2015-06-16

Developing a collaborative MOOC learning environment utilizing video sharing with discussion summarization as added-value

Al-Mousa, Mohannad Adel

Developing a Collaborative MOOC Learning Environment Utilizing Video Sharing with Discussion Summarization as Added-Value

By

Mohannad Adel Al-Mousa

A thesis submitted in partial fulfillment of the
requirements for the degree of
Master of Science in
Computer Science

Supervisor: Dr. Jinan Fiaidhi

Department of Computer Science
Lakehead University October
2014

Abstract

With the fast-growing Massive Open Online Courses (MOOC) community and the increase in the number of Learning Management Systems (LMSs) available online, the amount of shared information is massive. Current LMS - in particular MOOC providers - offer many advanced content delivery techniques: interactive video, active retrieval practices, and quizzes to enhance the pedagogical process. The main knowledge creation assets within MOOCs are encapsulated in other tools such as discussion forums, blogs, and wikis. Although these tools exist as separate entities within the platform, they still follow traditional techniques. We believe these tools need to be fully integrated to the main content and encourage spontaneous collaboration. From my experience with some MOOCs, the amount of collaboration and information-sharing is still overwhelming due to the massive number of participants and the limited range of collaborative tools. However, most of the shared information could be redundant or irrelevant. This information must be processed in order to provide the most concise knowledge. Therefore, we need to summarize this information from the discussions, blogs, and wikis and include the most relevant data in the course content. This thesis addresses this shortcoming by suggesting a new system with two primary components to accomplish this task. In the first component, we link the discussion tools to the main course content. Then, in the second component, we apply Natural Language Processing (NLP) techniques to present a summary of all shared content. We use techniques such as Term Frequency - Inverse Document Frequency (TF-IDF), stemming algorithm, Vector Space Model (VSM), and Cosine Similarity to rank the sentences. We then tune the TF-IDF values and boost the sentence ranks using the main content by delegating the first component's features. The next step involves choosing the most relevant sentence to build our summary. Finally, we evaluate our result using Recall-Oriented Understudy for Gisting Evaluation (ROUGE) system, which compares our automated summary to human extracted summaries. These results demonstrate that we can achieve high improvement summary compared to the baseline and other similar techniques.

Acknowledgments

My sincere appreciation is extended to my supervisor, Dr. Fiaidhi, Jinan for her direct supervision, guidance and leadership in this interesting research area throughout the years of Masters and the preparation of this thesis.

I would also like to extend my appreciation to the faculty members of the Computer Science department for all their help and support.

I would also like to thank Dr. Khoury, Richard, for sharing his precious time and priceless comments and continuous feedbacks on the NLP topic.

Above all, I give the utmost gratefulness and thanks to the Almighty Allah (God) for the patience, endurance, and determination He has granted me for the completion of this academic research.

Dedication

I dedicate my dissertation work to my father and mother (Mr. Adel Al-Mousa and Mrs. Weded Atarah), who have planted the seed in me to pursue this work, and encouraged me all these years.

I dedicate my dissertation work to my brothers (Amer, Ahmad, and Abdallah) and lovely sister (Dina), who have always been by my side with their unconditional support.

I dedicate my dissertation work to my Mosque friends, and to all other friends who have been a great support during my research.

But most importantly, I dedicate my dissertation work to my beloved wife (Fatima), who has been standing by my side throughout all the ups and downs with strength, enthusiasm and support. Also, to the apples of my eye, my lovely daughters Jenna and Abrar, for their continual supporting prayers, and for understanding when I couldn't be there to play with them.

Table of Contents

Abstract II

Acknowledgments III

Dedication IV

Table of Contents V

List of Tables VIII

List of Figures IX

List of Abbreviations XI

Chapter 1 Introduction 1

1.1 Overview 1

1.2 What is MOOC? 2

1.3 MOOC and the LMSs 4

1.4 MOOC General Architecture and Platforms 6

1.4.1 New MOOC Platforms 7

1.4.2 Existing LMS Platforms 7

1.5 MOOC Techniques 8

1.5.1 Design Techniques 8

1.5.2 Development Techniques 9

1.5.3 Deployment Techniques 10

1.6 MOOC Tools/Technologies 10

1.6.1 Content Management Tools 11

1.6.2 Social Media Tools 11

1.6.3 Discussion and Collaborative Tools 11

1.6.4 File Management Tools 12

1.6.5 Notification Tools 12

1.6.6 Assessment Tools 13

1.6.7 Analytical Tools 13

1.7 Natural Language Processing....13
1.7.1 Information Retrieval (IR) and NLP....14
1.7.2 Microtext....15
1.7.3 Tasks in NLP....15
1.7.4 The Summarization Problem....16
1.8 Conclusion:....17
1.8.1 Benefits....17
1.8.2 Motivation:....17
1.8.3 Research Goals:....19
Chapter 2 Frameworks for Building MOOCs....20
2.1 Comparing MOOC Frameworks....20
2.1.1 Moodle for MOOC Framework....21
2.1.2 OpenMOOC Framework....24
2.1.3 Google CourseBuilder (GCB) Framework....25
2.2 MOOC Development Trends....28
2.2.1 Interactive Videos:....29
2.2.2 Social Media Learning:....30
2.2.3 Crowdsourcing:....30
2.2.4 Data Analysis and Processing:....31
2.2.5 NLP Support....31
2.3 Proposed MOOC Forum-Media Context Summarization System Architecture....34
Chapter 3 Implementation of the MFMCS System....36
3.1 An Overview....36
3.1.1 Methodology....37
3.2 The MFMCS Architecture....38
3.2.1 The MFMCS Functionalities....39
3.2.2 JSMPW Architecture (First Component)....40

3.2.3 Summarization Application Architecture (Second Component) 48
3.2.4 Database Layer 70
Chapter 4 Evaluating the MOOC Video Summarization Capability 74
4.1 Experimentation Setup 74
4.2 Evaluation Criteria 74
4.3 Evaluation Results 78
Chapter 5 Conclusion and future work 86
5.1 Conclusion 86
5.2 Future Work 87
Appendices 88
Appendix I 88
Evaluation Environment 88
Appendix II 92
Porter Stemming Algorithm Diagram 92
Appendix III 93
Stopwords List 93
Appendix IV 95
Partial Closed Caption of the YouTube Video 95
Appendix V 96
Sample Discussion Dataset and Associated Summaries 96
Appendix VI 104
MFMCS System setup 104
Appendix VII 107
MFMCS System code 107
Bibliography 108

List of Tables

Table 1.1: Comparison between cMOOC & xMOOC 4
Table 2.1: MOOC platforms comparison summary 28
Table 3.1: Document/Discussion (10183) sample 56
Table 3.2: Document/Discussion (10183) segmented sentences 59
Table 3.3: Document/Discussion (10183) stemmed tokens for each sentence 59
Table 3.4: Cosine similarity of document (10183) 67
Table 3.5: Comparison between initial score & boosted score 68
Table 3.6: Extracted sentences as summary of document/discussion (10183) 69
Table 4.1: 2 by 2 Sentence contingence table 77
Table 0.1: Long English Stopwords list 93
Table 0.2: Discussion Forum module files changes 105

List of Figures

Figure 1.1: Traditional LMS 5
Figure 1.2: MOOC platform LMS 6
Figure 1.3: Knowledge creation cycle 14
Figure 2.1: Typical university system architecture 22
Figure 2.2: Moodle system architecture 23
Figure 2.3: OpenMOOC Architecture 25
Figure 2.4: Architecture of UniMOOC (Peco & Luján-Mora, 2013). 27
Figure 2.5: LexRank algorithm (computing centroid score) (Erkan & Radev, 2004) 33
Figure 2.6: High level architecture of the MFMCS system 35
Figure 3.1: High level MFMCS system architecture 39
Figure 3.2: MFMCS system use case diagram 40
Figure 3.3: JSMPW GUI 41
Figure 3.4: Content page load sequence diagram 43
Figure 3.5: New discussion sequence diagram 44
Figure 3.6: View & reply discussion sequence diagram 45
Figure 3.7: JSMPW (client side layer) architecture 46
Figure 3.8: I&K wrapper tab-click pseudo code 47
Figure 3.9: View discussion summary sequence diagram 49
Figure 3.10: Search discussions & search suggestions sequence diagram 50
Figure 3.11: Search suggestions example 51
Figure 3.12: Search result example 52
Figure 3.13: Summarization application components architecture on the server side layer 53
Figure 3.14: Search result controller, ForumPosts class, and API call pseudo code 54
Figure 3.15: MFMCS summarization algorithm 57
Figure 3.16: Discussions summarization activity diagram 58
Figure 3.17: Tokenization function code snippet 60
Figure 3.18: NLP parser class diagram 60
Figure 3.19: Calculate IDF function pseudo code 63
Figure 3.20: Estimated size of Google's index (Kunder, 2014) 63
Figure 3.21: Compute TF-IDF pseudo code 64
Figure 3.22: Penalize TF-IDF pseudo code 65

Figure 3.23: Cosine similarity illustration. sim(d1,d2) = cos θ (Manning, Raghavan, & Schütze, 2009)....66
Figure 3.24: Boost sentence score pseudo code....68
Figure 3.25: MFMCS annotation table schema in XML format for Moodle installation....70
Figure 3.26: JSMPW & Summarization application DB tables....71
Figure 3.27: Code snippet of the annotation class model....72
Figure 3.28: LINQ to Entity code snippet sample....72
Figure 4.1: Configuration settings....79
Figure 4.2: ROUGE measures for both MFMCS enhanced and basic settings....80
Figure 4.3: ROUGE metrics for the best MFMCS configuration summary....82
Figure 4.4: Recall measure for the best MFMCS configuration summary using different ROUGE evaluation configurations....83
Figure 4.5: Comparing the MFMCS enhanced summary with 7 other summarization systems using different ROUGE metrics....85
Figure 0.1: The MVC architecture....90
Figure 0.2: Porter stemming algorithm flowchart....92
Figure 0.3: JSMPW files sctructure....104

List of Abbreviations

Abbreviation	Meaning	1st Page #
AI	: Artificial Intelligence	16
API	: Application Programming Interfaces	23
AWS	: Amazon Web Services	21
CC	: Closed Caption	47
cMOOC	: Connectivist MOOC	3
CMS	: Content Management Systems	4
DOM	: Document Object Model	43
DUC	: Document Understanding Conference	75
FN	: False Negative	77
FP	: False Positive	77
GCB	: Google Course Builder	25
GUI	: Graphical User Interface	40
I&K	: Information and Knowledge	37
IR	: Information Retrieval	14
JSMPW	: JavaScript Media Player Wrapper	36
JSON	: JavaScript Object Notation	37
LCS	: Longest Common Subsequence	75
LINQ	: Language-Integrated Query	37
LMS	: Learning Management System	2
LUMM	: Lakehead University Moodle MOOC	38
MFMCS	: MOOC Forums-Media Context Summarization	36
MIT	: Massachusetts Institute of Technology	1
MOOC	: Massive Online Open Courses	1
Moodle	: Modular Object-Oriented Dynamic Learning Environment	21
MSVS	: Microsoft Visual Studio	37
MT	: Machine Translation	14
MVC	: Model View Controller	7
NER	: Named Entity Recognition	14
NLG	: Natural Language Generate	16
NLP	: Natural Language Processing	13
OER	: Open Educational Resources	17
OOP	: Object Oriented Programming	39
OTS	: Open Text Summarizer	31
PLATO	: Programmed Logic for Automatic Teaching Operations	1
POS	: Part-Of-Speech	14
QA	: Question Answering	14
ROUGE	: Recall-Oriented Understudy for Gisting Evaluation	31
SaaS	: Software as a Service	20
TF-IDF	: Term Frequency - Inverse Document Frequency	33
TN	: True Negative	77
TP	: True Positive	77
VLE	: Virtual Learning Environment	8
VSM	: Vector Space Model	66
XML	: Extensible Markup Language	37
xMOOC	: MOOCs that are not based on Connectivism	3

Chapter 1
Introduction

1.1 Overview

Over the past decade, the idea of remote education in the form of correspondence study was received by students with high response rates. Between 1893 and 1899, the average increase of new correspondence students was nearly 100% from the previous year, and beyond 1899, the number of enrolled students was higher than 100,000 annually (Clark, 1906). Throughout the following years, as technology evolved, the tools for correspondence study evolved as well. For example, colleges and universities began broadcasting educational lectures via their own radio stations in 1922, which was defined as "Wireless College", at institutions such as Tufts College, Curry College, University of Iowa, and many others in the United States and other countries (Tufts College to Give Radio Lecture Course., 1922). Another method of course material delivery was by mail, whereby the school would mail out the course material to the registered students and instruct them to submit their work the same way. Soon after the digital age started, courses became available via the World Wide Web, by which some or all course material were accessible by students online, which saved institutions time and course material delivery costs. In the early 1960's, the University of Illinois initiated Programmed Logic for Automatic Teaching Operations (PLATO), which was one of the first public online systems. The PLATO system was designed for a small scale until the mid-1970s, when it expanded with the capability of supporting up to one thousand users (Woolley, 1994). Then, the concept of e-learning emerged, where schools started utilizing online-based software to deliver course content and initiate a small online community from the students in the classroom. In 2001, Massachusetts Institute of Technology (MIT) aimed to publish course materials for the public on the World Wide Web. Soon thereafter, Open University in UK followed the same footsteps with the OpenLearn, mimicked by a few other large institutions (Liyanagunawardena, Adams, & Williams, 2013). A new term called Massive Online Open Courses (MOOCs) was introduced in 2008 to describe the new era of online learning. The term was introduced by Dave Cormier to describe the "Connectivism and Connective Knowledge" course, which was designed and led by George Siemens and Stephen Downes, considered the founders of MOOCs. The course, also known as CCK08 (Mackness, Mak, & Williams, 2010), was offered for the first time in Canada by the University of Manitoba in the fall of 2008. The course had approximately 2200 registered

students, of which only twenty-four were accredited, having paid applicable fees and completed the university's proctored exam. At the completion of that course, MOOCs increased dramatically and many universities, institutions, and even non-profit organizations began opening their courses to the public on MOOC platforms. The fall of 2012 witnessed an unusual and unprecedented number of registered students for MOOCs in the history of e-learning. As edX, the non-profit joint organization from Harvard and MIT boasted 370,000 registered students (PAPPANO, 2012). Today, there are over 240 MOOC providers, of which over 30 are joint efforts with some of the world's most renowned universities. Therefore, due to that massive growth, the year of 2012 was declared "the year of MOOC" as was described in The New York Times on November 4, 2012 by Laura Pappano (PAPPANO, 2012).

1.2 What is MOOC?

The fear of expensive education, limited seats, or even not being accepted to a certain program or well-known school began to unveil with the boom of new era of eLearning called MOOC. So what is MOOC? And why did it become so attractive to students, teachers, profit, and non-profit organizations? Even some of the most well-known post-secondary schools are interested in this new concept! What makes it so special and different from other eLearning methods or techniques? Is it a new Learning Management System (LMS)? Or even beyond?

Massive Open Online Courses (MOOC) is a new design of online courses used to enhance the quality of learning and the level of knowledge, available to an unlimited number of interested students throughout the world.

As the name suggests, MOOC is *Massive*: there is no limit of the number of students registering for the course. *Open*: there are no restrictions on students joining the course by any means; not by geographical area, time zone, economical status, and most of the time, not even knowledge level, as most MOOCs do not require prerequisite courses. Furthermore, it is open in the sense that the participation is in public; everyone has access to other enrolled students' contributions and shared knowledge (Downes, The LMS and the MOOC, 2012). The only restriction is included in the next term, being *Online*: the course is provided over the internet, therefore students must have an internet connection to be able to participate. The course materials – lecture notes, assignment, self-assessments, and exams – are usually made available online after the lecture time is over, and most of the time it would be available as a self-read course, even once the course is completed. Finally, the courses have most of the characteristics

of other regular courses, such as registration, a series of meetings between a start and end date with a preset start and end time, an instructor, course content, assignments, quizzes, tests, exams where applicable, and most importantly, due dates for the assessments. Finally, students receive a certificate of course completion, which is usually obtained from the instructor, not the university.

Like other courses held in a conventional classroom, instructors use teaching aids to enhance the learning process, and deliver the course content to the students in the best possible way in order for the student to follow and comprehend. Similarly, MOOC also utilizes teaching aids. In a traditional classroom, teaching aids have been advancing as technology advances. MOOC, on the other hand, along with all available online aid techniques – such as video recording, blogs, discussion forums, and others as we will refer to later in this chapter – integrates the existing social networks and active interaction between the course content and the student to enhance connectivity and collaboration between participants (McAuley, Stewart, Siemens, & Cormier, 2010). The integration of social media increases the comfort level of new users to use the MOOC system as it wouldn't be much different from their daily-used social tool, whether it is Facebook, Twitter, Wiki, Blog or other social media.

The initial MOOC developed by Downes and Siemens with CCK08 was based on the Connectivism theory. Learners had the autonomy, connectivity, creativity and the social network to generate knowledge (Siemens, 2012) (Mackness, Mak, & Williams, 2010). This MOOC was noted as Connectivist MOOC (cMOOC). Soon after, another MOOC ideology surfaced with persevering financial support from large institutions and corporations. This type of MOOC was referred to as xMOOC.

Table 1.1 demonstrates the main differences between cMOOC and xMOOC as highlighted in (Siemens, 2012) and (Panchenko, 2013).

Table 1.1: Comparison between cMOOC & xMOOC

Criteria	cMOOC	xMOOC
Focus	Knowledge creation and generation	Knowledge duplication
Teaching approach	More of creativity approach from learners	More traditional approach directed by instructor
Knowledge Organization	Non-structured information; distributed knowledge	Structured course content
Initiation	Individual initiatives	University-supported projects
Platform	Not controlled	Controlled
Designed by	Volunteers	Companies/organizations
Financial support	Not sponsored	Widely sponsored by large institutions and corporations

1.3 MOOC and the LMSs

LMSs are another variation of Content Management Systems (CMS) that are used broadly for higher education. LMSs provide higher control over learners' workflow than CMSs, which "may facilitate multiple content-developers within the same course," (Hamuy & Galaz, 2010). They are usually the primary communication tool for correspondence education, yet they are very supportive to face-to-face classrooms. Simonson defined LMS as a system to manage educational content for learners, that assists instructors with the administration of the content, and often tracks learners' contribution and progress (Simonson, 2008). In comparison to MOOC platforms, traditional LMSs are usually designed to manage course content, to communicate with learners, and to track their progress according to pre-defined curricula and objectives, whereas a MOOC environment seems to pay less attention to managing the environment, but focuses on the knowledge that is created and shared between participants as the course progresses (Downes, 2012).

A MOOC platform or MOOC environment is not a new LMS. However, it is a new methodology for online learning that incorporates many tools to assist knowledge creation and

sharing with openness in mind. Therefore, existing LMS companies started extending their platforms to incorporate MOOCs in their LMS or creating MOOC platforms under a different name. Some of these names include MOODL[1], CourseSites by Blackboard[2], and Canvas by Instructure[3]. In addition, new platforms were created as a MOOC platform such as Coursera[4], Udacity[5], and edX[6]. Most of these are supported by highly-ranked post-secondary institutions such as MIT, Stanford, UC Berkeley, Harvard and large corporations including Google.

To visualize some of the key differences between traditional and MOOC platforms, we provided the following two figures (Figure 1.1 & Figure 1.2). Figure 1.1 highlights the main components of traditional LMS platform and demonstrates the linkage between its entities. Figure 1.2 illustrates the MOOC platform including most of its components, demonstrating the linkage between them.

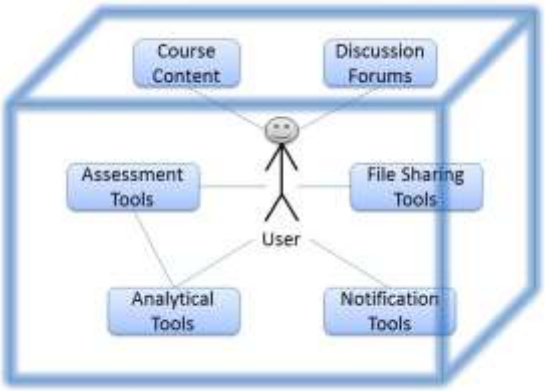

Figure 1.1: Traditional LMS

Traditional LMSs are usually proprietary-enclosed solutions. Their main components are linked to the user, with only a slight connection with the other components. Therefore, the user is the only connection between information, whether within the course content, discussion forum, assessments, or shared files. The entire system is restricted to one educational entity with no connection to the outside world. Also, the tools have become very classical, and in most cases do not attract learners.

[1] http://learn.moodle.net/
[2] https://www.coursesites.com
[3] http://www.instructure.com/
[4] https://www.coursera.org/
[5] https://www.udacity.com/
[6] https://www.edx.org/

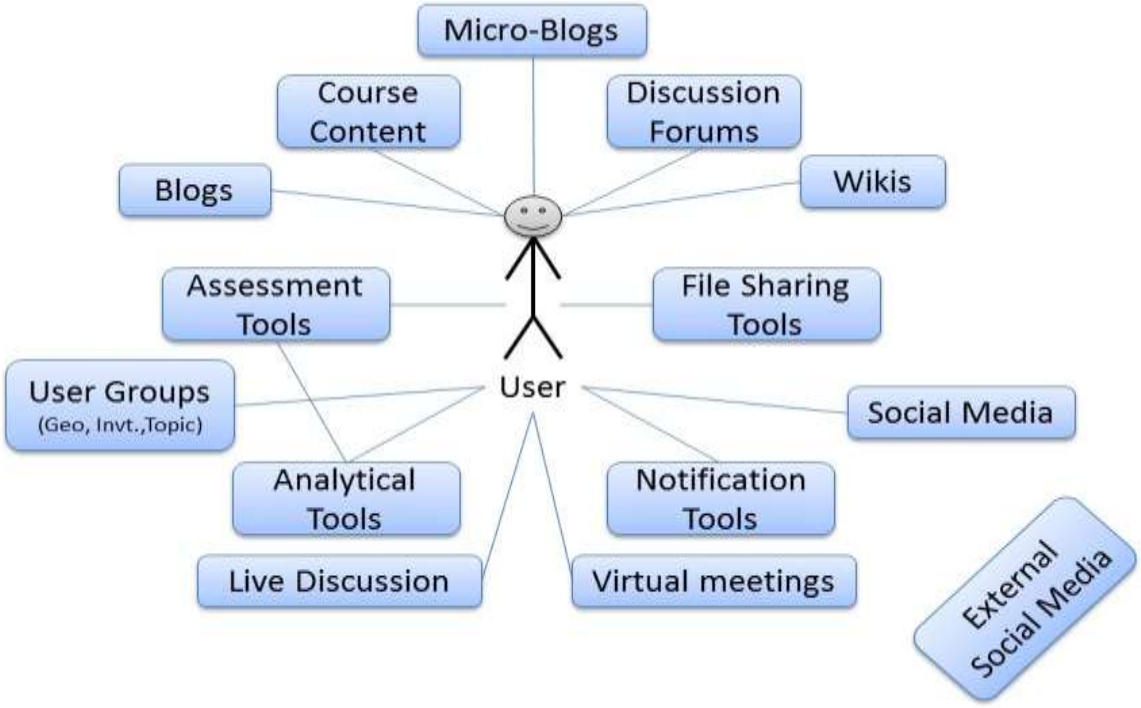

Figure 1.2: MOOC platform LMS

MOOC platforms, on the other hand, contain a larger variety of tools that provide learners with their learning method of choice. As Downes wrote when describing the difference between a MOOC course and a traditional course, "MOOC is completely voluntary" - voluntary in the participation choice, as well as in the method of participation (Downes, 2013). Although the variety of tools is much larger in this platform, there is little connection with each other. Yet, the advantage lies in the connection to the outside using the external social media, as well as the fact that it is an open system, usually accessible on the cloud.

1.4 MOOC General Architecture and Platforms

Although MOOC is relatively new, there are already many platforms that have started to provide MOOCs. In this section, we will discuss some of the key MOOC providers and highlight their main features. Most MOOC providers include the main features of its platform as shown in Figure 1.2. Some offer free courses, and others charge to obtain the certificate of completion if required by the student. There are also other differences, as we will describe below.

MOOC platforms are mostly one of two types; new platforms that started as a MOOC platform, or an extension and expansion to an existing LMS. Some of the new MOOC platforms include edX, Coursera, Udacity, and Canvas, which were developed using MVC (Model View

Controller) framework, which segregates between the logic, presentation, and commands. The MVC framework offers great development flexibility and continuous upgrades and enhancements to the platform. As for the second group of MOOCs, they are an extension of an existing LMS such as Moodle, CourseSites by Blackboard, and OpenCourses by Desire2Learn. These LMS developers decided to utilize their existing platforms with additional Web 2.0 and Web 3.0 tools and technologies to enhance user experiences, and to be able to keep up with the new wave of online learning. Although their MOOCs might appear as a new platform, they are fully integrated with existing features of their LMS.

1.4.1 New MOOC Platforms

- **edX:** edX is one of the major MOOC platforms known for its open source platform and its supporters of international institutes like MIT, Harvard and Berkeley. edX is aiming to expand globally in many languages to all nations. In November 2013, they collaborated with the Queen of Jordan to start a new Arabic-language MOOC portal (Agarwal, 2013). Also, edX is starting a joint effort with Google to create a new MOOC platform - MOOC.org - to assist educational institutes, teachers, and businesses to easily build and host MOOC courses. This new platform was launched in January 2014. Of course, edX was funded by the large educational institutes mentioned above.
- **Coursera:** Coursera is the fastest reaching MOOC platform with funds near $85 million, over 4,000 users, and most importantly 45% of course completion rate. Coursera places great emphasis on data analysis, which relates mainly to each participant's behaviors with respect to viewing of course material, participation, quizzes, practices, completion of assignments, posts, and sharing. The goal is to improve the platform and personalize user experience.
- **Course Builder:** Course Builder is a Google App Engine, providing the functionality to allow educators to build their MOOCs and to host them on Google's appspot.com. The benefit is that these courses are fully integrated with Google's platform and most, if not all of its features, start from email, social media, file collaboration, content sharing, Wikis, Blogs, or discussion forums. The setup is simple as Google provides a step-by-step tutorial on how to get started with you MOOC course.

1.4.2 Existing LMS Platforms

- **Moodle:** Moodle is an open source web-based application that serves as a Course Management System, which is sometimes referred to as a LMS or Virtual Learning

Environment (VLE). Moodle equips educators and LMS seekers with the base platform tools to create effective online learning websites and provides them with a flexible and customizable platform. In fact, Moodle is considered one of the most suitable platforms to accommodate MOOC courses. Moreover, Moodle is integrated with various operating systems, web services, and databases. The main programming language used is PHP, in addition to the basics of HTML, CSS and JavaScript. It is usually deployed as a client-server application, similar to most other websites. The database could be hosted either on the same server, or remotely. There are many MOOC platforms that use Moodle open source to launch their new MOOC site. Moodle launched its own MOOC platform in September 2013, targeting teachers as their main audience. The goal of the first course was "to collaboratively learn how to use Moodle to teach courses online" (Ng, 2013). After that, other Moodle sites hosted their MOOC on Moodle such as Open2study, Uneopen, and Wiziq.

1.5 MOOC Techniques

MOOC techniques may vary from one platform to another, or even in two different courses within the same platform. Most MOOC techniques are the same as those used in traditional classrooms, such as having a start and end time, due dates, quizzes, assignments, and a final exam. However, MOOC platform might handle these differently based on the tools and technologies available to MOOC. Another technique that might be difficult to apply in classical classrooms, yet to be used in MOOC, is to create a sense of community between learners by integrating social media in the communication between participants. In this section, we discuss some of the main MOOC techniques in three categories. Although each technique is listed under one category, some techniques might have an overlap with other categories.

1.5.1 Design Techniques

Design techniques are those that hold the MOOC's connectivism theory as a main driver. They also draw the core MOOC principles and values to be achieved. Here are some of these techniques:

- **Personalized curriculum:** MOOC course content is designed specifically for online use whereby the lecture is divided into smaller parts, each of which represents a coherent topic. Therefore, these topics could be taken separately based on the student's need, knowledge, and available time. Furthermore, each student can test their gained knowledge at the end of each topic, either in a small quiz, questions, or in a demo. This way, the student could focus

on the actual topic rather potentially being lost in the middle of the class while the instructor moves on to other topics.

- **Active retrieval practice:** Incorporating the retrieval of knowledge into MOOCs improves the learning process. As research suggested, retrieval of knowledge is not simply restating the facts, "the act of reconstructing knowledge itself enhances learning." (Karpicke & Blunt, 2011). Retrieval practice is not limited to online learning. However, in an online setting, all students are required to go through the same process. Take for example if the instructor asks a question during a lecture, one student might answer the question while possibly half of the students were not aware that there was a question posed in the first place. In MOOC, every student is forced to answer any question posted during the video and engage with the material (Koller, 2012).
- **One-on-one tutoring:** Peer tutoring has been proven to give the best learning result. Using this technology, MOOC can simulate, to an extent, a tutor who is customized and personalized for each student. Furthermore, if the course content itself is satisfactory for some students, they could act as tutors for other students.
- **Global community:** Having a massive number of students all over the world creates a sense of community around the MOOC course, as students have the opportunity to connect with each other by posting a question and providing an answer in an impressively short response time (Koller, 2012).

1.5.2 Development Techniques

These are techniques that implement the design, and insure that the tool used achieves the design goal. For example, the use of self-based courses and customizable tools targets the personalized curriculum technique in design. Also, imbedding short quizzes within the course material targets active retrieval practice. Below are some additional development techniques:

- **Automated grading:** The automated grading is interactive, as it provides instant feedback to students and gives them the opportunity to take the quiz again. With the exception of open-ended questions, the technology has been improving to allow automated grading in many fields like math, science, financial models, and in programing.
- **Peer grading:** When it comes to disciplines that require critical thinking, automated grading does not work. Coursera has used peer grading to solve this issue. Note that peer grading does not only solve the problem of grading a massive number of assignments in such disciplines as sociology and business, but the study conducted by Sadler and Good showed

that the grades provided by students and teachers are correlated. Furthermore, self-grading was also correlated. More importantly, peer grading improves the learning process whereby students get the chance to concur with the knowledge, and can explore the solutions from different point of view (Sadler & Good, 2006) (Koller, 2012).

- **Data collection and analysis:** The amount of data could be collected and the analysis applied could derive the learning process to a different dimension. This can be done by understanding the students' patterns and providing more interactive and personalized responses to students, which ultimately improves the pedagogical process.

1.5.3 Deployment Techniques

Deployment techniques are the delivery means for MOOC platform or MOOC courses.

- **Open-source platform:** Systems like Moodle, OpenMOOC[7], and edX are open-source-based platforms that offer MOOC courses. These systems provide the option and flexibility of either being an independent MOOC platform setup, or simply hosting a MOOC course in an existing platform. Some of the programing languages used in these systems includes PHP, Python, Ruby on Rails, and JavaScript.
- **Proprietary system:** Blackboard's CourseSite, Desire2Learn's Open Course, and Instructure's Canvas are examples of these systems. Some offer limited courses, hosting, and functionalities free of charge, but in order to get the full package, organizations are required to purchase the license.
- **Software as a Service (SaaS):** mooc.org is a newly-built platform by Google and edX to allow creating and hosting MOOC courses. Initially, Google started its own Course Builder platform which provided users with the ability to create MOOC courses, and deploy them on the Google app engine (Kolowich, 2013).

1.6 MOOC Tools/Technologies

There is an extensive number of tools that could be used in MOOC, the extent of which we might not be able to capture in this thesis, however we will point out the most widely used tools in MOOC and highlight their importance to the MOOC platform. On the other hand, most of these tools can fall into one or more of these categories:

[7] http://openmooc.org

1.6.1 Content Management Tools

Example of these tools: video, audio, presentation, and text.

