender systems focus efforts on usability concepts of their systems, leaving questions on the low-level PersonalTVware manage. To validate the PersonalTVware in a case study, a context-aware recommender system was implemented as a concept proof.

This paper is organized as follow: section II discusses the related works, section III describes the PersonalTVware

Manuscript received February 16, 2012; revised March 27, 2012.

Authors are with the Laboratory of Computer Architecture and Network, University of São Paulo (USP), São Paulo, Brazil (e-mail: fsilva@larc.usp.br; luizgpa@larc.usp.br; gbressan@larc.usp.br).

infrastructure; section IV describes the implementation and evaluation and section V presents the conclusion and future work.

II. Related Works

Recently in the scientific community it is possible to find some researches that address the use of context and recommender systems. Regarding recommender systems for digital TV, there are several systems. Zhang and Zheng [5] propose a recommender system for Digital TV based on fuzzy logic to infer preferences extracted from user's usage history. Blanco et al. [1] present an approach to personalized recommendation content that explores the concepts of the semantic Web to infer TV programs of interest to the user. Zhiwen, Xingshe, Yanbin and Jianhua [6] propose an approach to recommend TV programs to multiple users through the union of their users' profiles. In contrast with the PersonalTVware, the above-mentioned works fail to exploit the user's context during the recommendation process.

Lucas and Zorzo [7] propose a recommendation system for multi-user environment that considers the user's preferences discovered by the viewing history using the data mining technique to offer recommendation of TV programs. Differently from the PersonalTVware, this system does not perform inferences of contextual preferences from the user's context. Pessemier, Deryckere and Martens [8] present a proposal for a system that uses some contextual information (mood, location) explicitly obtained as extra information to filtering user-generated content on a social network site. No data mining technique about the context was adopted, and contextual information about the time was not considered as in PersonalTVware.

III. PersonalTVware Infrastructure

A. Solution Architecture

The PersonalTVware architecture consists of two subsystems: user's device and service provider. Fig. 1 outlines a detailed view of the overall architecture of PersonalTVware and their respective subsystems. In the subsystem user's device, the *Recommendation Controller module* connects a recommender system or applications to PersonalTVware modules, and coordinates the recommendation process. Besides, it is also responsible for the *Relevance Feedback* method [9] which computes interactively the *Contextual User Profiles* with instantiated genre information, and stores it in a knowledge base to be explored by inference techniques.

A *Contextual User Profile* is the result of the aggregation of the user contextual information, user personal data profile, and the genre of TV program considered relevant in a certain context. At PersonalTVware, the *Relevance Feedback* method can be performed either explicitly or implicitly. In the explicit way, users can select from the recommended TV programs those considered relevant to the context. The implicit way explores the ratio β between the time that the user watched a TV program (T_r) and its total duration (T_t) [9]. If the value obtained is greater than the threshold (50% in our experiments) the genre of the TV program will be considered relevant, and the *Contextual User Profile* is obtained.

$$\beta = T_r / T_t \in [0,1] \quad (1)$$

The *User Context Manager module* manages the access, acquisition of automatic way and conversions of the user ś contextual information such as its identity, day of the week, period of the day, location and how to access recommendation through a fixed or portable device, and the TV Program considered relevant. Such information is obtained through calls to the operating system and represented by means of metadata structures in XML Schemes and XML document.

The *User Profile Manager module* is responsible for creating e updating the user's profile through the explicit information provided by the user, for instance, the personal data (language, gender, age, occupation) and preferences (director, actor, author, keyword related to the subject of interest and a title of the TV program). The profile information will be described in accordance with the TV-Anytime metadata specifications [5], making the representation structured and aligned to the Digital TV systems. Besides, due to privacy and security reasons, the user profile is just stored in the user's device.

The *Context Interpreter module* consists of components that enable the learning and inference of contextual preferences by genres of TV programs. For that, upon request of the recommendation, PersonalTVware obtains the current *Contextual User Profile* without instantiated genre information being applied to the technique used in the inference task. Hence, from the knowledge base the interpreter returns the genres (*class*) inferred.

The PersonalTVware implements different data mining and machine learning techniques: decision tree classifier, naïve Bayesian classifier, back-propagation (a neural network), and case-based reasoning technique. The characteristic techniques are meticulously described in [4]. The interpretation or classification task can be understood as the search for a function, through supervised learning, that permits the correct associations of each record *Xi* to a single categorical label, *Yi*, named *class.*

The *Context-Based Filter module* is responsible for filtering of TV programs that are likely to be relevant to the user. The filtering process exploits the contextual information (date, time and place of origin of the TV program and user), explicit and contextual preferences (as the genre inferred by the *Context Interpreter module*), and TV programs description metadata. The content-based filtering technique was used as the recommending mechanism.

