在高等教育基于本体的混合方法，以课程推荐

由

Mohammed Essmat易卜拉欣

的论文中的需求部分履行提交博士学位奖

朴茨茅斯大学的哲学

能源与电子工程学院

技术学院

朴次茅斯大学，

2019年月

11

摘要 可供学生使用的与课程相关的信息，其数量和复杂性都在迅速

网络上增加。这种潜在的信息过载挑战了标准信息

随着用户发现越来越难找到相关信息，检索模型。

教育领域是一个已经被这个问题影响的主要领域之一。

选择大学的高等教育课程可能非常繁琐且极其

对于学生来说，复杂。个性化的推荐系统可能是推荐的有效方法

向潜在学生相关课程。现有的主要方法

基于关键词的在推荐过程中不能满足单个用户的需求。

尽管模型使用协作过滤，但通常缺少历史信息。

另一个缺点是他们没有提供的课程的全面知识

与学生最相关。

这项研究提出了一种新颖的基于本体的混合方法，从而推荐个性化

通过整合有关所有可用信息课程以满足学生的个性化需求

课程的并支持学生根据自己的职业目标选择课程，。本论文

做出了三点主要贡献：首先，，提出了一个基于本体的全面的

通过结合几种个性化课程推荐框架，称为OPCR

协作过滤，基于内容的过滤，人工智能技术

本体表示和知识管理等。开发了一套基于本体的

推荐算法，用于个性化推荐。该框架

能够自动提取，集成数据，以向学生提供适合

他们需求的建议。它不仅减少了信息过载，而且
提高了推荐准确性。其次，提出了本体模型，用于提取和
从多种来源整合信息，，有助于提高的质量
通过克服课程信息的异质性推荐。此外，它还具有
诸如通用性之类的属性，使它可以在不同推荐系统
随用户兴趣和对象属性而变化的域中使用。最后，的
开发并基于OPCR框架个性化推荐系统
评估了。该系统可在线访问，以供研究人员和开发人员使用。结果
表明，与，使用分层相关基于本体的推荐算法可
仅考虑关键字的过滤方法相比概念的产生更好的结果
相似性。另外，当本体相似性时，系统的性能得以改善
利用项目的简档和用户的简档之间的。

I

声明

尽管已注册上述学位的候选人，但我尚未注册任何其他
研究奖。本文所体现的结果和结论是指定的工作，
候选人尚未获得其他任何学术奖项。

首先，我要感谢最伟大和最的宇宙创造者

致谢

首先，我要感谢最伟大和最的宇宙创造者

仁慈，没有他的意志，我将不可能实现我希望愿望
获得博士学位的。

我要感谢许多启发了我个人和
职业的人。首先，我要感谢我的第一任主管Linda Yang博士
，他向我介绍了人工智能和信息检索系统领域。我
永远不会有勇于向前的方法和理论研究
没有她在我巨大的信心。我几乎想不出什么成就
没有她的帮助，，而且她为我提供了自主探索新想法的自由。在
过去的四年中，我得到了她的指导，鼓励，宽容，自由，
信任，热情好客和她的友谊。

我还要感谢技术学院的工程学院
朴茨茅斯大学为我提供了高质量的研究环境。我也感谢
我的同学们的鼓励和友谊。我要
感谢斯坦福大学所有朋友和Protégé工具的成员，
尤其是Michael和Alexander在整个提供的帮助，支持和宝贵见解
本体模型开发过程中。我感谢我的同事和其他研究人员
对本论文的间接或直接贡献。我热烈感谢我的同事们，
尤其是工学院的博士生（海德，的帮助和支持
穆塔达，马吉，西克里，阿里，迈克尔，马哈茂德，普里西拉，谢内多，艾哈迈德等）。我
非常感激赞助使我的博士工作成为可能的资金来源
由伊拉克伊拉克教育发展高级委员会（HCED）的。

我要对父母，兄弟姐妹的长期深表谢意
支持和鼓励，尤其是他们多年来的对我的信任

。我的父母，我永远无法感谢您为我所做的一切。我也非常
感激我挚爱的妻子Zahraa Albayati，他一直陪伴着我
，为帮助我继续学业做出了巨大的牺牲。感谢
您的耐心和关心，因为您，我的生活充满了爱和幸福。最后，我
想向我美丽的儿子Warth和Siraj表达我的爱意，感谢的耐心和
他们在博士期间理解。

三，

献身

于世界上所有的孤儿和穷人，特别是在我伊拉克国家
所属的，我希望这项成就能够使他们的脸上露出笑容，并擦干
眼泪。

出版物清单

1.期刊论文

❖Ibrahim，ME，Yang，Y.，Ndzi，DL，Yang，G.和Al-Maliki，M.，2019年。基于本体的个性化课程推荐框架。*IEEE Access*，第*7页*，第5180-5199页。DOI：10.1109 / ACCESS.2018.2889635❖

易卜拉欣，我，杨，Y.（2019）。《自动本体构建》高等教育中信息集成的，*《信息科学学报》*（**正在审查中**）

2.会议论文

❖Ibrahim，ME，Yang，Y.和Ndzi，D.，2017年7月。使用本体进行个性化课程推荐应用程序。在 *国际会议*
计算科学及其应用 （第426-438页）中。湛史普林格。

❖易卜拉欣（ME）和杨（Y。），2019年1月。基于本体的网络爬网方法，用于在教育领域检索资料。在 *第11届国际国际*
Agent与人工智能会议上。科学出版社。

缩写

IR信息检索

RS推荐系统

CF协同过滤

CBF基于内容的过滤

IE信息提取

KNN k近邻

OWL本体的Web语言

OPCR基于本体的个性化课程推荐

OM本体映射

UCAS高校招生服务

DB数据库

HTML 超文本标记语言

TF期限频率

IDF逆文档频率

VSM向量空间模型

高等教育

NSS全国学生调查

HECoS高等教育学科分类

DOM文档对象模型

目录

VIII

图列表

表列表

第十二

章引言

“教育是最重要的。你可以用它来改变世界的最强大的武器”

纳尔逊·曼德拉

本章讨论这展示了这一研发是如何及时动机
对于它试图解决的问题，以及目的和目标的Research
打算实现。随后，突出了本论文在知识方面的贡献
和技术。最后，说明了本文的内容
。

1.1动机

从理论上讲，数字内容的不断增长应增加的机会

发现符合个人需求的内容。但是，传统信息的用户

系统可能会遇到信息过载的情况，因为只有少数几个项目的领域内

在用户感兴趣（Bollen，Knijnenburg，Willemsen和Graus，2010
年）。

研究表明，，学生会因大量可用信息而超负荷工作

选择课程时（Huang，Chen和Chen，2013年）。当交流了的信息时，就会发生信息过载

超出处理能力大量。利用

教育技术的先进特性，可以以多种访问更高效，

格式和不同类型的更复杂的信息环境

信息资源。信息的传播给过多的信息

学生带来了（Kalyuga，2011年）。如今，可供使用的与课程相关的信息范围

学生正在迅速增加（Bhumichitr，Channarukul，Saejiem，Jiamthapthaksin和

Nongpong，2017年）。从大量网站中查找与课程相关的信息是一个

艰巨而耗时的过程。有效的搜索将包括所有相关

有关课程内容，教育机构和有关职业信息的信息

特定课程主题的。帮助学生从众多做出正确选择，

可用课程中以满足他们的个性化需求是一个真正的挑战（Huang，Zhan，

Zhang和Yang，2017
）。

如此丰富的信息意味着学生需要搜索，组织和使用

可以使他们符合个人目标，兴趣和当前知识水平的资源。

这可能是一个耗时的过程，因为它涉及到访问每个平台，搜索

可用的课程，仔细阅读每门课程的教学大纲，然后选择看起来也是最

1个

适合学生（Apaza，塞万提斯，Quispe医师，与月神，2014）。大量的

信息导致需要帮助学生选择，组织和使用相资源

与他们的目标，兴趣和知识匹配的（Farzan＆Brusilovsky，2006）。

Bendakir和Aïmeur的报告指出，追求教育的学生面临两个挑战：

众多的课程可供选择，并且缺乏对要遵循的课程

和顺序的了解（Bendakir＆Aımeur，2006
年）。

选择课程的过程可能非常繁琐且极其复杂。

如今，学生可以快速找到与大学及其提供的课程有关的信息

使用在线资源（Huang等，2013）。但是，仅仅因为更多课程

现在可以从大学网站上获得信息并不能自动意味着

学生具备对所有课程进行评估的认知能力（Ibrahim，Yang和Ndzi，2017年）。

相反，他们面临着一个被称为“信息超载”的问题（Z.

Zhang，Zhou，＆Zhang，
2010）。

在研究开始时开发的人工智能方法现已应用于

信息检索系统。推荐系统（RS）提供了一种有前途的

信息过滤方法（Garcia，Sebastia和Onaindia，2011），因为它们可以帮助用户找到最

合适的项目（Jannach，Zanker，Felfering和＆Friedrich，2011）。有许多在线

当前系统可用于查找和搜索课程。但是，

这些方法都没有针对性地向用户提供个性化建议，这些建议

提供有关特定相关课程的全面信息。

这项研究的动机之一是减少用户面临的信息过载

希望选择大学课程时。教育信息发布在互联网

以不同的格式上，因此提取符合用户搜索有用信息

查询的提出了巨大挑战（Alimam＆Seghiouer，2013）。的异质性

课程信息和个人用户需求使得决策过程非常繁琐和

复杂。测量项目概念的本体层次结构是一种

有前途的方法，可以帮助解决异质性问题（Bach＆Dieng-Kuntz，

2005
）。

此外，尽管有些课程标题相似，但每种标题可能会导致不同的职业道路

（DS＆K，2015年）。 Sandvig and Burke argued in their research work that a lack of knowledge

regarding which appropriate item to choose from a large number of items means that people

need to seek an advisor or guidance(Sandvig & Burke, 2005). Providing comprehensive

knowledge about a satisfactory item that a user may wish to select is another challenge because

the difference in a user's tastes and preferences will influence the degree of user satisfaction.

For example, a person seeking to choose a university course degree will need to acquire relevant

information regarding the course, not simply the course subject content, but also the reputation

of the university, the facilities provided, career opportunities and so forth. Therefore, the need

to establish a comprehensive framework that can extract and integrate information from

multiple sources and align this data in a unified form is another motivation for this research.

Recommender systems offer a promising approach to information filtering (Garcia et al., 2011) as they help users to find the most appropriate items (Jannach et al., 2011). Based on the needs of each user, the recommendation system will generate a series of specific suggestions (Ren, Zhang, Cui, Deng, & Shi, 2015). Recommender systems have been used to provide recommendations in a variety of domains such as e-commerce, news, movies, music, research papers, course materials among others. The education domain has used recommender systems for different purposes such as e-learning applications, academic advice, course material suggestions and so forth. Many online systems are currently available that can be used to find and search for courses (S. Wang & Sapporo, 2006) which use tools based on the users' prior knowledge of the courses (H. Zhang, Yang, Huang, & Zhan, 2017), keyword-based queries (Khan, 2000; ucas.com, 2018), collaborative filtering (CF) (Carballo, 2014) (T. Huang et al., 2017), data mining and association rules (Noakes, Arrott, & Haakana, 1968; H. Zhang et al., 2017) and content-based filtering (CBF) models (Lotfy & Salama, 2014). Despite the strong influence of existing course recommendation systems and how useful they can be, there are certain significant limitations such as:

- Models based mainly on keywords fail to address an individual user's needs in the recommendation process.

• Although models use collaborative filtering and data mining such as association rules and

decision trees, there is often a lack of historical information that makes this approach

challenging to adopt. For instance, new students who wish to use the system do not have

sufficient information about the model and are therefore unable to generate any

recommendations.

• The shortcoming of models that use content-based filtering is that current approaches are

based only on a specific subject recommendation rather than an entire university course.

Moreover, the similarity calculation in these models is based on the weighted average of

features and does not take into account user interaction with the system, such as the rating

value of recommended items.

• Another shortcoming of current models is that they do not provide comprehensive

knowledge regarding the course that is most relevant to the student. For example, students

need to know what future career the course will lead to and require information regarding

this aspect, as well as the quality of the facilities of the educational institution itself that

will be providing the course.

Categorising the needs of students and their areas of interest enables an appropriate course to be recommended. It is possible to help students to select a course by developing methods that

will both integrate the data from multiple heterogeneous data sources and allow this to rapidly

establish valuable course-related information (Huang et al., 2013).

All these facts provided the motivation to develop a new approach to overcome the information

overloading phenomenon and to obtain comprehensive knowledge regarding the recommended

items. Two research problems need to be addressed. First, how to integrate all available

information about courses, including the course modules, job opportunities and the users'

interests and build a relationship between the relevant information. Second, with all the

integrated information, how to recommend the most relevant courses to meet user's individual

needs.

1.2 Aims and objectives

This thesis aims to tackle the problem of information overloading. It develops a practical

framework based on the methods proposed in the research that can have a realistic application

with an impact within the scope of an education recommender system. The framework supports

data integration and course recommendation applications. Involving algorithms enables

intelligent course recommendations to be produced based on data integrated from multiple

sources. The ultimate aim is the ability to provide a personalised recommendation from a wide range of data sources, focusing on a student's individual needs when choosing a course.

The aim of the thesis research can be divided into the following specific objectives:

1. To study the state of the art of recommender systems, particularly focusing on those that

have been applied in the education domain. Also to examine the tools available to

students for assistance in decision making when choosing a suitable course to meet their

personal needs.

2. To study the tools of preference modelling, concentrating on the methods that employ

user profiles, and primarily analyse how they deal with the problems of initialisation and

dynamic updating of the profile. To design a model to dynamically manage user and item

profiles that provides improvement to the performance of conventional recommender

systems.

3. To develop methods to integrate data from multiple heterogeneous data sources which

will allow a user to quickly access valuable course-related information based on the user's

preferences thereby assisting the user to choose course relevant to their career direction.

4. To develop a framework that can be used by perspective students who plan to choose

university courses. The framework should be able to provide personalised

recommendations that meet with the individual student's needs by combining different

types of recommendation techniques. It should be able to support automatic data

extraction, integration and personalised course recommendations.

5. Based on the framework, to design and implement a personalised course recommendation

system to demonstrate the feasibility of the proposed approaches in a real application.

