Master Erasmus Mundus in
Color in Informatics and Media Technology (CIMET)

Automatic Semantic Annotation for Media Learning Objects

Master Thesis Report

Presented by

Laksmita Rahadianti

and defended at

Høgskolen i Gjøvik

Academic Supervisor: Prof. Faouzi Alaya Cheikh

Jury Committee:
Prof. Damien Muselet
Prof. Hubert Konik

Automatic Semantic Annotation for Media Learning Objects

Laksmita Rahadianti

2012/07/15

Abstract

Recent advances in technology have provided alternative solutions and approaches to everyday tasks. One of which is in education, where the learning process is no longer confined to conventional classrooms. E-learning or distance learning are now a popular alternative which involves the use of learning objects for teachers to convey instructional content. These learning objects are accessible to students on a digital repository and can come in many different forms. Recently multimedia files such as videos are also used. In order to enable efficient indexing and retrieval of these videos, they must be encapsulated into effective media learning objects. Although many querying methods exist, users are most accustomed to query by keyword methods, giving text based queries to retrieve corresponding videos. Generally these keywords are selected manually, but this method is not favorable because manual annotation is restrictive to a set of words, subjective to the annotator, and overall a labor intensive process. This research explores semantic keyword selection methods for automatic video annotation. Cross document annotation is used to extract potential keywords by taking into consideration surrogate documents, e.g. transcript, slides, lecture notes, etc. These potential keywords are then refined based on a set of preselected seed words in order to obtain highly related keywords, based on WordNet and visualness similarity scores. Three novel objective scoring methods are proposed to select top-ranking keywords based on visualness similarity and word sense disambiguation. These developed methods are then evaluated based on questionnaire responses of selected keywords for a set of videos. The three developed objective scoring methods correlate well with the scores of the subjectives responses and generally outperform the traditional term frequency inverse document frequency (TF-IDF) method. The proposed LVD-F method obtains the highest precision and recall of all.

Summary

E-learning is a popular application that utilizes the recent technology of the Internet in the academic field. These e-learning systems allow students to receive education without boundaries of time and space. E-learning systems are essentially on-line repositories of learning objects (LO). Learning objects are any type of object that contains pedagogical value. These learning objects come in many different forms, starting from textual objects like e-books or lecture slides to media files such as images or lecture videos. The domain of this research are these media learning objects (MLO) created from lecture videos.

E-learning systems must allow users to be able to retrieve whichever content they might need from the whole database, in order to get the maximum personal benefit of the learning process. Many different querying methods exist, but until now the most popular method is still query by keyword. This is not an issue with LOs in the form of text, but lecture videos are more challenging. The lecture videos need to be formulated into an effective MLO, which includes annotating the videos with appropriate textual keywords to enable search by keyword. The problem is to determine which words are the best suitable to use as annotations. This process can be done manually but many disadvantages follow. The manual process is restrictive, labor intensive, and subjective, therefore an automatic method is needed.

The entire duration of a lecture video may not be entirely relevant for a user's need as an entire lecture video may contain multiple units of information. Therefore these videos can be segmented temporally into shorter segments which can later be processed individually. Next, annotation is performed on these segments, enabling search and retrieval even within a single video. The annotation itself utilizes cross document annotation concepts by including an additional text source for keyword extraction. Not all of these words are suitable to use as keywords, hence the need of keyword selection.

This thesis first investigates the performance of a common statistical scoring method, Term Frequency Inverse Document Frequency (TF-IDF). This conventional method is then improved by introducing semantics. The semantic network chosen was WordNet, due to the extensive literature about it and the vast research on it. The keyword selection is performed semantically by employing a scoring method to determine which keywords are the most meaningful. After words are scored, they are then sorted according to the descending scores, and a list of the n top words are taken as the selected refined keyword set.