The basic philosophy of content-based filtering technique is to filter TV programs that have high similarity to user's profile. For this, the filter compares the TV program description metadata with the compatible description in the user's profile to get a personalized list of TV Programs [5].

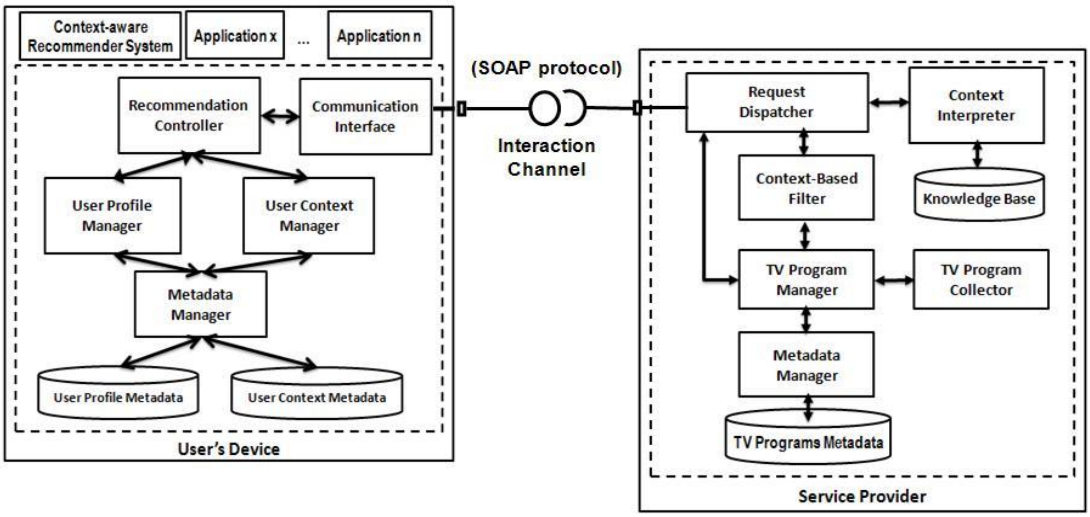

Fig. 1. PersonalTVware architecture.

Thus, through this list, user can select the TV program considered relevant (*Relevance Feedback*). This action will allow the system to learn new *Contextual User Profiles* with instantiated genre information. For reasons of limited resources of the computing devices, both the *Interpreter Context module* and *Context-Based Filter module* are located in the subsystem service provider.

The *TV Programs Manager module* manages the handling of information regarding TV programs. Such information is also described according to the TV-Anytime metadata specifications [5]. The *TV Program Collector module* is responsible for analyzing the metadata transmitted by Service Information (SI) and Web for collecting information referring TV programs. Finally, the *Communication Interface module* enables conveying information between the user's device subsystem and the service provider subsystem through a Web Service client that transmits requests via SOAP (*Simple Object Access Protocol*) over the interaction channel. In the service provider subsystem, the *Request Dispatcher module* is a Web Services-based interface that receives requests coming from the user ś device subsystem, and transmits them to the appropriate modules according to the type of the received request.

A. A Context-aware Recommendation Strategy

Assuming that the user has defined the user's profile, also that the PersonalTVware has already been trained. The recommendation process begins when the *Recommendation Controller module* receives a request from the recommender system to retrieve a TV program listing it to a particular user identified by its user identification (ID) code. The *User Context Manager module* is activated to capture and represent the contextual information of this user such as identification, location, date and time of the interaction and the type of the access device that should generate a history of the user's contexts.

Later, the *Recommendation Controller module*, having the user's ID, checks its respective information from the current context and the personal data via the *User Context Manager module* and the *User Profile Manager module*. Such information is aggregated to create the current *Contextual User profile* without instantiated genre information. Thus, the *Contextual User Profile* forwarded by the *Recommendation Controller module* through the *Communication Interface module* to *Request Dispatcher module* which should trigger the *Context Interpreter module* to infer contextual preferences for TV program genres by means of a data mining and machine learning technique selected.

