DOKUZ EYLÜL UNIVERSITY
GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

ONTOLOGY BASED RECOMMENDATION SYSTEM IN E-LEARNING FOR TURKISH

by
Mehmet MİLLİ

January, 2016
İZMİR

ONTOLOGY BASED RECOMMENDATION SYSTEM IN E-LEARNING FOR TURKISH

A Thesis Submitted to the
Graduate School of Natural and Applied Sciences of Dokuz Eylül University
In Partial Fulfillment of the Requirements for the Master of
Science in Computer Engineering

by

Mehmet MİLLİ

January, 2016
İZMİR

M.Sc THESIS EXAMINATION RESULT FORM

We have read the thesis entitled **"ONTOLOGY BASED RECOMMENDATION SYSTEM IN E-LEARNING FOR TURKISH"** completed by **MEHMET MİLLİ** under supervision of **ASST. PROF. DR. ÖZLEM AKTAŞ** and we certify that in our opinion it is fully adequate, in scope and in quality, as a thesis for the degree of Master of Science.

Asst. Prof. Dr. Özlem AKTAŞ

Supervisor

Asst. Prof. Dr. Derya BİRANT

(Jury Member)

Asst. Prof. Dr. Özgür TAMER

(Jury Member)

Prof.Dr. Ayşe OKUR

Director

Graduate School of Natural and Applied Sciences

ACKNOWLEDGEMENTS

I would like to express my gratitude and appreciation to my supervisor, Asst. Prof. Dr. Özlem AKTAŞ, for her constant support, encouragement, guidance, advice and criticism throughout this study. It was a great honor to work with her for this work.

I would like to offer my special thanks to my friends, Emre ÜNSAL, Mete Uğur AKDOĞAN, Emel ALKIM, Mehmet CENGİZ, Mustafa BATAR, Rıdvan SÖYLER, Muhammed YALÇIN, and Reza DABBAGHZADEH for their help, support and cheerful presence through not only the course of this study but also any course of my life.

I am greatly indebted to my family especially my twins Musa MİLLİ, and my fiancé Nursel SÖYLEMEZ for their support, patience and help. It would not have been able to complete this thesis without their support and help.

Finally, I would like to thanks to the members of my jury Asst. Prof. Dr. Derya BİRANT, Asst. Prof. Dr. Özgür TAMER for reviewing this thesis.

Mehmet MİLLİ

ONTOLOGY BASED RECOMMENDATION SYSTEM IN E-LEARNING FOR TURKISH

ABSTRACT

Rapid development of Internet usage today, increasing the quality and diversity of services offered to users has led to continual rise of data on the Internet day after day. It became a major problem how to store the collected data and how to interpret with them when using again. Among the huge mass of data, finding a product they wanted has become impossible with classical methods for users. For previous decades semantic web technologies and recommender systems have been used for work out these problems.

These technologies are two of the most popular techniques studied by both academia and industry which are commonly used for e-commerce, entertainment sites, and social networks. However, usage of semantic web applications and recommender systems in education field is limited. In this thesis these technologies are used for e-learning field. These technologies are combined in this thesis in order to cope with some RS problems such as data sparsity and cold start.

The subject of this study contains of the science and technology lesson subjects. Its main aim is to guide primary and secondary school students. To reach this aim, first step is creating ontology of subject of science and technology lessons by using protégé ontology editor. Then this ontology has been questioned with SPARQL. Second step is obtaining recommendation list by using collaborative filtering (CF). CF finds users interest about items that they have never seen and taste before, using their before ratings. Last step, is combining first and second steps. This step is implementing parallelized hybridization design approach to our project.

Keywords: Semantic web, ontologies, recommender systems, e-learning.

TÜRKÇE İÇİN E-ÖĞRENME ORTAMLARINDA ONTOLOJİ TABANLI ÖNERİ SİSTEMİ

ÖZ

Günümüzde internet kullanımın hızla gelişmesi, kullanıcıya sunulan servislerin çeşitliliğinin ve kalitesinin artması internet üzerindeki verinin gün geçtikçe sürekli artmasına neden olmuştur. İnternet üzerindeki bu verinin nasıl saklanacağı ve tekrar kullanılırken nasıl yorumlanacağı büyük bir problem haline gelmiştir. Kullanıcıların bu kadar büyük veri yığınları arasında istedikleri ürünü bulmaları klasik yöntemler ile imkânsızlaşmıştır. Son yıllarda anlamsal web teknolojileri ve tavsiye sistemleri bu sorunları çözmek için kullanılmaktadır.

Bu teknolojiler genelde e-ticaret, eğlence siteleri ve sosyal ağlarda kullanılan hem akademik hem de endüstri alanında çalışılan en popular iki tekniktir. Fakat anlamsal web uygulamalarının ve öneri sistemlerinin eğitim alanında kullanımı oldukça sınırlıdır. Bu tez kapsamında bu teknolojiler e-öğrenme ortamları için kullanılmıştır. Öneri sistemlerinin soğuk başlangıç ve seyreklik gibi bazı sorunlarını çözüm getirebilmek için bu teknolojiler bu tez de melez bir yöntemle birleştirilmiştir.

Bu çalışma Fen ve Teknoloji dersinin konularını içerir. Çalışmanın amacı ilk ve ortaokul öğrencilerine rehberlik etmektir. Bu amaca ulaşmak için birinci aşamada fen ve teknoloji dersi konularının ontolojisi protégé ile oluşturulmuştur ve oluşturulan bu ontoloji SPARQL ile sorgulanmıştır. İkinci aşamada işbirlikçi filtreleme (IF) kullanılarak önerilecek liste elde edilir. IF kullanıcılara daha önce görmedikleri ve değerlendirmedikleri, ilgisini çekebilecek konuları kullanıcının önceki ürün değerlendirmelerini kullanarak bulur. Son aşama birinci ve ikinci aşamanın birleşmesinden oluşur. Bu aşama da paralel melez yaklaşımı çalışmaya uyarlanır.

Anahtar kelimeler: Anlamsal web, ontolojiler, öneri sistemleri, e-öğrenme.

CONTENTS

Page

LIST OF FIGURES

Page

LIST OF TABLES

Page

CHAPTER ONE
INTRODUCTION

1.1 General

Rapid development of Internet usage today, increasing the quality and diversity of services offered to users has led to continual rise of data on the Internet day after day. It became a major problem how to store the collected data and how to interpret with them when using again. Moreover, among the huge mass of data users finds their requirements has become impossible with classical methods for them. Therefore various technologies emerged for cope with this problem. One of them is semantic web (SW) technology. It provides a computer environment than can be understood by software agents which could allow the machines to understand web contents itself. SW has been used web search machines, digital libraries, automatic web service, and distributed computing applications until now.

Another technology is recommendation systems (RS). It is software tools and techniques providing suggestions for users by using their items rate and items features. The main goal of RS is to help users in finding their requirements. RS has been used e-commerce, entertainment industry, service industry, and social networks applications until now.

In recent years, fast development of computer and information technologies provides increasing usage of Internet intensely in many areas. Thanks to the development of internet, various technologies in various sectors has been developed such as e-commerce, finance, and communications. One of these areas is e-learning applications. E-learning approach is the effective use of technological tools and applications in learning. It has also known as computer based education, web based education or distance education. The benefits and drawbacks of e-learning have been debated; it still continues to develop rapidly.

Before, SW and RS used many areas effectively but their usage of e-learning is limited. When they use effectively in this areas, they may be suitable for e-learning applications. In the future they may be benefit tools for students.

In this study, ontology based recommendation system in e-learning for Turkish (OBReSET) has been created. The subject of this study contains the science and technology lesson subjects of third, fourth grade of primary school and all grade of secondary school. The main purpose of the applications is while students studying lesson on the e-learning environment, guide them to learning material which supplying their requirements.

1.2 Organization of the Thesis

This study composes of six main chapters. These chapters are semantic web and ontologies, recommendations systems, methods and approaches in ontology based recommendation system, evaluation metrics and conclusion and future work. First chapter is this section which in proposed system and introduction of thesis are explained. Other chapters are explained briefly below;

In the second chapter, the evolution of World Wide Web and the concept of SW technologies are described. XML and RDF file structure are explained. Advantages of XML and RDF, SW standards and layers of semantic web, the creating tools and interfaces are discussed. How semantic web technology differs from existing web data is explained.

In the third chapter of the thesis, information RS are given and presented RS types especially collaborative filtering (CF), content based filtering (CBF), and hybrid systems (HS). Limitations and problems of CF and CBF approaches such as cold start and data sparsity are given. Solutions of these limitations and problems are discussed. Lastly, the emergence of the semantic recommendation is explained.

What is present in the fourth chapter of the thesis is the applicability of the features which defines Web 3.0 and SW to the e-learning education. Methods and

materials which are used for this thesis are given. How to create similarity matrix by using Pearson Correlation (PC), Cosine Similarity (CS) and Adjustable Cosine Similarity (ACS) approaches is explain. How to selection neighborhood by using threshold method and k neighborhood algorithm are explained. Protégé which is used creating ontology and Jena which is used querying RDF files are described.

In the fifth chapter of the thesis, the application which is created in the study are evaluated. The success and accuracy of the application has been measured by using most known methods such as mean absolute error (MAE), mean square error (MSE) root mean square error (RMSE) and receiver operating characteristic (ROC).

In the last chapter of the thesis, present conclusion of the study and contribution to academic areas. Then, future works is discussed.

CHAPTER TWO
SEMANTIC WEB AND ONTOLOGIES

2.1 The Progress of World Wide Web

Web is an interconnected information pool which is pervaded all over the world (Bansal, Kona, Blake, & Gupta, 2008). Nowadays, web environment is being improved every day with the help of improvement in the field of technology (Figure 2.1). Starting with 'ARPANET' (Advanced Research Projects Agency Network) known as the ancestor of the web, web technology had taken a major step forward until Web 3.0. Developed as a military defense project in the late 1960s, ARPANET is the first step in evolution of packet switching in Internet (Leiner et al., 2009).

Web 1.0 made by web servers and with users not being able to change data, only possible features was to read, see and listen had a one way type. There wasn't any human interaction, and because of that Web 1.0 is described as static web. Moreover web sites were not sufficiently designed because of lack of technical information and education in the field. In brief, Web 1.0 was a platform to serve information on the web in a passive way.

With the help of Web 2.0, this structure had become a democratic environment in a way that users can interact with it. Web 2.0 provided the ability for users to make contents, edit them, and comment, in other words, the possibility of interacting had happened. In addition, progress in the field of designing had an improving effect in personal web sites. In addition, users could share any content they want in their personal web page.

As well as improvement in design and interaction, with Web 2.0, other platforms used to connect each other. That why Web 2.0 is also known as Mobile Web. In Web 1.0 users could only surf the web in desktop computers but with Web 2.0, it cover wider platforms such as laptops, mobile phones and tablet etc. This made Internet usable in any field including: educational, shopping, banking, finance, and even in

our daily lives. Facebook.com, Wikipedia.com, Twitter.com with the features they provide, had been ancestors of Web 2.0.

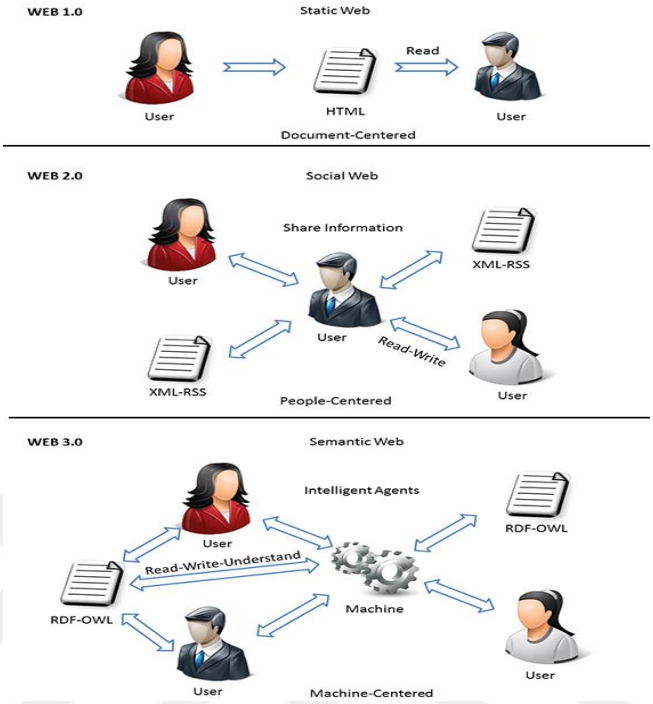

Figure 2.1 Evolution of World Wide Web (WordPress, n,d.)

Web 3.0 technologies and SW applications are mostly studied in the academic and industrial areas recently. According to Tim Berners Lee SW is an extension of current web, not another web (Berners-lee, Hendeler, & Lassila., 2001). Today's most of the web contents are created for human's consumption, so machines and computers are not suitable to understand these contents. The main aim of SW is to create more intelligent web contents, allows machine interaction, and processing information without any supervisor (Mohebbi, Ibrahim, & Idris, 2012).

2.2 Semantic Web

Computers have the ability to present the web content in a formal way however, they are not capable of understanding and interpreting them meaningfully. SW technology provides a computer environment than can be understood by software agents which could allow the machines to understand web contents itself. Creating SW content in an environment that can be understood by software

agents,(Frauenfelder, 2004). When considered from this point of view the semantic web can be considered as a global data network.

Although, SW is not artificial intelligence, it can use as an artificial intelligence technology. This technology can produce intelligent data which can be understood by computers. The terms of intelligent data refers to ability to solve well-defined problems only by well-defined operations on existing well-defined data of the machine. Rather than wanting computers to know human language, people should take effort to make more understandable data to enable the creation of intelligent data more easily (Maedche & Staab, 2000).

Despite the fact that the many SW applications are still being under development phase some of the completed applications are available today. DBpedia, CIA World Factbook, GeoNames are the examples of successful applications. But if all the Web content uses SW technology these applications would gain meaning, and work properly. One of the biggest problems in this phase is that some large parts of the data in Web 2.0 field are not converted to RDF format yet. When in content of Web 2.0 converts to RDF format, all the web technology will move to SW.

2.3 Semantic Web Application Areas

SW technologies can be used in a variety of application areas. Some of these areas are following.

2.3.1 Semantic Based Web Search Machines

Web content which are defined by ontologies, should inquire smarter than normal search engines. Swoogle, Onto Search are two of the many examples in this field.

2.3.2 Software Agent Based Distributed Computing Applications

Software agents will provide collation and use of data that are defined, structured and interpreted by ontologies. This will allow making most of the currently imagined applications become real.

2.3.3 Ontology Based Enterprise Information Management

With the global economy, in addition to traditional sources such as labor, capital, and inventory management, it's becoming so important to manage information as a resource of knowledge in the organizations and is emerging as important factor productivity. SW technologies will provide corporate information effectively managed and used.

2.3.4 Semantic Based Digital Libraries

SW technologies provide effective classification and indexing information. In this way providing to operate digital libraries each other and accessing the data in digital libraries easier.

2.3.5 Automatic Web Service Discovery, Activation, Mutual Operable and Traceability

Web services technology has recently been the most talked about and will lead to new opportunities in the web environment.

2.4 SW's Standards and Protocols

The standards of SW are being defined since 1994, within World Wide Web Consortium Corporation. Protocols like XML, HTML, XHTML, RDF, RDF-S, OWL, RIF and SPARQL are technologies have given by W3C until now (Figure 2.2).

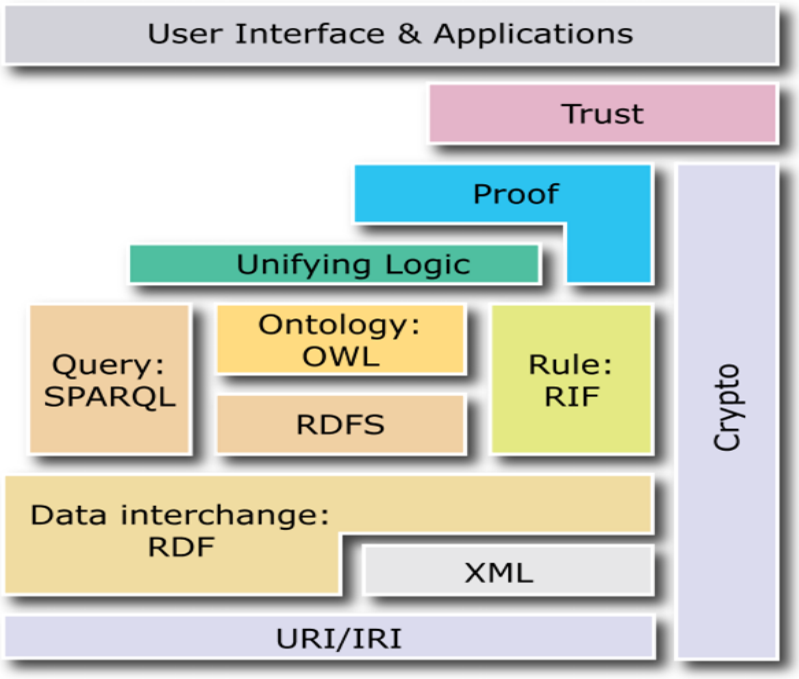

Figure 2.2 SW protocols which are defined by W3C (SW Layers, n,d.)

2.4.1 Universal Resource Identifier / Internationalized Resource Identifier

URI is a character set made by W3C, which is used to identify the name of a resource on the Web such as URL of internet site, document, image, table and etc. Syntax is made of protocol, domain, port, path, string and fragment id, which is written in order. From the point of duty and process IRI is nothing different than URI, but defined on an extended ASCII character set, this way it has support for more languages like Arabic, or Chinese.

URIs is very important, providing both the core of the framework itself and the link between RDF and the Web (Figure 2.3). W3C has defined two main URI standards based on SW (Sauermann, Cyganiak, & Völkel, 2008).

- Accessible by Web: Defined URI must be accessible by both humans and programs.
- Should Be Consistent: It's must not have conflict about if URI is defining concepts or documents. One URI should only point out a web document, or a real life concept.

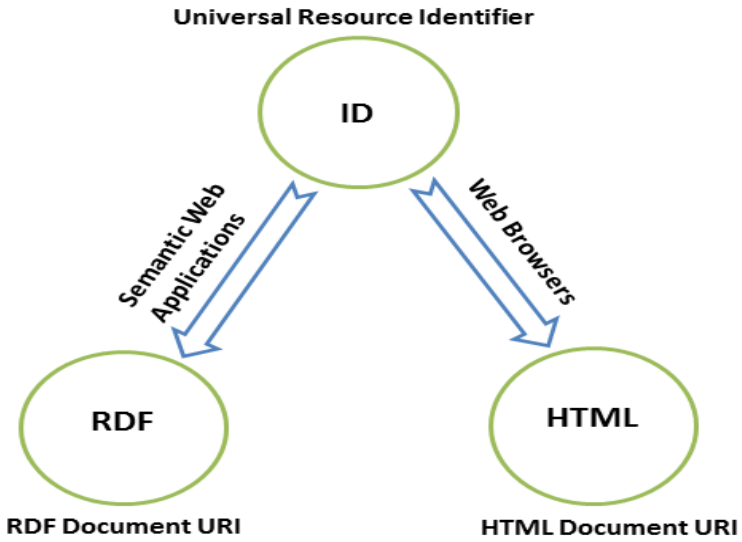

Figure 2.3 URIs is a link between RDF document and HTML document

2.4.2 Extensible Markup Language

XML stands for the Extensible Markup Language developed by W3C. Semi-structured data term is gained acceptance with XML (Kanne & Moerkotte, 1999). It is a meta markup language for some of web contents such as literal information and e-commerce requirements. It is used to make the process part of data transformation in data transformative systems. XML documents represent knowledge by using tree structure with some additional information. These tree structures compose of tags, elements and attributes (Figure 2.4).

As seen in Figure 2.4, there is a simple xml example including a root node represented BOOKS tag and there are 4 child nodes represented BOOK tag. Book node contains the ISBN information, and information stored directly on the node is called an attribute. The book's title and author information is stored in child nodes called title and author.

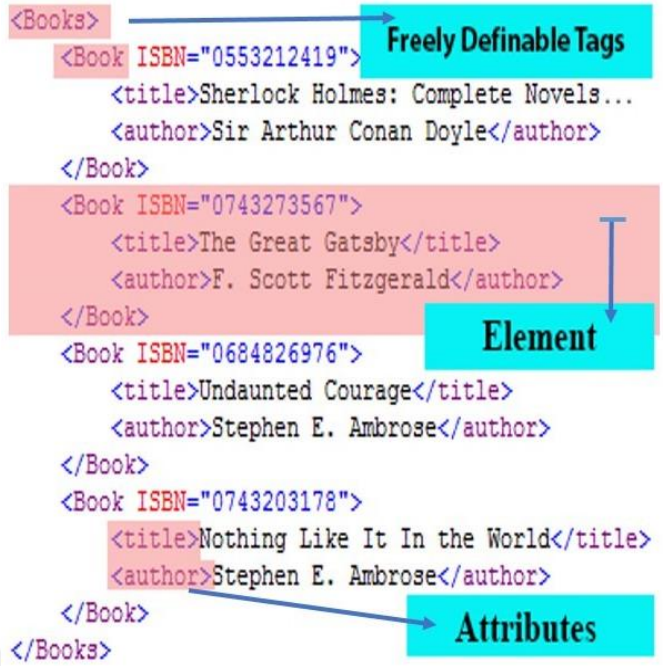

Figure 2.4 Example of XML document

Before XML standard invention, when moving data from a software or database to another platforms, there were a lot of problems encountered. To transform the structure of transferring set of information to other systems structure required complex process and took too much of time. Also there was a need for common markup language which is understandable by both humans and computer. XML is the standard to satisfy these needs.

