DESIGN AND DEVELOPMENT OF MEDICAL RECOMMENDATION SYSTEM FOR HOME CARE SERVICE FOR GERIATRICS

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

SALİHA İREM BEŞİK

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
COMPUTER ENGINEERING

DECEMBER 2015

Approval of the thesis:

DESIGN AND DEVELOPMENT OF MEDICAL RECOMMENDATION SYSTEM FOR HOME CARE SERVICE FOR GERIATRICS

submitted by **SALİHA İREM BEŞİK** in partial fulfillment of the requirements for the degree of **Master of Science in Computer Engineering Department, Middle East Technical University** by,

Prof. Dr. M. Gülbin Dural Ünver
Dean, Graduate School of **Natural and Applied Sciences** ________________

Prof. Dr. Adnan Yazıcı
Head of Department, **Computer Engineering** ________________

Prof. Dr. Ferda Nur Alpaslan
Supervisor, **Computer Engineering Department, METU** ________________

Examining Committee Members:

Prof. Dr. Nihan Kesim Çiçekli
Computer Engineering Department, METU ________________

Prof. Dr. Ferda Nur Alpaslan
Computer Engineering Department, METU ________________

Assoc. Prof. Dr. Pınar Karagöz
Computer Engineering Department, METU ________________

Assist. Prof. Dr. Aybar Can Acar
Informatics Institute, METU ________________

Assist. Prof. Dr. Kasım Öztoprak
Computer Engineering Department, Karatay University ________________

Date: ________________

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last Name: SALİHA İREM BEŞİK

Signature :

ABSTRACT

DESIGN AND DEVELOPMENT OF MEDICAL RECOMMENDATION SYSTEM FOR HOME CARE SERVICE FOR GERIATRICS

Beşik, Saliha İrem
M.S., Department of Computer Engineering
Supervisor : Prof. Dr. Ferda Nur Alpaslan

December 2015, 99 pages

Demands and expectations for health care have gradually increased with the longer life expectancy and decline in birth rate, however the resources reserved for health services are relatively limited. The countries with aging population problems are trying to develop new systems to obtain more effective usage of current resources. The aging population and resultant chronic illnesses has become a real problem for Turkey as well. The increase in elderly population results in more demand for health care because of aging-associated physical or mental limitations and chronic illnesses. Research illustrates that home care services for seniors speed up the healing process.

The aim of the thesis is developing a medical recommendation system (RHCS) which generates treatment and care plan recommendations to assist health professionals to make decisions on treatment process of geriatrics. This developed recommendation system will be a part of an integrated patient based e-health platform which provides a home health care for those elderly people who need care, including all of the actors (particularly relatives of elderly people) involved in the nursing period.

One of the distinctive points of this study lies in the methodology used which is empowering collaborative filtering recommendation approach with historical data of geriatric patients. Its ontological-based approach, electronic health record structure, compatibility with ICD-10 and ATC clinical classification systems also makes this study prominent.

RHCS has evaluated by both offline experiments with historical patient data taken by Ankara Numune Hospital and user studies conducted with 13 doctors. The results are measured by three different types of evaluation metrics, and it is showed that in each case RHCS is a successful system to generate reliable and relevant recommendations. As a future work, RHCS will be adapted to integrate with a rule-based clinical decision support system.

Keywords: Recommendation Systems, Collaborative Filtering, Ontology, Data Mining, Similarity Measures, Feature Weighting

ÖZ

YAŞLI HASTALARIN EVDE BAKIM HİZMETİ İÇİN MEDİKAL TAVSİYE SİSTEMİ TASARIM VE GELİŞTİRİLMESİ

Beşik, Saliha İrem
Yüksek Lisans, Bilgisayar Mühendisliği Bölümü
Tez Yöneticisi : Prof. Dr. Ferda Nur Alpaslan

Aralık 2015 , 99 sayfa

Yaşam süresinin uzaması ve doğum oranındaki düşüş ile yaşlanan dünyamızda, sağlık hizmetine yönelik talep ve beklentiler her geçen gün artmakta, bununla birlikte bu hizmetlere ayrılan kaynaklar ise çok daha kısıtlı kalmaktadır. Yaşlanan nüfus sorunuyla karşı karşıya kalan ülkeler mevcut kaynağın daha etkin kullanılmasına yönelik yeni hizmet sunum modelleri geliştirmeye çalışmaktadır. Yaşlı nüfus ve ona bağlı olarak ortaya çıkan kronik hastalık yoğunluğu ülkemiz için de önemli bir gerçek haline gelmiştir. Yaşlı nüfustaki artış hem yaşlanmadan kaynaklı fiziksel/ mental kısıtlamalar ve hem de kronik hastalıklar dolayısı ile sağlık hizmetine daha talepkar bir yönelme anlamına gelmektedir. Bununla birlikte araştırmalar, yaşlının evinde aldığı bakım hizmetine daha iyi cevap verdiğini ve iyileşme sürecinin hızlandığını göstermektedir.

Bu çalışma ile sağlık personellerine tedavi ve bakım önerilerinde bulunarak geriatri hastalarına uygulanacak tedaviyi belirleme konusunda yardımcı olabilecek klinik bir tavsiye sistemi (RHCS) geliştirilmesi amaçlanmaktadır. Geliştirilen klinik tavsiye sistemi, yaşlı ve bakıma muhtaç bireylerin sağlık bakım hizmetleri sürecine evini de katabilen; yaşlının yakınları başta olmak üzere bakım sürecine dahil olan tüm aktörleri kapsayan; öğrenen ve öneren yapısı ile tedavi / bakım sürecini iyileştirme amaçlı koordinasyon ve karar destek mekanizmalarına olanak sağlayan hasta merkezli bir e-sağlık platformunun bir parçası olarak kullanılacaktır.

Bu çalışmanın ayırt edici noktalarından biri işbirlikçi filtreleme yöntemini geriatri hastalarının eski medikal kayıtlarını da kullanarak güçlendirmesidir. Ontoloji tabanlı yaklaşımı, elektronik sağlık kaydı (ESK) altyapısı, ICD-10 ve ATC klinik sınıflandırma sistemleri ile uyumluluğu da bu çalışmayı önemli hale getirmektedir.

RHCS'yi değerlendirmek için, Ankara Numune Hastanesi'nden alınan hasta verileri üzerinde çevrimdışı testler (offline experiments) ve 13 doktor ile kullanıcı araştırmaları (user studies) yapılmıştır. Sonuçlar üç farklı değerlendirme metriği kullanarak incelenmiştir ve sonuçlar RHCS'nin güvenilir ve ilgili tavsiyelerde bulunduğunu göstermektedir. İleriye dönük bir çalışma olarak; RHCS, kural-tabanlı bir karar destek sistemine entegre olarak çalışmaya uygun hale getirilecektir.

Anahtar Kelimeler: Ontoloji, Tavsiye Sistemleri, Makine Tabanlı Öğrenme, Veri Madenciliği, Özellik Ağırlıklandırma

To my beloved husband

ACKNOWLEDGMENTS

I would like to express my sincere gratitude and appreciation to my supervisor, Prof. Dr. Ferda Nur Alpaslan for her constant encouragement, support and friendship throughout this study. I have been amazingly fortunate to work with her.

I would like to thank FONET IT Solutions Inc for their interest in this study, and all opportunities they have provided at their company. This work is also supported by TÜBİTAK-TEYDEB Technology and Innovation Funding Programs (1511) (Project No: 1130007).

I would like to thank to the members of my thesis examining committee, Prof. Dr. Nihan Kesim Çiçekli, Assoc. Prof. Dr. Pınar Karagöz, Asst. Prof. Dr. Aybar Can Acar and Asst. Prof. Dr. Kasım Öztoprak for their suggestions and constructive criticism. I would like to thank also Dr. Emre Sezgin. He influenced and supported this work scientifically and his contribution were highly valuable for me.

A special gratitude and love goes to my family; my mother M. Ayşın Tanrıseven, my father Kuddusi Tanrıseven and my brother Eren Tanrıseven for their unfailing support.

I am greatly indebted to my best friend Hazal Çelik for being a wonderful friend in every way. Very special thanks also go to Aybike Şimşek Dilbaz for giving me a shoulder to lean on, whenever I need.

Above all, my deepest expression of appreciation goes to my loving husband, Hasan Tayyar Beşik for his precious love and moral support. Without him, none of my dreams for the completion of a master degree would have ever come true.

TABLE OF CONTENTS

LIST OF TABLES

TABLES

LIST OF FIGURES

FIGURES

LIST OF ABBREVIATIONS

SNOMED-CT	Systematized Nomenclature of Medicine - Clinical Terms
RHCS	Medical Recommendation System for Home Health Care Service
ICD-10	International Statistical Classification of Diseases and Related Health Problems 10th Revision
ATC	Anatomical Therapeutic Chemical
SSI	The Republic of Turkey Social Security Institution
HIBCC	The Health Industry Business Communications Council
WHO	World Health Organization

CHAPTER 1

INTRODUCTION

In this introductory chapter, first the problem is defined and the motivation behind the study is stated. Then, the contribution of this study is explained. Finally, the structure of the thesis is presented.

1.1 Problem Definition

This thesis study focuses on the design and development of a medical recommendation system for home care service for geriatric patients called RHCS. RHCS aims to help health professional by generating a treatment or care plan list as ATC codes.

1.2 Motivation

There are several motivation points for our study. Some of the key ones can be listed as follows:

1. **Growth in elderly population:** According to United Nations report presented in 2013, the population is aging in nearly all the countries of the world. Elderly population in Turkey is 5.7 million in 2012 with a proportion of 7.5% and this population will reach to 8.6 million people with a proportion of 10.2% in 2023 [1]. Although Turkey is behind other developed countries in terms of aging population [2], this increase can be considered as a reason to develop home health care systems.

2. **Increase in chronic diseases:** United States National Center for Health Statistics defined chronic disease as “one lasting 3 months or more”. Chronic diseases are more common in aged people [3]. According to Chronic Diseases Report prepared by Health Ministry of Turkey in 2006 [4] [5], approximately 22 million elderly people have at least one chronic health condition.

   Growth in chronic diseases is an important motivation behind home care health systems because of two key reasons. First reason is that chronic illnesses are by far the most leading cause of the mortality. Approximately 60% of all deaths in the world are caused due to chronic diseases [6] and 71% of deaths in Turkey are based on chronic diseases [7]. Home health care systems for geriatrics can help to prevent such deaths based on chronic diseases. Second key reason is related to cost. Diagnosing and treating chronic diseases are more costly than acute diseases. Approximately 75% of the health expenses are separated for chronic diseases in all over the world [8]. Home health care systems can provide a cost-effective solution to decide a proper treatment and care plan for geriatrics with chronic diseases.

3. **Unsatisfactoriness of current home health care system:** Current situation of home care services in Turkey are explained in detail in Chapter 2.1. As in stating in this chapter, there is no single software supported by ministry of health used for home care services. There are only some business-related software solutions [9][10]. Patient information are stored in hard copy files and there is no computer-based system. It can be come to conclusion that current home health care system does not satisfy the demands and it motivates us to develop computer-based home health care systems.

4. **Living preferences of elderly people:** According to a state planning organization study conducted in Turkey, 36% of aged people live with their relatives, 63% of them live alone and 1% live in different institutions. In urban cities, the proportion of elderly people who live alone is much more, approximately 70% [11]. Besides, research shows that home care services speed up the recovery process of patients with providing home atmosphere [12]. Within this scope, it can be deduced that there is a demand for home care services in order to provide a more qualified life for elderly people living alone.

5. **Drug usage of elderly people:** The amount of drugs used grows with the increase in age [13]. While deciding treatment plan, health professionals have a risk to miss some drugs. This risk becomes higher when amount of drugs which should be given is increased. Recommending treatment plans can help health professionals to prevent this type of risks.

1.3 Contribution of this Study

This study is significant in a number of ways. In this section, the contributions of the study from the theoretical and practical perspectives are discussed. The major areas of contribution are:

- RHCS which is a home care health service with medical recommendation system help geriatrics to remain safely at home and avoid unnecessary hospitalization. With home atmosphere, home nursing has positive impact on care / treatment process of geriatric patients. It also reduces the cost. Therefore, RHCS provides a cost-effective solution.
- RHCS helps health professionals in terms of making decisions on treatment plans and makes long-term follow-up easier with continuity of care.
- RHCS uses a collaborative filtering recommendation approach which is empowered by historical data of patients. So, it generates recommendations by considering both medical records of different patients and historical medical records of patients themselves.
- In Turkey, there is a barcode standardization in health system. RHCS follows this barcode standardization. It also follows the international standard ATC classification system to provide interaction with different systems.
- RHCS is compatible with ICD-10 coding mechanism which leads to several benefits to our system. It may lead to fewer errors in diagnoses compared to textual diagnosis data. It can be used with ontology to classify diagnoses. It increased to ability to work in tandem with other services using ICD-10 too. All in all, it improves research studies.

- RHCS can be integrated with different hospital management systems.
- RHCS is an ontology-based system to provide a common terminology with different systems, units and user groups. It makes the system more advantageous compared to other similar systems by means of interoperability, scalability and expandability.
- The verification and validation of care and treatment process is considered as a very important output for the improvement of the system. Therefore, system evaluated carefully through both offline experiments with historical patient data taken by Ankara Numune Hospital and user studies conducted with 13 doctors. Offline experiment results are evaluated by three well-known types of metrics which are precision, recall and f-measure.

1.4 The Structure of the Thesis

This thesis consists of 8 chapters. The remaining 7 chapter is structured as follows.

In **Chapter 2**, the studies conducted to determine the system requirements are described. The current situation of home health care services in Turkey is stated through a user study, related clinical home care service softwares are discussed and clinical recommendation systems in literature are analyzed in detail.

In **Chapter 3**, some important background information about clinical classification systems and terminology are explained in order to comprehend the data set used within the scope of thesis. Clinical classification systems namely ICD-10 classification and ATC classification are explained and barcode technology used in health care in Turkey is presented.

In **Chapter 4**, ontology and in particular SNOMED-CT are explained in detail. It is also described that why and how SNOMED-CT is used.

In **Chapter 5**, recommendation problem is formalized, recommendation techniques are explained with their pros and cons, and a comparison drawn between recommendation techniques is presented. Besides, some of the similarity measures in literature are explained. Finally, experimental settings used in literature and most common evaluation metrics are stated.

In **Chapter 6**, our system, RHCS, is explained in detail. This chapter is divided into four main subsections. Firstly, the system architecture is presented. Secondly, data preparation process is explained. Thirdly, it is described how to determine the similarity in RHCS. Finally, implementation of RHCS is stated.

In **Chapter 7**, the experimental results and evaluation of RHCS are demonstrated. The results by different evaluation metrics are compared and discussed.

In **Chapter 8**, the thesis is concluded and the possible future work to improve the system is addressed.

CHAPTER 2

RELATED WORK

This chapter represents the preliminary research conducted before implementing the system, RHCS. RHCS is a clinical recommendation system implemented for home-care service. Firstly, in order to fully-understand the system requirements, it is looked through the current situation of health care services in Turkey through a user experience study. Then, related home care service solutions / end products are analyzed. Finally, a detailed study on clinical recommendation systems in literature is conducted.

2.1 Current Situation of Home Care Services in Turkey

User Experience study is an important task to comprehend the problem and the requirements. An interview with home-care services health professionals worked in Ankara Numune Hospital is carried out in order to gather user requirements and understand the current situation of Home-Care Services in Turkey. The system, RHCS, is explained and their feedback and suggestions are asked. Their contributions can be listed as follow:

- In Turkey, there is no single software supported by Ministry of Health used for home care services. There are only some business related products ([9],[10]).
- The information about home care patients are stored in hand-written documents. There is no clinical data repository in electronic environment [14].
- There are eight standard forms (i.e. "Home Care Service Application Form", "Home Care Service Information Form", "Home Care Service Patient Evalua-

tion Form", "Home Care Service Patient Treatment Plan Form", "Home Care Service Medical Analysis Request Form", "Home Care Service Medical Consultation Request Form", "Home Care Service Patient Transfer Form", "Home Care Service Service Termination Form") which are used for all home-care service providers. These forms are all available online [15].

- These forms consist of valuable information for home care patients consisting demographic information, epicrisis, vital signs, complaints, addictions, physical examinations, laboratory procedures, surgical procedures, treatment plans and consultation reports.
- There is a clinical data repository for the geriatric patients who are inmates in Ankara Numune Hospital. The data in this repository can be used for our system, RHCS.
- Clinical data repository does not include all features stored in forms used for home care patients. The mutual features can be considered more important for RHCS.

2.2 Related Products

The crucial reason to work on related products is determining user needs and system requirements of our medical recommendation system for home care services. There are several national and international home care services. Some of the products / solutions are described in the following:

- **Acıbadem Mobile Chronic Healthcare Services:**
  This software [9] provides remote follow-up of patients with chronic diseases like diabetes, heart conditions, hypertension and hypotension etc. The patients who benefit from this service are able to use the devices like electrocardiography (ECG or EKG), digital scales, digital sphygmomanometers (device used to measure blood pressure) and glucometers (device to monitor glycaemia) at home. The measurement values taken from these devices are sent to data transfer module automatically and they are transferred from this module to the healthcare professionals.

- **Kardelen Home Care Service:**

  This product [10] is developed for medical home care providers and it basically helps proper professionals appoint to the treatment and/or care tasks and follow-up the patients. Besides, this solution includes a variety of features, including patient admission, establishing a diagnosis, consultation request, patient referral, presenting medical reports, treatment-purpose processes, nursing processes and appointment scheduling. It is possible to transfer data between hospitals using this service via a secure server.

- **Horizon:**

  Horizon [16] is developed by McKesson Company and it can be integrated with other products of the company. It comprises both clinical and financial operations related to home care processes. Home care providers are able to use Horizon either by setting up it with obtaining a license (on-promise) or by accessing it on web (on-demand). System stakeholders including patients, patient relatives and healthcare professionals can access information about care and treatment plan and get involved in the system according to their roles. It also includes patient education materials and a dataset for drug interactions.

- **Agencycore:**

  Agencycore [17] is a home health software developed by Axxess Technology Solutions. It enables home health agencies to manage their workflow to deliver quality care to their patients. It serves different purposes such as administration, billing, scheduling and human resources. It also contains several properties, including automatic generation of care plans from assessments, integrated drug-drug and allergy interaction check and integrated medication and diagnosis lookups.

- **AxisCare Total Homecare Management:**

  AxisCare [18] is developed in accordance with the requirements of home care agencies and it includes a marketing module that agencies can manage and/or coordinate current clients and marketing activities. It provides different facilities, including organizing workflows, scheduling appointments, patient admission, billing and salary payment. It provides a "telephony" functionality which enables health professionals to communicate patients and patient relatives. Health professionals can share the changes and/or updates with patients via SMS or voice call. AxisCare is 100% web-based.

- **Health Care First:**
  HealthcareFirst [19] is a company which works in the field of home care software solutions. HealthcareFirst home care is a web-based and mobile-based software including several functionalities and modules like scheduling of the appointments, human resources, patient follow-up and billing. It stores patient medical records which include the demographic information of the patients and care and treatment plans applied to the patients.

- **Allscripts Homecare:**
  Allscripts Homecare software [20] is developed for home care agencies to provide a quality care. It automates clinical, administrative and financial processes of both large home care organizations and small home care companies. It is a fully-integrated system comprises several modules including patient admission, scheduling appointments, planning treatments and billing. It is a web-based system and it supports industry requirements like transaction standards and code sets to work with different systems.

As the current home care services are analyzed, the common approaches listed below can be examined:

1. These systems are all web-based. It is important to make system easily accessible for the stakeholders.

2. Applications are not patient-based in general. Only a small amount of them contains patients and patient relatives (Acıbadem [9] and Horizon [16]). Others are standard automation systems which models administrative and financial affairs.

3. It can be also inferred that for home care systems being able to be integrated with different systems is prominent.

4. Almost all of them keep up with the changes in health industry (new procedures, new approaches and new technologies).

5. Some of them contains rule-based decision systems particularly for drug interactions.
6. None of them includes a clinical recommendation system.
7. Applications are generally hosted by a service provider and made available to users over a network which is Software as a Service (SaaS) software distribution model.
8. All systems keep medical records of patients which include their demographic information, their chronic diseases, their medical allergies and the drugs they used.

2.3 Related Research

In the literature, there are several studies for generating clinical recommendation system.

In "Drug-Recommendation System for Patients with Infectious Diseases" Shimadaa et. al. developed a clinical decision support system in order to recommend drugs for patients who have infectious diseases. It aimed to help health professionals particularly doctors to select a drug appropriately [21].