After receiving the contextual preference inferred by the *Context Interpreter module*, the *Recommendation Controller module* updates the user's profile through the *User Profile Manager module* with the previously inferred preference. Later, the *Recommendation Controller module* gets the contextual and explicit preferences and aggregates them to the contextual information to send them through the *Communication Interface module* to the *Request Dispatcher module* which should trigger *Context-based Filter module.*

Finally, the *Context-based Filter module* carries out the filtering of the TV programs through filter expressions for comparing the contextual information (date, time and place of origin of the TV program and user), contextual and explicit preferences with the TV programs metadata obtained through the *TV Program Manager module*. The result of the filtering is a personalized list of TV programs sent to the *Recommendation Controller module*, and later presented to the user by the recommender system. Through this recommendation list the user can apply the *Relevance Feedback* method in an explicit way. Thus, the PersonalTVware should learn new *Contextual User Profile* with instantiated genre information.

IV. Implementation and Evaluation

A. PersonalTVware Implementation

The implementation of PersonalTVware was done in Java. In terms of Web Service technology, it was adopted Apache Axis 1.4 [10]. The interpretation and validation of XML documents are performed by a parser that uses the DOM interface (Document Object Model).

For implementation of the *Context Interpreter module* was used the Lucene framework, and the APIs of Weka tool [4]. It contemplates series of data mining and machine learning

developed in Java. The Weka uses a file in ARFF format (Attribute-Relation File Format) that contains a set of records to be used in the inference task (knowledge base with the *Contextual User Profiles*). For metadata manipulation on the server side and TV program filtering, it was used the XQuery language by mean of XML Database.

B. Case Study

The case study consisted of the development of a context-aware recommender system for digital TV, which uses the API PersonalTVware (user's device subsystem) to provide TV programs recommendations. The recommender system has been implemented as an interactive application (Xlet) based on API JavaTV [11] and run over Digital TV receivers. For the service provider subsystem, it has been installed on an application server. Fig. 2 shows the recommendation screen with a personalized list of TV programs and information about the current context of the user.

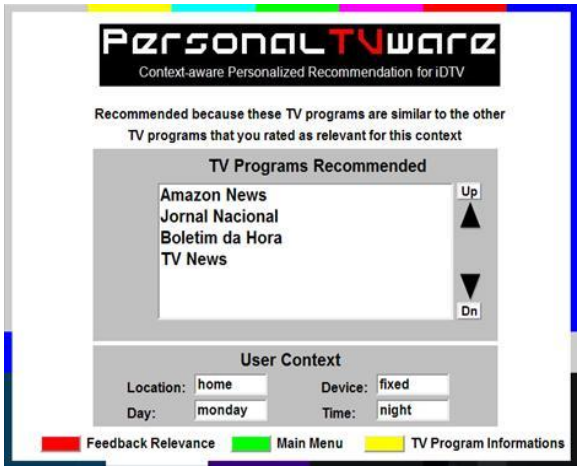

Fig. 2. Recommender system screen.

Through this screen (see Fig. 2) the user can select the TV program considered relevant (Relevance Feedback method).

Fig. 3. Detailed information about the TV program screen.

Through this screen (see Fig. 3) the user also can access detailed information (title, synopsis, date, time and channel) of each TV program.

C. Experiments and Evaluation

In order to evaluate the quality of the recommendations, a number of experiments have been conducted. The experiments were carried out with 10 users who used the system with various *Contextual User Profiles* during a four-week period. Furthermore, the experiments involved a collection of TV programs metadata describing titles, synopsis, genre, cast, among others. In total, this contained metadata for 2426 of 14 channels corresponding to one complete week's TV programming.

The test mode used was Holdout [12] which is one of the most common for assessing the efficiency of a classifier. In this method, the given data are portioned into tow independent sets, a *training set* and *test set.* Typically, two-thirds of the data are allocated to the training set, and the remaining one-third is allocated to the test set.

The training set is used to derive the predictive model (or knowledge model), the efficiency of which is estimated with the test set. Thus, after a training period of the recommender system that lasted a week, a training set (knowledge base) was generated corresponding to a total of 2197 *Contextual User Profiles* with instantiated genre information. After that, the following experiments were performed in the course of the last week.

The first experiment carried out by the users consisted of selecting the TV programs considered relevant among those that were recommended by each data mining and machine learning technique supported by PersonalTVware. The objective was to evaluate which technique provided the most appropriate recommendation to the *Contextual User Profile.*

In the second experiment, it was not employed the data mining and machine learning techniques. Therefore, only the user's profile explicitly defined was used. The objective was to evaluate the quality of the recommendations without the use of the user's context.