HTML is the most used markup language, but it has a lot of problems such as some restrictions, hardly readably for human and etc. XML is developed to overcome the limitations of HTML (Benoît Marchal, n.d.) However XML is not a replacement for HTML. The biggest difference between XML and HTML or any other markup language is that developer can add self-defined by the way of users prefers thanks to supporting Unicode character system.

2.4.2.1 Advantage of Using XML

XML documents have more advantages than other type of documents which are unstructured (Schenkel, 2003).

- There are many additional tools to create or store data.
- Web contents created by XML are easy to read both from humans and machines.
- XML data types are very flexible and customizable. So, users define their own special tags that they can use endlessly.
- XML can be used as an exchange format to enable users to move their data between similar applications
- XML languages are widely used and supported by other languages and programs.
- XML is easily processed because the structure of the data is simple and standard.
- XML documents are semi-structured. This, XML can be use like a database and they are used for data transform.

Although there are many advantages of XML, it has some structural problems and lack of its limitations. Researchers which study in this area, think that these problems are solved by using RDF.

2.4.3 Resource Description Framework

Resource Description Framework (RDF) is standardized data model or language used for representing information on the Web. RDF is a family of World Wide Web Consortium (W3C) specifications and it is also used knowledge management applications (Punnoose, Crainiceanu, & Rapp, 2012). Main goal of RDF is providing data for applications rather than directly to human. In other ways RDF provides a software tool for publishing both human-readable and machine-processable vocabularies designed on the web (Miller, 1998). Another goal of RDF is data representing as a collection of <subject, property, and object> triples can easily be stored in a relational database.

Approach of RDF is based on identifying resources by using web identifiers and describing resources in terms of simple properties and property values. This way objects and concepts can be expressed with descriptions and values. RDF metadata

model uses URI, IRI or URI references (URIref) for identify resources. For this reason a source using RDF statement is everything that is identifiable by URIref.

2.4.3.1 Basic Notation of RDF Statement

RDF files have set of statement and these statements compose of three elements. These statements are known as RDF triples in literature. These triples are similar to the base sentences being used in daily dialog. RDF triples are used to express the given knowledge piece by piece (Jentzsch, Usbeck, & Vrandecic, 2014). These triples are given a follow.

- **Subject:** The part that identifies the thing the statement is about is called the subject. They may be URI reference or unnamed resource.

- **Predicate:** Describes some relations between resources. They also called an object property. They must be URI reference.

- **Object:** The part that identifies the value of that property is called the object. They may be URI reference, unnamed resource or literal information.

2.4.3.2 RDF Graphs on Example of RDF Statement

RDF graphs are used to visualize the statements of RDF. An RDF graph can be visualized as a node and directed-arc diagram, in which each triple is represented as a node-arc-node link as seen Figure 2.5. A statement is represented by a node for the subject, a node for the object and an arc for the predicate.

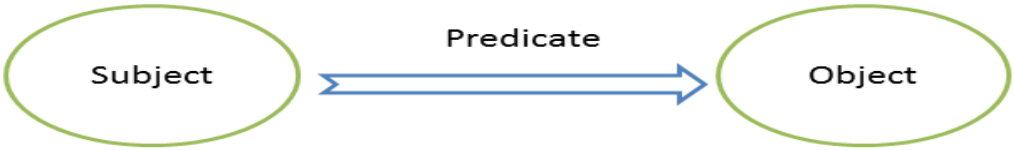

Figure 2.5 Simple statement graph template

RDF data model can be illustrated by concrete examples. Consider the following statement.

http://www.example.org/mehmet lives Istanbul.

In this statement http://www.example.org/mehmet is a subject; Istanbul is an object and live is a predicate. As shown that subject of statement is resource, but object is literal. But sometimes object of statement may be another resource. As shown in Figure 2.6 the predicate represents other resources. We write above example again.

http://www.example.org/mehmet live http://dbpedia.org/resource/Turkey/Istanbul

Figure 2.6 Simple statement graph template

2.4.3.3 Advantages of RDF

- RDF statements compose of triples so they can be implemented and store efficiently. Other models requiring variable-length fields would require a more costly and more cumbersome implementation.

- RDF reduces ambiguity. Because, global identifiers are used in RDF files.

- The RDF model is essentially the canonicalization of a (directed) graph, and so as such has all the advantages (and generality) of structuring information using graphs.

- RDF provides open world assumption so incremental data integration and data merging are easier.

- RDF syntax is layered thus the basic serialization syntax allows for quite a powerful encoding.

2.4.4 Resource Description Framework Schema

Resource description framework schema (RDFS) is representation of a type system that extends the data model of RDF. It defines a set of words in system that would be used in a specific field.

Table 2.1 RDF classes

Class Name	**Comment**
rdfs:Resource	The class resource, everything.
rdfs:Literal	This represents the set of atomic values, e.g. textual strings.
rdfs:XMLLiteral	The class of XML literals.
rdfs:Class	The concept of Class.
rdf:Property	The concept of a property.
rdfs:Datatype	The class of datatypes.
rdf:Statement	The class of RDF statements.
rdf:Bag	An unordered collection.
rdf:Seq	An ordered collection.
rdf:Alt.	A collection of alternatives.
rdfs:Container	This represents the set Containers.
rdfs:ContainerMembersh ipProperty	The container membership properties, rdf:1, rdf:2, ..., all of which are sub-properties of 'member'.
rdf:List	The class of RDF Lists.

RDF defines resources as classes, properties and values in the form of definitions. But upon these, application-specific classes and properties should be defined either. RDFS is used to determine the application-specific classes and properties. An RDF Schema declaration is expressed in the basic RDF Model and Syntax Specification and consists of classes shown as Table 1.1 and properties shown as Table 1.2. In other words, the RDF Schema mechanism provides a type system for RDF models, a

vocabulary of the valid terms that can be used to describe resources (Theoharis, Christophides, & Karvounarakis, 2005).

RDF Schema actually does not contain application-specific classes and properties, only provides a framework for them. RDF Schema classes are similar to the class hierarchy in object-oriented programming languages. This feature provides the ability to define resources as classes and sub-classes. Each RDFS is also a RDF resource.

Table 2.2 RDF properties

Property Name	**Comment**	**Domain**	**Range**
rdf:type	The subject is an instance of a class.	rdfs:Resource	rdfs:Class
rdfs:subClassOf	The subject is a subclass of a class.	rdfs:Class	rdfs:Class
rdfs:subPropertyOf	The subject is a sub property of a property.	rdf:Property	rdf:Property
rdfs:domain	A domain of the subject property.	rdf:Property	rdfs:Class
rdfs:range	A range of the subject property.	rdf:Property	rdfs:Class
rdfs:label	A human-readable name for the subject.	rdfs:Resource	rdfs:Literal
rdfs:comment	A description of the subject resource.	rdfs:Resource	rdfs:Literal
rdfs:member	A member of the subject resource.	rdfs:Resource	rdfs:Resource
rdf:first	The first item in the subject RDF list.	rdf:List	rdfs:Resource
rdf:rest	The rest of the subject RDF list after the first item.	rdf:List	rdf:List
rdfs:seeAlso	Further information about the subject resource.	rdfs:Resource	rdfs:Resource
rdfs:isDefinedBy	The definition of the subject resource.	rdfs:Resource	rdfs:Resource
rdf:value	Idiomatic property used for structured values.	rdfs:Resource	rdfs:Resource
rdf:subject	The subject of the subject RDF statement.	rdf:Statement	rdfs:Resource
rdf:predicate	The predicate of the subject RDF statement.	rdf:Statement	rdfs:Resource

2.4.5 Simple Protocol and RDF Query Language

RDF triples is the basic structure to access web content created in accordance to the RDF standards. If users have a RDF resource, they could access other RDF resources by using its properties. Because RDF triples make relations between resources. These way users can reach the information they need in the least possible time. But to access information among huge mass of data on the web causes the loss of both time consuming and labor. For these reason SPARQL query language has been developed to examine or queried web contents created RDF standards.

As it is shown in the Figure 2.6 its query type and working is similar to SQL's as seen Just as SQL provides a standard query language across relational database systems, SPARQL provides a standardized query language for RDF graphs or resources (Segaran, Evans, & Taylor, 2009).

There are four main commonly used query types supported by SPARQL; these are SELECT, ASK, DESCRIBE and CONSTRUCT queries. The explanation of these query types are given below;

- *SELECT* query serves to return the whole or a portion of the desired data from studied the data set to fit a given query pattern directly.

- *ASK* query returns the response that data is available which meets the query patterns in the data set or not.

- *DESCRIBE* query, URI returns the description of the data set that resource identified by the query pattern or directly to RDF.

- *CONSTRUCT* query, looks for patterns in the data set given by the query and again produces a schema (graph) matching with the query template (SPARQL Protocol).

Moreover, SPARQL also supports aggregation, subqueries, negation, creating values by expressions, extensible value testing, and constraining queries by source RDF graph (Erling & Mikhailov, 2009).

Table 2.3 Example of SPARQL query that answer the question "what are all the country capitals in Africa?"

```
PREFIX ex:  < http:// example.com/exampleOntology#>
SELECT  ?capital ?country
WHERE {
   ?x  ex:cityname  ?capital  ;
       ex:isCapitalOf  ?y .
   ?y  ex:countryname  ?country  ;
      exisInContinent   ex:Africa   .
}
```

2.4.6 Ontologies

Generally ontology is term used in the philosophy of science. In late 90's it is used as an artificial intelligence term. Ontologies are one of the components which have the most important role in creation of semantic web. There are many definition of ontology in academic field. Some of these definitions are given below.

- Shared formal conceptualizations of particular domains, ontologies provide a common understanding of topics that can be communicated between people and application systems (Decker et al., 2000).

- Ontologies are key requirements for building context-aware systems (Chen, Finin, & Joshi, 2003).

- Ontology can be used to define and specify spatial data semantically as well as machine understandably (Wang, Gong, & Wu, 2007).

- Ontology is a formal specification of a shared conceptualization (Gruber, 1995).

Ontologies should be well defined in order to understand by everyone and adapt themselves to new requirements. Well-defined ontologies should be clear, reusable, interoperable, and scalable.

The main purpose of the ontology is to define reference set of concepts which are the same concepts that can be used to refer to the same thing. Furthermore it can be used to support variety of task in diverse research such as information retrieval, natural language processing, knowledge management and etc. Nowadays it can be used in industrial, commercial web sites. In order to use ontologies effectively, current web content should be converted to RDF format.

Ontologies are classified in various ways according to the ability of showing detail. One of the most widely accepted classification of ontologies has made by Guarino (Guarino, 1998). He divided ontologies into four basic categories. These categories are following (Figure 2.7).

2.4.6.1 Top Level (Upper) Ontologies

These types of ontologies describe general and abstract concepts or terms such as time, space, event, action and etc. These are independent from a particular problems or domains. Actually they are created with the expansion of the domain ontology. They are composed for very high number of user and their scope is very large. Some examples of upper level ontologies are Sensus (Swartout, Patil, Knight, & Russ, 1996), CYC (Lenat, 1995), CORBA (Mowbray & Zahavi, 1995) and WordNet (Miller, 1995).

2.4.6.2 Domain and Task Ontologies

These ontologies describe concepts or terms which are part of world, task or subject. These concepts related with a generic domain such as medicine, electronic or a generic task such as selling, and scheduling. They represent the concept more specialization than upper ontologies. However, scope of these represent narrower

than upper level ontologies. Some examples of domain and task ontologies are DOLCE, OpenCyc, SUMO.

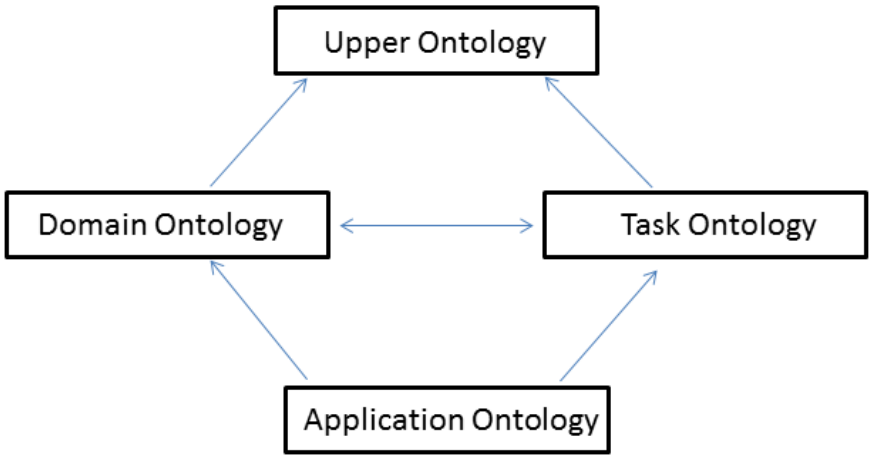

Figure 2.7 According to Guarino classifying of ontologies

2.4.6.3 Application Ontologies

These types of ontologies describe particular domain or particular task. Although they are narrower than domain and task ontologies, they represent the concepts more in detail than them.

2.5 SW and Ontology Components

Ontologies consist of a variety of components. Some of ontologies compose of only individuals, classes, attributes, and relationships. They have also known as light-weight ontologies. Additionally ontologies may contain other components such as function terms, restrictions, rules, axioms, and events. These ontologies called heavy-weight ontologies. Some of these components are explained below.

2.5.1 Classes

They are also known as concepts, terms, types, collections and etc. They are the most important component of the ontologies. They provide common structure for a group of similar object. They are in a hierarchical structure so they may have

subclasses or parents. They can contain values or value restrictions for properties and relationships. Classes in ontologies are similar to any object-oriented programing language classes such as C# or java. As it is shown Figure 2.8 Person, Module, and Document are example of classes. Generally Object class or Thing class are root class in ontologies.

2.5.2 Individuals

They are basic level component of ontologies. The also known as instances, elements, objects, particulars, and etc. They are specific member of class. For example Mehmet is individuals of class Lecturer in Figure 2.8. Limitless individuals which belongs only one class may be created in ontology.

2.5.3 Relationships

Relationships also called object property represent a type of interaction between classes of the ontology (Benjamins & Gómez-pérez, 1999). Ontologies include additional types of relations such as Join, Read, is_about knows in Figure 2.8 but some of types are standard such as is-a, is-a-part-of, is-a-subclass-of, instance-of and etc.

2.5.4 Properties

They are also known as attributes and data property. They are used to describe classes by assigning their attributes. They have type, label, and value. The object property represents a property which links between classes, while the data property references a literal value. In Figure 2.8, Lecturer class has two properties called Faculty and Tel_Num.

2.5.5 Functions

They are a special case of relations. They can be used in place of an individual term in a statement.

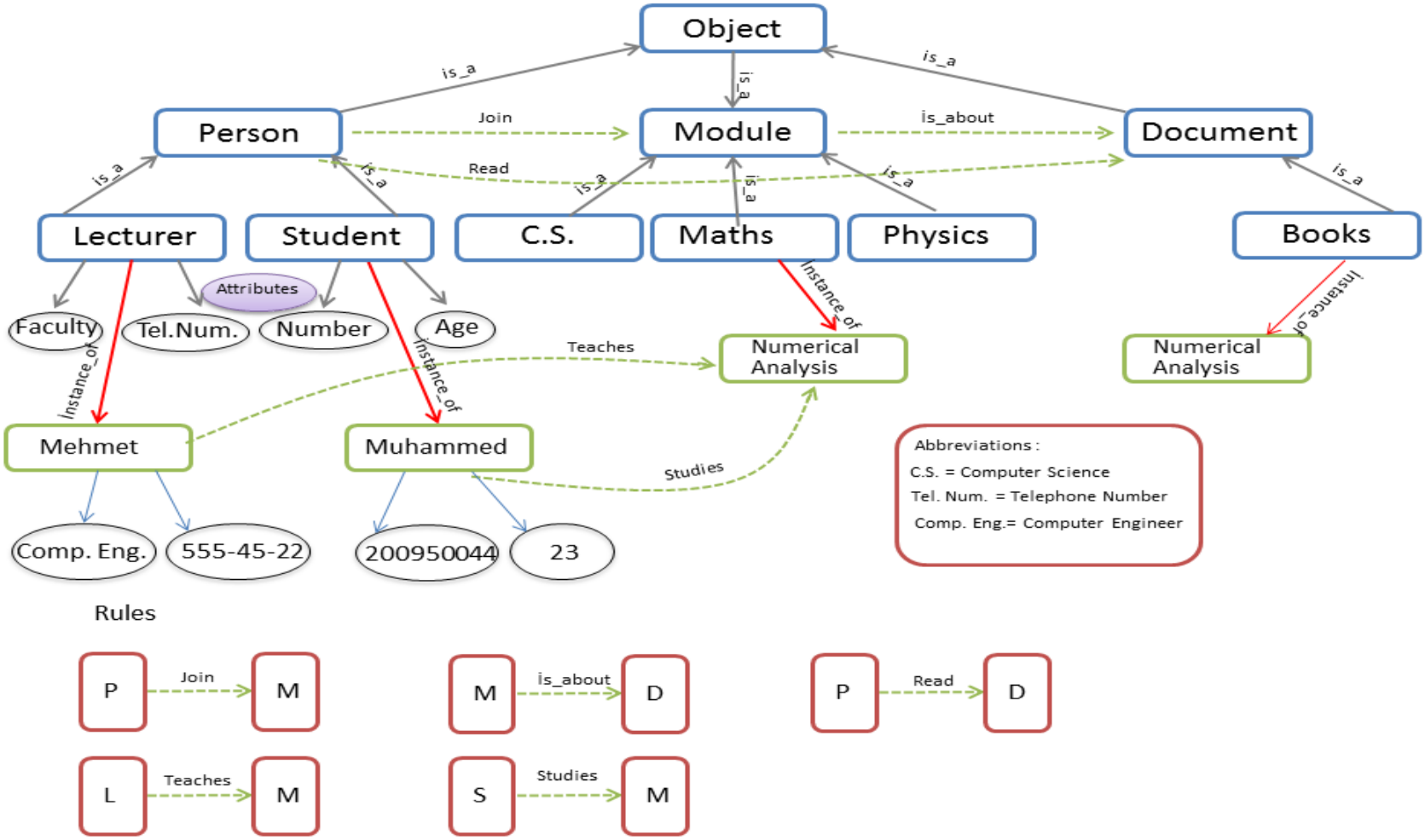

Figure 2.8 Sample ontology model

2.5.6 Axioms

They are used to model sentences that are generally true. They verifying the correctness of the input information specified in the ontology or deducing new information (Winer, 2011).

CHAPTER THREE
RECOMMENDATION SYSTEMS

Rapid development of Internet usage today, increasing the quality and diversity of services offered to users has led to continual rise of data on the Internet day after day (photos, music, movie, blog etc.). It became a major problem how to store the collected data and how to interpret with them when using again. Among the huge mass of data finding a product they wanted has become impossible with classical methods for users. Before the recommendation systems (RS) in classical methods web sites were designed to present the same content for users without any customization. Today, this web sites present their products considering users of some information such as gender, age, etc. and product features such as price, color, height, etc. Thus RS help users find the product they want without dealing with huge collections of items. RS are used by websites, while offering user specified suggestions.

The main goal of RS is to help users in finding their way through huge databases and item collections, by filtering and suggesting relevant items considering the users preferences such as tastes, interests, or priorities (Bellogín & de Vries, 2013). In literature, as it will be made many definition of RS but there are a few most commonly used definitions. Some of these definitions are following.

- RS are software tools and techniques providing suggestions for items to be of use to a user (Sharma & Ugrasen, 2012).

- RS produce a ranked list of items on which a user might be interested, in the context of her current choice of an item (Debnath, Ganguly, & Mitra, 2008).

- RS are a personalized information filtering technology used to identify a set of items that will be of interest to a certain user (Deshpande & Karypis, 2004).

- RS are computer-based intelligent technique to deal with the huge mass of information on internet and product overload (Vozalis & Margaritis, 2003).