These are the tools that handle the main content of the course and they are the core tools of MOOCs as they are the course artifacts or the delivery tool of the artifacts that contain the initial knowledge shared. The designer of the MOOC platform usually incorporates more than one of these tools to improve interaction. The content is initially uploaded or posted by the course instructor using one of these tools, either via video, audio, presentation, or text document. In MOOCs, the contents are relatively short in time or small in size in order to enhance the student's comprehension and gain of knowledge. Furthermore, the contents usually incorporate an assessment tool to concur the student's understanding by applying a retrieval practice of the main idea into the lecture. Although, we see the integration of the tools, content management, and assessment tools, the content does not hold all the knowledge that the students require. Based on the original MOOC, or what is now called cMOOC, knowledge is created and generated during the course as participants contribute to the course material through their discussion about the topic (Siemens, 2012). Although there is an area for discussion, contribution and knowledge sharing in MOOC – as we will see in the discussion tools – is not integrated or linked to the contents like the assessment tools, which makes discussions sometimes seem to be irrelevant information rather than concrete knowledge.

1.6.2 Social Media Tools

Example of these tools: Facebook, Twitter, Google+, Linked in, and My Space.

These are the tools that create the learning community and encourage collaboration between participants to support the pedagogical process in MOOC. They also open the learning materials to the outside world and attract interest from others, as users begin to share their MOOC activities and experiences within their own social networks.

1.6.3 Discussion and Collaborative Tools

Example of these tools: Discussion forums, Real-time desiccations, Wikis, Blogs, Micro-blogs, and Virtual study groups/meeting.

Discussion forums are the main and most common tool of communication used in LMS. Most of the time – especially in traditional LMSs - it is the only tool that allows collaboration between participants. Therefore, it has become a standard feature of LMSs. This feature also

carried its way to MOOC platforms. The main purpose of discussion forums is to create a learning community by encouraging students to share their knowledge and contribute with their own discussion, either in a form of a question, answer, or comments, similar to discussions in a classroom. However, the tool was not used for its initially-intended pedagogical purpose. Students were not enticed to participate, as neither the system nor the designers promoted such features. Rather, the instructions forced involvement, and rendered the feature an assessment tool to measure class participation (Morris & Stommel, 2013). Take for example ANGEL Learning Discussion forum, which featured automated scoring rubrics that monitor students' behaviors and evaluate them based on rules set by the instructor (ANGEL Learning, 2009).

The LMS can also integrate other discussion tools such as real time discussions, Wikis, Blogs, and Micro-blogs, all of which can facilitate engagement between participants and encourage collaboration in a way that suits each individual. The new MOOC platforms do use some of these tools to create the learning community, yet most of the time they are used in isolation of each other and of the main course content.

1.6.4 File Management Tools

Example of these tools: Google Drive, Microsoft Office 365, and Drop Box.

Shared and collaborative files are the tools referred to in this category. Shared files simplify collaboration between contributors. Some of these tools allow single access to files, while others – which are better for collaboration – allow multiple synchronous access. For example, users can access the same file and work on the same document, and are able to see their changes instantly. The best example of these tools is Google Drive, which offers the basic and most widely used collaboration files: documents, presentations, spreadsheets, and drawings, and has the option to link other existing file types from a preset list. Due to the fact that these tools are usually used for collaborative deliverables, MOOC does not need to use most of their functionalities. However, MOOC focuses on collaborative and shared knowledge, which make Wikis and Blogs the perfect tools for that purpose.

1.6.5 Notification Tools

Example of these tools: Emails, RSS feed, and Posts,

Notification tools are very important in the MOOC platform as well as in any LMS. They could be used for direct, group, or broadcast communication. However, not everyone has the

desire or the time to constantly check emails. Especially with the massive number of learners in the system, the quantity of emails could be overwhelming. Therefore, we need a comprehensive yet smart tool that allows customization and personalization of notifications. Having such a tool encourages collaboration and fast response between participants by enabling notification with the most convenient tool that fits the learner's objectives. Furthermore, these tools can be integrated with the student's daily used accounts so it does not become an additional burden. One way to incorporate personalized notification is to partner with social media sites and enable one login feature. Using one login via the different social media sites or email accounts, the MOOC course is linked to that site and can push notifications to the learners via the most convenient method to the learner.

1.6.6 Assessment Tools

These are the tools that assist instructors to create course assessments to record students' grades and progress. Most assessments need to be the automated-grading type, using close-ended questions. However when it comes to open-ended questions, the assessment must use a different schema or style (i.e. peer assessment).

1.6.7 Analytical Tools

These tools are used mainly in the background by administrators to provide important statistics about courses, usage, and users' habits. These statistics are used by designers and decision-makers to improve the platform. Coursera, for example, studies these statistics carefully and makes modifications accordingly to improve user experience.

1.7 Natural Language Processing

Imagine the amount of information shared by MOOC users on the online Discussion Forums, Blogs, Wikis, or even the Micro-Blogs. Digesting this information requires a huge effort by users. Therefore, we examined automated summary generation to provide the users with the most relevant knowledge from the discussions. For this task, we have used some of the Natural Language Processing (NLP) techniques. NLP refers to the interaction between human natural language and the computer. This includes many tasks such as understanding the language, classifying the text, language modeling, extracting, and generating text.

NLP has been around since the 50s after the publication of "Computing Machinery and Intelligence" by Alan Turning, followed by a demonstration of automatic machine translation in

1954 (Hutchins, 2005). NLP tackles many problems, some of which have mostly been solved like spam detection, Part-Of-Speech (POS) tagging, and Named Entity Recognition (NER). For some other problems, there has been good progress done so far. Examples of these applications include Sentiment Analysis and Machine Translation (MT). Finally, there still exist other applications that are challenging to use, such as Question Answering (QA), Summarization, and Dialog Understanding.

1.7.1 Information Retrieval (IR) and NLP

Information Retrieval (IR) is the process of undertaking a large, or it could be huge, collection of script, document, or human language for the purpose of searching or obtaining useful information, which can then be presented as knowledge. The life cycle of the data and knowledge creation starts at the repository, whereby all data is saved in raw form with a massive number of records that are not used until they leave the repository. These records are then retrieved with specific criteria to create information concerning the same criteria. Then, further processing is applied on this information in the attempt to create meaningful knowledge that leads to human's decisions. Finally, new raw data is returned to the repository to close the cycle.

NLP is one of the means to process the information retrieved, specifically the human's natural language of it. Once the information undergoes the NLP pipe line, the outcome is a new form of knowledge created. This knowledge supports the human brain to make new decisions which in turn can create new data redirected to the repository (see Figure 1.3 for illustration).

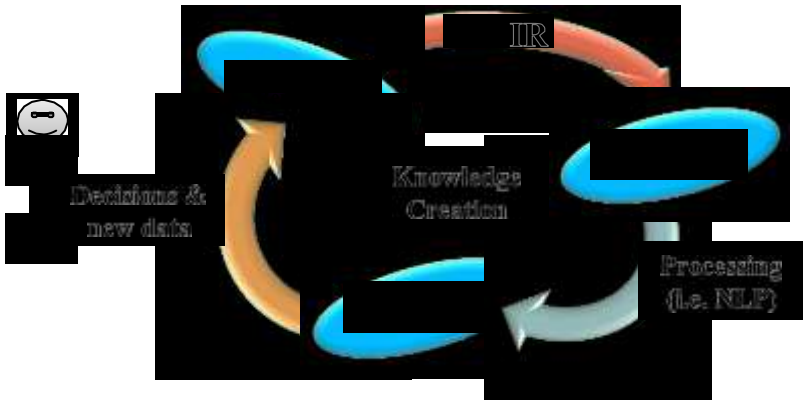

Figure 1.3: Knowledge creation cycle

1.7.2 Microtext

Microtext is a new term defined by Jeffrey Ellen in his survey paper "*All About Microtext*" (Ellen, 2011). The term refers to any text that is generally short in length, semi-structured, and contains informal language and grammar. The most obvious example of Microtext is micro-blogs (i.e. twitter site), and text exchanged in chat rooms. Other examples with same characteristics but could be lengthier include emails, blogs, wikis, and discussion forums.

1.7.3 Tasks in NLP

Research within IR and NLP covers a wide range of tasks, some of which we will be using in this research, including:

- **Sentence breaking:** Usually, sentences ends with a full stop which makes it easy to determine its boundaries. However, the full stop is also used for other punctuations like abbreviations, titles, and URLs. It is the job of sentence breaking task to determine the type of the period whether it is an end of sentence or not. Furthermore, this task sets clear sentence boundaries for further processing.
- **Text tokenizing:** There are many methods to tokenize text, the most common one being Whitespace Tokenizer. Its main task is to divide text into the appropriate phrases whether as single or multiple words.
- **Stemming:** Stemming is the process of returning the word to its stem, or morphological root in most cases. This is done by removing the word's inflectional endings.
- **Topic detection:** This goes one step further than clustering the information, since the clustered text could be unsupervised and labeled with the topic/theme of the cluster.
- **Text similarity:** The task of finding and appreciating similar chunk of texts based on a given relationship.
- **Automated summarization: Automated** summarization is still one of the experimental research areas, since there could be many variations of techniques based on the text format, length, and context system. As an example: summarizing a regular document is different than summarizing a micro-blog due to the length and the topics that could be covered in each.

1.7.4 The Summarization Problem

Summarization is the act of creating a summary; which is defined by (Jones K. S., 1999) as:

> "*A reductive transformation of source text to summary text through content reduction by selection and/or generalisation on what is important in the source*"

The definition suggests three main points in the reduction process: summarization, selection, and importance scaling of the text. What we mean by importance scaling is assigning an importance level to each segment of the text based on specific linguistic characteristics. These characteristics include, but are not limited to, language structure, meaning, and context (Martinet & Palmer, 1960). Understanding these characteristics using a computer is certainly not a simple task, and this is the roll of Artificial Intelligence (AI) in the context of IR.

Automated summarization is the process of creating a summary, as defined above, using a computer program. The main challenges involve encoding the computer program to understand natural language, its ability to determine ideas in the source text, and to rank those ideas based on their importance. Although there has been a wave of interest since the last decade, there is still much work to be done when it comes to automated summarization.

There are two main methods for automatic summarization, with everything else falling back to one of these methods: extraction and abstraction. Extraction is the process of selecting the most valuable and related information from the source text and presenting it in the summary. Abstraction involves the concept of Natural Language Generate (NLG), as it requires paraphrasing of the source text in order to generate a summary based on the meaning of the important idea. While abstraction is the more favourable method, it is also more challenging as it requires heavy machinery to handle complex algorithms for using Natural Language Understanding and Generation (Das & Martins, 2007). Within each of these methods, many summarization branches exist such as single vs. multi-document summarization, as well as properly grammatically and formatted/structured text vs. semi or non- formatted/structured text. These are in addition to the new short text format, unstructured chat/discussion/email writings, especially in the era of high speed communication and smart devices that prompt users for short communicated messages.

1.8 Conclusion:

1.8.1 Benefits

MOOCs have returned to society with obvious benefits. The fact that they are open comprises, in itself, increased accessibility and flexibility, and are accommodating to any individual despite his/her age, gender, or financial status (as most courses are free). The online framework eliminates time zone and geographical restrictions, while adding to the flexibility and choice of courses. This format also offers the developers with flexibility to provide a wide range of educational tools and delivery formats, which in turn improves the pedagogical process. Therefore, the advantages and benefits of MOOCs are not only paying back to a few individuals, but to the entire globe by benefiting all nations.

1.8.2 Motivation:

With benefits come challenges. The he discussion above describes some of the challenges in the pedagogical process that aggregated from the way LMSs and instructors use some of the above-mentioned tools and technologies. Even in MOOC platforms, it is clear that some of its components are disconnected from the main and initial course content, and even newly-created knowledge is secluded in different formats throughout the system. Take for example discussion forums, new posts, questions, answers, or comments, all of which can have new knowledge created by learners or instructors, but which could be lost or hard to locate between various posts. The same issue applies to blogs and micro-blogs. With wikis, the situation is a bit different but not much better, since the wiki page is edited and information is usually added in its logical place. Yet, someone going through the course material might not know that there is a wiki answering his/her curious question. That is because these discussion tools are currently missing the link to the main topic.

These tools can provide a better interaction for the pedagogical process. If we can incorporate new online technologies such as WEB 2.0, WEB 3.0, E-Learning 2.0, Open Educational Resources (OER), and 3D simulations, we can improve learning and collaboration.

We believe it is time to start thinking outside the box with respect to discussion tools, and we need to take the discussion to the outside world and link it with its related knowledge. In other words, we need to link the discussions with their related knowledge outside their containers, either through a forum, wiki, blog, or micro-blog.

Once we have the discussions linked to the main content, we need to apply some of the NLP techniques to be able to provide a meaningful condensed version of the related information.

Also, having social media is an asset, whereby internet users are noticeably active on social media sites such as Facebook, Twitter, and Google+. Therefore, discussion tools need to integrate the social aspect to encourage spontaneous collaboration between students.

The current social media sites – mainly those can integrate with MOOC platforms – do incorporate to an extent the social media feature to the platform as a whole, but not for the purpose of the discussion and most importantly, not for discussion forums.

With all existing technologies and sophisticated tools incorporated in LMSs that support and encourage collaboration and knowledge sharing, MOOC discussion forums noticeably lack in participation. For example in edX, only 3% participate in discussion forums, whereas in Stanford MOOCs, the highest participation level was 10%, with the average being less than 5% (Hill, 2013).

Therefore, we need to enrich the interaction and collaboration by connecting the course content with each other in a convenient way to all participants (instructors and students). This will connect the initial course content with the isolated and scattered cells of knowledge in the discussion tools (Discussion Forums, Wikis, Blogs, and Micro-Blogs). We must then summarize this knowledge for the users in the same portal. In this thesis, we are proposing new techniques that we believe can help learners connect, contribute, collaborate, and comprehend the knowledge. It will also provide a summarized version of the discussions to get the benefit of the full material at once. Ultimately, it will improve the pedagogical process. The two new techniques are:

1. Course content annotation linked to the discussion forum, wiki, blog, or sticky notes in a form of micro-blogs.
2. Information collection and summarization.

1.8.3 Research Goals:

To accomplish the above-stated goal, we will follow these steps:

- Obtain an open-source MOOC platform
- Set up a server to host the MOOC platform (web host), database, and email server.
- Design a new media player wrapper plug-in as an interface to wrap the existing media player. The wrapper would ultimately handle three media formats (video, audio, and text). This interface will:
  - Allow users to initiate and link to new discussion forum, Wiki, Blog, or Micro-Blog as an annotation at a specific time of the video and audio content or at a specific word or sentence in text content.
  - Display an up-to-date summary of all the discussions had by all users.
  - Require some database additions to handle the linkage and media timing of the annotations.
- Finally, demonstrate the result of the summarization algorithm and use some of the existing metrics to evaluate our result in comparison to existing tools.

The new design will visualize to the users the various knowledge related to the course content or to any specific topic. It will also encourage spontaneous discussions participation, since users will have the ability to post or reply without losing track of the media or of their train of thought.

Chapter 2

Frameworks for Building MOOCs

In the previous chapter, we briefly descried the history and time line development of eLearning systems. In addition, we stated that MOOCs have their own variations of developments and improvements. This chapter provides the background information that is required to have a complete understanding of the proposed work. The chapter is divided as follows: Firstly, we will conduct a comparison between a few existing MOOC platforms. The second part views the main developments and future trends of current MOOCs. One of the potential developments that can be incorporated in MOOC platforms to improve the learnability is Text Summarization, which is one of the NLP tasks that has not reached its maturity state yet. Finally, we will determine the right MOOC platform to work with as well as the best tools with which to integrate the discussion forums and summarization algorithm into the main content.

2.1 Comparing MOOC Frameworks

A MOOC framework is a programming framework for digital networked learning intended to assist decision makers/admin/designers/academics plan and deploy such new learning environments based on goals, context, and expertise. Any MOOC framework should emphasize the role of technology/media, learner profile, social learning, quality, and interactive & participatory learning/teaching. Different MOOC frameworks were compared based on the following criteria:

- Software license: open source, proprietary, mixed, or Software as a Service (SaaS) subscription.
- Deployment model: client server web hosting, dedicated hosting, or cloud hosting.
- Development method: in-house, off-the-shelf, or (SaaS) subscription

As we have mentioned in Chapter 1, we did list a few MOOC platforms. Some of these platforms provide only a MOOC hosting service such as Coursera[8], edX[9], Stanford[10], Udacity[11], and many others. These websites provide their platform as a MOOC hosting or listing service, and are in most cases limited to certain institutes and/or include a registration fee for the MOOC.

[8] Coursera website: https://www.coursera.org
[9] edX website: https://www.edx.org
[10] Stanford classes website: https://class.stanford.edu
[11] Udacity website: https://www.udacity.com

Certainly, the software is not available for us to use or customize with the exception of edX, which was released in March 2013 based on XBlocks, yet there could be costs attached to using the platform as we will describe below.

On the other hand, some existing proprietary LMSs launched their own version of MOOC platform that leveraged its LMS features, yet is a separate platform or website in most cases. Some of these platforms include Canvas provided by Instructure[12], Open Courses by Desire2Learn[13], and CourseSites provided by Blackboard[14], all of which are also out of reach since they are proprietary platforms.

Finally, we will look into open source platforms from two points of view: existing platforms that extended their LMS to accommodate MOOCs, such as Moodle[15] and Sakai[16], and new open source MOOC platforms such as edX, OpenMOOC[17] and Google CourseBuilder (GCB)[18].

As mentioned, edX started as an enclosed platform only to offer MOOC hosting, but in March of 2013, they released the open edX code to the public with the support of Stanford University, Google, MIT and others. However, installing and configuring Open edX might incur some costs since the platform is provided with an Amazon Web Services (AWS) Cloud Formation template comprised of a number of AWS resources that are not free as referenced in the edX configuration manager readme file. For this reason, edX was excluded from the selected platforms.

2.1.1 Moodle for MOOC Framework

Moodle – which stands for Modular Object-Oriented Dynamic Learning Environment – is an open source LMS that has been around since 2001. The platform has numerous features and tools that evolved as a result of educators' feedback and experimentations, and developers' dedication. To enhance support for users and educational institutes who were interested in making Moodle their LMS platform, Moodle, since the start, launched its community site, which used Moodle itself and acted as a central knowledge base for the Moodle community (Dougiamas, About Learn.Moodle.net, 2014). As mentioned, Moodle features a wide range of

[12] Instructure website: http://www.instructure.com
[13] Desire2Learn Courses website: https://opencourses.desire2learn.com
[14] https://www.coursesites.com
[15] https://moodle.org
[16] https://sakaiproject.org
[17] http://openmooc.org
[18] Google CourseBuilder website we built with google app engine: https://lugomooc.appspot.com

tools that can be leveraged toward any pedagogical style, and hence MOOCs as well. The LMS contains the common discussion tool (discussion forums), and includes Wiki pages for courses, a blogs for users, a messaging system, a calendar, progress tracking, etc. Being the top LMS with 73 million users throughout the globe, and the third by total with 87,000 customers, Moodle didn't have any challenges in attracting MOOC audiences (Barrish, 2014). In fact, Moodle launched its first MOOC platform "learn Moodle" website back in September 2013[19]. "*The website aims to facilitate learning and collaboration to inspire better teaching everywhere*" (Dougiamas, About Learn.Moodle.net, 2014). The first MOOC hosted on the site was "Teaching with Moodle: An Introduction", which run for 4 weeks in September 2013 (Dougiamas, About Learn.Moodle.net, 2014). Moodle can be one of many systems run by an educational institute and it can be easily integrated with other systems to create the full picture as in Figure 2.1 (Hunt, 2012). Nonetheless, Moodle can be used perfectly as a stand-alone system.

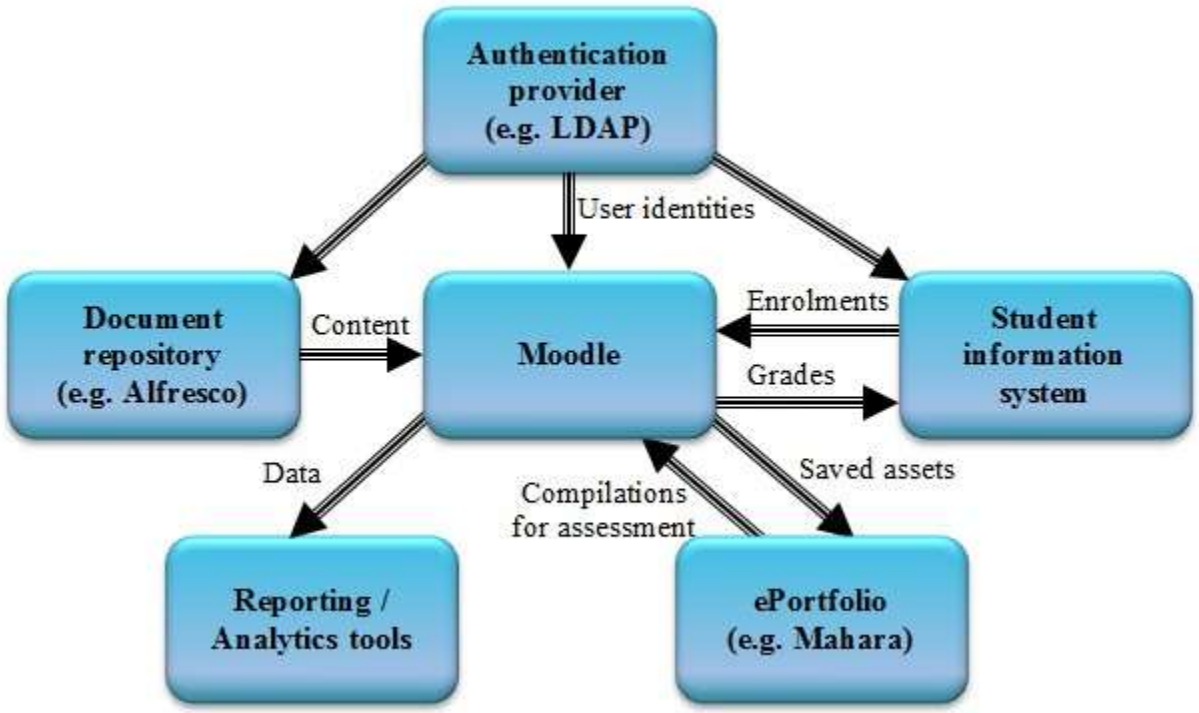

Figure 2.1: Typical university system architecture

When we take a closer look at the center box, we find that Moodle consists of – by name – many modules and plug-ins built using PHP code that is setting in a web server, the web application is connected to one of the supported databases: MySQL, PostgreSQL, Microsoft SQL

[19] http://learn.moodle.net

Server, or Oracle, and finally the file/content storage (moodledata folder) to manage the uploaded files (see Figure 2.2).

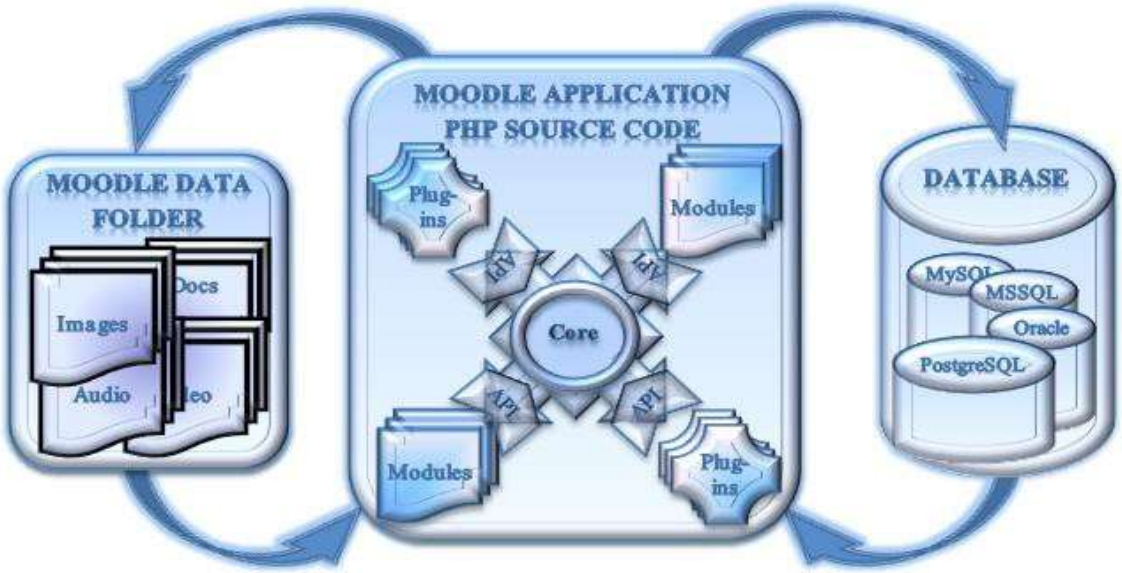

Figure 2.2: Moodle system architecture

Moodle's core application is the core infrastructure that is required to build the LMS. It also includes all necessary components needed for the plug-ins to communicate with the core. For instance, Moodle core handles courses and activities, user's authentication, authorization and full profile, the Yahoo user interface JavaScript library, upgrading functionality, and finally, logs and statistics. Plug-ins, on the other hand, are categorized into specific types, and communicates with the core using Application Programming Interfaces (APIs) just like modules. For instance, "an authentication plug-in and an activity module can communicate with Moodle core using different APIs". Activities, such as forums – and resources, such as pages, are some of the most important plug-ins in addition to other types of plug-ins such as themes, language packs, etc. (Moodle Developers, 2014).

Moodle uses a web server to host the core application and extensions (plug-ins and modules). This fact may frame a Moodle LMS with scalability limitation depending on the system implementation design. Moodle can be implemented all on one server: as a web server, a database server, and a file server, and this design is not suitable for MOOC. The other approach is to use several web servers with a load balancer and multi-cluster database servers. Finally, a separate file server for storage, as this is more appropriate for MOOC, yet it could be costly to build such an infrastructure unless it already exists. Another option would be to host Moodle on

the cloud with one of the cloud providers such as LIIP. Going with the latter approach would suggest that the software is to be used as a service, as described below:

> *"The University of Applied Sciences of Northwestern Switzerland (FHNW) asked Liip to host its high-performance learning management system. To ensure that the system is continually updated, the FHNW opted for the Moodle "Software-as-a-Service" solution."*
>
> (Liip AG, 2014)

Finally, Moodle can be used in both forms as a software provided, which is the more economic option, yet not scalable or SaaS, yet more expensive and can defeat the purpose of using an economic open source application since cost is one of our drivers. However, it is still a strong candidate since we are not going for a full scalable MOOC platform at this point. Furthermore, once we test our new feature in one platform, we can easily transform it to another platform.

2.1.2 OpenMOOC Framework

OpenMOOC is an open source MOOC platform licensed under the Apache license 2.0. The platform implements a complete open MOOC LMS (OpenMOOC Home Page, 2012). The platform is developed using PHP for the authentication component as it is based on SimpleSAMLphp, and uses Python/Django for the main engine and the other components. The platform is designed with interactive video to manage the content and intelligent discussion forums for users and teachers to communicate. The MOOC course in OpenMOOC is divided into units that consist of knowledge pills, which is a collection of short videos to construct a deliverable topic and other supporting material such as links, documents, and practice material (short questions, quizzes, etc.). Figure 2.3 demonstrates the architecture of OpenMOOC as it is described in the OpenMooc website[20]. The authentication component (*Identity Provider*) is based on SimpleSAMLphp[21] which is based on PHP programing language. Then, there is the main engine, *Moocng,* which allows the creation and management of courses. Finally, the third component, *Askbot*, is a Q&A platform acting as the main communication tool between teachers and students (OpenMOOC Home Page, 2012).

[20] www.openmooc.org
[21] https://simplesamlphp.org

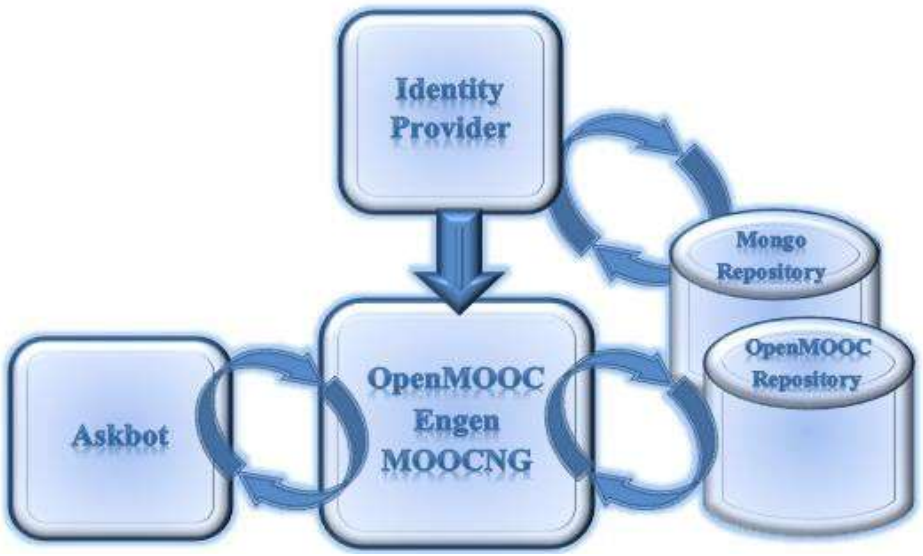

Figure 2.3: OpenMOOC Architecture

The final version of OpenMOOC v1.5 integrated additional features such as learning analytics and peer review questions, in addition to the existing features from previous versions such as Wikis, students' messaging system, etc. (OpenMOOC Home Page, 2012).

2.1.3 Google CourseBuilder (GCB) Framework

Google CourseBuilder is an open source MOOC platforms, a Google project using Google AppEngine. Therefore, the course would act as an application hosted on Google AppEngine utilizing Google's powerful available resources. Notwithstanding, limited resources are assigned to each free application (course) and for more and full scalability application, Google offers a paid service for additional resources. GCB uses Python as the main development language. It offers easy and a simple start-up with its sample course deployments process, since the course is hosted on Google AppEngine, no environment setup overhead is required. To deploy the initial course, and to be able to use Google AppEngine, the administrator must have Python run time environment, as well as Google AppEngine, and finally the GCB sample application for setting up an initial MOOC. One of the advantages of using GCB is the beauty of integrating the course with other Google products in order to have the full communication experience. That includes, but is not limited to: Google groups and discussion forums, Gmail, Drive, Blogger, Hangout, and we should not forget the most powerful search engine Google search. However, additional work is required to integrate these tools. Another great feature with GCB is the one login system that gives users access to all Google tools.

With all Google capabilities and scalabilities, there are limitations and drawbacks to what can be achieved with GCB. Creating courses requires technical knowledge with Python, HTML, and JavaScript to be able to customize a course and integrate it with other Google products. Since the project is integrated with Hangout for live discussions, as well as limited office hours, it is limited to the number of users that can connect to the Hangout session, which is 10 to 15 individuals, depending on the account type. With Hangout messaging, the limit increases to 100 users, but registration for the course is still unlimited (Bishop, 2013). The University of Alicante researchers in Alicante, Spain has developed a very successful MOOC platform based on GCB called UniMOOC[22]. To overcome some of the above limitations with GCB, the UniMOOC consortium consists of different committees: a steering committee, a scientific committee that handles developments of the educational model, a technical committee which defines the technical requirements, and an organizing committee in charge of the educational resources developments and the implementation of technical requirements (Peco & Luján-Mora, 2013). Their architecture, as we will see in Figure 2.3, consists of: GCB MOOC eLearning platform v1.51, NOSQL database, existing Google products, two custom-developed UniMOOC systems – massive email and accreditation – that have been integrated with the GCB, and the virtual secretary system that acts as a second component in the accreditation system. That in turn interacts with Mozilla open badges system and stores students' achievements.