1.3 Major contributions

This research addresses the existing gap and investigates an approach by which to automatically

extract and integrate course information based on ontology technology and to enhance the

performance of a recommendation system by reducing information overloading in the

education domain. The aggregation of ontology domain knowledge into the recommendation

process is one of the solutions that can overcome the limitations of conventional recommender

systems. Ontology-based (OB) recommenders systems are knowledge-based and use ontology

to represent knowledge about the items and the users in the recommendation process. In

addition, user profiling that is based on ontology, item ontology and the semantic similarity between two ontologies is used to overcome the new user problem.

The main contributions of this thesis lie in the following points:

1. It contributes to the knowledge of current recommender systems by adding insight as to how existing problems are usually tackled and why there still remain shortcomings.

2. It defines a novel Ontology based Personalised Course Recommendation (OPCR) framework by combining several artificial intelligence techniques including collaborative filtering, content-based recommendations, ontological representation and management of knowledge. A set of ontology based recommendation algorithms are developed for personalised recommendation. The framework is thus capable of automatic data extraction, integration and personalised course recommendations to provide students with suitable recommendations to meet with their needs. It aims to not only increase the precision metrics but also to reduce information overloading.

3. The ontology model, designed to extract and integrate information from multiple sources, contributes to improving the diversity of recommendations by overcoming the heterogeneity of course information. In addition, it features properties such as generality

which enable it to be used in different recommendation system domains which change with the user's interests and the object's attributes.

4. A personalised recommendation system is developed and evaluated. The system is available online as open access for users.

1.4 Thesis organisation

This thesis is organised as follows. Chapter 1 Introduction presents a contextualisation of the work and offers a brief explanation of its motivations. The general concepts are clarified and a description of the contributions of the thesis is provided.

Chapter 2 discusses background details and related work and research on recommendation systems and the aspect of ontology. It also highlights different recommendation algorithms and the main challenges faced by general recommender systems, particularly in the education domain. Attention is mainly focused on the collaborative and content-based systems with the strengths and weaknesses of each model being pointed out and an analysis of the research trends in the area of recommender systems is provided.

Chapter 3 presents the Ontology based Personalised Course Recommendation (OPCR) framework and its components in great detail, constituting the core of this work.

Chapter 4 expands the proposed ontology model and its modules and also discusses in detail the recommendation filtering algorithms that are used within the framework.

Chapter 5 continues with the implementation of the actual recommender system, namely OPCR, and its intermediate steps until the generation of recommendations followed by the results of this implementation.

Chapter 6 discusses details of the different approaches of OPCR evaluation followed by the results of user satisfaction measurements.

Chapter 7 concludes by summarising the goals of the thesis, defines the contributions provided and ends with an analysis of future directions for this research.

Finally, the thesis includes the bibliographic references used for its elaboration and 4 annexes that provide information relevant to the thesis.

CHAPTER 2 BACKGROUND AND RELATED WORK

"Research is to see what everybody else has seen, and to think what nobody else has thought." Albert Szent

This chapter discusses the background to the thesis topic and related work regarding the

relevant literature. It provides an up-to-date, state of the art solution in the field of intelligent recommender systems (RS) within the education domain that may be useful, not only to scientists working within this field but to designers and developers of intelligent recommender systems in other domains.

It also discusses how this research relates to previous works undertaken in this area and in what ways it significantly differs from these. Section 2.1 aims to present the theory concerning the recommendation system approaches, a set of features for recommender systems is defined, and a review of the current state of the art is conducted together with an explanation as to why current solutions are not sufficient to address the problem of information overloading.

Section 2.2 refers to recommender systems in a specific field that in this thesis, is course recommendations in higher education, reviews the main shortcomings of current solutions in this area and explains how the proposed system addresses the information overloading problem in the field. Section 2.3 discusses the aspects of ontology in recommendation systems, particularly in the education domain, and how using ontology can extract and integrate information from multiple sources for utilisation in a unified form in order to enhance both the performance of the recommender system and user satisfaction. Moreover, related concepts such as ontology construction, ontology mapping and main challenges are discussed. Finally, section 2.4 presents a summary of the chapter.

2.1 Recommender Systems: main approaches and challenges

The rapidly increasing scope of the internet has given users the facility to choose from an enormous range of information, whether this is information concerning their education, experiences in their world or information that enables them to maintain their lifestyle.

Essentially, to offer a straightforward description, a recommender system can provide recommendations (suggestions) to users in different contexts, such as when they have to choose between a large numbers of items or when they wish to receive suggestions. Recommender systems become particularly helpful in situations where there is an *information overload*

problem, that is, the remarkable array of choices makes the search and selection a challenging task for the user. Information overload was a term introduced to represent the feeling of exhaustion and confusion that occurs because of the cognitive energy required to manage the number of information users has to deal with.

Recommender systems produce a set of technologies and algorithms from various fields such as information retrieval, machine learning, marketing, education, economics and many others.

It has become popular since the mid-1990s, contributing solutions to the problem of information overload on the World Wide Web. Different approaches have been manipulated, each with their advantages and shortcomings. Given the fact that recommender systems are generally established to solve real-world problems, the field is exciting and fulfilling to both the academic domain and business world.

Resnick and Varian (Kembellec, Chartron, & Saleh, 2014) define a recommender system as “a system able to learn users' preferences about different items and use these preferences to propose new items that users might be interested in”. Burke (Burke, 2002) adds a new notion regarding the definition of recommender systems, “a recommender system must be able to provide individualised recommendations and guide users in a personalised way”.

Recommender systems began to attract consideration from researchers in the early nineties (Goldberg, Nichols, Oki, & Terry, 1992). Research into recommender systems spread beyond information retrieval and filtering analysis and began to be applied to a variety of different domains. The object of using recommender systems is to overcome information overload by retrieving the most appropriate information and services from a massive amount of data.

Recommender systems are used by many e-Commerce websites, such as Amazon, to help customers to find appropriate products (Linden, Smith, & York, 2003). In recommender systems, the items can be recommended based on specific information which can be acquired from the demographic data of customers, an analysis of the past purchasing behaviour of consumers as a prediction for future buying behaviour or from the top overall sellers (Adomavicius et al., 2011). However, their application has been extended to fields such as movies, music suggestions, news, bookstores and education (Al-Badarenah & Alsakran, 2016; Cantador, Bellogin, & Castells, 2008; Cui & Chen, 2009; Hsu, 2008; Jones & Pu, 2009; DH Park, Kim, Choi, & Kim, 2012). The main aim of using recommender systems is to reduce information overload by the retrieval of the most relevant information and services from a vast amount of data.

Furthermore, recommender systems can be defined primarily as software programmes that attempt to recommend items to users by predicting users' item preferences based on various types of information, including information about the items, the users and the interactions between users and items. The performance of recommendation systems is influenced by many factors that can affect the quality of recommendations according to the application domain. In their work, Martinez and Lhadj (Martinez & Lhadj, 2013) highlighted the main factors that can influence the results of a recommendation system as shown in Table 2.1.

The widely utilised recommendation system filtering techniques can be categorised into four main approaches. The content-based filtering (CBF) approach recommends items similar to those preferred in the past by the user (Lops, de Gemmis, & Semeraro, 2011). The collaborative filtering (CF) approach recommends items preferred by users with similar needs or interests (Kim, 2013). Knowledge-based filtering recommends items whose features meet users' needs and preferences based on particular domain knowledge (Ruotsalo & Hyvönen, 2007). A hybrid recommendation system is an approach that combines two or more recommendation techniques to overcome the typical shortcomings of each approach (Adomavicius & Tuzhilin, 2005a; Burke, 2002;Ibrahim, Yang, Ndzi, Yang, & Almaliki, 2018). In the following subsections, these techniques are described in detail, and their respective advantages and shortcomings are studied. **Factor** **Description** User factor This includes all basic characteristics such as background, demographics and language Personal factor Behaviour, flexibility to accept or reject recommendations, interest, mood, motivation, trust, intuition and honesty, privacy, awareness of other options, bookmarks, needs, interaction weight, interaction preferences, interactions between users Recommendation Quality, credibility, measurability and weight of the recommendation, reliability, classification, date and time. It is also recommended to include an explanation about why a resource is recommended and who the contributors are. Resources Content, thesaurus, taxonomy, tags, keywords, ratings, reviews, summary, contributors, date, number of votes System Accessibility, usability, parameters, goal, initial data, data analysis techniques, design, architecture, graphical interface Table 2. 1 Factors that influence the recommendation for recommender systems

2.1.1. Collaborative filtering system (CF)

Collaborative filtering is one of the broadly used techniques in recommender systems in order

to overcome the information overloading problem. The idea behind this technique is to assist

people in making their own decisions based on the opinions of other people who share similar interests (Kaminskas & Bridge, 2014). A large community of users is required in order to be able to collect and analyse an immense amount of information regarding user behaviour and characteristics as shown in Fig 2.1. Collaborative filtering systems suffer from a *cold start problem* when there is a lack of data regarding a current user (Adibi & Ladani, 2013). To tackle this problem, the system can offer the top rated item.

Collaborative filtering is considered to be the most popular and widely implemented technique in recommender systems. Since 1990, numerous recommender systems based on the collaborative filtering technique have been created and developed in the worlds of academia and business. These systems have been utilised in many disciplines and uses include suggesting courses, news, articles, movies, products, books, web pages and so forth (Cui & Chen, 2009; Herlocker, Konstan, & Riedl, 2000; Linden et al., 2003; Ray & Sharma, 2011; Ren et al., 2015).

According to previous researchers, there is a number of collaborative filtering algorithms that can be applied to generate recommendations. Collaborative filtering algorithms are mainly divided into two classifications; *memory based* and *model-based* algorithms (Bagherifard, Rahmani, Nilashi, & Rafe, 2017).

Memory-based collaborative filtering utilises user-item rating data to measure the similarity between users or items to provide recommendations (Zhao & Shang, 2010). It is widely used in commercial systems. The fact that the similarity between users is computed by utilising only rating data means that the system is flexible for any products. Nevertheless, the main flaw in this technique is that, since it is calculated using only rating data, the similarity cannot take into

User profile & contextual parameters

Recommendation list

Community data

Figure 2. 1 Collaborative filtering recommendation

account the causes that led to a good or bad rating. Therefore, two users might have liked the same item but for certain different reasons. In Chapter 4 of this thesis, a new method is presented that allows a recommender system to include this new dimension. The popular memory-based method is called the neighbour method that is divided into user-based and item-based. In the user-based method, the similarity between users in their consumption models is used to calculate recommendations. For a target user, the preferences of similar users and the neighbours can assist in recommendations (Desrosiers & Karypis, 2011).

In the neighbour formation case, the similarity between the target user and all other users has to be computed. Several algorithms provide a measure for user similarities such as Pearson's correlation (Tang & McCalla, 2009a) and the cosine-based approach (Chang, Lin, & Chen, 2016) which will be explained next.

$$sim(u_i,u_j) = \frac{\sum_{b_k \in K_{ij}} (r_i(b_k) - \bar{r}_i)(r_j(b_m) - \bar{r}_j)}{\sqrt{\sum_{b_k \in K_{ij}} (r_i(b_m) - \bar{r}_i)^2} \cdot \sqrt{\sum_{b_k \in K_{ij}} (r_i(b_m) - \bar{r}_j)^2}} \quad (2.1)$$

Pearson's correlation coefficient is a measure for the strength and direction of a linear correlation between two variables. Eq.2.1 presents the Pearson correlation coefficient that computes the similarity between two users. In this equation, K_{ij} refers to the set of items that are rated by both users u_i and u_j. $r_i(b_m)$ is u_i's rating from item b_m and $\bar{r}_i$ is the average rating value for user u_i.

In the cosine-based approach, the users are admitted as vectors u_i and u_j in a m-dimensional space, where $m = |K_{ij}|$. The vectors thus represent the rating values for the items that were rated by both users. The similarity is then computed as the angle between those two vectors, as shown in Eq.2.2.

$$sim(u_i,u_j = \cos(\vec{a}_i, \vec{a}_j)$$

$$sim(u_i,u_j = \frac{\vec{a}_i \cdot \vec{a}_j}{|\vec{a}_i| \cdot |\vec{a}_j|}$$

$$sim(u_i,u_j) = \frac{\sum_{b_k \in B_{ij}} r_i(b_k) \cdot r_j(b_k)}{\sqrt{\sum_{b_k \in B_{ij}} r_i(b_m)^2} \cdot \sqrt{\sum_{b_k \in B_{ij}} r_j(b_m)^2}} \quad (2.2)$$

In the item-based approach, the similarity between items with familiar users is employed. The idea behind this is that items that are similar to those that the user has previously rated or utilised are good candidates for a recommendation (Sarwar, Karypis, Konstan, & Riedl, 2001).

Model-based collaborative filtering uses the user-item rating data to build a model that is then used to make predictions for unrated items. The process of building a model can be performed using different statistical and machine learning algorithms (Kumar, 2011). Machine learning techniques inspire these methods, for example, Bayesian networks, artificial neural networks (Salakhutdinov, Mnih, & Hinton, 2007) and latent factor models (Koren, Bell, & Vol insky, 2009).

Cold start problem

These systems face a cold start problem (Schein, Popescul, Ungar, & Pennock, 2002) when there is no available data regarding a current user. The cold start problem occurs when the recommender system cannot draw any assumptions for users or items about which it has not yet collected sufficient information. The first time a user visits a specific recommender system, for instance, none of the items will have been rated. Therefore, the recommender system does not know what the likes and dislikes of that user are. The same problem occurs when a new item is added to the recommender system. Since nobody has ever rated that item, the recommender system cannot know to which other items it is similar. Consequently, the recommender system cannot recommend the item until a large number of users have rated it.

However, one of the advantages of collaborative filtering is that it does not need or require any knowledge about the items in the database, the matrix of users' ratings is the only input necessary.

2.1.2. Content-Based Filtering Models

Content-based filtering is a conventional method which is applied when information overload problems need to be dealt with (Pazzani & Billsus, 2007a). This filtering technique recommends items for the user based on information regarding previously evaluated items for that user. However, this technique suffers from over-specialisation, as it is incapable of determining unexpected items and the user will only receive recommendations for items similar to those that the user has rated before. This problem of novelty is also known as the serendipity problem.