According to [9] the efficiency of a recommendation system can be measured by the following metrics: *precision*, *recall* and *F-measure*. The *precision* can be used to measure the ability of the system to present only relevant items. While *recall* may be used to measure the ability of the system to provide all the relevant items, the *precision* and *recall* metrics are conflicting by nature i.e., when the system has a high r*ecall* the *precision* is low and vice versa. Thus, it can be used *F-measure* in a single formula that combines the *precision* and *recall* metrics. According to [9] the closer to one is to measure the value of *F-measure* the more efficient the system.

D. Outcomes

Table I presents the averages of the metrics obtained from recommendations based on data mining and machine learning techniques and that using only the user's profile. The results indicate that the quality of the recommendations was better through case-based reasoning technique.

TABLE I: COMPARISON OF THE AVERAGE OF METRICS

Technique	Metrics		
	Precision	***Recall***	***F-measure***
C4.5 algorithm	0,861	0,620	0,680
Naive Bayesian Classifier	0,797	0,593	0,636

Technique	Metrics		
	Precision	*Recall*	*F-measure*
Multilayer Perceptron	0,774	0,582	0,606
Case-based Reasoning	0,834	0,920	0,841
User's Profile	0,511	0,174	0, 247

According to Table I, the average *F-measure* of the case-based reasoning technique is closer to 1. Therefore, the system showed superior performance as compared to the other tested methods. The reason for this result was less discrepancy between the measures of *precision* and *recall* achieved by the system.

The C4.5 algorithm showed a slightly better *precision* compared to other methods because returned fewer TV programs not relevant to the user's context, but its *recall* was less than the case-based reasoning technique, resulting in the reduction of the value of its *F-measure*. This result was due to the fact that C4.5 algorithm returns relevant TV programs, but not all relevant TV programs to the user's context returned.

Additionally, it can be observed that the quality of the recommendations was lower when only the user's profile was employed without the use of context, which shows that the lack of exploration of the context directly impacts the quality of recommendations. It was also possible to note that there were not large variations among the other learning techniques.

Fig. 4 illustrates the Recall-Precision Graph [9] which integrates *precision* and *recall*, to evaluate the system performance. In the graph, each dot is a pair of recall-precision value. The curve much closer to the upper right-hand corner of the graph (where the *recall* and *precision* are maximized) indicates the best performance.

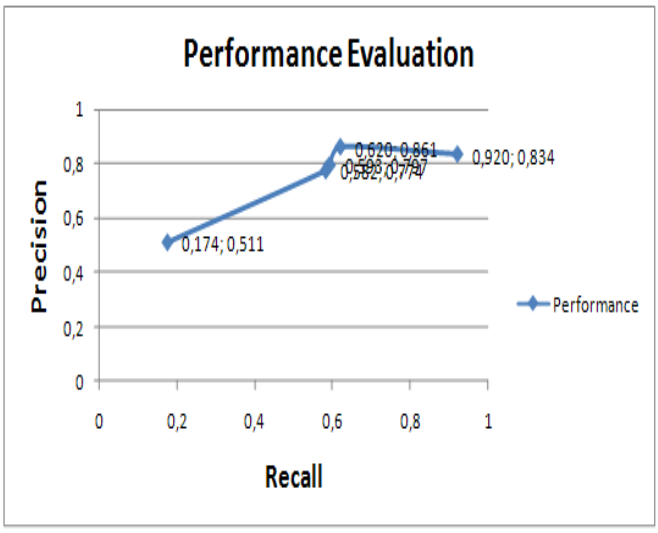

Fig. 4. Recall-precision graph.

This can be explained by the fact that the case-based reasoning technique presents the greatest number of relevant TV programs among all relevant TV programs available to the user's context. Through the graph (see Fig. 4) it can be noted that without the notion of context, the user's profile can only provide general recommendations resulting in a reduction of the system performance.

V. Conclusion

A software infrastructure entitled PersonalTVware that supports the development of some context-aware recommender systems for Interactive Digital TV has been introduced. From these experimental results, we concluded that the exploitation of both user's profile and context can improve the performance of a recommender system, especially when data mining and machine learning techniques are employed. Regarding the task of favorite's genre inference from the *Contextual User Profile*, it is worthy to note that case-based reasoning technique provides best quality recommendations.

Users who participated in the experiments evaluated positively the results of the recommendations of TV programs presented. In future work, we will implement new data mining techniques such as Support Vector Machine (SVM) [12]. Finally, we will also investigate the incorporation of other categories of context (infrastructure, system, application, domain and social) to improve recommendation quality.

References

[1] Y. Blanco, J. Pazos, M. Lopes, A. Gil, and M. Ramos, "AVATAR: An Improved Solution for Personalized TV Based on Semantic Inference," *IEEE Trans. on Consumer Electronics*, vol. 52, no. 1, pp. 223-231, February 2006.