[22] http://unimooc.com/

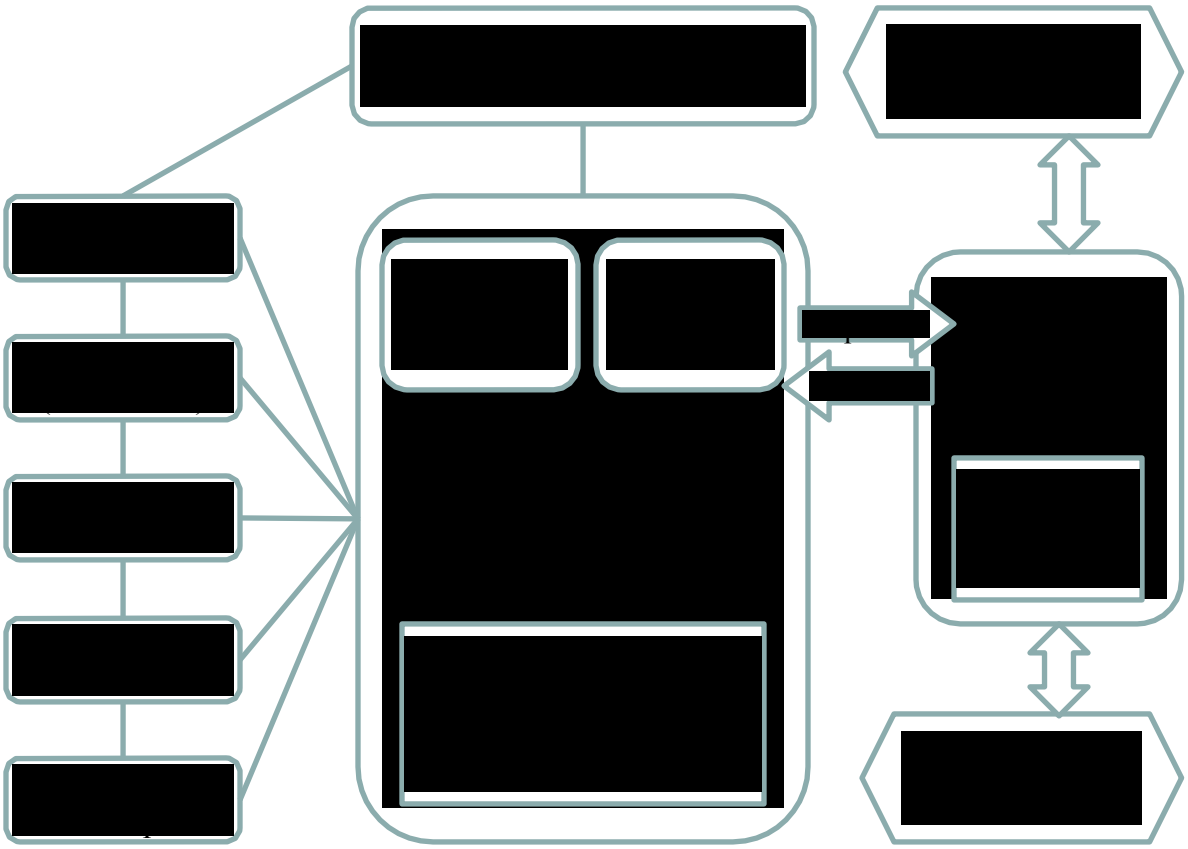

Figure 2.4: Architecture of UniMOOC (Peco & Luján-Mora, 2013).

As we can see in Figure 2.4, there was a significant amount of development done toward this project. Mostly, to integrate external systems together in order to achieve the pedagogical requirements set by the scientific committee. Another challenge was the fact that limited resources were available with Google's free version of projects and products, yet as pointed out by Peco & Luján-Mora, the cost was low compared to other web hosting (Peco & Luján-Mora, 2013). As a result, GCB can be a simple and quick launch of a basic MOOC platform, or it could be a massive project integrated with many Google products and other custom-built modules or features.

The following Table 2.1 compares the three MOOC platforms described above.

Table 2.1: MOOC platforms comparison summary

Criteria	Moodle for MOOC	GCB	OpenMOOC
Software license	Open source	Open source	Open source
Supported languages	PHP, JavaScript, CSS	Python, JavaScript, CSS	PHP, Python/Django, CSS
Deployment model	Client server	Cloud hosting	Client server
Installation & Configuration	Successful setup in our environment.	Successful setup and deployed on Google's Appspot.com	Incompatible with our environment.
Supported features			
Discussion forums	√	√	√
Wikis	√	x	√
Blogs	√	x	x
Micro-Blogs	x	x	x
Content media	Video, Documents, Audio	Video, documents	Video, Documents

From the table above, both GCB and Moodle platform can be used for our experimental study, however since Moodle is a full LMS platform and is fully integrated with its own discussion forums, Wikis, and Blogs, it would be more suitable to add our new feature. The only challenge would be that although Moodle does not recommend using JavaScript, the platform is still compatible with it.

2.2 MOOC Development Trends

The original MOOC, which was developed by Downes and Simons, is based on the connectivism theory (Downes, 2012). Downes describes learning in his introduction of his latest book "Connectivism and Connective Knowledge: Essays on meaning and learning networks" as follows:

> "*Learning is the creation and removal of connections between the entities, or the adjustment of the strengths of those connections. A learning theory is, literally, a theory describing how these connections are created or adjusted*"
>
> (Downes, 2012)

The initial MOOC by Downes and Simons was later referred to as cMOOC. Because other MOOC providers started their MOOCs through their own version of MOOC platforms or embedded version of their LMS that accommodated MOOCs, these MOOCs are referred to as xMOOC. Not many developments were made on cMOOC, however xMOOC witnessed enormous developments and improvements that enhanced the courses and platforms with helpful tools for administrators, authors, and users. Hence, this section focuses on the main developments and trends of the xMOOC platforms.

2.2.1 Interactive Videos:

The focus on video to be the main content delivery option for MOOCs opened up a wide development opportunity for video interaction. One of the most obvious developments with MOOCs is interactive video lectures, noticed from the evolving changes to video lectures within the current MOOC platforms such as the ones mentioned in this chapter. Movies and educational videos are no longer a static story-telling process. Rather, they have integrated interaction contents that are seamlessly joined with the video to allow the user to participate by providing feedback on a quick quiz question or by applying some object movements using a mouse and keyboard interacting with the videos. Interactive videos provide students with self-paced learning ability, and enable them to repeat the video for mastery learning. This is not available with live online courses taught by an instructor (Du, 2014).

One of the newer projects for video interaction is the Popcorn JavaScript library created by Mozilla[23]. This project allows the user to produce video with dynamic embedded content, like displaying a live map, current weather, or a Wikipedia link. In addition, users can embed a simple text, image, or a popup text window on top of the video[24]. The use of Popcorn JavaScript library enables interaction between the video/audio time line and other HTML elements on the page (Cuypers & Knopper, 2013). This great tool is very effective for video production to capture the users' interests, but is not so valuable at making them interact with the video.

Although these developments have made videos interactive and dynamic, they are still linear, meaning that the video always starts on one side and always ends on the other side. What we could see in the near future are non-linear videos, where the media will be offered based on the user's interaction with it. For example, if a user answers a quiz question incorrectly, the

[23] http://popcornjs.org
[24] https://popcorn.webmaker.org

video would automatically replay the section of the video to help the user review the relevant material and provide them with the opportunity to select the correct answer. Furthermore, this service could expand to offer material based on the level of the individual user's knowledge.

2.2.2 Social Media Learning:

Social media is driving a large number of projects, due to its massive reach and influence on users (network). Social media also has a power of influence since it acts similar to word of mouth: friends recommending to friends. For that reason, there is no wonder social media sites are the marking hub for all types of advertisements. Hence, social media could leverage this aspect to become a knowledge hub as well. Social learning is the kind of learning that takes place in asocial setting, which usually facilitates a smooth and unburdened knowledge-sharing environment.

> *"Learning would be exceedingly laborious, not to mention hazardous, if people had to rely solely on the effects of their own actions to inform them what to do. Fortunately, most human behavior is learned observationally through modeling: from observing others, one forms an idea of how new behaviors are performed, and on later occasions this coded information serves as a guide for action."*
>
> *(Bandura, 1977).*

Social learning is not a new concept; it has been around since 1977 when Albert Bandura first wrote about his Social Learning Theory. However, with the new social media and social network sites, this theory could be reshaped to create an online social learning environment that is prompted at request and only when needed. Social learning, although it could have a wider reach yet in most cases, could have a shallower knowledge due to the same social setting and learning style, unless of course, the social setting itself is reshaped to fit a deeper pedagogical setting.

2.2.3 Crowdsourcing:

By definition, Crowdsourcing is the process of collecting needed information, ideas, or services through a large number of people or focused community (Howe, 2006). MOOC creates the perfect setting for crowdsourcing projects. When large numbers of interested learners are

joined, they could act as a focus group, it is a great opportunity to collect information, analyze it, and present it as organized knowledge (Alario-Hoyos, et al., 2013).

2.2.4 Data Analysis and Processing:

What is more important than collecting the data in the crowdsourcing process is analyzing it, extracting information, and presenting the knowledge in a condensed format to the users. This is a very important task, since the massive amount of information generated from collaborative tools like discussion forums can be overwhelming (Alario-Hoyos, et al., 2013). Applying some NLP tasks like text summarization, question answering, or word tagging can assist with this direction to convert the information into knowledge.

2.2.5 NLP Support

Interactive tasks in network learning require large amounts of natural language or textual media. One can use NLP for providing extra support to MOOCs through processes like POS tagging, question answering, relationship extraction, and summarization.

As discussed in chapter 1, text summarization is still in the early stages, especially with the rapid/promptly social media tools like online posts, micro-blogs, discussion forms, and even emails from small devices. These types of online tools are pushing the language to shift from the well-formatted/structured language to a short/concise, unformatted/unstructured text that falls in the Microtext category. In this section, we will discuss a few of the summarization systems that perform well in a multi/single, and well formatted/structured documents: Open Text Summarizer (OTS)[25], Dragon Toolkit[26], and MEAD System[27]. We will focus on the basic summarization algorithm within these systems as they support – with the exception of OTS – many NLP tasks and furthermore, they implement some of the evaluation algorithms like Recall-Oriented Understudy for Gisting Evaluation (ROUGE)[28] which we will discuss later in this thesis.

- **Open Text Summarizer (OTS):** This is open source software available as a library and command line tool. OTS is a single document summarization system whereby users can input text into the program and the output would become highlighted summarized text. OTS evaluates each sentence in the text and assigns an importance level to them. Finally, utilizing

[25] http://libots.sourceforge.net/
[26] http://dragon.ischool.drexel.edu/
[27] http://www.summarization.com/mead/
[28] http://www.berouge.com/Pages/default.aspx

the threshold summary percentage of the original text, the most important sentences will be highlighted as a summary (Yatsko & Vishnyakov, 2007)[29]. The main idea lies in the occurrences of a term and represents its importance in the article/document. Many of the same terms represent an important idea in the document. The OTS algorithms lie in the following steps:

1. [illegible]
2. [illegible]
3. [illegible]
4. [illegible]
5. [illegible]
6. [illegible]

OTS has proven its high performance in summarizing documents as it outperformed similar systems such as Subject Search Summarizer, Copernic Summarizer, and Essence (Yatsko & Vishnyakov, 2007).

- **Dragon Toolkit:** Dragon Toolkit is open source Java-based development package in the area of IR and text mining TM. The package's applications include classification, clustering, summarization, and topic modeling. The toolkit integrates packages of NLP tools and enables it to present text collections with various schemes; most importantly text relationships. The scalability features of the toolkit distinguish it from other systems such as Weka, as it is designed for large scale applications with small memory requirements. The toolkit implements LexRank summarization algorithm and ROUGE system for summary evaluation. The LexRank algorithm summarizes text based on sentence salience. After clustering the sentences, the system evaluates the centrality of each sentence and extracts the most important sentences for the summary (Erkan & Radev, 2004). Figure 2.5 demonstrates the pseudo code of computing the centroid score:

[29] http://libots.sourceforge.net/

```
   input  : An array S of n sentences, cosine threshold t
   output: An array C of Centroid scores
 1 Hash WordHash;
 2 Array C;
 3 /* compute tf×idf scores for each word */
 4 for i ← 1 to n do
 5     foreach word w of S[i] do
 6         WordHash{w}{"tfidf"} = WordHash{w}{"tfidf"} + idf{w};
 7     end
 8 end
 9 /* construct the centroid of the cluster */
10 /* by taking the words that are above the threshold*/
11 foreach word w of WordHash do
12     if WordHash{w}{"tfidf"} > t then
13         WordHash{w}{"centroid"} = WordHash{w}{"tfidf"};
14     end
15     else
16         WordHash{w}{"centroid"} = 0;
17     end
18 end
19 /* compute the score for each sentence */
20 for i ← 1 to n do
21     C[i] = 0;
22     foreach word w of S[i] do
23         C[i] = C[i] + WordHash{w}{"centroid"};
24     end
25 end
26 return C;
```

Figure 2.5: LexRank algorithm (computing centroid score) (Erkan & Radev, 2004)

The algorithm utilizes the Term Frequency - Inverse Document Frequency (TF-IDF) word weight above the threshold to be the centroid of the cluster. It then computes the score for all sentences in the document.

- **MEAD System:** MEAD is open source software developed in the University of Michigan in 2000, offering summarization and evaluation of multi-lingual text. The tool supports four summarization algorithms: TF-IDF, centroid, position-based, and query-based. The evaluation algorithms include recall, precision, kappa, and relative utility, as well as other content-based measures such as cosine, word overlap, and bigram overlap (Radev, et al., 2006).

MEAD was developed using Perl. In addition to the core system, it requires additional Perl modules (XML related modules). MEAD also depends on an external software package to run.

There are two summarization baselines available with MEAD: lead-based and random. A lead-based baseline traverses through each document and selects the first sentence, then the second, and so on based on the sentence features. Random, on the other hand, selects sentences randomly from each cluster. Both baselines are limited to the number of selected sentences to the

desired summary size. Note that sentence selection in both baselines take into account the sentence features, which is the basis of sentence scoring (Radev, et al., 2006).

MEAD scores each sentence based on specific features. By default, MEAD uses Position, Centroid, and Length. Other default sentence features are also available such as the number of named entities, and anaphora. In addition to the default features, MEAD includes features such as SimWithFirst: computes the similarity with the first sentence; IsLongestSentence: assigns a score of one for the longest sentence (otherwise zero); and three additional query-based similarity measures. The sentence score is calculated based on three stages (cluster, document, and sentence) exploiting the features of that stage. Initially, the clustering features are applied. Then document features' checks are applied to each document in the cluster. Finally, the sentence features are applied for each sentence in the cluster. Based on these three stages, the sentence is assigned a score that determines its rank.

There are a few other online tools available to perform automated summarization such as: Text Compactor[30] (based on OTS system), Free Summarizer[31], Online summarize tool[32], Automatic Text Summarizer[33], and Web Summarizer[34]. Additionally, there are other available summarization APIs which we did not examine, as their algorithms are not available and they incur user fees.

2.3 Proposed MOOC Forum-Media Context Summarization System Architecture

We will build our system using two main components. The first will act as an integration tool between the main content and all other discussion and knowledge-sharing tools. Furthermore, this component presents the different knowledge-sharing tools in a centralized page with the main content. The knowledge-sharing tool on which we will focus during this research is discussion forums. Yet, the system design will have the flexibility to accommodate other tools in the future such as wikis, blogs, and micro-blogs. We chose YouTube videos as the main content delivery tool because of its wide use for educational videos and its simplicity to link to any LMS. We chose discussion forums because they are the main discussion tool that exists in

[30] http://www.textcompactor.com
[31] http://freesummarizer.com/
[32] http://www.tools4noobs.com/summarize
[33] http://autosummarizer.com
[34] http://www.websummarizer.com/Pages/Default.aspx

every LMS. Although wikis and blogs have not been used as much in the past, they are now being incorporated into the LMS, especially in MOOC platforms.

The second component we will implement uses NLP tasks to generate an extraction summary of all discussions made for the video. The following is a high level architectural diagram, which we will discuss in detail in the implementation chapter.

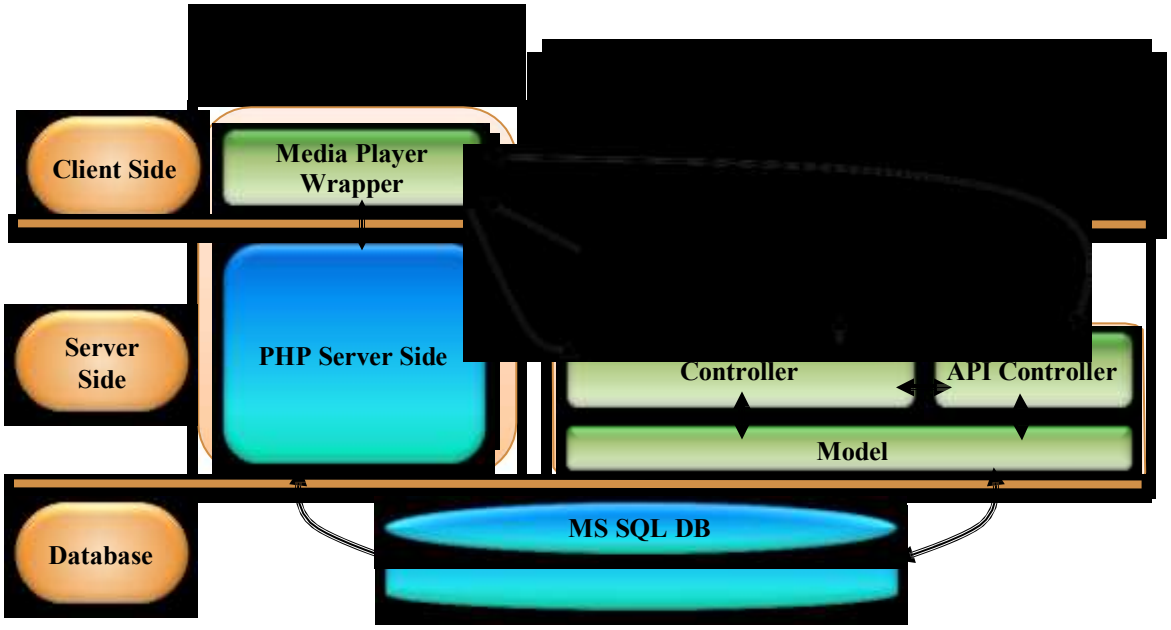

Figure 2.6: High level architecture of the MFMCS system

In the next chapter, we will discuss our proposed system in more detail, focusing on design and implementation.

Chapter 3
Implementation of the MFMCS System

3.1 An Overview

We have seen in chapter 1 how current MOOCs and LMS, in general, have information and shared knowledge distributed between the teaching contents and all collaborative tools[35] within the platform. These tools have no direct relationships/links with the main course contents (e.g. video content). These collaborative tools are the hub for providing effective teaching and learning experience. If such collaborative tools are not designed well in order to guide learners toward the correct goals, we will end up with a tool that deters learners from focusing on the main learning tasks as they would need to switch back and forth between these collaborative tools and the learning contents. What is more challenging is the massive amount of newly-created knowledge, as some MOOCs may have thousands of registered users. In this case, learning can become frustrating instead of being an entertaining activity. Therefore, we aim in this research to encourage spontaneous knowledge creation and sharing, and to present concise key points of the shared discussions related to the video content. This chapter introduces a new development as a value-added MOOC that aims to encourage learners to engage with the contents without losing their focus on the main learning task. The architecture of this added value MOOC is comprised of two components described as a combiner and summarizer. The combiner is presented as a media player wrapper component, and integrates the collaborative tools with the main content delivery tools, mostly videos in MOOCs, in one interface. While the summarizer applies NLP the summarization task to generate an extraction summary of all shared knowledge that is related to the main content. The summarizer is represented by the server side Summarization Application component.

The current chapter demonstrates the implementation of our MOOC Forums-Media Context Summarization (MFMCS) system. The MFMCS consists of two main components: the JavaScript Media Player Wrapper (JSMPW), and the NLP summarization application. Initially, we will discuss our methodology in achieving the research goals which includes the development of the MFMCS system. Then, we present a high level architecture of the MFMCS system followed by its functionalities. We will then zoom in on each component, the JSMPW on the

[35] Collaborative tools such as: Discussion Forums, Wikis, Blogs, and Micro-Blogs.

client side and the Summarizer application on the server side, describing their functionalities and implementation in detail, including examples and pseudo codes for the key parts of the implementation. Finally, we will present the database layer, additions and changes, and its importance to the system.

3.1.1 Methodology

To address the research goals of this thesis, we have developed the MFMCS system that will link the video content with users' discussions that take place within the discussion forums tool in one interface. Then, reduce the overflow of information in the forum by presenting a summary of the key points of the discussions. We will achieve this by creating two components in the MFMCS system. The first component is the JSMPW, which resides on the client side and links the discussion forums with the video content. The second is the summarization application that resides on the server side, and is mainly responsible for the summarization task. Finally, we will store all the data in a relational database.

We used Microsoft Visual Studio 2013 (MSVS) as a main development tool for both components, whereas Microsoft SQL Server 2012 was used to store our relational database. The following three points describe in more detail the various integrated libraries, programing languages and seawares in our development of each component:

1. **Client Side (JSMPW Application):** The client side application was developed using **HTML**, **JQuery**, and **CSS** to design the JavaScript Media Player Wrapper (JSMPW). The JSMPW consists of video wrapper, modal dialog, and the Information and Knowledge (I&K) wrapper.
2. **Server Side (Summarization Application):** The server side application was developed using the MVC architecture whereas **ASP.NET** web pages (**Razor**) syntax were used to design the view. **C#** programing language was used for the control and model components. We also used Language-Integrated Query (**LINQ**) in most of our development to interact with the datasets retrieved and posted from and to the database. Finally, JavaScript Object Notation (**JSON**) and Extensible Markup Language (**XML**) formats were used to transfer data between the system components. The server side application also contained a set of APIs, which we developed to enhance reusability and maintainability of our core application. We have created two APIs that are used by both the client side and server side

annotation web applications: the annotation API, and the Natural Language Processing API. We will discuss them in detail in section 3.2.

3. **Database:** On the database side, we used Microsoft SQL Server 2012 to hold Moodle's Database named LUMM, which stands for Lakehead University Moodle MOOC. The Summarization Application required access to one of Moodle's tables in the LUMM in addition to four new tables, which were created for the Summarization Application database changes. Details are contained in section 0.

3.2 The MFMCS Architecture

Figure 3.1 below shows the MFMCS application in three different layers, namely Client Side, Server Side, and the database. The highlighted green boxes represent our contribution as MFMCS system. Two main components were developed as part of the MFMCS system: the JSMPW within Moodle application, and the entire Summarization Application with its two sub applications. The first component is on the client side and is required to connect the discussion forums from the Moodle application to the video content. It performs this task by allowing the users to begin a new discussion at any point in time during the video. The server side holds Moodle's PHP server side code in addition to our second component, the Summarization Application, whose main objective is to provide a NLP summarization solution for all discussions related to the video content. In addition, the Summarization Application acts as a facilitator between the JSMPW and the database. Finally, the third layer is the MSSQL server relational database that stores all data in records within tables in the LUMM database[36].

[36] LUMM stands for Lakehead University Moodle MOOC, used as an application code and database name.

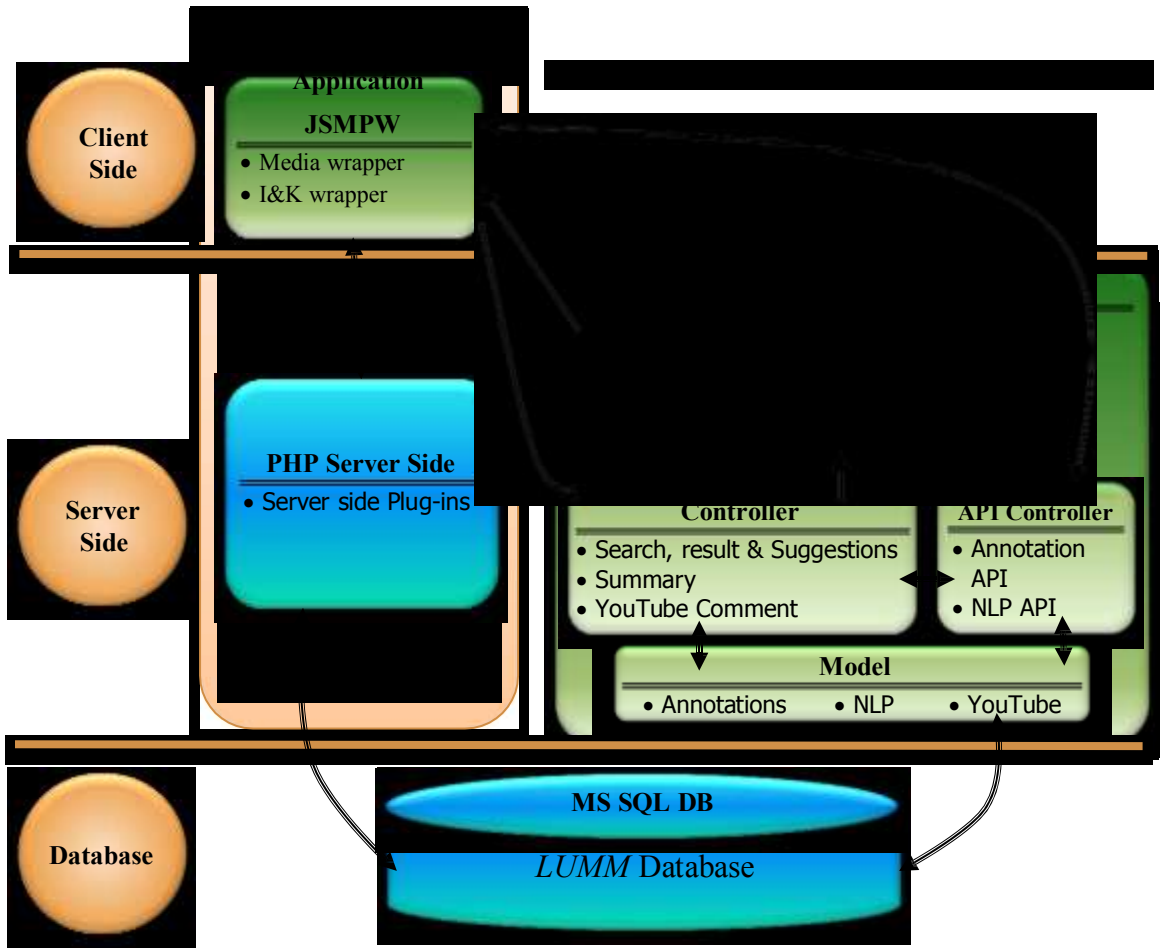

Figure 3.1: High level MFMCS system architecture

The JSMPW – first component of the MFMCS system – is designed as a plug-in and developed using jQuery. This code is designed in a class-like Object Oriented Programming (OOP) architecture using the jQuery class library. The second component of the MFMCS system is the Summarization Application developed in two sub-applications: a web application and APIs. Communication between both components occurs via HTTP protocol, which we will demonstrate in detail in the coming sub-sections of each component. In this section, we will explain the functionalities of the MFMCS system and the architecture of each layer in detail.

3.2.1 The MFMCS Functionalities

The MFMCS system functionalities are demonstrated below in Figure 3.2. The system has five actors: Users (who can be students or instructors) and four system actors (who could be internal or external to the MFMCS system): NLP API, Annotation API, YouTube API, and Moodle's Discussion Forums Module. The user is the primary actor and can initiate any of the

main use cases (on the left side): Display Discussions, Search Discussions, Create New and View Existing Discussions via Display New Discussion page, and Click Annotation Icon use cases respectively, and finally, View YouTube Comments and Discussion Summary.

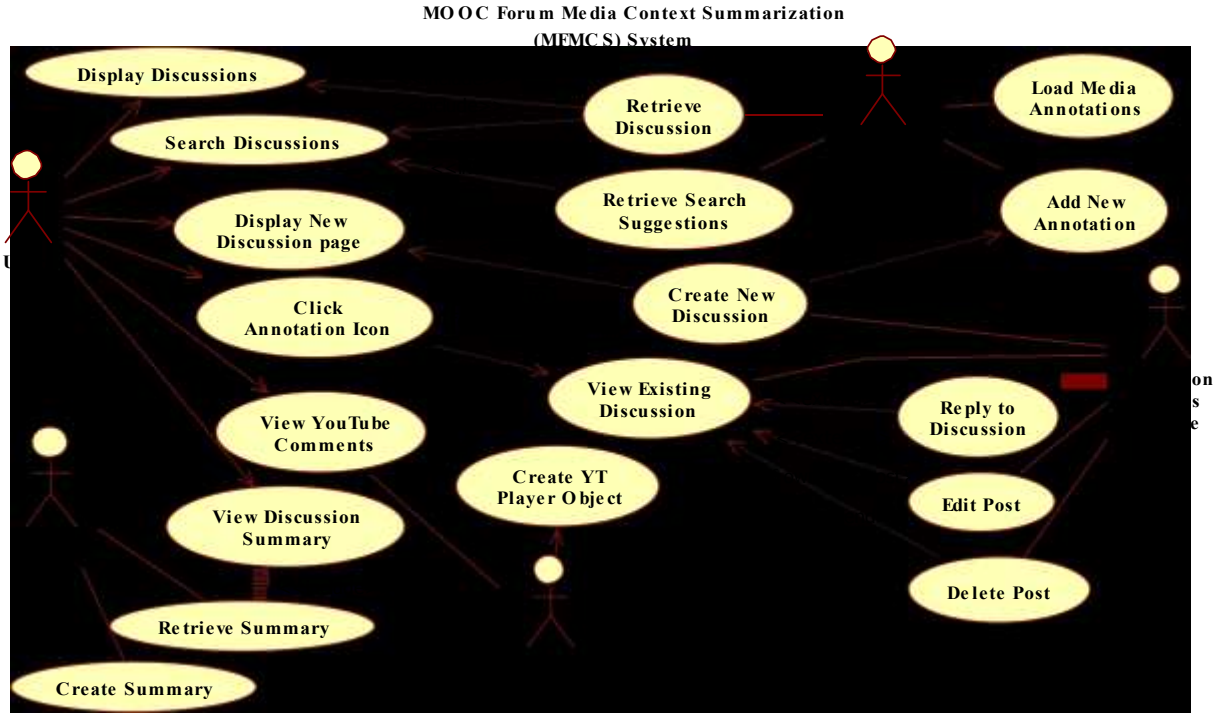

Figure 3.2: MFMCS system use case diagram

In the next three sections, we will explain in detail the architecture, main use cases, and main scenarios for each component of the MFMCS system.

3.2.2 JSMPW Architecture (First Component)

In this sub-section, we will discuss the Graphical User Interface (GUI) design and the full architecture of the JSMPW. The JSMPW design is encapsulated in a jQuery/JavaScript file that acts as a plug-in for the Moodle LMS, and is designed as a plug-in to wrap the YouTube default player. Moodle has a media plug-in that contains many features such as Info box, Logobox, external caption files, livestream, search bar, and snapshot; however these features cannot link the discussions from the discussion forums, wikis, or blogs with the video. Therefore we decided to build our video player wrapper on top of the default YouTube Player thus the reason for using only the default player. We designed the JSMPW as a plug-in to match Moodle's design style and have the flexibility of working with different MOOC platforms. The JSMPW was developed as our first component of the application, and we will exploit its features for the NLP algorithm

in the second component. Although this is a new tool of its kind and we have not seen a similar existing implementation that serves the pedagogical field in the LMS, it is the closest system that we found with a similar idea that was implemented for annotating micro-blogs (comments) for audio files at a specific time during the video. The website that implements this is Sound Cloud[37]. We applied the same concept on LMS using discussion forum annotations. The screen capture in Figure 3.3 below illustrates the main components of the JSMPW:

Figure 3.3: JSMPW GUI

1. **Media Wrapper**: This is the main wrapper that surrounds Moodle's video player and contains the following components to incorporates Moodle's discussion forums to the media time line:

[37] https://soundcloud.com/

a. **Annotation Menu**: Contains the list of collaboration tools that are possible to annotate on the media38.
b. **Annotation Icons**: Represents a new discussion topic at a specific point of time during the video. With their mouse, users can hover over any of these icons to view its discussion title. They can also click on the icon to display the full discussion, including its reply threads. Once the discussion is open, users can post a reply to any post.
c. **Annotation Bar**: A progress bar containing a video seek and many annotation icons. Users can click on any of these icons to view the discussion posts.