Meisamshabanpoor and Mahdavi studied medical decisions for disease recognition, treatment and time of period needed for recovery. In their article "Implementation of a Recommender System on Medical Recognition and Treatment", their proposed system is explained. They used classification techniques and collaborative filtering recommendation approach [22].

Duan, Street and Lu generated nursing care plan recommender system which is explained in their article "A Nursing care plan recommender system using a Data Mining Approach". They proposed a recommender system to provide a ranked list of nursing plans based on historical data and the list is updated as new items are entered. Association-rule measures (*support* and *confidence*) and a novel approach named as "information value" that expects which selections may improve the future rankings are used [23].

Hoens, Blanton and Chawla have a research on generating a reliable medical recommender system considering privacy. In their article "Reliable Medical Recommendation Systems with Patient Privacy", they explain a physician recommending system. Patients can rate physicians based on their satisfactions and the system considers these ratings to generate a recommendation. Two important features of their research are secure processing architecture and anonymous contributions architecture. Secure processing architecture provides patients to contribute encrypted ratings and the recommendations are generated over encrypted data. Anonymous contributions architecture provides patient to submit their ratings anonymously. In order to have a more reliable system, dishonest users and physicians cannot tamper with ratings. They evaluated their recommendation system in terms of reliability of recommendations and system performance [24].

Rodriguez et. al. explained their medical recommendation system SemMed in their article "SemMed: Applying Semantic Web to Medical Recommendation System". They aimed to assist health professionals by recommending possible drugs or medications by using Semantic Web Technologies. They used a ontology in OWL format with three main related classes which are diseases, allergies and medicines. The system generates drug recommendations by using information about diagnosis, drugs already taken and allergies. The recommendable drugs for a patient PM are determined by the given formula 2.1 where AM represents a set of all drugs which can be used to treat diseases of the patient, DM represents a set of drugs which associated with the patient, $IMDM$ represents a set of drugs which interacts with the currently prescribed to the patient, IMA represents a set of drugs which interacts with allergies of the patient [25].

$$PM = AM - (DM \cup IMDM \cup IMA) \tag{2.1}$$

Lim, Husain and Zakaria described their personalized recommender system in the article named "Recommender System for Personalized Wellness Therapy". Their system generated personalized wellness treatment recommendations using an Artificial Intelligence technique, hybrid case-based reasoning. They proposed an online consultation form to users. Users state their wellness concerns on consultation form and the system tries to find similar cases by case-based reasoning. If there is no suitable similar cases, the system provides recommendations by rule-based reasoning [26].

In "IAServ: An Intelligent Home Care Web Services Platform in a Cloud for Aging-in-Place", Su and Chiang described IAServ as a personalized health-care service implemented as a web service and deployed in a cloud computing setting. IAServ cannot be directly classified as a medical recommendation system, rather it is a clinical decision support system. IAServ generates personalized care plan by using the patient's ontological profile and formulated rules. [27].

Our proposed medical recommendation system is different from others in some aspects. The comparison between related studies and RHCS can be seen in Table 2.1.

Table 2.1: Comparison of RHCS and other proposed systems.

Reference	**Features**
Shimada et al., 2005 [21]	• Topic: – Drug Recommendation – To patients – Infectious Diseases • Methodology: – Decision Tree Classifier • Personalization: Yes • Ontology Usage: No

Table 2.1: Continued

<table>
<tr><th>Reference</th><th>Features</th></tr>
<tr><td>Meisamshabanpoor
and Mahdavi, 2012 [22]</td><td>• Topic:
– Diagnosis Recommendation
– Treatment Recommendation
– Prediction of length of treatment
– To Patients
• Methodology:
– Classification
– Collaborative filtering
– Pearson Correlation Coefficient
• Personalization: Yes
• Ontology Usage: No</td></tr>
<tr><td>Duan, Street and Lu,
2008 [23]</td><td>• Topic:
– Nursing care plan recommendation
– To health professionals
• Methodology:
– Association-rule measures
– "Information value"
• Personalization: No
• Ontology Usage: No</td></tr>
</table>

Table 2.1: Continued

Reference	Features
Hoens, Blanton and Chawla, 2010 [24]	• Topic: – Physician recommendation – To patients – Considering Privacy & Reliability • Methodology: – Collaborative Filtering – Encrypted data • Personalization: Yes • Ontology Usage: No
Rodriguez et al., 2009 [25]	• Topic: – Drug recommendation – To health professionals • Methodology: – Web Semantic Techniques – Ontologies • Personalization: No • Ontology Usage: Yes

Table 2.1: Continued

Reference	Features
Lim, Husain and Zakaria, 2013 [26]	• Topic: – Welness therapy recommendation – To patients • Methodology: – Hybrid case-based reasoning – Weighted Average Near Neighbour algorithm • Personalization: Yes • Ontology Usage: No
Su and Chiang, 2013 [27]	• Topic: – Intelligent Home-care Web Services – Cloud Computing • Methodology: – Decision Support System – Intelligent Agent • Personalization: Yes • Ontology Usage: Yes

Table 2.1: Continued

Reference	**Features**

RHCS	• Topic:
	  – ATC code Recommendation
	  – To health professionals
	• Methodology:
	  – Collaborative Filtering
	  – Weighted Hamming Distance
	• Personalization: No
	• Ontology Usage: Yes

CHAPTER 3

CLINICAL CLASSIFICATION AND TERMINOLOGY

In this chapter, ICD-10 classification, barcode technology in healthcare in Turkey and ATC classification system are presented. Since in the proposed system RHCS, these clinical classification systems and terminologies are used, they are explained in detail.

3.1 Barcode Technology in Healthcare in Turkey

Barcode is a unique code in the form of numbers to identify the products, namely drugs in healthcare. The European Article Numbering (EAN) Association, now known as GS1, is established as an international standards organization for barcode technology [28]. Turkey is one of the countries which uses the barcode standards set by GS1 [29]. GS1 defines different barcoding standards which are EAN-8, EAN-12, EAN-13, EAN-14 and EAN-128 for different purposes. The Health Industry Business Communications Council (HIBCC) is to facilitate barcode technology in healthcare. HIBCC is more specialized than GS1 [30]. The barcode technology principals in healthcare in Turkey can be listed as follows [29] [31]:

- All of the drugs that will be reimbursed by the Social Security Institution must have a unique barcode and register to "Republic of Turkey Medical Devices and Drugs Databank".
- Drugs can use either GS1 EAN-13 or HICC barcode standards.
- Barcodes of drugs can be started with "868", "869" or "550".

- Barcodes starting with "868" and "869" are assigned for Turkey by GS1 EAN-13.

- Barcodes of drugs starting with "550" are produced in that hospital. In general, they are used for dermatological disorders.

3.2 ATC classification system

ATC classification system is used for classifying drugs according to their active ingredients. ATC classification system is controlled by the World Health Organization Collaborating Centre for Drug Statistics Methodology (WHOCC). Activecingredients are divided into different groups "according to the organ or system on which they act and their therapeutic, pharmacological and chemical properties" [32]. Principals for ATC classification can be listed as follows [33] [32] [34]:

- Drugs are classified in groups at five different levels. As an example, the complete classifications of *metformin* and **cetirizine** are illustrated in Table 3.1.

- Plain medicinal products are products with one active substance or products which in addition to one active component contain auxiliary substances. Plain medicinal products of same active substance have same ATC code. Thus, in the ATC system, all plain *metformin* preparations are given the code *"A10BA02"* and all plain *cetirizine* preparations are given the code *"R06AE07"*.

- Medicinal products containing two or more active ingredients can be considered as combination products. As an example, Table 3.2 illustrates a combination product, *combinations of lidocaine and prilocaine*, and corresponding ATC code for this combination product.

- A medicinal product can be given more than one ATC code if it has clearly different therapeutic uses. For instance; *Acetylsalicylic acid (aspirin)* is used amongst other for pain and for cardiovascular disease. *Aspirin* is classified as cardiovascular medicine in ATC code *"B01AC06"*, for pain as ATC code *"N02BA01"*.

Table 3.1: The ATC classifications of metformin and cetirizine.

(a) The classification of metformin.

A	Alimentary tract and metabolism (1st level, anatomical main group)
A10	Drugs used in diabetes (2nd level, therapeutic subgroup)
A10B	Blood glucose lowering drugs, excl. insulins (3rd level, pharmacological subgroup)
A10BA	Biguanides (4th level, chemical subgroup)
A10BA02	Metformin (5th level, chemical substance)

(b) The classification of cetirizine.

R	Respiratory System (1st level, anatomical main group)
R06	Antihistamines for Systemic Use (2nd level, therapeutic subgroup)
R06A	Antihistamines for Systemic Use (3rd level, pharmacological subgroup)
R06AE	Piperazine derivatives (4th level, chemical subgroup)
R06AE07	Cetirizine (5th level, chemical substance)

Table 3.2: The example of ATC codes for combination product.

N01BB02	lidocaine
N01BB04	prilocaine
N01BB20	combinations of lidocaine and prilocaine

- Each medicinal product has a corresponding barcode and each barcode has a corresponding ATC code.

  In Table 3.3, barcode *"8699525092366"* is *"HITRIZIN 10 MG 10 TABLET"* and it corresponds to ATC code *"R06AE07"*.

  Barcode *"8699546080274"* corresponding to *"TALCID 500 MG 40 ÇİĞNEME TABLETİ"* and barcode *"8699546700288"* corresponding to *"TALCID SÜSPANSİYON 500 MG/5ML 200 ML"* are considered as almost the same by health professionals. They correspond to the same ATC code *"A02AD04"* and so these medicinal products are similar to each other by ATC classification system as well.

  Barcode *"8699788751406"* corresponding to *"RANIVER 50 MG/2 ML 10 AMPUL"* and barcode *"8699518750402"* corresponding to *"ULCURAN 50 MG 10 AMPUL"* are considered as same by health professionals. They correspond to the same ATC code *"A02BA02"* and so these medicinal products are similar to each other by ATC classification system as well.

Table 3.3: The example of medicinal product barcodes and corresponding ATC codes.

Barcode	**Product**	**ATC_Code**
8699525092366	HITRIZIN 10 MG 10 TABLET	R06AE07
8699546080274	TALCID 500 MG 40 ÇİĞNEME TABLETİ	A02AD04
8699546700288	TALCID SÜSPANSİYON 500 MG/5ML 200 ML	A02AD04
8699788751406	RANIVER 50 MG/2 ML 10 AMPUL	A02BA02
8699518750402	ULCURAN 50 MG 10 AMPUL	A02BA02

- In order to treat a certain disease, health professionals can use different drugs with same or similar active ingredients. Different drugs with same ingredients have similar effects on treatment. Medicinal products having ATC codes same until 3rd level. For instance, *lidocaine* with ATC code *"N01BB02"* and *prilocaine* with ATC code *"N01BB04"* can be considered as similar.

3.3 ICD-10 Classification System

The International Classification of Diseases (ICD) is a standard classification system developed by World Health Organization. It is "the standard diagnostic tool for epidemiology, health management and clinical purposes" [35]. 117 countries used ICD-10 standards and Turkey is one of these countries. ICD-10 was purchased from WHO in 1995 by the Health Project Coordination Unit of Health Ministry in Turkey and then translated into Turkish by a professional committee. ICD-10 Turkish version has been developed by including the updates of WHO until 2005. Since 2005, in health institutions affiliated to Ministry of Health, it has been obligatory to use ICD-10 [36].

Table 3.4 illustrates an example of diagnoses determined by health professionals and corresponding ICD-10 codes as diagnosis codes.

Table 3.4: The example of diagnoses and corresponding diagnosis codes as ICD-10 codes.

Diagnosis name	**Diagnosis code**
Esansiyel (primer) hipertansiyon	I10
Hipertansif kalp hastalığı	I11
Hipertansif kalp hastalığı, kalp yetmezliği (konjestif) ile birlikte	I11.0
Hipertansif kalp hastalığı, kalp yetmezliği (konjestif) olmaksızın	I11.9

CHAPTER 4

ONTOLOGY

In this chapter, the details of the ontology, SNOMED-CT, were given. The reason behind why to use SNOMED-CT and the structure of SNOMED-CT are explained in detail.

4.1 Definition of Ontology

In computer science "ontology" has a different meaning than philosophy and metaphysics. Tom Gruber, a researcher at Stanford University, defines "ontology" as "an explicit specification of conceptualization". It is used for knowledge sharing and reuse [37]. Ontology is an agreed-upon vocabulary compromising set of semantically related "concepts" in order to exchange information in a domain. Instead of creating a new ontology, SNOMED-CT is used as clinical terminology.

4.2 SNOMED-CT

Systematized Nomenclature of Medicine was created by the College of American Pathologists (CAP) in 1975. SNOMED was combined with Clinical Terms Version 3 (CTV3) in 2002. The merged product was called SNOMED Clinical Terms, which was shortened to SNOMED CT. The International Health Terminology Standards Development Organisation (IHTSDO) distributes SNOMED-CT around the world until 2007. According to United States National Library of Medicine (NIH), SNOMED-CT is one of the required standards in interoperability specifications of the United

States Healthcare Information Technology Standards Panel. SNOMED-CT is also accepted as a common global language for health terms within other IHTSDO Member countries which are Australia, Canada, Denmark, Lithuania, Sweden, the Netherlands, New Zealand, the United Kingdom and the United States [38].

Electronic Health Records (EHR) include clinical contents of patients and SNOMED-CT helps to interpret these EHRs through a standardized way. The use of SNOMED-CT benefits individuals, populations and healthcare in several ways including "reducing costly duplications and errors", "removing language barriers", "raising quality of care" and so on [39].

4.2.1 The structure of SNOMED-CT

SNOMED CT is a clinical healthcare terminology file that includes three main types of components which are concepts, descriptions and relationships [40].

- Concepts are unique clinical definitions which are organized into hierarchies. Related concepts range from general to spesific within a hierarchy.
- Descriptions are textual explanations of concepts in order to make concepts human readable.
- Relationships are links between related concepts.

Figure 4.1 illustrates general SNOMED CT design and development. Every concept has a unique identifier and concepts are organized into hierarchies by means of |is-a| relations. "Body structure", "Clinical finding", "Event" and "Substance" are some of the top-level hierarchies. |is-a|, |due to|, |causative agent|, |finding site| and |has active ingredient| are some of the types of relationships.

In order to make the usage of SNOMED CT more understandable, an example can be given [40]:

The concepts described in "Descriptions" as |bacterial pneumonia| and |viral pneumonia| both are linked with |is a| relationship to |infective pneumonia| concept. |infective pneumonia| concept has an |is a| relationship to the more general concept |pneumonia|. Hence, |bacterial pneumonia| and |viral pneumonia| concepts are also linked with general concept |pneumonia| by |is-a| relationship.

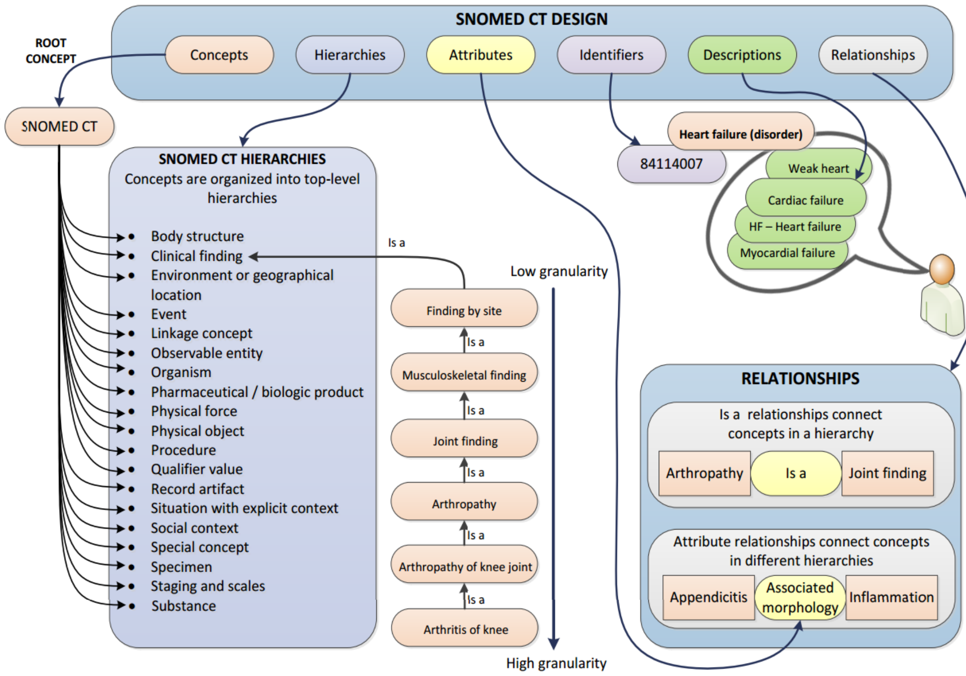

Figure 4.1: SNOMED CT Design and Development.
[40]

Within SNOMED-CT Release, it can be reached maps to other code systems and classifications including ICD-10. In ICD-10 mapping file, there are mapping between SNOMED CT concept ids and ICD10 codes. There are also SNOMED-CT descriptions of concepts and ICD descriptions as textual data.

Table 4.1 illustrates an example mapping between SNOMED-CT and ICD-10. *"Pneumonia in mycosis"* is coded as *"J17.2"* in ICD-10 classification. The corresponding SNOMED-CT concept is |*Pneumonia in aspergillosis (disorder)*| and the corresponding concept id is *"111900000"*. Different ICD-10 codes can map into same SNOMED-CT concept ids since they have different classification methodologies.

Table 4.1: Example of Mapping between SNOMED-CT and ICD-10.

Concept id	**SNOMED-CT description**	**ICD-10 code**	**ICD-10 description**
111900000	Pneumonia in aspergillosis (disorder)	J17.2	Pneumonia in mycosis
307726001	Anemia in ovarian carcinoma (clinical finding)	C56	Malignant neoplasm of ovary
307726001	Anemia in ovarian carcinoma (clinical finding)	D63.0	Anemia in neoplastic disease
50620007	Diabetic autonomic neuropathy (disorder)	E14.4	Unspecified diabetic disease with neurologic complications
50620007	Diabetic autonomic neuropathy (disorder)	G99.0	Autonomic neuropathy in endocrine disease

CHAPTER 5

RECOMMENDATION SYSTEM

In computer science, the term "recommendation (recommender) system" can be defined in many ways. Some of the possible definitions are as the following:

- "Recommender systems are software tools and techniques providing suggestions for items to be of use to a user" [41].
- "Recommender systems are personalized information filtering technology used to either predict whether a particular user will like a particular item (prediction problem) or to identify a set of N items that will be of interest to a certain user (top-N recommendation problem)" [42].

These definitions can be varied by domain specific recommendation systems. In the scope of the thesis, only medical recommendation systems are considered. By narrowing the scope, it can be had a more clear understanding for the problem and the proposed system RHCS.

Clinical databases consists of different health states of patients, like laboratory results, diagnosis codes, treatment plans and health reports. With the rapid increase in data have been collected in clinical databases, the size of search space has become dramatically large. Thus, medical recommendation systems become more important to deal with the information overload problem in clinical databases. The task of medical recommendation systems is recommending different medical information like diagnosis and treatment plans.

In this chapter, firstly, the medical recommendation problem is defined. Secondly, some of the significant problems related to recommendation approaches are stated. Thirdly, main recommendation approaches along with the limitations are explained. Finally, different similarity measures in literature are described.

5.1 Medical Recommendation Problem Definition

Our aim to develop a medical recommendation system is generating a recommendation list of care and treatment plans to health professionals. It is a guide for the health professionals and it is an input for the clinical decision support system. In literature, recommender systems can be categorized into two: Prediction problem and top-N recommendation problem. In this thesis, it is primarily focused on top-N recommendation problem rather than the prediction problem.

5.2 Problems of Recommendation Systems

In this section, some of the common problems of recommendation systems are explained.

5.2.1 Cold-start Problem

Cold-start problem (early rate problem [43], first rater problem [44]) is the problem caused by "giving recommendations to novel users who have no preference on any items" [45]. Patients not having medical records can cause cold-start problem.

5.2.2 Gray sheep Problem

Claypool et al. [43] first used the term "gray sheep". This problem refers to users, in our case patients, who do not "consistently agree or disagree with any group of people". Hence, they cannot be put into a group easily and such patients cannot "benefit from collaborative fltering" recommendation approach.

McCrae et al. [46] used the term "black sheep" to refer users, in our case patients, who are "opposite group". They are totally different from others. For such patients, it is impossible to generate recommendations by using collaborative filtering approach.

5.2.3 Stability - Plasticity Problem

Stability - plasticity problem is used commonly in artificial intelligence. It is basicly that once a system trained on a given data, it cannot learn anything new [47]. For particularly recommendation systems using content-based and collaborative filtering approaches, it is difficult to adapt them to changes in preferences [48].