This thesis proposes three (3) different methods of scoring keywords objectively, utilizing semantic concepts such as semantic similarity, visualness, and word sense disambiguation (WSD). The visualness measure is adapted to the lecture videos to be used in these methods, and two different WSD methods are used. Visualness with Disambiguation by Category (VDC) and Visualness with Lesk Disambiguation (VLD) are hence developed. Additionally, semantics are also combined with statistics, creating a combined score from visualness and frequency in the Lesk Visualness and Disambiguation with Frequency of Occurrence (LVD-F) method.

The performance of these objective scoring methods are then compared against the subjective answers. An on-line survey was conducted in order to determine which words would be chosen subjectively by e-learning system users. The Borda count scoring method is used to be able to rank these chosen words in a single list. The top words are then taken as the subjective words, which are treated as the ground truth. The results obtained by the developed objective scores are then compared to this.

These three (3) novel semantic scoring methods for keyword selection correspond well with the words selected by users, and the LVD-F method outperforms all the other methods in terms of precision and recall. The proposed semantic methods show better results than conventional scoring methods such as TF-IDF, indicating that semantic annotation is indeed necessary. WordNet is also a very robust and reliable semantic network to use for semantic analysis. Additionally, cross document annotation also proves to be a potential alternative to extract potential keywords for annotation.

Preface

First of all, I thank Allah for all His grace, for if was not for Him I would not be here. I would also like to thank my family back home: Mama, Papa, Pandu, and Karin; as well as my grandmother Eyang; for their continuous support throughout these years abroad. Next I would like to thank my fiancee and best friend, Brahmastro Kresnaraman, for all his love and encouragement for me in all I do, especially through these years apart. I love you all.

I would like to thank my supervisor Prof. Faouzi Alaya Cheikh for his help and guidance during my thesis work. I would also like to extend my gratitude to PHD fellow Ali Imran Shariq, whose input and critique has been very inspiring during the course of this research. I thank my two external reviewers, Prof. Damien Muselet and Prof. Hubert Konik, whose input on my presentation helped me complete my final report. Finally I want to thank all the other CIMET professors that have taught me during the 2 years of my master program in Universite Jean Monnet Saint Etienne, Universidad de Granada, and Gjøvik University College.

I next want to mention my CIMET colleagues of cohort 3 who have become a family I never thought I would have: Pamela, Lynn, Mike, Vignesh, Alexandra, Natalia, Annick, Torres, Medina, Kicha, Owais, Oscar, Piotr, Janet, Rahul, Alina, Alexandru, Melkamu; and especially Remy, Kiks and Tatiana. I thank you for all the times we had together, I will never forget you all. To Marialena and Vamsi, who have been throughout all 4 semesters at the same mobility with me. Thank you for all the laughs and the tears, I love you guys and will truly miss you both.

Last but not least I would like to thank everybody who has participated in my online survey. All of these people have given a great contribution to the completion of my thesis.

Laksmita Rahadianti, 15/07/2012

Contents

List of Figures

List of Tables

1 Introduction

Recent advances in technology have provided humans with alternative solutions and approaches to everyday tasks. One of which is in the education field, in which the learning process is no longer confined to the conventional classroom. These advances together with the widespread integration of the Internet worldwide make the possibility of distance learning, or e-learning, possible.

The concept of distance learning diminishes the need for a student to be present in person at a physical classroom. The learning process is instead carried out by making use of an on-line e-learning system, which can be accessed by both the lecturer and the student in order to deliver or retrieve the educational content. Sometimes e-learning is not used. Although students are still required to attend conventional classes, the e-learning system is often used simultaneously as an additional feature to enhance the learning experience.

Through these e-learning systems, students can access educational content that would normally be distributed in class; such as lecture slides, books, or handouts; in their digital form. These materials containing pedagogical value are called learning objects (LO). While slides, books, or handouts are in the form of textual data, pedagogical value can also be conveyed in different forms. Learning objects are also found in the form of multimedia, such as images and video. These types of files can also be rich in pedagogical content, forming media learning objects (MLO). sE-learning systems consisting of MLOs are essentially multimedia databases, which require certain organization and structure so that information can be retrieved effectively [2]. E-learning system users should be able to retrieve learning objects to cater their specific information needs as shown in the simplified e-learning scheme in Figure 1. It is clear here that information retrieval for such systems is crucial.