2. **I&K Wrapper:** Contains a set of tabs that helps the user get direct access to shared information. It also includes the newly extracted summary tab that provides users with new knowledge based on the video discussions. The following are the tabs within this wrapper:
   a. **Search Tab**: Allows the user to search for any discussion posted on the current media that is related to a search query that the user provides. The results are then displayed to the user under the search field.
   b. **Discussion Tab**: Lists all discussions posted to the current video in addition to any discussion threads available.
   c. **YouTube Comments Tab**: Displays a list of the latest 25 YouTube comments posted on the YouTube page. These comments are retrieved by calling the YouTube Comments API via the YouTube comments controller in the Summarization Application Web App.
   d. **Discussion Summary Tab**: Displays an automated extracted summary of all discussions on the current video.

The GUI interface in Figure 3.3 is the entry point for the primary actor (users) in Figure 3.2 to initiate any of the associated use cases. Similarly, any of the other actors can invoke the associated use case(s) at different trigger points. Next, we will walk through some scenarios to explain the functionalities of the main use cases:

[38] The MFMCS system is designed to accommodate in additional to the Discussion Forums, Wiki, Blog, and Micro-Blog. We only implemented the discussion forums in this thesis and will have the rest as future work.

1. **Load Media Annotations Use Case**: As soon as the course content page is requested via the browser, the page checks if all objects in the Document Object Model (DOM) are loaded and ready for use. Then the page loads the JSMPW JavaScript file. After that the JSMPW instantiates a YouTube player object by calling the YouTube iFrame API. Once the JSMPW receives confirmation that the YouTube player is ready it will create the media and I&K wrappers and display them on the page. It will then initiate a call to the Annotation API to retrieve all annotations for the video and will finally display Annotation on the page. See the use case scenario illustration in Figure 3.4 below.

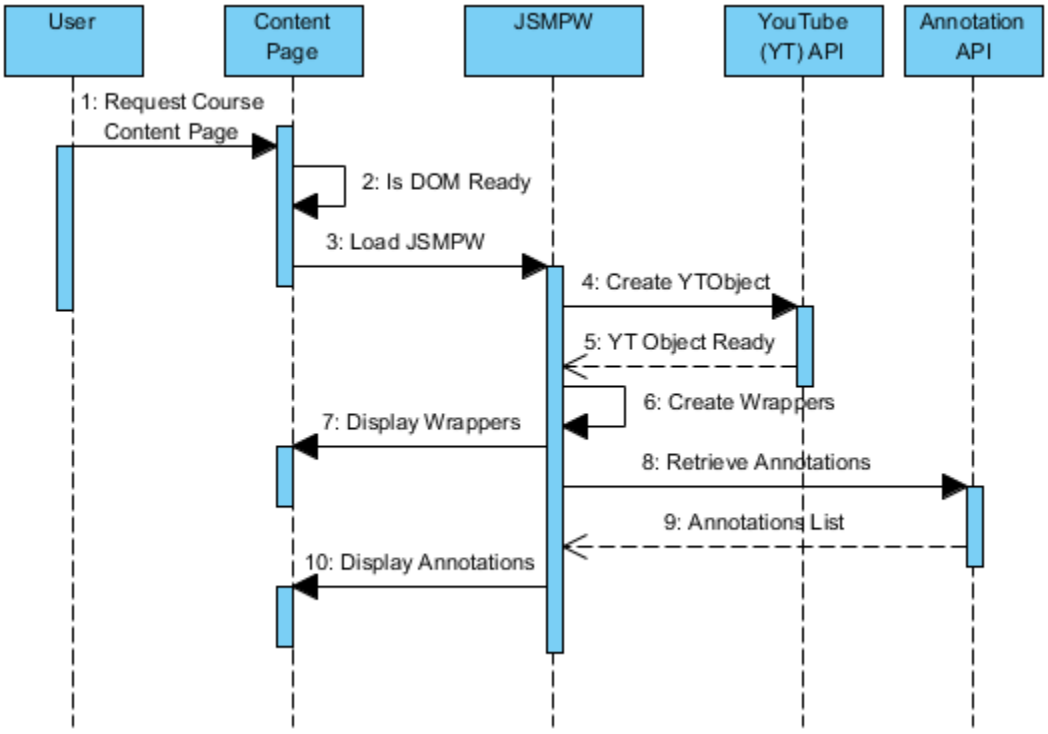

Figure 3.4: Content page load sequence diagram

2. **Create New Discussion Use Case**: The user can then initiate a new discussion forum page use case by clicking on the Discussion forum menu item shown in Figure 3.3, which will trigger the discussion forum module in Moodle to display its content in a dialog modal on top of the playing media. After loading the Discussion Forum form and the "Post to Form" button is selected, a new discussion is then created. Also, the "Add New Annotation" use case is triggered and a new annotation record is created by the annotation API. Both actors (Annotation API and Discussion Forum Module) are responsible for formatting the data entered by the user and storing it to the database. Figure 3.5 illustrates the above scenario in the sequence diagram of creating a new discussion use case.

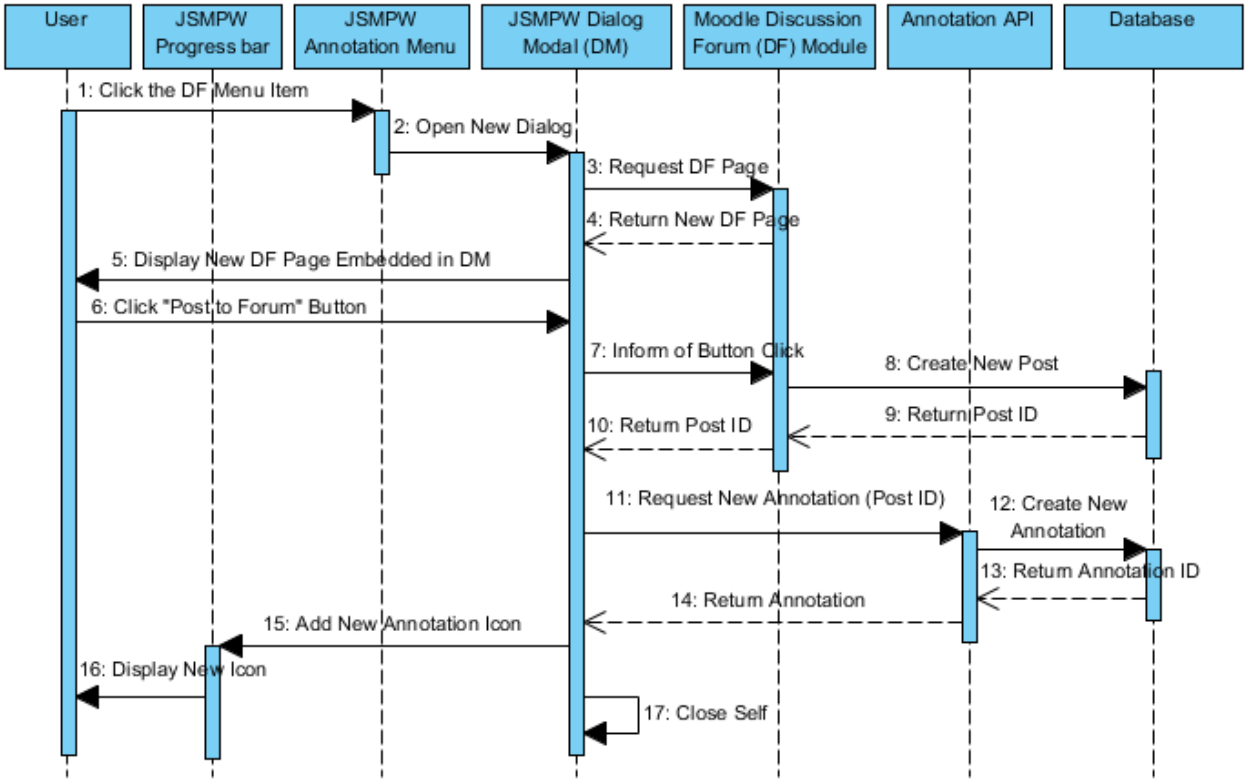

Figure 3.5: New discussion sequence diagram

3. **View Existing Discussion Use Case**: Another important use case is the View Existing Discussion by clicking the annotation icon. Then, the New Dialog Modal will open requesting discussion, passing the post ID from the discussion forum module. The requested discussion is then retrieved from the database and displayed to the user within the dialog modal. The user can click the reply button in the dialog modal to add a reply thread to the discussion. The discussion forum module receives the request from the dialog modal and returns the Reply form page to be displayed to the user. Once the user inputs the reply text and clicks the "Post to Forum/Submit" button. The dialog modal informs the discussion forum of the form submission. The new post will be stored in the database by the discussion forum module. Finally, the dialog modal will close itself after the confirmation. Figure 3.6 illustrates the sequence diagram of this use case scenario.

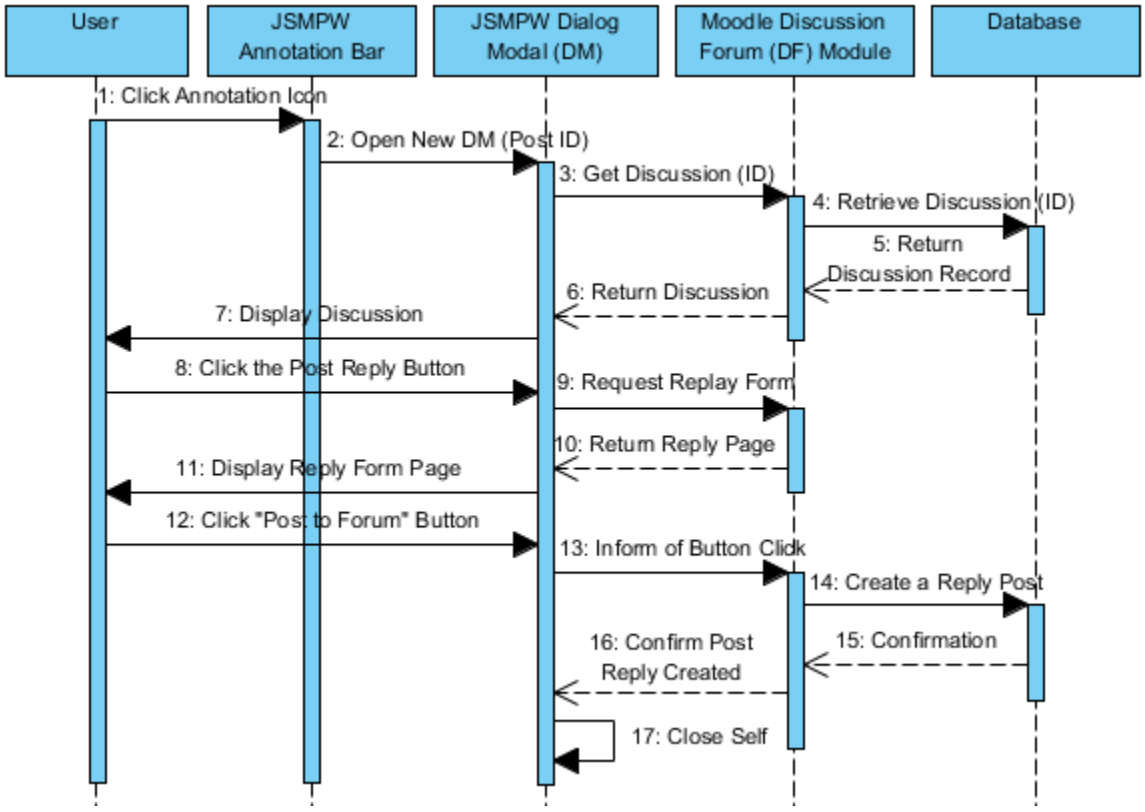

Figure 3.6: View & reply discussion sequence diagram

The architecture of the JSMPW in Figure 3.7 shows the main components of the JSMPW and its interaction with other components and external resources.

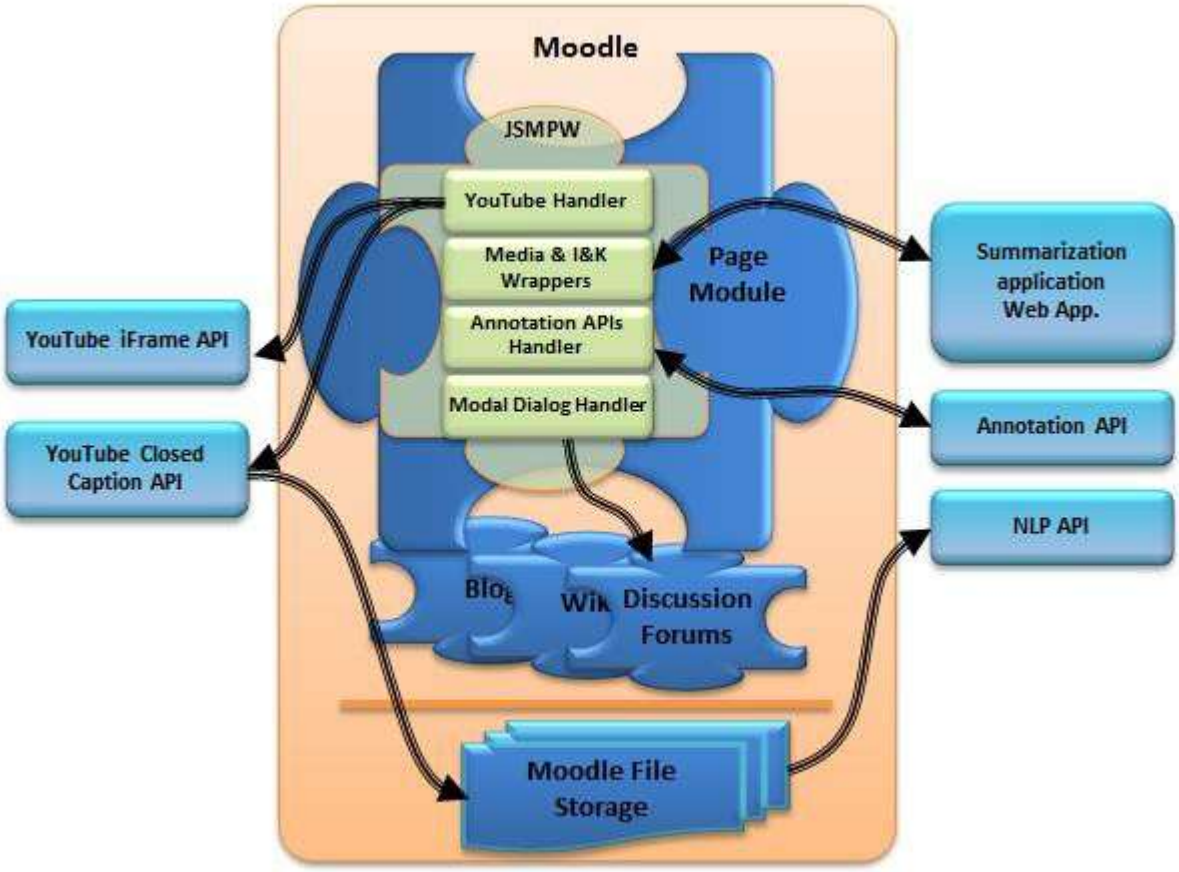

Figure 3.7: JSMPW (client side layer) architecture

As we explained in section 2.1.1 of chapter 2, Moodle is built as the Moodle core, modules, and plug-ins that communicate via internal APIs. As presented in Figure 3.7 above, the JSMPW interacts with various components, some of which are internal to Moodle whereas others are external. The pseudo code below in Figure 3.8 demonstrates the tabs in the I&K Wrapper connects to the Web App from the Summarization Application.

```
1.  I&K Wrapper Tabs Clicked event() {  // This event is triggered when the any tab is clicked
2.     Fade out the background
3.     Open the Tabs wrapper
4.     Set the focus on the selected tab in the I&K Wrapper
5.     Switch (clicked tab){    // Tabs could be: Search, Discussions, YouTube Comments, & Summary
6.        Case Search:
7.           Display Search page  // Use AJAX call to retrieve the Search Form page
8.        Case discussions:
9.           Display Discussions       // Use iFrame to imbed the discussion page in the I&K Wrapper.
10.       Case YouTube Comments:
11.          Display YouTube Comments page  // Use AJAX call to retrieve the YouTube Comments page
12.       Case Summary:
13.          Display Summary page  // Use AJAX call to retrieve the Summary page
14.    }
15. }
```

Figure 3.8: I&K wrapper tab-click pseudo code

Next, we will explain each of the JSMPW units and the main components shown in the above diagram.

- **JSMPW units**: The JSMPW is a filter[39] plug-in in Moodle used by the page module. We designed it with four main units to handle different functionalities:
  - **YouTube Handler**: The YouTube handler responsible for any calls to YouTube APIs, such as the YouTube iFrame API, which creates the YouTube player object, and the YouTube timed text API, responsible for capturing the Closed Caption (CC).
  - **Media & I&K Wrappers**: The media wrapper is a class-like OOP that can handle different types of media objects using the polymorphism model. The I&K wrapper is a container for the search, list of discussions, YouTube comments, and discussion summary pages.
  - **Annotation API Handler**: The main purpose of this unit is to post and retrieve the annotations for the particular media by making various calls to the Annotation API.
  - **Modal Dialog Handler**: This unit is based on **jQuery.Simplemodal.js** library explained in Appendix I to display the embedded version of Moodle's discussion forums to the user on top of the playing media. Using the Modal Dialog help users stay on the same page and not navigate away from the main presentation.

[39] Filter plug-in is one of Moodle's classifications of different types of plug-ins.

- **Moodle page modules:** This is one of Moodle's core modules, which is responsible for displaying the course content page. We modified the "view.php" file to include references to our JSMPW JavaScript and CSS files.
- **Moodle plug-ins:** Moodle contains many plug-ins that hold instances of information such as Discussion Forum, Wiki, and Blog. The JSMPW makes calls to these plug-ins[40] to be displayed within its Modal Dialog unit on top of the media (video). These plug-ins connect directly to the database to store and retrieve their data.
- **Summarization Application:** The Summarization Application is a container for two sub applications, which are the web app and the APIs.
    - **Web App.**: This is a server side application designed using the ASP.NET MVC architecture. The web application contains web pages such as search form, search result, search suggestions, summary, and YouTube comments page. The web application is responsible for rendering the view data to the client side after applying the business rules in the model. The model exchanges data with the database by a direct connection via ADO.NET[41] or via the Annotation APIs. One of the models obtains YouTube comments directly from an external YouTube feed API.
    - **APIs:** We built two APIs that can be accessed internally and externally: the annotation API and the NLP API. The annotation API is the main API for adding/posting, retrieving, and searching for annotations. The NLP API is the main API that applies all the NLP rules – which we will explain in the next section – to generate the discussion summary.

3.2.3 Summarization Application Architecture (Second Component)

The server side in Figure 3.1 contains Moodle's PHP server side code and the Summarization Application and acts as a middle tier between the JSMPW and the database. The Summarization Application works in parallel with Moodle's PHP server side application. The Summarization Application contains a web application and two APIs. The web application is designed based on the MVC design pattern and is responsible for presenting the API's response data to the client. It is also responsible for the search, and YouTube Comments features.

[40] The initial design included all three modules Discussion Forum, Wiki, and Blog, but only implemented the Discussion Forum.
[41] http://msdn.microsoft.com/en-us/library/aa286484.aspx

Within the Summarization Application, the web application functionalities are presented by the following use cases from Figure 3.2: Display Discussions, Search Discussions, View YouTube Comments, and View Discussion Summary. The APIs sub application is represented by the two-system actors (Annotation API & NLP API) and the following use cases: Retrieve Summary, Create Summary, Retrieve Discussions, Load Annotations, and finally, Add New Annotation.

1. **View Summary Use Case**: When the user clicks the Discussion Summary tab from the I&K wrapper, the JSMPW initiates a call to the summary controller in the Summarization Application Web App. The controller selects the summary view and attempt to retrieve an up-to-date summary via the annotation model. If no summary exists, the controller then triggers an NLP API call to create one. The NLP API controller informs its model with a request for summary. The NLP model performs the summarization process to create a summary. Then the model stores the updated summary to the database. Finally, once the up-to-date summary is in the database, the annotation model notifies the summary view to display the summary to the user. Figure 3.9 below is a sequence diagram of the view discussion summary use case where the summary in the database is not up-to-date.

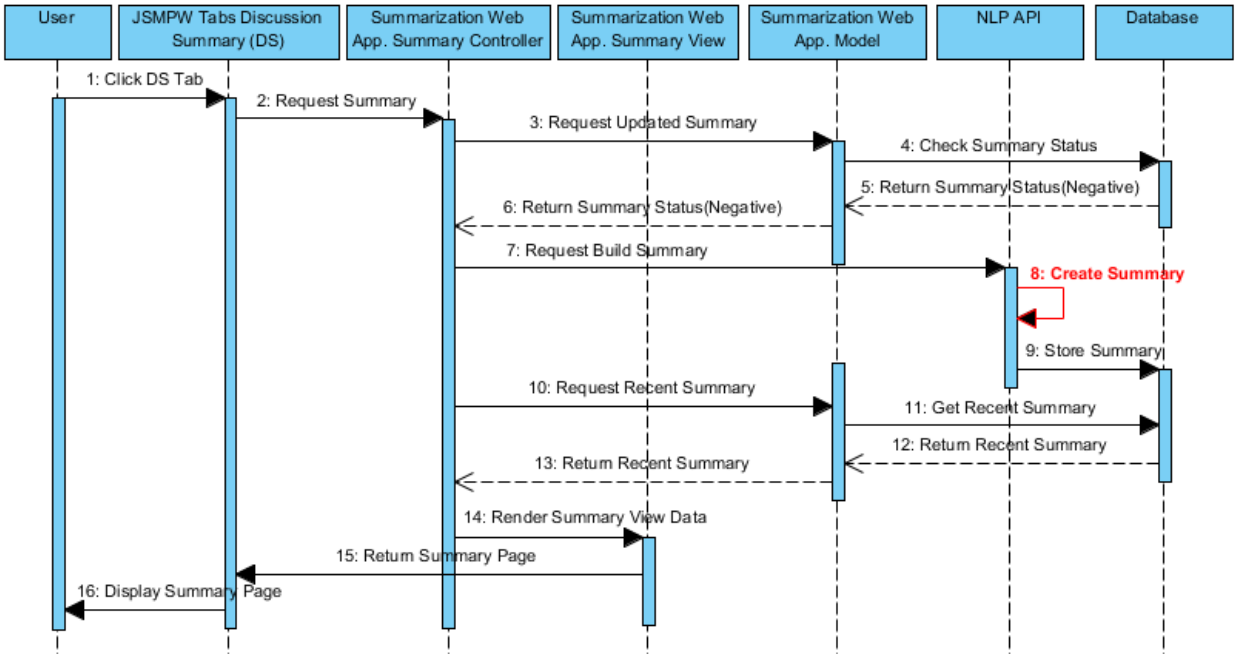

Figure 3.9: View discussion summary sequence diagram

Step 8: Create Summary" listed in red in Figure 3.9 is the enhanced summarization algorithms that the NLP API applies in order to generate an extraction summary of the discussions posted on the current video. The summary is boosted by the JSMPW annotations. We exploit both the closed caption as a whole, as well as the annotation context (the closed caption surrounding the annotation time) during the summarization process. Later in this section, we will explain these algorithms step-by-step, and demonstrate a summarization example of one discussion.

2. **Search Discussions Use Case:** The search discussion use case includes the Retrieve Search Suggestions use case. Figure 3.10 demonstrates the sequence diagram of the search discussion and its included search suggestions use cases. A detailed scenario follows.

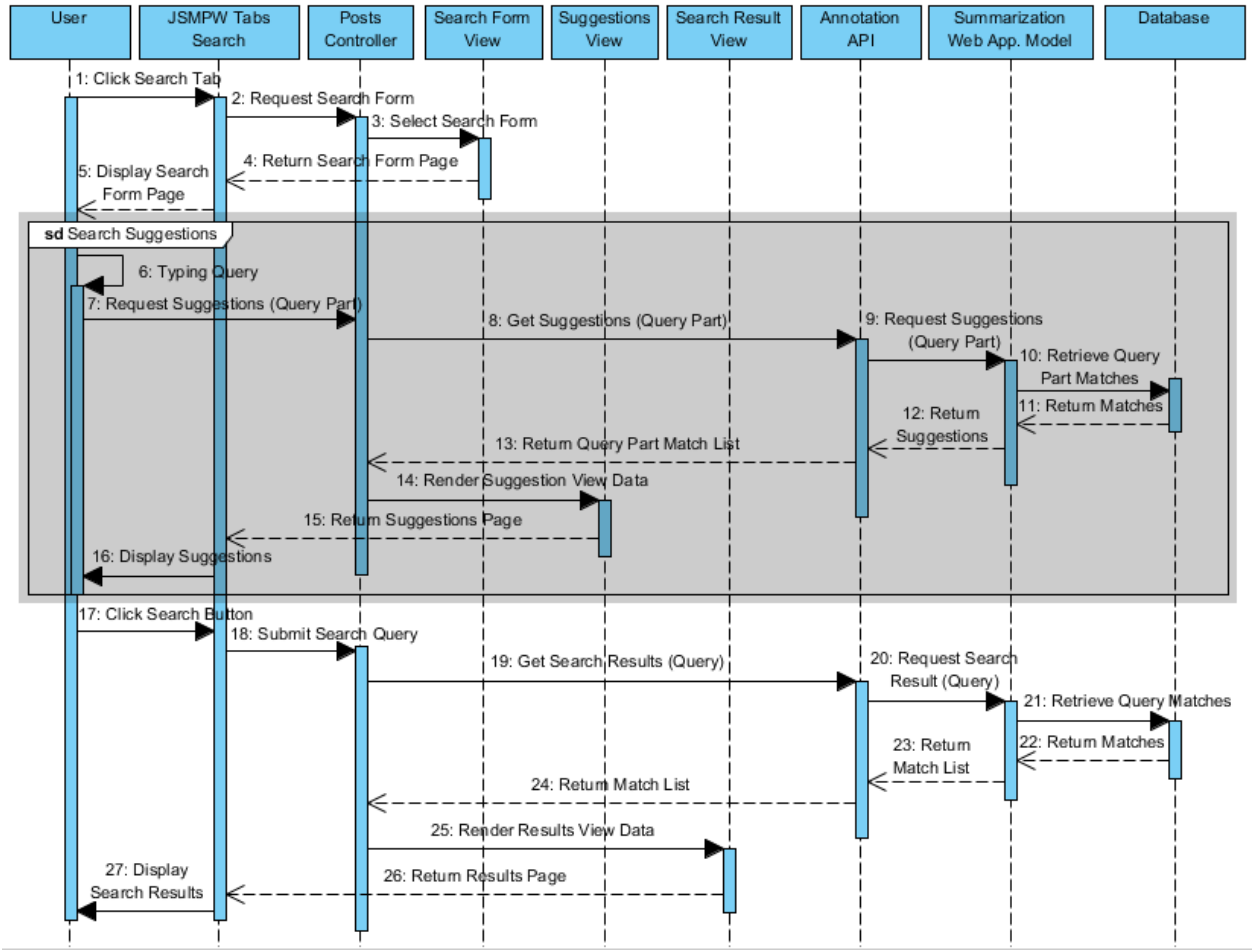

Figure 3.10: Search discussions & search suggestions sequence diagram

When the user clicks the Search tab, the JSMPW sends a request to the Posts controller in the Summarization Application Web App. The Posts controller selects the search form view and

renders it to the JSMPW to be displayed to the user. The user can then start typing the search query. As the typing starts and the number of characters exceed three, the search suggestions use case starts. For example, if the user is searching for the word "global", the search suggestions will start appearing after the user types the letters "glo", so the query part is now "glo". This query part is sent to Posts controller. The controller then calls the annotation API. The annotation API controller receives the query part and request suggestions from the model. The model contacts the database and retrieves the query part matches from the discussion posts stored in the forum table in the database. The database returns a list of matched posts that contains the query part to the model, the model to the API, and the API back to the Posts controller, for example a list might look like this "global warming, The glob is". The controller selects the suggestions view, and then renders the suggestions view data to the JSMPW, which will be displayed to the user. See search suggestions example in Figure 3.11.

Figure 3.11: Search suggestions example

This process happens in a loop (see step 6 in Figure 3.10) for each character change in the search field. When the user clicks the Search button (step 17 in Figure 3.10), the JSMPW submits search query to the Posts controller, the controller sends the request to the annotation API, and the API controller sends the request to the Summarization Application model. Then, the model retrieves the matched posts from the Forums table from the database. The matched posts are returned from the database to the Summarization Application model to the API then to the Posts controller. After that, the controller renders the results view data to the JSMPW to be displayed to the user. An example search result for the term "Global Warming" would be the all posts in the discussions contains the query. See Figure 3.12 below.

Figure 3.12: Search result example

Next, we demonstrate in Figure 3.13 high level architecture of the Summarization Application and the interactions within and outside of its components.

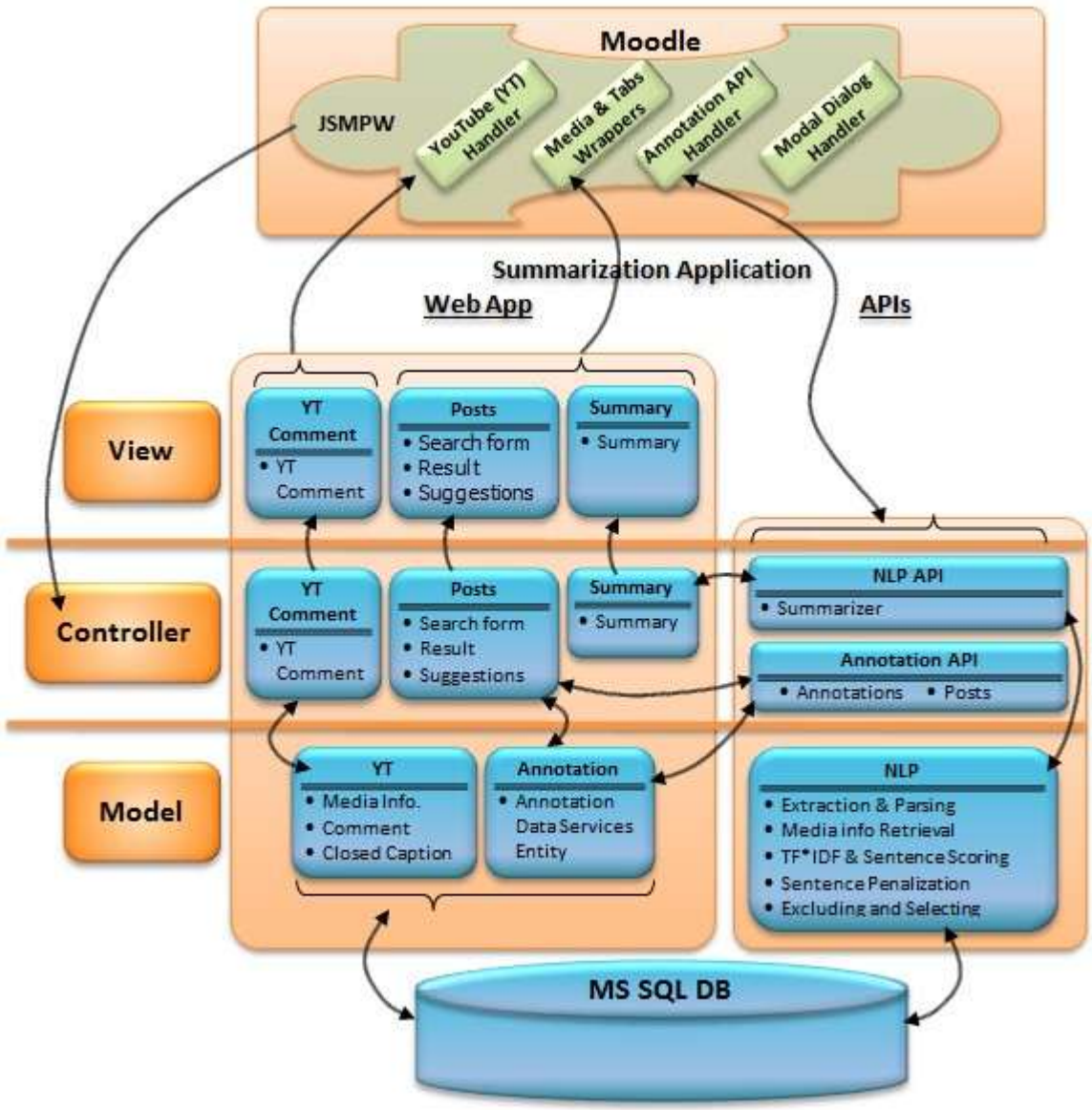

Figure 3.13: Summarization application components architecture on the server side layer

When the user clicks any tab from the I&K wrapper, the JSMPW initiates a call to the Summarization Application Web App. The appropriate controller selects the view requested and informs the model with the requested action. The model applies its logic based on the request and accesses the database directly. In some cases, like in the search and summary tabs, the controller initiates an API call, and the API controller communicates with its model which connects to the database. Finally, the model notifies the selected view with the updated data.