5.3 Recommendation Techniques

The techniques that are used in recommendation systems are mainly divided into six categories which are non-personalized recommendation, collaborative recommendation, content-based recommendation, knowledge-based recommendation, demographic recommendation, utility-based recommendation and hybrid recommendation techniques. In this section, these recommendation system techniques are explained in detail.

5.3.1 Non-personalized Recommendation

Non-personalized recommendation is one of the simplest recommendation approaches. For each patient, the proposed recommendation is identical with the others independent from the patients. The recommendation list can be determined based on the popularity of the treatment plans. The most popular treatment plan can be defined as the most frequent one. As it is illustrated in Figure 5.1, there are six patients, three of them are treated with plan $\text{Treatment}_\text{A}$, two of them are treated with plan $\text{Treatment}_\text{B}$ and one of them is treated with plan $\text{Treatment}_\text{C}$. So, the most frequent treatment plan is $\text{Treatment}_\text{A}$. The system recommends $\text{Treatment}_\text{A}$ to the target patient Patient_N.

The advantage of this method is that it is easy to implement compared to other recommendation approaches. However, the recommendations for all users are identical and might not appeal to everyone in the system because of lack personalization [49].

5.3.2 Collaborative Recommendation

The Collaborative Filtering (CF) approach is one of the most widely used recommendation approaches. It is a process of filtering or selecting information from the dataset. The system generates recommendations to target patient based on the other similar patients. As it is illustrated in Figure 5.2, there are three patients, $\mathrm{Patient}_1$ is treated with treatment plans $\mathrm{Treatment}_A$ and $\mathrm{Treatment}_D$, $\mathrm{Patient}_2$ is treated with treatment plan $\mathrm{Treatment}_B$ and $\mathrm{Patient}_3$ is treated with treatment plans $\mathrm{Treatment}_C$ and $\mathrm{Treatment}_D$. Assume according to the similarity metric defined by the recommendation system, $\mathrm{Patient}_N$ is similar to $\mathrm{Patient}_1$. So, the system recommends the treatment plans of $\mathrm{Patient}_1$ which are $\mathrm{Treatment}_A$ and $\mathrm{Treatment}_D$ to target patient $\mathrm{Patient}_N$.

The collaborative filtering recommendation approach can be categorized in to two categories as *memory-based* and *model-based* [50].

5.3.2.1 Memory-based Collaborative Filtering

Memory-based collaborative filtering uses the entire set or a sample set of the patient-treatment plan matrix to generate a recommendation. There are two different types for memory-based collaborative filtering: *user-based collaborative filtering* and *item-based collaborative filtering*.

5.3.2.1.1 User-based Collaborative Filtering

In user-based collaborative filtering that is also known as neighborhood-based algorithm, every patient is a part of a group of similar patients (neighbors). The recommendation generated for target patient is based on the recommendation lists of the neighbor patients of the target patient. According to Hiralall and Kowalczyk [49], the general algorithm used for this approach can be summarized into three following steps:

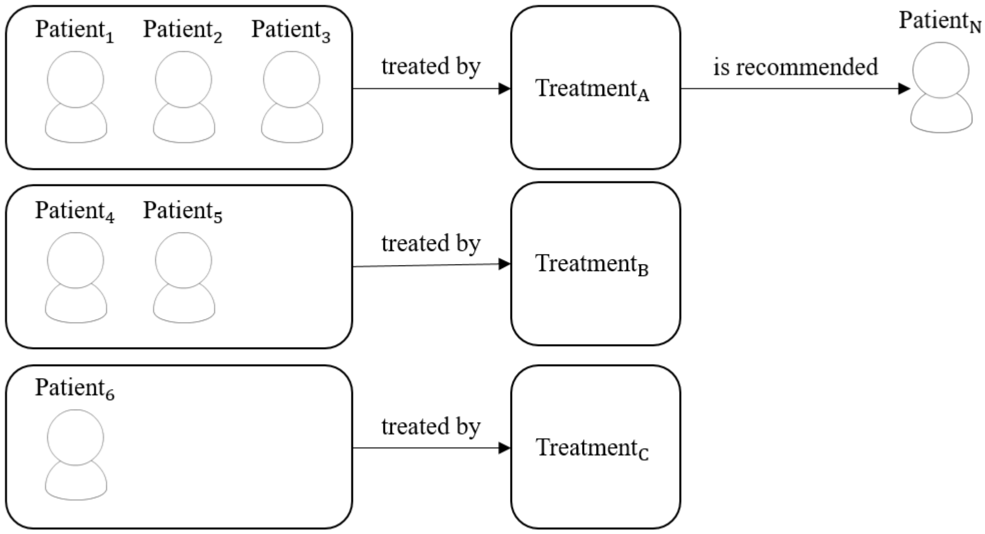

Figure 5.1: Non-personalized recommendation based on popularity.

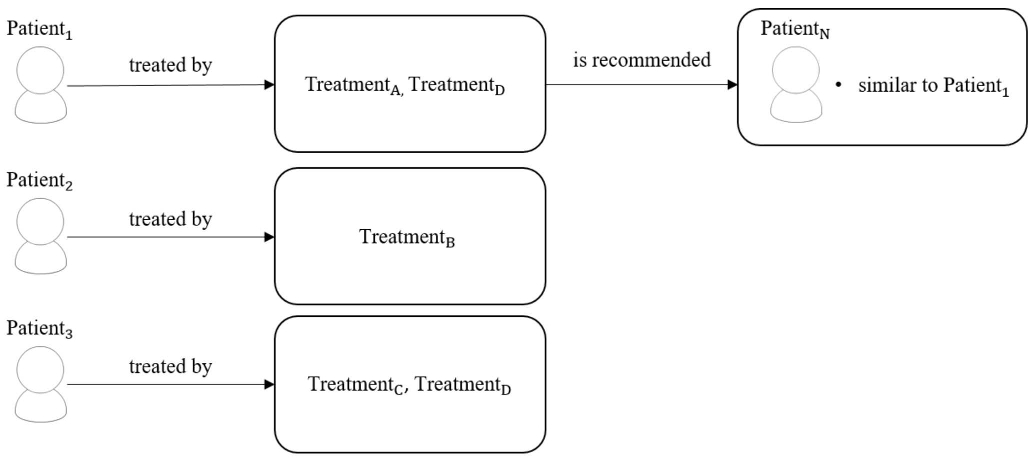

Figure 5.2: Collaborative recommendation based on user similarity.

1. Assign a weight to all patients with respect to similarity with the target patient.
2. Select k neighbor patients which have highest weights.
3. Generate a recommendation list as a weighted combination of the selected neighbors' treatment plans.

Table 5.1 summarizes how to generate the recommendation list for $\mathrm{Patient_N}$ (target patient), R_N, by means of User-based Collaborative Filtering. It is tried to find the K nearest neigbor patients which have highest similarity measures $s_{u,N}$. The value $s_{u,N}$ is a similarity measure between the patient $\mathrm{Patient_u}$ and the target patient $\mathrm{Patient_N}$. There are different similarity measures which are described in detail in Chapter 5.5. K is a predefined number which can be determined as a certain value or can be determined empirically.

Table 5.1: User Based Collaborative Filtering Algorithm to generate recommendations for $\mathrm{Patient_N}$.

$P \leftarrow$ set of all patients

$T \leftarrow$ set of all treatments for each patient.

$P = \{Patient_1, Patient_2, ..., Patient_N\}$ where $Patient_N$ is target patient.

$T = \{T_1, T_2, ..., T_N - 1\}$ where T_1 is the treatment applied for $Patient_1$.

$N \in \mathbb{R}_{>0}$ where N is the size of P.

$S \leftarrow$ set of similarity measures.

$S_{1,N} \leftarrow$ similarity measure between $Patient_1$ and $Patient_N$.

$S = \{S_{1,N}, S_{2,N}, ..., S_{N-1,N}\}$

$K \in \mathbb{R}_{>0}$ where K, a predefined number, is the size for recommendation.

$S_{highestK} = \{S_{1,N}, S_{2,N}, ..., S_{K,N}\}$ set of highest K similarity measures where $S_{1,N} > S_{2,N} > ... > S_{N-1,N} > 0$.

$R_N = \{T_1, T_2, ..., T_K\}$ recommendation list for $Patient_N$.

5.3.2.1.2 Item-based Collaborative Filtering

Linden et al. [51] proposed item-based collaborative filtering as an alternative to user-based collaborative filtering in which they match similar treatment plans, rather than matching similar patients. In this approach, similarities between pairs of treatment plans T_i and T_j can be computed with different similarity methods.

In Table 5.2, there are list of patients and lists of recommendable treatment plans. It can be thought patients as *users* and treatment plans as *items*. The values can be "0" or "1" or "?". "0" means that the patient did not treated with that treatment plan. "1" means that patient treated with treatment plan. "?" means that there is no information about that patient and treatment plan correlation. The similarity between treatment plans can be determined by using the patient-treatment plan matrix. Alternatively, domain knowledge about treatment plans can be used as well. Domain experts can generate a utility function for the similarity between treatment plans. Utility function is explained in Chapter 5.3.5.

Table 5.2: Example of Patient-Treatment Plan Matrix.

	T_1	T_2	$\cdots$	T_i	$\cdots$	T_j	$\cdots$	T_N
$Patient_1$	0	0		1		1		1
$Patient_2$	0	1		0		1		1
$\vdots$								
$Patient_L$	1	1		1		?		1
$\vdots$								
$Patient_{N-1}$	0	1		?		0		0
$Patient_N$	1	0		?		?		0

5.3.2.2 Model-based Collaborative Filtering

Memory-based collaborative filtering systems may have problems in terms of speed and scalability. Particularly for the systems generating real-time recommendations on very large datasets, memory-based collaborative filtering approach can be more problematic [49]. Model-based approach uses information to build a model to generate recommendations and there is no need to use whole dataset every time [52]. Thus, it is beneficial in terms of speed and scalability. There are different approaches to learn a model. It can be grouped these approaches into three:

1. **Probabilistic approach:** In the perspective of probabilistic approach, the collaborative filtering can be defined as calculating a probability score for a patient-

treatment plan pair, given the target patient profile or the previous scores. Bayesian networks and clustering use this approach [53].

2. **Enhancement to memory-based algorithms:** In memory-based recommendation, similarity scores for patients and/or treatment plans are calculated and used to generate recommendations. The same idea can be enhanced by using in model-based recommendation approach. The similarity scores can be stored as a model and these stored scores can be used to generate recommendations. Only some of the most similar entities can be stored so the size of the dataset can be limited. According to research conducted by Sarwar et al. [52], storing a limited number of entities does not affect so much the accuracy of the recommendations.

3. **Linear algebra problem:** Generating recommendations can be defined as performing linear algebra operations on a patient-treatment plan matrices. Singular Value Decomposition can be used to reduce the dimensionality of the dataset. According to Sarwar et al. [52], the reduced space can improve accuracy of the recommendations.

One of the advantages of collaborative filtering is that there is no need to knowledge about domain, so it is domain-independent. The memory-based approach is more advantageous than model-based approach since the implementation of the algorithm is simpler, updating the database is easier and the quality of recommendations are generally better [54]. On the other hand, the model-based approach requires low memory and CPU-time . There are several drawbacks of the approach. It has cold-start, gray sheep and the stability or plasticity problems. The quality of the recommendations is affected by the size and the quality of the dataset [48].

5.3.3 Content-based Recommendation

The main idea behind content-based filtering is that treatment plans with similar features can be recommended similarly. It requires additional information about treatment plans such as features of them and how these features related to each other to define the similarity measure. The relevance between treatment plans and patients are also needed. The relevance can be obtained by means of historical medical data of the patients. For instance, in order to recommend a treatment plan to a target patient Patient_N, the content-based recommendation system examines the similarities among the treatment plans Patient_N has treated in the past. Only the treatment plans that have a high similarity to historical medical records of Patient_N would be recommended. A simple example is illustrated in Figure 5.3. Target patient Patient_N has two historical medical records which are Treatment_A and Treatment_B. Since Treatment_D is more similar to historical records, it is recommended to Patient_N.

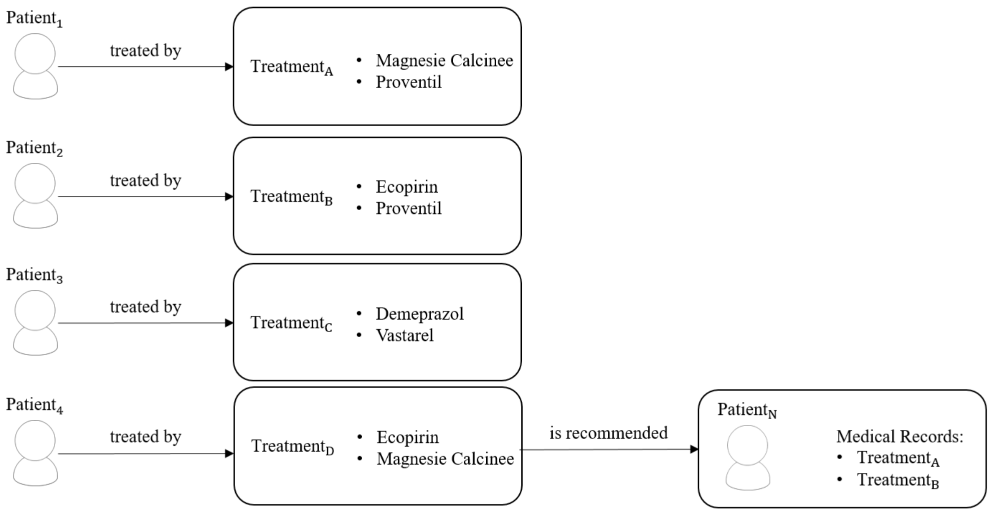

Figure 5.3: Content-based recommendation.

The medical records of the patients can be stored as vectors of keywords. The main work to classify vectors is using Information Retrieval classification techniques like Rocchio [55] and Winnow [56].

The content-based filtering approach does not require any knowledge about domain, it is domain-independent. It works well if the recommendable treatment plans are represented as a set of features properly. There are some crucial drawbacks of the approach. Assigning features to treatment plans is a hard task, the features are sometimes manually assigned. The content of the features effects the quality of the recommendation and it would be problematic. If the dataset includes unique recommendable treatment plans which are not similar to each other, the approach will be unsuccessful to generate a proper recommendation. It also suffers from the cold-start problem and the stability or plasticity problem [48].

5.3.4 Knowledge-based Recommendation

Knowledge-based recommendation systems generate recommendations based on knowledge about patients and treatment / care plans. As it is shown in Figure 5.4, recommender has a knowledge on relationship between need and/or preference of patients and possible recommendations. It makes inference on this knowledge and reasons out which recommendation meet the patient's needs. The structure of knowledge can be in any type.

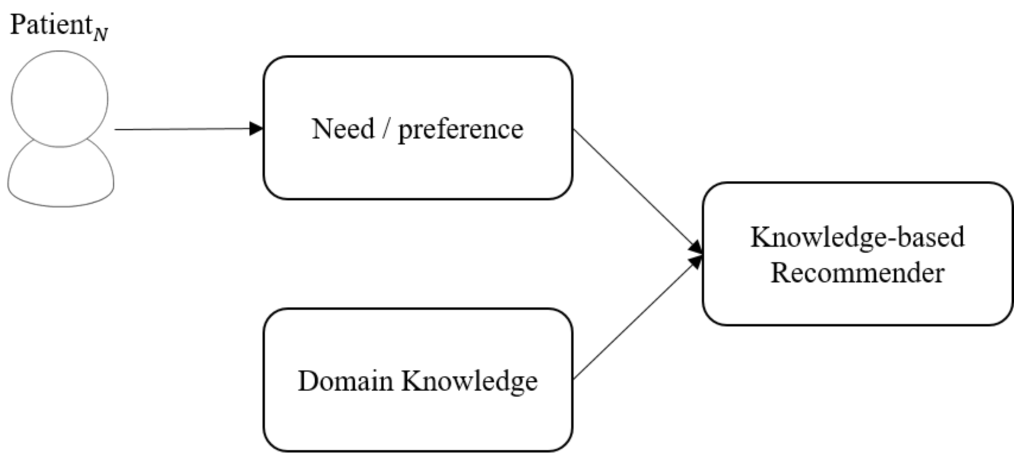

Figure 5.4: The diagram of the knowledge-based recommendation systems.

For instance, in [48], EntreeC, which is a recommender system for restaurants, uses cascaded hybrid recommendation system with knowledge based and collaborative filtering recommenders. EntreeC generates restaurant recommendations by using its knowledge base and preferences of users. EntreeC may recommend the top-rated vegetarian restaurants for a new user preferring to eat vegetarian.

As it is illustrated in Figure 5.5, there is a domain knowledge about the target patient Patient_N. Patient_N has *allergic asthma* and uses the medicine named *Proventil*. There are three different treatment plans and there are knowledge about their compatibility with different allergies and medicines. Patient_N needs a treatment plan which is compatible with *allergic asthma* and the medicine named *Proventil*. The medical recommender system looks for the treatment plans that match these needs. In this system, the treatment plan that matches the patient's need is $\text{Treatment}_\text{B}$.

Knowledge-based recommendation technique avoids cold-start and sparsity problems because it does not depends on the historical data of the patients [49]. This system does not need to store any medical records, it is solely use domain knowledge. The needs and/or preferences for each patients should be provided to generate recommendations. It is easy to make new recommendations when the needs of patients changes, it avoids "stability or plasticity" problem. The disadvantage of the knowledge-based recommendation technique is the need of domain knowledge. There are three types of domain knowledge required for a knowledge-based recommender: Catalog knowledge which is knowledge about recommendation items and their features, functional knowledge which is the knowledge about how to map patient's needs and recommendation items and user knowledge which is some knowledge about the patient. It could be hard to provide the domain knowledge for all patients. Inference part may be also difficult. Finding the best recommendation requires some knowledge engineering [48].

5.3.5 Utility-based Recommendation

Formalization of our recommendation problem can be done as follows: P is the set of all patients and T is the set of all treatments and/or care plans in the recommendation system. Both the space T, which is the set of all treatment plans, and P, which is the set of all patients, can be very large. The utility function u_t is a measure for relatedness of treatment plant t to patient p is defined as given in Equation 5.1.

$$u_t : P \times T \rightarrow R \tag{5.1}$$

R is a set to define relatedness which contains non-negative integers or real numbers within a certain range. In order to recommend a treatment plan for a target patient $t \in T$, a recommendation system tries to find such a treatment plan t' that maximizes the utility of the patient p. Therefore, for each patient $p \in P$, the system tries to recommend t' as given in Equation 5.2.

$$\forall p \in P, t' = \arg\max_{t \in T} u_t(p, t) \tag{5.2}$$

In Table 5.3, there are list of patients, lists of symptoms and the recommendable treatment plans. It can be thought patients as *users* and treatment plans as *items*. The values can be "0" or "1". "0" means that the patient do not have that symptom. "1" means that patient have that symptom. In the proposed recommendation systems, in which the utility function is represented by the score determined by the symptoms. Some of the treatment plans of the patients are missing. The recommendation system tries to make predictions on the missing treatment plans by using the utility function. The data known are manually entered by health professionals. Our aim is recommending a new treatment plan for the usage of health professionals. The problem can be described as making treatment plan recommendation for unknown patient-treatment plan pairs. For example in Table 5.3, element at the fifth row and fifth column of the matrix is "?" which means that for "$Patient_4$ the treatment plan has not entered.

Table 5.3: Example of Patient-Treatment Plan Matrix.

Patients	$Symptom_1$	$Symptom_2$	$\cdots$	$Symptom_n$	Treatment Plans
$Patient_1$	0	0		1	$Treatment_A$
$Patient_2$	1	1		1	$Treatment_B$
$Patient_3$	1	0		0	$Treatment_A$
$\vdots$					
$Patient_M$	1	0		1	?

$P = \{Patient_1, Patient_2, ..., Patient_M\}$ of patients, $T = \{T_1, T_2, ..., T_N\}$ of treatment plans.

Scores in a recommendation system can be represented by a matrix which is called patient-treatment plan matrix. In a m-by-n patient-treatment plan score matrix, m rows represent the patients and n columns represent the treatment plans. An example of matrix used for recommendation is illustrated in Table 5.4. Scores are calculated by using utility function. In Table 5.4, the score values are not calculated as a real example, they are random numbers used only for illustration reason. The recommender system recommend treatment plans to the target patient based on the calculated scores. Recommendation system recommend a list of treatment plans that have the highest scores among the calculated scores of the target patient. For example, in the given Patient-Treatment Plan Score Matrix in Table 5.4, $Patient_N$ is the target patient. First, the system calculates scores of the $Patient_N$ for each treatment plans (e.g.