Another goal of RS is to eliminate the need for browsing the item collections by presenting the user with items of interest (Nathanson, Bitton, & Goldberg, 2007). The origin of the RS is based on information retrieval. Developing of these systems, since foundation, at 1990's center became a separate research topic (Adomavicius & Tuzhilin, 2005). But recent years RS are complex systems containing different techniques such as text mining, artificial intelligent, machine learning, text analysis, semantic methods and etc. Nowadays recommendation plays an increasingly important role in our daily lives. RS are used many variety of web sites commonly. However RS are becoming an important commercial software tool some e-Business web sites. Thus millions of companies are implementing this systems into their sales strategy. Companies use RS showed an increase sales and profit. RS gave a strategic advantage over companies not use this systems. The companies which use these systems are increasing their income, by finding products which as more chance to be liked by customer(Rashid, Lam, Karypis, & Riedl 2006).

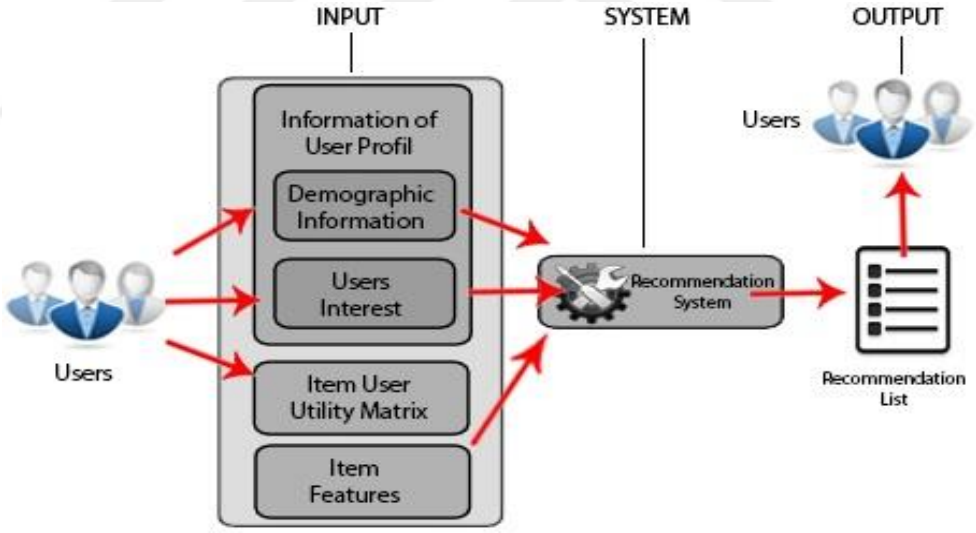

Figure 3.1 Model of showing how RS work

Web sites use the RS that suggests a special product to the user. RS collect some information about user's interest while they surfing and shopping on the company's web sites. Some RS use users demographic information, user interest, product features, others use both of them (Figure 3.1.). These systems suggest product that users have not seen before by using these information. Companies suggest a wide

variety of items through the e-commerce web sites, some companies offer product in a million of items or collections.

RS can be applied to many areas. However, product RS are commonly used in e-commerce, music, books, document and news. But field of use has been greatly extended in recent years. RS has become usable in restaurants, hotels, financial services, social networks and insurance companies. For companies the main purpose of using RS by companies is to increase the sales. Many of them are gaining advantage to them self and customers by using RS.

3.1 RS Application Areas

RS technologies can be used in a variety of application areas. Some of these areas are following.

3.1.1 E-Commerce Applications

These applications are sites which RS are mostly used. They filter products to which users could like more and recommend them to users. RS increase their income by turning potential buyers which navigate sites into a real buyer, cross-selling and ensuring commitment (Schafer, Konstan, & Riedi, 1999). For example Amazon, and eBay are the most known e-commerce sites used these systems.

3.1.2 Content Based Applications

Filtering newspaper, document, articles or email. People nowadays, instead of buying magazines or newspapers in the morning, prefer to read it from the source website in any other time of the day using their mobile phones, computers or other devices which can access the internet. In these sites, mostly content which is related to the users previously read articles are recommended. News sites, academic articles are examples of these websites such as BBC, CNN, etc.

3.1.3 Entertainment Applications

Film, music, video, radio and television programs. Like e-commerce applications they filter these entertainment product to which users could like more and suggest it to users. YouTube, and Movie Lens are most known examples of these sites.

3.1.4 Service Industry

Hotel, travel, online ticket reservation and selling, and etc. The best examples are resort agents and airline companies. Booking, and TripAdvisor are the good example in this applications.

3.1.5 Social Network Applications

Recommending friends or applications in social networks which are mostly used nowadays. Facebook, Twitter, Instagram are mostly used social network sites.

3.2 Advantages of RS

RS, since the beginning of their usage, have strategic advantage both for companies which use these systems and for users which navigate their website.

3.2.1 For Company

RS technology provides advantages in many areas for company. Some of these advantages are following.

3.2.1.1 Increases Sales

RS greatly increase selling incomes for companies by suggesting them product which have more chance to be liked. According to the experiences of companies that use RS, sales are expected to increase between 10 to 35 percent. According to 2006 sales figures, 35% of Amazon's sales are done through RS. Netflix in 2012 reported that 75% of what its users watched came from recommendations.

3.2.1.2 Cross-Sell

Cross-selling is the action of selling an additional service or products which are relevant current item to an active customer for increase sales. One of the biggest problems of e–commerce sites are users which only buy same product and not be interested in other products. These systems can make the user buy any other product which is similar to the product their buying by recommending it. For example in daily usage: When you want to buy a product, users which previously bought a product or checked that, also bought or checked B product. This way chance of selling other item after suggesting get increased.

3.2.1.3 Loyalty

When you bought a product from an e-commerce website, and you are satisfied from the product itself and after sale service, in another time when you need similar product, you will prefer to buy from that site. In addition to that, when you enter a website, if it knows you, and knows what products you prefer, and suggest you products with high accuracy, you will feel valuable.

3.2.2 For Costumer

RS technology provides advantages in many areas for Costumer. Some of these advantages are following.

3.2.2.1 Prevent Loss of Time

RS prevent wasting of time for searching billions of data by selecting products which user could be interested in from huge mass of data and item collections.

3.2.2.2 Help to Find Right Product

There will be many related and unrelated products in list for a user which is searching a product in a site which does not use recommendation systems. More choices will confuse the users and may be forced to buy product that is lower than the standards of product which he/she was going to buy. But RS is filtering most of unrelated product, so they prevent to choose wrong product to users.

3.2.2.3 Confidence

One the other problems in internet selling is confidence problem between site and user. When recommendations for the user is more related to that specific user and has high accuracy, user confidence will increase.

3.3 Data Collections

When websites try to recommend products for users use the information user gave when signing up to the website, or products he or she bought, rated and traces left when viewing contents on the web. The traces they left can help in guessing what to suggest from products which they had never seen but have a chance to be liked and bought. In this context, RS are suggesting products with using of information filtering systems by using traces of users left when surfing in the website (Belkin & Croft, 1992). Data can be collected from user in two ways as explicit and implicit.

3.3.1 Explicit Data Collections

In explicit data collection, user must perform in action. They are self-assessments made by answers of questions posed directly to users. If a user leaved a comment for a product or rated a product, is explicit kind of information. Although these data are healthy but in practice they are not much usable. Because gaining these information from user needs time and users don’t spend time on these things. Most of the time, product evaluation forms of users remain empty (Figure 3.2).

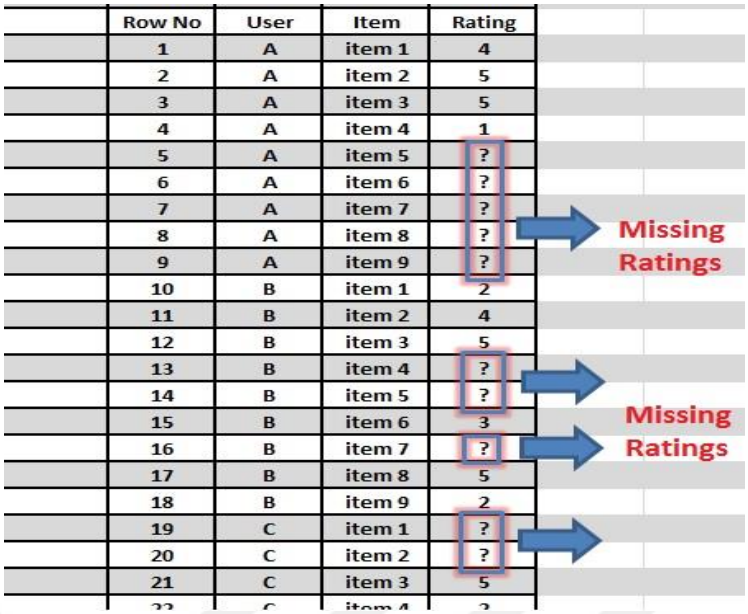

Figure 3.2 A variable set which shows products that user evaluation and missing ratings of products

Examples of implicit data collection include the following;

- Asking a user to rate an item on a sliding scale.
- Asking a user to search.
- Asking a user to rank a collection of items from favorite to least favorite.
- Presenting two items to a user and asking him/her to choose the better one of them.
- Asking a user to create a list of items that he/she likes.

3.3.2 Implicit Data Collections

In implicit data collection, tracking technology gathers behavioral data. Implicit data sets are kind of interpretation. By analyzing user behavior within the system, trying to understand in what that specific user has interest. For example, if a user always watches same kind of movies when he or she is online, it can be understood that he or she likes that type of movies without directly need to comment or rate any movie. Although amount of these type of data in comparisons to other types are much more, but these are all assumptions. Their accuracy is open to discussion in

comparison to explicit data type. Examples of implicit data collection include the following:

- Observing the items that a user views in an online store.
- Analyzing item/user viewing times.
- Keeping a record of the items that a user purchases online.
- Obtaining a list of items that a user has listened to or watched on his/her computer.
- Analyzing the user's social network and discovering similar likes and dislikes.

3.4 Recommendation Approaches

RS are one of the most studies areas intensively in industrial, academic, and educational fields. There are many approaches in literature. However, the most know approaches are content based filtering (CBF) and collaborative filtering (CF). Studies show that both gave good results. New approaches have emerged as a result of these studies.

In CBF, items are grouped in specific properties. When user register a system firstly user profile is created for every user. User profile is defined by items which user examined, liked or bought before. Based on this user profile, list of item recommendation is defined. Pure CBF ignore the preferences of other users (Schein & Popescul, 2002).

Approaches has advantages and disadvantages in comparison to each other, as in every field. Previous decades many researchers revealed hybrid systems that these two approaches are used together to eliminate the their disadvantages (Resnick, Varian, & Editors, 1997). The results of their experiments performed has proved that rate of success of hybrid approaches to be higher.

RS are classified according to their prediction approach. The recommender systems can be divided into five main categories (Figure 3.3).

- Content based filtering (Pazzani & Billsus, 2007), (Barranco & Martínez, 2010), (Chen, Jang, & Lee, 2011).

- Collaborative filtering (Herlocker, Konstan, Borchers, & Riedl, 1999), (Gong, 2010).

- Knowledge based systems (Burke, 2000), (Felfernig, Friedrich, Jannach, & Zanker, 2006).

- Hybrid systems (Burke, 2002), (Salter & Antonopoulos, 2006).

- Semantic recommendation (Ruotsalo, 2010), (Pukkhem, 2013).

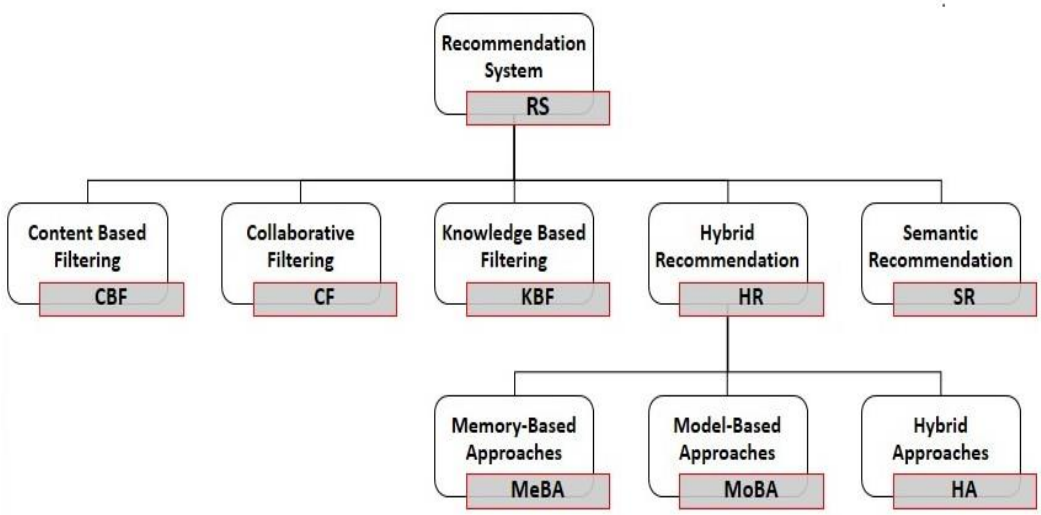

Figure 3.3 Recommendation approaches

3.4.1 Content Base Filtering

In RS the roots of the content-based approaches is based on information retrieval and information filtering research (Baeza & Ribeiro, 1999). Firstly, text based applications as documents, websites, news contents and messages are created on this subject. Content based systems only use active users preferences, instead of using preferences of all users (Balabanović & Shoham, 1997).

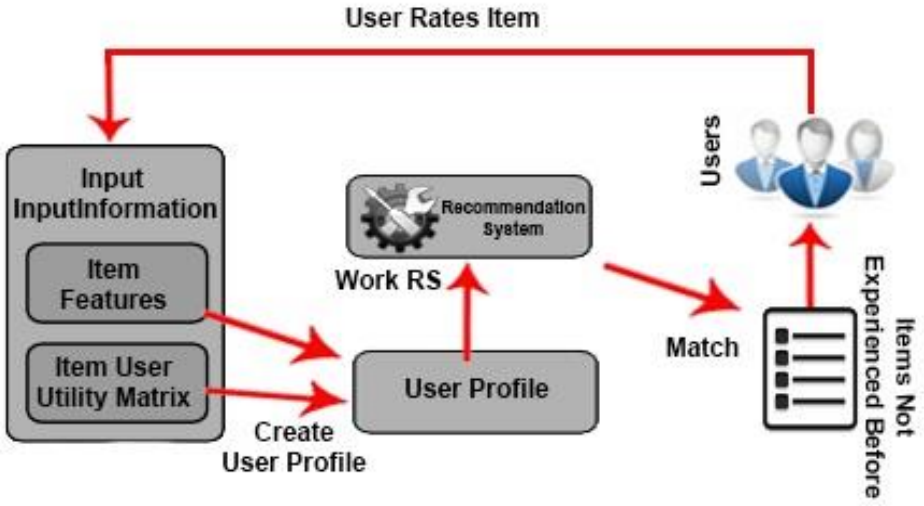

Figure 3.4 Model of showing how CBR systems work

CBF systems utilize past ratings u(c,si) of user c when suggesting u(c,s) which is interest level of user c to item s that user c has never seen before. siєS denotes the set of items that have similar properties of item s.

A profile is created for every user in the system. This profile is based on evaluations, preferences user made previously and sometimes demographic (sex, age, education level) details (Figure 3.4). This system works by comparing the user profile vector and the item properties vector with each other. In the result of this process, item similar to user profile is recommended to the user. As example, in a movie recommendation system, specifications of the movie user watched in the past are being compared with other movies that user didn't watch. Movies with similar characteristics are recommended to the user. If there is not a profile about the user, items that user viewed or rated previously are used to create a profile.

3.4.1.1 Advantages of CBF

CBF approach provides advantages in many areas for information sector. Some of these advantages are following.

- **Independent from Other Users:** In CBF, because of using only active user's reviews, when creating user profile vector, recommendation will happen independent from other users.

- **Clarity:** It can be found that how recommendation lists are created and how systems work using content properties and definitions.

- **New Item Problem:** Recommendation done is based on items properties so a new item can be suggested using its own properties. Thus, content based algorithms don't have new item problem and can recommend new item.

3.4.2 Collaborative Filtering

CF is a filtering type in which, to find how a user will rate an item which is never seen before, by comparing previously given rates of that user with other users' rates. This technique is based on, similar users have similar tastes. Users which rate items similar to the rates of active user, have similar tastes with active user. A user is similar to active user based on how much rates of user in the system is close to rates of active user. In guessing stage, the users which are most similar to active user, will have active role when deciding what items active user could like.

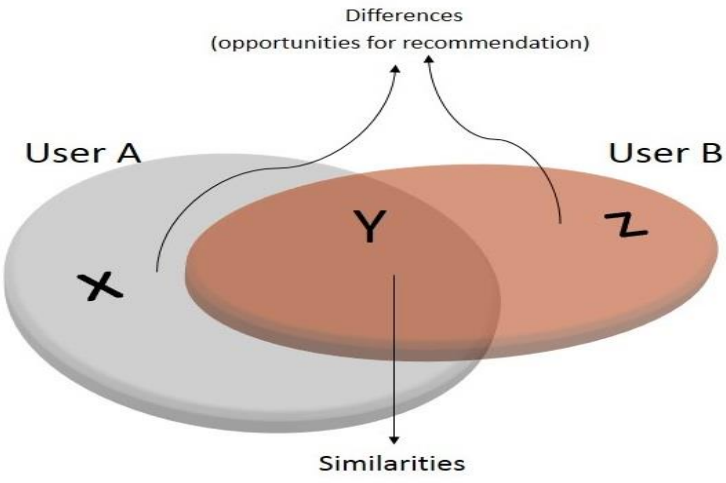

Figure 3.5 The representation of principles of collaborative filtering

Where A (X + Y) is the set of items rated by Alice and B (Y + Z) is the set of items rated by Bob. Region Y is the set of items rated by both users. Region X is the set of items rated by Alice but Bob has never seen or tried before. Region Z is the set of items rated by Bob but Alice has never seen or tried before. The CF says that the similarity calculation made through the region Y. At the result of calculation there is positive correlation between Alice and Bob we define they are similar users, so most probably they have similar tastes. According to this inference Alice most probably likes the items which are in region Z (Figure 3.5).

In Collaborative filtering systems, rates of users about items are stored in matrix. There could be many user and products registered to a system. Since size of these matrices are huge, it can sometime cost too much As seen in Table 3.1 if there are m items and n users in the system, total user rating of this system will be O(mXn).Since it is impossible for very user to review and rate every item, big part of these matrices are empty.

Table 3.1 User item matrix

		ITEM						
		Item 1	**Item 2**	**Item 3**	**Item 4**	**Item 5**	**…**	**Item m**
USERS	**User 1**	3	?	5	3	4	…	?
	User 2	?	3	?	1	?	…	4
	User 3	1	?	5	2	4	…	2
	User 4	5	2	1	?	?	…	5
	User 5	?	?	1	5	4	…	4
	…	…	…	…	…	…	…	…
	User n	5	1	?	?	2	…	?

CF systems utilize past ratings u(cj,s) of user c when suggesting u(c.s) which is interest level of user c to item s that user c has never seen before. cj ε C denotes the set of users that much similar to active user u.

CF algorithms are divided in to two categories as: model based and memory based approaches (Breese, Heckerman, & Kadie, 1998). Both of two approaches have advantages and disadvantages in ram requirement, speed, guessing accuracy,

reusability, easy understandable and in other ways. In any system, to define efficiency of the approaches to choose, filed of usage (item, movie, education and etc.), size of item-user matrix (user number and item number) and similar criteria should be considered to decide.

In CF, when reviewing similarities between users consider item-user matrix's rows are items and columns are users, rates given to each item are distinct line vectors. In another words, when calculating similarities, similarity between users are calculated. These approaches are named user based (user-user) Collaborative filtering.

Table 3.2 User-user row vector and item-item column vector

		ITEM						
		Item 1	**Item 2**	**Item 3**	**Item 4**	**Item 5**	**…**	**Item m**
U S E R S	**User 1**	3	?	5	3	4	…	?
	User 2	?	3	?	1	?	…	4
	User 3	1	?			4	…	2
	User 4	5	2	1	?	?	…	5
	User 5	?	?	1	5	4	…	4
	…	…	…	…	…	…	…	…
	User n	5	1	?	?	2	…	?

Another approaches is item based (item-item).In this approaches, similarities between products are reviewed. When reviewing similarities between items, each rate of items from each user are calculated as distinct column vectors (Table3.2). In another words when calculating similarities, considering only columns vector.

3.4.2.1 Advantage of Collaborative Filtering

CF approach provides advantages in many areas for information sector. Some of these advantages are following.

- **Independent from Item Profile:** These systems do not need item profile and they are not obligation to keep detailed information about the properties of the

item. Thus, there is not any comment or information about products in these systems.

- **Independent from User Profile:** Pure CFs do not based on user's demographic information when they suggest the item for users. So these systems do not need this information.

3.4.3 Knowledge Based Filtering

CF algorithms only use rates which users give to items but, content based algorithms use user profiles and item properties information. Both approaches have advantages and disadvantages in comparison to each other. But, there are situations which both approaches are not sufficient. So, choosing one of these approaches each time may not be the best choice. For example, we don't purchase house, car or computer frequently. Thus, user-item matrix is not used actively. CF and CBF algorithms does not produce decent results where there are small number of ratings.