Figure 3.14 is a pseudo code for the search discussion controller method, the Forum Posts class, and the search API controller method. The pseudo code below demonstrates how the

search controller creates an instance of the Forum Post model (class) and calls the Annotation API to retrieve the search results. Finally, the API method returns the search result based on the parameters received.

```
1.  Search Result Controller () {       // This Controller is triggered when the search button is clicked
2.      Get URL Parameters (Video ID, Search Query)
3.      // The Annotation API call passes the parameters in the URI for the routing table to understand
        it
4.      Create ForumPosts Model = Call Annotation API http://.../api/Posts/{Video ID}/{Search Query}
5.      Render The Results Model to the Results View
6.  }
    ================================
7.  // ForumPosts class in the Model, matches the schema of the Database [Forum posts] table.
8.  // The Class stores the data in its properties as part of a ForumPosts instance
9.  partial class ForumPosts
10. {
11.     long id { get; set; }
12.     long discussion { get; set; }
13.     .
14.     .
15.     .
16.     string subject { get; set; }
17.     string message { get; set; }
18. }
    ================================
19. // GetPostsbyID method in the Annotation API Controller
20. [Route("api/Posts/{videoID}/{searchQuery}")]       // Method Routing signature
21. GetPostsByvID(VideoID, SearchQuery)         // Method name and expected parameters
22. {
23.     // The Database call is done using LINQ to Entity framework
24.     Select Posts from [Forum Posts] table Joined with [Annotation] table in the database where
        [Annotation][Video id] = VideoID & [Forum posts][message] contains (SearchQuery)
25. }
```

Figure 3.14: Search result controller, ForumPosts class, and API call pseudo code

The first layer in the above architecture is the View layer, which consists of the application view components that are displayed in the I&K wrapper of the JSMPW.

- **Search form page:** The Search form provides users with the ability to search for any keyword in the discussion. The result would be the discussion that contains the keywords.
- **Results page:** This page is called once the search is submitted to display the search results.
- **Suggestions page:** The search box provides suggestions after typing the first 3 characters in the search field. The suggestions are retrieved from the original posts/discussions.
- **Summary page:** The summary page is responsible for displaying the summary of the discussions.

- **YouTube Comments page:** The YouTube Comment page is an additional feature we added to display some of the actual comments on the YouTube page.

The next layer is the communication (controller) layer, which contains the MVC controllers and the API controllers.

- **Posts controller:** The Posts controller handles the interactions of the discussion forum's posts via the annotation's API with both the model and view. Posts interactions include a list of all posts/discussions, search, and search suggestions.
- **Discussion Summary controller:** This controller is responsible for the decision to either retrieve the updated summary from the model, or to call the NLP API to generate an up-to-date summary based on the recent discussions.
- **YouTube Comments controller:** This controller sends to the model the YouTube video ID and triggers the YouTube Comments view to display the data obtained from the model.
- **Annotation API controller:** This is an API controller, for the annotation information from adding new annotation, and for retrieving all or part of the annotations for a specific media. The main difference from the MVC controller is that there is no view for this controller, as it acts as an API. This API is usually called from the post controller in our application.
- **NLP API controller:** This controller is similar to the annotation API controller in the setup, but it handles the discussion summary controller calls. This API is responsible for all NLP tasks. Ultimately, it is responsible for creating the summary from all discussions about the media and storing it to the database.

The final layer in our application is the business logic and data layer, represented as the model in the MVC design pattern.

- **Annotation model:** This is the main objects holder for the annotations and discussions. Using the .NET entity framework, this model defines both the conceptual and storage models and the mapping between them (Microsoft). It is also directly linked to the database via the ADO.NET entity data model classes.
- **YouTube model:** This model handles YouTube feeds and APIs objects such as comments' fees, video information, and closed caption. The model makes a direct call to an external YouTube APIs like feeds (for the media information and comments), or obtains data from a source file like closed caption. The model is responsible for serializing and de-serializing the

.NET objects data types into XML or JSON-formatted data types for communication over the HTTP protocol.

- **NLP model:** This model holds the core of NLP tasks, algorithms, and data objects. The main tasks are listed below with a detailed description of the summarization steps taking place in the NLP model (see Figure 3.16 for more details). Also, Figure 3.15 describes the full summarization algorithms on the NLP API as part of the MFMCS system. The tasks listed below are based on their chronological execution order of the summarization process. We will also walk through a real summarization scenario providing details of each task to demonstrate its impact on the summary. For example, Table 3.1 contains one of the discussion posts and its two replies that were submitted online - Discussion ID (10183). We treat the discussion and all its replies as one document in the summarization process.

Table 3.1: Document/Discussion (10183) sample

subject : Political Neutrality An advocacy group that is non-political and not funded by oil companies, gives it a degree of legitimacy that others who oppose global warming do not have. With the focus on the search for the truth, the lecturer allows for more openness and free thinking.
subject : Re: Political Neutrality The world needs people who believe and are passionate. I have no problem with activists, climate activists or environmentalists - on the contrary - if it wasn't for some of these people, but - you have to consider people's vested interests and take precautions to make sure those interests don't get in the way.
subject : Re: Political Neutrality If someone has a habit of dishonesty of course one would be foolish to take their word. Likewise if someone is respected for their honesty and diligence, I'm more likely to take them at face value. I am always skeptical, and if one investigates further and finds that the asserter hasn't published the relevant work (or in this case hasn't published anything for 10 years), makes assertions that are not supported by any evidence, and upon further investigation, finds that the assertions are actually directly contradicted by real world evidence, then it would be foolish not to discount the assertions.

```
1.  Initialize the Corpus.                  // Creates an instance of DFCorpus class
2.  Connect to the database.                // Establish and opens a database channel.
3.  // The following line creates the Select Query by joining forums & Annotation tables to retrieve the
    posts related to the specified video ID.
4.     Posts List = Select all posts from forums table for the specified video ID.
5.  Close Database connection
6.  For each Post in Posts List {           // Loop through all selected Posts
7.     // parent is a column indicates the parent thread to the post if equal to zero then this is a new
       discussion.
8.     If parent=0 then
9.        Create new Paragraph (Post's discussion)      // New instance of DFParagraph class
10.       // The new DFParagraph instance applies the segmenting of the paragraph and instantiates
          instances of DSentence class. Then the new DSentence instance applies the tokenization of each
          sentence and instantiates instances of DPhrase class. Finally the DPhrase instance call the
          stemming algorithm and stores the term and stemmed term.
11.       Create new document (Paragraph)               // New instance of DFDocument class
12.       Add new document to Corpus
13.    Else
14.       Add post to existing document.
15.    End if
16. }  // End for each loop
17. Get and Load Video Information (video ID)           // Creates video Info Document instance
18. Create and Add Video Information document to Corpus // Add document to the Corpus
19. // Start the summarization tasks
20. For each document in Corpus {                      // Loop through all documents in the Corpus
21.    Calculate IDF Values()
22.    Calculate TF*IDF()
23.    Penalize Terms TF*IDF()
24.    Calculate Sentence Score()
25.    Boost Sentence Score()
26.    Exclude Sentences()
27.    Select Summary Sentences()
28. }  // End for each loop
29. Save Selected Sentences to the Database     // Store the summary sentences to the DB
```

Figure 3.15: MFMCS summarization algorithm

The first line in the algorithms above is the initialization of the corpus. The corpus is the first and main object that contains all other objects. Lines 6-14 contain the process of building the corpus by creating and adding new discussions to the corpus (lines 9 & 10), or adding posts/threads to existing document. Lines 15 &16 are designed to get the media information (title, description, & closed caption) and load it to the database (*Media Info*) table. It then creates a video in the document and adds it to the corpus. From line 2 to line 17 is the extraction and parsing steps. After that, starting from line 18 to 26, the summarization process starts by iterating through the documents/discussions list in the corpus and applying each of the function calls in the loop. More details and pseudo code of each of the functions follow at each step below.

- **Data extraction & parsing:** The first step is to retrieve the data from the database (line 4 in the MFMCS summarization algorithm, Figure 3.15 above). Moodle's discussion forum plug-in stores all posts in the forums table in the database. The NLP API retrieves these posts via the NLP model from the forums table from the DB. Once these posts are obtained, we start the parsing process (see Figure 3.18 for more details). The main class (DFCorpus) represents a corpus, each discussion is a document (DFDocument object) in the corpus, and each thread in the discussion represents a paragraph (DFParagraph object). We then parse the paragraph into sentences (DSentence object), and finally each sentence into phrases (Phrase objects). Since each post item in the database represents a paragraph, we focus on segmenting sentences and parsing phrases/terms (see segmented sentences example in Table 3.2, and stemmed tokens of each sentence in Table 3.3 We used regular expressions for the segmenting and tokenizing tasks (refer to the tokenizer code snippet in Figure 3.17). For segmenting and tokenization standards, we followed the English punctuation standards referenced from The Blue Book of Grammar and Punctuation (Straus, 2008). After parsing and obtaining the terms, we filter out the non-Stopwords (see Appendix II) then apply Porter's stemming algorithm to obtain the stem of each valuable term. Appendix II Porter Stemming Algorithm Diagram provides the full

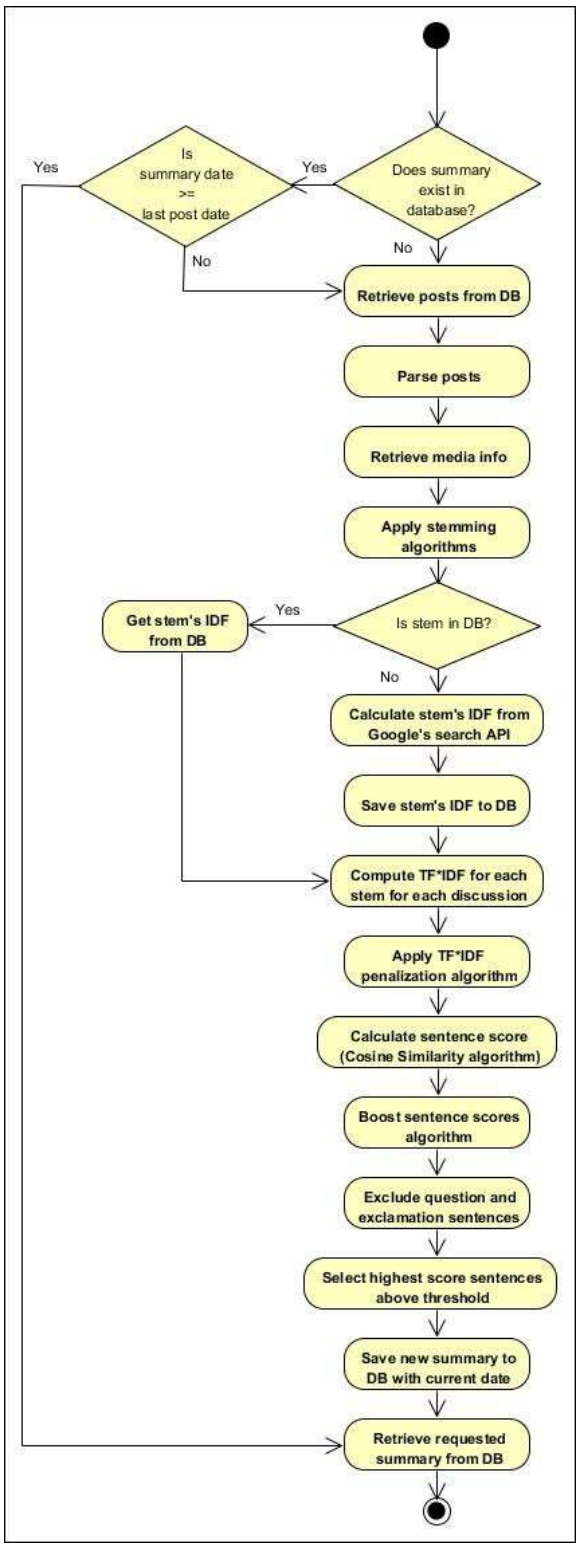

Figure 3.16: Discussions summarization activity diagram

stemming flowchart for Porter's algorithm. These terms will be used in further NLP tasks. Going back to our example, as a result of the parsing and stemming algorithms, we exclude Stopwords like an, that, none…etc. Then we stem the remaining vocabularies using Porter's Stemming algorithms. For example: the stem of "advocacy" is "advocaci", "political" is "polit", and "funded" is "fund", etc. (see full stemmed terms for discussion (10183) in Table 3.3).

Table 3.2: Document/Discussion (10183) segmented sentences

#	Sentence
S1	An advocacy group that is non-political and not funded by oil companies, gives it a degree of legitimacy that others who oppose global warming do not have
S2	With the focus on the search for the truth, the lecturer allows for more openness and free thinking
S3	The world needs people who believe and are passionate
S4	I have no problem with activists, climate activists or environmentalists - on the contrary - if it wasn't for some of these people, but - you have to consider people's vested interests and take precautions to make sure those interests don't get in the way
S5	If someone has a habit of dishonesty of course one would be foolish to take their word
S6	Likewise if someone is respected for their honesty and diligence, I'm more likely to take them at face value
S7	I am always skeptical, and if one investigates further and finds that the asserter hasn't published the relevant work (or in this case hasn't published anything for 10 years), makes assertions that are not supported by any evidence, and upon further investigation, finds that the assertions are actually directly contradicted by real world evidence, then it would be foolish not to discount the assertions

Table 3.3: Document/Discussion (10183) stemmed tokens for each sentence

#	Stemmed Tokens
S1	advocaci, group, polit, fund, oil, compani, degre, legitimaci, oppos, global, warm
S2	focus, search, truth, lectur, open, free
S3	world, peopl, passion
S4	problem, activist, climat, environmentalist, contrari, wasnt, peopl, vest, interest, precaut, make, dont
S5	habit, dishonesti, foolish, word
S6	likewis, respect, honesti, dilig, face
S7	skeptic, investig, find, assert, hasn, publish, relev, work, case, year, make, support, evid, direct, contradict, real, world, foolish, discount

Figure 3.17 below is the code snippet for the tokenization function

```
1.  public static string[] Tokenizes(string text)
2.  {
3.     // Strip all HTML line 4-6.
4.     text = Regex.Replace(text,
       @"</?\w+((\s+\w+(\s*=\s*(?:"".*?""|'.*?'|[^'"">\s]+))?)+\s*|\s*)/?>",  "");
5.     text = Regex.Replace(text, @"(^\.)|((http?)|www)([^""'  ]+)", "");
6.     text = Regex.Replace(text, "<[^<>]+>", "");
7.     text = Regex.Replace(text, "[0-9]+", ""); // Strip numbers.
8.     text = Regex.Replace(text, @"(http|https)://[^\s]*", ""); // Strip urls.
9.     text = Regex.Replace(text, @"[^\s]+@[^\s]+", ""); // Strip email addresses.
10.    text = Regex.Replace(text, "[$]+", ""); // Strip dollar sign.
11.    text = Regex.Replace(text, @"@[^\s]+", ""); // Strip usernames.
12.    text = Regex.Replace(text, @"(\n)|(\t)|(\r)", ""); // Strip new line & tab chars
13.    // Tokenize and also get rid of any punctuation
14.    return text.Split(" @$/#.-:&*+=[]?!(){},''\">_<;%\\- ".ToCharArray());
15. }
```

Figure 3.17: Tokenization function code snippet

- **Media Info retrieval:** We retrieve the title, description, and closed caption for the YouTube video to be used in our NLP algorithms. We retrieve both the title and description from the feeds API during the summarization process and the closed caption from the YouTube timedtext API. The retrieved media info stored in a (*VideoInfoDocument*) object as part of the corpus (*DFCorpus*) class (see Figure 3.18 below for more details).

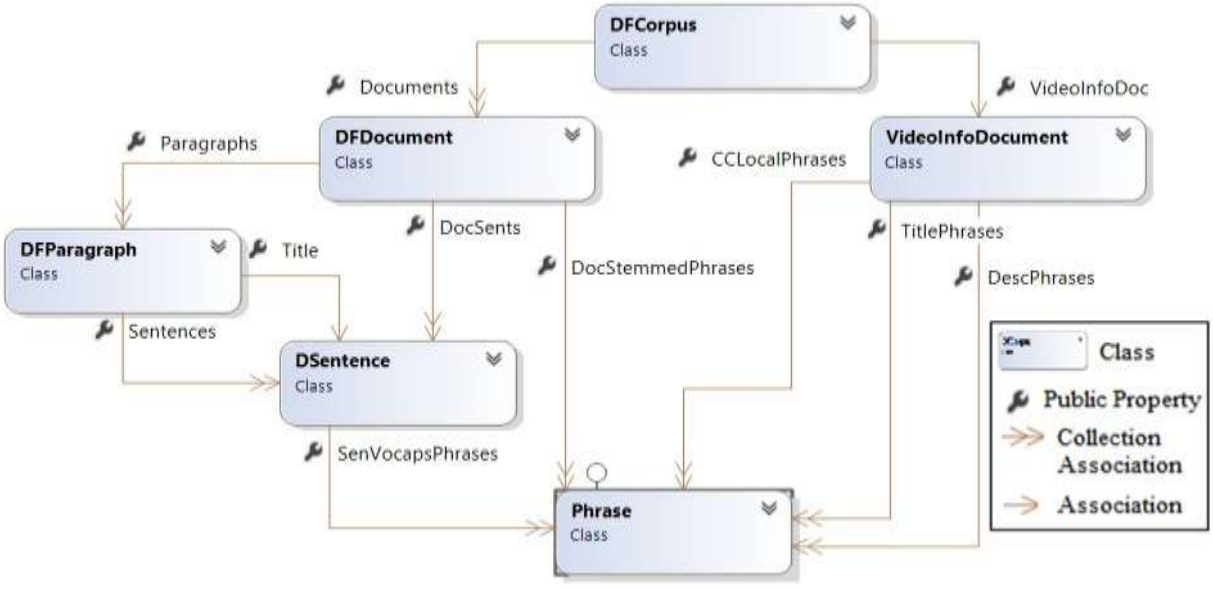

Figure 3.18: NLP parser class diagram

The above diagram in Figure 3.18 illustrates the class diagram of the NLP Parser within the NLP API sub application. The main class (*DFCorpus*) can contain one video information document (*VideoInfoDocument*) instance and a list of discussions (*DFDocument*) documents. The Video information document contains three lists of phrase instances: *TitlePhrases*: contains phrases of the title; *DescPhrases*: contains phrases of the description; and *CCLocalPhrases*: contains phrases of the closed caption around the time of the discussion annotation on the video time line, which represents the context in the video of the annotated discussion. On the other hand, each discussion document (*DFDocument*) contains a list of paragraphs (*DFParagraph*) represented in on post/thread in a discussion, list of sentence (*DSentence*) as a result of the segmentation, and a list of terms (*Phrase*) as a result of the tokenization step. A paragraph (*DFParagraph*) also contains a title sentence and a list of segmented sentence (*DSentence*). Finally each sentence (*DSentence*) is tokenized into phrases (*Phrase*).

DFCorpus Class: Responsible for storing an object of the list of documents. also it is responsible for executing the main summarization algorithm. This class contains the summarization algorithm methods (lines 19 to 25) in addition to the corpus initialization step in line 1 as shown in Figure 3.15.

DFDocument Class: Mainly responsible for storing instances of paragraphs, sentences, and phrases for each discussion as one document and calculating the Term Frequency (TF) of each document.

DFParagraph Class: Mainly responsible for segmenting the paragraph into sentences and storing them in a list of DSentence instance object. An example of segmented paragraph into sentences is demonstrated in Table 3.2.

DSentence: Responsible for tokenizing each sentence in to phrases (instances of DPhrase). An example of tokenized sentences into terms/phrases is demonstrated in Table 3.3.

DPhrase Class: Responsible for storing the term, stemmed version of the term – by calling Porter Stemmer tool, and storing the TF-IDF value for the term, which will be calculated further in the algorithm. An example of stemmed tokens is demonstrated in Table 3.3.

- **Compute IDF:** As pointed in Chapter 2, TF-IDF is one of the main techniques that provide an indication of the relevant importance of a term to a document within a large document context. The following equation (3.1) is the basic equation for the TF-IDF as defined by Jurafsky and Martin in (Jurafsky & Martin, 2000) and (Manning, Raghavan, & Schütze, 2009):

$$\text{[illegible]} = \text{[illegible]} * \text{[illegible]}_{\text{[illegible]}_t}^{N} \tag{3.1}$$

Where:

[illegible]$_{td}$: [illegible]
[illegible]$_{t}$: [illegible]occurrences [illegible]
[illegible] [illegible]
[illegible]$_t$: [illegible]

The first step we apply from the above equation is calculating the Inverse Document Frequency ($\text{IDF} = \log\frac{\text{N}}{\text{n}_\text{t}}$); this step is also referred to as "training the system" where the training dataset (corpus) is used to compute the IDF value for all terms in the corpus. Another variation of the IDF equation is

$$\text{IDF} = \log\frac{\text{N}}{1 + \text{n}_\text{t}} \tag{3.2}$$

One is added to the denominator of the log to avoid division by zero (Becker, 2013). To calculate the IDF, there is usually training dataset/corpora that containing a large number of documents from a wide range of topics. Although this approach has been mostly the chosen technique, we chose to go with an open approach and use the entire web as our corpus. We treat each web page as a document and perform a search to get the number of pages contains the term we used for that Google Web Search API, which returns the result count, and the number of pages, as part of the search result list. Knowing the number of Google's indexed web pages, Figure 3.20 below (Kunder, 2014), is approximately 47 billion at the time of performing the search, we were able to apply equation (3.2) in the following pseudo code to compute the IDF value for each term.

```
1.   Calculate IDF Values() {
2.      Get Phrases List for this (current) document     // Each document has Doc Stemmed phrases list
3.      For each Phrase in Phrases List {                // Loop through the Phrases list
4.         If exist in database then                     // Check if we already have the Phrase in the DB
5.            Get IDF value from database        // Retrieve IDF value form DB, it was calculated before
6.         Else                            // Otherwise Make a call to Google's Web search API
                                                passing the Phrase as parameter to search for.
7.            Call Google Web Search API (search for Phrase) and get the result count.
8.            Compute IDF using the following equation ??? = ???(N/(1+???))     // Compute IDF
9.            Save Phrase and IDF value to the database  // Save the Phrase and its IDF to the DB
10.        End If
11.     }
12.  }
```

Figure 3.19: Calculate IDF function pseudo code

Figure 3.20: Estimated size of Google's index (Kunder, 2014)

Using the Google search API, we find the result count for the "advocaci" and "polit" stems which are 2,350 and 1,200,000 respectively. If we use equation (3.2), the IDF value would be 16.8108173901 and 10.5755804905 respectively.

- **Compute TF & TF-IDF:** Next, we can compute the Term Frequency (tf = $f_{t,d}$) for each term within each document. The TF value can be normalized with different methods (Manning, Raghavan, & Schütze, 2009). In our implementation, we normalize the term frequency by the length of the document. The document length is defined by the number of unique non-Stopwords/vocabularies in the document. Once we have both the TF and the IDF values, we can compute $w_{t,d} = tf * idf$. Therefore the final equation would be:

$$w_{td} = \frac{f_{t,d}}{\frac{d}{d_l}} * idf \frac{N}{1 + t} \quad (3.3)$$

Where:

$w_{t,d}$ = the weight of term t in document d

The full algorithm of the TF-IDF computation is described in Figure 3.21 below.

```
1.  Calculate TF*IDF() {
2.     Get Phrases List for this (current) document      // Each document has Doc Stemmed phrases list
3.     For each Phrase in Phrases List {                 // Loop through the Phrases list
4.        If Not Stopword then                           // Exclude Stopwords form TF*IDF Computation
5.           Get the Phrase frequency/count (TF) in the document/discussion
6.           Normalize Phrase frequency                  // Divide TF by document length
7.           Compute TF*IDF using the following equation        w_td = tf/d_l * idf
8.        End If
9.     }
10. }
```

Figure 3.21: Compute TF-IDF pseudo code

Referring back to our example, the term frequency of the "advocaci" term in this discussion/document is $tf = 1$. The length of the document is $d_l = 60$. Therefore the weight of this term is $w_{td} = 0.2601802898$. Similarly the term "polit" would have a $w_{td} = 0.1762596748$.

- **TF-IDF penalization:** One of the main enhancements that we provided to the basic summarization approach is to exploit the characteristics of the first component (JSMPW) to serve the second component during the summarization process. We apply this step to resolve one of the most challenging problems in Microtext summarization, which is topic drift, defined as a deviation of a thread in the discussion away from the main subject (Hobbs, 1990). This is very common in multi-threaded, multi-users chat, micro-blogs, and discussion

forums platforms. To reduce or eliminate this phenomenon, we modify the TF-IDF equation

in (3.3) to include the normalized term frequency of the closed caption of the video lecture d_{cc} see equation (3.4). After we find the term frequency for each term in the closed caption, we normalize it by the maximum term frequency (Manning, Raghavan, & Schütze, 2009). Then we multiply it by the TF-IDF value from equation (3.3). This way we insure to exclude – from the discussion – terms that are out of context or not directly related to the video topics. Furthermore, the normalization step by the max term frequency scales the discussions terms based on their frequency in the closed caption; the maximum term will hold its discussion TF-IDF value and the rest would be scaled down. This normalization technique allows the main topic to stand out in the discussion. The following is the final equation for the term weight ($w_{t,cc,d}$):

$$w_{t,cc,d} = \frac{tf_{t,cc}}{\max(tf_{t,cc})} * \log\frac{N}{1+df_t} \quad (3.4)$$

Where:

$tf_{t,cc}$: the number of occurrences of term t in the closed caption of document d.

df_t: the number of documents containing term t in the whole corpus.

N: the total number of documents in the corpus.

The TF-IDF penalization algorithm is described in Figure 3.22 below.

```
1.  Penalize TF*IDF() {
2.     Get CCPhrases List for the Closed Caption (CC) in the Video Info document //Stemmed phrases list
3.     Get the MAX CCTF for CCPhrases List
4.     Get Phrases List for this (current) document      // Each document has Doc Stemmed phrases list
5.     For each Phrase in Phrases List {                 // Loop through the Phrases list
6.        If Phrase exist in CCPhrases then              // Exclude StopWords form TF*IDF Computation
7.           Get CCPhrase TF_tcc from CCPhrases List     // The Term count in the closed caption
8.           Normalise TF_tcc by MAX CCTF
9.           Compute new weight using w_t,cc,d = tf_t,cc / max(tf_t,cc) * tf_t,d * log N/(1+df_t)
10.       Else
11.          The new weight w_t,cc,d = 0                 // Since TF_tcc = 0 then pwt,cc,d = 0
12.       End If
13.    }
14. }
```

Figure 3.22: Penalize TF-IDF pseudo code

From our previous example, the closed caption did not include the first term stem "advocaci" so the value for $tf_{advocaci} = 0$. This will make the weight of the term set to zero indicating that this term is not relevant to the video. Hence, penalization prevents topic drift. On

the other hand, the stem "polit" did exist in the caption, $f_{stem} = 4$, and the maximum term frequency in the caption is $\max(f_{stem}) = 48$. Therefore, the penalization of the stem = 0.0833333333. Hence, $w_{stem} = 0.0146883062$.

- **Calculate sentence's score:** To calculate the sentence score, we adopted the cosine similarity algorithm, within the Vector Space Model (VSM), where we calculated the cosine similarity for each sentence and every other sentence in the document by multiplying the unit vectors of the sentences as in the equation (3.5) (Manning, Raghavan, & Schütze, 2009):

$$sim(d_1, d_2) = \vec{v}(d_1) \cdot \vec{v}(d_2) \quad (3.5)$$

Figure 3.23 illustrates a simplified two dimensional graph of two terms (gossip and jealous) in three documents and search query. Documents are represented as unit vectors where its nodes are the weights of each term.

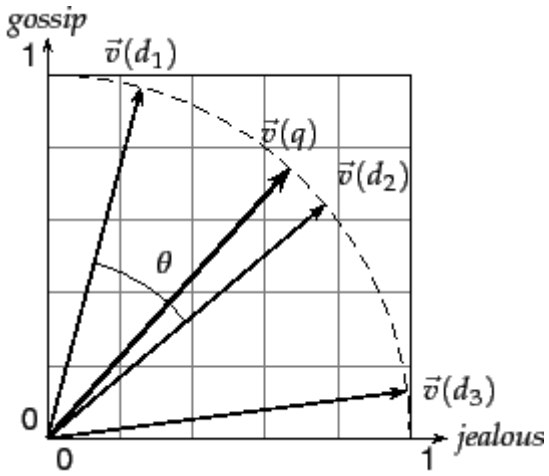

Figure 3.23:Cosinesimilarityillustration.sim(d1,d2)=cos0(Manning, Raghavan, & Schütze, 2009)

Once we have the similarity of every two sentences, we sum the total similarities for each sentence as the following equation (3.6):

$$\text{score}(S_i) = \sum_{0}^{n} sim(d_1, d_2) \{k \neq i, 0 < [k, i] < n \mid i, k, n \in N\} \quad (3.6)$$

Where:

S_i: is the i_{th} sentence $\{i , 0 < i \leq n\}$
n: is the number of sentences

From our example, after we apply the above equation (3.6), the following Table 3.4 demonstrates the cosine similarity between every two sentences in the document:

Table 3.4: Cosine similarity of document (10183)

	S1	S2	S3	S4	S5	S6	S7	Sentence Score ∑
S1	0	0	0	0	0	0	0	0
S2	0	0	0	0	0	0	0	0
S3	0	0	0	0.0379324517	0	0	0.3738636656	0.4117961174
S4	0	0	0.1517298069	0	0	0	0.1066812853	0.2584110923
S5	0	0	0	0	0	0	0	0
S6	0	0	0	0	0	0	0	0
S7	0	0	0.3738636656	0.1066812853	0	0	0	0.4805449510

- **Boost sentence score:** In this step, we again exploit the JSMPW component annotation time of each discussion to obtain the local closed caption, which is the closed caption surrounding the discussion annotation time. The local closed caption represents the context of the discussion in the video. Then we apply the cosine similarity and add the $\mathrm{sim}(s_i, \mathrm{local}\ s_{cc})$ value to the sentence score. In addition to the local closed caption, we also apply the same method to the media title sentence and each sentence in the description. Therefore, the final sentence score equation would be as follow:

$$\mathrm{BScr}(S_i) = \text{[illegible]} + \text{[illegible]} + \sum_{k=0}^{n} \text{[illegible]} \quad \{k \neq i\} \qquad (3.7)$$

Where:

$\mathrm{local}\ s_{cc}$: the local closed caption around the annotation time

s_{title}: the media title sentence

s_x: the x_{th} sentence in the description

n_{desc} : the number of sentences in the description

The full algorithm of the boosted sentence score method is shown in Figure 3.24 below.

```
1.  Boost Sentence Score() {
2.     Tokenize Title from Video Info Document object          // use the same RegEx tokenizer
3.     Tokenize description from Video Info Document object
4.     Get document's annotation time          // annotation time in the video from the JSMPW
5.     Get local/context closed caption surrounding annotation time
6.     Tokenize local closed caption
7.     For each sentence in the document {
8.        // Compute the Cosine Similarity between the current sentence in the loop and video
          information document sentence (title, description, and Local closed caption) and add to the
          sentence score calculated in the previous step.
9.        Sentence score += Compute cosine similarity between sentence & title
10.       Sentence score += Compute cosine similarity between sentence & description
11.       Sentence score += Compute cosine similarity between sentence & Local closed caption
12.    } // End for each Loop
13. }
```

Figure 3.24: Boost sentence score pseudo code

The discussion sample in Table 3.1 was posted at second 36 of the video time frame. Based on a configuration value ANNOT_BOUNDRY that determines the number of seconds surrounding the annotation time, the application extracts the caption as one sentence and refers to it as local closed caption sentence ☐☐☐☐ ☐☐. Looking at the sample closed caption in Appendix
IV, we can get the sentences starting from second "12.469" to "59.219" inclusive having ANNOT_BOUNDRY=25 seconds. Then we apply equation (3.7) for each sentence in the document to calculate the Cosine Similarity with the local closed caption, the title of the video, and every sentence in the description. In our example, Table 3.5 shows the difference in the sentence score before the boost (column 1) and after the boost (column 2).