$\text{Treatment}_\text{A}$, $\text{Treatment}_\text{B}$, $\text{Treatment}_\text{C}$, $\text{Treatment}_\text{D}$ and $\text{Treatment}_\text{E}$). Suppose that in our example, the number of the items in recommendation list is 2. Therefore, the recommendation list consists of $\text{Treatment}_\text{E}$ and $\text{Treatment}_\text{B}$. There can be no predefined size for the recommendation list. It can be a threshold value. Suppose that the threshold score in our example is 3. In other words, if the score is above 3, it means that the patient can be treated by the corresponding treatment plan. In this case, the recommendation list for Patient_N consists of $\text{Treatment}_\text{E}$, $\text{Treatment}_\text{B}$ and $\text{Treatment}_\text{A}$.

Table 5.4: Patient-Treatment Plan Score Matrix Example.

	$\text{Treatment}_\text{A}$	$\text{Treatment}_\text{B}$	$\text{Treatment}_\text{C}$	$\text{Treatment}_\text{D}$	$\text{Treatment}_\text{E}$
Patient_1	4.7	3.2	3.2	2.0	1.6
Patient_2	3.0	4.5	1.4	3.0	1.9
Patient_3	4.7	3.1	1.8	1.8	2.6
$\vdots$					
Patient_N	3.2	4.1	2.4	2.4	4.5

Utility-based recommendation techniques are very similar to knowledge-based techniques. In several studies [57] [49], there is no separate technique named as utility-based, they consider utility-based as knowledge-based. Both knowledge-base and utility-based recommendation systems do not generate recommendations based on generalizations about users, but rather they make inference on the match between a patient's need and the set of recommendation options available. However, knowledge-based recommendation systems and utility-based recommendation systems differ from each other. Knowledge-based recommendation systems require background knowledge of how recommendation items meet the patient's needs and descriptions of patient's needs or interests. However, utility-based recommendation systems require utility-functions over recommendation items that describe patient's needs. In utility-based technique, there is no need for knowledge engineering.

Utility-based recommendation systems make recommendations based on a computation named utility function related to each treatment plans for the patients. *Tête-à-Tête* and PersonaLogic use utility-based recommendation techniques. Each use different ways to create patient-specific utility functions [58]. The profile of patients are these patient-specific utility functions and the recommender system aims to find the best match accordingly.

Utility-based recommendation techniques avoid ramp-up (cold-start) and sparsity problem because they do not generate their recommendations on historical records of the patients. One of the most important benefits of this technique is that it can factor many different features apart from recommendation item-specific features, such as reliability of a treatment plan, availability of a medicine and delivery schedule of a medicine in order to contribute to the value of treatment. Learner-based recommendation systems (demographic, collaborative and content-based) have stability or plasticity problem, whereas utility-based recommendation systems are sensitive to changes of patient's needs. The main problem of this technique is how to create a utility function for each patient is a difficult task. It is not learner-based so its ability for suggestion is static (system does not learn)[48].

5.3.6 Demographic Recommendation

Michael Pazzani researched the demographic-based recommendation approach in his article named "A Framework for Collaborative, Content-Based and Demographic Filtering" [59]. Patients are classified according to their demographic data. The aim is to learn a "pattern" between demographic data and treatment plans. As it is illustrated in Figure 5.6, the recommendation systems tries to make recommendations based on demographic information on the gender, height, weight etc. of patients. Demographic information of $\text{Patient}_{\text{N}}$ (target patient) is similar to the demographic information of patient in the category for $\text{Treatment}_{\text{A}}$. Hence, the system recommends $\text{Treatment}_{\text{A}}$ to $\text{Patient}_{\text{N}}$.

The advantage of a demographic recommendation technique is that the system can make recommendations without any other data like symptoms and medical records. Knowledge about the treatment plans is not needed; so the technique is domain-independent. The disadvantage is that demographic classification could be erroneous. Since medical decisions depends on several different attributes, it cannot be relied on only demographic data. For example not all females whose weight and height are the same can be treated with the same plan. Patients with unusual demographic data result in low correlation coefficient with other patients. Recommendations for them are very difficult to find and they also cause odd (weird) recommendations for their

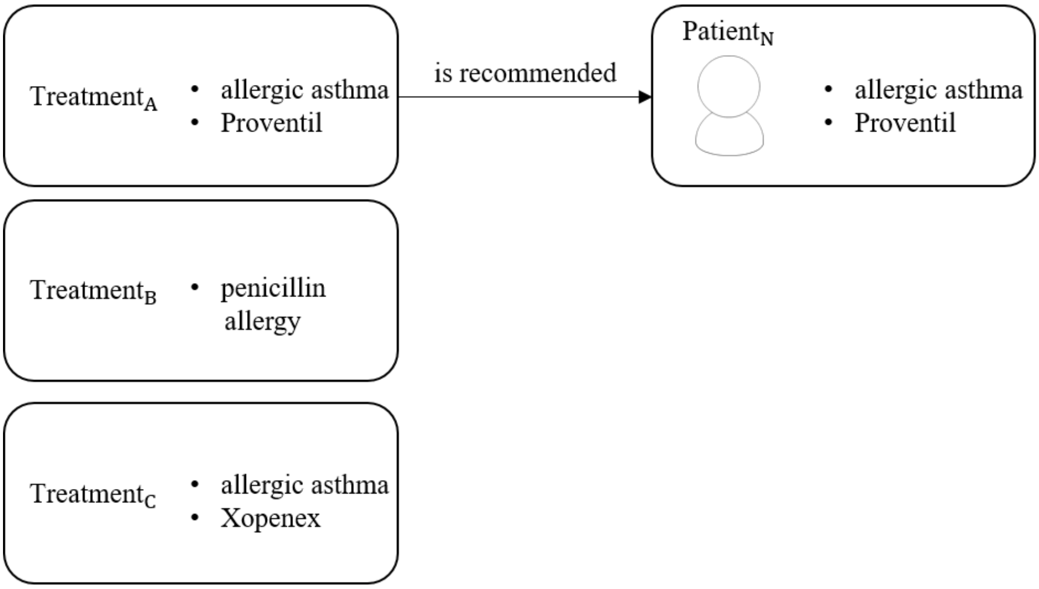

Figure 5.5: Knowledge-based recommendation.

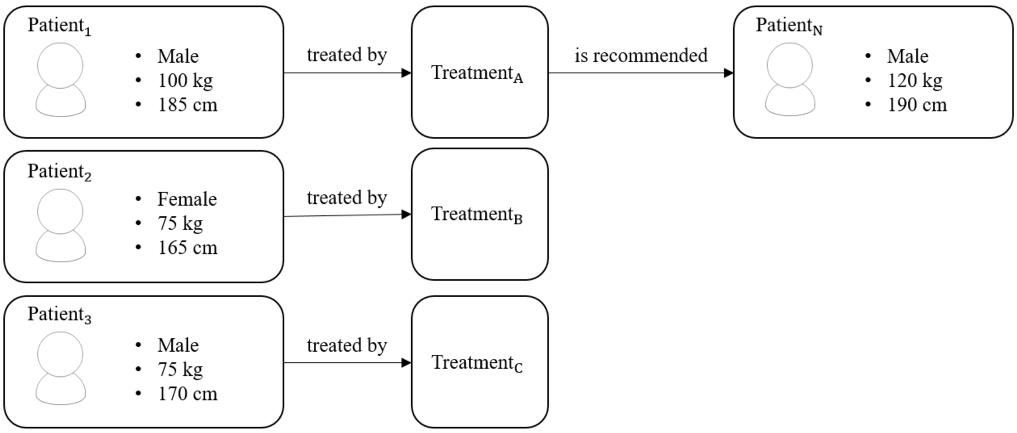

Figure 5.6: Demographic-based recommendation based on popularity.

correlated users, this problem is called the gray sheep problem and is discussed in [43]. Another challenge is the difficulty to change a created profile of a patient once the taste of the customer changes. This is called the stability or plasticity problem [48].

5.3.7 Hybrid Recommendation

Hybrid recommendation is referred as a recommendation approach that utilizes multiple recommendation approaches in order to generate recommendations. Each of the techniques described above has known shortcomings or limitations. Hybrid recommendation systems are generally implemented by combining multiple techniques to cope with the limitations of these techniques. There are mainly seven hybridization techniques which are described briefly below [60].

- **Weighted:** In this hybridization technique, the recommendation approaches used have initial weights. These weights are initially equal and changed over time with observations of the system. Evaluation results or errors can be such observations. The independent recommendation systems are combined to produce a single recommendation system according to their final weights. P-Tango [43], Pazzani's combination hybrid [59] and Towle & Quinn's hybrid system [61] are example recommendation systems which use weighted hybridization technique.

- **Switching:** In this hybridization technique, hybrid recommendation system switches among recommendation approaches used. The selection is based on a switching criterion. For instance, a switching hybrid medical recommendation system generates recommendations using different approaches for different patients by using switching criterion and analyzing the profiles of the patients. DailyLearner system [62] and Tran & Cohen's hybrid recommender system [63] are example recommendation systems which use swithcing hybridization technique.

- **Mixed:** A mixed hybrid recommendation system presents all recommendations which are generated by different recommendation systems in a single combined list. The PTV system [64] and autonomous agent ProfBuilder [65] are example recommendation systems which use mixed hybridization technique.

- **Feature Combination:** In this hybridization technique, the features derived from independent recommenders are combined to produce a single recommendation. For instance, Basu proposed a hybrid system in their article [66], this system which combines collaborative features, ratings of the users, with the content features improves the performance of pure collaborative recommendation approach in terms of precision.

  Condliff et al. [67] proposes an example recommendation system which use both feature combination and meta-level hybridization techniques in their study.

- **Feature Augmentation:** In feature augmentation hybridization technique, the output of one recommendation approach is used as a part of the input to another approach. In [68], a content boosted collaborative filtering system which uses feature augmentation hybrid approach is proposed. In this system, first content-based recommendation predicts unknown ratings. Then, collaborative filtering uses the results provided by content based recommender as input to make predictions about ratings of users.

  Libra system [69] and the GroupLens research system [70] are example recommendation systems which use feature augmentation hybridization technique.

- **Cascade:** In cascade hybridization technique, recommendation systems are strictly prioritized, one recommender which has worse results cannot modify decisions made by a stronger one, but can only refine them. After primary recommendation approach processes on data and produces an output, secondary approach can only change the score (ranking) of this output but cannot make changes on its content. Also, in cascade hybrid systems, if the prior recommender has very good results, there is no need to run the other recommendation systems.

  In [58], EntreeC, which is a recommender system for restaurants, uses cascaded hybrid recommendation system with knowledge based and collaborative filtering recommendation systems. EntreeC generates restaurant recommendations by using its knowledge base and preferences of users. Collaborative filtering is used to break the ties and further generate restaurant recommendations after knowledge-based recommender.

- **Meta-level:** In meta-level hybridization technique, one recommendation technique is used to construct some sort of model, then this model is used by other recommendation techniques as input. It is kind of similar to the feature augmentation hybridization technique, but they differ. In meta-level hybridization technique, one recommender produces a model by using learning algorithms and the other recommender uses this model. In feature augmentation hybridization technique, the output of one recommender is used as a part of an input to another recommender.

  Fab [71] [72], a document recommender, uses a meta-level hybrid system with "collaboration via content" structure. Content-based recommender constructs user profiles as vectors of weights for the terms/keywords. Collaborative filtering recommender then uses these user profiles to determine the similarity between users. Fab also uses cascade technique.

  LaboUr system [73] is another meta-level hybrid recommendation system example.

Hybrid recommendation systems are advantageous since they try to overcome the limitations of the other recommendation techniques. Hybrid recommendation systems combine two or more recommendations to have better results. The disadvantage is that such systems could be hard to implement and use.

One of the most popular hybrid system is the combination of content-based and collaborative filtering. The comparisons of the different hybridization techniques is difficult to handle. Which hybridization technique should be used depends on the characteristics of the recommendation systems being combined. A small distinction can be helpful in order to establish the trade-offs between hybridization techniques. Two cases can be set which are *uniform case* and *non-uniform case*. In uniform cases, one recommender performs better than another over the whole space of recommendation, and in non-uniform cases, recommenders have different strengths and weaknesses in different parts of the space. For the uniform cases, cascade hybridization, feature augmentation hybridization, feature combination hybridization and meta-level hybridization can be more advantageous than the others. For the non-uniform cases, switching hybridization could be more preferable.

Burke [48] [60] summarizes some of the studies in hybrid recommender system in the Figure 5.7. There are some remarks about the table. It is for personalized-recommendation techniques. Since utility-based recommendation technique is a special case of knowledge-based recommendation technique, knowledge-based and utility-based techniques are combined.

	Weighted	Mixed	Switching	Feature Combination	Cascade	Feature Aug.	Meta-level
CF/CN	P-Tango	PTV, ProfBuilder	DailyLearner	(Basu, Hirsh & Cohen 1998)	Fab	Libra	
CF/DM	(Pazzani 1999)						
CF/KB	(Towle & Quinn 2000)		(Tran & Cohen, 2000)				
CN/CF							Fab, (Condliff, et al. 1999), LaboUr
CN/DM	(Pazzani 1999)			(Condliff, et al. 1999)			
CN/KB							
DM/CF							
DM/CN							
DM/KB							
KB/CF					EntreeC	GroupLens (1999)	
KB/CN							
KB/DM							

(CF = collaborative, CN = content-based, DM = demographic, KB = knowledge-based / utility-based)

Redundant
Not possible

Figure 5.7: Example hybrid recommendation systems.
[48] [60]

The gray fields illustrate the redundant combinations. There are four order-insensitive hybridization techniques: Weighted, Mixed, Switching and Feature Combination. The order in which the recommendation techniques applied makes no difference with these hybridization approaches. For instance, a CN/CF weighted system is not different from a CF/CN one. There are 24 redundant spaces in the table.

Apart from redundant spaces, there are some combinations are not possible. For feature combination hybridization technique, knowledge-based recommendation technique does not represent a possible hybrid because knowledge-base can take into account any kind of data. The demographic recommendation technique is similar to collaborative filtering technique since they only differ in terms of the features they used. Hence, there is no need to distinguish content-based/demographic (CN/DM) meta-level hybrid from a content-based/collaborative (CN/CF) one.

5.4 Comparison of Recommendation Techniques

As it is described in Table 5.5, all recommendation approaches have some strengths and weaknesses.

Table 5.5: Comparison of Recommendation Techniques.

Technique	**Strengths**	**Weaknesses**
Non-personalized	- Easy to implement - No need to complex data	- Lack of personalization
Collaborative	- Domain-independent - Quality can be improved over time - Low CPU time for model-based	- Quality is dependent on dataset size - Cold-start - Gray sheep - Stability vs. plasticity - Lots of memory and high CPU-time for memory-based
Content-based	- Domain-independent - Quality can be improved over time	- Quality is dependent on dataset size - Cold-start - Stability vs. plasticity
Knowledge-based	- No cold-start - No need to historical data - Sensitive to changes on data	- Knowledge engineering required - Not learn
Utility-based	- No cold-start - No need to historical data - Sensitive to changes on data	- Need to formulize a utility function - Not learn

Table 5.5: Continued

Technique	**Strengths**	**Weaknesses**
Demographic	- Domain-independent - Quality can be improved over time	- Need to gather demographic information - Quality is dependent on dataset size - Cold-start - Gray sheep - Stability vs. plasticity

5.5 Similarity Measures

Determining similarity is one of the key tasks for content-based and collaborative filtering recommendation systems. In literature, there are several similarity measures. In our case, we need to find similarity between patients. Each patient can be represented as a vector. Each of the attributes of patient data can be considered as a dimension of the vector. This is neither classification nor clustering problem. Similarity is basically defined as "closeness". When two patient vectors are closer to each other, then these patients are similar. In this section, some of the prominent similarity measures which can be used in medical recommendation systems are explained.

Table 5.6: Formulas used to determine similarity.

Definitions:
$P \leftarrow$ patient vector space
$\vec{v}(p_{\mathrm{m}}) \leftarrow$ vectorial representation of $\mathrm{patient_m}$
$\vec{v}(p_{\mathrm{n}}) \leftarrow$ vectorial representation of $\mathrm{patient_n}$
$\vec{v}(p_{\mathrm{m}}) = \{m_1, m_2, ..., m_k\}$
$\vec{v}(p_{\mathrm{n}}) = \{n_1, n_2, ..., n_k\}$
$m_i \leftarrow$ the value for i^{th} attribute for $\mathrm{patient_m}$ where $0 < i < k$
$n_i \leftarrow$ the value for i^{th} attribute for $\mathrm{patient_n}$ where $0 < i < k$
$w_i \leftarrow$ weight for i^{th} attribute where $0 < i < k$
$\vec{v}(p_{\mathrm{m}}), \vec{v}(p_{\mathrm{n}}) \in P$
$0 < i < k; m_i, n_i, w_i \in \mathbb{R}_{>0}$
Formulas
$d_{Manhattan} \leftarrow$ Manhattan distance
$d_{Manhattan} = \sum_1^k \lvert m_i - n_i \rvert$
$d_{WeightedManhattan} \leftarrow$ Weighted Manhattan distance
$d_{WeightedManhattan} = \sum_1^k w_i \times \lvert m_i - n_i \rvert$
$d_{Euclidean} \leftarrow$ Euclidean distance
$d_{Euclidean} = \sqrt{\sum_1^k \lvert m_i - n_i \rvert^2}$
$d_{WeightedEuclidean} \leftarrow$ Weighted Euclidean distance
$d_{WeightedEuclidean} = \sqrt{\sum_1^k w_i \lvert m_i - n_i \rvert^2}$
$d_{SquaredEuclidean} \leftarrow$ Squared Euclidean distance
$d_{SquaredEuclidean} = \sum_1^k \lvert m_i - n_i \rvert^2$
$d_{WeightedSquaredEuclidean} \leftarrow$ Weighted Squared Euclidean distance
$d_{WeightedSquaredEuclidean} = \sum_1^k w_i \lvert m_i - n_i \rvert^2$

Table 5.6: Continued

$d_{Minkowski} \leftarrow$ Minkowski distance

$d_{Minkowski} = \sqrt{\sum_1^k |m_i - n_i|^k}$

$d_{WeightedMinkowski} \leftarrow$ Weighted Minkowski distance

$d_{WeightedMinkowski} = \sqrt[k]{\sum_1^k w_i |m_i - n_i|^k}$

$d_{Chebyshev} \leftarrow$ Chebyshev distance

$d_{Chebyshev} = \max_{1 \le i \le k} |m_i - n_i|$

$d_{Hamming} \leftarrow$ Hamming distance, # $\otimes$ is bitwise XOR;

$d_{Hamming} = \sum_1^k m_i \otimes n_i$

$d_{WeightedHamming} \leftarrow$ Weighted Hamming distance

$d_{WeightedHamming} = \sum_1^k w_i \times (m_i \otimes n_i)$

$sim_{Cosinus} \leftarrow$ Cosinus similarity

$$sim_{Cosinus} = \frac{\sum_1^k m_i \times n_i}{\sqrt{\sum_1^k m_i^2} \times \sqrt{\sum_1^k n_i^2}}$$

$sim_{WeightedCosinus} \leftarrow$ Weighted Cosinus similarity

$$sim_{WeightedCosinus} = \frac{\sum_1^k w_i \times m_i \times n_i}{\sqrt{\sum_1^k w_i \times m_i^2} \times \sqrt{\sum_1^k w_i \times n_i^2}}$$

$sim_{Jaccard} \leftarrow$ Jaccard Coefficient similarity

$$sim_{Jaccard} = \frac{\vec{v}(p_{\mathrm{m}}) \bigcap \vec{v}(p_{\mathrm{n}})}{\vec{v}(p_{\mathrm{m}}) \bigcup \vec{v}(p_{\mathrm{n}})}$$

$sim_{PearsonCorrelation} \leftarrow$ Pearson Correlation Coefficient similarity

$sim_{PearsonCorrelation} = \frac{\sum_1^k (m_i - \bar{m}) \times (n_i - \bar{n})}{\sqrt{\sum_1^k (m_i - \bar{m})^2} \times \sqrt{\sum_1^k (n_i - \bar{n}^2}}$; where $\bar{m}$ and $\bar{n}$ are respective means.

In Table 5.6, formulas for two vectors which have all non-negative real number values are used. Distance and/or similarity functions provide a way to measure how close two elements are, where elements do not have to be numbers but can also be different arbitrary objects. A typical distance for real number vectors is absolute difference.