Unlike the collaborative filtering approach, the content-based filtering approach uses content that the user has liked in the past in order to suggest similar content and, as opposed to collaborative filtering, does not utilise other users' interests to issue recommendations (Burke, 2002; Pazzani & Billsus, 2007b). This approach analyses the information from items that have been previously rated. The process of recommendation essentially involves locating suitable

matches between the user profile and the characteristics of the items. One of the advantages of

content-based methods is that they can deal efficiently with the new item problem, that is, they

can recommend new items for which there is no user feedback, unlike collaborative filtering

algorithms.

Furthermore, content-based filtering has proved popular for producing recommendations for

information items when, for example, the user marks/buys certain items of interest and the

system then offers the items which are most similar items to the user's favourite items. These

systems need a great deal of detail about the items in the database to be able to recommend

similar ones and about the user profile that describes what the user likes. On the other hand, it

does not require a large community of users as Fig. 2.2 shows. Every content-based

recommender system has the following three goals:

- to analyse item descriptions and documents
- to build a user profile
- to compare favourite items with other items in the database

The architecture of such a recommender system was published in (Lops et al., 2011). The

authors describe the three main components that contribute to reaching the above goals:

- *The content analyser* provides extraction of structured, relevant information (usually keywords) from texts to be able to process these further.

User profile & contextual parameters

Item features

Recommendation list

Figure 2.2 Content-based recommendation

• *The profile learner* gathers information about user interests such as what item they select,

rate or leave other feedback for (Ruthven & Lalmas, 2003), employs machine learning

algorithms (Han, Kamber, & Pei, 2012) and creates a user profile.

• *The filtering component* utilises the user profile to recommend relevant items by matching

the user profile to corresponding items in the database. It uses different similarity metrics

(Zezula, Dohnal, & Amato, 2006), cosine similarity, being one of the most often used as

presented in Eq.2.2, creates a ranked list of items and suggests these to the user.

Proposals for content-based recommendation algorithms attract perspectives and algorithms

from different domains such as information retrieval, semantic web and machine learning. For

instance, term-weighting models from information retrieval were employed in early proposals

for web recommendations (Balabanović & Shoham, 1997). However, in the current content-

based filtering approaches, the data is not modelled correctly. Using ontologies to model the

data provides a better modelling quality and thus more suitable recommendations because

better-modelled items mean more rigorous user preferences have modelling abilities which can

produce more accurate similarities (Maidel, Shoval, Shapira, & Taieb‐Maimon, 2010).

Methods using semantic web technologies have also been introduced for content-based

recommendations, as in the case of news recommendation (Cantador et al., 2008; Kumar,

2011), or movie and music recommendations leveraging Linked Open Data (Ostuni, Di Noia,

Mirizzi, & Di Sciascio, 2014). Regarding the use of machine learning techniques, Mooney and

Roy (Mooney & Roy, 1999) used Bayesian classifiers for book recommendations, and Pazzani

and Billsus (Pazzani & Billsus, 1997) used numerous techniques such as Bayesian classifiers,

clustering, decision trees and artificial neural networks for website recommendation.

Limitations of the content-based approach

Content-based recommender systems have several limitations which have been identified in

the literature (Burke, 2002; Ekstrand, Harper, Willemsen, & Konstan, 2014). The most relevant of these are:

• Limited content analysis. Content-based recommendations are constrained by the features that are explicitly associated with the items to be recommended. For example, content-based course recommendations can only be based on material written about a course: course title, course fee, university name. The effectiveness of these techniques thus depends on the available descriptive data. Therefore, in order to have a sufficient set of features, the content should be either in a form that can be automatically parsed by a

computer or in a form in which the features can be manually extracted easily. In many cases, these requirements are very difficult to fulfil. There are certain domains where automatic feature extraction is complicated and to assign features manually is often not practical. For instance, even if a recent attempt underlines the need for further research in this direction (Li, Ogihara, & Li, 2003), it is much harder to apply automatic feature extraction methods to multimedia data such as graphical images, video streams and audio streams, than it is for text content. A recent trend is to enrich content representation by means of external knowledge sources, such as ontology based ones. The Explicit Semantic Analysis (ESA), introduced in (Markovitch & Markovitch, 2006), proposes an indexing technique based on content gathered from Wikipedia articles. An early attempt of coupling content-based filtering based on ontology with techniques for knowledge infusion is proposed in (Musto, 2010).

- Content over-specialisation. Content-based filtering based on ontology retrieves items that score highly against a specific user profile. Content-based techniques cannot recommend items that are different from anything the user has seen before. Thus, for instance, a person with no experience in ambient music will never receive recommendations about that genre if he has never enjoyed something at least similar to it. To overcome such limitations, it may be appropriate to introduce an element of randomness in the recommendations (Maes & Sharadanand, 1995). Alternative approaches, such as that implemented in DailyLearner (Billsus & Pazzani, 2000), propose to filter out items not only if they are too different from user's preferences but also if they are too related to something the user has viewed previously. Furthermore, in (Zhang, Callan, & Minka, 2002) a set of five redundancy measures is provided in order to evaluate whether a document that is deemed to be relevant contains some novel information as well.

- Cold-start. Before a content-based recommender system can really grasp user preferences and provide reliable recommendations, each user has to rate a sufficient number of items.

However, the recent explosion of Web 2.0 and social platforms has changed the rules for

user profiling since, in principle, it is possible to reuse the information the user has

already provided (such as comments, posts, tags or data gathered from social networks)

and to exploit such information as a starting point to incrementally build and model the

user profile. In this area, a recent trend is represented by social media-based user profiling

(Bu et al., 2010).

It is significant to note that content-based techniques are not based on the real content of items.

For instance, in the context of course recommendation, a content-based filtering system does

not concern itself with the content of the course, it is merely based, at best, on the description,

keywords or course title. Most of the time, the “content” is simply the genre, author or other

metadata. Also, content-based filtering methods do consider the textual content that has been

written about items by users, blogs or whatever. They generally apply semantic analysis by

using ontologies.

2.1.3 Knowledge-Based Filtering Models

These techniques are usually utilised for an explicit representation of knowledge, as with a case-based reasoning system, ontology or other forms of rule systems. Items are recommended to users according to inference about a user's interests (Middleton, De Roure, & Shadbolt, 2009). A case-based system relies on the notion of using past problem-solving experiences as a primary source to solve a new problem.

A good example of this knowledge-based method (KB) is presented in the work of Burke (Burke, 2002). In his work, the model was designed to help a user to find restaurants that matched his/her preferences through the use of interactive dialogue. The user could change the retrieved suggestion by refining the search query based on user interest until achieving the suitable option. The other type of knowledge-based method is the use of semantic similarity to recommend items to users. Ontologies have been applied to a variety of recommender systems to reduce content heterogeneity and to improve content retrieval. For example, in (Obeid, Lahoud, El Khoury, & Champin, 2018), good results to cope with content heterogeneity have been obtained by using subsumption hierarchies to generalise user profiles.

Furthermore, the concept of the semantic web has been used to improve e-learning. In (Yang, Sun, Wang, & Jin, 2010), Yang et al. proposed a semantic recommender system approach for use in e-learning to help learners to define suitable learning objectives. Moreover, the system could assist instructors by suggesting new resources that could be adopted to enhance the syllabus of the course. This system has been built with a query keywords extension and uses both semantic relations and ontology reasoning. The authors in (Ren et al., 2015) presented a personalised ontology-based recommendation system which is similar to the two approaches mentioned above. It represents items and user profiles in order to provide personalised services that use semantic web applications. The evaluation shows that the semantics-based methods of the recommender system improve the accuracy of the recommendations.

A recommendation system based on ontology can also solve the cold start problem which occurs when using information from the past is insufficient (Zhou, Yang, & Zha, 2011). Indeed, this problem occurs due to an initial lack of ratings for new users and hence it becomes impossible to make reliable recommendations. An ontology-based model has been proposed for e-learning personalisation which would recommend learning objectives by judging the past preference history of learners. Like traditional systems, this system suffers from a new user problem and is limited to learning objectives only (Ambikapathy, 2011). Ontology structures significantly improve the ontology which can lead to increased accuracy (Bagherifard et al., 2017). For instance, all of the “IS-A”s relations in the ontology for measuring semantic similarity were considered to be similar in a hierarchical tree in which the associations between the concepts were shown by “IS-A”. Calculating the similarity between the two concepts is made less accurate by this.

The knowledge-based recommender system has certain advantages and disadvantages (Adomavicius & Tuzhilin, 2005b; Sieg, Mobasher, & Burke, 2010). On the positive side, it has no cold start problem; the system can recommend items to a new user based on simple knowledge of his/her preferences. It will therefore not require the user to rate or buy very many items in order to provide satisfactory recommendations. On the negative side, the knowledge-based method system faces a scalability problem where more time and more effort are needed to calculate the similarities for a larger case-base compared to other standard recommendation techniques. The knowledge-based method has a further weakness, which is that the system needs to include some information about items, users and functional knowledge in order to produce recommendations.

2.1.4 Hybrid Based Filtering

Models

Hybrid systems utilise a combination of the methods mentioned above to overcome their disadvantages and utilise their strengths (Burke, 2002). For instance, collaborative filtering techniques fail to handle the new-item problem, ie they are unable to recommend items that have not yet been rated. However, new items characteristics (information) are generally available and can be used with content-based methods as shown in Fig. 2.3. The first type of hybridisation is to select different approaches and to allow each to produce a separate ranked list which is then merged into one final list (Claypool et al., 1999). Several other hybrid

approaches are based on one model which uses the second model only to overcome shortcomings, eg the cold start problem or to improve user profiles (Pazzani, 1999).

Burke (Burke, 2002) defined the taxonomy for the hybrid recommendation systems which he

listed into the following seven categories:

- **Weighted**: Several recommendation component scores are combined numerically. This

  category aggregates scores from each factor using an additive formula. For instance, the

  easiest combined hybrid would be a linear combination of recommendation scores, as in

  P-Tango (Claypool et al., 1999). This method initially provides equal weight to

  collaborative and content-based recommenders. However, the weighting is constantly

  adapted according to users' feedback on predictions.

- **Switching**: From the available recommendation components, the system chooses a

  particular component and applies the one selected. As an illustration, the recommendation process starts with a content-based recommender, but switches to a collaborative one

when the confidence level on the recommendations already presented are not sufficient

as shown in the work of Billsus and Pazzani (Billsus & Pazzani, 2000). However, this

switching hybrid does not entirely avoid the ramp-up problem, since both the

collaborative and content-based systems feature the new user problem.

• **Mixed**: Different recommender systems produce their recommendations that will be

introduced together. This method is based on the merging and presentation of multiple

rated lists into a single rated list. This method is implemented in the PTV system (Smyth

& Cotter, 2000)

User profile & contextual parameters

Community data

Item features

Figure 2.3 Hybrid systems

Recommendation list

• **Feature Combination**: Contributing and actual recommenders are the two different recommendation components that exist for this category. The working of an actual recommender is dependent on the data modified by the contributing recommender which throws features of one source onto the other component's source.

- **Feature Augmentation**: This category is similar to the feature combination hybrid with the only difference being that the contributor gives novel characteristics. It is more flexible than the feature combination method.

- **Cascade**: This category plays the role of tiebreaker. Here, every recommender is assigned a certain priority and, according to that assigned priority, lower priority recommenders play a tiebreaker role over those with higher priority. Usually, one recommendation system is applied to generate a set of candidates and the second algorithm filters and re-ranks this to produce the final list. The cascade hybrid is generally more efficient than a weighted one that applies all of its methods to all items.

- **Meta-level**: The model generated by one of the recommenders is used as the input for another recommender. As stated in (Burke, 2002): "this differs from feature augmentation: in an augmentation hybrid, we use a learned model to generate features for input to a second algorithm; in a meta-level hybrid, the entire model becomes the input."

2.2. Recommendation systems in the higher education domain

"Applying to university is a big decision, and we want to ensure that all students, whatever their background, have the key facts at their fingertips to help them make the

right choice for them.”

Dr Vince Cable

As highlighted in section 2.1, the recommendation problem widely applies to many domains although its main uses are within the fields of e-commerce and entertainment. Most developed systems aim to help the user to decide which products to buy (or consume). Successful application of e-commerce recommenders are reported elsewhere (Sarwar, Karypis, Konstan, & Riedl, 2000).

This success, as mentioned earlier, has motivated the implementation of recommender systems in the educational domain (Manouselis, Vuorikari, & Assche, 2010). In this domain, the ultimate goal is that learners acquire knowledge and that educators support the learning process.

In other words, there are clear differences that impinge on how to design recommender systems for each domain (Drachsler, Hummel, & Koper, 2008a). Recommending items in the education domain, either directly or indirectly, has the goal of improving the decision making process and the selection of appropriate courses. In the education domain, a recommendation system is an intelligent agent that suggests different alternatives to students which takes into account, as a starting point, the previous action from other students with approximately the same characteristics, such as academic performance and other personal information (Park, 2017). It is a fact that, before taking a course, the student has to enrol on the course. However, the most notable aspect of this process is not the enrolment itself, but the decision before that

which has
to be taken(Ibrahim et al., 2018).
In this section, the most important literature regarding these types of recommendations
systems
with reference to the dimension of the items recommended and the different
approaches that
have been used to recommend items in the education domain will be reviewed.
However,
recommender systems in education are entirely different from recommender systems in
e-
commerce, as they have to consider not only the students or the educator's preferences
for
certain learning materials but also how this material may help them to obtain their goals
(Bozo,
Alarcón, & Iribarra, 2010). In table 2.2, a comparison has been made of the important
differences and factors between a general purpose recommendation system and an
educational
recommendation system.
Difference factor General recommendation
system
Educational recommendation
system The goal In fields such as e-commerce, a user is looking to buy a product of a
specific quality and in a specific price range (Nikos Manouselis, Drachsler, Vuorikari,
Hummel, & Koper, 2011)
Educational recommender systems help the user or a group of users to find suitable
resources and learning activities for optimum achievement of learning goals and the
development of competences in less time (Drachsler et al., 2009; Drachsler, Hummel, &
Koper, 2008b) The context Most recommender systems share factors such as networks
and peer information (J. Hu & Zhang, 2008; Ramadoss & Balasundaram, 2006; Santos
& Boticario, 2009; T. Tang & Mccalla, 2004)
The context of educational recommender systems is pedagogically related. Factors that
should be considered as part of the context are pre and post-requisites, timeframe and
instructional design (Abel, Bittencourt, Henze, Krause, & Vassileva, 2008; Santos &
21
Boticario, 2009), and social networks (Q. Yang et al., 2010) User influenced by
Recommender systems are mostly based on user tastes, personal preferences or what
a user likes or dislikes (Santos & Boticario, 2009; TY Tang & McCalla, 2009b)
Highly influenced by pedagogical factors such as learning history, knowledge,
preferences, processes, strategies, styles, patterns, activities, feedback,

misconceptions, weaknesses, progress and expertise (Abel et al., 2008; Gasparini, Lichtnow, Pimenta, & Oliveira, 2009; Masters, Madhyastha, & Shakouri, 2008; Prieto, Menéndez, Segura, & Vidal, 2008; Wan, Ninomiya, & Okamoto, 2008; Q. Yang et al., 2010)

Table 2.2 Difference of the factors in a general recommender system and an educational recommender system

A significant number of recommender systems have been propo sed in the education domain, as well as in teaching and academic guidance. In the education domain, the target users are students, teachers or academic advisors and the recommendable items are educational materials, universities or information such as courses, topics, student performance and the field of study.