[2] A. Gediminas and A. Tuzhilin, "Toward the Next Generation of Recommender System: A Survey of State-of the-Art and Possible Extensions," *IEEE Trans. on Knowledge and Data Engineering*, vol. 17, no.6, pp. 149-156, June 2005.

[3] A. K. Dey, "Understanding and Using Context," *ACM Personal and Ubiquitous Computing*, vol.5, no.1, pp.4-7, February 2001.

[4] I. H. Witten and E. Frank, *Data mining: Practical Machine Learning Tools and Techniques*, 2rd ed. San Francisco, C.A.: Morgan Kaufmann Publishers, 2005, ch.6.

[5] H. Zhang and S. Zheng, "Personalized TV Program Recommendation based on TV-Anytime Metadata," in *Proc. 9th Int. Symp. on Consumer Electronics (ISCE 05)*, 2005, pp. 242-246.

[6] Y. Zhiwen, Z. Xingshe, H. Yanbin, and G. Jianhua, "TV Program Recommendation for Multiple Viewers Based on User Profile Merging," *User Modeling and User-Adapted Interaction*, vol. 16, no. 1, pp.63-82, March 2006.

[7] A. Lucas and S. Zorzo, "Personalization for Digital Television Using Recommendation System Strategy," in *Proc. 16th Int. Conference System, Signals and Image*, 2009, pp. 1-4.

[8] T. D. Pessemier, T. Deryckere, and L. Martens, "Context Aware Recommendations for User-generated Content on a Social Network Site," in *Proc. 7th European Interactive Television Conference*, 2009.

[9] Y. Zhiwen and Z. Xingshe, "TV3P: An Adaptive Assistant for Personalized TV," *IEEE Trans. on Consumer Electronics*, vol.50, no. 1, pp. 393-399, February 2004.

[10] Apache Axis, "Apache Web Service Project", [Online] Available: http://ws.apache.org/axis/

[11] Sun Microsystems, "Sun JavaTV: Java Technology in Digital TV", [Online] Available: http://java.sun.com/products/javatv/

[12] J. Han and M. Kamber, *Data Mining: Concepts and Techniques*, 2rd ed. San Francisco, C.A.: Morgan Kaufmann Publishers, 2006, pp. 337-342.

Fabio Santos da Silva is a Professor at the University of the State of Amazonas (UEA), Brazil and researcher at Laboratory of Computer Architecture and Network (LARC-USP). He received his PhD in Science – Digital Systems (2011) at Polytechnic School of University of São Paulo (EP-USP), Brazil. His main research interests include Ubiquitous Computing, Context-Aware Computing, Recommender Systems, Information Retrieval, Data Mining, Interactive TV (Digital TV, IPTV, and Connected TV) and Application in Distance Education. At Genius Institute of Technology was responsible for development of middleware GINGA based on NCL/Lua and Java

technologies, interactive Digital TV Applications, mobile and Web applications.

Luiz Gustavo Pacola Alves holds a M.Sc. in Electrical Engineering and B.Sc. in Computer Engineering from Polytechnic School of University of São Paulo (EP-USP), Brazil. He has published academic papers on his main research areas which include Interactive Digital TV (metadata, middleware and applications), Context-Aware Computing and Collaborative Systems (CSCW). He worked as a researcher engineer at Laboratory of Computer Architecture and Network (LARC-USP) from 2002 to 2008. Since joining SAMSUNG Electronics in 2009 he has worked with broadcasters, consumer electronics manufacturers, software industries, universities and regulatory governmental agencies to establish technical standards for Brazilian Digital Television System, called SBTVD. Recently he has been focusing on collaboration on GINGA Test Suite specification, a collection of tests to drive the interoperability among the GINGA (SBTVD middleware based on NCL/Lua and Java technologies) implementers.

Graça Bressan got her PhD in Electronic Engineering (1986) at Polytechnic School of University of São Paulo (EP-USP), Brazil. Nowadays, at the Computer Engineering Department at EP-USP, she teaches and develops research on Computer Network, Digital TV and Performance Analysis and has published articles and has oriented Msc and PhD degree students. Her current research interests includes Computer Networks and Digital TV focusing in the aspects of Distributed Systems, Distributed Middleware, QoS Mechanisms, Collaborative Virtual Environment, Middleware for Digital TV, Video-Conferencing, Modeling and Performance Analysis of Networks, and Application in Distance Education.