5.6 Evaluating Recommendation Systems

Evaluation is "a structured process of assessing the success of a project in meeting its goals" [74]. After implementation process, evaluation is one of the crucial tasks in almost every study. It is needed to evaluate the algorithms used in the system, RHCS. In this chapter, first the experimental settings are introduced. Shani and Gunawardana [75] classified experiments in three which are offline experiments, user studies, and online experiments. It is also discussed how to draw conclusions from the conducted experiments by explaining some of the well-known evaluation metrics.

5.6.1 Experimental Settings

In this section, three different experimental settings that can be used for evaluation are described.

5.6.1.1 Offline Experiments

Offline experiments are performed by simulating the behavior of users interacting with the system through using historical dataset [75]. Historical dataset is a pre-collected dataset. It can be assumed that user behavior after implementing system will be similar enough to the user behavior on historical data. Hence, trustworthy conclusions based on this simulation can be drawn. Offline experiments are advantageous over other experimental settings with requiring no interaction with real users and being a cost-effective solution. However, the assumption which users' behavior when interacting with a system will be similar to the users' behavior prior to that system's deployment can be erroneous in some cases. Thus, the results by offline experiments can be insufficient and misleading for a reliable evaluation.

5.6.1.2 User Studies

User studies are research studies conducted by a set of test users in order to understand user behaviors and information needs [75]. Test users are real (end) users which interact with the system to perform several predetermined tasks. While the test users interacting with system, their behavior is observed and some quantitative measurements are collected. There can be several different quantitative measurements such as the time taken to perform each task and what portion of the tasks are completed. User studies also enable us to ask test users qualitative questions via powerful questionnaires such as whether the test user perceived the task as easy to complete, whether the test user perceive the task as understandable, or whether the test user thought the recommendations were relevant. These type of qualitative questions are very important to interpret user behavior and quantitative results.

User study is the only experimental setting enables us to collect qualitative data. This is the most important advantage of this approach.

User study has also some drawbacks:

- Collecting test users is a hard task.
- Test users can be volunteers or paid. Compensation of paid test users can be expensive.
- Test users are asked to perform several tasks. These tasks may be repeatedly performed in order to compare user behavior on successive usages and on first usage. It is almost impossible to test all possible usage scenarios and this testing process requires time. Therefore, it may be needed to conduct user studies with a small set of test users and a small set of tasks.
- It can be come to the conclusion that user studies may be disadvantageous both in terms of time and monetary value.
- There is one more prominent challenge that it should be considered whether the test users represent the real system users properly or not. Even when test users represent the true population of real users adequately, the results of experiments may still be biased as test users do not interact with system unconsciously, they

are aware that they are participating in an experiment and they may provide some misleading information.

Although there are negative aspects of user study method, it is still really beneficial to evaluate systems.

5.6.1.3 Online Experiments

Online experiments are conducted with real system users that perform real tasks. This type of experiment provides the most trustworthy results. Hence, many real world systems like Google, Microsoft, Amazon, Ebay/Paypal and Facebook employ online testing systems. There are different types of online experiments such as A/B testing, Multifactor experiments, Conditional execution and so on. A/B testing is one of the most common approaches in online experiments which is "randomly assigning real users to one of two variations of a service" [76]. Multifactor experiments are experiments including more than one factor which are evaluated independently [77]. Conditional execution are conducted when there is a dependency on a condition. For instance, when a case is valid only if another case is occurred [76].

Online experiment is superior to other types of experiments by providing more realistic inferences. In online experiments, a sample set of real users performed some real tasks. Users should be selected randomly in order to have a fair evaluation. Providing randomness is a challenging process. Other extrinsic factors like user interface and underlying algorithms which may affect user behaviors should be fixed [75].

This type of experiments cannot be conducted before system deployment and so it is risky to cause user dissatisfaction which is an undesirable case particularly for commercial systems. For these reasons, it is more acceptable to perform an online experiment after an offline study and/or user study [76].

5.6.2 Evaluation Metrics

After conducting experiments, the results are interpreted by means of some evaluation metrics. In this section, some evaluation metrics that are commonly used in literature are explained.

Sarwar et. al categorize evaluation metrics in to two [52]:

- Statistical accuracy metrics
- Decision-support accuracy metrics

Statistical accuracy metrics are known as predictive accuracy metrics. As the name implies, these metrics are used to evaluate recommenders which focus on prediction problem. Predictive accuracy metrics are used to measure how close the recommender system's predicted rating scores are to the eventual user rating scores [78]. The most commonly used ones are mean absolute error (MAE), mean squared error (MSE) and root mean squared error (RMSE).

MAE, MSE and RMSE are calculated as in Table 5.7 where N is the number of predictions, p_i is the predicted rating for item i and r_i is the eventual user rating for item i.

Table 5.7: Formula for MAE, MSE and RMSE.

$$MAE = \frac{\sum_{i=1}^{N} |p_i - r_i|}{N}$$

$$MSE = \frac{\sum_{i=1}^{N} |p_i - r_i|^2}{N}$$

$$RMSE = \frac{\sqrt{\sum_{i=1}^{N} |p_i - r_i|^2}}{N}$$

Decision-support accuracy metrics are known as classification accuracy metrics [78]. They are used to measure how effective a recommender system generates relevant or irrelevant recommendations [52]. These accuracy metrics are generally used to evaluate recommenders which focus on top-N recommendation problem. The most commonly used ones are precision, recall and f-measure.

The relevant and irrelevant recommendations generated by a recommender system can be displayed in a two-by-two *confusion matrix* as shown in Table 5.8.

Table 5.8: Confusion Matrix.

<table>
<tr><td></td><td></td><td colspan="2">Recommended</td></tr>
<tr><td></td><td></td><td>Relevant</td><td>Irrelevant</td></tr>
<tr><td rowspan="2">Actual</td><td>Relevant</td><td>True Positive (tp)</td><td>False Negative (fn)</td></tr>
<tr><td>Irrelevant</td><td>False Positive (fp)</td><td>True Negative (tn)</td></tr>
</table>

Precision (Equation 5.3) is to measure that within all recommendations how many is relevant.

$$precision = \frac{True\ positives}{True\ positives + False\ positives} \tag{5.3}$$

Recall (Equation 5.4) is to measure that within all recommendable or relevant items how many is recommended.

$$recall = \frac{True\ positives}{True\ positives + False\ negatives} \tag{5.4}$$

F-measure also known as balanced F-score or F_1 score (Equation 5.5) is the harmonic mean of precision and recall.

$$f - measure = \frac{2 \times precision \times recall}{precision + recall} \tag{5.5}$$

CHAPTER 6

RHCS: A MEDICAL RECOMMENDATION SYSTEM FOR HOME HEALTH CARE SERVICE

In this chapter, the medical recommendation system named as RHCS is explained in detail. Firstly, the system architecture is presented. Secondly, it is talked about how to prepare the data. Then, it is clarified how to determine similarity. Finally, the implementation details are stated.

6.1 System Architecture

Figure 6.1 is a sample representation of the system architecture for the medical recommendation system, RHCS.

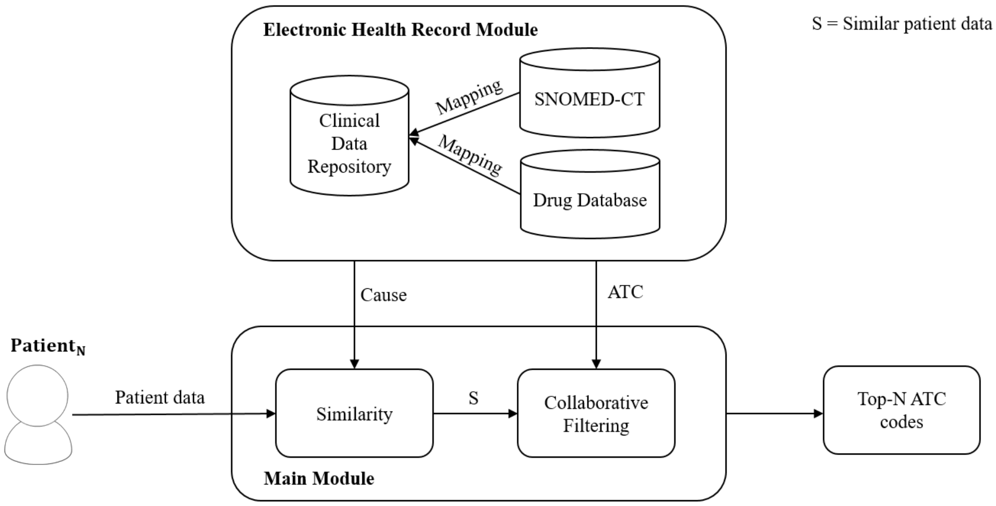

Figure 6.1: System architecture for RHCS.

There are two major modules which are "Electronic Health Record Module" and

"Main" module. Electronic Health Record Module used to prepare patient data used. Main module is used to generate recommendation list. It generates top-N treatment plans as ATC codes. It is stated how to decide the size of recommendation list (N) in Chapter 7. This decision is made empirically after some logical statements.

6.2 Data Preparation

The main dataset named "Clinical Data Repository" is a patient database of Ankara Numune Hospital taken between 10-03-2015 and 15-05-2015 which include the data of inmate patients who are older than 65 (geriatrics). Figure 6.2 is a sample representation of this clinical database.

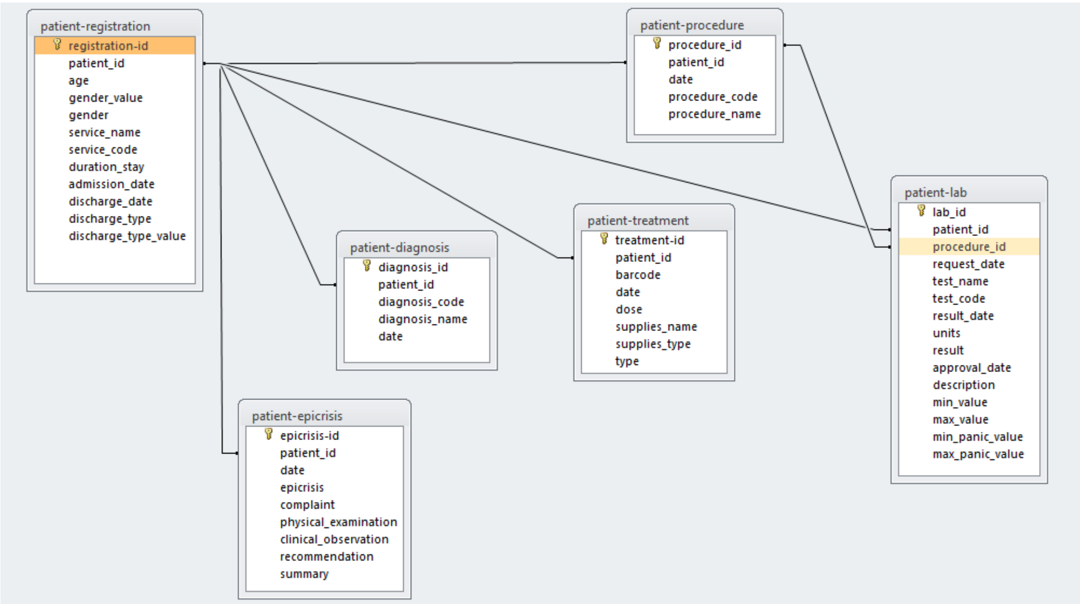

Figure 6.2: The structure of the Clinical Data Repository.

In our clinical data repository, there are six related tables. The basic information about the tables is given below:

patient-registration: This table stores general information about patients. It mainly stores patient registration id (*registration_id*), age, sex, hospital department which they admit to (*service_name*), the hospitalization duration and the type of discharge.

patient-epicrisis: This table stores epicrises and the summary of the procedures applied to patients. It is related with the table *patient-registration* on *registration_id* attribute.

patient-treatment: This table stores information about treatment plans such as drugs and drug dosages used. It is related with the table *patient-registration* on *registration_id* attribute.

patient-diagnosis: This table stores diagnosis information of patients. It is related with the table *patient-registration* on *registration_id* attribute.

patient-procedure: This table includes information about all procedures applied to the patients during their hospitalization. It is related with the table *patient-registration* on *registration_id* attribute.

patient-lab: This table includes information about laboratory procedures applied to patients during their hospitalization. Results of laboratory procedures and the numeric values used to determine whether patients fall within the normal range or not are also stored. It is related with the table *patient-registration* on *registration_id* attribute and the table *patient-procedure* on *procedure_id* attribute.

In this study, it is not used any artificial data in order to have a platform being suitable to real-life scenarios. There are two explicit important knowledge sources which are a drug database taken from SSI and a medical ontology (SNOMED-CT).

The clinical data repository is a large real-world database which has several inaccurate (noisy), incomplete and inconsistent data entries, so the data should be preprocessed. The major tasks used in data preprocessing can be seen in Figure 6.3.

The first task for data preparation is data cleaning. The second task is data integration which is used to integrate different data sources to work together. The final task is data reduction and transformation. In this task, the important attributes used for the medical recommendation system, RHCS, are selected and transformed into a more convenient format.

The data preparation process including data cleaning, data integration, and data reduction and data transformation can be explained in seven phases.

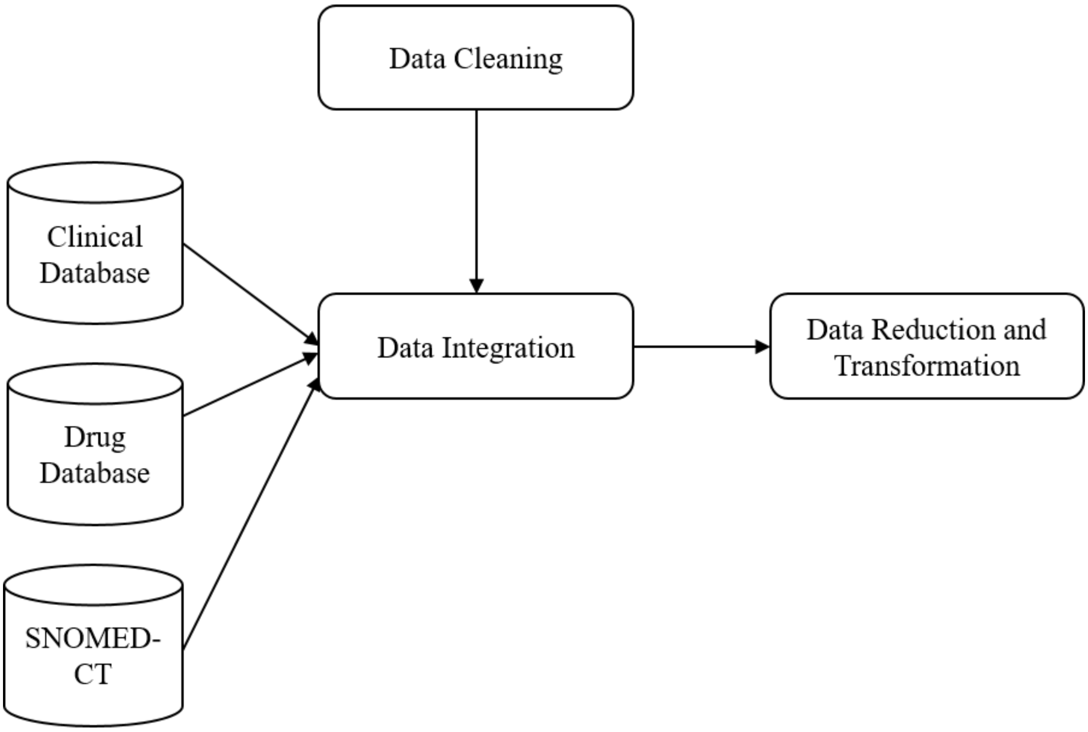

Figure 6.3: Data mining techniques used for RHCS.

6.2.1 Phase 1: Preparing patient-registration table

Ignoring the tuple with missing some prominent features is a common way to get rid of incomplete data. In our case, we ignored the tuples with missing *registration_id* which is a unique id given to each patients at their first registration. We eliminated such data since filling these features can cause some erroneous situations.

In Turkey, patients can admit to any hospital department. After physical examination, health professionals can transfer patients to any other hospital department. The service information in our dataset are related to the hospital departments where patients first admitted to. Therefore, service information is not very meaningful for our usage and we do not use these information in RHCS.

The numeric values about this phase can be seen in Table 6.1.

Table 6.1: Patient registration data information.

	Before Phase-1	**After Phase-1**
Number of patient-registration data	2866	2854

6.2.2 Phase 2: Preparing patient-epicrisis table

Epicrisis is a medical case history which is used by health professionals to diagnose. Epicrisis information includes complaints and/or syndromes of patients. Patient complaint is a proper metric for our medical recommendation system. The patient complaint data are all textual and manually entered by health professionals, so there are many misspellings and noisy data. We corrected these misspellings and grouped the similar complaints into one. From the *patient_epicrisis* table, we selected only *complaint* field. For each selected complaints, seperate fields were created. If a complaint is made by a patient, the corresponding value is entered as "1". "0" is entered, otherwise.

The numeric values about this phase can be seen in Table 6.2.

Table 6.2: Epicrisis data information.

	Before Phase-2	**After Phase-2**
Number of patient-epicrisis data	6125	6002
Number of different complaints	527	367

6.2.3 Phase 3: Preparing patient-diagnosis data

In our recommendation system RHCS, we used diagnosis_code information. We ignored the tuples with missing diagnosis_code which is an indispensable attribute for our recommendation system. There are some inconsistencies between diagnosis codes and diagnosis names. Although the names of the diagnoses are the same, the codes of them are different. Such data are also eliminated.

After data cleaning, we integrated SNOMED-CT ontology to determine the relationship between diagnoses. Diagnosis_codes are in the form of ICD-10. In order to determine the relationship between diagnoses, we need to use the clinical ontology SNOMED-CT. In SNOMED-CT, as we explained in Chapter 4, there are concepts, descriptions and relationships. In the release of SNOMED-CT, we have also SNOMED-CT to ICD-10 mapping. We converted diagnosis_codes to corresponding SNOMED-CT codes. By means of these SNOMED-CT concept codes, we can state the relationships between different diagnoses.

While joining the tables on *registration_id*, we also removed patient-registration data not including *diagnosis_code* information.

The numeric values about this phase can be seen in Table 6.3.

Table 6.3: Diagnosis data information.

	Before Phase-3	**After Phase-3**
Number of patient-diagnosis data	5088	5041
Number of different diagnosis_codes	654	343
Number of patient data	2854	2453

6.2.4 Phase 4: Preparing patient-procedures data

Patients underwent several surgical and laboratory procedures. Information about these procedures can be helpful to generate a more proper recommendation list. For the surgical procedures, we do not have any result information. We only know whether a patient underwent the surgical procedure or not. However, for laboratory procedures, we have detailed result data in *patient-lab* table. Therefore, in order to prepare the procedures data, we first looked through the list of surgical procedures. Because we used *patient-laboratory* table for the laboratory procedures, we eliminated these procedures from patient-procedure data. After eliminating laboratory procedures, we eliminated the surgical procedures which are unnecessary for our context. Table 6.4 illustrates an example of unnecessary surgical procedures, since being an intensive-care patient or not is not worth to consider for our system.

We also grouped the similar procedures into one. Table 6.5 illustrates an example of such similar surgical procedures. By having less number of procedures, we can deal with the sparsity problem.

The numeric values about this phase can be seen in Table 6.6.

For each selected surgical procedures, separate fields were created. If a surgical procedure is applied to a patient, the corresponding value is entered as "1". "0" is entered, otherwise.

Table 6.4: Example of Unnecessary Surgical Procedures taken from Clinical Data Repository.

procedure_code	**procedure_name**
552.001	Birinci Basamak Yoğun Bakım Hastası
P552001	(P*) Birinci basamak yoğun bakım hastası
P552002	(P*) İkinci basamak yoğun bakım hastası
P552003	(P*) ÜÇüncü basamak yoğun bakım hastası
704.230	Acil hemodiyaliz/yoğun bakım ve hasta başında hemodiyaliz
552.001-1	Birinci Basamak Yoğun Bakım Viziti
552.002	İkinci Basamak Yoğun Bakım Hastası
552.002-2	İkinci Basamak Yoğun Bakım Viziti
552.003	ÜÇüncü Basamak Yoğun Bakım Hastası
552.003-3	ÜÇüncü Basamak Yoğun Bakım Viziti
510.090	Yoğun bakım

Table 6.5: A Grouping Example for Similar Surgical Procedures taken from Clinical Data Repository.

procedure_code	**procedure_name**
801.750	Eklem grafisi (İki yön) mukayeseli
801.770-11	Eklem grafisi (Tek yön) tek eklem
801.770-2	Eklem grafisi (Tek yön) tek eklem
801.770-9	Eklem grafisi (Tek yön) tek eklem
801.770-8	Eklem grafisi (Tek yön) tek eklem
801.770-13	Eklem grafisi (Tek yön) tek eklem
801.780-4	Eklem grafisi (İki yön) tek eklem
801.780-7	Eklem grafisi (İki yön) tek eklem
801.780-5	Eklem grafisi (İki yön) tek eklem
801.780	Eklem grafisi (İki yön) tek eklem
801.780-3	Eklem grafisi (İki yön) tek eklem
801.780-8	Eklem grafisi (İki yön) tek eklem

Table 6.6: Procedure data information.