In this thesis, to evaluate the proposed methodology, the focus was how to reduce information overloading for the student who is required to make a decision and thus to provide them with the most appropriate university course. The word “course” in the thesis refers to the programme of studies such as undergraduate courses or postgraduate courses. Many types of research have been undertaken, using different techniques and algorithms, that have been used to recommend courses to students. For instance, Sandvug and Burke presented the Academic Advisor Course Recommendation Engine (AACORN) that used a case-based reasoning approach which utilised knowledge acquired from previous cases in order to solve new problems (Sandvig & Burke, 2005). Their system used both the course histories and the experience of past students as the basis of assisting future students in course selection decision making.

At the same time, it was noticed that the intended future career of students was an essential factor which could influence their decision to choose a particular course (Huang, 2017). Farzan and Brusilovsky proved this by using a reported course recommendation system based on an adaptive community (Farzan & Brusilovsky, 2006). They employed a social navigation approach to analysing the students' assessments of their career goals in order to provide

22

recommendations for courses. The main object of this approach was to obtain the students' explicit feedback implicitly, as part of their natural interaction with the system.

In this respect, Artificial Intelligence techniques could help to develop and improve the decision making and reasoning process of humans to minimise the amount of uncertainty there is in active learning to ensure a lifelong learning mechanism (Bobadilla, Ortega, Hernando, & Gutiérrez, 2013). The challenge for recommender systems, therefore, is to better understand the student's interest and the purpose of the domain (Ibrahim, Yang & Ndzi, 2017). An association mining based recommender system has been developed for recommending tasks that are related to learning which is most suitable for learners based on the performance of the targeted student and other students who are similar to them (Noakes et al., 1968). A course recommendation system has been proposed that would check how similar university course programmes are to the students' profiles.

The proposed framework in this thesis is comprehensive in that it combines content based filtering and collaborative filtering with an ontology technique in order to overcome the problem of overloading information. It does this by utilising a similar hierarchal ontology to map the course profiles with the user (student) profile. The new approach develops two novel methods to extract and integrate data from multiple sources and then align the data appropriately. This ontology mapping of the different data improves the ability to obtain a comprehensive knowledge of the recommended items. The approach tackles the new user problem by calculating the ontology similarity that exists between the users' profiles by measuring the user rates for each item. The proposed recommender system is used to evaluate the hierarchy ontology similarity there is between the item profiles and the users' profiles before the student chooses a course to match his/her requirements and enrols on the programme.

2.2.1 Course Recommender System

Research shows that students are overloaded by a large amount of information available when choosing a course (Huang et al., 2013). Nowadays, the range of course-related information available to students is still rapidly increasing (Diem, 2015). This abundance of information has created the need to help students to choose, organise and use resources that match their individual objectives, interests and present knowledge (Farzan & Brusilovsky, 2006). Dr Cable, a British politician who was the Secretary of State for Business, reported, “Applying to university is a big decision, and we want to ensure that all students, whatever their background, have the key facts at their fingertips to help them make the right choice for them".

and Aïmeur report students pursuing education are faced with two challenges: a myriad of courses from which to choose, and a lack of knowledge about which courses to follow and in what sequence (Noakes et al., 1968). Also, the heterogeneity of course information and the user's personal needs make the decision process very tedious and complicated (Ge, Chen, Peng, & Li, 2012). At the same time, Wendy Hodgkiss, a careers adviser with the 'Which? University' organisation reported, “Don't just 'grab' something in a panic. Do some research first and make sure you really want to go to the relevant university and course”[1].

Amer and Jamal showed that course choice decision is influenced by the bac kground of the student and personal or career interests (Al-Badarenah & Alsakran, 2016). However, offering more course information on university websites does not necessarily infer that the students will have the cognitive ability to evaluate them all as alternatives. Instead, it confronts students with a problem usually termed as "information overloading" (Huang et al., 2017; Obeid et al., 2018; Shrivastav & Hiltz, 2013; Sieg et al., 2010). Evaluating all course alternatives themselves is a challenging task for students, even when some search tools do exist. How to automatically find the relevant course to match with the students' needs is a pressing problem (Ibrahim et al., 2018). Many online systems have been made available to help students to find and compare different courses across different universities such as UCAS[2], ukcoursefinder[3], unistats[4] and comparetheuni[5]. However, these tools have been recognised as working based on previous user's knowledge of courses only or on keyword-based queries. Furthermore, for a given course query, a student will receive hundreds of results in a random order thus, again, students will be overloaded with information and potentially irrelevant results.

Good progress has been made in the course recommendation system which aims to support students to find suitable courses. Excellent work has been achieved in building a course recommendation system based on a collaborative filtering approach (Carballo, 2014; Ray & Sharma, 2011). In addition, CourseAgent is a significant work which is a community-based course recommendation system that uses the social navigation approach (Farzan & Brusilovsky, 2006) to produce recommendations for courses based on a student's estimation of their appropriate career goals. The main focus of this method is to collect explicit feedback from students implicitly, as part of their natural communication with the system. The basic and obvious advantage of the system to students is as a course administration system that stores

information about courses they have chosen and facilitated communication with their advisors.

[4]http://comparetheuni.com

[5]http://www.ukcoursefinder.com

[1] retrieved from http://www.hefce.ac.uk at 25 Oct. 2018

[2]https://www.ucas.com [3]https://unistats.direct.gov.uk

Sandvig and Burke reported a new course recommendation system called AACORN, which is

a case-based reasoning approach(Sandvig & Burke, 2005). The system utilises the experience

of previous students and their course histories as a place to start to advise course selection. In

order to discover the similarities between course histories, the system uses a metric broadly

used in bio-informatics named the edit distance. The system demands details of a partial history

of the courses taken by a student before it can supply effective recommendations.

The RARE recommender system combines association rules with user preference data to

recommend relevant courses (Noakes et al., 1968). RARE was used on real data derived from

the Department of Computer Science at the Universite de Montreal. It analyses the past

behaviour of students regarding their choice of course. More explicitly, it formalises association

rules that beforehand were implicit. These rules allow the system to predict recommendations

for new students. A solution to handle the cold start problem, which is central for recommender

systems, is also proposed in RARE.

PEL-IRT refers to the Personalised E-Learning system which applies item response theory

(Chen, Lee, & Chen, 2005). It recommends suitable course material to students, bearing in

mind both the difficulty of the course material and student ability. When utilising PEL-IRT,

students can choose course categories and units and can use relevant keywords to search

interesting course material. Once the course material has been suggested to students, and they

have browsed through the information, the system requires them to answer two questionnaires.

This explicit feedback is employed by PEL-IRT to re-evaluate the students' abilities and to

customise the course material difficulty featured in the recommendation.

Recently, academics have found that personalisation is an influential factor used to increase the

accuracy of recommendations and information retrieval (Huang et al., 2017; Salahli, Özdemir,

& Yaşar, 2013). Punj and Moore realised that recommendation agent that can filter and

integrate information and offer feedback influenced the user's decision more in comparison to

agents that are only aware of the alternative options (Punj & Moore, 2007). Furthermore, it was

realised that the relevant course recommendations outcome integrates useful information from

multiple useful sources such as jobs sites, social networks and other related educational data sources (Huang et al., 2013).

The Course Recommender System is based on several different collaborative filtering algorithms such as user-based (Cone, 2011) and item-based (Sarwar et al., 2001), OC1 (Murthy, Kasif, Salzberg, & Beigel, 1993). The Course Recommender System (Mahony & Smyth, 2007)

is based on a variation of the widely-used item-based collaborative filtering algorithm. The purpose of a module recommender system is to facilitate and enhance the online module selection process by recommending optional modules to students based on the core modules that they have selected. Applying historical enrolment data points leads to a very encouraging performance concerning both recall and coverage. Certain recent research has focused on using course recommender systems in niche areas such as civil engineering professional courses (Zhang, 2009) and physical education courses at universities (Liu, Wang, Liu, & Yang, 2010).

From a study of the literature, it is obvious that the recommendation technology applied in the field of education can facilitate the teaching and learning processes. Considering the significance and importance of education, the assistance of a recommendation system can improve efficiency and increase the validity of learners in the actual educational situation.

All the above studies highlight the importance of course recommendation systems.

However,

all the current systems that provide information regarding a suitable course for students use a

single data source such as the students themselves, courses histories or university records.

However, the proposed search in this thesis seeks to build a new course recommendation system

based on integrating information about courses from multiple data sources such as university

websites, job websites and social networks. It will provide the students with recommendations

that meet their personal needs, interests and career aspirations and therefore support the

decision making process. At the same time, it will reduce information overload and

heterogeneity through the use of ontology-based data integration of the search results of the

relevant course. Accordingly, this creates the need for software that can automatically avoid

the irrelevant choices, gathering information about the choices and allowing users to see only

the more appropriate options that best match their needs.

Nevertheless, despite the high impact of course recommendation systems and how useful they

are, there remain certain significant limitations, such as:

• Models based mainly on keywords fail to address the individual user's needs in the

recommendation process. Although models use collaborative filtering and data mining such as association rule and decision trees, there is often a lack of historical information

that makes it challenging to use this approach. For instance, new students who want to

use the systems cannot generate any recommendations since the system has insufficient

information about them.

• The shortcoming of models that use content-based filtering is that current approaches are

based only on a specific subject recommendation rather than a whole university course.

Moreover, the similarity calculation in these models is based on the weighted average of

features and does not take into account user interaction with the system, such as the rating

value of recommendation items.

• An additional shortcoming of the current models is that they do not provide

comprehensive knowledge about the course that is most relevant to the student. For

example, students need to know what future career the course will lead to and require

information about this aspect, as well as the quality of the facilities of the educational

institution that will be providing the course.

By categorising the needs of students and their areas of interest, it is possible to recommend an

appropriate course. It is possible to help students to select a course by developing methods that will both integrate the data from multiple heterogeneous data sources and allow this to rapidly set valuable course-related information (Huang et al., 2013). By using ontology, the user will be able to obtain precise knowledge about the course (Wang & Sapporo, 2006). It is possible to build a relationship between the relevant information on the internet including the course modules, job opportunities and the users' interests. Ontology provides a vocabulary of classes and properties that can be used to both describe a domain and emphasise knowledge sharing (Hongji Yang, Zhan Cui, & Brien, 1999). The use of semantic descriptions of the courses and the student profiles allows there to be both qualitative and quantitative reasoning about the matching, as well as the required information about the courses and the student's interests, which is required in order to refine the selection process of which course to take.

A novel hybrid filtering system is proposed in this thesis, based on both the content based and collaborative filtering methods and using an ontology as the way to overcome the problem of information overloading which has been a key challenge when facing the building of an effective recommendation system. The proposed approach uses an ontology for data extraction and integration from multiple data sources. Data integration that is based on ontology is used in the ontology-based metadata. It uses a combination of model-based and memory-based ontology in collaborative filtering to provide a high-quality recommendation.

User profiling that is based on ontology, item ontology, the semantic similarity between two ontologies and the proposed OKNN algorithm is used in the collaborative filtering aspect to overcome the new user problem (more details of OKNN are provided in chapter 4). On the

other hand, item-based ontology and semantic similarity are both applied in content-based filtering to overcome the new item cold start problem. In order to make the measurement of semantic similarity more accurate, a hierarchy concept similarity approach is used in the content-based filtering. This measures the "IS-A" degree between the two nodes of item ontology which was found to yield a more precise recommendation list for the target user.

2.2.3 Challenges in the course recommendation system

There are many challenges that it is necessary to overcome for the implementation of education recommender systems to be effective. In this section, the main challenges found in the literature and ways to address these are summarised.

Information overload: This problem refers to when, in an environment such as the internet,

the amount of pedagogical content is overwhelming and widely spread over the generating network (Wang, 2008). This leads to information overload, making it difficult for students to find and evaluate quality information regarding the most suitable learning resources (Gasparini, Lichtnow, Pimenta, & De Oliveira, 2009; Yang & Wu, 2009).

Lack of structure in the data: One of the primary difficulties and essential characteristics of a recommender system is the process by which the data is structured. (Pearce, 2008). For example, in social learning environments, information tends to be classified in just one category thereby reducing the number of options to the user (Abel et al., 2008). In addition, a predefined structure does not exist in a social network (Nikos Manouselis, Drachsler, Verbert, Santos, & Konstan, 2014) and information cannot be reused in other systems because of a lack of structure which hinders the interoperability among recommender systems (Nikos Manouselis & Vuorikari, 2009).

New user and cold start problem: This problem occurs when there are no ratings for new resources or when a new user has not yet rated any items (Tang & McCalla, 2009a). Suggested ways to overcome this challenge include 1) a knowledge provider can be the first starter; consequent users can contribute to this elaboration (Rafaeli, Dan-Gur, & Barak, 2005); 2) use of artificial learners (Tang & McCalla, 2009a); 3) use of information related to the completion of activities and similar preferences (Manouselis et al., 2014).