Moreover, time factor plays an important role for RS. As time passed, accuracy and validity of the recommendation could be decreased. For example, in a RS for a computer, rating from 5 years before may not be efficient using content based algorithms (Jannach, 2004). Generally in these situation KBF is the best perform than other approaches.

3.4.4 Hybrid Recommendation Systems

Hybrid recommendation systems are produced by combining two or more recommendation approach. These systems are mostly created by combining content based filtering collaborative and knowledge filtering. Hybrid approaches mostly are created when other approaches are insufficient. Hybrid systems are produced to solve disadvantages) of other approaches such as cold start, data sparsity, limited content analysis (Miranda,Claypool, & Gokhale., 1999).

In many articles, when comparing HS and pure CF, pure CBF, and KBF methods, researchers shown that hybrid systems produce more accurate results (Soboroff, 1999), (Melville, Mooney, & Nagarajan, 2001). With combination of CBF and CF approaches in different ways, approaches below are emerged. Although RS have advantages but also it has missing components to be resolved and aspects to be improved.

The next generation RS must observe better user movements, item content information should be interpreted better, item information should be in a way that both understandable by computers and humans, should contain dynamic methods that can be integrated to any system and the accuracy of the results found should be based on a measurable basis. With considering all these problems, semantic recommendation approach is seen as a solver to the problems listed section 3.5.

3.4.5 Semantic Recommendation Approach

SW has been extensively studied in both academic and industrial means, which is discussed in the field of Web 3.0 in last years. SW systems create important data model not only for web based systems but also for other information systems. In semantic recommendation approach the recommendation process is generally based on concept diagram or an ontology describing acknowledge based and uses SW technologies (Figure 3.6).

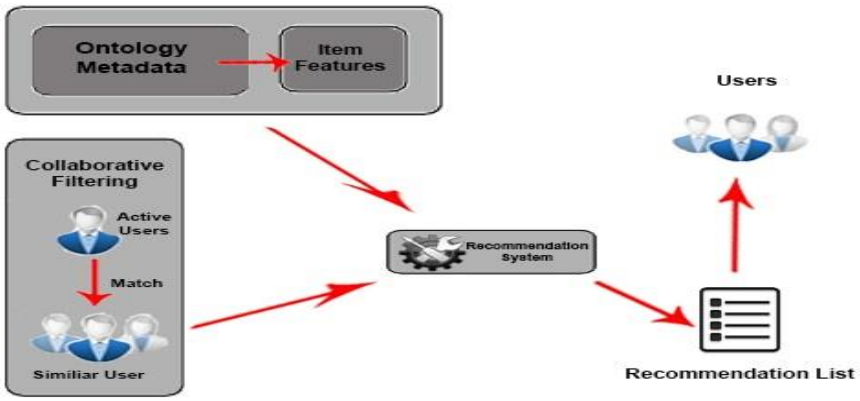

Figure 3.6 Model of showing how SW work

Ontologies which show certain concepts in a domain and relation between them are frequently being used is a form of knowledge representation nowadays. SW can appear in the category of knowledge-based recommender systems. Because, the SW systems are based on a knowledge-base. Ontologies which are base of SW is believed to solve problems listed above about recommendation systems. SW systems are used cold start and data sparsity problems of CF system (Wang & Kong, 2007).

3.5 Problems of Recommendation Systems

Although RS approaches intensively used in academic and industrial studies, still they have unsolved problems. The common problems for RS approaches are cold start (new item, user and system), data sparsity, scalability and limited contend analysis. These problems are explain briefly;

3.5.1 Cold Start

Cold start problem in recommendation system can be divided into three categories; new system, new user, new item (Milli & Milli, 2015).

3.5.1.1 New System

When establishing new RS there is no data about user preferences, so it is difficult to give the good advice. The user's rates items over time and the input data of the system increases. Thus allows RS to give better advice. We think that SW cope with this problem until the system collects enough data.

3.5.1.2 New User

When a new user registration there is no history of this user, so the system couldn't predict what the new user interested in. To deal with this problem some RS want to the user to rate a set of item when registering. However most of users do not

want to rate any items due to their insufficient time. We think that semantic web cope with this problem until the new user rate some items.

3.5.1.3 New Item

Like new user problem when an item is added the system, there is no past information of this item, so the system cannot recommend it to the user. This problem refers to new item problem in literature. We think that semantic web approach work out new item problem until the item is rated by some users.

3.5.2 Data Sparsity

In RS, when considered number of products and number of users, although it seems to be working with huge matrix, but most of the matrix discussed is empty (Figure 3.7). Data set which we try to produce suggestion on, systems recommendation power and correctness of the recommendations are increased as follows. Wang Shuliang and his friends, combined cloud model with CF, to solve this problem. Thus they increased correctness of recommendation in extreme sparse datasets (Wang, Xie, & Fang, 2011).

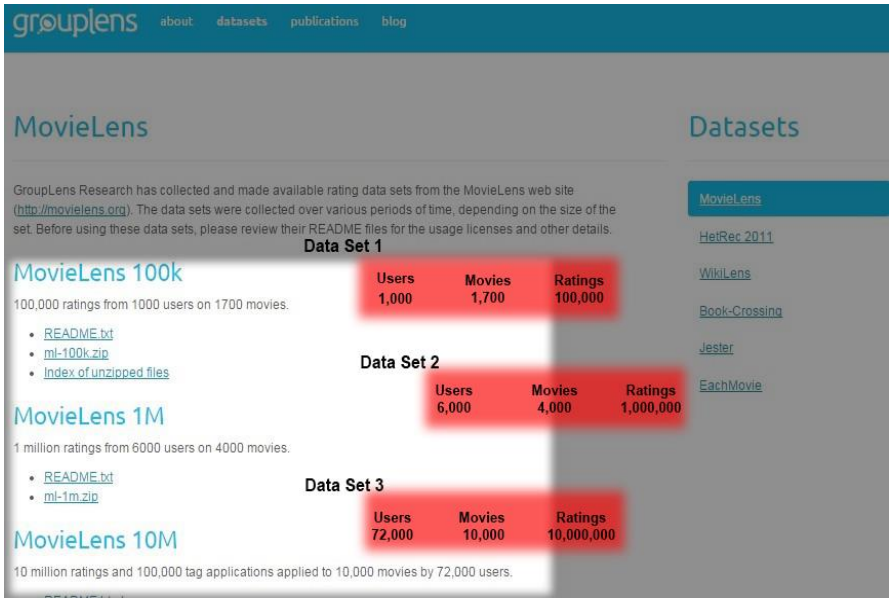

Figure 3.7 Three dataset accessed by everyone on MovieLens

Many companies bring into data set to researchers especially movie and music data set in order to study on it. When analyzing all of movie data set, it can be seen that they are extreme sparse such as MovieLens data set which is used many times by researches. The statistical details of these data set are given Table 3.3.

Another approach to cope with this problem is to use demographic information (gender, age, area code, education and employment information) to find users similar to active user (Pazzani, 1999).

Table 3.3 Example of sparsity data set

Source	**Dataset**	**Users**	**Items**	**Total Rate**	**Expected Rate**	**Sparsity Rate**
Movie Lens	Data Set 1	1,000	1,700	100,000	1,700,000	94.1177
Movie Lens	Data Set 2	6,000	4,000	1,000,000	24,000,000	95.8333
Movie Lens	Data Set 3	72,000	10,000	10,000,000	720,000,000	98.6111
Netflix	Prize Data Set	480,189	17,770	100,480,507	8,532,958,530	98.8225

3.5.3 Over-Specialization

In CBF only items which are similar to user profile are recommended. Thus, same items are recommended to user many times. For example for a user which never visited a Chinese restaurant, another Chinese restaurant will never be recommended. To solve this problem, genetic algorithms are used in order to bring different items in similar to user profile item list by some researchers (Sheth & Maes, 1993).

3.5.4 Scalability

User-item matrix are not static matrixes. Each day, rows (user) and column (item) of matrix and rates are changed. Thus, with increase of complexity of systems which contain thousands of users and items there appears to be a serious scalability issues.

3.5.5 Limited Content Analysis

In CBF, items name and type have natural limits. If there is not enough properties about items user is interested in or not, none of CBF systems will produce accurate solution.

Another problem in analyzing content is that, if two different item are defined with exact same properties, difference between these items will not be understandable. Especially in text-based systems, because properties are defined as important keywords in the document, content based filtering cannot differ a well-written document and poor document (Shardanand & Maes, 1995)

CHAPTER FOUR
DESIGN AND IMPLEMENTATIONS OF ONTOLOGY BASED RECOMMENDATION SYSTEM IN E-LEARNING FOR TURKISH (OBReSET)

In this chapter a system named ontology based recommendation system in e-learning for Turkish (OBReSET) is develop in order to an information content management application which is used for guide primary and secondary school students to suitable web content for them using SW technologies and RS.

Firstly overview of the model is given. Then the architectural design details of OBReSET which include the SW analyzing, proposed CF method, and proposed hybrid approach are introduced. In section 4.2 SW analyzing is introduced. CF approach used in OBReSET is mentioned in section 4.3. Finally, proposed hybrid approach is explained in section 4.4.

4.1 Overviewed of the Proposed Model

Figure 4.1 shows proposed system structure. The proposed system consists of 3 main phases. First phase is creating ontology about primary and secondary school lessons and computing semantic analyses. In this phase, we generated set of items by using semantic similarity measure that utilizes taxonomy similarities between items.

Second phase is item based and user based CF. In this phase, most similar items set are generated among which never seen before with using of user's previous preferences. Firstly we calculated similarity between items, then selected nearest neighbors, and lastly we generate prediction list for active users. Every process of item-based collaborative filtering we used a variety of methods. In chapter 5, we compared and discussed about accuracy and performance of method with considering result from calculation.

Last phase, is implementing parallelized hybridization design approach to our project. This phase is combining first and second phase.

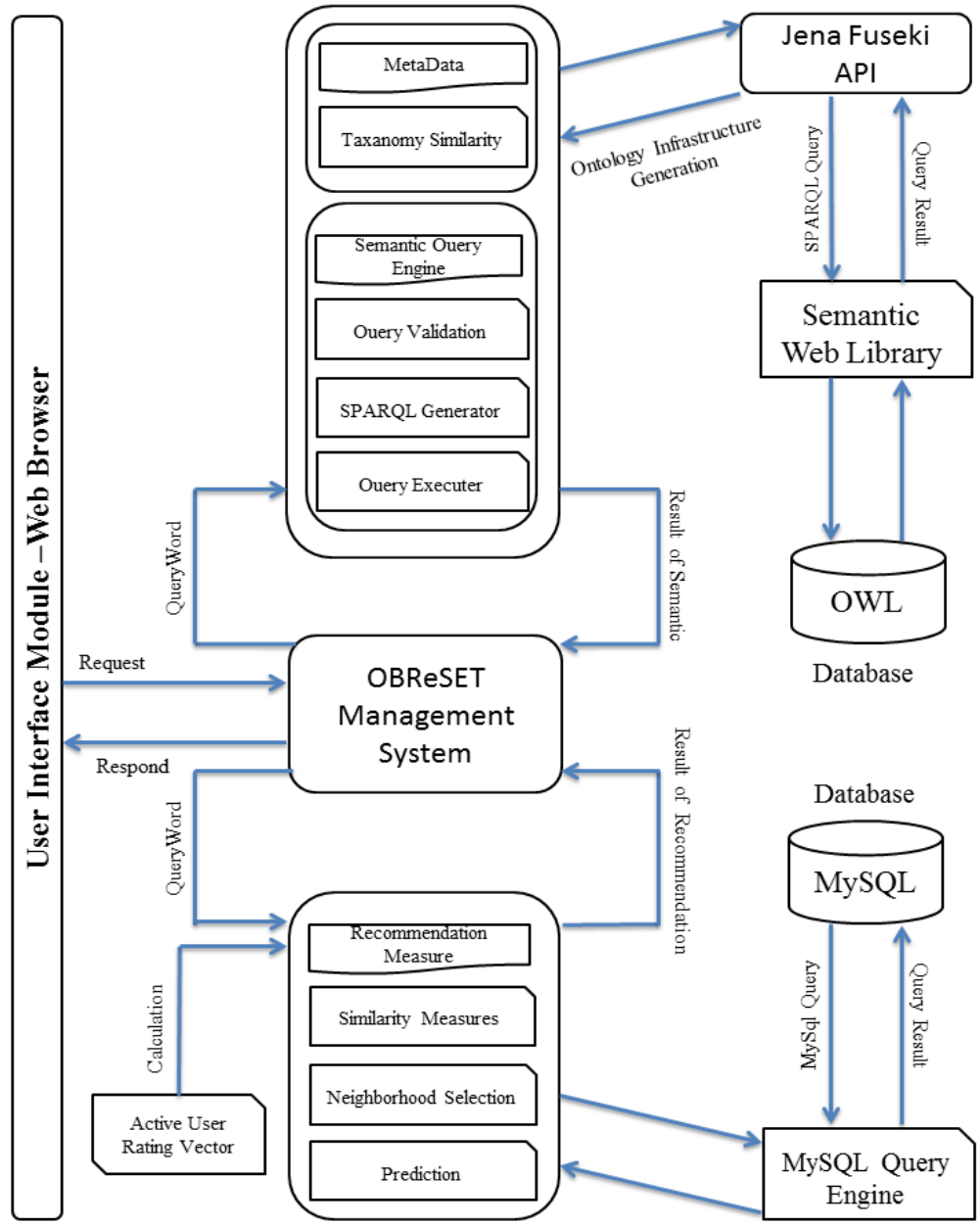

Figure 4.1 Ontology based recommendation system management framework

4.2 Semantic Similarity Calculation

We use semantic web technology and ontologies for reducing the data sparsity and cold start problems of pure collaborative recommendation. Thus we deal with some limitations of RS systems. At first ontology of this domain was created to calculate semantic similarity between science and technology lessons. In ontologies, there are 3 different ways to calculate similarity between concepts. These are taxonomy similarity, relation similarity and attribute similarity. Only taxonomy similarity was used in our project. Because there are not any relation and data property in our project. Taxonomy similarity calculates based on hierarchical order in between concepts.

A number of semantic similarity approaches have been developed in the previous decade. We calculate the semantic similarity measure by using "IS-A" taxonomy. We utilized two different approaches from the studies of Wu & Palmer and Li, Bandar, and Mclean in our thesis in order to calculate semantic similarity. We utilized the taxonomic similarity to resolve syntactic ambiguity. These approaches are described below.

4.2.1 Wu & Palmer Approach

The first method which we used in our thesis is Wu and Palmer measure (Wu & Palmer, 1994). This method based on the distance between two concepts in ontology. The distance is showed in Figure 4.2. Taxonomy similarity of Wu and Palmer approach's measure is given below;

$$Tax.Sim.Cal._{Wu\ \&\ Palmer}\ (C_1, C_2) = \begin{cases} \dfrac{(2N_3)}{N_1 + N_2 + 2N_3}, & if\ \ C_1 \neq C_2 \\ \\ 1 \quad , & if\ \ C_1 = C_2 \end{cases} \tag{4.1}$$

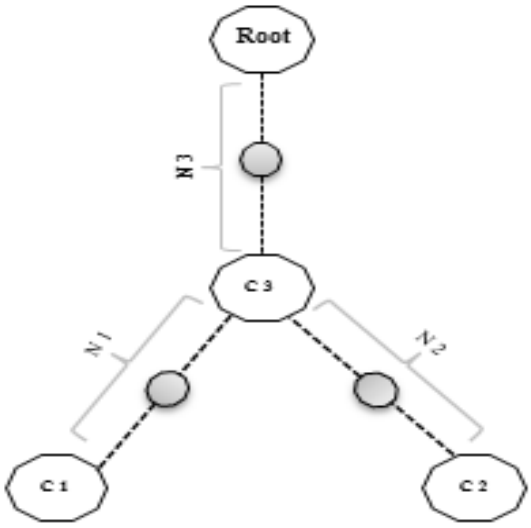

Figure 4.2 The class similarity measure

In (4.1) formula in order to calculate the class similarity between C_1 , C_2 and C_3 is the closest common node (upper-class) of these nodes. N_1 is the number of nodes on the path from C_1 to C_3. N_2 is the number of nodes on the path from C_2 to C_3. Thing class is root node. N_3 is the number of from root node to C_3.

Assume that in our project we try to calculate the taxonomy similarity between "Bileske_Kuvvet" and "Eko_Sistemler" classes using (4.1). N_1 is the path from "Bileşke_Kuvvet" to "Fen_Bilgisi" which is the closest common node of these classes. Moreover N_2 is the path from "Eko_Sistemler" to "Fen_Bilgisi" and N_3 is the path from "Fen_Bilgisi" to "Thing" which is the root node. Based on the values of N_1, N_2, and N_3 similarity from these classes can be computed.

$$Tax.Sim.Cal._{Wu\ \&\ Palmer}\ (Bileşke\_Kuvvet, Eko\_Sistemler) = \frac{2 \cdot 4}{3 + 3 + 2 \cdot 4}$$

$$= {}^{8}/_{14}$$

$$= 0.5714$$

This result shows that "Bileske_Kuvvet "and "Eko_Sistemler" class are not very similar.

4.2.2 Li, Bandar & Mclean's Approach

The second similarity measured we used is Li, Bandar & Mclean's method (Li, Bandar, & McLean, 2003). They develop a different taxonomy similarity. This method is based on distance between class as Wu and Palmer's approach. Taxonomy similarity of this approach's measure is given below;

$$Tax.Sim.Cal._{Li,Bandar\ \&\ Mclean} = \begin{cases} e^{\alpha l} - \dfrac{e^{\beta h} - e^{-\beta h}}{e^{\beta h} - e^{-\beta h}}, & if\ \ C_1 \neq C_2 \\ \\ 1 \quad , & if\ \ C_1 = C_2 \end{cases} \tag{4.2}$$

In (4.2) where, l is the shortest path length between C_1 and C_2, h is the depth of subsume in the hierarchy semantic nets. α is a constant and β is a is a smoothing factor.

In our project suppose that we try to calculate the taxonomy similarity between "Bileske_Kuvvet" and "Eko_Sistemler" classes using (4.2). l which is the shortest path from "Bileske_Kuvvet" to "Eko_Sistemler" is 6. h which is their most specific upper- class is 4. Optimal values of α and β are 0.1 and 0.6 respectively. Based on the values of l, h, α and β similarity from these classes can be computed.

$$Tax.Sim.Cal._{Li,Bandar\ \&\ Mclean}\ (Bileşke_{Kuvvet}, Eko_{Sistemler}) = e^{0.1\cdot 6} - \frac{e^{0.6\cdot 4} - e^{-0.6\cdot 4}}{e^{0.6\cdot 4} + e^{-0.6\cdot 4}}$$

$$= 1.8221 - 0.9836$$

$$= 0.8385$$

The result which obtained Li, Bandar and Mclean's approach is more highly correlated than obtained previous method result.

4.2.3 Our Ontology Methodology and Ontology Creation Steps

Ontology development is an important and an iterative process that should be considered. To create ontology based system in a specific field, first of all the field

should be understood thoroughly (Milli, Ünsal, & Aktaş, 2015). Thus, ontology developing methodologies help in understanding ontology field. There are many ontology developing methodologies in literature. For example Skeletal Methodology was created by Uschold and King (Uschold & Gruninger, 1996). One of them is 101 methodologies (Noy & McGuinness, 2001). In this project, due to the ease of use, understandable, applicable and general acceptance by the most of developer and researcher 101 methodology is selected. This methodology is based on 7 basic steps. These basic steps of methodology are given below;

1. Determine the domain and scope of the ontology
2. Consider re-using existing ontologies
3. Enumerate important terms in the ontology
4. Definite classes and the class hierarchy
5. Determine the data type and the object properties of classes
6. Determine the restrictions of the data type and the object properties
7. Creating individuals (instances)

We create our ontology with the help of these methodology steps. After then in this section the creation of our ontology is explained in detail.

4.2.4 Determine the Domain and Scope of the Our Ontology

In the first step of this methodology used for creating ontology, areas and scope of the developed application should be extensively discussed. Well-defined ontologies should answer following questions.

- What is the domain that the ontology will cover? Our ontologies domain contains the science and technology lesson subjects of third, fourth grade of primary school and all grade of secondary school.

- What we are going to use the ontology? Science and technologies ontology is going to be use for helping students and teachers.

- What types of questions the information in the ontology should provide answers? The Proposed system can respond to student's questions about subject of which they will work, respectively. Moreover it suggest to student about relevant subject or concept.

- Who will use and maintain the ontology? Although the ontology which created for students and teachers, everyone can used after enrolled the web sites. We explain how to extend and maintain the ontology in chapter five.

This step is also known as the decision and plan making process. But maybe some details are not seen before. Some of the decisions can be changed later, when developing process of ontology.