	Before Phase-4	**After Phase-4**
Number of patient-procedure data	24500	19884
Number of different surgical procedures	910	653

6.2.5 Phase 5: Preparing patient-laboratory data

In *patient-lab* table, there are detailed information about laboratory procedures. We eliminated some unnecessary laboratory procedures and we grouped the similar procedures into one. The numeric values about this phase can be seen in Table 6.7.

Table 6.7: Laboratory Procedure data information.

	Before Phase-5	**After Phase-5**
Number of patient-laboratory data	59766	47014
Number of different laboratory procedures	550	372

For each selected laboratory procedures, seperate fields were created. These newly created fields were filled with the numbers, from 1 to 5, according to the situation of the patients. For each laboratory procedures, there are five different fields required to evaluate the situation of patients which are *result*, *min_value*, *max_value*, *min_panic_value* and *max_panic_value*. If these fields are all numeric data and there are no missing value, Algorithm 6.1 is used to calculate the value for each corresponding laboratory procedures. For each selected laboratory procedures *p*, if it is not applied to a patient, we set "0" to the corresponding value *v*. If result (*r*) is between *minvalue* and *maxvalue*, in other words if it is in normal range, we set "1" to the corresponding value *v*. If result (*r*) is smaller than *minvalue* and bigger than *minpanicvalue*, in other words if it is in low - critical range, we set "2" to the corresponding value *v*. If result (*r*) is bigger than *maxvalue* and smaller than *maxpanicvalue*, in other words if it is in high - critical range, we set "3" to the corresponding value *v*. If result (*r*) is smaller than *minpanicvalue*, in other words if it is in low - panic range, we set "4" to the corresponding value *v*. Finally, if result (*r*) is bigger than *maxpanicvalue*, in other words if it is in high - panic range, we set "5" to the corresponding value *v*.

For some patient-laboratory correlations, some of the required fields to calculate the value for laboratory data are missing and/or they are textual data. We manually evaluated such laboratory procedures. Table 6.8 is an example for manual evaluation. In this example, there is no information about the fields which are *min_value*, *max_value*, *min_panic_value* and *max_panic_value*. We have only textual result data. "Negatif" and "-" result values mean there is no bacteria and/or infection in urinary

Algorithm 6.1 Pseudo code for the algorithm to state the values for laboratory procedures data.

$p \leftarrow$ set of selected laboratory procedures
$r \leftarrow$ result for corresponding laboratory procedures
$minvalue \leftarrow$ min_value information for corresponding laboratory procedures
$maxvalue \leftarrow$ max_value information for corresponding laboratory procedures
$minpanicvalue \leftarrow$ min_panic_value information for corresponding laboratory procedures
$maxpanicvalue \leftarrow$ max_panic_value information for corresponding laboratory procedures
$v \leftarrow$ calculated value for corresponding lab procedure
$0 \leftarrow$ N/A situation (there is no result value for the corresponding lab procedure)
$1 \leftarrow$ normal range situation (result value is in reference range)
$2 \leftarrow$ low - critical range situation (result value is not in reference range, it is less than min_value and does not go beyond the panic values)
$3 \leftarrow$ high - critical range situation (result value is not in reference range, it is more than max_value and does not go beyond the panic values)
$4 \leftarrow$ low - panic range situation (result value exceeds the min_panic_value)
$5 \leftarrow$ high - panic range situation (result value exceeds the max_panic_value)
for all p **do**
  if $r = NULL$ **then**
    $v = 0$
  else if $(r > minvalue)$ **and** $(r < maxvalue)$ **then**
    $v = 1$
  else if $(r > minpanicvalue)$ **and** $(r < minvalue)$ **then**
    $v = 2$
  else if $(r > maxvalue)$ **and** $(r < maxpanicvalue)$ **then**
    $v = 3$
  else if $(r < minpanicvalue)$ **then**
    $v = 4$
  else
    $v = 5$
  end if
end for

culture, so we set v as 1. "Positif" and "+" result values mean there is an bacteria and/or infection in urinary culture, so we set v as 3. "Berrak" and "Normal" result values mean the colour/transparency of the urine is clear and transparent, so we set v as 1. "Az Bulanık" result value means the urine has low-turbitidy, so we set v as 3. "Çok Bulanık" result value means the urine has high-turbitidy, so we set v as 5.

Table 6.8: A Manual Evaluation Example for Laboratory Procedures taken from Clinical Data Repository.

test_name	**result**
İdrar Tetkiki Tam Otomatik	Negatif
İdrar Tetkiki Tam Otomatik	Pozitif
İdrar Tetkiki Tam Otomatik	Berrak
İdrar Tetkiki Tam Otomatik	Az Bulanık
İdrar Tetkiki Tam Otomatik	Çok Bulanık
İdrar Tetkiki Tam Otomatik	Normal
İdrar Tetkiki Tam Otomatik	+
İdrar Tetkiki Tam Otomatik	-

6.2.6 Phase 6: Preparing patient-treatment data

In *patient-treatment* table, there are information about drugs used. First, we eliminated the data without barcode information. Chapter 3.1 consists of detailed information about barcode system in Turkey.

In clinical data repository, there are information about both drugs and consumable materials used in treatment phase. We only used the drugs, therefore we cleaned the consumable material data.

Recommending name of drugs is not a proper way for medical recommendation systems, because there are many equivalent drugs. Although the equivalent drugs have the same active ingredients, they have different names. Instead of recommending drug names, we recommended the active ingredients of drugs. In order to determine the active ingredients of drugs, we used a drug database taken from Social Security Institution (SSI). In drug database, there are drug barcodes and the corresponding active ingredients. We integrated drug database as an additional data source. The barcode information we had were mapped with active ingredients in drug database.

We created a field named *ingredient* to store these mapped active ingredients.

The numeric values about this phase can be seen in Table 6.9.

Table 6.9: Treatment data information.

	Before Phase-6	After Phase-6
Number of patient-treatment data	199999	129856
Number of different drugs	1628	1157
Number of different ATC codes	-	197

6.2.7 Phase 7: Transformation phase.

Transformation phase is one of the most challenging and time-consuming parts of our data preparation process. In order to understand why do we need to transform data, we have to fully comprehend our data and its weaknesses. After each of these six phases, we have almost prepared (clean, meaningful and structured) data. For each patient, we have information about complaints, diagnoses, surgical and laboratory procedures and the active ingredients of the drugs (ingredient) used.

Assume we use collaborative filtering algorithm and want to generate a top-N recommendation list to a target patient ($Patient_N$) by using the data provided after these six phases. Then, it is generally enough to find only few similar patients to that target patient. It is because the average number of drugs used for each patient is approximately 6 and N should be selected as near to this average number in order to have better evaluation results. This situation restricts RHCS to generate diverse treatment recommendations and also decrease the success of RHCS. Hence, we need to transform our data into a more convenient format.

We know all drugs given to patients during their stays in hospital, however we do not know the direct reason behind usage of these drugs. The reason could be based on a diagnosis or a complaint or the result of a laboratory procedure. In order to determine the reason to use of drugs, we used different sources which are clinical diagnosis and treatment guidelines and books [79][80][81][82][83][84]. By referring these guidelines and books, we manually determined the causes (complaints or diagnoses or laboratory procedures) which can be correlated to each drug used for each cause.

In these clinical guidelines and books, the process to decide treatment plans is directly related to make a diagnosis. Even in many cases, diagnosis is enough to make a decision on treatment plans. The problematic part is that, in our database, health professionals entered only one major diagnosis to each patient. Other secondary ilnesses and causes are not entered as diagnoses. However, in this transformation phase, we created one attribute; namely *"cause"*; by transforming diagnosis, complaints and laboratory procedures and *"cause"* attribute can be considered like "diagnosis".

Table 6.10 is an illustrative example for patient data before transformation phase. In this example, a patient have complaints (Hipertansiyon, Halsizlik, Ateş, Öksürük, Burun Tıkanıklığı), is diagnosed as Akut Sinüzit, have laboratory results being outside of reference range (Glikoz-3 and Albumin-2) which 3 corresponds to high-critical range and 2 corresponds to low-critical range and have used drugs which are listed according to their ATC codes. ATC codes are listed according to alphabetical order, there is no other ordering mechanism used before transformation phase.

Table 6.10: An Example of Patient Data Before Transformation Phase.

Complaints	**Diagnosis**	**Lab - Value**	**ATC**
Hipertansiyon, Halsizlik, Ateş, Öksürük, Burun Tıkanıklığı	Akut Sinüzit	Glikoz - 3, Albumin - 2	A10A, B05A, C09C, J01C, J01E, J01F, M01A, N02B, R01A, R05C, R05D

In transformation phase, we manually determined the direct correlation between causes and ATC codes. We also calculated priority score (%) for ATC codes. Table 6.11 illustrates the patient data used in Table 6.10 after transformation phase. In this example, complaints, diagnosis and laboratory procedures are all considered as causes. We tried to find the correlation between causes and ATC codes. Some of the causes do not have a corresponding ATC code and some of the causes can have more than one corresponding ATC codes. There can be some overlapping conditions as well, for instance *"R05C"* and *"R05D"* ATC codes can be used for both "Öksürük" and "Akut Sinüzit".

The numeric values about this phase can be seen in Table 6.12.

Table 6.11: An Example of Patient Data After Transformation Phase.

Cause	**ATC**	**Priority** (%)
Akut Sinüzit	J01C	70
	J01E	30
	J01F	10
	M01A	20
	R01A	60
	R05C	60
	R05D	40
Albumin düşüklüğü	B05A	90
Ateş	N02B	90
Burun Tıkanıklığı	R01A	95
Glikoz yüksekliği	A10A	30
Hipertansiyon	C09C	20
Öksürük	R05C	70
	R05D	70

Table 6.12: Transformation phase numeric information.

	Before Phase-7	**After Phase-7**
Number of different causes	-	561
Number of different ATC codes	-	197

Priority score (Equation 6.1) is measured according to frequency, basically. In the example table Table 6.11, for "Akut Sinüzit", there are 7 different ATC codes used as treatment plans. For each of these ATC codes, priority scores are calculated. For instance, the priority score of "J01C" ATC code is to measure that within all patients diagnosed as "Akut Sinüzit", how many uses drugs with "J01C" ATC code.

$$priority - score_{d,ATC} = \frac{\#\ of\ patients\ with\ d\ using\ ATC}{\#\ of\ patients\ with\ d} \tag{6.1}$$

In priority-score calculation, there is an exceptional case. Some ATC codes can be used for more than one cause. For example "R05C" ATC code is used for both "Öksürük" complaint and "Akut Sinüzit" diagnosis. First, we decided the main cause correlated with this ATC code. In this case, main cause in order to use "R05C" is "Öksürük" complaint. Hence, $priority - score_{Oksuruk,R05C}$ is calculated as in Equation 6.1. However, for "Akut Sinüzit" diagnosis, the calculation differs. In such exceptional cases, the priority score of ATC code ATC for cause d is calculated as in Equation 6.2 where c is main cause to use ATC. For instance, for $priority-score_{Akut\ Sinuzit,R05C}$, c is "Öksürük".

$$priority - score_{d,ATC} = \frac{\#\ of\ patients\ with\ d\ not\ having\ c\ who\ use\ ATC}{\#\ of\ patients\ with\ d\ not\ having\ c} \quad (6.2)$$

This transformation phase is prominent in order to determine the purpose of usage of these drugs. After this phase, we have some causes and their corresponding ATC values independent of patients' other data.

6.3 Similarity Measures

Finding similarity is one of the major tasks for content-based and collaborative filtering recommendation systems. In our medical recommendation system RHCS, we used collaborative filtering recommendation approach. We explained the details about our recommendation approaches in Implementation part (Chapter 6.4). In clinical recommendation systems using collaborative filtering technique, we need to measure similarities between patients (*users*) in the clinical data repository. In this section, the data processing we did and the similarity measure used in our medical recommendation system are explained in detail.

6.3.1 Data Processing

First, we have studied on how to determine the similarity measure used in RHCS. As in clarified in Data Preparation section (Chapter 6.2), we have several causes of different illnesses and corresponding treatment plans as ATC codes to these causes. Such entries can be considered as "patient entries" since these cause-ATC mappings are belongs to patients and more than one entry can be related to a patient.

Our aim is finding similarity between target patient ($Patient_N$) and different patient entries in database. The data structure of target patient ($Patient_N$) is different from other patient entries in database because target patient data is like patient data before Phase 7 which is transformation phase.

We represented each patient entry in database as vector and clinical data repository as a common vector space. Table 6.13 is used to define the vectorial representation of patient entries in our database.

Table 6.13: Pseudo code for the algorithm to define vectorial representation for patient entries in database.

$P \leftarrow$ patient vector space $\vec{v}(p_{\mathrm{m}}) \leftarrow$ vectorial representation of patient-entry$_{\mathrm{m}}$ $\vec{v}(p_{\mathrm{m}}) \in P$ $diagnosis\_code_{\mathrm{m}} \leftarrow$ corresponding SNOMED-CT concept code if cause information of patient-entry$_{\mathrm{m}}$ is related to diagnosis. "null" if cause is not diagnosis-related. $complaint_{\mathrm{i_m}} \leftarrow$ the information about patient-entry$_{\mathrm{m}}$ whether it is related to complaint$_{\mathrm{i}}$ or not. $lab_{\mathrm{i_n}} \leftarrow$ corresponding value information if cause information of patient-entry$_{\mathrm{m}}$ is related to laboratory procedure lab$_{\mathrm{i}}$. "0" if cause is not related. $diagnosis\_code \leftarrow$ string $complaint_{\mathrm{i}} \in 0, 1$, where $0 < i < 368$ $procedure_{\mathrm{i}} = 0$, where $0 < i < 654$ $lab_{\mathrm{i}} \in 0, 1, 2, 3, 4, 5$, where $0 < i < 373$ $\vec{v}(p_{\mathrm{m}}) = \{diagnosis\_code_{\mathrm{m}}, complaint_{1_{\mathrm{m}}}, ..., complaint_{367_{\mathrm{m}}}, procedure_{1_{\mathrm{m}}}, ..., procedure_{653_{\mathrm{m}}}, lab_{1_{\mathrm{m}}}, ..., lab_{372_{\mathrm{m}}}\}$

We also represented target patient ($Patient_N$) as vector. Table 6.14 shows how to define the vectorial representation of target patient.

Vectorial representation makes our similarity problem more understandable and simpler. Since our aim is recommending a treatment plan, namely active ingredients of the drugs, ingredient attribute is our *class*. We use diagnosis, complaint and lab as *attributes* to determine the similarity. Surgical procedures are not used to generate recommendations, they are used to inform health professionals.

Table 6.14: Pseudo code for the algorithm to define target patient vector.

$\vec{v}(p_{\mathrm{n}}) \leftarrow$ vectorial representation of target patient $\mathrm{Patient_N}$

$diagnosis\_code_{\mathrm{n}} \leftarrow$ corresponding SNOMED-CT concept code for diagnosis code of $\mathrm{Patient_N}$

$complaint_{\mathrm{i_n}} \leftarrow$ the information about $\mathrm{Patient_N}$ whether s/he has $\mathrm{complaint_i}$ or not.

$procedure_{\mathrm{i_n}} \leftarrow$ the information about $\mathrm{Patient_N}$ whether s/he was underwent to surgical procedure $\mathrm{procedure_i}$ or not.

$lab_{\mathrm{i_n}} \leftarrow$ the result of $\mathrm{Patient_N}$ for laboratory procedure $\mathrm{lab_i}$.

$diagnosis\_code_{\mathrm{n}} \leftarrow$ string

$complaint_{\mathrm{i_n}} \in 0, 1$, where $0 < i < 368$

$procedure_{\mathrm{i_n}} \in 0, 1$, where $0 < i < 654$

$lab_{\mathrm{i_n}} \in 0, 1, 2, 3, 4, 5$, where $0 < i < 373$

$\vec{v}(p_{\mathrm{n}}) = \{diagnosis\_code_{\mathrm{n}}, complaint_{1_{\mathrm{n}}}, ..., complaint_{367_{\mathrm{n}}}, procedure_{1_{\mathrm{n}}}, ..., procedure_{653_{\mathrm{n}}}, lab_{1_{\mathrm{n}}}, ..., lab_{372_{\mathrm{n}}}\}$

Patient-entry vectors are multidimensional vectors, but only one of these dimensions is used. For instance, if cause is related to diagnosis, $diagnosis_code$ is the only dimension which has a value different than "0" or "null".

Target patient vector is a multidimensional vector as well. Each of its attributes can be considered as a dimension of the vector.

6.3.2 Algorithms

In Table 5.6, different similarity measures are formulated. These measures are for vectors which have all non-negative real number values. In our case, patient vectors have non-numeric attribute values thus instead of using absolute difference metric, we used a different metric. In order to calculate distance between patient entry $Patient_m$ and target patient ($Patient_n$), we generated a specialized weighted Hamming distance measure as given in Equation 6.3.

$$d - Weighted\ Hamming_{m,n} = (w_1 \times \phi_{mn}) + \sum_{i=2}^{k} w_i \times (n_i \otimes m_i) \tag{6.3}$$

where m_i is the value for i^{th} attribute for patient_m, n_i is the value for i^{th} attribute for patient_n and w_i is the weight for i^{th} attribute. ϕ_{mn} is a distance value used in order to determine distance between diagnosis codes of patient_m and patient_n. ϕ_{mn} is "1" if diagnosis codes are the same. ϕ_{mn} is "0.5" if diagnosis codes are related to each other. We used SNOMED-CT ontology to decide whether diagnosis codes are related to each other or not. Diagnosis codes are all SNOMED-CT concept identifiers (IDs). If there is a relation between two diagnosis codes stated in "Relationship" table in SNOMED-CT, these two diagnosis codes are considered as related to each other. If not, they are different and the value of ϕ_{mn} is "0".

The similarity, $sim_{WeightedHamming}$, between target patient (Patient_n), $\vec{v}(p_\text{n})$ and the patient entry in database (patient $_\text{m}$), $\vec{v}(p_\text{m})$, is calculated by the formula given in Equation 6.4.

$$sim - Weighted\ Hamming_{m,n} = \frac{1}{d - Weighted\ Hamming_{m,n}} \tag{6.4}$$

,

Similarity is measured according to three major attributes which are diagnosis, complaints and laboratory procedures. We have to determine whether these attributes are equally important or not. If not, we also have to find a way that how determine the importance of these attributes. There is no straight-forward way to do it. In Equation 6.3, w_i is used as the weight for i^{th} attribute of target patient. Weight can be considered as importance of the attribute.

A metric related to frequency score was used to determine the weights of each attribute. This metric is named as "priority score" and the formula is given in Equation 6.1 and Equation 6.2. We calculated "priority score" for each cause. Weights w_i are equal to this priority score.

6.4 Implementation

Our RHCS system takes target patient data as input and generates a treatment plan list as ATC codes accordingly.

In Chapter 5, different recommendation approaches were explained with their pros and cons. We used collaborative recommendation technique which is commonly used in clinical recommendation systems in our RHCS system. The reasons behind the usage of collaborative filtering are related both our dataset and our aim of this study. These reasons can be listed as follows:

- Our dataset is not appropriate for the usage of demographic, knowledge-based, utility and content-based recommendation techniques.
- Non-personalized recommendation technique is very simple technique and it is not proper for our aim. It generates recommendation lists independent from patient data.

We used user-based collaborative-filtering recommendation technique. The general algorithm used for this approach can be summarized into three following steps:

1. Group patient entries into three (diagnosis or complaint or laboratory procedure) according to their attributes.
2. For each patient, measure similarity with the target patient.
3. Until we have "K" patient entries,
   (a) For each group, select patient entries with the highest similarity scores.
4. If size of selected patient entries bigger than "K", remove patient entries with minimum similarity scores until we have "K" patient entries.
5. Generate a recommendation list consisting of ATC codes of the selected patient entries.

Algorithm 6.2 is used to determine the recommendation list for $\mathrm{Patient}_{\mathrm{N}}$ (target patient), R_N, by means of User-based Collaborative Filtering. We categorized patient entries into three according to type of their attributes. We have set of diagnosis related patient entries (P_d), set of complaint related patient entries (P_c) and set of laboratory procedures related patient entries (P_l). For each group, we tried to find total K nearest neighbor patient entries which have highest similarity measures $s_{u,N}$. The value $s_{u,N}$ is a similarity measure between the patient $\mathrm{Patient_u}$ and the target patient $\mathrm{Patient_N}$. We explained our similarity measure in detail in Chapter 6.3. K is a predefined number which can be determined as a certain value or can be determined empirically. We determined it according to the results we get in evaluation step.