Cognitive overload: This refers to the effort required during the process of selecting useful resources or assigning accurate ratings (Rafaeli et al., 2005). This takes place particularly when there is raw data, when the user is unable to ask the right question, when pedagogical resources

are not properly defined by the expert (DeLong, Desikan, & Srivastava, 2006), when resources are not classified and when there are no existing summaries, keywords or other types of descriptors (Yang, Huang, Tsai, Chung, & Wu, 2009). One way to overcome this issue is to use content analysis techniques such as data mining to find keywords or structures (Yang et al., 2009).

Quality of the recommendation and trust: Another problem arises when users do not trust the system and the recommendations. The probability that a user will perform an action based on the recommendations is often too low (Chang et al., 2005; Schulz et al., 2001). For that reason, it is suggested that the quality of a recommendation is always defined (Zheng & Li, 2008), that it should be made clear whether a recommendation is either precise or simply relevant, that biased recommendations are reduced as much as possible (Schulz et al., 2001) and that it is made clear where the recommendations come from (Rafaeli et al., 2005) or how new items are added.

2.3。 Ontology

Modelling information at the semantic level is one of the main purposes of using ontologies (Guarino et al., 2009). This section gives a detailed review of ontology including ontology definitions and a description of the ontology development process. It then discusses the application of ontology to data extraction and integration and the use of ontology in recommender systems.

2.3.1 Ontology Definition

The original definition of “ontology” in computer science was provided by Gruber (Gruber, 1993) as an “explicit specification of a conceptualisation”. Borst defines ontology as a “formal specification of a shared conceptualisation” (Borst, 1997). Coelho *et al*. gave a new definition of ontology “as a knowledge domain conceptualisation into a computer-processable format which models entities, attributes, and axioms”. Ontology is typically made up of vocabulary and relationships between concepts (Coelho, Martins, & Almeida, 2010). According to Antoniou and Harmelen (Antoniou & Harmelen, 2008), ontologies are concept properties, disjointedness statements, value restrictions and specifications of logical relationships between

objects. Ontologies have provided a tool for formally modelling the structure of a system based

on the relationships that emerge from its observation.

The term taxonomy is used when the ontology contains only “IS-A” relationships, and normally

the use of the word ontology is restricted to systems that support a rich variety of relationships

between the concepts, including logical propositions that formally describe the relationship.

Many ontology classifications have been established (Grazia, Bono, Pieri, & Salvetti, 2004).

For example, ontology can refer to the specific domains that can provide conceptual modelling

of a particular domain.

Ontologies can be classified into three categories - Domain Ontology, Upper Ontology and

Application Ontology. Domain Ontology represents the vocabulary related to a generic domain

such as education, medicine or automobiles; or any generic task or activity such as selecting or

diagnosing by specialising the terms introduced in the top-level Ontology. Upper Ontology,

also called Foundation Ontology, is a model of the common objects that can apply to a wide range of Application Ontology. An Application Ontology defines concepts of a particular domain and task. In the application domain, Upper Ontology, as well as Domain Ontology, can be integrated with Application Ontology. Practical descriptions on ontologies have shown their importance in several respects:

- An ontology involves the factorisation of knowledge. Like the oriented object approach, knowledge is not repeated in each instance of a concept (Rinku & Aravind, 2016).

- An ontology provides a unified framework to reduce or eliminate ambiguities and conceptual and terminological confusion (Daramola, Adigun, & Ayo, 2009).

- An ontology can significantly increase the performance of search engines. Through the semantics provided, an ontology can address problems such as the noise and silence of the traditional search engines (Ringe & Francis, 2012).

- An ontology can support the sharing and reuse of knowledge (Shvaiko & Euzenat, 2013). the researcher can reuse existing ontologies and, if adapted to meet with their need, will reduce the time of creating an ontology from scratch.

- An ontology implements mechanisms of deductive reasoning, automatic classification, information retrieval and ensures interoperability between systems.

In the following sections, this discussion will focus on all of these aspects of the ontology domain. The importance of ontology in the field of education will be explained, and the architecture of the framework and the procedure of acquiring and representing an ontology in that domain will be specified. Ontologies are classified according to their level of dependence

on a particular task or point of view into three categories (Kawtrakul, Suktarachan, & Imsombut, 2003):

- Upper-level ontologies are domain independent and intended to capture and represent the semantics of the real world to support large applications. An example of this type is the "Cyc project" which attempts to capture and encode large amounts of common sense knowledge about the real world.

- Domain ontologies specify concept relationships between concepts and inference rules for specific domains in a specific way (eg travel reservations, soccer and gourmet food).

- Application ontologies describe concepts relative to a task domain such as the reasoning process for medical diagnosis. According to the classification mentioned above, the ontology developed in this thesis is categorised as an application ontology which is to be utilised within the e-Government service domain.

In this thesis, ontology has been used in the three main areas of the proposed framework - the data gathering component used an ontology to extract information from multiple sources based on the hierarchy structure of information, the ontology model component, used to construct and map ontologies, and the recommendation engine component. The following subsection describes the background to using ontology in the relevant domain of this thesis.

2.3.1 Ontology Representation

The main principal elements in ontologies are concepts, relations, axioms and instances. The definition of each element, according to (Noy et al., 2001), is presented below:

A **Concept** (also known as a term or a class) is the essential abstract component of a domain. Typically, the class represents a group of common properties owned by many members. Also, classes are arranged in hierarchical graphs on two levels. Higher level classes are called parent classes, and the subordinate levels are called child classes. A graph of concepts might organise classes in a lattice or a taxonomic view; for example, the class “Faculty” could have many subclasses, such as “Department” and “College”. Moreover, the concepts might have many different distinguishable properties.

A **Relation** (also known as a slot) is used in the ontology structure to provide a declaration for the relationships between concepts in a specific domain. In order to specify the two classes involved in a particular relationship, one of them will be described as a “Domain” and the other

one as a “Range”; for instance, the relationship “Work” can have the concept of “Employee” as a domain and “Faculty” as a range.

An **Axiom** (sometimes called a facet or role restriction) is utilised in the ontology to force restrictions on the values of both classes and instances. Logic-based languages, such as first-order logic, have been developed in order to express these constraints. Furthermore, these languages can be used as the verification process for the consistency of the ontology structure.

An **Instance** (also known as an individual) is a relationship between ontology concepts in relation to their real values; for instance, “Iraq” could be an instance of the class “Asian countries”, or simply “countries”.

2.3.2 Data Extraction Using Ontology

Vast amounts of information can be found on the web (Vallet *et al*, 2007). Consequently, finding relevant information may not be an easy task. Therefore, an efficient and effective approach which seeks to organise and retrieve relevant information is crucial (Yang, 2010).

With the rapid increase of documents available from the complex WWW, more knowledge regarding users' needs is encompassed. However, an enormous amount of information makes pinpointing relevant information a tedious task. For instance, the standard tools for web search engines have low precision as, typically, some relevant web pages are returned but are combined with a large number of irrelevant pages mainly due to topic-specific features which may occur in different contexts. Therefore, an appropriate framework which can organise the overwhelming number of documents on the internet is needed (Pant *et al.*, 2004).

The educational domain is one of the domains that have been affected by this issue (Almohammadi, Hagras, Alghazzawi, & Aldabbagh, 2017). As the contents of the web grow, it will become increasingly challenging, especially for students seeking to find and organise the collection of relevant and useful educational content such as university information, subject information and career information(Chang *et al.,* 2016). Until now, there has been no centralised method of discovering, aggregating and utilising educational content (Group, 2009) by utilising a crawler used by a search engine to retrieve information from a massive number of web pages. Moreover, this can also be useful as a way to find a variety of information on the internet (Agre & Dongre, 2015). Since the aim is to find precise data on the web, this comprehensive method may not instantly retrieve the required information given the current size of the web.

Most existing approaches towards retrieval techniques depend on keywords. There is no doubt that the keywords or index terms fail to adequately capture the contents, returning many irrelevant results causing poor retrieval performance (Agre & Mahajan, 2015). In this thesis, a new approach to web crawler based on ontology is proposed which is used to collect specific information within the education domain. In this thesis, the approach focuses on a crawler which can retrieve information by computing the similarity between the user's query terms and the concepts in the reference ontology for a specific domain. For example, if a user seeks to retrieve all the information about master's courses in computer science, the crawler will be able to collect all the course information related to the specific ontology designed for the computer science domain.

The crawling system described in chapter 3 matches the ontology concepts thus giving the desired result. After crawling concept terms, a similarity ranking system ranks the crawled information. This reveals highly relevant pages that may have been overlooked by focused standard web crawlers crawling for educational content while at the same time filtering redundant pages thereby avoiding additional paths.

2.3.3 Ontology Mapping

Ontology mapping is also known as ontology matching or ontology alignment. Ontology mapping or matching is different from ontology merging. Ontology mapping tries to make the source ontologies consistent and coherent with one another while keeping them separate. In contrast, ontology merging aims to create a single coherent ontology that includes the information from all the sources. Ontology mapping is used to “establish correspondences among the source ontologies, and to determine the set of overlapping concepts, concepts that are similar in meaning but have different names or structure, and concepts that are unique to each of the sources” (Noy et al., 2001). It is also defined by Kalfoglou and Schorlemmer (Kalfoglou & Schorlemmer, 2003) as follows, “Given two ontologies O1 and O2, mapping one ontology onto another means that for each entity (concept C, relation R, or instance I) in ontology O1, it tries to find a corresponding entity, which has the same intended meaning, in ontology O2”. This research defines ontology mapping so as to find a set of semantic correspondences between similar elements in different ontologies.

Various works have been developed to support the mapping of ontologies. An interesting survey which gathered more than 30 works is presented in (Kalfoglou & Schorlemmer, 2003).

In(De Bruijn et al., 2006) other surveys can be found regarding ontology alignment. In most

approaches, heuristics are described for identifying corresponding concepts in different ontologies, eg comparing the names or the natural language definition of two concepts and checking the closeness of two concepts in the concept hierarchy. PROMPT (Natalya & Musen, 2004) is an algorithm for ontology merging and alignment based on the identification of matching class names. A few approaches, such as RDFT (Omelayenko, 2002), use the comparison of the resources to determine a similarity between concepts, but the problem is that the structures of all data instances are heterogonous. RDFT proposes an approach to the integration of product information over the web by exploiting the data model of RDF which is based on directed label graphs. RDFT discovers a similarity between classes (concepts) based on the instance information for this class, using a machine-learning approach. Like RDFT, GLUE (Kalfoglou & Schorlemmer, 2003) is a system which employs machine learning technologies to semi-automatically create mappings between heterogeneous ontologies. An ontology is considered here as a taxonomy of concepts, and the problem of matching is reduced to “for each concept node in one taxonomy, find the most similar node in the other taxonomy”. The problem with GLUE is that the reliability of the results is related to the quantities and the degree of correction of all examples used by machine learning. S-Match Semantic Matching

(Giunchiglia, Shvaiko, & Yatskevich, 2004) is an approach to matching classification

hierarchies. Semantic matching addresses the problem of when there are two different

classification hierarchies, where each hierarchy is used to describe a set of documents, ie each

term in the classification hierarchy describes a set of documents.

2.4 Summary

This chapter described related works and concepts that have been discussed in this thesis.

Recommended systems (RSs) provide a promising approach to information filtering as they

help users to find the most appropriate items. Based on the needs of each user recommendation

system, a series of specific suggestions will be generated. It's highlighted the main

recommendation approaches and explained the principles of similarity calculation of each

approach. In addition, the drawback and advantage of each approach has been detailed.

Despite the high impact of the course recommender system and how useful it is, there are

certain significant limitations in the current researches and approaches. Approaches based

mainly on the keywords failed to address the individual user's needs in the recommendation

process. Although models use collaborative filtering, and data mining such as association rule and decision tree, there is often a lack of historical information that makes it challenging to

adopt this approach. For instance, new students who wish to use the systems do not have sufficient information about the model and therefore cannot generate any recommendations. The shortcoming of approaches that use content-based filtering is that current approaches are based only on a specific subject recommendation rather than an entire university course. Moreover, the similarity calculation in these models is based on the weighted average of features and does not take into account user interaction with the system, such as the rating value of recommendation items. Another shortcoming of the current models is that they do not provide comprehensive knowledge about the course that is most relevant to the student. For example, students need to know what future career the course will lead to and require information about this aspect, as well as the quality of the facilities of the educational institution itself that will be providing the course.

Our focus in this thesis were how to apply content based approach and collaborative based approach utilising ontology in education domain. A novel hybrid filtering approach is proposed in this research, based on both the CBF and CF methods and using ontology as a way by which to overcome the problem of information overloading which has been a key challenge when consideration is given to building an effective recommendation system. The research used ontology for data extraction and integration from multiple data sources. Data integration that is based on ontology is used in the ontology-based metadata. It utilises a combination of model-based and memory-based use of ontology in CF to provide a high-quality recommendation.

User profiling based on ontology used in the CF to overcome the new user problem. On the other hand, item-based ontology and semantic similarity are both applied in CBF to overcome the new item cold start problem.

CHAPTER 3 OPCR: ONTOLOGY-BASED PERSONALISED COURSE RECOMMENDATION FRAMEWORK

"Customers don't know what they want until we've shown them"

Steve Jobs

3.1 Introduction

Choosing a higher education course at university is not an easy task for students. A wide range of courses is offered by individual universities whose delivery mode and entry requirements all differ. Finding relevant information regarding higher education from a large number of websites is a challenging and time-consuming process. Helping students to make the correct choice from a myriad of available courses in order to meet their individual needs is a testing experience. Such abundant information means that students need to search, organise and use the resources that can enable them to match their individual goals, interests and current level of knowledge appropriately. This can be a time-consuming process since it involves accessing each platform, searching for available courses, carefully reading every course syllabus and then choosing the one that is most suitable for the student. However, simply because more course information is now provided by universities on their websites does not automatically mean that students possess the cognitive ability to evaluate each of the courses. Instead, they are confronted with a problem that is termed "*information overloading*". To counter this, artificial intelligence methods are now being applied to information retrieval systems. Recommendation systems provide a promising approach to information filtering as they help users to locate the most apposite items. Based on the requirements of each user's recommendation system, a series of specific suggestions can be generated. Thus, a personalised recommendation system can be an effective way of suggesting relevant courses to prospective students.