4.2.5 Consider Re-Using Existing Ontologies

One of the most important characteristics of ontologies is reusability. Before starting the creation process of ontology, similar ontologies creating in this area should be investigated from semantic search machine or ontological resources such as Swoogle. OntoSearch, and Watson. The development of existing ontologies may be more useful for academic area instead of developing a new ontology in the same subject.

Another characteristic of ontologies is that it can combine with other ontologies. Sometimes reusing existing ontologies may be a requirement if your system needs to interact with other applications. Many ontologies are already available in electronic form and can be imported into the ontology development environment to change or expand it.

As it is mentioned before, the use of our ontologies is limited in e-learning for Turkish applications. So this step is not applicable in our project because we were not able to find any related existing ontologies. Nevertheless, before the creating OBReSET we searched whether there are similar ontologies to ours or not. However

we did not found relevant ontologies already exist and start developing our ontologies.

4.2.6 Enumerate the Important Terms in the Ontology

The object properties (relation between each other) and their data properties of the terms going to be used in this project should be documented as a comprehensive list without any distinction and worrying about ordering them. We declare all the terms that describe important concepts and their characteristics (Table 4.1). In our project when ontology terms are being listed by using subject of science and technology lesson and terms, we benefited from official website of "Talim Terbiye Kurulu" (TTK, n.d.).

Table 4.1 Some of the terms that will be used in the proposed ontology

Konu Alanı	**Ünite Başlıkları**	**Kazanım Sayısı**	**Öngörülen Süre**	**Ders Saati %**
Fiziksel Olaylar	Çevremizdeki Işık ve Sesler	8	21	19,4
	Işığın Görmedeki Rolü	1	3	
	Işık Kaynakları	1	6	
	Sesin İşitmedeki Rolü	3	6	
	Çevremizdeki Sesler	3	6	
Canlılar ve Hayat	Canlılar Dünyasına Yolculuk	6	21	19,4
	Çevremizdeki Varlıkları Tanıyalım	1	3	
	Ben ve Çevrem	1	4	
	Doğal ve Yapay Çevre	2	4	
	Bilinçli Tüketici	1	6	
	Sağlıklı Yaşam	1	4	
Fiziksel Olaylar	Yaşamımızdaki Elektrikli Araçlar	4	22	19,4
	Elektrikli Araç-Gereçler	1	6	
	Elektrik Kaynakları	2	8	
	Elektriğin Güvenli Kullanımı	1	8	

Table 4.1 Some of the terms that will be used in the proposed ontology (cont.)

Dünya ve Evren	Gezegenimizi Tanıyalım	3	9	8,4
	Dünya'nın Şekli	1	3	
	Dünya'nın Yapısı	2	6	
Canlılar ve Hayat	Vücudumuzun Bilmecesini Çözelim	8	21	19,5
	Destek ve Hareket	2	6	
	Soluk Alıp Verme	2	6	
	Kanın Vücutta Dolaşımı	1	6	
	Egzersiz Yapalım	3	3	
Fiziksel Olaylar	Kuvvetin Etkileri	4	12	11,1
	Kuvvetin Cisimler Üzerindeki Etkileri	1	6	
	Mıknatısların Çekim Kuvveti	3	6	
Madde ve Değişim	Maddeyi Tanıyalım	11	27	25
	Maddeyi Niteleyen Özellikler	1	3	
	Maddenin Hâlleri	2	3	
	Maddenin Ölçülebilir Özellikleri	2	3	
	Maddenin Isı Etkisiyle Değişimi	2	4	
	Madde ve Cisim	1	3	
	Saf Madde ve Karışım	1	3	
	Karışımların Ayrıştırılması	1	5	
	Karışımların Ekonomik Değeri	1	3	

4.2.7 Define the Class and Class Hierarchy

This step is the phase of defining the class. There are many methods used when determining class place order in literature. But the commonly used method is Uschold and Gruninger's method mentioned in their work. These methods are defined as below (Figure4.3);

4.2.7.1 Up-Down Approach

The development process starts from the upper (top level) class and continues towards lower (bottom level) classes. A path is followed from the most public class to the most private class.

4.2.7.2 Down-Up Approach

Starts the development process by identifying lower classes then by grouping upper classes are created. A path is followed the most private classes to the most public class.

4.2.7.3 Hybrid Approach

The development process starts with definition of the most strike classes or concepts. Then most general concepts and most specific classes are created properly. A development process is a combination of the up-down and down-up approaches.

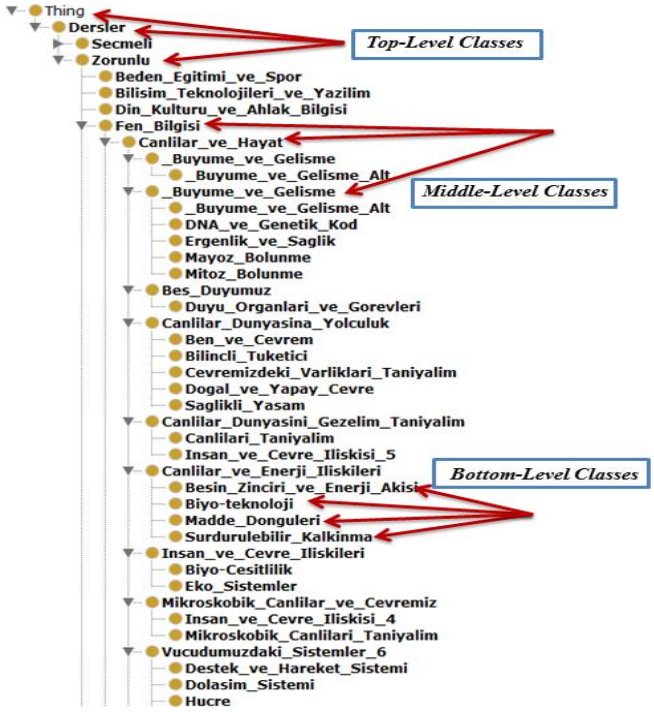

Figure 4. 3 Hierarchical design of science and technology lesson

We define classes in a hierarchical order by using a list created in section 4.2.6. Defining class hierarchy and concept properties is a nested process, so it should be made simultaneously. Some of the terms in ontologies can be added or changed in the creating process. None of these three methods is inherently better than any of the others. If list of classes to be created and their position in the hierarchy is defined completely, using the general to the particular approach provides convenience to the developer. In this study, we chose top-down approach in order to proceed faster. Since classes of ontology to be created are in a systematic list from upside to down.

The class hierarchy represents an “is-a” relation. For example a class B is a subclass of the A class and a class C is a subclass of B class, if so C is also subclass of A class. As it is shown that in Figure 4.4 a class Isi_ve_Sicaklik is a subclass of the Maddenin_Degisimi and Maddenin_Degisimi is a subclass of the Madde_ve_Degisim, if so also Isi_ve_Sicaklik is a subclass of Madde_ve_Degisim. Another way to think of the taxonomic relation is as a “kind-of” relation. For example Madde_ve_Degisim is a kind-of Fen_Bilgisi.

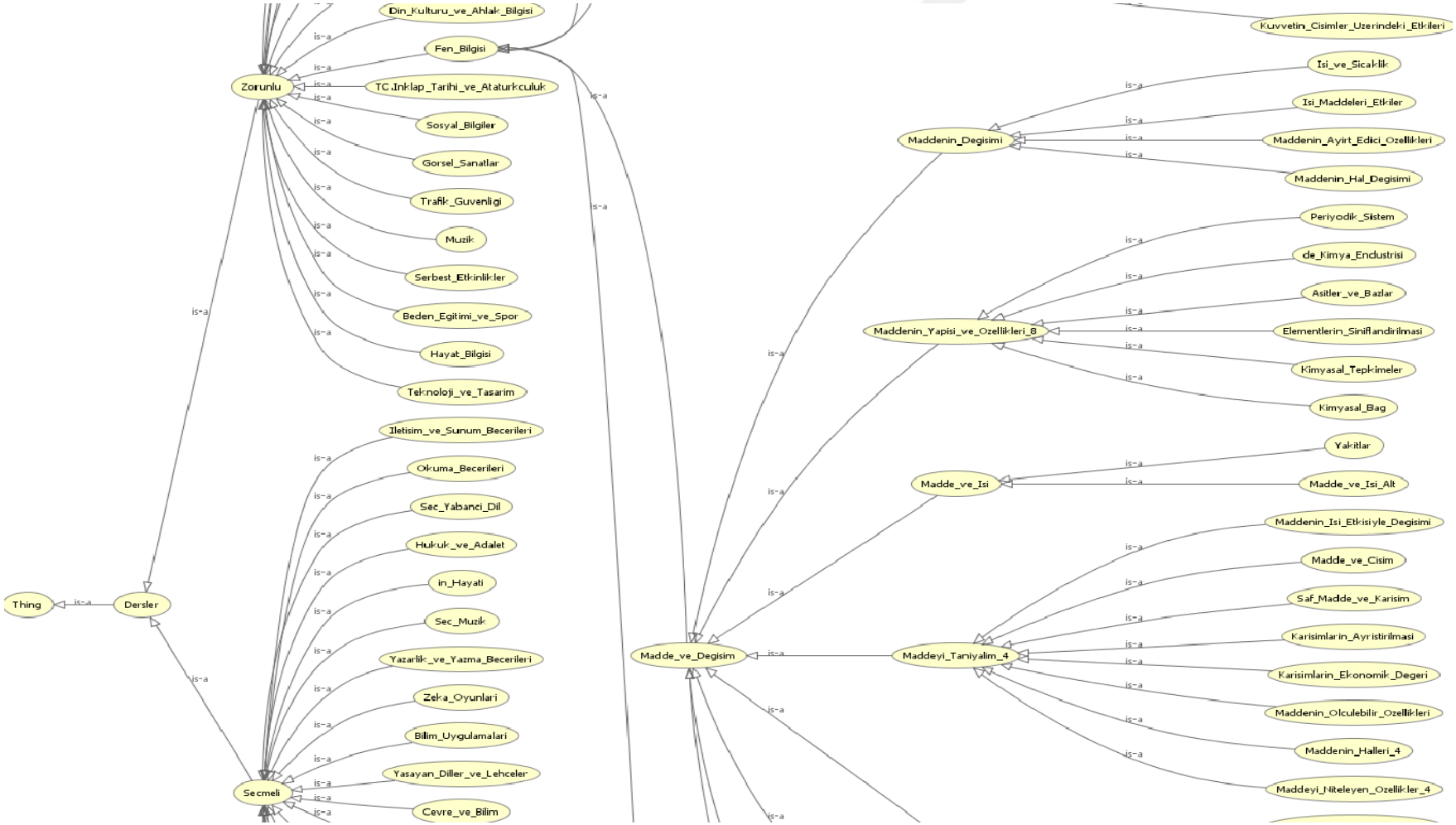

Figure 4.4 Class hierarchy created by using protégé editor and this hierarchy visualization by OWLViz which is protégé's plug-in

4.2.8 Determine the Data Properties and the Object Properties and Classes

Only determine the classes and hierarchical structure between them is not enough to show the information to be given to users clearly. Attributes are utilized to define semantic relation and characteristics of classes in ontologies. There are two kinds of properties; Object properties and data properties. Object properties define already created relationship between two classes, internal or external parts and the characteristics of the class. Data properties are used to describe classes by assigning their attributes.

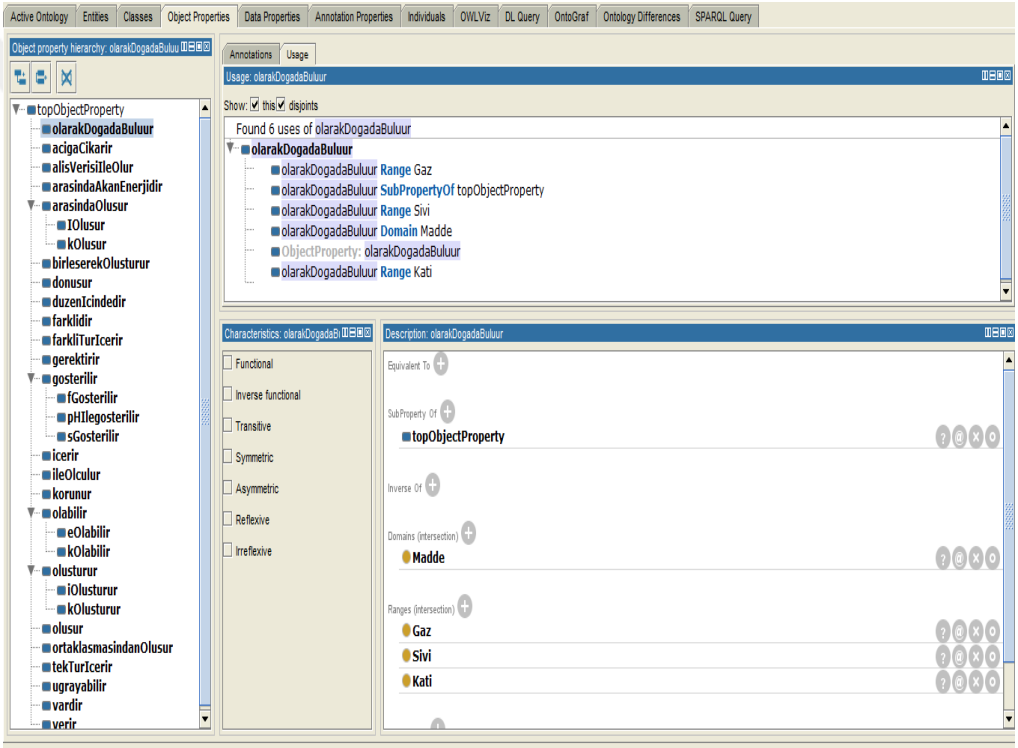

Figure 4.5 Object properties created Protégé OWL editor

For example in this study “olarakDogadaBulunur” property is defined. While the domain of this property is “Madde” class, range of this property is “Katı”, “Sıvı and “Gaz” classes. If we look to this example as type of RDF triple, we can simply understand that: “Madde katı, sıvı, gaz halinde doğada bulunur.” This means the following sentence in English. “Substance can be found as solid liquid and gas in nature” (Figure 4.5).

4.2.9 Determine the Restriction of the Data Type and the Object Properties

Data type properties may have different restrictions describing value type, number of values, allowed values and other. For example in this project some of value types are integer such as “protonSayisi”, “genlesmeKatSayisi”, “erimeDonmaNoktasi”, “PhDegeri”, “ozKutlesi”, and etc. Some of value types are string such as “simgesi” (Figure 4.6).

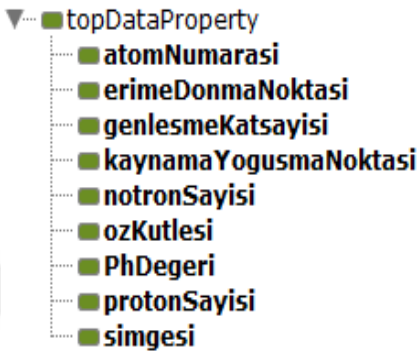

Figure 4.6 Data properties created Protégé OWL editor

4.2.10 Create Individuals

Last step of creating ontologies is to create individuals related to previously defined classes. Firstly we selected class of individuals to be added. Then we created an individual instance of that class. There is not any limitation in individual number.

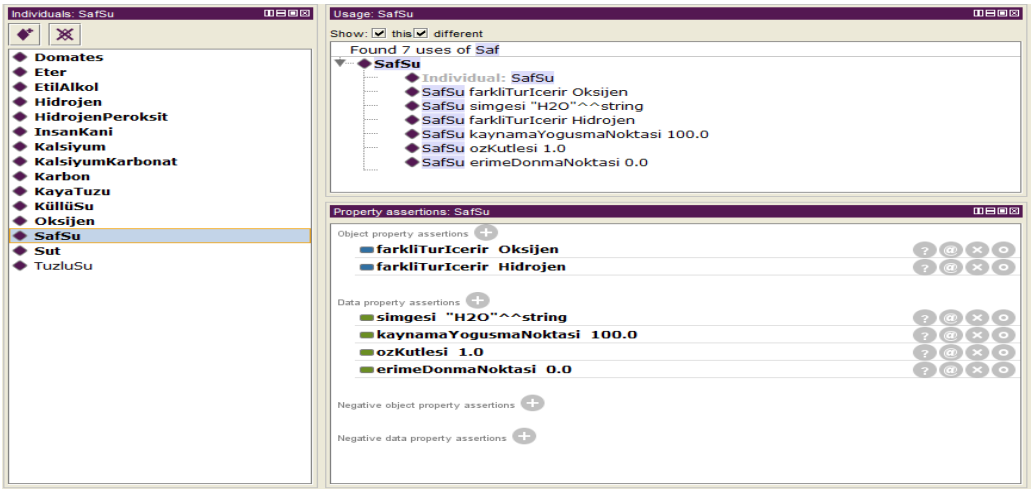

Figure 4.7 Individuals of classes and their values

Individuals related to classes “Atom” and “Madde” are defined in this study (Figure 4.7). Data property assertions and object property assertions of “Bileşik” class are defined.

4.3 Structure of RS in Proposed System

In RS, we had to compute some measures in order to obtain item list which suggested to users. We explain them in this section. These measures consist of three phases. These phases are given below;

1. Similarity Computation
2. Neighborhood Selection
3. Prediction Computation

There are many methods at every phases. We obtain the entire RS mechanism with combined these methods. The reason of there are so many of methods is that different methods work efficiently for different situation. Small changes to be made in any of these steps can lead to big changes in the system's performance and accuracy. Therefore, we should be careful when choosing a method to combine in order to obtain optimum system performance.

Table 4.2 User-Item matrix

	User 1	**User 2**	**User 3**	**User 4**	**User 5**
Mitoz_Bölünme	3		3	3	5
Saf_Maddeler		3		1	
Bileske_Kuvvet	2	?	5		2
Gunes_Sistemi	3	2	1		
Eko_Sistemler		4	3	4	2

After this section in examples we use the values in Table 4.3 when calculating the similarity between “Bileşke_Kuvvet” and “Eko_Sistemler”.

4.3.1 Similarity Calculation Methods

In the first step of CF algorithms similarities between active user and the other users were calculated (Herlocker, Konstan, & Riedl, 2002). In CF algorithms there are several similarity methods have been used such as cosine vector similarity (CS), adjusted cosine vector similarity (ACS), Pearson correlation coefficient (PCC), Manhattan distance (MhD), Euclidean distance (ED), Chebyshev distance (CD), Minkowski distance (MnD) and etc. We used CS (Hamers et al., 1989), ACS (Sarwar, Karypis, Konstan, & Riedl, 2001), and PCC (Hauke & Kossowski, 2011) methods for creating user-user and item-item similarity matrix in our project .The fundamental difference between the similarity computation in user-based CF and item-based CF is that in case of user-based CF the similarity is computed along the columns of the Table 4.2 but in case of the item-based CF the similarity is computed along the rows of the Table 4.2. We explain the item-based algorithm in this section and we create item-item similarity matrix.

4.3.1.1 Pearson Correlation Coefficient

One of the most common used measures of correlation in science is the Pearson Correlation. It is used for indicate the linear relationship between two sets of data. A PCC method measures how highly correlated are two variables and is measured from -1 to +1. If the PCC is 1, the two data are perfectly correlated but otherwise PCC is -1, these data are not correlated. PCC between item a and item b is expressed by the formula as follows;

$$\mathrm{Sim}(a,b) = \frac{\sum_{u\in U}\left(R_{a,u} - \overline{R_a}\right)\left(R_{b,u} - \overline{R_b}\right)}{\left(\sqrt{\sum_{u\in U}\left(R_{a,u} - \overline{R_a}\right)^2}\right)\left(\sqrt{\sum_{u\in U}\left(R_{b,u} - \overline{R_b}\right)^2}\right)} \tag{4.3}$$

Where $R_{a,u}$ shows the ratings of item a given by users, $\overline{R_a}$ average rating of item a and $R_{b,u}$ shows the rating of item b given by users, $\overline{R_b}$ average rating of item b. U is the set of users rate to both item a and item b. Suppose that we try to find similarity between “Bileske_Kuvvet” and “Eko_Sistemler” using PCC.

"Bileske_Kuvvet" item rating vector is = {2, 5, and 2};

"Eko_Sistemler" item rating vector is = {3, 4, and 2};

The calculating similarity between items is measured by observing all the users who have rated both items. In our example we should consider *user 3* and *user 5* ratings. Therefore, the matrix of items should be as follows;

"Bileske_Kuvvet" item rating vector is = {5, and 2};

"Eko_Sistemler" item rating vector is = {3, and 2};

If two items are not rated any common users, a correlation-based similarity measure could not detect any relation between these items. After we determined items matrixes we can compute similarity based on the values from these matrixes.

$$\mathrm{Sim}(a,b) = \frac{(5-3)\,(3-3)+(2-3)(2-3)}{\left(\sqrt{(5-3)^2+\,(2-3)^2}\right)\left(\sqrt{(3-3)^2+\,(2-3)^2}\right)}$$

$$= {}^{1}/_{\sqrt{5}}$$

$$= 0.447$$

The result from PCC similarity shows that "Bileske_Kuvvet" and "Eko_Sistemler" are low correlated each other.