Algorithm 6.2 Pseudo code for the User-based Collaborative Filtering algorithm used in RHCS to generate top-K recommendation list for $\mathrm{Patient_N}$.

$P \leftarrow$ set of all patient entries.
$T \leftarrow$ set of all treatment plans for each patient entry.
$Patient_N \leftarrow$ target patient data.
$P_d \leftarrow$ set of all patient entries whose causes are diagnoses.
$P_c \leftarrow$ set of all patient entries whose causes are complaints.
$P_l \leftarrow$ set of all patient entries whose causes are laboratory procedures.

$P = \{Patient_1, Patient_2, ..., Patient_M\}$
$T = \{T_1, T_2, ..., T_M\}$ where T_M is the treatment applied for $Patient_M$.
$M \in \mathbb{R}_{>0}$ where M is the size of P.

$S_{1,N} \leftarrow$ similarity measure between $Patient_1$ and target patient $Patient_N$.
for i=1 **to** M+1 **do**
  calculate similarity scores $S_{i,N}$
end for

$K \in \mathbb{R}_{>0}$ where K, a predefined number, is the size for recommendation.
$S_{selectedK}$ set of selected K similarity scores.
$temp = 1$

Algorithm 6.3 Pseudo code for the User-based Collaborative Filtering algorithm used in RHCS to generate top-K recommendation list for $\text{Patient}_{\text{N}}$ (Continued).

repeat
  find patient entry $temp_d$ in P_d with maximum similarity score
  if $temp_d > 0$ **then**
    add $temp_d$ to $S_{selectedK}$
    add 1 to $temp$
  end if
  find patient entry $temp_c$ in P_c with maximum similarity score
  if $temp_c > 0$ **then**
    add $temp_c$ to $S_{selectedK}$
    add 1 to $temp$
  end if
  find patient entry $temp_l$ in P_l with maximum similarity score
  if $temp_l > 0$ **then**
    add $temp_l$ to $S_{selectedK}$
    add 1 to $temp$
  end if
until $temp = K$

$size \leftarrow$ size of set $S_{selectedK}$.
for i=0 **to** K-*size* **do**
  find minimum similarity score min in $S_{selectedK}$
  remove min from $S_{selectedK}$
end for
Assume $S_{selectedK} = \{S_{1,N}, S_{2,N}, ..., S_{K,N}\}$
$R_N = \{T_1, T_2, ..., T_K\}$ recommendation list for $Patient_N$.

CHAPTER 7

RESULTS AND EVALUATION

In this chapter, the evaluation results of offline experiments and user study are illustrated and a detailed discussion about results is given.

7.1 Evaluation Results of Offline Experiments

First, we evaluated RHCS by offline experiments (as explained in Chapter 5.6.1.1). We used patients in our dataset (pre-collected data) as test users (target patients). Table 7.1 shows numeric information about Offline Experiments.

Table 7.1: The Numeric Information about Offline Experiments.

Number of patients before data preprocessing	2866
Number of patients used after data preprocessing	2453
Number of patients tested	2453
Number of all ATC codes recommended	14817
Average number of ATC codes used per patient	6.04

Before data preprocessing, we had 2866 patients and 413 of them removed while data processed. We have 2453 patients used in RHCS and we tested all of these 2453 patients. For each of these test patients, we aimed to generate top-K recommendation plans. We tried to determine K which is number of ATC codes generated empirically. The average number of ATC codes used per patient is measured as 6.04 and so it is 6 approximately. As a logical interpretation, we picked two close numbers to this average number 6 as K which are 5 and 10. Hence, we evaluated RHCS both K=5 and K=10. We generated both top-5 and top-10 treatment plans as ATC codes and we evaluated the results according to three different evaluation metrics which are precision, recall and f-measure. The approximate evaluation results are illustrated in table 7.2.

Table 7.2: The Evaluation Results of Offline Experiments.

	TP	FP	TN	FN	precision (%)	recall (%)	f-measure (%)
K=5	11438	827	447973	3379	93.26	77.19	84.47
K=10	14804	9726	439074	13	60.35	99.91	75.25

In our dataset, number of ATC codes used per patient is varied from 3 to 10. As a more detailed evaluation, we also looked through the evaluation results for patients grouped by number of ATC codes per them. Figure 7.1, Figure 7.2 and Figure 7.3 illustrates the results for precision, recall and f-measure metrics accordingly. A detailed analysis about evaluation results is given in Chapter 7.3.1.

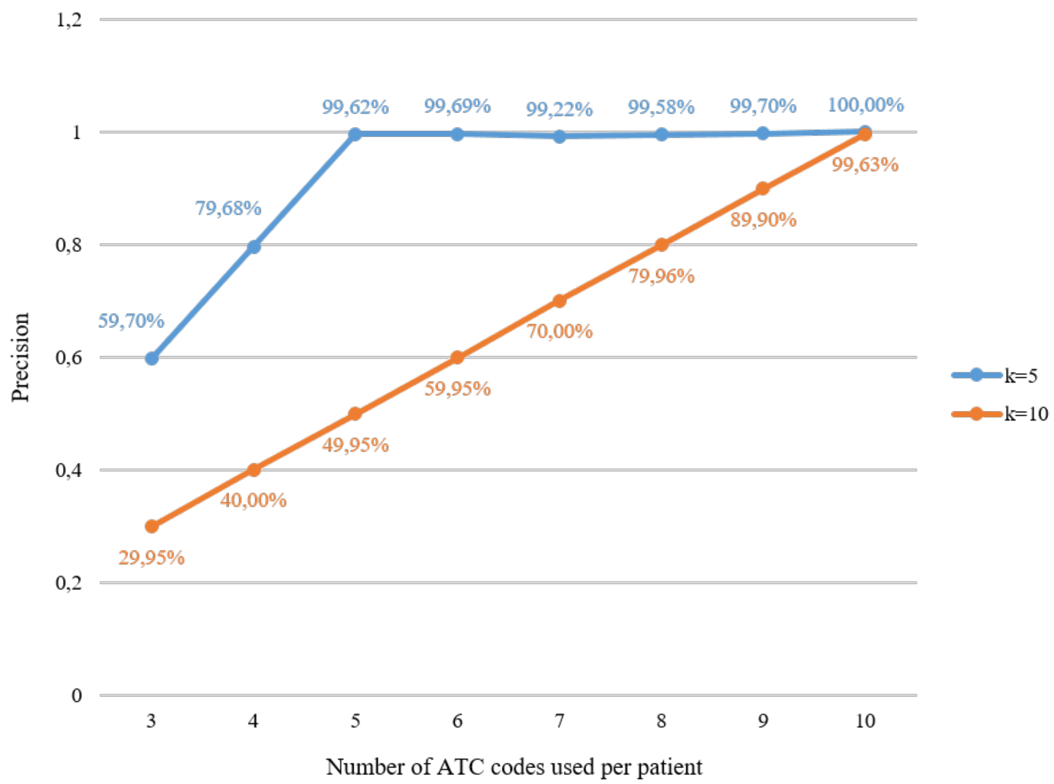

Figure 7.1: Precision for K=5 and K=10.

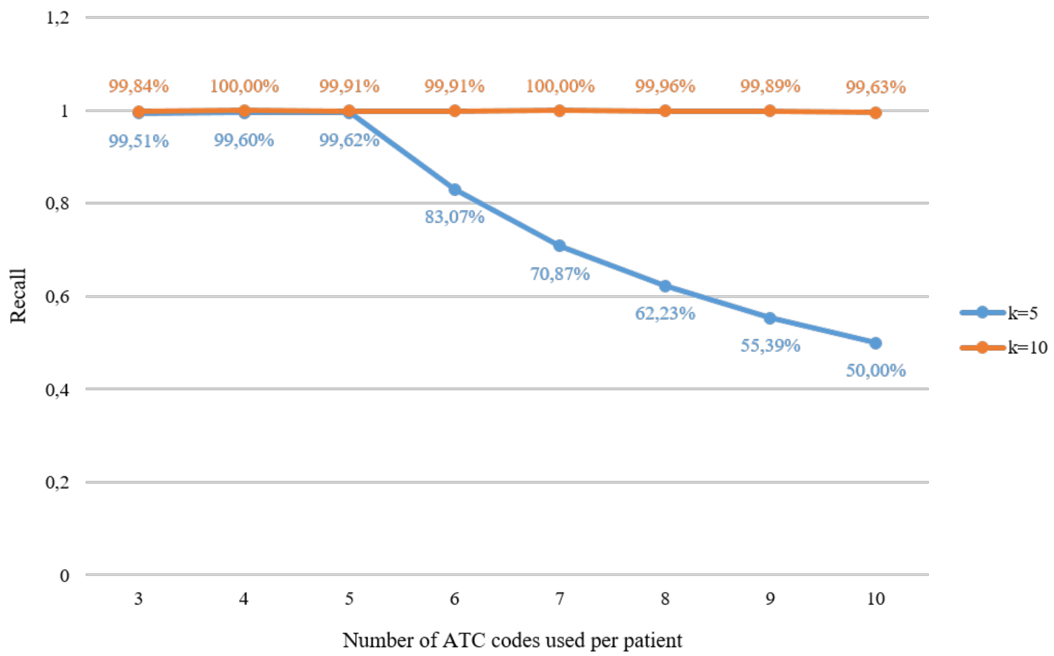

Figure 7.2: Recall for K=5 and K=10.

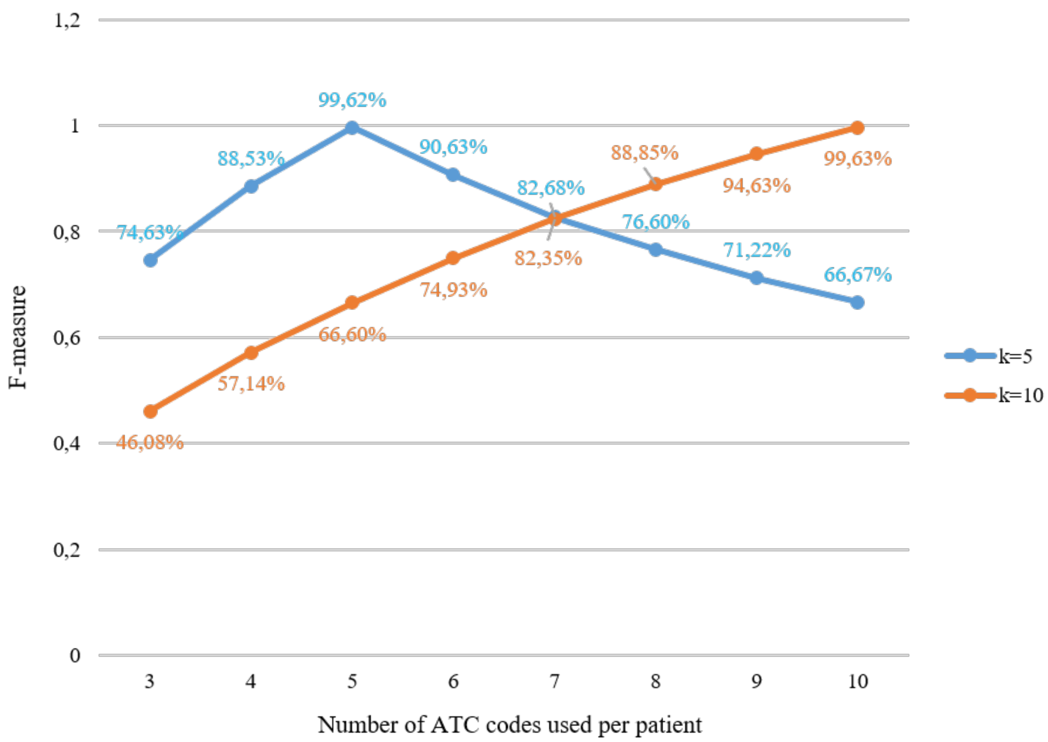

Figure 7.3: F-measure for K=5 and K=10.

7.2 Evaluation Results of User Study

User study is conducted with real system users (medical doctors) that perform some predetermined tasks. As it is illustrated in Table 7.3, there are 13 doctors participated in user study. These doctors are asked to evaluate 8 recommendation lists each having 10 ATC codes generated by RHCS for 8 different patient data.

In offline experiments, we showed that number of ATC codes per patient affects evaluation results. For instance, precision measured for patients with 10 ATC-codes is higher than precision for patients with 3 ATC-codes. Therefore, selecting random patients without considering number of ATC codes used for them may result biased evaluation results. In order to build a more reliable experiment set, we randomly select one patient from each of 8 different groups classified by number of ATC codes. Hence, we have 8 patients in total.

Doctors evaluated ATC codes generated for these 8 patients one by one picking scores between "1" to "5". "1" means not-related and "5" means very related. This scoring mechanism enable doctors to scale relatedness of ATC codes. The decision-support accuracy metrics we used in Chapter 7.1 do not allow such a scoring mechanism. Instead, they evaluated RHCS by categorizing the generated ATC codes as related or not related.

Table 7.3: The Numeric Information about User Study.

Number of patients used for user study	8
Number of doctors participated in user study	13

We generated a formula (Equation 7.1) to evaluate online experiment. Our aim is to measure relavancy of generated ATC codes for a given patient. "1" is 0% relevant and 2 means 25% relevant, 3 means 50%, 4 means 75% and 5 means 100% relevant. For each doctor and patient pair a relevancy score is calculated by the Equation 7.1 where $\#\ of\ 5\ scores_{doctor,patient}$ is the number of "5" scores given by doctor *doctor* to generated ATC codes for patient *patient* and K is the number of ATC codes generated which are 5 and 10 respectively.

$$
\begin{aligned}
r_{doctor,patient,K}(\%) = \frac{1}{K} * (\# \, of \; 5 \; scores_{doctor,patient} * 100 \\
+ \# \, of \; 4 \; scores_{doctor,patient} * 75 \\
+ \# \, of \; 3 \; scores_{doctor,patient} * 50 \\
+ \# \, of \; 2 \; scores_{doctor,patient} * 25 \\
+ \# \, of \; 1 \; scores_{doctor,patient} * \; 0)
\end{aligned}
\tag{7.1}
$$

We have 104 scores in total given by 13 doctors for each K=5 and K=10. Table 7.4 illustrates the relevancy scores calculated for K=5. Table 7.5 illustrates the relevancy scores calculated for K=10.

Table 7.4: Relevancy Scores calculated for K=5.

	P1	**P2**	**P3**	**P4**	**P5**	**P6**	**P7**	**P8**
D1	100	100	100	100	100	100	100	100
D2	95	100	95	100	100	100	100	95
D3	100	100	100	100	100	100	100	100
D4	100	100	100	100	100	100	100	100
D5	100	100	100	100	100	100	100	100
D6	100	100	100	100	100	100	100	100
D7	100	100	100	100	100	100	100	100
D8	95	100	95	95	100	95	100	100
D9	100	100	100	95	100	100	100	100
D10	90	100	100	100	100	95	100	100
D11	100	100	100	100	100	100	100	100
D12	100	100	100	100	100	100	100	100
D13	100	100	100	90	100	100	100	95

The success of RHCS according to a doctor is determined by Equation 7.2 which is mean of relevancy scores related to that doctor. Table 7.6 illustrates the relevancy score of system RHCS according to different doctors participated in user study when K is equal to 5 and Table 7.7 illustrates the doctor-based relevancy scores of RHCS when K is equal to 10.

$$
relevancy - score_{Di,K} \; (\%) = \frac{\sum_{j=1}^{8} r_{Di,Pj,K}}{8} \tag{7.2}
$$

Table 7.5: Relevancy Scores calculated for K=10.

	P1	**P2**	**P3**	**P4**	**P5**	**P6**	**P7**	**P8**
D1	100	100	100	100	100	100	100	100
D2	95	100	97.5	97.5	100	97.5	100	92.5
D3	92.5	95	100	100	100	100	100	100
D4	95	100	100	100	100	100	100	100
D5	100	100	100	100	100	100	100	100
D6	92.5	95	100	92.5	100	100	100	100
D7	100	100	100	100	100	100	100	100
D8	92.5	100	92.5	97.5	100	95	100	87.5
D9	100	100	100	97.5	100	100	100	100
D10	87.5	92.5	100	100	100	95	100	100
D11	100	100	100	100	100	100	100	100
D12	95	100	92.5	100	100	100	100	100
D13	100	95	100	95	100	100	100	97.5

In order to evaluate the overall relevancy of RHCS, we use Equation 7.3 which is mean of relevancy scores measured for each doctor. Table 7.8 shows the relevancy score we calculated for K=5 and K=10.

$$relevancy - score_K\ (\%) = \frac{\sum_{i=1}^{13} relevancy - score_{Di,K}}{13} \tag{7.3}$$

7.3 Discussion

RHCS generates treatment plan list as ATC codes for patients. We evaluated RHCS by means of offline experiments and user studies. In this section, we discussed the evaluation results.

7.3.1 Discussion on Offline Experiment

We use three different evaluation metrics which are precision, recall and f-measure. We measured the percentages for both K=5 and K=10.

Table 7.6: Relevancy Scores for each doctor calculated for K=5.

Doctor	Relevancy Score
D1	100
D2	98.125
D3	100
D4	100
D5	100
D6	100
D7	100
D8	97.5
D9	99.375
D10	98.125
D11	100
D12	100
D13	98.125

Table 7.7: Relevancy Scores for each doctor calculated for K=10.

Doctor	Relevancy Score
D1	100
D2	97.5
D3	98.4375
D4	98.4375
D5	100
D6	97.5
D7	100
D8	95.625
D9	99.6875
D10	96.875
D11	100
D12	98.4375
D13	98.4375

Table 7.8: Relevancy Score for overall system RHCS for K=5 and K=10.

	Relevancy Score
K=5	99.32692308
K=10	98.60576923

7.3.1.1 Precision Results

Precision is the percentage of relevant recommendations within all recommendations. Recall is an important metrics in order to find out whether generated ATC codes are relevant or not.

Figure 7.1 is the precision vs. number of ATC codes used per patient graph. For both K=5 and K=10, the precision values getting higher with the number of ATC codes per patient is increased as it is expected. When we generate a recommendation list with the size of 5, we can get 60% as the maximum precision value for a patient who uses 3 ATC codes. However, for a patient who uses more than 5 ATC codes, the precision value can reach to 100%.

Precision scores we get for K=5 are greater than the scores for K=10. This is also an expected situation as K (number of recommendations) is denominator to measure precision and higher denominator results in lower ratio.

The precision values for overall system RHCS are illustrated in Table 7.2. Precision for K=5 is approximately 93.26% and precision for K=10 is approximately 60.35%.

7.3.1.2 Recall Results

Recall is the percentage of having truely recommended ATC codes within all relevant or recommendable ATC codes. It is an important metrics in order to find out whether there is an ATC code being not generated when it is should be.

Figure 7.2 is the recall vs. number of ATC codes used per patient graph. For K=5, patients who use 5 or more ATC codes have lower recall values than those using 3 or 4 ATC codes. This is because number of ATC codes per patient is denominator to measure recall. When we generate a recommendation list with the size of 5, we can get 50% as the maximum recall value for a patient who uses 10 ATC codes. However, for a patient who uses 5 or less ATC codes, this value can reach to 100%.

For K=10, the numbers of ATC codes used per patient do not affect recall values too much since there is no patient who uses more than 10 ATC codes.

In general, recall scores we get for K=10 are greater than the scores for K=5. For recall, we are not interested in denominator part which is number of ATC codes per patient because we cannot change these number. In order to have higher recall values, we have to increase nominator part which is the number of truely recommended ATC codes (*true-positive*). For K=10, we have a higher chance to have more truly recommended ATC codes. Hence, it is expected to have higher recall scores for K=10.

The recall values for overall system RHCS are illustrated in Table 7.2. Recall for K=5 is approximately 77.2% and recall for K=10 is approximately 99.9%.

7.3.1.3 F-measure Results

F-measure is the harmonic mean of precision and recall.

Figure 7.3 is the f-measure vs. number of ATC codes used per patient graph. F-measure is a measure to make use of both precision and recall evaluation metrics and so it is difficult to find a direct correlation between f-measure values and number of ATC codes used per patient.

The f-measure values for overall system RHCS are illustrated in Table 7.2. F-measure for K=5 is approximately 84.47% and f-measure for K=10 is approximately 75.25%.