There are many online systems currently available that can be used to find and search for courses which use tools based on the users' prior knowledge of the courses and keyword-based queries. However, these approaches fail to address the needs of an individual user in the recommendation process. Moreover, the models use collaborative filtering and data mining, such as association rule and, as there is often a lack of historical information, this makes it challenging to adopt these approaches. For instance, new students who wish to use the systems do not have sufficient information about the model, and therefore no recommendations can be provided. On the other hand, the approaches that use content based filtering focus only on a

specific subject recommendation rather than an entire university course. Moreover, the similarity calculation in these models is based on the weighted average of features and does not take into account a user's interaction with the system, such as the rating value of the recommended items.

This chapter presents a novel approach that personalises course recommendations so that the individual needs of users are suitable matched. The proposed approach has developed a framework of an ontology-based hybrid filtering system, the ontology-based personalised course recommendation (OPCR). OPCR is a modular framework for the creation of knowledge-based recommender systems that utilise ontology as their source of knowledge. The motivation behind this approach is to overcome not only the limitations that experts impose on the automation and maintenance of such systems but also to address the cold start problem experienced by a new user of the system by using ontology matching between both the user profile and the course profiles. The proposed architecture makes use of AI to obtain the required information, where possible, in an effort to minimise the tasks required by the ontology. OPCR tackles the problem of information overloading by actually limiting the available courses the student has to examine as possible choices. In addition, OPCR uses dynamic ontology mapping between user profiles and courses profiles that lead to a reduction in the time taken to search relevant courses and improves the performance of the system.

A hybrid recommender method based on ontology is proposed in this work. The approach firstly aims to extract and integrate information from multiple sources based on ontology. The information sources are then classified into three primary sources; course information sources, student information sources and career information sources. Integrating this information using ontology will obtain optimal results.

Moreover, the second objective is to build dynamic ontology mapping between the user profiles and the item profiles that will help to reduce information overloading. In order to offer an appropriate recommendation to users, two main filtering approaches, CBF and CF, have been combined and the result is thus a combination of memory-based and model-based methods. In CF, several techniques, such as user profiling that is based on ontology, item ontology and KNN are used to overcome the information overload problem and to improve scalability and accuracy.

On the other hand, item-based ontology and semantic similarity are applied in content-based filtering to solve the new user issue and also to improve accuracy. The final objective is to put

forward a list of recommendations and ask the user to assign a rating to each recommendation.

The user then provides feedback on the recommendation list and carries out a re-ranking. User feedback has been used to evaluate the system and to improve its accuracy, as is shown in

greater detail in the evaluation chapter. This work aims to increase the accuracy and

performance of the recommender system by combining the hybrid method (CBF and CF) with

enhanced ontology.

3.2 Framework Architecture Design

OPCR has been built for situations where there is a need to identify a relevant university course

program for a particular student. Within its scope, the definition of what is relevant, as well as

the possible use of the qualification, is defined by how the course will meet the individual needs

of a student. The framework is extremely flexible since it can be adapted to any item domain

that meets the specific requirement of having 'objectively' relevant items. Although the

difference might not be immediately obvious, often the 'relevant' course is selected on a

completely subjective basis, such as when selecting which movie to watch at the weekend or

when buying books or clothes. Making a generalised suggestion in these circumstances can be

unfounded but the situation is very different when choosing a university course programme

such as BSc, MSc and so forth. During course selection, one can find specific features to

quantify, for example, the ranking or location of the university. In addition, the intended

application of a degree defines the entry requirements for courses, the course fee should match the budget of the student, and the course units/modules should be relevant. The other important factors with regard to the selection of a course are the location of the university providing the course and the type of employment that will be available following completion of the course. All these factors will impact on the decision making process when choosing the most appropriate university course. The factors will be different for each student based on their personal requirements. OPCR is therefore designed with these pertinent factors in mind.

The proposed ontology-based personalised course recommendation framework (OPCR) focuses on recommending courses to students by utilising a hybrid filtering approach that combines both content-based filtering and collaborative based filtering with ontology support. As shown in Fig.3.1, OPCR consists of four main layers. The first layer, data gathering, consists of all the information resources and the data collection module. This is used to extract useful information from multiple sources. The second layer is the database that is used to store all of the items and user information. The middle layer is the core functional part that includes the

ontological data model and the recommender engine. Each of these components will explain in detail in the following sections. The final layer is the user application layer, consisting of the user interface, which is responsible for user interaction with the framework, for searching items and for providing feedback on the recommendation list. Every layer and module in the framework both links and interacts with the others, based on the input and output of each one.

The framework comprises the following steps:

(1) Extract all the useful information from multiple sources for the system.

(2) Build profiles of the courses by extracting all the useful information regarding course features and organising that information in the system database. Consideration is given to the ontology hierarchy of the course features.

(3) Build the student profile by obtaining student information via both explicit and implicit approaches. Different user attributes have been identified which can be used to profile the student into the OPCR system as well as the user ratings of the recommended courses.

(4) Build dynamic ontology mapping in order to link the user profile and the item profile.

(5) Analyse user queries and calculate the similarity between the user profile and the course profile by employing ontology matching and cosine similarity.

(6) Use a collaborative filtering technique in order to obtain top N users that are similar to

the current user by using an ontology-based k nearest neighbour (OKNN) algorithm.

The final step suggests the recommended list of courses to the user and obtains feedback from

the user. The purpose of each of these steps is explained in the following sections. All the

modules, which have been fully developed using Open Source Free Technologies (OSFT), are

organised in a traditional client-server structure. The most novel aspect of the system is the

careful combination of different technologies, which has led to the development of an

application that uses advanced artificial intelligence techniques in an efficient way, presenting

a low execution time. These techniques are totally hidden from the users who simply interact

with a user-friendly client application that presents information on maps and lists that are very

easy to manage.

The modularity of the framework allows components to be swapped with virtually no

modifications required to other parts of the system. For instance, the web crawler functionality

could be swapped with a pre-existing database, provided that the database contained all the

required information. This also enables the independent modification and extension of each

component. For example, the Data Collection is responsible for extracting the course data from

a webpage but is so flexible that it can be used to extract any data in a different domain with

only certain adaptations to the item attributes. Moreover, Ontology Model can easily be adapted

for use in any domain. The following sections discuss each module in more detail.

3.3 Main Components

This section presented all the modules that have been developed for the framework from the

server aspect that includes the data gathering module, the ontology model and the recommender

engine as highlighted in Fig. 3.1. Each of these modules works sequentially and in correlation.

Figure 3.1 OPCR main architecture

In the following subsections, the structure of each module is explained, and the input/output

data for each is described.

3.3.1 Data Gathering

As it was decided that a content-based recommender system technique should be the primary approach for the provision of recommendations, different formats of information were required to be gathered to support this system. Fortunately, all of this information is available through sources that are publicly available. This includes websites in HTML format, such as the universities' websites for course information and recruitment websites for career information and Microsoft Excel documents that have been uploaded to the internet, such as statistical information regarding the reputation of educational institutions, for example, the NSS scores for universities. The data from both the student and course ontology is prepared and pre-processed into the correct format for the recommendation engine by the pre-processing data component. To obtain information about each course from the websites of all the universities was a time-consuming task as each university publishes its course information in a different format. Extracting precise information from various websites is always a challenging task in the domain of information engineering, so a web crawler was therefore customised that browses the web page automatically. The web crawler scrapes information from a web page and then sorts this into the system database. The reformulated queries are allocated to web crawlers and APIs that search for specific course information and jobs.

The web crawler analyses the web page based on a definition of the features of each course and then extracts the feature values. Each extracted feature value belongs to one of the features defined in this paper. Five features of the courses are used in this study: course title, course major subject, course fee, university location and the language of the course. On the other hand, the feature constructed in the user ontology is based on item ontology. The implicit information such as user feedback and the rates of the recommendations have been collected and added to the user profile for later use when it is then utilised to locate a top-rated neighbour that is similar to the target user.

The main challenge of the data gathering process was the building and customising of the web crawler to extract data from the web pages. All course and university information are available from the universities' websites. However, there is no suitable dataset available from which to generate a synthetic dataset to implement the proposed framework. Visiting the website of each

individual university and extracting information is a huge challenge and extremely time-consuming.

Moreover, the web pages of each university use a different layout, and the data is sorted in a particular format. This problem led to the use of the UCAS website in order to extract information regarding courses for universities throughout the United Kingdom. UCAS is one of the most popular higher education websites that details course information (undergraduate, postgraduate, etc.) for all UK universities. The challenge with using UCAS is that no API is provided with which to extract course information. This led to the building and customising of a data extraction API that could extract useful course information and save this in the system database to be used when needed for implementation and evaluation purposes.

Based on a similar concept, to extract job information the crawler was adapted and customised to obtain all useful information regarding jobs and save this in the system database. The Java technique was used to build the crawlers, and HTML was used to create the interface for the crawler. A new approach has been used in the data extraction module that extracts the data based on a hierarchy relationship between course information or job information. The idea is that, before extracting the information, the hierarchy relation between them needs to be defined as an example, the MSc course for computer programming will be defined as a subclass, with computer science defined as the main class. Furthermore, computer science will form part of information technology as a field of study. Extracting information based on a hierarchy

relationship will help the system to avoid over-looping when creating a query in the database.

The following subsections present further discussion regarding the data collection of course

information, job information and other relevant information that has been used to improve

recommendation quality. In Fig 3.2 main architecture of proposed crawler, the crawler consists

of several stages; it begins with construction domain ontology which it uses as a reference of

similarity between the user query and the web contents. The user query adjusts to generate

query based ontology concepts and uses Term Frequency-Inverse Document Frequency (TF-

IDF) for identifying terms for query expansion.

Firstly, we describe how information retrieval can be achieved in the ontology. For instance, if
D is the number of documents annotated by concepts from an ontology O, the document is
represented by vector d of concept weights. For each concept $x \in O$ annotating d, dx is the
importance of x in document d. It can be computed by using the TF-IDF algorithm as shown in
Eq. 3.1:

$d_x = \max^{ferq}{}_y freq_{x,d}$

${}_{x,y} \log^{|D|} n_x$

Where $freq_{x,d}$ is the number of occurrences of *x* in *d*, $max_y freq_{y,d}$ is the frequency of the most repeated instance in *d*, and nx is the number of documents annotated by *x*, then cosine similarity
between the query and the document is used as the relevance score for ranking the documents
as shown in Eq.3.2.

cosine similarity(d,q) = $\sum d_i . q_i$

$\sqrt{\sum d_i^2} . \sqrt{\sum q_i^2}$

(3.1)

(3.2)

Where d the ith term in the vector for document and q the ith term in the vector for the query.

The ontology-based query used as an input to the search engine module. The output of this
phase is a set of documents which would be used for the crawling system and furthermore

Figure 3.2 Proposed crawler main architecture

43

operate as a way by which to check all the web pages for validity (ie HTML, JSP etc.). If it is

valid, it is parsed, and the parsed content is matched with the ontology and, if the page matches, it

will be indexed otherwise, it will be discarded. Architecture of the proposed approach is illustrated

in Fig.1. The user interacts with the crawler using a simple interface designed to allow the user

query insert.

3.3.1.1 Course Crawler Module

The web crawler has two main tasks, to acquire the web pages featuring relevant products and to extract from these the useful information that is needed for the recommendation process. The

crawler was used to extract course information from the UCAS webpage. Both of the tasks of

the crawler utilise the Document Object Model (DOM) that is used to describe all the elements

of a webpage. As mentioned in the previous section, the primary source for university course

information in the United Kingdom is UCAS. UCAS details more than 80,000 courses for

different fields of study at different universities. In order to validate the framework, MSc

courses will be the target of the crawler as a case study. Most students face difficulty in finding

which master's programme is more relevant to their background or which will match the type

of career they are considering. Through the webpage, the crawler can target specific objects in

a webpage that might represent links or other relevant information. For example, the bullet

point list of a course description is represented in the DOM by a list object.

Fig.3.3 shows a sample of a course page from the UCAS website. A sub-module of the crawler,

the content extractor, adopts the same approach to identify the required information from the

product webpage. The information required in this case is the unique DOM identification

characteristic of the area that contains the required text. Four areas are extracted from each

course webpage; the university name and course topic, the course details, the entry

requirements and fees and

funding.

This approach makes both the crawler and the content extractor very flexible since they can be customised for virtually any type of university course programme following a structural pattern. The content extractor can also be expanded to identify any other important areas of information that might be required in the future such as course modules/unit, assessment method and so forth. Furthermore, both components are virtually infallible. However, these advantages come at the cost of scalability since, for each new course that needs to be scanned, a new set of DOM characteristics needs to be identified. The flexibility that is required is also the reason why generic extraction frameworks have not been used. The outcome of the web crawler presents

courses names, accompanied by a detailed table of attributes that includes names, values and headings, as well as any possible bullet point descriptions. Table 3.1 shows the outcome of the web crawler for the course. Similar data tables are created for all scanned courses

Feature	Value
Course title	Artificial Intelligence
Course qualification	MSc
Course URL	https://digital.ucas.com/courses/details?coursePrimaryId=7fea172a-efdd-4c02-9950-edf34da09124&academicYearId=2018
Course description	Artificial Intelligence (AI) forms part of many digital systems. AI is no longer seen as a special feature within software, but as an important development expected in modern systems. From word-processing applications to gaming, and from robots to the Internet of Things, AI tends to be responsible for controlling the underlying behaviour of systems. Such trends are forecast to grow further.
University name	University of Aberdeen
Field of study	Information technology
Main subject	Computer science

Major subject Artificial Intelligence

Course UK Fee £6,300

Course international fee £15,000

Course location University of Aberdeen

Figure 3.3 HTML structure of a course webpage

King's College

Aberdeen

AB24 3FX

Entry requirements Our minimum entry requirement for this programme is a 2:2 (lower

second class) UK Honours level (or an Honours degree from a non-

UK institution which is judged by the University to be of equivalent

	worth) in the area of Computing Science. Key subjects you must have covered: Java, C, C++, Algorithms problem solving and Data Structures.
Course duration	12 months
Course language	English
Course mode	Full-time

Table 3.1 Example result of the web crawler for course information

3.3.1.2 Job Crawler Module

The second web crawler was used to extract job information from recruitment webpages such as Indeed.com. The job information crawler used the same principles as the course information crawler. For each job item that has to be scanned, a new set of DOM characteristics also needs to be identified. Fig. 3.4 shows a sample of a job webpage from the Indeed website. The crawler has been customised to obtain useful information about jobs featured by Indeed.co.uk, one of the most popular recruitment sites used to search for jobs in the United Kingdom. All information that the crawler extracted about the job is based on course majors. The information regarded as useful about the jobs was a job title, job description, company name, job location, job salary and reviews of the job. Table 3.2 shows the outcome of the web crawler for the job.