Table 3.5: Comparison between initial score & boosted score

	☐☐☐☐☐☐☐(☐☐) from EQ	☐☐☐☐☐☐☐(☐☐) from
S1	0	1.2642248821
S2	0	0.0580768940
S3	0.4117961174	0.4117961174
S4	0.2584110923	0.9215403919
S5	0	0
S6	0	0
S7	0.4805449510	0.5854839465

- **Exclude sentences:** Question or exclamation-type sentences are valuable in the scoring schema, however they are not valuable in a summary of the discussion. Therefore, we excluded these sentences from the list of potential summary sentences. We achieved this by negating their scores. In our example, the sentences were neither question nor exclamation sentences, so we did not exclude any. Assuming we had one sentence ended with a question mark "?" and had a boosted score $\mathrm{BScr}(S_i) = 0.5436200343$, the new boosted score would be: $\mathrm{BScr}(S_i) = -0.5436200343$. Therefore, this sentence would not be selected for the summary, as it is not above the threshold.
- **Sentence selection for summary:** The final step is to select the average similar sentences, which are going to have the highest Boosted Score (BScr) as in equation (3.7). We select only sentences that are above the specified threshold. The threshold is defined in a configuration file as a percentage above the average BScrs. Once the best scored sentences are selected, we save these sentences to the database and then the summary view page retrieves this information from the database once it is triggered by the summary controller. From the discussion posted in the example above, we calculate the average sentence score = 0.648224446. Based on a 25% threshold, the minimum boosted score selected is $\mathrm{BScr}(S_i) = 0.8102805580$. As a result we have only S1 and S4 as part of the summary for this document/discussion refer to Table 3.6 below.

Table 3.6: Extracted sentences as summary of document/discussion (10183)

#	Sentence
S1	An advocacy group that is non-political and not funded by oil companies, gives it a degree of legitimacy that others who oppose global warming do not have
S4	I have no problem with activists, climate activists or environmentalists - on the contrary - if it wasn't for some of these people, but - you have to consider people's vested interests and take precautions to make sure those interests don't get in the way

The above steps are conducted in iterations equal to the number of documents/discussions posted as annotations on the video.

3.2.4 Database Layer

As part of the JSMPW installation to Moodle's platform, we create new tables and add them to the Database used by the server side component Summarization Application. We add four new tables: Annotation, URLParams (helper table), MediaInfo, TermsIDF table. To add new tables to Moodle's database as part of a new module installation, the table schema must be presented in an XML format. Then, the PHP installation code generate a create table statement meets the database requirement and executes the statement to create the new table. The following code snippet in Figure 3.25 is the Annotation table schema in an XML format.

```
1.  <TABLE NAME="jsmp_annotations" COMMENT="Defines jsmediaplayers">
2.    <FIELDS>
3.      <FIELD NAME="id" TYPE="int" LENGTH="8" NOTNULL="true"
        SEQUENCE="true"/>
4.      <FIELD NAME="annid" TYPE="int" LENGTH="8" NOTNULL="true" DEFAULT="0"
        SEQUENCE="false"/>
5.      <FIELD NAME="anntype" TYPE="char" LENGTH="50" NOTNULL="true"
        DEFAULT="0" SEQUENCE="false"/>
6.      <FIELD NAME="annaction" TYPE="char" LENGTH="50" NOTNULL="true"
        DEFAULT="0" SEQUENCE="false"/>
7.      <FIELD NAME="anntitle" TYPE="char" LENGTH="50" NOTNULL="true"
        DEFAULT="0" SEQUENCE="false"/>
8.      <FIELD NAME="anntime" TYPE="int" LENGTH="8" NOTNULL="true"
        DEFAULT="0" SEQUENCE="false"/>
9.      <FIELD NAME="mediaid" TYPE="char" LENGTH="255" NOTNULL="true"
        SEQUENCE="false"/>
10.     <FIELD NAME="mediaduration" TYPE="int" LENGTH="8" NOTNULL="true"
        DEFAULT="0" SEQUENCE="false"/>
11.   </FIELDS>
12.   <KEYS>
 13.     <KEY NAME="primary" TYPE="primary" FIELDS="id"/>
14.   </KEYS>
15.   </TABLE>
```

Figure 3.25: MFMCS annotation table schema in XML format for Moodle installation

Figure 3.26 is a Database Diagram which demonstrates the four new tables, Moodle's existing forums table, and their relationships.

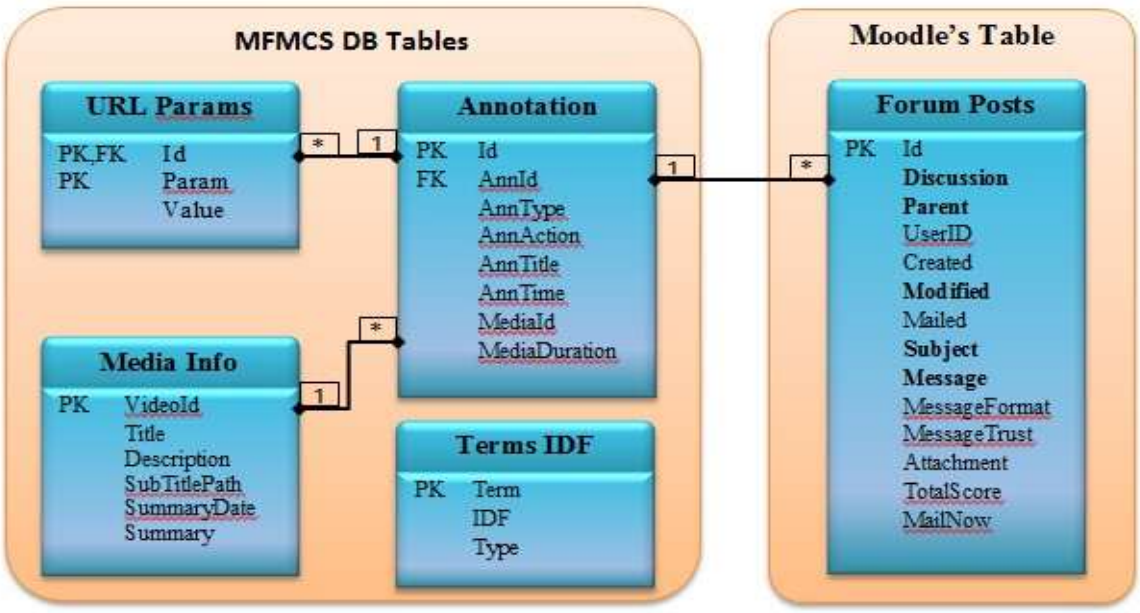

Figure 3.26: JSMPW & Summarization application DB tables

- **Forum Posts:** An existing Moodle table that holds the discussion forum data/posts. The main columns that we use are: Discussion (the ID of the main post), Parent (the ID of the parent post), Modified (the last modified data), Subject (the title of the post), and Message (the actual post text). Each record in this table represents a post/thread in a single discussion. Hence, one discussion ID would have many records (posts).
- **Annotation:** Stores the new annotation's information such as: time of annotation reference to the media time line, ID of the post, type of the post (currently we have Discussion Forums (DF) type only), and other meta-information about the annotation. The AnnID is a foreign key to dissection/post ID in the Forum Post table that links each annotation with many posts from the forum table.
- **URL Params:** A helper table used to store the required parameters for each annotation discussion/post direct URL, to be able to load it in the modal dialog. One annotation record can have multiple parameters in this table.
- **Media info:** Holds the media information and the generated summary for all discussions on the media. For each media-information record, there is many annotations linked by (MediaInfo.VideoID=Annotation.mediaID).
- **Terms IDF:** Used to store the IDF value for each term in the discussions. Its main purpose is to avoid redundant IDF computations and to call to Google's search API, especially when there is a cap on the Google API calls.

The database tables are accessed in the .NET framework as part of the ADO.NET Entity Data Model. After establishing a connection to the database server, we add an ADO.NET Entity Data Model to the application and then map the database tables to newly-created .NET entity classes, one for each table needed by the application. The class contains mapped database columns as properties, and can be accessed like any .NET property in a regular class. Figure 3.27 is a code snippet of the Annotation class representing the Annotation table in addition to its relationship with the URLParam table. Each Annotation record can have many URLParam records.

```
1.  public partial class Annotation
2.  {
3.      public Annotation() {
4.          this.urlparams = new HashSet<URLParam>();      // Instantiation of a collection
5.      }
6.      public int id { get; set; }
7.      public int annid { get; set; }
8.      public string anntype { get; set; }
9.      public string annaction { get; set; }
10.     public string anntitle { get; set; }
11.     public int anntime { get; set; }
12.     public string mediaid { get; set; }
13.     public int mediaduration { get; set; }
14.     public virtual ICollection<URLParam> urlparams { get; set; } //Collection of URLParam
15. }
```

Figure 3.27: Code snippet of the annotation class model

Once we create the ADO.NET Entity Data Model, we use *LINQ to Entity* to access the data in the database represented by the class. A sample LINQ to Entity call is demonstrated in Figure 3.28, where the code returns the selected posts for the specified videoID that matches the searchQuery from the joined tables *ForumPosts* and *Annotation* on both *videoID* and *annid*.

```
1. return db.ForumPosts
2.       .Where(p => db.Annotations.Any(e  => e.mediaid == videoID && p.discussion == e.annid))
3.       .Where(r => r.message.Contains(searchQuery) || r.subject.Contains(searchQuery));
```

Figure 3.28: LINQ to Entity code snippet sample

In conclusion, we have presented in this chapter our MFMCS system as a new added value tool to MOOC platforms. The MFMCS system resolves the learning distraction represented by switching tools, and resolves the challenges by linking the discussion tools to the main course content. Also, this resolves the massive amount of shared knowledge by providing a modified summarization algorithm solution that summarizes the discussions shared on a particular video content in the context of the video content itself.

In the next chapter will focus on the experiment setup, discuss different evaluation criteria used to evaluate our results compared to other systems, conclusion, and finally future work.

Chapter 4

Evaluating the MOOC Video Summarization Capability

In this chapter we will describe the evaluation of the MFMCS system. We will start by a brief description of the experimentation setup. Then follow by discussing the evaluation system and criteria. Finally demonstrate the evaluation results compared to other summarization systems, mentioned in chapter 2.

4.1 Experimentation Setup

After setting up Moodle LMS we installed the JSMPW. Next, we created a course titled "*Global Warming Facts*". The course consisted of one video lecture based on existing YouTube video titled "*Why "Global Warming" Failed & Why Climate Change is Real*"[42]. We invited 25 students to register in the course and participate in the discussion to get their feedback and experience about the integration of the discussion forums and the main content video lecture. Eighteen students enrolled in the course and only seven were actively engaged in the discussion forums about the video. After the discussion was completed, we asked four persons to watch the video and summarize the discussions posted online using the extraction summarization technique. The summarization was based on extracting a full or partial sentence from the discussions as long as the sentence is related to the video. For example, if the video discusses Global Warming facts but mentions nothing about its economic issues, then the summary should not include any sentence about Global Warming economic issues. These four summaries are considered ideal summaries. Next, we evaluated the automated summary – generated by the MFMCS system – by comparing it to these human (ideal) summaries.

4.2 Evaluation Criteria

The first component of the MFMCS system is the JSMPW component, which is responsible for integrating the discussion forums with the video content/course lecture of the MOOC platform. We evaluated this component based purely on user's feedback after using the tool.

On the other hand, we evaluated our MFMCS extracted summary using more systematic measures, utilising the Recall-Oriented Understudy of Gisting Evaluation (ROUGE) system. A

[42] https://www.youtube.com/watch?v=5c4XPVPJwBY

full description of the software is available on the ROUGE website[43]. We have requested to download the software form the available download ROUGE request form on the same ROUGE website. The request was sent to the author Chin-Yew Lin, who in turn sent us a download link for the software, and we downloaded the latest available version, ROUGE-1.5.5. The system provides an automated evaluation of system generated (candidate) summaries compared to human written (ideal) summaries. ROUGE evaluates summaries based on five different evaluation metrics (Lin, 2004):

1. **ROUGE-N**: Where N value between 1 and 9. ROUGE computes matching n-grams between candidate and ideal summaries. This metric favors matching more shared grams between the ideal summaries.
2. **ROUGE-L**: Similar to ROUGE-N with the exception of having the flexibility of the gram size, the value of N. In this metric ROUGE matches the Longest Common Subsequence (LCS) between the two summaries. Then computes the length ratio of the overlapped LCS to the ideal summary's length. One advantage of this metric is that no consecutive LCS matches are required, yet a term in-sequence is a must, just like ROUGE-N.
3. **ROUGE-W**: Weighted Longest Common Subsequence metric was introduced to evaluate consecutive matches of LCS in ROUGE-L.
4. **ROUGE-S**: This is a Skip-Bigram co-occurrence statistics that evaluates a set of any pair of terms matching within their sentence. ROUGE-S measures the Skip-Bigram overlaps ratio between the candidate and the ideal summaries. For instance, an ideal sentence "*greenhouse effect on the environment*" has a total number of maximum Skip-Bigrams C(5,2)[44] = 10. The following Bigrams: {*"greenhouse effect", "greenhouse on", "greenhouse the", "greenhouse environment", "effect on", "effect the", "effect environment", "on the", "on environment", "the environment"}*.
5. **ROUGE-SU**: Similar to ROUGE-S with the addition of including Unigram in the case its second gram does not appear in the sentence.

ROUGE evaluates system generated summaries (candidates) based on provided human summaries (ideals). The ROUGE system has proved its validity of its evaluation method in the Document Understanding Conference (DUC) data. To determine the quality of any candidate,

[43] http://www.berouge.com/Pages/default.aspx
[44] Combination: C(5,2) = 5!/(3!*2!) = 10.

summary ROUGE compares the candidate summary with an ideal summary by counting the overlapping terms between the two (Lin, 2004). By the name ROUGE uses the Recall measure to evaluate summaries. However, the latest version of ROUGE (starting from version ROUGE-1.5.1 and up) included two additional measures: Precision and F-Score. Based on the author revision notes of the software, these where added to deal with different candidates and ideal summaries lengths.

The Recall measure demonstrates how well the automated summary retains valuable information from the original document based on the human ideal summaries. In more technical terms, Recall is the percentage of the overlap units between candidate and ideal summaries out of ideal's summary sentences. The Precision measure, on the other hand, weighs the valuable information retained within the automated summary. Precision represents the percentage of the overlap units between system (candidate) and human (ideal) summaries out of the candidate's summary sentences. Finally, the F-Score is the weighted harmonic mean that combines the two measures that assesses trade-off between Recall and Precision (de Oliveira, Torrens, Cidral, Schossland, & Bittencourt, 2008) (Bysani, 2010).

The focus on Recall or Precision measures varies from one application to another. For instance, in the summarization context if we are more interested in presenting a shorter correct summary, we focus on high precision (the ratio of matches to the total candidate summary, in other words "of the things we are selecting, what % is correct?"). While if we need to present a more correct summary (relevant to the ideal summary), then we look for a higher recall (the ratio of matches to the total ideal summary, in other words "of the correct items, what % did our system select?"). For the summarization applications in general and our MFMCS summary in particular, the Recall measure is the most informative measure as presented in (Lin & Hovy, 2003). Table 4.1 demonstrates the two-by-two contingence table for the computation of precision and recall. To compute Precision, we consider the number of correctly-selected sentences that are in the ideal summary and selected by the system (TP,) and the incorrectly-selected sentences by the system (candidate) summary that are not in the ideal summary (FP). See equation (4.1) below. To compute Recall, we consider the number of correctly-selected sentences that are in the ideal summary and selected by the system (TP), and the sentences that are in the ideal summary but were falsely not selected by the system (candidate) summary (FN). See equation (4.2) below (Olson & Delen, 2008).

Table 4.1: 2 by 2 Sentence contingence table

Sentences	In Ideal Summary	Not in ideal Summary
In System Summary	True Positive (TP)	False Positive (FP)
Not in System Summary	False Negative (FN)	True Negative (TN)

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \tag{4.1}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{4.2}$$

Finally, based on the application needs, we favor the Recall measure, Precision, or we trade-off one on the cost of the other. The F-Score can provide us with the evaluation score accounting this trade-off. Equation (4.3) is the F-score equation based on both Precision (P) and Recall (R) values. The final version of ROUGE (version 1.5.5) computes all three measures with a default (α = 50%) trade-off for the F-score (Olson & Delen, 2008).

$$\text{F} - \text{score} = \frac{1}{\alpha\frac{1}{P} + (1-\alpha)\frac{1}{R}} = \frac{(\beta^2+1)PR}{R^2 P + R} \tag{4.3}$$

For example, if we consider the previous sentence in ROUGE-S5 metric, "greenhouse effect on the environment" as an ideal sentence, we then have a candidate summary sentence "greenhouse bad effect environment" which has a C(4,2) = 6. The matched bigrams are {"greenhouse effect", "greenhouse environment", "effect environment"}. Therefore ROUGE-S5 would have a Recall of (3/(3+7)) = 0.33, a Precision of (3/(3+3)) = 0.5, and finally an F-score of (α=.5)= 0.39.

In the next section, we will examine our results with these three measures from one or more ROUGE metrics. However, as mentioned earlier for the summarization system, we will first focus on the Recall measure to present a more correct summary (relative to the ideal summaries) rather than a shorter correct summary that could be missing a lot of the key sentences in the discussion, as we need to evaluate the summary by knowing the percentage of the correct

sentences retained from the ideal summaries. For example, if the human summary has 10 sentences, we need to know how many sentences the system selected out of these 10.In other words, we need to have a concise summary, and reduce the non-relevant sentences selected so we should mind the Precision measure in the evaluation. Therefore, we will examine the F-Score with a particular trade-off ratio. We chose (α=25%)[45] trade-off ratio in computing the F-Score.

To be able to use ROUGE, we first need to obtain the ideal/human summaries, and generate an extraction summary using the MFMCS system. Initially, we compared multiple samples of MFMCS system-generated summaries to determine the best configurations that provide the best summary results. Then we compared the best configured automated summary to other automated summaries generated by each of the following systems or online summarization tools:

1. Text Compactor Tool: based on Open Text Summarizer (OTS) algorithms[46]
2. The Dragon Toolkit: based on the LexRank algorithms[47]
3. The MEAD system[48]
4. The following online summarization systems: Automatic Text Summarizer[49], Free Summarizer[50], Online Summarization Tool[51], Sumplify[52]

Finally, we ran the ROUGE system using different ROUGE metrics to compare both our best MFMCS summary and the external online summaries against each of the human ideal summaries to evaluate our summary to other systems.

4.3 Evaluation Results

The JSMPW tool received very positive feedback as a new tool to integrate the discussion forums with the main course content (video lectures) within the MOOC and LMS platforms. We have contacted 25 users to participate in the course and join the discussion on the video. At the end of the course, we had seven students who were active on the course discussion with a total course enrolment of eighteen students. We invited all 18 students to send us feedback and comments about the MFMCS system in general and on the JSMPW integration tool in particular.

[45] Favor Recall α=0, favor Precision α=1.
[46] Text Compactor web page: http://www.textcompactor.com/ and the OTS page: http://libots.sourceforge.net/
[47] Dragons web page: http://dragon.ischool.drexel.edu/default.asp
[48] http://www.summarization.com/mead/
[49] http://autosummarizer.com/
[50] http://freesummarizer.com/
[51] http://www.tools4noobs.com/summarize/
[52] http://sumplify.com/

The feedback was all positive and encouraging of the new tool and idea. One of the user's testimonials about the new system was:

> *"I love this idea, very interesting and ideas like this are needed to pave the way for the future of education"*

The rest of feedback we received from other users where no less of a value, but shorter in words. The full list of feedback we received from all users is listed below:

"*Nice application*", "*I like the way you linked the discussion*", "*The Annotation seems nice*", "*Well done*",

For the summarization component results, we have generated 10 different summaries with 10 difference configurations using our MFMCS system. Then we evaluated them with different ROUGE metrics to be able to find the best configurations for the system that generate the best summary. Once we had the best summary, we again ran different ROUGE metrics to evaluate the MFMCS best summary with the other automatic online summaries to evaluate how our system performs against other available summarization systems. Figure 4.1 lists the configuration settings for one of the summaries. Lines 5 and 6 listed in red in Figure 4.1 are the main two configurations that impacted our MFMCS generated summary, providing the available dataset. Line 10 enables the TFIDF penalization to prevent drifted topics from appearing in the summary; and Line 11 enables sentence score boost using the closed caption context for each discussion in addition to the video title and description. We also have other configurations that could be useful for future use or larger datasets such as: DOC_SENT_THRUSHOLD and SENT_STEM_THRUSHOLD used for the minimum document size (minimum number of sentences) to be considered for summarization and the minimum sentence size (minimum number of stemmed terms) to be illegible for scoring. The NORMALIZE_XXX_TF switch is to enable/disable TF normalization.

```
1.  The configuration settings for the current run are:
2.  LOGMetrics:True
3.  DOC_SENT_THRESHOLD:2
4.  SENT_STEM_THRESHOLD:2
5.  SENT_SCORE_THRESHOLD:0.1
6.  ANNOT_BOUNDARY:30
7.  NORMALIZE_DOC_TF:True
8.  NORMALIZE_SNT_TF:True
9.  NORMALIZE_CC_TF:True
10. PENALIZE_TFIDF:True
11. BOOST_SENT_SCORE:True
```

Figure 4.1: Configuration settings

We have modified two main configuration settings to find the best summary settings: SENT_SCORE_THRUSHOLD, and ANNOT_BOUNDRY. We tested ANNOT_BOUNDRY using various values between 5 and 30 seconds. The result showed that having a closed caption of 30 seconds before and of 30 seconds after the annotation time is the best value for the annotation boundary. Therefore, we fixed the value to 30 seconds and changed the sentence score threshold value between 0.0% and 40% above the document's sentences average score. Figure 4.2 demonstrates two ROUGE metrics for the MFMCS summarization system, using both the basic scoring setting – as a baseline – without our contribution (in red), and the enhanced summarization settings after applying both TFIDF penalization and sentence score boost (in green). On the x-axis, we have the three measures Recall, Precision, and F-Score (having α=0.25, favoring Recall) for the seven different sentence score thresholds. On the y-axis we have the ROUGE score between 0-1. The enhancement outperforms the baseline scoring setting for all measures: Recall, Precision, and F-Score.

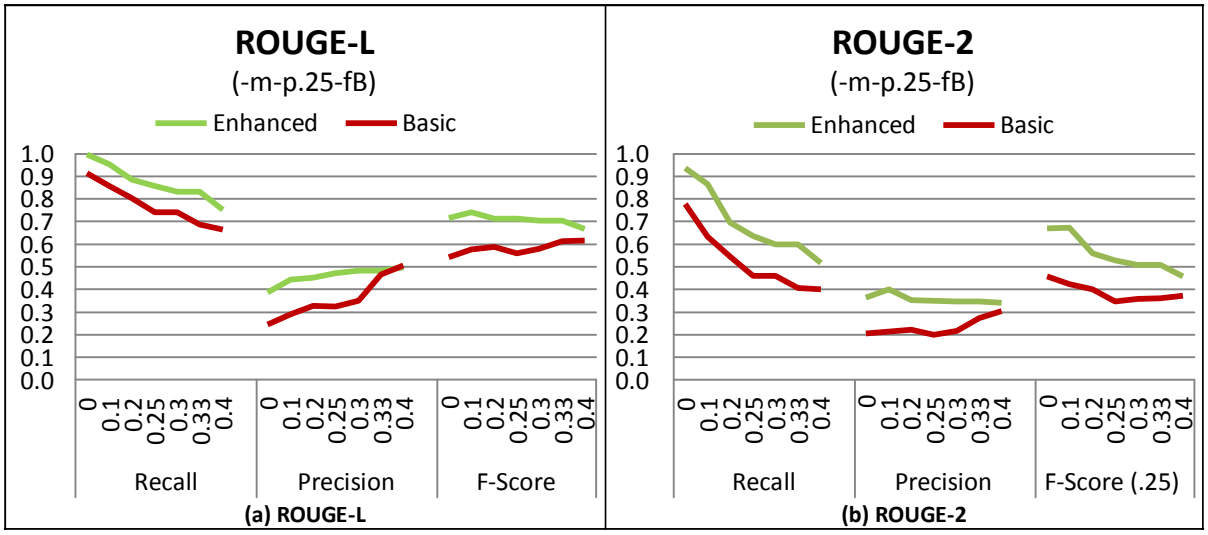

Figure 4.2: ROUGE measures for both MFMCS enhanced and basic settings

We notice in Figure 4.2 (a) and (b) above that we obtain the best Recall value when the threshold is set to 0.0%. Hence, we select all sentences above the document's average score.

To further evaluate the best MFMCS summary, we applied variations of ROUGE tests using different ROUGE configurations. Figure 4.3 below shows four different configurations for ROUGE measures:

(a): represents ROUGE measures evaluations after applying Porter stemming on both candidate and ideal summaries option (m), removing Stopwords option (s), and evaluating the metrics based on the best overlapping scores of all ideal summaries option (-f B). For example, the full ROUGE-L command is (*./ROUGE-1.5.5$RV.pl -e data -a -c 95 -t 0 -m -s -p .25 -f B Settings.xml*).

(b): represents ROUGE measures evaluations after applying Porter stemming on both candidate and ideal summaries option (m), removing Stopwords option (s), and evaluating the metrics based on the average overlapping scores of all ideal summaries option (-f A). For example, the full ROUGE-L command is (*./ROUGE-1.5.5$RV.pl -e data -a -c 95 -t 0 -m -s -p .25 -f A Settings.xml*).

(c): represents ROUGE measures evaluations for both candidate and ideal summaries including Stopwords and without applying any stemming algorithms, and evaluating the metrics based on the best overlapping scores of all ideal summaries option (-f B). For example, the full ROUGE-L command is (*./ROUGE-1.5.5$RV.pl -e data -a -c 95 -t 0 -p .25 -f B Settings.xml*).

(d): represents ROUGE measures evaluations for both candidate and ideal summaries including Stopwords and without applying any stemming algorithms, and evaluating the metrics based on the average overlapping scores of all ideal summaries option (-f A). For example, the full ROUGE-L command is (*./ROUGE-1.5.5$RV.pl -e data -a -c 95 -t 0 -p .25 -f A Settings.xml*).

Note that for all other options, we chose the default with the exception of option (-p) favoring the Recall when computing the F-Score with a trade-off factor α=0.25

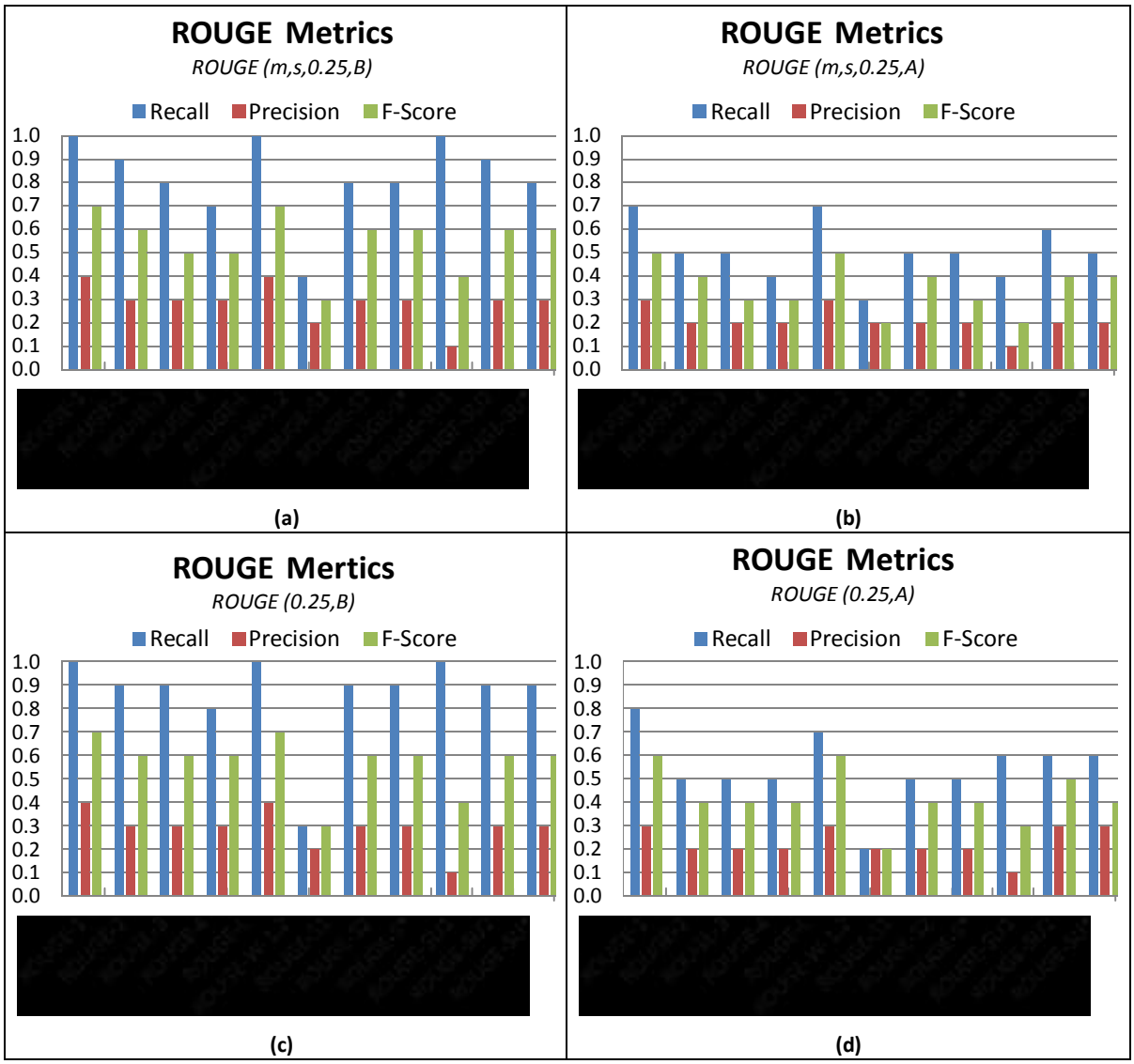

Figure 4.3: ROUGE metrics for the best MFMCS configuration summary

Even when computing the ROUGE measures using the average overlapping score of the candidate summary with all ideal summaries Figure 4.3 (b) & (d), our MFMCS enhanced summary performs at a high Recall score, about 0.78 with ROUGE-1 and ROUGE-L including Stopwords, without stemming Figure 4.3 (d).