4.3.1.2 Cosine Similarity

This metric is generally used to determine similarity between two vectors that measures the cosine of the angle between them (Salton & McGill, 1983). The CS method measures how highly correlated is two variables and is measured from 0 to 1. Moreover this method is utilized to find the differences between two texts in text mining branch of computer science. This method can be applied to collaborative filtering by simulating users to document, items to words and user rates to word frequencies. CS between *item a* and *item b* is expressed by the formula as follows;

$$\text{Sim}\,(a,b) = \cos\left(|\vec{a}|.\,|\vec{b}|\right) = \frac{\vec{a}.\vec{b}}{|\vec{a}|.\,|\vec{b}|} \tag{4.4}$$

Where $\vec{a}$ is ratings vector for *item a*, $\vec{b}$ is ratings vector for *item b*. $|\vec{a}|$ is a magnitude of vector $\vec{a}$ and $|\vec{b}|$ is a magnitude of vector $\overrightarrow{b}.\ \overrightarrow{a.b}$ indicate their inner product *a* and *b*. Formula (4.4) can also be expressed as follows.

$$\text{Sim}\,(a,b) = \frac{\sum_{u\in U} R_{a,u}\ .\ R_{b,u}}{\sqrt{\sum_{u\in U}(R_{a,u})^2}\ .\sqrt{\sum_{u\in U}(R_{b,u})^2}} \tag{4.5}$$

Based on proximity of the result from (4.5) if it is closer to 1, these items or documents are highly correlated to each other. If it is closer to 0 these items or documents are low correlated. Sometimes this similarity result can be 1. However in this case do not always indicate that correlation between these two items is perfect. This case shows that there is a constant coefficient between two documents (Tan & Steinbach, 2006).

Suppose that like a previous method we try to find similarity between "Bileske_Kuvvet" and "Eko_Sistemler" using CS method.

$$\text{Sim}\,(a,b) = \frac{(5*3)+(2*2)}{\left(\sqrt{5^2+2^2}\right)\left(\sqrt{3^2+2^2}\right)}$$

$$= {}^{19}/_{\sqrt{377}}$$

$$= 0.9785$$

The result from CS similarity shows that "Bileske_Kuvvet" and "Eko_Sistemler" are perfectly correlated each other.

4.3.1.3 Adjustable Cosine Similarity

There are a many disadvantages of CS. An example of its disadvantages is when calculating the similarity between items by using CS, the average of items or users ratings vector are not taken in the account. Therefore the calculation results could not

be satisfying for many developers. Adjustable Cosine Similarity measurement is a modified form of cosine similarity to remove its drawbacks. In item-based CF the fundamental difference between the PCC and ACS is that in case of PCC is computed to considering items ratings average, ACS is computed to considering users ratings average in data collections. ACS between item a and item b is expressed by the formula as follows;

$$\text{Sim}(a,b) = \frac{\sum_{u\in U}\left(R_{a,u} - \overline{R_u}\right)\left(R_{b,u} - \overline{R}_u\right)}{\left(\sqrt{\sum_{u\in U}\left(R_{a,u} - \overline{R}_u\right)^2}\right)\left(\sqrt{\sum_{u\in U}\left(R_{b,u} - \overline{R}_u\right)^2}\right)} \quad (4.6)$$

Where $R_{a,u}$ denotes the ratings of item a given by users, and $R_{b,u}$ denotes the ratings of item b given by users. $\overline{R_u}$ shows average rating of users u. U is the set of users rate to both item a and item b.

Like PCC, the ACS measures how highly correlated are two variables and is measured from -1 to +1. If the value of ACS result is 1, the two data are perfectly correlated but otherwise value of ACS result is -1, these data are not correlated. We would like to calculate similarity between "Bileske_Kuvvet" and "Eko_Sistemler" using ACS.

$$\text{Sim}(a,b) = \frac{\big((5-3)(2-3)\big)\big((3-3)(2-3)\big)}{\left(\sqrt{(5-3)^2+(3-3)^2}\right)\left(\sqrt{(2-3)^2+(2-3)^2}\right)}$$

$$= {}^{-2}/_{2\sqrt{2}}$$

$$= -7071$$

The result from ACS similarity shows that "Bileske_Kuvvet" and "Eko_Sistemler" are very low correlated each other.

For similarity computation we used three methods which are explained above in order to calculate similarity between items. Most of researchers and academician have studied and discussed about which method is more accurate and performance

for years. When we looking these studies and according to our proposed systems, PPC is the more accurate and performing than other methods. These results are discussed in chapter 5 of our thesis.

4.3.2 Neighborhood Selection

In the second step of CF algorithms is neighborhood selection by utilizing similarities of item-item matrix created previous step. This matrix consists of huge mass of data and storage of them is almost impossible due to the memory limitations. Moreover most of values in this matrix are unnecessary for us. They are not used prediction phases. Therefore we should eliminate the low correlated items or users to reducing the size of this matrix. In order to obtain more accurate item list suggested active users, we use only most similar users or items when calculating the prediction. In this section the most important thing is how we make the neighbors selection process. There are several ways to cope with this problem. Some of these ways are explained below;

4.3.2.1 k-Nearest Neighbor Filtering

k-Nearest Neighbor Filtering is a simple algorithm that stores all available cases and classifies new cases based on a similarity measure. The classical k-NN neighborhood selection algorithms still one of the most popular and prominent methods used in the RS community. Probably the most challenging issue in this method is how to choose the value of k. Therefore k should be selected carefully in order to improve efficiency and accuracy of this algorithm. If k is selected too large, system can consume large memory to store neighborhood list and other process of systems work slowly such as prediction process. On the other hand if k is selected too small, the system will be sensitive to noise points (Tintarev & Masthoff, 2011). Therefore we consider to a number of factor such as size of similarity matrix, memory of our system and etc., when choosing the optimal value of k.

4.3.2.2 Threshold Filtering

Another method of neighborhood selection is defining a threshold value. We determine a minimum similarity value instead of fixed neighborhood value. This method is more flexible than k-NN filtering method. As previous method the most challenging issue is how to choose the minimum threshold value. Many times determining the optimum threshold value may be impossible. In this case developer determine it by using trial and error method.

4.3.2.3 %k-Nearest Neighbor Filtering

The main difference between %k-NN Filtering and k-NN filtering is that in k-NN the number of neighbors to be selected is fixed; in %k-NN the number of neighbors is flexible. The value of k changes depending on the number of its similar items.

4.3.2.4 Negative Filtering

Previous methods select neighborhood by calculating positive similarity correlation between items. Its reverse is also possible. In this method neighborhood selection can calculate by using negative similarity correlation. However this method is not commonly used academic community.

In neighborhood phase we used k-NN Filtering, %k-NN Filtering and Threshold methods. It is difficult to determine the most efficient method among these approaches. However in three of these methods it is seen from the result of study that the number of neighbors is very important. The value of k can affect directly to result of proposed system. These results discuss chapter 5 of our thesis.

4.3.3 Prediction Computation Methods

The most important step in a CF system is to generate the output list in terms of prediction. After creating item-item similarity matrix by using methods explained 3.3.1 and choosing their neighbors by using methods explained 3.3.2, then we make

prediction. Prediction computation is the last phase to obtaining recommendation list. In order to predict unknown rating of item a rated by user u in item-based CF, we consider three such techniques are given below;

4.3.3.1 Basic Average

This method is also known most basic approach. Calculated similarity matrix in section 4.3.1 are used to the neighborhood selection. We find an arithmetic average of most nearest neighbors selected in section 4.3.2. Prediction formula of a rating for an item given by user u is the following;

$$\text{Pred. R}_{\text{a,u}} = \frac{1}{\text{N}} \sum_{\text{i}' \in \text{I}'} \text{R}_{\text{i}',\text{u}} \tag{4.7}$$

I' denotes the set of N number of most similar items to item a and $R_{i',u}$ is the rating given by user u on the item i'. $Pred.R_{a,u}$ show that prediction rate given by user u on item a. Although these methods show a good performance it is not preferred due to its low accuracy.

4.3.3.2 Weighted Average

In this method the similarities between the item and neighbors utilize as weight vector. Formally, using the notion shown in (4.8) we can denote the prediction $Pred.R_{a,u}$ as;

$$\text{Pred. R}_{\text{a,u}} = \frac{\sum_{\text{i}' \in \text{I}'} (\text{R}_{\text{i}',\text{u}}) \left(\text{Sim}(\text{a}, \text{i}')\right)}{\sum_{\text{i}' \in \text{I}'} \text{Sim}(\text{a}, \text{i}')} \tag{4.8}$$

Sim (a,i') show similarity value between item a and item i^i. The big difference between WA and Basic Average is that WA considered similarity between items when computing $Pred.R_{a,u}$. WA method is most commonly used approaches by researches. Their results are more accuracy than basic average.

4.3.3.3 Adjusted Weighted Average

The method called AWA is taking into account the different rating scales of different items. Each item is perceived differently by users in item-based CF. we can denote the prediction $Pred.R_{a,u}$ as;

$$\text{Pred.}\,\mathrm{R_{a,u}} = \overline{\mathrm{R}}_{\mathrm{i}} + \frac{\sum_{\mathrm{i}'\in \mathrm{I}'}(\mathrm{R_{i',u}})\big(\mathrm{Sim(a,i')}\big)}{\sum_{\mathrm{i}'\in \mathrm{I}'}\mathrm{Sim(a,i')}} \tag{4.9}$$

Sim (a,i') show similarity value between item a and item i^i. The big difference between WA and Basic Average is that WA considered similarity between items when computing $Pred.R_{a,u}$. WA method is most commonly used approaches by researches. Their results are more accuracy than basic average.

AWA is the best accuracy result among these methods but its performance reduces concretely, particularly when working huge mass of item-item matrix.

We try these three methods to compute unknown ratings explained above in our project. Except these methods there are varieties of prediction approaches in recommendation community while the most accurate results are obtained AWA in terms of a lot of data set. Because AWA takes into account how items or users perceive the rating scales. But in this study for our data set we observed that SA method gave the best results. These results are discussed in chapter 5 of our thesis.

4.4 Parallelized Hybridization Design

This phase we implemented parallelized hybridization design approach to our project (Jannach & Zanker, 2010). This phase is combining the first and the second phases (Figure 4.8). System's general output is returned to active user which is the intersection of item set from first phase and item set from second phase.

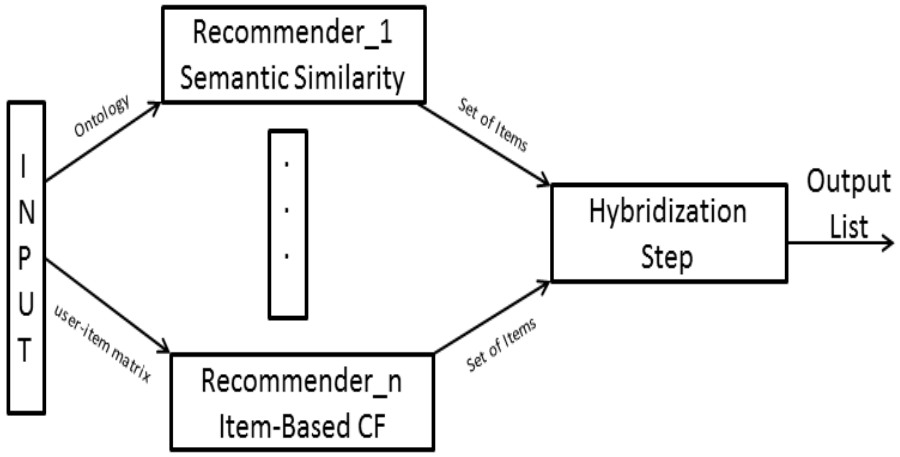

Figure 4.8 Simple architecture of proposed system

4.5 Structure of User Interface

In our project we design two different web sites. First web site is for student, administrator and other users. We create our project as web based in order to easy accessible. If users login this site with their username and password they can rate web content and they can create their item user matrix or change it. In this situation, when user searches something, user-based CF algorithm is worked background of web site otherwise item-based CF algorithm is worked. Users can observe subject which is studied recently. Moreover in order to do some operations such as manage user operation, change web content and etc. we design administrator panel. Second web site is for computation of accuracy and performance of the develop systems.

4.6 Characteristic of Data Set Created Manually

Data set used for recommendation systems and ontological researchers should be created by natural way in order to measure performance and accuracy of algorithms properly. However data sets used for scientific and academic studies are formed after costly and challenging process. Sometimes they take for years to create dataset. Researchers may have to define their data sets manually about their study area, due to limitations of time and budget. Despite intensive searching we did not find any data set relevant to our scope of our project. Most of data sets on web are out of the scope of our study area.

Before we begun to create our data set we had analyzed other data set which is composed by natural way such as Movielens and Netflix movie dataset mentioned in chapter two. To simulate Movilens data set we determine several criterion such as data sparsity, rating scales number of items and users when we creating our data set. Characteristic of data set which we used in our project are given following;

- Consist of 500 users.
- Consist of 182 items (title of lessons or subject).
- Sparsity rate of data set is almost %8.
- Rates are given by users on items randomly.

CHAPTER FIVE
TEST, RESULTS AND EVALUATION METRICS

In this chapter, firstly evaluation metrics which we used in our experiments are describe briefly. Then, proposed system is tested and results of this test details are presented. After that we compared the results obtained from different methods. Finally, the results of the test are given in tables and charts. In this project, the rate of the test data and education data are optional. But generally 20% of rating data is used as test set and rest of our data is used as training set. Our data set contains 500 users and 182 items (lesson of primary and secondary school) and 7408 rates are given by the users on item. We do not consider users whose number of rates under 20% due to quality of similarity between user-user and item-item.

5.1 Evaluation Metric

In our experiment**s**, we use two different type of methods in order to evaluate results. Some of these methods are statistical accuracy criteria such as mean absolute error (MAE), root mean square error (RMSE). Second type of these methods decision support accuracy criteria such as receiving operating characteristic (ROC). Before we compare the results, these methods are explain briefly;

5.1.1 Statistical Accuracy Criteria

There are a different statistical accuracy method. We used three of them in this project. They are given below;

5.1.1.1 Mean Absolute Error (MAE)

In statistics, one of the most commonly used method for calculating accuracy is Mean Absolute Error. It has been utilized as a standard statistical metric to measure how to close predictions value are to the actual value. Some of these results which obtained from predictions would be negative value. Therefore results from MAE are

taken absolute value in order to compute the error average of any systems properly. The MAE formula is given by;

$$\mathrm{MAE} = \frac{1}{\mathrm{N}} \cdot \sum_{\mathrm{u}=1}^{\mathrm{N}} \left| \mathrm{R_{i,u}} - \widehat{\mathrm{R}}_{\mathrm{i,u}} \right| \tag{5.1}$$

N shows number of prediction $R_{i,u}$ is the actual rate given by user u on item i. $\hat{R}_{i,u}$ indicates predicted value of proposed system.

5.1.1.2 Mean Square Error (MSE)

In statistics, the mean squared error (MSE) is a measure of how close a predicted value line is to actual data points. Every data point are taken the distance vertically from the point to the corresponding value on the curve of the error, and square the value. Then add up all those values for all data points, and divide by the number of points minus two. The taking squares is done so negative values do not cancel positive values. The smaller the MSE, the closer the fit is to the data. The MSE formula is given by;

$$\mathrm{MSE} = \frac{1}{\mathrm{N}} \cdot \sum_{\mathrm{u}=1}^{\mathrm{N}} (\mathrm{R_{i,u}} - \widehat{\mathrm{R}}_{\mathrm{i,u}})^2 \tag{5.2}$$

As in the MAE formula, N shows number of prediction $R_{i,u}$ is the actual rate given by user u on item i. $\hat{R}_{i,u}$ indicates predicted value of the proposed system.

5.1.1.3 Root Mean Square Error (RMSE)

The Root Mean Square Error (RMSE) is a generally utilized to measure of the difference between predicted values of proposed systems and the actually values observed from the obtained data set. RMSE is also known as the root mean square deviation (RMSD). Firstly, predicted values are squared in order to transform positive numbers and then take the square root of these values. RMSE is expressed by the formula as follows;

$$RMSE = \sqrt{\frac{1}{N} \cdot \sum_{u=1}^{N} \left(R_{i,u} - \hat{R}_{i,u}\right)^2} \quad (5.3)$$

As in the MAE method, N shows number of prediction $R_{i,u}$ is the actual rate given by user u on item i. $\hat{R}_{i,u}$ indicates predicted value of proposed system.

5.1.2 Decision Support Accuracy Criteria

We used Receiver Operating Characteristic method as a decision support technique in order to calculate our system accuracy. It is explain briefly below;

5.1.2.1 Receiver Operating Characteristic

The Receiver Operating Characteristic (Van Rijsbergen, 1979) analysis is one of the most used decision support technique. The ROC was first developed by electrical engineers to account for perceptual detection of stimuli. It is also known that ROC Curve. The ROC analysis has been used in medicine, radiology, psychology and other areas for many decades. Especially ROC is used to diagnose to some diseases such as breast cancer. More recently, it has been introduced to machine learning. ROC Curve plotted between 0 and 1. Result from ROC is 1 in the case shown that values are clustered properly. If it is 0.5 in the case shown that values are clustered randomly.

TP=True Positive= It is shown that recommended products or items which is interested by user.

FP=False Positive= It is shown that did not recommend products or items which is interested by user.

FN=False Negative= It is shown that recommended products or items which is not interested by users.

TN= True Negative=It is shown that did not recommend products or items which is not interested by users.

- **Sensitivity Calculation:** It is proportion of true positives. The ability of the system on correctly predicting the condition in cases it is really present.

$$\mathrm{SEN} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}} \tag{5.4}$$

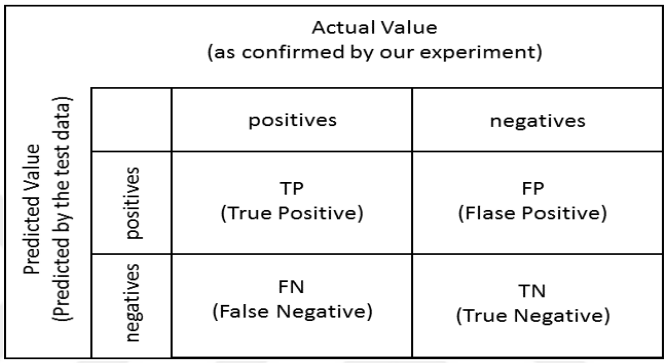

Figure 5.1 Contingency table (confusion matrix) for a binary classifier (Fawcett, 2003)

- **Specificity Calculation:** The ability of the system in correctly predicting the absence of the condition in cases it is not present.

$$\mathrm{SPE} = \frac{\mathrm{TN}}{\mathrm{TN} + \mathrm{FP}} \tag{5.5}$$

- **Efficiency Calculation.** The arithmetic mean of Sensibility and Specificity. In practical situations, sensibility and specificity vary in reverse directions. Generally, when a method is too responsive to positives, it tends to produce many false positives, and vice versa. Therefore, a perfect decision method (with 100% specificity and 100% specificity) rarely is conceived, and a balance between both must be obtained.

$$\mathrm{EFF} = \frac{\mathrm{SEN} + \mathrm{SPE}}{2} \tag{5.6}$$

- **Accuracy Calculation:** The proportion of correct predictions, without considering what is positive and what is negative. This measure is highly dependent on the data set distribution and can easily lead to wrong conclusions about the system performance.

$$\mathrm{ACC} = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{P} + \mathrm{N}} \tag{5.7}$$

All of these methods which were used in this study perform on user-item data set using different value and approach which were mentioned in chapter 4.3. After accuracy rate were computed for different evaluation methods which were explained chapter 5.1. Results from every step of this study were compare with each other. Then, it is aimed to observe which technique has the best performance on which evaluation approach. Some parameters were kept constant in order to obtain reliable result while experimental studies were conducted studies.

5.2 Experimental Results

Many of experiments are conducted and obtained result from these experiments during the project. However important result and graphics are given in this chapter;

5.2.1 Comparison of Neighborhood Selection Methods

In this experimental work, results from k-NN, %k-NN and Threshold approaches are compared with each other in order to observe the effect of selecting different k or threshold value. The statistical details of the results from different k value are shown in Table 5.1.

The following parameters will be used in the next three experiments.

- Algorithm : CF (I-I)
- Similarity Measures : PCC
- **Neighborhood Selection : k-NN - %k-NN - Threshold**
- Prediction Method : SA
- **Evaluation Approach : MAE – MSE – RMSE - ROC_4**

Table 5.1 The effect of the different k value on result in k-NN approach

k	**10**	**20**	**30**	**40**	**50**	**60**	**70**	**80**	**90**
MAE	0.830	**0.813**	0.856	0.869	0.888	0.907	0.924	0.917	0.930
MSE	**0.789**	0.799	0.895	0.943	1.003	1.047	1.083	1.062	1.094
RMSE	**0.886**	0.898	0.947	0.972	1.001	1.024	1.039	1.033	1.046
ROC_4	0.792	**0.797**	0.778	0.765	0.751	0.742	0.738	0.730	0.724

In the first experiment the effect of the k value on result in k-NN approach is presented. k-NN approach was implemented on item-item similarity matrix which is obtained from PCC. After then select different k values and calculate predictions. This predictions were evaluated using the MAE, MSE, RMSE, ROC_4 methods and compared with their results.