Feature	**Value**
Job title	Machine Learning/Artificial Intelligence Engineer

Company's name EF

Job description Machine Learning/Artificial Intelligence Engineer. We are looking for Machine Learning/Artificial Intelligence engineers to help us build the most intelligent system. Job location London

Job salary £35,000 - £45,000 a year

Job education requirement MSc Artificial intelligence

Job review 30 reviews

Table 3.2 Example result of the Web Crawler for job information

3.3.1.3 NSS Score Data Collector

The national student survey (NSS) is one of the significant factors students have to consider when choosing a university course programme in the United Kingdom. NSS contains much statistical information regarding a university such as teaching average, student satisfaction, lab facilities and so forth. The NSS report is published every year with details of all educational institutions in the UK. The NSS score collector has extracted the NSS information and sorted this into SQL format in the system database.

3.3.1.4 University Rank Collector

One of the critical factors that can affect decisions when choosing a university course is the reputation of the university (Brown, Varley, & Pal, 2009). OPCR has extracted the rank for each university from the popular education website such as theguardian.com[1] and sorted this in the system database in SQL file format. OPCR used the university rank as one of the factors in the final scoring function for recommendations to compute the similarity between each course in the database.

3.3.1.5 Feature Extraction

Identifying different attributes is necessary for course profiling (Lee, 2011). In order to construct a course ontology, the factors that most influence a student when making a decision

Figure 3.4 HTML structure of a job webpage.

to choose a university course needed to be identified. These factors then formed the main

classes of the ontology. A survey of students at the University of Portsmouth was carried out

to discover the most important factors that had influenced their choice of university course.

The programme title, fee, location and prominence of a course were all factors that appeared

to be the most important when the students determined their choice of university for higher education (HE) study.

It was found that issues of institutional prominence maintain a fairly high profile in students'

decision-making. The overall reputation of the institution and the National Student Survey

score (NSS) of final year undergraduate students are both significant.

3.3.1.6 User Profile data collector (UPDC)

This component is responsible for collecting user profile information either explicitly or

implicitly. User information includes demographic data, academic information and career

interests as well as information regarding the user rating of recommendation items. This

information was extracted based on two approaches. The first approach was to ask the student

information when he/she registered with OPCR as shown in Fig 3.1 OPCR has a user interface,

which is web-based using HTML, that allows the user to make interactive by completing the

user profile and providing feedback for the recommendations. The registration information for

the user includes personal information such as name, age, gender and postal address together

with academic information such as educational background, the field of study, interest area,

skills and so forth. Information obtained by asking the user for this directly is referred to as

explicit. The second approach, implicit information, is based on feedback received from the user regarding the recommended item and how they rated the recommendation.

3.3.2 Ontology Model

With the growing need for more effective use of ontology in a wide range of applications, there has become an increasingly high demand for the creation of a suitable construction approach for ontology. Ontologies are constructed using two methods, i) the manual process (semi-automated) with a supervised approach; and ii) the automated process with an unsupervised approach (Zeng, Zhu, & Ding, 2009). Thus far, construction approaches for most of the applications have applied a semi-automated process. Ontologies are constructed manually using various methods such as machine learning techniques and data mining techniques that can build an ontology of high quality. However, from the perspective of performance, this can

be costly and time-consuming and to maintain and manage the created ontology can be a struggle. This, therefore, means that this approach is less feasible for a wide degree of applications.

The aggregation of ontology domain knowledge into the recommendation process is one of the solutions that can overcome the limitations of conventional recommender systems. Ontology-based (OB) recommenders are knowledge-based systems that use an ontology to represent

knowledge about the items and users in the recommendation process. In education, ontology-

based recommender systems employ ontological knowledge regarding the students and the

courses to map a student to relevant university courses which meet their individual needs.

Ontologies play an important role in knowledge representation as well as in knowledge sharing

and are reused in these systems. Previous studies have shown that aggregation of ontology

domain knowledge regarding the learner and the learning resources improves the accuracy and

quality of recommendations as well as alleviating other drawbacks associated with

conventional recommendation techniques such as information overloading and cold-start

problems (Adomavicius and Tuzhilin 2005; Zhao et al. 2015b).

The automated approach can construct ontology without the input of user mediation. However,

several problems exist in this unsupervised method such as inconsistency and, in particular, the

appropriate handling of missing information of concepts and their relationships deserve

mention. In addition, the factor of inconsistency plays a crucial role in diminishing the

effectiveness of this method. Therefore, the construction of high quality and efficient ontology

remains an open research problem. To overcome the limitations in the available construction

process, an automated ontology approach is proposed.

The ontology model includes the construction of dynamic ontologies for the user and the items

that map these ontologies in order to gain a comprehensive knowledge for the

recommendations. After building the ontologies and mapping them, the result will be used as

an input in the recommender engine. In the proposed approach, ontologies are used to model

knowledge regarding the course content (the course profile), knowledge about the user (the

student profile) and domain knowledge (the taxonomy of the domain being learned). Within

the domain of knowledge aspect, the term ontology refers to both the formal and explicit

descriptions of the domain concepts (Ibrahim et al., 2017). These are frequently considered to

be a set of entities, relations, functions, instances and axioms. By enabling the users or contents

to share a common understanding of the knowledge structure, ontologies provide applications

with the ability to interpret the context of the student profiles and the course content features

based on their semantics.

In addition, the hierarchical structure of the ontologies allows developers to reuse the domain

ontologies (for example, in computer science and programming language) (Singto &

Mingkhwan, 2013) in order to describe the learning fields and to build a practical model without

the need to begin again from scratch. The present work has constructed three ontologies. Firstly, the course ontology, secondly the student ontology and thirdly, the job ontology. The protégé tool has been used to evaluate the ontologies with the hierarchical mapping between the ontology classes that are used to compute the similarity between them. Knowledge, represented by the ontologies, has been combined into one single ontology. The ontology model thus created significantly helps to reduce information overloading.

In order to understand how do find semantic similarity between the items, two important modules need to be explained in this section; firstly, the automatic dynamic ontology construction module (including item ontology and user ontology) and the ontology mapping module for which more details will be provided in chapter 4.

3.3.2.1 Dynamic ontology construction

The difference between static and dynamic ontology is that the dynamic depends on certain parameters changing that can be considered globally to be situations. Static and dynamic ontologies are suitable examples of the static and dynamic concepts of classical physics (Middleton et al., 2009). There are generally several ways to make a given static ontology become a dynamic one; it simply depends on what is to be defined as being changing objects.

However, ontologies developed by static approaches consist of terms that are limited in their knowledge base due to a lack of updating. A dynamic ontology-based model is proposed to classify the extracted terms and to build a knowledge base for a specific domain. It is a challenge to obtain a well classified corpus. Even if a corpus is available, it may be classified incorrectly due to fewer terms being classified because of the limited and static nature of the classifiers. To overcome this, the use of an ontology-based model is proposed in order to classify the terms and prepare the knowledge base.

Ontology is a data model that characterises knowledge about a set of classes or concepts and the relationships between them (Antoniou & Van Harmelen, 2008). The classes define the types

of attributes or properties that are common to individual objects within the class. The following modules explain our proposed dynamic ontology model: Document Analysis, Ontology Construction as shown in Fig.3.5.

Figure 3.5 Dynamic Ontology Construction

There are many existing methods of constructing ontologies available. In the present work, the “Ontology Development 101”' approach developed by Natalya Noy and Deborah McGuinness (X. Zhou et al., 2004) is followed. The language used to write the ontology is OWL 2 Web Ontology Language (Kadima & Malek, 2010) and the protégé tool Version 5.2 (Jambhulkar & Karale, 2017) has been used to build the model. In order to construct this ontology, the following steps have been considered:

1. Determine the domain and scope of the ontology

   In this thesis, higher education has been determined as the domain, and master's courses in Computing and Business Management have been determined as the scope of the ontology.

2. Take into account reusing existing ontology

In education, many ontologies were found that model this aspect of the domain.

However, no ontology was found that could be reused to serve our intended purpose.

Despite this, current ontologies have been used as a guideline to model the common

concepts of the new ontology.

3. Enumerate the domain terms

The ontology is defined as a taxonomy that helps to describe different aspects of the

domain, such as the student, course and career. Some concepts are further divided into

subclasses that would improve the classification of the instances of these classes.

4. Determine the classes and the class hierarchy

The classes are defined as a group of individuals or instances that represent a class

where all of the members share the same concepts. When the classes are ordered

hierarchically, this is termed a taxonomy. Inference engines use hierarchies to denote

inheritance relationships. Classes are defined by following the combination

development process, which is a combination of both bottom-to-top and top-to-bottom

approaches. When this approach is followed, the important terms are first defined, and

then generalisation and specialisation takes place.

5. Define the relationships between classes

   The relationship that exists between class members in an ontology is termed the properties. There are two types of properties: object and data properties. Object properties represent the binary relations that exist between members of the classes, such as the relationship between a student and the courses. Here, a property called *HasSelected* has been defined which is used to represent this relationship. Data properties link an individual to a data literal, such as a student's ID, and it was found that, by analysing users belonging to a particular profile, they have a similar interest in course ontology. Thus attributes such as *offereCourse*, *HasCareer,* etc. can help to decide initial recommendations for a user according to his/her profile. In addition, this work on the recommendation of courses has focused mainly based on CBF and the attributes in the course vector such as course title, main subject of course and location.

   The user nodes in the user profile ontology are linked to course attributes in the course ontology using a *hasFeildOfStudy*, *HasLocation* relations. The course ontology is linked with job ontology using a *LeadTo* relation.

3.3.2.2 Ontology Mapping Module

After constructing all the local ontologies in the framework, it is essential to discover the links between these ontologies. Mapping ontologies will help to obtain a comprehensive knowledge to answer queries asked by users. However, as each domain uses its own set of ontologies, an interoperability issue arises when exchanging information among these domains. To overcome this interoperability issue, an ontology mapping module (OMM) proposed a new mapping algorithm to establish a mapping between ontologies in the framework. The ontology mapping algorithm focuses mainly on improving the efficiency and accuracy of the mapping process whilst also addressing other issues regarding the declarative and expressivity of the mapping

representation. The mapping between the two ontologies is performed at two levels. The first level maps the concepts between them while the second level matches the properties for a given set of mapped concepts. The mapping process has been discussed in more details in chapter 4.

The mapping process includes severl steps, starting with two ontologies which are going to be mapped as its input. The derivation of ontology mappings takes place in search of candidate mappings. The similarity computation determines the similarity values of candidate mappings. Hypotheses are then generated using a rule base. This rule base contains a set of deductive rules which may be enriched with new rules proposed by domain experts. The “best” similarity hypothesis is selected. Each step can be repeated for multiple rounds and exchanges messages with the previous step if necessary.

In OPCR, the recommender systems utilise domain ontologies to enhance personalisation because, in CF clustering, the interests of the user are modelled more effectively and accurately through ontologies and the application of a domain-based inference method. OPCR will use an ontology method to improve accuracy and to enhance personalisation in the CF aspect of the hybrid recommendation system using ontology and content of items. Ultimately, content-based features and ontology can be considered for the improvement of personalised recommendation and accuracy in the CF aspect by combining memory-based and model-based techniques.

3.3.3 Recommendation Component

After constructing the ontology models, in this section, the recommender engine is now introduced. OPCR used a hybrid method that combined the CBF and CF filtering approaches with supporting ontology model mapping, and this is the core component of the framework. In chapter 4, how each element of the hybrid approach works is explained in detail. Furthermore, the recommendation component has the following modules:

3.3.3.1 Recommendation Engine

The Recommendation Engine (RE) is a tool that contains one or more different types of an algorithm that have been combined to recommend items to users based on user preferences. In order to provide a personalised recommendation to users, a series of stages will be implemented, and a different score will be calculated at each stage according to the weighting of that stage. The details of each stage will be discussed in different scenarios in chapter 4.

The OPCR recommends courses to students according to their profiles as well as in relation to similar users with whom they share a comparable academic background and who, when they used the system, had highly rated the courses. The user is required to provide basic personal information and academic information such as field of study, main subject of the course, interest area and the type of skills that he/she has. In the case of recommending a course at UK universities, there are certain important factors that need to be considered by the user such as the range of the course fee, the range of the university ranking and also the range of the degree of NSS score. After all this information has been provided by the user, the recommendation algorithms will then be implemented according to the CBF and CF filtering approach in order

to provide the user with relevant recommendations. However, in a situation where the user

wishes to search for a course using OPCR without first providing user profile information, the

find requester will search for the course in the database according to a similarity between the

given keywords and the course titles using TF-IFD technique. The keywords relevancy is

calculated by multiplying term frequency with the inverse document frequency.

3.4 User Interface Component

The User Interface (UI) component facilitates the interaction between users and the system of

a service provider. The users' interaction is performed through user registration, user login,

user on-demand requests and corresponding recommendations and user feedback. HTML was

used for the system interface. The UI component consists of two main modules: user module

and administrator module. The user module contains several activities such as registration onto

the system, login to the system, conducting a general course search, adding ontology concepts,

undertaking a personal search based on ontology and displaying a list of recommendations to

obtain the user rating for the recommendation items. Through the UI component, a user can perform the following actions:

- Login to the system every time
- Create a user profile through the registration page
- Search courses and create a personalised recommendation list
- Add, update and delete items from the user profile and rate recommendation list items
- Provide feedback regarding the received recommendation

The administrator module has many activities that make the framework more flexible and able

to adjust, delete, add and manage ontology models classes without the need to access the back-

end of the framework. The administrator module also allows adjustment to the weights of the

algorithms in the framework by using an algorithm weight adjustment page. The module

interactive interface between the system and active users. It will be generated based on the

respective user's information including the user's demographic information, personal interests

and requirements within a given domain of education. Making a framework that enables a user

to modify their personal information dynamically can lead to a reduction in the time taken to

create a new ontology.