We notice in Figure 4.3 (a, b, c, and d) that ROUGE metrics (on the x-axis) vary from one another, even though they all evaluate the same dataset. We also notice that they all fluctuate in the same pattern, even when we use different ROUGE configurations see Figure 4.4. The main fluctuation driver for these metrics is the dataset type, as Chin-Yew Lin concludes in (Lin, 2004). For example Lin found that for single document summarization ROUGE-2, ROUGE-L,

ROUGE-W, and ROUGE-S worked very well. While ROUGE-SU4 and ROUGE-SU9 evaluated very short and headline-like summaries prominently. For multi-document summarization, ROUGE-1, ROUGE-2, ROUGE-S*, and ROUGE-SU* performed reasonably well when removing Stopwords. For our dataset and based on the result shown in Figure 4.4, we found that ROUGE-1, and ROUGE-L worked very well.

Figure 4.4: Recall measure for the best MFMCS configuration summary using different ROUGE evaluation configurations

Figure 4.4 above demonstrates the Recall measure for the best MFMCS configuration summary using different evaluation configuration in ROUGE. On the x-axis, we have the different ROUGE metrics, and the y-axis contains the Recall measure. The legend contains the ROUGE configurations as follow (m= apply stemming, s= remove Stopwords, B= evaluate based on the best human summary match, and A= evaluate based on the average human summaries matches).

Finally, using the same discussion dataset and the MFMCS summary, we compared the performance of our MFMCS system against the three summarization systems discussed in Chapter 2: Dragon Toolkit[53], MEAD[54] system and OTS[55]. In addition, we included the following similar online summarization systems in our comparison: Automatic Text Summarizer[56], Free Summarizer[57], Online Summarization Tool[58], and Sumplify[59]. Figure 4.5 (a)

[53] http://dragon.ischool.drexel.edu/default.asp
[54] http://www.summarization.com/mead/
[55] http://www.textcompactor.com/ based on the OTS: http://libots.sourceforge.net/
[56] http://autosummarizer.com/

demonstrate how our MFMCS enhanced summary, marked in red, has a higher Recall score for all ROUGE metrics tested. Although our MFMCS-enhanced summary has an average Precision score compared to all other systems as in Figure 4.5 (b), Figure 4.5 (c) demonstrate a higher F-Score with (α=0.25). The drop in the Precision score in Figure 4.5 (b) indicates that our MFMCS-enhanced summary has a relatively larger summary with undesired sentences compared to the other systems. The discussions dataset and the full summary generated by the MFMCS system, all online systems, and all human summaries are available in Appendix V.

[57] http://freesummarizer.com/
[58] http://www.tools4noobs.com/summarize/
[59] http://sumplify.com/

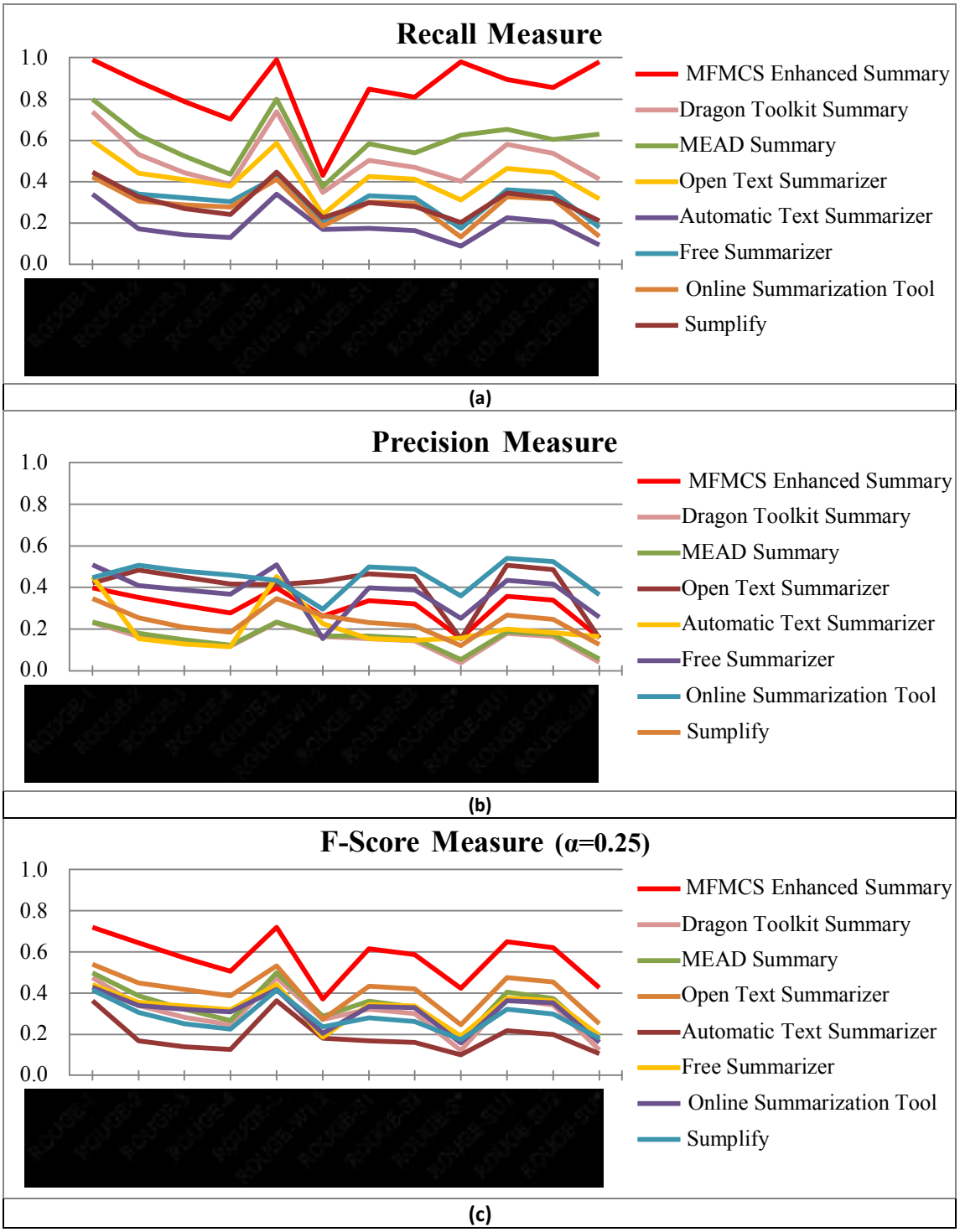

Figure 4.5: Comparing the MFMCS enhanced summary with 7 other summarization systems using different ROUGE metrics

This concludes our results chapter. In the final chapter, we will derive the thesis conclusion and suggest additional changes and enhancements to the MFMCS system as a future work.

Chapter 5
Conclusion and future work

5.1 Conclusion

We have discussed in Chapter 1 the main connectivism theory behind cMOOC, and how the knowledge is created by the users participating in MOOC. The implementation of the MFMCS system we have presented in this thesis supports the connectivism theory presented by Downes and Siemens back in 2003. We have created a new tool that integrates the main knowledge sharing and creation hubs to the main course contents delivery tools, which we demonstrated by integrating the discussion forums with the video lectures, where users are able to create new knowledge and link it to the main course content spontaneously. Part of the future work is to expand to full integration with all other tools and content formats. The JSMPW allows users to initiate a new discussion post at a specific time of the lecture video to associate the discussion with that moment of the video as an annotation. Then, it allows others to share their feedback and knowledge in the same discussion through their replies.

As we have received good feedback from the users who used the system, we consider that this implementation – from a cMOOC and pedagogical point of view – supports and encourages spontaneous knowledge-sharing and creation within the MOOC platform, and it is the missing piece of the current MOOC platforms. This is primarily because the JSMPW links users to both the main learning artifacts to any newly-created knowledge in the same page, and users don't have to toggle back and forth between different pages.

The second important fact about the JSMPW is its role as a building stone for the enhanced summarization algorithms we have designed – as a second component – in the summarization application. We designed an enhanced extraction summarization algorithm for the shared discussion posts. We utilized our first component (the JSMPW) to present a contextualised extraction summary for all discussion posts based on the video content and the time of the discussion within the video.

We based our summarization algorithm on some of the existing summarization techniques: stemming, Stopwords removal, TF-IDF computation, and finally sentence scoring based on cosine similarity. In addition, we utilized the closed caption of the video lecture and the annotation time of the discussion posts to eliminate topic drifts within the discussion, and boosted the sentences scores by the local closed caption of the discussion – around the

annotation time of the discussion – to present contextualized summary based on the video content.

Our summarization application showed improvement in the generated summary over the existing summarization systems tested in section 4.3. We have compared our automatically-generated summary to summaries generated by humans, and used the ROUGE system to evaluate the quality of our summary. As the results suggest in the previous chapter, the enhanced algorithm demonstrates improvement over the basic algorithms. Furthermore, the enhanced algorithms surpass some of the existing summarization algorithms currently available online, and that were tested in the previous chapter.

5.2 Future Work

For our initial component, we see the full integration of different knowledge-sharing tools such as Blogs, Wikis, and micro blogs, is essential. In addition, the integration should be compatible with all other course content formats and sources, such as documents, slides, audio, different video format (i.e. HTML5).

Within the NLP tasks, summarizations systems are still in their premature stage, and much room for improvement is there to explore. We have touched on one side of summarization tasks, the summarization of discussion forums within the educational environment. Yet, we can still enhance the system and the learning environment with one or more of the following future exploration areas:

1. Apply full integration between knowledge tools and course content.
2. Generate an abstraction summary.
3. Generate a custom interest-based summary, by allowing users to search for one or more keywords, with the result being a summary of the discussion related to the search query.
4. Create a rating system for the generated summary and retrain the system based on the user's ratings to provide a user preferred summaries based on user's profile.

Appendices

Appendix I

Evaluation Environment

The evaluation environment was prepared to host the MFMCS system as part of the online LMS Moodle. Initial environment and application setup are detailed in the following sub-sections.

Environment Setup

- **Windows Server 2012 R2:** We installed Windows Server 2012 R2 on one machine and used it to host both the Web Server and Database Server. Although we installed Ubuntu Server to set up other MOOC platforms such as edX and OpenMOOC, we did not use it, since none of these platforms was the choice of implementation.
- **SQL Server 2012:** The main database was managed using Microsoft SQL Server 2012. We used one Database instance and one application database. To connect to the database from the application, we used SQL Server authentication.
- **Web Server:** An IIS 8.5 was installed on the windows server to host the web and the API applications and linked it to the **lakeheadu.ca** domain with a subdomain **at5024-lumooc.**

Application Setup

- **Moodle platform:** As we pointed out in the conclusion of the previous chapter, Moodle was our platform choice to host our MOOC course, and we installed the latest version of Moodle 2.7 on our server. This installation included Moodle's core application and the additional plug-ins. Moodle platform was described in detail in section 2.1.1. The Moodle platform was implemented in The ***LUMM*** application folder in the ***Default Web Site*** and we used port 80 in IIS to be available on the World Wide Web. Moodle's Data folder was placed on a different drive in the server (G:\LUMM\MoodleData). This folder stores all the publicly-accessed files that are needed by the application.
- **Summarization Application:** The Summarization Application is the application we developed to handle the new pages that are rendered to Moodle main pages through the **I&K** wrapper; such as Search, Discussion, YouTube comments, and discussion summary pages. This application also contains the APIs developed for the annotations and NLP.

These pages and APIs are part of the ***AnnService Web Site,*** which is hosted on our IIS webserver and uses port 8080 to be accessed from the World Wide Web.

External Resources

- **JQuery libraries:** The main jQuery/JavaScript **JSMPW** file utilizes some of the existing jQuery plug-ins, in addition to the main jQuery library. These jQuery tools and plug-ins were used to enhance the users' experience when interacting with our system. Below are the list of JavaScript/jQuery plug-ins that were used in the JSMPW:
  - **jQuery.Class.js:** jQuery lightweight class-like implementation mimics classes in Object Oriented Programming (OOP) (Meyer, 2010).
  - **jQuery.Simplemodal.js:** jQuery lightweight plug-in that acts as a modal dialog framework. Provides a powerful modal dialog interface developments (Martin, 2013).
  - **jQuery.tools.expose.min.js:** a small functionality from the jQuery UI tools plug-in responsible for making HTML sections stand out from the surroundings elements (Three Dub Media, 2008).
- **YouTube APIs:** Several YouTube APIs were used in our application. But the most important YouTube API we used was the YouTube iframe Player API[60]. This API enabled us to embed YouTube video in an HTML iframe tag and control the video using JavaScript. The iframe API provides a great flexibility of serving as an HTML5 player when flash player is not supported by the browser. This API creates a YouTube JavaScript object and allows JavaScript to control the media playing. We also used the YouTube Feed API to retrieve YouTube video metadata such as title, description, current time, duration, and comments. Finally, we used YouTube timedtext API to retrieve the video closed captions and used it for further NLP summarization tasks. We padded the API calls with extra parameters to receive the data in JSON format to parse it into our .NET object within the Annotation API Application.
- **ASP.NET MVC:** MSVS provide developers with the ability to design their applications using the Model, View, and Controller (MVC) architecture. The MVC architecture comes with great benefits such as application layers segregation, code maintenance, and flexibility for expansion with new requirements.

[60] YouTube iframe API reference page: https://developers.google.com/youtube/iframe_api_reference

- **Model:** holds the business logic and application data, and is responsible for retrieving and updating the application database, processing the requests coming from the controller, and finally, notifying the view when there is a change in its data state.
- **View:** contains the user interface of the web application, receiving notifications from the model to update its instances and displaying it to the user.
- **Controller:** its main responsibility is to handle a user's requests and send it to the model. Figure 0.1 represents the standard MVC architecture as described in (Kanjilal, 2013).

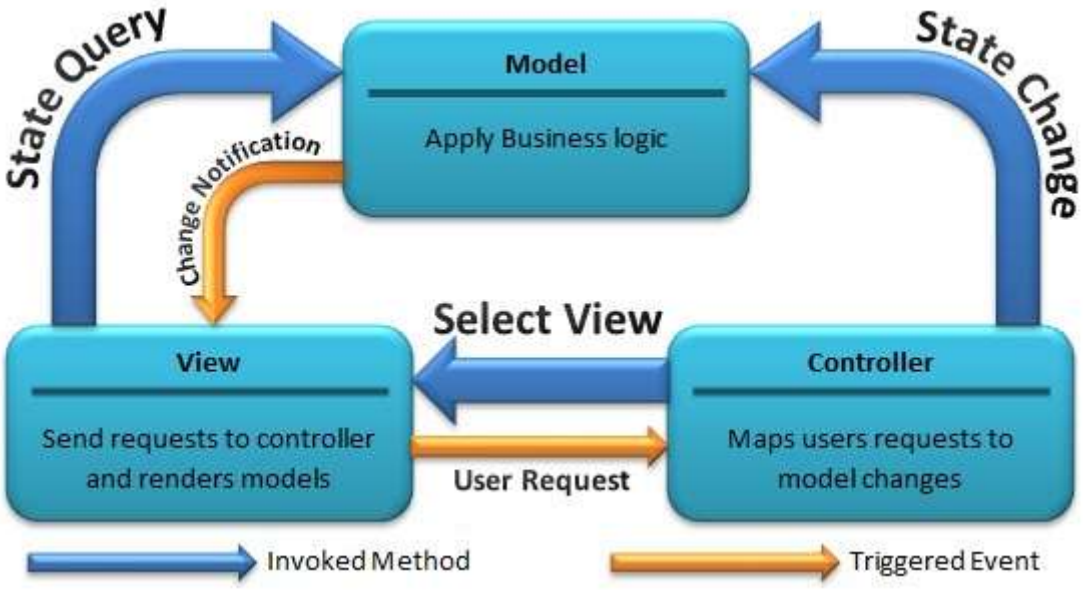

Figure 0.1: The MVC architecture

The user submits the request via the controller, which then sends the request to the model for further processing, with the option of selection the appropriate view for the request. The model then applies the business rules and communicates with the database to accommodate the controller's request. In addition, the model notifies the selected/appropriate view to update its instance with the new data processes. The view can also interact with the model to render its content.

- **ASP.NET Web API:** a framework that allows the developers to build a web service that utilizes the HTTP protocol. The service can be accessed by clients or even by the server using its API. The API responds to the requests with standard JSON or XML formatted data.

  - **Request flow and routing in ASP.NET MVC & Web API**

The routing process in ASP.NET MVC and Web API is very important task to handle and direct the requests and API calls. We will explain this process in brief and simple terms to get an idea of how the API requests are handled within our application. When a user sends a request to an APS.NET MVC application, the request goes through the HTTP Routing, and then to the route handlers. The route handlers send the request to the HTTP handlers, which select the appropriate controller associated with the request, and contact its function directly. The controller then selects the appropriate view to be presented back to the user. It is worth pointing out that routing is the most important stage in the request flow since it identifies the appropriate controller based on the request. All routings need to be registered in the Global.asax in order for the routing handler to be able to direct user's requests (Trivedi, 2013).

There is one main difference in the ASP.NET Web API routing which is the HHTP methods. The Web API uses the HTTP methods, which are the controller's methods or actions. Like GET, POST, UPDATE, and DELETE actions, these do not exist in ASP.NET MVC.

- **Google web search API:** We have used Google web search API mainly to obtain the number of web pages mentions a specific word, and then calculated the Inverse Document Frequency (IDF) which we outlined in detail in section 3.2.33.2.3 above.
- **English Stemmer:** This is one of the existing libraries we used for the summarization task. The existing implementation "English Stemmer" uses Porter stemming algorithm; see Appendix II for more details. Before we apply the stemming algorithm, we exclude all Stopwords and then stem the remaining. Stopwords are a list of common words in the language that do not have additional value in the sentence, such as: *the, in, after*. For the Stopwords list, we used a long English list obtained from one of Google's projects "Collection of stop words in 29 languages"[61] (see Appendix III for more details).

[61] The "Collection of stop words in 29 languages" project by Google https://code.google.com/p/stop-words/

Appendix II

Porter Stemming Algorithm Diagram

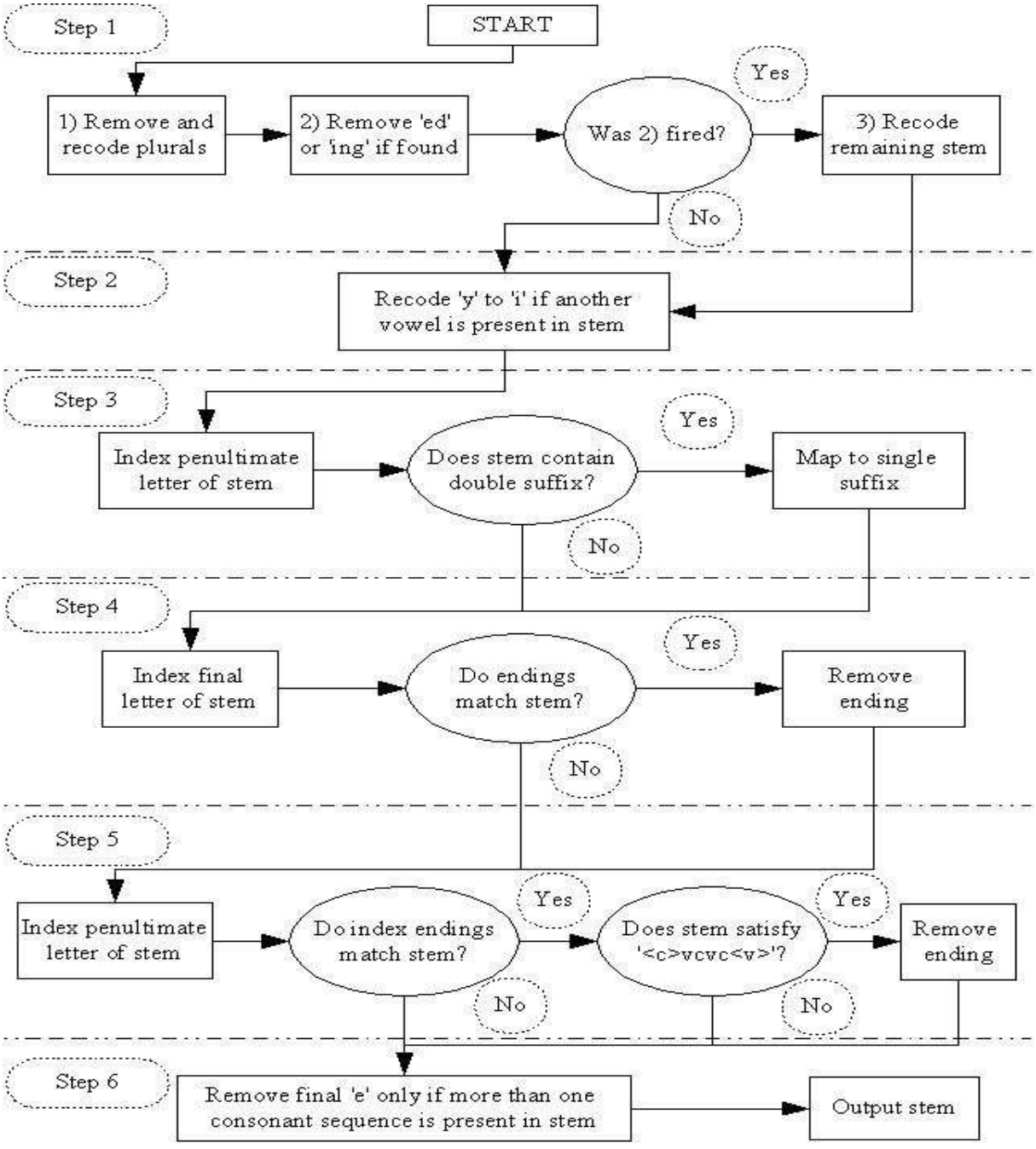

Figure 0.2: Porter stemming algorithm flowchart[62]

[62] The Lancaster Stemming Algorithm site:
http://www.comp.lancs.ac.uk/computing/research/stemming/Links/porter.htm

Appendix III

Stopwords List

Long English Stopwords list obtained from the "Collection of stop words in 29 languages" project by Google.

Table 0.1: Long English Stopwords list[63]

a	a's	able	about	above	according	accordingly
across	actually	after	afterwards	again	against	ain't
all	allow	allows	almost	alone	along	already
also	although	always	am	among	amongst	an
and	another	any	anybody	anyhow	anyone	anything
anyway	anyways	anywhere	apart	appear	appreciate	appropriate
are	aren't	around	as	aside	ask	asking
associated	at	available	away	awfully	b	be
became	because	become	becomes	becoming	been	before
beforehand	behind	being	believe	below	beside	besides
best	better	between	beyond	both	brief	but
by	c	c'mon	c's	came	can	can't
cannot	cant	cause	causes	certain	certainly	changes
clearly	co	com	come	comes	concerning	consequently
consider	considering	contain	containing	contains	corresponding	could
couldn't	course	currently	d	definitely	described	despite
did	didn't	different	do	does	doesn't	doing
don't	done	down	downwards	during	e	each
edu	eg	eight	either	else	elsewhere	enough
entirely	especially	et	etc	even	ever	every
everybody	everyone	everything	everywhere	ex	exactly	example
except	f	far	few	fifth	first	five
followed	following	follows	for	former	formerly	forth
four	from	further	furthermore	g	get	gets
getting	given	gives	go	goes	going	gone
got	gotten	greetings	h	had	hadn't	happens
hardly	has	hasn't	have	haven't	having	he
he's	hello	help	hence	her	here	here's
hereafter	hereby	herein	hereupon	hers	herself	hi
him	himself	his	hither	hopefully	how	howbeit
however	i	i'd	i'll	i'm	i've	ie
if	ignored	immediate	in	inasmuch	inc	indeed
indicate	indicated	indicates	inner	insofar	instead	into
inward	is	isn't	it	it'd	it'll	it's
its	itself	j	just	k	keep	keeps
kept	know	knows	known	l	last	lately
later	latter	latterly	least	less	lest	let
let's	like	liked	likely	little	look	looking
looks	ltd	m	mainly	many	may	maybe

[63] The "Collection of stop words in 29 languages" project by Google https://code.google.com/p/stop-words/

me	mean	meanwhile	merely	might	more	moreover
most	mostly	much	must	my	myself	n
name	namely	nd	near	nearly	necessary	need
needs	neither	never	nevertheless	new	next	nine
no	nobody	non	none	noone	nor	normally
not	nothing	novel	now	nowhere	o	obviously
of	off	often	oh	ok	okay	old
on	once	one	ones	only	onto	or
other	others	otherwise	ought	our	ours	ourselves
out	outside	over	overall	own	p	particular
particularly	per	perhaps	placed	please	plus	possible
presumably	probably	provides	q	que	quite	qv
r	rather	rd	re	really	reasonably	regarding
regardless	regards	relatively	respectively	right	s	said
same	saw	say	saying	says	second	secondly
see	seeing	seem	seemed	seeming	seems	seen
self	selves	sensible	sent	serious	seriously	seven
several	shall	she	should	shouldn't	since	six
so	some	somebody	somehow	someone	something	sometime
sometimes	somewhat	somewhere	soon	sorry	specified	specify
specifying	still	sub	such	sup	sure	t
t's	take	taken	tell	tends	th	than
thank	thanks	thanx	that	that's	thats	the
their	theirs	them	themselves	then	thence	there
there's	thereafter	thereby	therefore	therein	theres	thereupon
these	they	they'd	they'll	they're	they've	think
third	this	thorough	thoroughly	those	though	three
through	throughout	thru	thus	to	together	too
took	toward	towards	tried	tries	truly	try
trying	twice	two	u	un	under	unfortunately
unless	unlikely	until	unto	up	upon	us
use	used	useful	uses	using	usually	uucp
v	value	various	very	via	viz	vs
w	want	wants	was	wasn't	way	we
we'd	we'll	we're	we've	welcome	well	went
were	weren't	what	what's	whatever	when	whence
whenever	where	where's	whereafter	whereas	whereby	wherein
whereupon	wherever	whether	which	while	whither	who
who's	whoever	whole	whom	whose	why	will
willing	wish	with	within	without	won't	wonder
would	would	wouldn't	x	y	yes	yet
you	you'd	you'll	you're	you've	your	yours
yourself	yourselves	z	zero			

Appendix IV

Partial Closed Caption of the YouTube Video

<transcript>
<text start="2.45" dur="1.33">**in**</text>
<text start="3.78" dur="3.38">**Hi folks my name is Ben Davidson**</text>
<text start="7.16" dur="5.309">**I'm here on behalf of approximately 175,000 suspicious observers**</text>
<text start="**12.469**" dur="3.651">**we are a simple research group independent event**</text>
<text start="16.12" dur="4.42">**economic or political strings what does that mean**</text>
<text start="20.54" dur="3.12">**that means we take no corporate political**</text>
<text start="23.66" dur="4.9">**or other funding to support any side but the truth this may be a speech about the**</text>
<text start="28.56" dur="3.32">**flawsaglobalwarmingbutwe're not Petro folks**</text>
<text start="31.88" dur="3.37">**not oil boyce and we do not advocate deregulation**</text>
<text start="35.25" dur="3.42">**in fact first one to say that human pollution**</text>
<text start="38.67" dur="5.72">**presents an array of health problems respiratory endocrine central nervous**</text>
<text start="44.39" dur="2.239">**system the flora**</text>
<text start="46.629" dur="3.401">**fun water and atmosphere of our planet**</text>
<text start="50.03" dur="4.57">**contain poisonous and we put them there something in that vein**</text>
<text start="54.6" dur="4.619">**needs to change very sir that being said**</text>
<text start="**59.219**" dur="3.5">**are present here an argument for the superior already<**/text>
<text start="62.719" dur="3.521">**have been external climate forcing the current circumstance**</text>
…
…
</transcript>

Appendix V

Sample Discussion Dataset and Associated Summaries

Integrated discussions dataset:

subject : Political Neutrality
An advocacy group that is non-political and not funded by oil companies, gives it a degree of legitimacy that others who oppose global warming do not have. With the focus on the search for the truth, the lecturer allows for more openness and free thinking.

subject : Re: Political Neutrality
The world needs people who believe and are passionate. I have no problem with activists, climate activists or environmentalists - on the contrary - if it wasn't for some of these people, but - you have to consider people's vested interests and take precautions to make sure those interests don't get in the way.

subject : Re: Political Neutrality
If someone has a habit of dishonesty of course one would be foolish to take their word. Likewise if someone is respected for their honesty and diligence, I'm more likely to take them at face value. I am always skeptical, and if one investigates further and finds that the asserter hasn't published the relevant work (or in this case hasn't published anything for 10 years), makes assertions that are not supported by any evidence, and upon further investigation, finds that the assertions are actually directly contradicted by real world evidence, then it would be foolish not to discount the assertions.

subject : Human Responsibility
What assists with this argument is the fact that it is recognized that humans are to blame for excessive pollution and waste. Where others do not always recognize this, the lecturer here does note that this condition must change sooner than later.

subject : Re: Human Responsibility
I believe in taking care of the environment, buying only efficient products and recycling. But these alarmists and violent greens that burn other peoples Hummers really get me. Greenpeace is the worst of all, even one of its founders will have nothing more to do with them. They make me ashamed to say I am an environmentalist because it makes me part of the problem rather than part of the solution. Can't anyone see where rash actions lead? The blame for this is squarely on the UN itself for pushing Algorism and punishing skepticism and the green alarmists pointing fingers at oil companies instead of thinking things out rationally. Come on people, wake up, get off the bandwagon and start using constructive criticism.

subject : Re: Human Responsibility
By nature we waste resources and over-consume, This trend will continue for as long as there are goods to consume and there are goods that can be made to be consumed.

subject : Predictions
Through the use of trends, scientists and researchers can predict future events. The lecturer is making the announcement that the correlation between CO2 and temperature has been compromised by humans. Since the Industrial Revolution the balance between CO2 and temperature which remained constant has now been disrupted. This may cause a degree of inaccuracy for all future predictions as it is not constant.

subject : Re: Predictions
Nice correlation between T and CO2 level. However, given a correlation between two variables one has still to clarify which is doing what. One might of course notice that in the most recent "burst" CO2 gets well above the temperature curve, unlike the previous cases.

subject : Re: Predictions
There is no heat being added unless you are talking about variation in Solar output. The warming is supposed to be the result of Earth holding on to more heat and releasing it more slowly. The earth is not a black body radiator, but that doesn't even matter because CO2 isn't a big absorber of black body radiation at this temperature.

subject : Pause in Global Warming
The lecturer states that the reduction of sun spots could in fact reduce or even pause the effects of global warming. This lack of solar activity could in fact provide evidence that this is just one of many global warming phases that occurred over the centuries.

subject : Re: Pause in Global Warming

The bottom line is that all our heat comes from the sun. If it cools down so does the earth. Any variances due to different ocean currents, magnetic fields, etc., are only releasing stored energy from the sun.

At the end of the day, this planet will be a Dead Rock circling a spent sun.