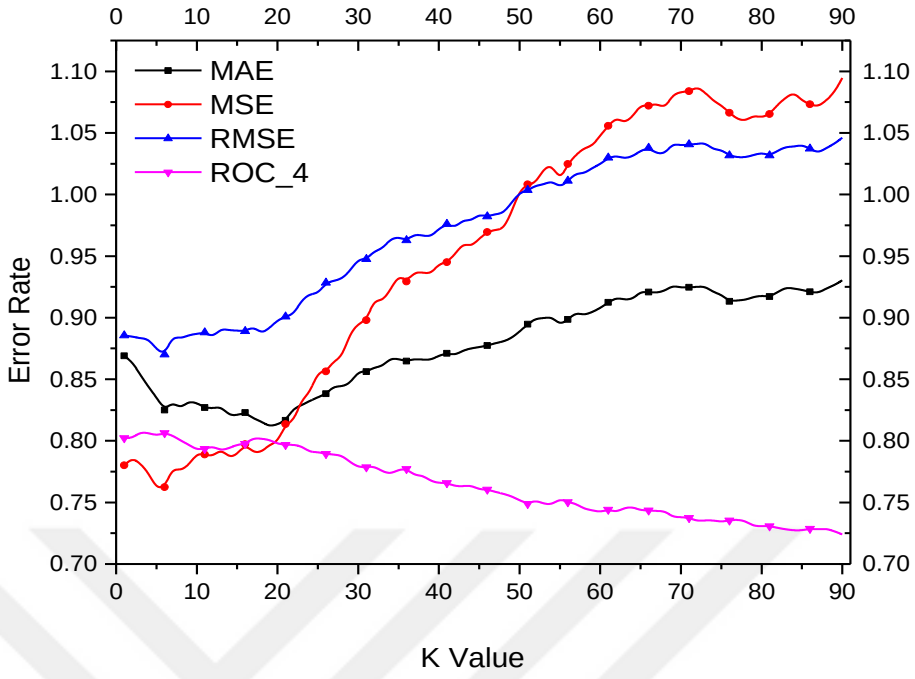

Figure 5.2 Error rates for k values in k-NN approach

Based on Table 5.1, the best result is 0.789 obtained from the MSE. However in term of average MAE approach provides best result. According to Figure 5.2 the best results were obtained from neighborhood value between 10 and 20. In previous works on MovieLens data set many researchers had found the optimum k values between 20 and 50. The main reason of obtaining varied results in this studies is different characteristic of data sets such as sparsity of data, user count, item count, etc. Therefore, this experiment show that optimum value of k may be change by different data set.

In the second experiment the effect of threshold value on result in threshold approach is presented. This approach was implemented on item-item similarity matrix which is obtained from PCC. In this experiment we worked with all threshold values which can be given (from -1 to 1) in order to illustrate effect of different values, and predictions were calculated. As first experiment this predictions were evaluated using the MAE, MSE, RMSE, ROC_4 methods and compared with their results. Based on Table 5.2, the best result is 0.780 obtained from the MSE. However in term of average MAE approach provides the best results.

Table 5.2 The effect of the different k value on result in k-NN approach

Value	MAE	MSE	RMSE	ROC_4
-1	0.956	1.165	1.081	0.713
-0.9	0.954	1.167	1.078	0.713
-0.8	0.948	1.144	1.071	0.715
-0.7	0.952	1.147	1.071	0.715
-0.6	0.951	1.139	1.067	0.713
-0.5	0.948	1.136	1.067	0.715
-0.4	0.942	1.128	1.060	0.715
-0.3	0.934	1.107	1.052	0.719
-0.2	0.929	1.108	1.050	0.724
-0.1	0.927	1.087	1.040	0.728
0	0.922	1.075	1.037	0.732
0.1	0.912	1.049	1.024	0.746
0.2	0.894	1.001	1.003	0.755
0.3	0.876	0.968	0.982	0.759
0.4	0.860	0.932	0.966	0.766
0.5	0.848	0.892	0.949	0.774
0.6	0.842	0.865	0.928	0.779
0.7	0.825	0.812	0.902	0.792
0.8	**0.810**	**0.780**	**0.880**	**0.798**
0.9	0.829	**0.780**	0.881	0.796
1	0.880	0.790	0.889	0.797
AVG	0.901	1.014	1.005	0.746

As it is mentioned before threshold value should be selected carefully in order to improve efficiency of methods. If this value is selected too large, rates of dissimilar neighbors may be taken an account. On the other hand if this value is selected too small the systems can be more sensitive to noise points.

As shown in Figure 5.3 for this data set the best threshold value is 0.8 or 0.9. The main reason of selected too large value is that our data set is too small.

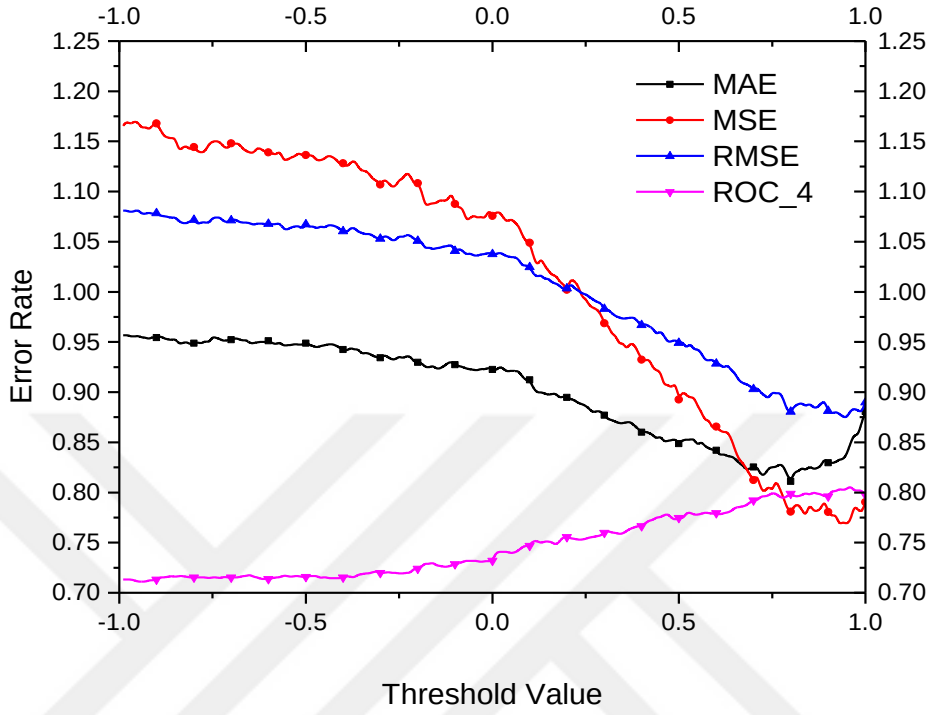

Figure 5.3 Error rates for threshold values in threshold approach

In the third experiment effect of k value on result in %k-NN approach is presented. This approach was implemented on item-item similarity matrix which is obtained from PCC.

Table 5.3 The effect of the different k value on result in %k-NN approach

K Val.	**MAE**	**MSE**	**RMSE**	**ROC_4**
10	**0.813**	**0.794**	**0.889**	**0.801**
20	0.868	0.930	0.967	0.777
30	0.899	1.019	1.012	0.748
40	0.926	1.083	1.041	0.734
50	0.931	1.101	1.048	0.721
60	0.935	1.109	1.054	0.721
Average	0.888	0.981	0.988	0.755

In this experiment we selected different k values and calculate predictions. This predictions were evaluated using the MAE, MSE, RMSE, ROC_4 methods and compared with their results. Based on Table 5.3, the best result is 0.794 were obtained from the MSE method. However in term of average MAE approach provides best results as in first experiment. The effect of the different k values on result in %k-NN methods are shown Figure 5.4.

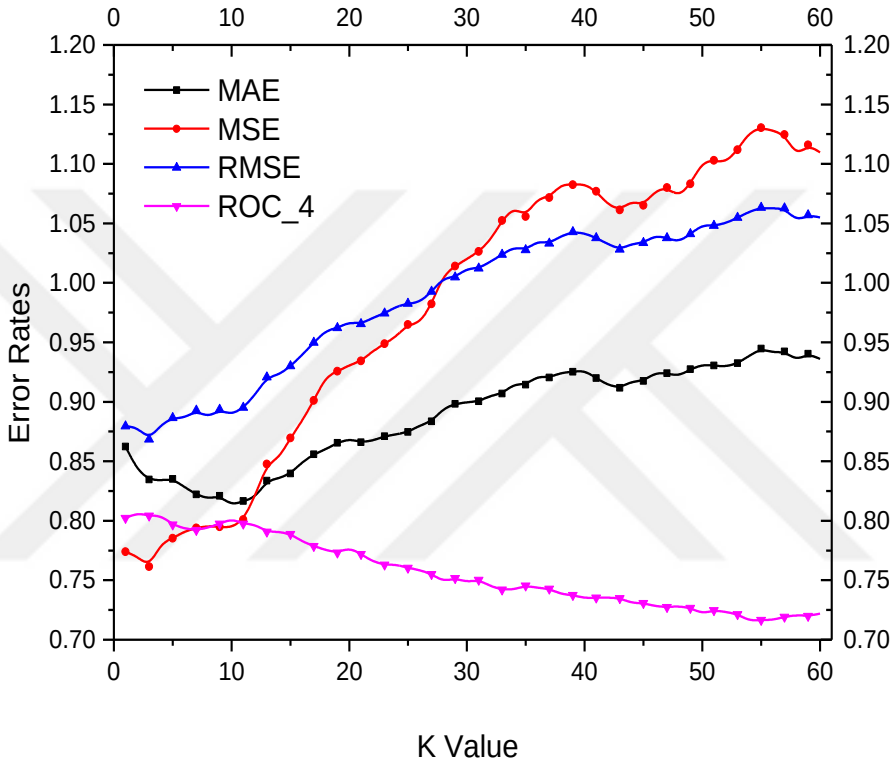

Figure 5.4 Error rates for k values in %k-NN approach

According to three experiments which statistical details are given above show that k-NN method has better performance than %k-NN and Threshold methods when compare with their results. But in studies which are conducted by other researches previously, in some cases it is observed that %k-NN approach has better performance than other approaches.

When analyzed these experiments results, it is clearly shown that best k values in k-NN approach are from about 5 to 15, and in %k-NN approach are from 3 to 7, and in Threshold approach are from 0.8 to 0.9.

5.2.2 The Effect of Similarity Calculations Methods on CF Algorithms

In this experiment, results from PCC, CS, and ACS approaches are compared with each other in order to observe their effect on item-item CF and user-user CF algorithms. The statistical details of the results from different similarity approaches are shown in table 5.4.

The following parameters will be used in the next experiments.

- **Algorithm : CF (I-I)-CF(U-U)**
- **Similarity Measures : PCC - CS - ACS**
- Neighborhood Selection : k-NN (k_Value=10)
- Prediction Method : SA
- **Evaluation Approach : MAE - MSE - RMSE - ROC_4**

Table 5.4 The effect of the different similarity methods on I-I CF and U-U CF algorithms

Alg.	Evaluation Metrics	Pearson Correlation	Cosine Similarity	Adjusted Cosine Similarity
Item-Item CF	**MAE**	0.830	0.829	**0.816**
	MSE	0.789	0.791	**0.772**
	RMSE	0.886	0.890	**0.881**
	ROC_4	0.792	**0.800**	**0.800**
User-User CF	**MAE**	**0.827**	0.828	0.835
	MSE	**0.783**	0.792	0.795
	RMSE	**0.889**	0.891	**0.889**
	ROC_4	0.798	0.794	**0.803**

Table 5.4 shows the results when compare with PCC, CS, and ACS approaches. When the results were analyzed seen that ACS similarity method provides better resluts than PCC, and CS methods.

5.2.3 The Effect of Prediction Methods on CF Algorithms

In this experiment, results from SA, WA, and AWA approaches are compared with each other in order to observe their effect on item-item CF and user-user CF algorithms. The statistical details of the results from different similarity approaches are shown in Table 5.5.

The following parameters will be used in the next experiments.

- **Algorithm : CF (I-I)-CF(U-U)**
- Similarity Measures : PCC
- Neighborhood Selection : k-NN (k-Value=10)
- **Prediction Method : SA – WA - AWA**
- **Evaluation Approach : MAE - MSE - RMSE - ROC_4**

The conducted experiment demonstrates that the usage of SA technique can outperform the usage of WA and AWA methods in terms of accuracy.

Table 5.5 The effect of the different prediction methods on I-I CF and U-U CF algorithms

Alg.	Evaluation Metrics	Simple Average	Weighted Average	Adjusted Weighted Average
Item-Item CF	**MAE**	**0.830**	0.846	0.922
	MSE	**0.789**	0.837	1.083
	RMSE	**0.886**	0.915	1.042
	ROC_4	**0.792**	0.783	0.729
User-User CF	**MAE**	**0.827**	0.842	0.939
	MSE	**0.783**	0.837	1.098
	RMSE	**0.889**	0.913	1.049
	ROC_4	**0.798**	0.787	0.727

5.2.4 Comparison of the Similarity Methods

In this experiment various k values were applied with different similarity methods in order to observe their effect on result from MAE. The statistical details of the results from different K values are shown in Table 5.6.

The following parameters will be used in the next three experiments.

- Algorithm : CF (U-U)
- **Similarity Measures : PCC - CS - ACS**
- **Neighborhood Selection : k-NN - %k-NN - Threshold**
- Prediction Method : SA
- Evaluation Approach : MAE

Table 5.6 The effect of different similarity methods with different k values on results

k Val.	Pearson Correlation	Cosine Similarity	Adjusted Cosine Similarity
10	**0.827**	0.828	0.835
20	**0.834**	0.844	0.844
30	**0.855**	0.861	0.859
40	**0.870**	0.877	0.884
50	0.898	**0.894**	0.901
60	**0.905**	0.910	0.911
70	0.922	0.924	**0.918**
80	**0.927**	0.936	0.930
90	**0.925**	0.941	0.932

In the first experiment different k values in k-NN approach and PCC similarity method were used together and observed their effect on results. Although the results very close to each other, PCC method provides better result than other methods in term of MAE approach. Figure 5.5 shows in three methods the best results are obtained by using k value between 10 and 20.

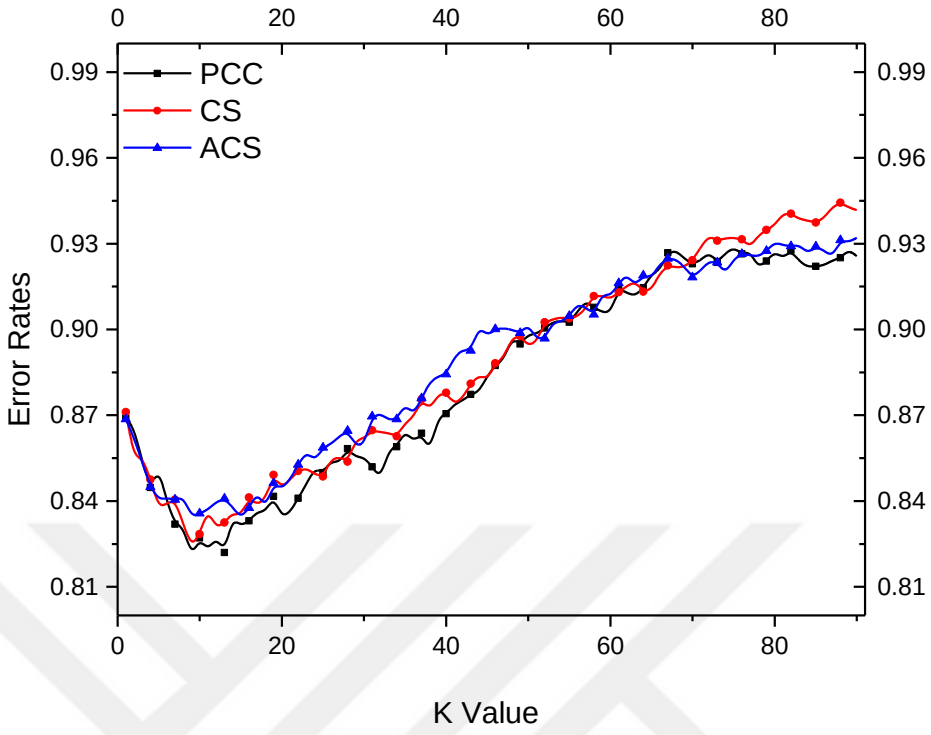

Figure 5. 5 Error rates for different k values with using different similarity methods

Second experiment is that effect of threshold value on result in threshold approach. This approach was implemented on user-user similarity matrix which is obtained from PCC, CS, and ACS.

In this experiment we worked with all threshold values which can be given (from -1 to 1) in order to illustrate effect of different values, and predictions were calculated. However in CS methods measurement scale between 0 and 1 as seen that Figure 5.6. As first experiment this predictions were evaluated using the only MAE methods and compared with their results. In Figure5.6 the results show that best results are obtained by using PCC method.

Table 5.7 The effect of different similarity methods with different threshold values on results

Value	PCC	CS	ACS	Value	PCC	CS	ACS
-1	**0.956**	-	0.971	**0.1**	**0.912**	0.955	0.916
-0.9	**0.954**	-	0.973	**0.2**	**0.894**	0.957	0.914
-0.8	**0.948**	-	0.967	**0.3**	**0.876**	0.957	0.891
-0.7	**0.952**	-	0.966	**0.4**	**0.860**	0.955	0.882
-0.6	**0.951**	-	0.964	**0.5**	**0.848**	0.955	0.876
-0.5	**0.948**	-	0.964	**0.6**	**0.842**	0.954	0.871
-0.4	**0.942**	-	0.963	**0.7**	**0.825**	0.952	0.856
-0.3	**0.934**	-	0.963	**0.8**	**0.810**	0.949	0.841
-0.2	**0.929**	-	0.963	**0.9**	**0.829**	0.912	0.840
-0.1	**0.927**	-	0.961	**1**	0.880	**0.879**	0.882
0	**0.922**	0.957	0.964	**AVG**	**0.901**	0.946	0.922

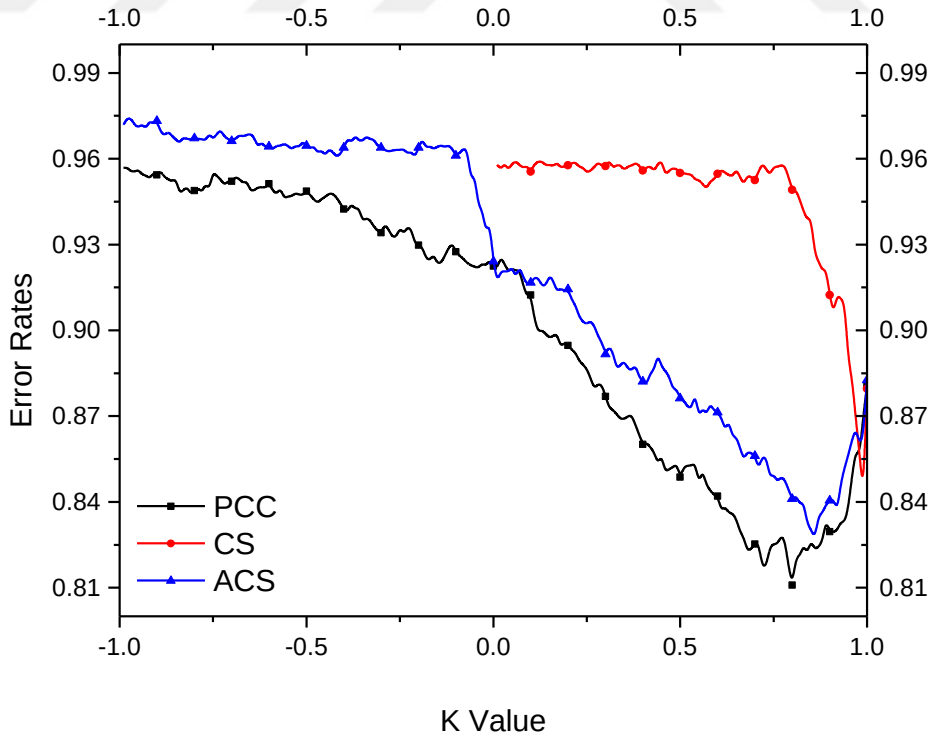

Figure 5.6 Error rates for different threshold values with using different similarity methods

In the third experiment the effect of k value on result in %k-NN approach is presented. This approach was implemented on item-item similarity matrix which is obtained from PCC, CS, and ACS.