3.5 Summary

This chapter has presented the details of the proposed ontology based personalised course recommendation framework (OPCR). OPCR is made up of different components that are important for the framework to work effectively. The proposed framework supports the development of personalised course recommendation systems. OPRCourse is designed to provide course recommendations to students based on the ontology concepts similarity between the course profile and the student profile. It is takes into account the personal information and academic information of the users in the recommendation process. Furthermore, an ontology mapping module is proposed in the OPCR framework to integrate information from multiple sources and to map this into a single unified module in order to obtain a comprehensive knowledge with which to answer users' queries. The aggregation of ontology domain knowledge into the recommendation process is one of the solutions that can overcome the limitations of conventional recommender systems. Ontology-based (OB) recommenders systems are knowledge-based and use ontology to represent knowledge about the items and the users in the recommendation process. In addition, user profiling that is based on ontology, item ontology and the semantic similarity between two ontologies is used to overcome the new user problem

Moreover, OPCR is a novel, personalised, adaptive dynamic hybrid recommender framework

which supports the solution to the information overloading and cold start user problems which

pertains to the difficulty of providing high quality recommendation to new users. OPCR

supports the representation, indexing, sharing and delivery of context information and provides

modular components that are common across
applications.

CHAPTER 4 ONTLOGY MODEL AND RECOMMENDATION ENGINE ALGORITHMS

4.1 Introduction

The rapid increase of information available on the WWW in different domains has caused a

problem of information overloading. As explained in chapter 2, the education domain is one of

the domains which have been influenced by this problem. Finding the appropriate education

facility or relevant course has become a challenging task for most students. Before they enrol

on a relevant university course, a student needs to be certain that the choice most successfully

matches their individual needs. Recommender systems represent a promising approach by

which to tackle the problem of information overloading. While university courses may feature

similar course concepts within the same field of study, the various courses may lead to different

type of career paths, therefore, the ontology model has been used to solve the semantic

similarity problem.

There are many factors that influence students when they seeking to make a decision regarding

the selection of a university course as mentioned in chapter 2 and information regarding these

factors is available in different formats and from different sources. The ontology model allows

information from multiple sources to be automatically integrated in order to obtain

comprehensive knowledge with which to respond to the queries raised by users/students.

Furthermore, modelling the information at the semantic level is one of the main goals of

utilising ontologies.

This chapter is dedicated to describing in detail both the construction of the ontology model

and the recommender engine algorithms that have been used in OPCRa and how this approach

can tackle information overloading and cold start problems experienced by a new user. In the

ontology model section, a new approach is introduced to automatically generate ontologies for

both the item (course/job domain) and the user (student domain). In addition, a new mapping

algorithm is proposed to map the similar concepts of the domain ontologies to obtain a

comprehensive knowledge of recommendations items. The mapped information is used as an

input in the recommender engine. OPCRa attempts to reduce the confusion experienced by the user when attempting to retrieve what he/she wants from the massive number of items (for a certain specified domain).

The second part of this chapter will discuss in depth the algorithms which have been proposed for the recommendation engine to produce recommendations for users. A hybrid filtering approach is used in OPCRa which combines the CBF filtering approach and the CF filtering approach and utilises ontology to enhance the recommender algorithms. Using ontology in the recommender system helped to increase the accuracy of user similarity. In the following subsections, more details will be discussed regarding the ontology model and the filtering algorithms in OPCRa.

4.2 Ontology model

The aggregation of ontology domain knowledge into the recommendation process is one of the solutions that can overcome the limitations of conventional recommender systems. Ontology-based (OB) recommenders are knowledge-based systems which use an ontology to represent knowledge about the items and users in the recommendation process. In education, ontology-based recommender systems use ontology knowledge about the students and the course

resources in order to map a student to relevant university courses which meet their individual

needs.

Ontologies are used in OPCRa to model knowledge about the course content (the course

profile), job content (the job profile), knowledge about the user (the student profile) and domain

knowledge (the taxonomy of the domain being learned). Within the domain of knowledge

representation, the term ontology refers to both the formal and explicit descriptions of the

domain concepts (Ibrahim et al., 2018). These are frequently perceived as a set of entities,

relations, functions, instances and axioms. By enabling the users or contents to share a common

understanding of the knowledge structure, ontologies give applications the ability to interpret

the context of the student profiles and course content features based on their semantics. In

addition, the hierarchical structure of the ontologies allows developers to reuse the domain

ontologies, for example, in computer science and programming language (Singto &

Mingkhwan, 2013), in order to describe the learning fields and to build a practical model

without the necessity of starting from
scratch.

In this thesis, three ontologies have been constructed to validate the proposed approach. Firstly, the course ontology, secondly the student ontology and thirdly, the job ontology. Fig.4.1 shows

Figure 4.1 Data flow and main related module in ontology model

the data flow and the related modules developed to build the ontology model. The protégé tool

has been used to evaluate the ontologies with the hierarchical mapping between the ontology

classes used to compute the similarity between them. Knowledge, represented by the

ontologies, has been combined into one single ontology. The ontology model that has been

created significantly helps to reduce information overloading.

4.2.1 Ontology Construction Module

All the useful information collected from multiple sources is presented as entities, and their relations are all described by the relationship in the relational database as mentioned in chapter 3 section (3.3.1). These relationships correspond to concepts in the ontology. The information is represented in tables in the database. Three main tables have been presented in the module which are a course information table, a student information table and a job information table.

When generic domain ontologies and domain ontologies are modelled, the ontologists should select the top-level ontologies to be reused. And then the application domain ontologies are built on top of them(Fernandez-Lopez & Corcho, 2010). In ontology construction module Zaemmoruchi and Grhomari approach has been used to build reference ontologies(Zemmouch & Ghomari, 2013) . Reference ontology is generally constructed using the concepts of other ontologies as inputs to an analysis and subsequent synthesis. So, the commonly used approach

in constructing the reference ontology is to survey all concepts from all established ontologies and analyze similarities and differences, in order to arrange concepts into a coherent representation of ontology domain knowledge.

Reference ontologies are primarily designed as extensions or specializations of high-level

ontologies that take a global view of multiple domains of reality, and do so in accordance with

principles of ontology science. The same principles and set of rules have been used to transfer

the information from each table to a local ontology. According to (Lei Zhang & Li, 2011), five

rules have been identified by which to construct an ontology from a relational database as

follows:

Rule 1: Read the information from tables in the database, map the relationships directly into

ontology concepts, defined as class C_i. Map a data table into a concept. The table name acts as

the concept name. The table attributes act as the concept properties.

Rule 2: For the relationship, R_i in a relational database, Supposed $P_i = PKey\ (R_i)$, $Fi = FKey\ (R_i)$, if $F_i\ (R_i) \subseteq P_i(R_i)$, *then* the attributes of foreign keys are removed from the properties of the concept. That is, the attributes of foreign keys are not considered as the properties of the

concept to map.

Rule 3: For the relationship in the relational database, if $FK(R_i) \neq 0$, $PK\ (R_i) \neq 0$, $FK(R_i) = PK(\ R_i)$, then the table is a bridge table. Two object properties (owl: ObjectProperty) of ontology are created from this. Both properties are reciprocal properties. Their domain and range are the

two ontology concepts corresponding to the relationships which are referenced respectively.

Rule 4: For the inheritance relations R and *sub R* in a relational database, these can be mapped into ontology object properties (owl: ObjectProperty) directly. The OWL class which corresponds to the table *sub R* is declared as a subclass. The corresponding class of table R is seen as a parent class. *Sub C* is a subclass of *C*.

Rule 5: For the relationships R_i and R_j in a relational database, if $F_i(R_i)$ is the foreign key attributes of R_i, and $F_i(R_i)$ is referenced by R_j. Moreover, they cannot meet rule 3 and rule 4. Then, object property (owl: ObjectProperty), “HAS_A” is added to the foreign key attributes.

The domain and range are C_i, C_j respectively.

The ontology construction process begins by extracting information and a relational model from the database and establishing a relation metadata model. Based on the analysis of the relational database model, it transfers the relational database into the ontology model using the

rules defined above. Ontology information is thus extracted from a relational database by mapping the database data into ontology instances according to the data conversion steps from the relational database to OWL ontology. Fig.4.2 shows an example of course ontology construction from the course table.

Figure 4.2 Construct course ontology from course table

4.2.2 Course Ontology

Identifying different attributes is necessary for course profiling (Lee, 2011). In order to construct a course ontology, the factors that most influence a student when they make a decision about the choice of a university course need to be identified. These factors then form the main classes of the ontology. Students at the University of Portsmouth were surveyed to ascertain the most important factors that had influenced their choice of university course. More than 200 students participated in this survey. They were given 20 factors that influenced their decision when choosing a university course programme and were then asked to rank these factors on a scale of 1-10. The 20 factors were classified into six categories, and the scores and standard deviations for each category were computed. The results are summarised in

Table 4.1.

Factors and key constituent element	Mean
Course information(Field of study, Courses , major subjects , course structure)	7.8
Course Fee	7.5
NNS score	7.4
Prominence (institutional reputation)	6.4
Location (institutional location)	6.9
Career	7.9

Table 4.1 Factors and keys constituent elements for selecting university courses

60

The title and fees of a course programme and the location and prominence of the university were all factors that appeared to be the most important when students determined their choice of university for higher education (HE) study. The following points can be noted:

- ❖ Taking 5.5 as the midpoint on a ten-point Likert scale, three of the seven factors had a mean score that was lower than this midpoint. It can be assumed therefore that the promotion, people and prospectus elements do not have a significant influence on the choices that students make regarding where to study for their higher education.
- ❖ Among the elements included in the programme factors, both the field of study and details regarding the course information appear to exert the most considerable

influence on the / choice of university course programme by students.

❖ The factor that was uppermost in the' decision-making frameworks of the students

was the issue of fees which had the greatest impact on university choice and the type

of career that could be achieved following completion of the course.

❖ It was found that issues of institutional prominence maintain a fairly high profile in

students' decision-making. The overall reputation of the institution and the national

student survey score (NSS) of teaching students are both significant.

The course attributes are considered when extracting the course profile including the essential

information, course information, as well as information regarding fees and university rankings

and the NSS score for the university. This information is used for knowledge discovery at a

later stage of the user profiling process. In Fig. 4.3 the main classes and subclasses of the course

ontology with instances are shown.

The course profile attributes will match the user profile features through ontology mapping.

Each class of the course profile will be a map to the equivalent class in the user profile.

Ontology reference is used to identify the equivalent classes in both the course profile and the user profile. The protégé tool was used for the construction and evaluation of the ontology model. Fig.4.4 shows the graphical representation of the course ontology in the protégé environment.

Fig. 4.4 shows the concepts hierarchy in the course ontology features parsed at different levels.

Some of the features contain a number of other aspects which are related to the main feature.

For example, *course_information*, as the main feature in the course profile, takes the position

at level 1 and includes four features which are *MajorSubject*, *FieldOfStudy, Course_title* and *MainSubject.* Each of these features is assigned by weight in the scoring function according to the degree of importance of the feature. In the example of the *Course_information*, the features of *course_title* and *MajorSubject* are given high importance in order to calculate the similarity between two courses. Therefore, when the user condicts a search, the ontology hierarchy will help the system to find the user's interests by checking the concept in the user profile feature and calculating the similarity in the *course_information* feature. For example, if the feature *FieldOfstudy* is Information Technology and *MainSubject* is computer sciences in the course profile, the system will calculate the similarity only with the courses under computer sciences instead of looping in the entire database. Thus, using the ontology model leads to an improvement of the performance of the recommender system.

Figure 4.3 Course ontology structure

Figure 4.4 Graphical representation of the course ontology in the protégé environment

4.2.3 Student Ontology

Firstly, the student profile needs to be modelled prior to recommending an appropriate course.

The user profile consists of two main parts. The first part is the personal and educational

attributes of the user and the second is the user's rating of the previously recommended course.

The personal and educational attributes include the user's individual personal information as

well as education and background information such as their hometown, gender, the field of

study, main subject, major subject, interest area, technical and non-technical skills. In Fig. 4.5

and Fig. 4.6, the graphical representation of student profile ontology in the protégé environment

is shown. A student profile can be defined as:

$$U= \{ a_1,a_2,\ldots\ldots\ldots\ldots, a_n\} \quad (4.1)$$

Where U is the user/student, a_i represents the user's *i*th attribute.

If a student has obtained an offer from the system in the past and rated the courses, that student

can be further definded as:

$$U_r = \{u,r\} = \{ a_1,a_2,\ldots\ldots, a_n ,r_n\} \quad (4.2)$$

Here, U_r is the user that received a recommendation for the courses from the system and has rated the courses.

Furthermore, in order to make a satisfactory recommendation, it is important to ensure that the characteristics of the recommended activities match the interests of the user. The course ontology is created for all the courses that are to be recommended to the user/student. The system recommends several courses in the streams of arts, information technology, science, social science, management, commerce, engineering, education and law. The student obtains a recommendation for any course depending on their eligibility, ie if the student has a graduate degree, the system can recommend any post-graduate course. If the student has a postgraduate degree, then either a research course or PhD can be selected, depending on the faculty. The proposed approach conducts an entrance test as an eligibility criterion for admission onto undergraduate and postgraduate engineering courses.

◦

Figure 4. 5 Graphical Representation of the Student Ontology

In the proposed system, there are three ontologies: course ontology, student profile ontology

and jobs ontology. There are three aspects of the local ontology construction process. These are

unstructured text documents from structured relational data sources and semi-structured data

sources. Unstructured text documents include four processes: data pre-processing, concept clustering, context extraction and local ontology construction. For more information regarding

local ontology construction from the unstructured text, see (Ibrahim et al., 2018) .

4.2.4 Job Ontology The intended future career of a student is an essential factor that can influence decision making

when selecting a university course (Farzan & Brusilovsky, 2006). Constructing a job ontology

is vital if a student is to understand the attributes of the planned employment role and career path. This information is extracted from a job website, such as Indeed.com. Job attributes include such information as job title, job description, job salary, job location and the required educational qualifications, as shown in Fig. 4.7. A graphical representation of the job ontology in protégé environment is shown in Fig. 4.8

Figure 4.6 Student ontology structure

Figure 4.7 Job ontology structure

Figure 4. 8 Graphical representation of the job ontology