7.3.2 Discussion on User Study

We provided an user study set with 8 different patients and 13 different medical doctors. We are asked that doctors to evaluate recommendation lists generated for these 8 patients. Doctors selected scores from 1 to 5 to determine the relevancy of ATC codes.

Table 7.4 is the doctor-patient matrix for K=5 and Table 7.5 is the doctor-patient matrix for K=10. The elements of matrices are relevancy scores. For instance, the element at second row and second column is the relevancy score of patient P1 according to doctor D1.

We cannot make a decision on that K=5 or K=10 results in better relevancy scores.

While generating recommendation list, we used a priority score mechanism which is explained in Chapter 6.4. Priority score mechanism is not a natural ordering so we cannot state that first ATC code in recommendation list must be more relevant than second recommendation. There are 4 cases that K=10 is better than K=5 in terms of relevancy scores. K=5 is better in other 100 cases.

Table 7.6 shows the mean of relevancy scores of all patients according to each doctor for K=5 and these scores between 98.125% and 100%. Table 7.7 shows the mean of relevancy scores of all patients according to each doctor for K=10 and these scores are between 95.625% and 100%.

Table 7.8 illustrates the mean of relevancy scores which we calculated for each doctor. For K=5, this score is 99.327% and for K=10, this score is 98.606%. These values are too close to each other, so it is not obvious that K=5 is superior than K=10 in terms of relevancy score.

7.3.3 Overall Analysis

We can summarize our findings as follows:

- In offline experiment, we assume that only ATC codes used for patient are recommendable. If generated ATC codes are not used by patients, we classified them as falsely recommended (FP). However, this assumption is not totally accurate. ATC codes generated by RHCS can be relevant in spite of not using in our clinical dataset. Hence, our precision percentages measured on offline experiment set may be under presented so it may be misleading.

- User study has some drawbacks. First problem is that it is not an objective method as the success related to personal decisions of doctors. Second drawback is that we can evaluate the relevancy, however we cannot learn whether there is a missing ATC code in recommendation list or not. So we cannot evaluate system by a metric like recall.

- Determining which K is more preferable depends on our aim. If we want to have a greater precision value than K should be selected as 5. If recall value is more important for us, than K should be 10. We prefered K as 10 and there are three main reasons behing this preference:

  - Because one of our motivation point is guiding health professionals in terms of reminding ATC codes, recall is an important metric for us.
  - Precision measured on offline experiment set can be misleading.
  - According to user study results, it is not worth to consider the difference between scores for K=5 and 10, both are acceptable.

- For K=10 on offline experiment set; precision is approximately 60.35%, recall is approximately 99.91% and f-measure is approximately 75.25%. For K=10 on online experiment set; the relevancy score is approximately 98.6%.

CHAPTER 8

CONCLUSION AND FUTURE WORK

Within this thesis study, a medical recommendation system named RHCS has been presented. RHCS is developed as a part of home health care service for geriatric patients. Some of the major contributions of this study can be listed as follows:

- We determined our system requirements through a user study conducted with health professionals who work in Numune Hospital.
- We did not use any virtual data. Patients data were taken from Numune Hospital.
- We followed standardizations of Minister of Health of the Republic of Turkey. RHCS is compatible with drug barcode standards and ICD-10 classification system.
- RHCS is ontology-based and it makes system advantageous in terms of interoperability, scalability and expandability.
- RHCS follows the international standard, ATC classification system, to provide interaction with different health care systems.
- RHCS can work for different patients outside of our clinical data repository.
- RHCS uses user-based collaborative filtering recommendation approach and it is empowered by historical data of patients.
- We conducted both offline experiments and a user study. Offline experiments are evaluated by precision, recall and f-measure. Offline evaluation results are

all higher than 60% and it demonstrates that RHCS is a successful recommendation system.

- 13 medical doctors are participated in user study. We evaluated user study through generating a relevancy score. We measured this relevancy score as approximately 98% and it shows an evidence that according to 13 medical doctors, RHCS generates relevant recommendations.

As a future work, we will use RHCS as a part of an integrated patient based home health care service. Some of the significant characteristics of this home health care service which we will develop can be stated as follows:

- Home health care service will involve all healthcare actors including patients and patient relatives to care, assessment and treatment process. System stakeholders will access information and get involved in the system according to their roles. Since it will be patient-based, patients can be actively involved in treatment and nursing period.
- In Turkey, currently there is no standard Electronic Health Record system. This study aims to develop an internationally standardised electronic health record to keep patient medical records in a more systematic way. It will follow some international standards to provide interaction with different systems.
- It will provide a platform to gather metrics or parameters about some illnesses and share them to the stakeholders with regard to their roles.
- It will provide an alert systems which makes automatic reminding and/or informing to stakeholders.
- It will provide a rule-based decision support system which enables high-quality service by simulating the decision-making ability of a health professional.
- This system will provide to access information regardless of time and place through web based platform-independent application.
- It will be compatible with mobile devices.

- It will be able to work with different hospital management systems and third party system or applications via API supports.

Besides, RHCS can be adapted to work with instantaneous data measurements of patients. We can obtain data by means of different biomedical sensors and medical devices like electrocardiography (ECG or EKG), digital scales, digital sphygmomanometers (device used to measure blood pressure) and glucometers (device to monitor glycaemia).

REFERENCES

[1] Barış Uçar. Population projections, 2013-2075. http://www.turkstat.gov.tr/PreHaberBultenleri.do?id=15844, Feb 2013. Online; accessed 20-Sep-2015.

[2] Dilek Aslan. Aged population is increasing in Turkey: Situation analysis. *Acta Medica*, 2:23–25, 2013.

[3] Definition of chronic disease. http://www.medicinenet.com/script/main/art.asp?articlekey=33490. Online; accessed 26-Sep-2015.

[4] H. Erdal Akalın. Dünyada ve Türkiye'de Kronik Hastalıklar. http://www.sasder.org/sunumlar/ulusal-kongre3/Kronik-Hastaliklar_EAkalin.pdf, 2014. Online; accessed 15-Sep-2015.

[5] Belgin Ünal, Gül Ergör, Gönül Dinç Horasan, Sibel Kalaça, and Kaan Sözmen. Türkiye kronik hastalıklar ve risk faktörleri çalışması. Technical report, Ankara, 2013.

[6] Chronic diseases and health promotion. http://www.who.int/chp/en/. Online; accessed 16-Sep-2015.

[7] Türkiye Kalp ve Damar Hastalıklarını Önleme ve Kontrol Programı: Birincil , İkincil ve Üçüncül Korumaya Yönelik Stratejik Plan ve Eylem Planı. Technical report, Ankara, 2010.

[8] H. Erdal Akalın, Mine Durusu Tanrıöver, and Faik Sayran. Sürdürülebilir Sağlık Sistemi için Kronik Hastalık Yönetiminde Elektronik Sağlık Kayıtlarının Rolü. http://www.tusiad.org.tr/__rsc/shared/file/ErdalAkalin-03072012.pdf. Online; accessed 19-Sep-2015.

[9] Health at Home. http://www.acibademmobil.com.tr/default/7/evde_saglik.asg. Online; accessed 20-Feb-2015.

[10] KarMed HBYS. http://www.kardelenyazilim.com/hbys.html. Online; accessed 20-Feb-2015.

[11] Fethiye Erdil. Yaşlanan toplum ve yaşlinin evde bakimi. pages 127–134, 2002.

[12] Ayşe Güler. Evde Bakım Hizmetleri. http://halksagligi.med.ege.edu.tr/seminerler/2004-05/EvdeBakimHizmetleri_Ayse.pdf, 2004. Online; accessed 26-Sep-2015.

[13] William E Hale, Ronald G Marks, and Ronald B Stewart. Drug use in a geriatric population. *Journal of the American Geriatrics Society*, 27(8):374–377, 1979.

[14] Resmi Gazete (2005). Evde Bakım Hizmetleri Sunumu Hakkında Yönetmelik. *Resmi Gazete*, Resmi Gazete Sayısı: 25751, 10.03.2005.

[15] Evde Sağlık Hizmetleri Koordinatörlüğü. `http://www.istanbulsaglik.gov.tr/w/sb/tedk/evdesaglik_formlar.asp`. Online; accessed 20-Feb-2015.

[16] Horizon Home Health Care. `http://www.hhhcs.com/`. Online; accessed 20-Feb-2015.

[17] Axxess. `http://www.axxess.com/`. Online; accessed 20-Feb-2015.

[18] Axiscare Homecare Management. `http://axiscaresales.com/`. Online; accessed 20-Feb-2015.

[19] HealthCare First. `https://app.healthcarefirst.com`. Online; accessed 20-Feb-2015.

[20] Allscripts homecare. `http://www.allscripts.com/products-services/products/homecare`. Online; accessed 20-Feb-2015.

[21] K. Shimada, H. Takada, S. Mitsuyama, H. Ban, H. Matsuo, H. Otake, H. Kunishima, K. Kanemitsu, and M. Kaku. Drug-Recommendation System for Patients with Infectious Diseases. In *AMIA Annual Symposium Proceedings*, page 1112, 2005.

[22] Meisamshabanpoor and Mehregan Mahdavi. Implementation of a recommender system on medicalrecognition and treatment. *International Journal of e-Education, e-Business, e-Management and e-Learning*, 2(4):315, 2012.

[23] L Duan, W Street, and D Lu. A nursing care plan recommender system using a data mining approach. In *Paper presented at the Proceedings of the 3rd INFORMS Workshop on Data Mining and Health Informatics*, 2008.

[24] T Ryan Hoens, Marina Blanton, Aaron Steele, and Nitesh V Chawla. Reliable medical recommendation systems with patient privacy. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 4(4):67, 2013.

[25] Alex Rodriguez, Edward Jimenez, Jesús Fernández, Martin Eccius, Jorge Marx Gomez, Giner Alor-Hernandez, Rubén Posada-Gomez, and Carlos Laufer. Semmed: Applying semantic web to medical recommendation systems. In *Intensive Applications and Services, 2009. INTENSIVE'09. First International Conference on*, pages 47–52. IEEE, 2009.

[26] Thean Pheng Lim, Wahidah Husain, and Nasriah Zakaria. Recommender system for personalised wellness therapy.

[27] Chuan-Jun Su and Chang-Yu Chiang. Iaserv: An intelligent home care web services platform in a cloud for aging-in-place. *International journal of environmental research and public health*, 10(11):6106–6130, 2013.

[28] About GS1. http://www.gs1.org/about. Online; accessed 16-Apr-2015.

[29] Ulusal Bilgi Bankası. http://www.sgk.gov.tr/wps/wcm/connect/SGK+Internet/genel_saglik_sigortasi/tibbi_malzeme/ulusal_bilgi_bankasi/. Online; accessed 16-Apr-2015.

[30] Learn More About HIBCC. http://www.hibcc.org/about-hibcc/. Online; accessed 16-Apr-2015.

[31] GS1 Barkod. http://www.gs1.tobb.org.tr/GS1barkod.php. Online; accessed 16-Apr-2015.

[32] ATC/DDD Index 2015". http://www.whocc.no/atc_ddd_index/. Online; accessed 20-Apr-2015.

[33] Purpose of the ATC/DDD system. http://www.whocc.no/atc_ddd_methodology/purpose_of_the_atc_ddd_system/ttp://www.gs1.tobb.org.tr/GS1barkod.php. Online; accessed 20-Apr-2015.

[34] Structure and principles. http://www.whocc.no/atc/structure_and_principles/. Online; accessed 20-Apr-2015.

[35] International Classification of Diseases (ICD). http://www.who.int/classifications/icd/en/. Online; accessed 21-Apr-2015.

[36] ICD-10 Turkish version of 10th Revision of Diseases and Health Problems International Classification. http://www.saglik.gov.tr/EN/belge/2-96/icd-10-turkish-version-of-10th-revision-of-diseases-and-.html. Online; accessed 21-Apr-2015.

[37] Thomas R Gruber. Toward principles for the design of ontologies used for knowledge sharing? *International journal of human-computer studies*, 43(5):907–928, 1995.

[38] What is SNOMED CT? http://www.ihtsdo.org/snomed-ct/what-is-snomed-ct. Online; accessed 21-Apr-2015.

[39] Why SNOMED CT? http://www.ihtsdo.org/snomed-ct/why-should-i-get-snomed-ct. Online; accessed 21-Apr-2015.

[40] SNOMED CT Starter Guide. Technical report.

[41] Francesco Ricci, Lior Rokach, and Bracha Shapira. *Introduction to recommender systems handbook*. Springer, 2011.

[42] Mukund Deshpande and George Karypis. Item-based top-n recommendation algorithms. *ACM Transactions on Information Systems (TOIS)*, 22(1):143–177, 2004.

[43] Mark Claypool, Anuja Gokhale, Tim Miranda, Pavel Murnikov, Dmitry Netes, and Matthew Sartin. Combining content-based and collaborative filters in an online newspaper. In *Proceedings of ACM SIGIR workshop on recommender systems*, volume 60. Citeseer, 1999.

[44] Akshi Kumar and MPS Bhatia. Community expert based recommendation for solving first rater problem. *International Journal of Computer Applications*, 37(10):7–13, 2012.

[45] Xuan Nhat Lam, Thuc Vu, Trong Duc Le, and Anh Duc Duong. Addressing cold-start problem in recommendation systems. In *Proceedings of the 2nd international conference on Ubiquitous information management and communication*, pages 208–211. ACM, 2008.

[46] John McCrae, Anton Piatek, and Adam Langley. Collaborative filtering. *http:// www. imperialviolet. org*, 2004.

[47] Dominic John Repici. The Stability-Plasticity Problem. `http://standoutpublishing.com/g/stability-plasticity-problem.html`. Online; accessed 2-May-2015.

[48] Robin Burke. Hybrid recommender systems: Survey and experiments. *User modeling and user-adapted interaction*, 12(4):331–370, 2002.

[49] Manisha Hiralall and Wojtek Kowalczyk. Recommender systems for e-shops. *Business Mathematics and Informatics paper, Vrije Universiteit, Amsterdam*, 2011.

[50] SongJie Gong, HongWu Ye, and HengSong Tan. Combining memory-based and model-based collaborative filtering in recommender system. In *Circuits, Communications and Systems, 2009. PACCS'09. Pacific-Asia Conference on*, pages 690–693. IEEE, 2009.

[51] Greg Linden, Brent Smith, and Jeremy York. Amazon. com recommendations: Item-to-item collaborative filtering. *Internet Computing, IEEE*, 7(1):76–80, 2003.

[52] Badrul Sarwar, George Karypis, Joseph Konstan, and John Riedl. Item-based collaborative filtering recommendation algorithms. In *Proceedings of the 10th International Conference on World Wide Web*, WWW '01, pages 285–295, New York, NY, USA, 2001. ACM.

[53] John S. Breese, David Heckerman, and Carl Kadie. Empirical analysis of predictive algorithms for collaborative filtering. In *Proceedings of the Fourteenth*

Conference on Uncertainty in Artificial Intelligence, UAI'98, pages 43–52, San Francisco, CA, USA, 1998. Morgan Kaufmann Publishers Inc.

[54] Model-based recommendation systems. `http://www.cs.carleton.edu/cs_comps/0607/recommend/recommender/modelbased.html`. Online; accessed 7-June-2015.

[55] Christopher D Manning, Prabhakar Raghavan, Hinrich Schütze, et al. *Introduction to information retrieval*, volume 1. Cambridge university press Cambridge, 2008.

[56] Nick Littlestone. Learning quickly when irrelevant attributes abound: A new linear-threshold algorithm. *Machine Learning*, 2(4):285–318, 1988.

[57] Alexander Felfernig, Michael Jeran, Gerald Ninaus, Florian Reinfrank, Stefan Reiterer, and Martin Stettinger. Basic approaches in recommendation systems. In *Recommendation Systems in Software Engineering*, pages 15–37. Springer, 2014.

[58] Robert H Guttman. *Merchant differentiation through integrative negotiation in agent-mediated electronic commerce*. PhD thesis, Citeseer, 1998.

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

[60] Robin Burke. Hybrid web recommender systems. In *The adaptive web*, pages 377–408. Springer, 2007.

[61] Brendon Towle and Clark Quinn. Knowledge based recommender systems using explicit user models. In *Proceedings of the AAAI Workshop on Knowledge-Based Electronic Markets*, pages 74–77, 2000.

[62] Daniel Billsus and Michael J Pazzani. User modeling for adaptive news access. *User modeling and user-adapted interaction*, 10(2-3):147–180, 2000.

[63] Thomas Tran and Robin Cohen. Hybrid recommender systems for electronic commerce. In *Proc. Knowledge-Based Electronic Markets, Papers from the AAAI Workshop, Technical Report WS-00-04, AAAI Press*, 2000.

[64] Barry Smyth and Paul Cotter. A personalised tv listings service for the digital tv age. *Knowledge-Based Systems*, 13(2):53–59, 2000.

[65] Ahmad M Ahmad Wasfi. Collecting user access patterns for building user profiles and collaborative filtering. In *Proceedings of the 4th international conference on Intelligent user interfaces*, pages 57–64. ACM, 1998.

[66] Chumki Basu, Haym Hirsh, William Cohen, et al. Recommendation as classification: Using social and content-based information in recommendation. In *AAAI/IAAI*, pages 714–720, 1998.

[67] Michelle Keim Condliff, David D Lewis, David Madigan, and Christian Posse. Bayesian mixed-effects models for recommender systems. In *ACM SIGIR'99 Workshop on Recommender Systems: Algorithms and Evaluation*, volume 15. Citeseer, 1999.

[68] Prem Melville, Raymond J Mooney, and Ramadass Nagarajan. Content-boosted collaborative filtering for improved recommendations. In *AAAI/IAAI*, pages 187–192, 2002.

[69] Raymond J Mooney and Loriene Roy. Content-based book recommending using learning for text categorization. In *Proceedings of the fifth ACM conference on Digital libraries*, pages 195–204. ACM, 2000.

[70] Badrul M Sarwar, Joseph A Konstan, Al Borchers, Jon Herlocker, Brad Miller, and John Riedl. Using filtering agents to improve prediction quality in the grouplens research collaborative filtering system. In *Proceedings of the 1998 ACM conference on Computer supported cooperative work*, pages 345–354. ACM, 1998.

[71] Marko Balabanović. An adaptive web page recommendation service. In *Proceedings of the first international conference on Autonomous agents*, pages 378–385. ACM, 1997.

[72] Marko Balabanović. Exploring versus exploiting when learning user models for text recommendation. *User Modeling and User-Adapted Interaction*, 8(1-2):71–102, 1998.

[73] Ingo Schwab, Alfred Kobsa, and Ivan Koychev. Learning user interests through positive examples using content analysis and collaborative filtering. *Internal Memo, GMD, St. Augustin, Germany*, 2001.

[74] National Centre of Sustainability at Swinburne University of Technology. *A Short Guide to Monitoring & Evaluation*, 1 edition, Jan 2011. Online; accessed 1-August-2015.

[75] Guy Shani and Asela Gunawardana. Evaluating recommendation systems. In *Recommender systems handbook*, pages 257–297. Springer, 2011.

[76] Eytan Bakshy, Dean Eckles, and Michael S Bernstein. Designing and deploying online field experiments. In *Proceedings of the 23rd international conference on World wide web*, pages 283–292. ACM, 2014.

[77] Ron Kohavi, Roger Longbotham, Dan Sommerfield, and Randal M Henne. Controlled experiments on the web: survey and practical guide. *Data mining and knowledge discovery*, 18(1):140–181, 2009.

[78] Jonathan L Herlocker, Joseph A Konstan, Loren G Terveen, and John T Riedl. Evaluating collaborative filtering recommender systems. *ACM Transactions on Information Systems (TOIS)*, 22(1):5–53, 2004.

[79] Marie A Chisholm-Burns, Barbara G Wells, Terry L Schwinghammer, Patrick Malone, Jill Kolesar, John Rotschafer, and Joseph DiPiro. *Pharmacotherapy Principles and Practice*. McGraw-Hill New York, third edition, 2010.

[80] Stephen J McPhee, Maxine A Papadakis, Michael W Rabow, et al. *Current Medical Diagnosis & Treatment 2012*. McGraw-Hill Medical, 2012.

[81] Fred F Ferri. *Ferri's Clinical Advisor: Instant Diagnosis and Treatment, 2002*. W.B. Saunders Company, 2003.

[82] Alan J Sinclair, John E Morley, and Bruno Vellas. *Pathy's Principles and Practice of Geriatric Medicine*. John Wiley & Sons, 2012.

[83] Sukhdev Chatu. *The Hands-on Guide to Clinical Pharmacology*, volume 14. John Wiley & Sons, 2011.

[84] Mark H Beers, Robert Berkow, et al. *The Merck manual of diagnosis and therapy.* Number Ed. 17. Merck and Co. Inc., 1999.