Let's hope it warms up, CO2 increases, plants grow and life becomes comfortable for a while. The alternative is not nice.

subject : Re: Pause in Global Warming

Warmer winters would mean fewer deaths, particularly among vulnerable groups like the aged. However, the same groups are also vulnerable to additional heat, and deaths attributable to heatwaves are expected to be approximately five times as great as winter deaths prevented. It is widely believed that warmer climes will encourage migration of disease-bearing insects like mosquitoes and malaria is already appearing in places it hasn't been seen before.

subject : New Ice Age

The evidence being presented is that with the lack of solar activity there may be a new Ice Age. Within Canada, the winter of 2013/2014 saw severe record breaking cold. Just over the past few years, we have witnessed colder and longer winters which may become more brutal based on this evidence.

subject : Re: New Ice Age

OK. This is what I have to say. I am not skeptical about the warming trend caused by atmospheric changes. I do think the current discussion in the media is limited. If we were about to enter an ice age, we might want more greenhouse gases to counter an ice age trend independent of the atmosphere and likely caused by astronomical features. The above discussion says an ice age is unlikely because of the warming forces.

subject : Re: New Ice Age

In a place where the temperature is always well below freezing, "global warming" is not going to melt all the ice. That doesn't mean it isn't a problem elsewhere. Even if there were no net ice loss on earth, if we're losing ice in places we need it (such as mountain ranges that supply people with drinking water), and accumulate it in places that have no humans at all (Antarctica), that's an enormous problem.

subject : Popular Science

We see the ground work for a popular science approach to reality. What sells, becomes the truth; what people want to hear, becomes the truth. Be it inaccurate science or a planned popular consensus disregarding truth, at the end of the day humanity cannot afford to play games with is environmental future.

subject : Re: Popular Science

Should we not wait for 100% certainty before taking action? Outside of logic and mathematics, we do not live in a world of certainties. Science comes to tentative conclusions based on the balance of evidence. The more independent lines of evidence are found to support a scientific theory, the closer it is likely to be to the truth. Just because some details are still not well understood should not cast into doubt our understanding of the big picture: humans are causing global warming.

subject : Re: Popular Science

Consensus in science is different from a political one. There is no vote. Scientists just give up arguing because the sheer weight of consistent evidence is too compelling, the tide too strong to swim against any longer. Scientists change their minds on the basis of the evidence, and a consensus emerges over time.

subject : Scientific Consensus

We find that there does not exist consensus amongst scientists in regards to climate change. While many support global warming, others see global cooling as the actual reality. What is really at risk is the truth. Scientists are people and can be enticed to support one position or the other. What starts as fact, could be skewed in such a way that it can be used to support one agenda or the other.

subject : Re: Scientific Consensus

The claim that there is some vast number of scientists that constitute a consensus and that agree with catastrophic warming is not only not science, it isn't even correct. We constantly see people pointing out that scientists supposedly prove this supposed consensus, where in fact there is still division.

subject : Re: Scientific Consensus

Scientists (and regular human beings) are also affected by cultural, social, and personal beliefs.... Rather than the traditional view that science is to be protected from biases and other imperfections of people, it turns out that science is inescapably infected with humanness.

Sample Automated MFMCS Summary:

An advocacy group that is non-political and not funded by oil companies, gives it a degree of legitimacy that others who oppose global warming do not have. I have no problem with activists, climate activists or environmentalists. On the contrary, if it wasn't for some of these people, but you have to consider people's vested interests and take precautions to make sure those interests don't get in the way. What assists with this argument is the fact that it is recognized that humans are to blame for excessive pollution and waste. Where others do not always recognize this, the lecturer here does note that this condition must change sooner than later. The lecturer is making the announcement that the correlation between CO2 and temperature has been compromised by humans. Since the industrial revolution, the balance between CO2 and temperature, which remained constant, has now been disrupted. One might of course notice that in the most recent 'burst', CO2 gets well above the temperature curve, unlike the previous cases. The warming is supposed to be the result of earth holding on to more heat and releasing it more slowly. The earth is not a black body radiator, but that doesn't even matter because CO2 isn't a big absorber of black body radiation at this temperature. The lecturer states that the reduction of sun spots could in fact reduce or even pause the effects of global warming. This lack of solar activity could in fact provide evidence that this is just one of many global warming phases that occurred over the centuries. Any variances due to different ocean currents, magnetic fields, etc., are only releasing stored energy from the sun. At the end of the day, this planet will be a dead rock circling a spent sun. Let's hope it warms up, CO2 increases, plants grow, and life becomes comfortable for a while. It is widely believed that warmer climates will encourage migration of disease-bearing insects like mosquitoes and malaria, which is already appearing in places it hasn't been seen before. The evidence being presented is that with the lack of solar activity there may be a new ice age. Within Canada, the winter of 2013/2014 saw severe record breaking cold. If we were about to enter an ice age, we might want more greenhouse gases to counter an ice age trend independent of the atmosphere and likely caused by astronomical features. In a place where the temperature is always well below freezing, 'global warming' is not going to melt all the ice. Even if there were no net ice loss on earth, if we're losing ice in places we need it (such as mountain ranges that supply people with drinking water), and accumulate it in places that have no humans at all (Antarctica), that's an enormous problem. The more independent lines of evidence are found to support a scientific theory, the closer it is likely to be to the truth. Just because some details are still not well understood should not cast

into doubt our understanding of the big picture: humans are causing global warming. Scientists change their minds on the basis of the evidence, and a consensus emerges over time. We find that there does not exist consensus amongst scientists in regards to climate change. What is really at risk is the truth.

Sample human summaries:

Human summary 1:

If someone has a habit of dishonesty, of course one would be foolish to take their word. Likewise, if someone is respected for their honesty and diligence, I'm more likely to take them at face value. I am always skeptical, and if one investigates further and finds that the asserter hasn't published the relevant work (or in this case hasn't published anything for 10 years), makes assertions that are not supported by any evidence, and upon further investigation, finds that the assertions are actually directly contradicted by real world evidence, then it would be foolish not to discount the assertions. What assists with this argument is the fact that it is recognized that humans are to blame for excessive pollution and waste. By nature, we waste resources and over-consume. This trend will continue for as long as there are goods to consume and there are goods that can be made to be consumed. Through the use of trends, scientists and researchers can predict future events. The lecturer is making the announcement that the correlation between CO2 and temperature has been compromised by humans. Since the industrial revolution, the balance between CO2 and temperature, which remained constant, has now been disrupted. This may cause a degree of inaccuracy for all future predictions as it is not constant. The bottom line is that all our heat comes from the sun. If it cools down, so does the earth. Any variances due to different ocean currents, magnetic fields, etc., are only releasing stored energy from the sun. In a place where the temperature is always well below freezing, "global warming" is not going to melt all the ice. That doesn't mean it isn't a problem elsewhere. Even if there were no net ice loss on earth, if we're losing ice in places we need it (such as mountain ranges that supply people with drinking water), and accumulate it in places that have no humans at all (Antarctica), that's an enormous problem. There is no vote. Scientists just give up arguing because the sheer weight of consistent evidence is too compelling, the tide too strong to swim against any longer. Scientists change their minds on the basis of the evidence, and a consensus emerges over time.

Human summary 2:

An advocacy group that is non-political and not funded by oil companies, gives it a degree of legitimacy that others who oppose global warming do not have. You have to consider people's vested interests and take precautions to make sure those interests don't get in the way. Humans are to blame for excessive pollution and waste. The lecturer is making the announcement that the correlation between CO2 and temperature has been compromised by humans. Since the industrial revolution, the balance between CO2 and temperature, which remained constant, has now been disrupted. The lecturer states that the reduction of sun spots could in fact reduce or even pause the effects of global warming. Let's hope it warms up, CO2 increases, plants grow, and life becomes comfortable for a while. The evidence being presented is that with the lack of solar activity, there may be a new ice age. An ice age is unlikely because of the warming forces. The more independent lines of evidence are found to support a scientific theory, the closer it is likely to be to the truth. Humans are causing global warming. Scientists change their minds on the basis of the evidence, and a consensus emerges over time. We find that there does not exist consensus amongst scientists in regards to climate change.

Human summary 3:

You have to consider people's vested interests and take precautions to make sure those interests don't get in the way. It is recognized that humans are to blame for excessive pollution and waste. Since the industrial revolution the balance between CO2 and temperature, which remained constant, has now been disrupted. However, given a correlation between two variables, one has still to clarify which is doing what. Lack of solar activity could in fact provide evidence that this is just one of many global warming phases that occurred over the centuries. With the lack of solar activity there may be a new ice age. The more independent lines of evidence are found to support a scientific theory, the closer it is likely to be to the truth. We find that there does not exist consensus amongst scientists in regards to climate change. We constantly see people pointing out that scientists supposedly prove this supposed consensus, where in fact there is still division.

Human summary 4:

By nature we waste resources and over-consume, this trend will continue for as long as there are goods to consume and there are goods that can be made to be consumed. Through the use of trends, scientists and researchers can predict future events. The lecturer is making the announcement that the correlation between CO2 and temperature has been compromised by humans. Since the industrial revolution, the balance between CO2 and temperature, which remained constant, has now been disrupted - nice correlation between T and CO2 level. However, given a correlation between two variables, one has still to clarify which is doing what. The bottom line is that all our heat comes from the sun. Any variances due to different ocean currents, magnetic fields, etc., are only releasing stored energy from the sun. The evidence being presented is that with the lack of solar activity there may be a new ice age. Consensus in science is different from a political one. Scientists just give up arguing because the sheer weight of consistent evidence is too compelling, the tide too strong to swim against any longer. There does not exist consensus amongst scientists in regards to climate change. Scientists are people and can be enticed to support one position or the other. What starts as fact could be skewed in such a way that it can be used to support one agenda or the other. Science is to be protected from biases and other imperfections of people, it turns out that science is inescapably infected with humanness.

Sample summary from Dragon Toolkit software:

This lack of solar activity could in fact provide evidence that this is just one of many global warming phases that occurred over the centuries. I am always skeptical, and if one investigates further and finds that the asserter hasn't published the relevant work(or in this case hasn't published anything for 10 years), makes assertions that are not supported by any evidence, and upon further investigation, finds that the assertions are actually directly contradicted by real world evidence, then it would be foolish not to discount the assertions. I am not skeptical about the warming trend caused by atmospheric changes. The lecturer is making the announcement that the correlation between Co2 and Temperature has been compromised by humans. The more independent lines of evidence are found to support a scientific theory, the closer it is likely to be to the truth. Just because some details are still not well understood should not cast into doubt our understanding of the big picture: humans are causing global warming. Scientists are people and can be enticed to support one position or the other. The Earth is not a Black body radiator, but that doesn't even matter because CO2 isn't a big absorber of black body radiation at this

temperature. We constantly see people pointing out that scientists supposedly prove this supposed consensus, where in fact there is still division. The above discussion says an ice age is unlikely because of the warming forces. The world needs people who believe and are passionate. The claim that there is some vast number of scientists that constitute a consensus and that agree with catastrophic warming is not only not science, it isn't even correct. In a place where the temperature is always well below freezing," global warming" is not going to melt all the ice. Just over the past few year we have witnessed colder and longer winters which may become more brutal based on this evidence. While many support global warming, others see global cooling as the actual reality. This may cause a degree of inaccuracy for all future predictions as it is not constant. The bottom line is that all our heat comes from the Sun. Scientists change their minds on the basis of the evidence, and a consensus emerges over time. If we were about to enter an ice age, we might want more greenhouse gases to counter an ice age trend independent of the atmosphere and likely caused by astronomical features. What assists with this argument is the fact that it is recognized that humans are to blame for excessive pollution and waste. Warmer winters would mean fewer deaths, particularly among vulnerable groups like the aged. Science comes to tentative conclusions based on the balance of evidence. The lecturer states that the reduction of sun spots could in fact reduce or even pause the effects of global warming.

Sample summary from Text Compactor website based on Open Text Summarization:

An advocacy group that is non-political and not funded by oil companies, gives it a degree of legitimacy that others who oppose global warming do not have. I have no problem with activists, climate activists or environmentalists - on the contrary - if it wasn’t for some of these people, but - you have to consider people's vested interests and take precautions to make sure those interests don’t get in the way. If someone has a habit of dishonesty of course one would be foolish to take their word. I am always skeptical, and if one investigates further and finds that the asserter hasn't published the relevant work (or in this case hasn't published anything for 10 years), makes assertions that are not supported by any evidence, and upon further investigation, finds that the assertions are actually directly contradicted by real world evidence, then it would be foolish not to discount the assertions. What assists with this argument is the fact that it is recognized that humans are to blame for excessive pollution and waste. Come on people, wake up, get off the bandwagon and start using constructive criticism. By nature we waste resources and over-consume. This trend will continue for as long as there are goods to consume and there

are goods that can be made to be consumed. Through the use of trends, scientists and researchers can predict future events. This lack of solar activity could in fact provide evidence that this is just one of many global warming phases that occurred over the centuries. The bottom line is that all our heat comes from the Sun. The above discussion says an ice age is unlikely because of the warming forces. In a place where the temperature is always well below freezing, "global warming" is not going to melt all the ice. Even if there were no net ice loss on earth, if we're losing ice in places we need it (such as mountain ranges that supply people with drinking water), and accumulate it in places that have no humans at all (Antarctica), that's an enormous problem. We see the ground work for a popular science approach to reality. Rather than the traditional view that science is to be protected from biases and other imperfections of people, it turns out that science is inescapably infected with humanness.

Sample summary from Free Summarizer website:

I am always skeptical, and if one investigates further and finds that the asserter hasn't published the relevant work (or in this case hasn't published anything for 10 years), makes assertions that are not supported by any evidence, and upon further investigation, finds that the assertions are actually directly contradicted by real world evidence, then it would be foolish not to discount the assertions. The lecturer states that the reduction of sun spots could in fact reduce or even pause the effects of global warming. This lack of solar activity could in fact provide evidence that this is just one of many global warming phases that occurred over the centuries. If we were about to enter an ice age, we might want more greenhouse gases to counter an ice age trend independent of the atmosphere and likely caused by astronomical features. In a place where the temperature is always well below freezing, "global warming" is not going to melt all the ice. Even if there were no net ice loss on earth, if we're losing ice in places we need it (such as mountain ranges that supply people with drinking water), and accumulate it in places that have no humans at all (Antarctica), that's an enormous problem. What sells, becomes the truth; what people want to hear, becomes the truth. While many support global warming, others see global cooling as the actual reality. The claim that there is some vast number of scientists that constitute a consensus and that agree with catastrophic warming is not only not science, it isn't even correct. Rather than the traditional view that science is to be protected from biases and other imperfections of people, it turns out that science is inescapably infected with humanness.

Appendix VI

MFMCS System setup

This setup is divided into two components setup: the JSMPW and the Summarization application.

Prior to installing the MFMCS system we need to setup the web server, database server, and Moodle LMS as per Appendix I.

1. JSMPW setup: Create a filter type Moodle plug-in as described in the Moodle development documentation website[64]. Add the following additional folders: css, javascript, and pix, which contain the required css, javascript, and images files respectively. The additional files and folder structure would be as demonstrated in Figure 0.3.

   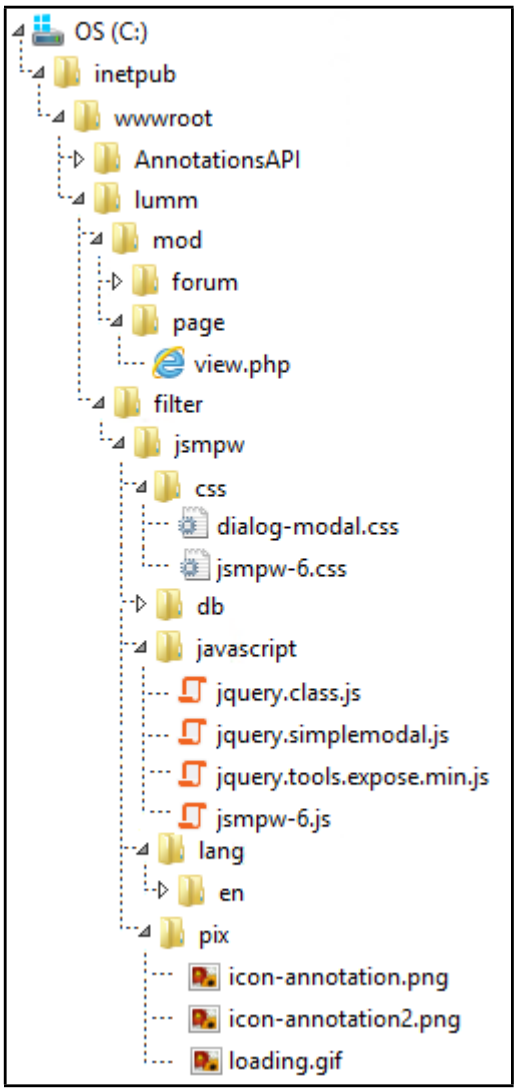

   Figure 0.3: JSMPW files sctructure

   Once the plug-in main folder (jsmpw) folder is placed under the filter folder, Moodle will automatically install the plug-in and create the database tables. The final step is to reference jQuery 1.10 version or higher and the (jsmpw) css and javascript files in the view.php file within Moodle's page module; refer to Figure 0.3. This page is responsible for loading the course weekly lectures content which includes the YouTube link to the video lecture. Note that Moodle has the jQuery library file in the lib folder under the main directory.

   We also modified some of the forum module files refer to Table 0.2.

   If the database tables created by the plug-in were not created correctly, access the database server and recreate them using the attached script in reference Appendix VII

[64] https://docs.moodle.org/dev/Filters

2. Summarization application: this application is published to the AnnotationAPI directory; see Figure 0.3 , using MSVS. Once the application is published we need to modify the Web.config file as follow: change the "HostServer" configuration value to "Server". Otherwise the application would work as if it is setup at localhost web server for development.

Table 0.2: Discussion Forum module files changes

File	Change type	Change	Change comment
View_embd.php	New	line 112	Add the following: /// Mohannad added $PAGE->set_pagelayout('embedded'); ///End Mohannad Addition Everything else is same as view.php
Index_embd.php	New	line 49	Add the following: /// Mohannad added $PAGE->set_pagelayout('embedded'); ///End Mohannad Addition Everything else is same as index.php And comment out this $PAGE->set_pagelayout('incourse');
Post_embd.php	New	line 503	Add the following: /// Mohannad added $PAGE->set_pagelayout('embedded'); ///End Mohannad Addition Everything else is same as post.php
Post_embd.php	Change	Line 516	Change from ('post.php', to ('post_embd.php',
Post_embd.php	Change	Line 609	Change from view.php to view_embd.php
Post_embd.php	Change	Line 684	Change from view.php to view_embd.php
Post_embd.php	Change	Line 686	Change from discuss.php to discuss_embd.php
Post_embd.php	Change	Line 727	Change from view.php to view_embd.php
Post_embd.php	Change	Line 729	Change from discuss.php to discuss_embd.php
Post_embd.php	Change	Line 779	Change from discuss.php to discuss_embd.php
Post_embd.php	Change	Line 805	Change from view.php to view_embd.php And comment the line out

Post_embd.php	Change	Line 688	Comment the line
			//redirect(forum_go_back_to("$discussionurl"), $message.$subscribemessage, $timemessage);
			And add the following:
			/// mohannad commented "Edit Form"
			$data = array();
			$data['d']=$discussion->id;
			$data['title']=$fromform->subject;
			echo json_encode($data);
			/// END Mohannad change

Post_embd.php	Change	Line 740	Comment the line:
			//redirect(forum_go_back_to("$discussionurl#p$fromform->id"), $message.$subscribemessage, $timemessage);
			And add the following:
			/// Mohannad Commanted "Forum Reply"
			$data = array();
			$data['d']=$discussion->id;
			$data['title']=$fromform->subject;
			echo json_encode($data);
			/// END Mohannad change

Post_embd.php	Change	Line 800	Comment the line:
			//redirect(forum_go_back_to("view_embd.php?f=$fromform->forum"), $message.$subscribemessage, $timemessage);
			And add the following:
			/// Mohannad Commanted "New Forum"
			$data = array();
			$data['d']=$discussion->id;
			$data['title']=$fromform->subject;
			echo json_encode($data);
			/// End Mohanand Changes

Appendix VII

MFMCS System code

JSMPW plug-in JavaScript file (jsmpw-6.js):

jsmpw-6.js

JSMPW plug-in CSS file (jsmpw-6.css):

jsmpw-6.css

Required image files:

loading.gif

icon-annotation.png

icon-annotation2.png

External libraries files:

dialog-modal.css

jquery.tools.expose.min.js

jquery.simplemodal.js

jquery.class.js

Summarization application code:

The Summarization application project contains over 70 files and 16 folders. For that reason we attached a compressed version of it below:

Summarization Application.zip

Database new tables create scripts:

Annotations.sql

Media Info.sql

URL Params.sql

Terms IDF.sql

Bibliography

Agarwal, A. (2013, Nov 7). *MOOCs for the Arab World*. Retrieved from edx.org: https://www.edx.org/blog/moocs-arab-world

Alario-Hoyos, C., Pérez-Sanagustín, M., Delgado-Kloos, C., Muñoz-Organero, M., Rodríguez-de-las-Heras, A., & Parada G., H. (2013). Analysing the impact of built-in and external social tools in a MOOC on educational technologies. In D. Hernández-Leo, T. Ley, R. Klamma, & A. Harrer, *Scaling up learning for sustained impact* (Vol. 8095, pp. 5-18). Springer Berlin Heidelberg.

ANGEL Learning. (2009). *Angel LMS - Discussion Forums*. Retrieved from Angel Learning: http://www.angellearning.com/products/lms/discussions.html

Bandura, A. (1977). *Social Learning Theory*. Englewood Cliffs, NJ: Prentice Hall.

Barrish, J. (2014, Jan 23). *Top LMS Software*. Retrieved Feb 09, 2014, from www.capterra.com: http://www.capterra.com/infographics/top-lms-software

Becker, K. (2013, Sep 13). *TF*IDF in C# .NET for Machine Learning*. Retrieved May 24, 2014, from Primary Objects: http://www.primaryobjects.com/CMS/Article157.aspx

Bishop, T. (2013, Jan 7). *Google Course Builder: Implications For The Learning Community*. Retrieved from Knowledge Vision: http://www.knowledgevision.com/google-course-builder-implications-for-the-learning-community

Bysani, P. (2010). Progressive Summarization: Summarizing relevant and novel information. *Master dissertation*. International Institute of Information Technology, Hyderabad.

Clark, J. J. (1906). *The Correspondence School--Its Relation to Technical Education and Some of Its Results* (Vol. 24). Science. doi:10.1126/science.24.611.327

Crowdsourcing - Definition and More. (n.d.). Retrieved Feb 25, 2014, from Merriam-Webster.com: http://www.merriam-webster.com/dictionary/crowdsourcing

Cuypers, H., & Knopper, J. W. (2013). Interactive Mathematical Videos. *CICM Workshops*. Retrieved from http://ceur-ws.org/Vol-1010/paper-16.pdf

Das, D., & Martins, A. F. (2007). A Survey on Automatic Text Summarization. *Literature Survey for the Language and Statistics II course at CMU, 4*, 192-195.

de Oliveira, P. C., Torrens, E. W., Cidral, A., Schossland, S., & Bittencourt, E. (2008). Evaluating Summaries Automatically – a system proposal. *Director, 3*(3), 1-8.

Dougiamas, M. (2013, Aug 29). *Why a Moodle MOOC?* Retrieved Jan 28, 2014, from Learn Moodle: http://learn.moodle.net/mod/page/view.php?id=50

Dougiamas, M. (2014, Jan). *About Learn.Moodle.net*. Retrieved Jan 2014, from Learn Moodle: http://learn.moodle.net/mod/page/view.php?id=40

Downes, S. (2012). *Connectivism and Connective Knowledge: Essays on meaning and learning networks.* Canada: National Research Council. doi:ISBN: 978-1-105-77846-9

Downes, S. (2012). *The LMS and the MOOC.* Retrieved from http://www.slideshare.net/Downes/the-lms-and-the-mooc

Downes, S. (2013). What a MOOC Does. Retrieved Nov 11, 2013, from http://halfanhour.blogspot.ca/2012/03/what-mooc-does-change11.html

Du, Y. (2014). Massive Open Online Course: The Implication to iSchool Education. *iConference 2014 Proceedings* (pp. 884 - 888). iSchools. doi:10.9776/14296

Ellen, J. (2011). All about Microtext - A Working Definition and a Survey of Current Microtext Research within Artificial Intelligence and Natural Language Processing.

Erkan, G., & Radev, D. R. (2004). LexRank: Graph-based Lexical Centrality as Salience in Text Summarization. *Journal of Artificial Intelligence Research, 22*, 457–479.

Grünewald, F., Meinel, C., Totschnig, M., & Willems, C. (2013). Designing MOOCs for the Support of Multiple Learning Styles. In F. Grünewald, C. Meinel, M. Totschnig, & C. Willems, *Scaling up Learning for Sustained Impact* (pp. 371-382). Springer Berlin Heidelberg. doi:10.1007/978-3-642-40814-4_29

Hamuy, E., & Galaz, M. (2010). Information versus Communication in Course Management System participation. *Computers & Education, 54*(1), 169-177. doi:10.1016/j.compedu.2009.08.001

Hill, P. (2013). MOOC Discussion Forums: barrier to engagement? *e-Literate*. Retrieved Nov 10, 2013, from http://mfeldstein.com/mooc-discussion-forums-barriers-engagement/

Hobbs, J. R. (1990). Topic drift. In *Conversational organization and its development* (Vol. 38, pp. 3-22). Ablex Publishing Corporation.

Howe, J. (2006). The rise of crowdsourcing. *Wired magazine, 14*(6), 1-4.

Hunt, T. (2012). Moodle. In A. Brown, & G. Wilson, *The Architecture of Open Source Applications* (Vols. Volume II: Structure, Scale, and Few More Fearless Hacks, pp. 230 - 248). Creative Commons: Attribution. doi:ISBN: 9781105571817

Hutchins, J. (2005). The history of machine translation in a nutshell.

Jones, K. S. (1999). Automatic summarising: factors and directions. In M. T. Inderjeet Mani, *Advances in Automatic Text Summarization* (pp. 1-12).

Jones, K. S. (2007). Automatic summarising: The state of the art. In *Information Processing & Management* (Vol. 43(6), pp. 1449–1481). doi:10.1016/j.ipm.2007.03.009

Jurafsky, D., & Martin, J. H. (2000). *Speech and Language Processing: An Introduction to Natural Language Processing,Computational Linguistics, and Speech Recognition.* Prentice Hall PTR.

Kanjilal, J. (2013, Dec). *Working with ASP.NET Web API*. Retrieved Apr 29, 2014, from Packt Publishing: http://www.packtpub.com/article/working-with-aspnet-web-api

Karpicke, J. D., & Blunt, J. R. (2011). Retrieval Practice Produces More Learning than Elaborative Studying with Concept Mapping. *Science, 331*(6018), 772-775. doi:10.1126/science.1199327

Koller, D. (Performer). (2012). What we're learning from online education. Retrieved from http://www.ted.com/talks/lang/en/daphne_koller_what_we_re_learning_from_online_education.html

Kolowich, S. (2013). *Google and edX Create a MOOC Site for the Rest of Us*. Retrieved from Chronicle of Higher Education: http://chronicle.com/blogs/wiredcampus/google-and-edx-create-a-mooc-site-for-the-rest-of-us/46413?cid=wc&utm_source=wc&utm_medium=en

Kunder, M. d. (2014, Jun 2). *The size of the World Wide Web*. Retrieved Jun 3, 2014, from World Wide Web Size: http://www.worldwidewebsize.com/

Liip AG. (2014). *LIIP Projects*. Retrieved Feb 14, 2014, from www.liip.ch: http://www.liip.ch/en/what/projects/moodle-as--software-as-a-service-

Lin, C.-Y. (2004). Looking for a Few Good Metrics: Automatic Summarization Evaluation - How Many Samples Are Enough? *In Proceedings of the NTCIR Workshop*, *Vol. 4*, pp. 1-10.

Lin, C.-Y., & Hovy, E. (2003). Automatic evaluation of summaries using n-gram co-occurrence statistics. *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1* (pp. 71-78). Association for Computational Linguistics.

Liyanagunawardena, T., Adams, A., & Williams, S. (2013). MOOCs: A Systematic Study of the Published Literature 2008-2012. *The International Review Of Research In Open And Distance Learning, 14*(3), 202-227. Retrieved from http://www.irrodl.org/index.php/irrodl/article/view/1455/2531

Mackness, J., Mak, S. F., & Williams, R. (2010). The Ideals and Reality of Participating in a MOOC. In *Proceedings of the 7th International Conference on Networked Learning 2010* (pp. 266-274).

Manning, C. D., Raghavan, P., & Schütze, H. (2009). Scoring, term weighting & the vector space model. In *Introduction to Information Retrieval* (pp. 109-133). Cambridge University Press. Retrieved from http://www-nlp.stanford.edu/IR-book/

Martin, E. (2013, Feb 17). *jQuery Simple modal Project*. Retrieved Mar 30, 2014, from Eric Martin Simple modal: http://www.ericmmartin.com/projects/simplemodal/

Martinet, A., & Palmer, E. (1960). *Elements of General Linguistics.* Faber & Faber.

McAuley, A., Stewart, B., Siemens, G., & Cormier, D. (2010). *The MOOC model for digital practice.*

Meyer, J. (2010, Jun 15). *A Simple, Powerful, Lightweight Class for jQuery*. Retrieved Mar 23, 2014, from Bitovi: http://bitovi.com/blog/2010/06/a-simple-powerful-lightweight-class-for-jquery.html

Microsoft. (n.d.). *.edmx File Overview (Entity Framework)*. Retrieved Jul 2014, from Microsoft msdn Documentations: http://msdn.microsoft.com/en-ca/library/vstudio/cc982042(v=vs.100).aspx

Moodle Developers. (2014, Jan). *Moodle architecture*. Retrieved Feb 2014, from Moodle Dev Documents: http://docs.moodle.org/dev/Moodle_architecture

Morris, S. M., & Stommel, J. (2013). The Discussion Forum is Dead; Long Live the Discussion Forum - See more at: http://www.hybridpedagogy.com/Journal/files/Discussion_Forum_is_Dead.html#sthash.ORIv4vxU.dpuf. *Hybrid Pedagogy*. Retrieved from http://www.hybridpedagogy.com/Journal/files/Discussion_Forum_is_Dead.html

Ng, X. (2013, August). *Moodle launches its first official MOOC with teachers in mind*. Retrieved from Moodle: http://moodle.com/moodle-launches-its-first-official-mooc-with-teachers-in-mind/

Nicholson, P. (2007). A history of e-learning. In *Computers and education* (pp. 1-11). Springer Netherlands.

Olson, D. L., & Delen, D. (2008). *Advanced Data Mining Techniques* (1st edition ed.). Springer. doi:ISBN 3-540-76916-1

OpenMOOC Home Page. (2012, Jul). Retrieved Mar 12, 2014, from openmooc.oeg: http://openmooc.org/

Panchenko, L. F. (2013). Massive Open Online Course as an Alternative Way of Advanced Training for Higher Educational Establishment Professors. *Education and Pedagogical Sciences*. Retrieved from http://pedagogicaljournal.luguniv.edu.ua/archive/2013/N1/articles/3/Panchenko_eng.pdf

PAPPANO, L. (2012, Nov 4). The Year of the MOOC. *The New York Times*. Retrieved from www.lexisnexis.com/hottopics/lnacademic

Peco, P. P., & Luján-Mora, S. (2013, October). Architecture of a MOOC based on CourseBuilder. *Information Technology Based Higher Education and Training (ITHET), 2013 International Conference on*, (pp. 1-8). doi:10.1109/ITHET.2013.6671045

Radev, D., Blitzer, J., Winkel, A., Allison, T., Topper, M., elebi, A. C., & Craig, M. (2006, Mar 21). *MEAD Documentation v3.10.* Retrieved Jul 2014, from MEAD Summarization: http://www.summarization.com/mead/

Sadler, P., & Good, E. (2006). The Impact of Self- and Peer-Grading on Student Learning. *Educational Assessment, 11*(1), 1-31. doi:10.1207/s15326977ea1101_1

Siemens, G. (2012). MOOCs are really a platform. Retrieved Nov 05, 2013, from http://www.elearnspace.org/blog/2012/07/25/moocs-are-really-a-platform/

Simonson, M. (2008). Course Management Systems. *Quarterly review of distance education, 8*(1), 7-11.

Straus, J. (2008). *The Blue Book of Grammar and Punctuation* (Tenth Edition ed.). San Francisco,, CA: Jossey-Bass. Retrieved from http://www.grammarbook.com/english_rules.asp

Three Dub Media. (2008, Jul 14). *jQuery Tools: The missing UI library from the web*. Retrieved Apr 06, 2014, from jQuery Tools: http://jquerytools.org/

Trivedi, J. (2013, Sep 13). *ASP.Net MVC Request Life Cycle*. Retrieved Jul 08, 2014, from C-sharp Corner: http://www.c-sharpcorner.com/UploadFile/ff2f08/Asp-Net-mvc-request-life-cycle/

(1922). *Tufts College to Give Radio Lecture Course.* Olympia (WA): daily recorder.

Woolley, D. R. (1994). *PLATO: The Emergence of Online Community*. Retrieved from http://thinkofit.com/plato/dwplato.htm

Yatsko, V. A., & Vishnyakov, T. N. (2007). A method for evaluating modern systems of automatic text summarization. *Automatic Documentation and Mathematical Linguistics*.