Table 5.8 The effect of different similarity methods with different k values on results

K Value	Pearson Correlation	Cosine Similarity	Adjusted Cosine Similarity
10	**0.888**	0.890	0.900
20	**0.929**	0.940	0.932
30	0.954	0.967	**0.945**
40	0.962	0.969	**0.952**
50	**0.964**	0.974	0.965
60	**0.966**	0.972	0.971

In this experiment we selected different k values and calculate predictions. This predictions were evaluated using the only MAE methods and compared with their results.

When compare the results, PCC method provides the best result in term of MAE approach. Figure 5.7 shows in the results of three methods and the best results are obtained by using k value between 2 and 5 for all methods.

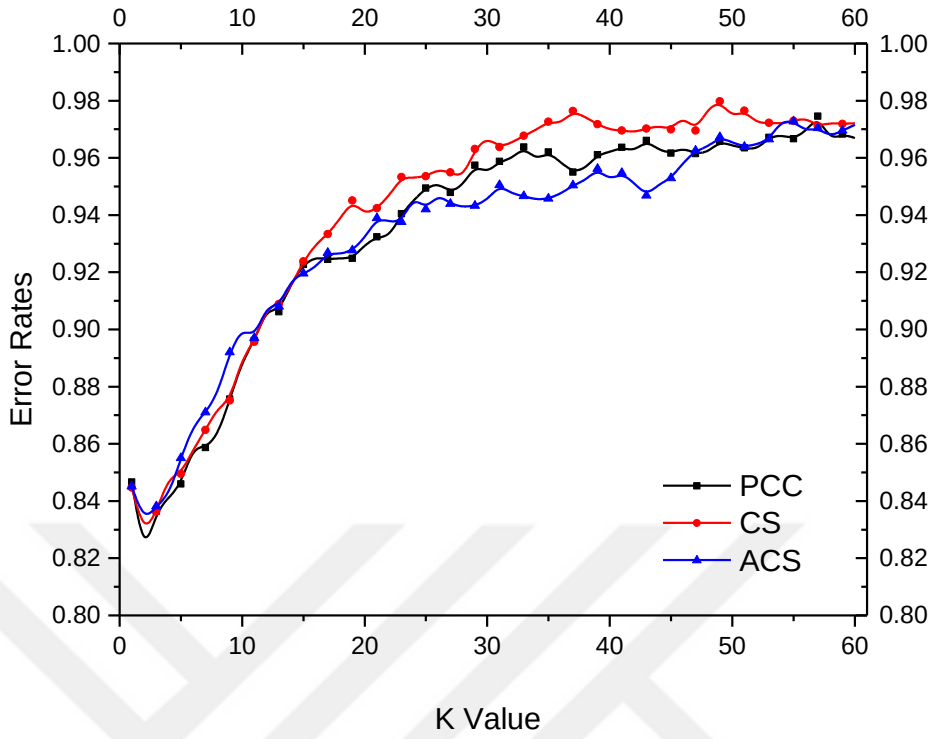

Figure 5.7 Error rates for different K values in %k-NN approach with using different similarity methods

The conducted three experiment demonstrates that the usage of PCC technique can outperform the usage of CS and ACS methods in terms of MAE accuracy method.

CHAPTER SIX
CONCLUSION AND FUTURE WORK

RS approaches generally used for e-commerce, news portals and entertainment (music and movie) applications until today. However, the implementation of RS methods in e-learning applications are limited. It is obvious that there is a gap in RS based e-learning applications in the literature. During this study, RS methods implemented for e-learning data set for science and technology lessons in order to guide primary and secondary school students in their education. Data set used in this study is prepared manually due to absence of data set in computer environment. The properties of previous data sets implemented by different researchers are used while preparing our data set such as data sparsity.

Although RS approaches intensively used in academic and industrial studies, still they have unsolved problems. The common problems for RS approaches are cold start (new item, user and system), data sparsity, scalability and limited contend analysis. In addition, previous researches are addressing these problems and offer some solutions. For instance, most of these solutions are focused on using variety of hybrid approaches such as CBF and CB.

In this study, we propose a hybrid system in order to solve cold start and data sparsity problem encountered most of the RS applications. The developed system integrate ontology, which is the back bone of semantic web, and CF which is an effective method used in RS. In order to integrate ontology and CF a parallelized hybridization design method mentioned in chapter 4 is implemented. The developed hybrid system is tested in our data set.

RS implementation process consist of three computational parts. Similarity calculation is initial part. In the first part we try PCC, CS, and ACS methods. The results obtained from these three methods show PCC has the highest accuracy as previous studies. In order to find nearest neighbor of active user k-NN, %k-NN, and threshold methods are used for neighborhood selection in the second part of our study. The conducted experiments demonstrate that the usage of k-NN method can

outperform the usage of %k-NN and threshold value. In the third part, SA, WA, and AWA methods are used to calculate prediction in order to generate recommend list. Experiments results reveals that SA provides the best accuracy for our data set. However, AWA prediction method gives the best results in previous studies, SA gives better results in our study. The reason of the difference in our results is the low similarity ratios due to the arbitrary rates of our data set. The accuracy of results are evaluated by using MAE, MSE, RMSE, and ROC metrics.

Ontology implementation process consist of two parts.101 methodology is used during the creation of the ontology in the first part. Subject of second and primary school science and technology lesson's ontologies are created in Protégé OWL Editor. Class hierarchy, relation with classes and data properties are also determined in this part. In the second part, the distance between classes are calculated by using Wu & Palmer's, and Li, Bandar & Mclean's approaches. Li, Bandar & Mclean's approach gives the best results for our data sets.

Generally pure RS cannot solve the cold start and data sparsity problems. RS cannot generate sufficient recommend list for users in some situations. As a result of this, proposed hybrid system generate an ordinary recommend list. Moreover, the quality of recommend list can be increased by using semantic web and ontology. This study demonstrates that proposed hybrid system solves the cold start and data sparsity problem encountered most of the RS applications.

In future subject of Mathematics, Turkish, History, Geography and other lessons can be added in our data set. The developed hybrid system will be tested with an expanded version of data set.

REFERENCES

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

Baeza-Yates, R., & Ribeiro-Neto, B. (1999). *Modern information retrieval*. (1st Ed.). New York Press.

Balabanović, M., & Shoham, Y. (1997). Fab: Content-based, collaborative recommendation. *Association for Computing Machinery*, *40* (3), 66–72.

Bansal, A., Kona, S., Blake, M. B., & Gupta, G. (2008). An agent-based approach for composition of semantic web services. In *2008 IEEE 17th Workshop on Enabling Technologies: Infrastructure for Collaborative Enterprises* (pp. 12–17). IEEE.

Barranco, M. J., & Martínez, L. (2010). A method for weighting multi-valued features in content-based filtering. *Lecture Notes in Computer Science (including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *6098* (3), 409–418.

Belkin, N. J., & Croft, W. B. (1992). Information filtering and information retrieval two side of the same coin. *Association for Computing Machinery*, *35* (12), 29–38.

Bellogín, A., & de Vries, A. P. (2013). Understanding similarity metrics in neighbour-based recommender systems. *Proceedings of the 2013 Conference on the Theory of Information Retrieval - ICTIR '13*, 48–55.

Benjamins, V., & Gómez-pérez, A. (1999). Knowledge-system technology: ontologies and problem-Solving methods. *European Conference in Artificial Intelligence*, *20*, 1–15. Retrieved November 11, 2015, from http://hcs.science.uva.nl/usr/richard/pdf/kais.pdf

Benoît Marchal. (n.d.). *XML by example*. *Production*. Retrieved Agust 9, 2015, from http://www.dcc.fc.up.pt/~zp/aulas/1112/pde/geral/bibliografia/XML By Example.pdf

Berners-lee, T., Hendler, J., & Lassila, O. (2001). The semantic web. *Scientific American*, *284*(5), 35–43.

Breese, J. S., Heckerman, D., & Kadie, C. (1998). Empirical analysis of predictive algorithms for collaborative filtering. *Proceedings of the 14th Conference on Uncertainty in Artificial Intelligence*, *461* (8), 43–52.

Burke, R. (2000). Knowledge-based recommender systems. In *Encyclopedia of Library and Information Systems 69*, 175–186.

Burke, R. (2002). Hybrid recommender systems: Survey and experiments. In *User Modelingand User-Adapted Interaction, 12*, 331-370.

Chen, H., Finin, T., & Joshi, A. (2003). An ontology for context-aware pervasive computing environments. *The Knowledge Engineering Review, 18* (3), 197–207.

Chen, Z. S., Jang, J. S. R., & Lee, C. H. (2011). A kernel framework for content-based artist recommendation system in music. *IEEE Transactions on Multimedia, 13* (6), 1371–1380.

Chong, E. I., & Eadon, G. (2005). *An efficient SQL-based RDF querying scheme.* Retrieved July 9 2015, from http://www3.ntu.edu.sg/home/bshe/SQLBasedRDF_vldb05.pdf

Debnath, S., Ganguly, N., & Mitra, P. (2008). Feature weighting in content based recommendation system using social network analysis. *Proceeding of the 17th International Conference on World Wide Web*, 1041–1042.

Decker, S., Harmelen, F. Van, Broekstra, J., Erdmann, M., Fensel, D., Horrocks, I., et. al. (2000). The Semantic Web - on the respective roles of XML and RDF. *IEEE Internet Computing*, *4* (October), 19.

Deshpande, M., & Karypis, G. (2004). Item-based Top-N recommendation algorithms. *Association for Computing Machinery Transactions on Information Systems*, *22* (1), 143–177.

Erling, O., & Mikhailov, I. (2009). RDF support in the virtuoso DBMS. *Studies in Computational Intelligence*, *221*, 7–24.

Fawcett, T. (2003). Receiver operating characteristic graphs. *Notes and partical considerations for redearchers, Machine Learning, 31.*

Felfernig, A., Friedrich, G., Jannach, D., & Zanker, M. (2006). An integrated environment for the development of knowledge-based recommender applications. *International Journal of Electronic Commerce*, *11* (2), 11–34.

Frauenfelder, M. (2004). Sir Tim Beners-Lee. *Technology Review*, *107*(8), 40. Retrieved July 25, 2015 from http://search.ebscohost.com/login.aspx?direct=true&db=f5h&AN=14750013&lang=es&site=ehost-live

Gong, S. (2010). A collaborative filtering recommendation algorithm based on user clustering and item clustering. *Journal of Software*, *5* (7), 745–752.

Gruber, T. R. (1995). *Toward principles for the design of ontologies used for knowledge sharing*? Retrieved August 25, 2016 from http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.89.5775&rep=rep1&type=pdf

Guarino, N. (1998). Formal ontology and information systems. *Proceedings of the First International Conference*, *46* (June), 3–15.

Hamers, L., Hemeryck, Y., Herweyers, G., Janssen, M., Keters, H., Rousseau, et. al. (1989). Similarity measures in scientometric research: The Jaccard index versus Salton's cosine formula. *Information Processing & Management*, *25* (3), 315–318.

Hauke, J., & Kossowski, T. (2011). Comparison of values of Pearson's and Spearman's correlation coefficients on the same sets of data. *Quaestiones Geographicae*, *30* (2).

Herlocker, J., Konstan, J., & Riedl, J. (2002). An empirical analysis of design choices in neighborhood-based collaborative filtering algorithms. *Information Retrieval*, 287–310.

Herlocker, J. L., Konstan, J. A., Borchers, A., & Riedl, J. (1999). An algorithmic framework for performing collaborative filtering. In *Proceedings of the 22nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval - SIGIR '99* (pp. 230–237). New York, USA: ACM Press.

Jannach, D. (2004). Preference-based treatment of empty result sets in product finders and knowledge-based recommenders. *Poster Proceedings of the 27th Annual German Conference on Artificial Intelligence, KI 2004*, 145–159.

Jannach, D., & Zanker, M. (2010). Introduction of recommendation systems. *ACM Symposium on Applied Computing 2010* Sierre, Switzerland, 22 March 2010

Jentzsch, A., Usbeck, R., & Vrandecic, D. (2014). An incomplete and simplifying introduction to linked data. *Perspectives on Ontology Learning*, 21–33.

Kanne, C.-C., & Moerkotte, G. (1999). Efficent storage of XML data. *Lehrstuhl Für Praktische Informatik 3 Universitat Mannheim Germany*, (December 13).

Leiner, B. M., Cerf, V. G., Clark, D. D., Kahn, R. E., Kleinrock, L., Lynch, D. C., et al. (2009). A brief history of the Internet. ACM SIGCOMM *Computer Communication Review*, 39 (5), 22–31.

Lenat, D. (1995). CYC: A Large-Scale Investment in knowledge infrastructure. *Communications of the ACM, 38* (11), 33–38.

Li, Y., Bandar, Z. A., & McLean, D. (2003). An approach for measuring semantic similarity between words using multiple information sources. *IEEE Transactions on Knowledge and Data Engineering*, *15* (4), 871–882.

Maedche, A., & Staab, S. (2000). Semi-Automatic engineering of ontologies from text. In *Proceedings of the 12th Internal Conference on Software and Knowledge Engineering* (pp. 231–239).

Melville, P., Mooney, R. J., & Nagarajan, R. (2001). Content-boosted collaborative filtering. *Proceedings of the 2001 SIGIR Workshop on Recommender Systems*, 9.

Miller, E. (1998). An Introduction to the resource description framework, *Journal of Library Administration*, *34* (3-4), 245-255.

Miller, G. A. (1995). WordNet: A lexical database for English. *Communications of the ACM*, *38* (11), 39–41.

Milli, M., & Milli, M. (2015). Ontology based recommender system with using dissimilar users. In *International Science and Technology Conference* (pp. 405–410). St.Petersburg.

Milli, M., Ünsal, E., & Aktaş, Ö. (2015). Creating ontology based concept maps which can be queried in computer environment. In *International Science and Technology Conference* (pp. 137–144). St. Petersburg.

Miranda, T., Claypool, M., Gokhale, A., & Sartin, M. (1999). Combining content-based and collaborative filters in an online newspaper. In *In Proceedings of ACM SIGIR Workshop on Recommender Systems*.

Mohebbi, K., Ibrahim, S., & Idris, N. B. (2012). Contemporary semantic web service frameworks: An overview and comparisons. *International Journal on Web Service Computing*, *3* (3), 65–76.

Mowbray, T. J., & Zahavi, R. (1995). *The essential CORBA : Systems integration using distributed objects*, xvi, 316 p. Retrieved September 21, 2015, from http://www.loc.gov/catdir/toc/onix04/95050760.html

Nathanson, T., Bitton, E., & Goldberg, K. (2007). Eigentaste 5.0 constant-time adaptability in a recommender system using item clustering. *Proceedings of the 2007 ACM Conference on Recommender Systems - RecSys '07*, 149–152.

Noy, N., & McGuinness, D. (2001). Ontology development 101: *A guide to creating your first ontology development*, *32*, 1–25.

Pazzani, M. J. (1999). A framework for collaborative, content-based and demographic filtering. *Artificial Intelligence Review*, *13*(5), 393–408.

Pazzani, M. J., & Billsus, D. (2007). Content-based recommendation systems, *The Adaptive Web*, LNCS 4321, pp. 325 – 341, 2007.

Pukkhem, N. (2013). Ontology-based semantic approach for learning object recommendation. *ACEEE International Journal on Information Retrival*, *3* (4). Retrieved from http://hal.archives-ouvertes.fr/hal-00942526/

Punnoose, R., Crainiceanu, A., & Rapp, D. (2012). Rya: A scalable RDF triple store for the clouds. *Proceedings of the 1st International Workshop on Cloud Intelligence*, 4.

Rashid, A. M., Lam, S. K., Karypis, G., & Riedl, J. (2006). ClustKNN: A highly scalable hybrid model- & memory-based CF algorithm. *Search*.

Resnick, P., Varian, H. R., & Editors, G. (1997). Recommender systems mende tems. *Communications of the ACM*, *40* (3), 56–58.

Ruotsalo, T. (2010). *Methods and applications for ontology-based recommender systems*. *science and technology*. Retrieved December 12, 2015, from http://lib.tkk.fi/Diss/2010/isbn9789526031514/

Salter, J., & Antonopoulos, N. (2006). Recommender agent : Collaborative and content-based filtering. *Analysis*, *21* (February), 35–41. Retrieved September 9, 2015, from http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1588800

Salton, G., & McGill, M. J. (1983). Introduction to modern information retrieval. *Introduction to Modern Information Retrieval*. Retrieved July 19, 2015 from http://search.ebscohost.com/login.aspx?direct=true&db=lxh&AN=ISTA2001897&site=ehost-live

Sarwar, B., Karypis, G., Konstan, J., & Riedl, J. (2001). Item-based collaborative filtering recommendation algorithms. *Proceedings of the 10th*, *1*, 285–295.

Sauermann, L., Cyganiak, R., & Völkel, M. (2008). Cool URIs for the semantic web *working draft W3C*, *49* (December 2008), 1–15. Retrieved from http://www.w3.org/TR/cooluris/

Schafer, J. Ben, Konstan, J., & Riedi, J. (1999). Recommender systems in e-commerce. *Proceedings of the 1st ACM Conference on Electronic Commerce EC 99*, *2001*, 158–166.

Schein, A., & Popescul, A. (2002). Methods and metrics for cold-start recommendations. *Proceedings of the 25th Annual International ACM Conference on Research and Development in Information*

Schenkel, R. (2003). *XML for beginners snake oil ?,* 1–55. Retrieved from http://resources.mpi-inf.mpg.de/d5/teaching/ss03/xml-seminar/talks/xml

Segaran, T., Evans, C., & Taylor, J. (2009). Programming the semantic web. *Semantic Web Services Processes and Applications 1* 54-62

Shardanand, U., & Maes, P. (1995). Social information filtering: Algorithms for automating "Word of Mouth." *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems - CHI '95*, 210–217.

Sharma, K. S., & Ugrasen, S. (2012). Design and implementation of architectural framework of recommender system for e-commerce, *International Journal of Computer Science and Information Technology & Security, 1*, 153–162.

Sheth, B., & Maes, P. (1993). Evolving agents for personalized information filtering. *Proceedings of 9th IEEE Conference on Artificial Intelligence for Applications*, 345–352.

Soboroff, I. (1999). Combining content and collaboration in text filtering. *International Joint Conferences on Artificial Intelligence, 99*, 86–91.

Swartout, B., Patil, R., Knight, K., & Russ, T. (1996). Toward distributed use of large-scale ontologies. *Proc. of the Tenth Workshop on Knowledge Acquisition for Knowledge-Based Systems*, 138–148. Retrieved August 29 2015 from http://scholar.google.com/scholar?hl=en&btnG=Search&q=intitle:Toward+Distributed+Use+of+Large-Scale+Ontologies#0

SW Layers. (n.d.). Retrieved November 28, 2015, from ttps://www.w3.org/2001/sw/

Tan, P., & Steinbach, M. (2006). Introduction to data mining instructor ' s solution Manual. *Names, 28*(1), 9–35, v.

Theoharis, Y., Christophides, V., & Karvounarakis, G. (2005). Benchmarking database representations of RDF. *Institute of Computer Science, Forth Vassilika Vouton, Heraklion, Greece* 685–701.

Tintarev, N., & Masthoff, J. (2011). Designing and evaluating explanations for recommender systems. *Recommender Systems Handbook*. (1) 479-511.

TTK. (n.d.). Retrieved November 20, 2015, from http://ttkb.meb.gov.tr/

Uschold, M., & Gruninger, M. (1996). Ontologies: Principles, methods and applications. *The Knowledge Engineering Review*, *11* (02), 93.

Van Rijsbergen, C. J. (1979). Receiver accuracy criteria evaluation metrics. *Information Retrieval*, 112–140.

Vozalis, E., & Margaritis, K. (2003). Analysis of recommender systems' algorithms. *Hercma*, 1–14. Retrieved Agust 18, 2015, from http://lsa-svd-application-for-analysis.googlecode.com/svnhistory/r72/trunk/LSA/Other/LsaToRead/hercma2003.pdf

Wang, R.-Q., & Kong, F.-S. (2007). Semantic-enhanced personalized recommender system. *Sixth International Conference on Machine Learning and Cybernetics*, (August), 19–22.

Wang, S., Xie, Y., & Fang, M. (2011). A collaborative filtering recommendation algorithm based on item and cloud model. *Wuhan University Journal of Natural Sciences*, *16* (1), 16–20.

Wang, Y., Gong, J., & Wu, X. (2007). Geospatial semantic interoperability based on ontology. *Geo-Spatial Information Science*, *10* (3), 204–207.

WordPress. (n.d.). Retrieved November 20, 2015, from https://oye2906.wordpress.com/tag/web-1-0/

Winer, D. (2011). Review of ontology based storytelling devices. *Culture, Computation: Essays in Honour* Retrieved Agust 8, 2015, from http://www.judaica-europeana.eu/docs/Winer_Ontology_Storytelling_svt.pdf

Wu, Z., & Palmer, M. (1994). Verb semantics and lexical selection. *32nd Annual Meeting on Association for Computational Linguistics*, 6.