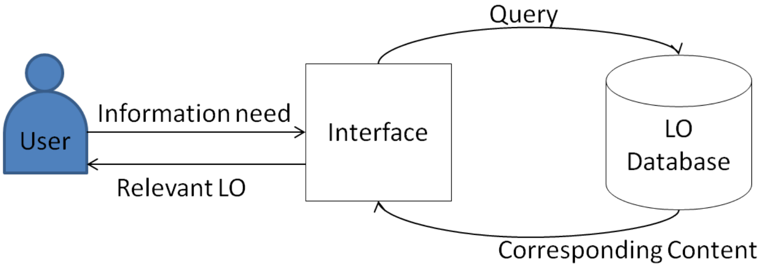


Figure 1: An example of a simplified e-learning system.

1.1 Problem Statement

It is necessary MLOs in an e-learning system to be structured optimally. An optimal MLO should be able to encapsulate pedagogical content that is able to convey lessons in the same way as conventional teaching can. As previously mentioned, it is crucial for e-learning system users to be able to search and retrieve certain content that they desire. Although various methods for multimedia retrieval exist, most users are still accustomed to retrieve information using text-based queries. Therefore, in the formation of MLOs, the necessity of textual keywords are then needed to associate with the videos for information retrieval purposes. A reliable and valid scheme for tagging learning objects is hence necessary [3]. Meaningful meta data is essential to explain the particular LO correctly, including tags and annotations.

A potential MLO can be formed by lecture videos. A video of a lecturer giving a lecture that would normally be done in class, can be recorded and later distributed. Classroom lectures are rich in pedagogical content because it can contain examples, explanations, and often also questions and discussions. A recording of this nature would form a very useful Video Learning Object (VLO) which can be distributed through e-learning systems.

In order to be able to get the maximum benefit of a certain lecture video, students need to be able to retrieve specific videos or parts of videos that are relevant to their needs. Therefore, the creation of appropriate VLOs complete with tags and annotation is necessary.

Although the whole duration of an entire lecture video usually about a certain topic, it is also most probably covers many different sub-topics. It is common for a student to prefer finding a certain section of a lecture video according to the information needed at the time, without having to go through the whole video. Therefore annotation along the temporal domain is needed. The tagging and annotation is then done on each of these segments (Figure 2).

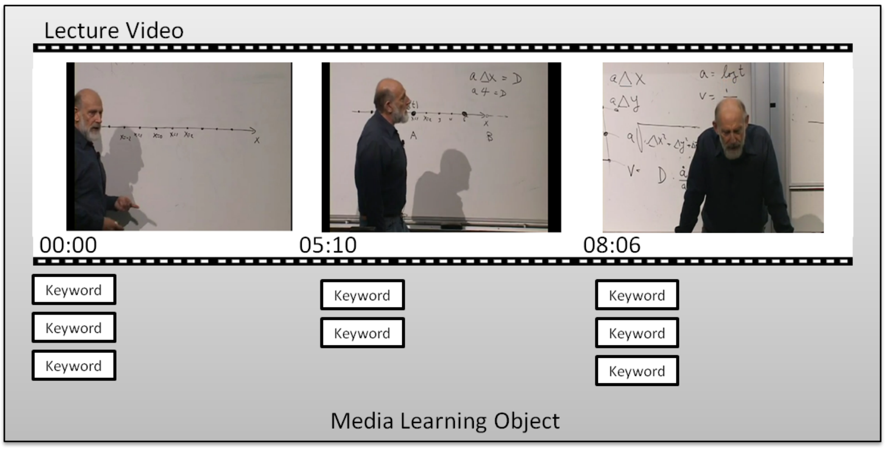


Figure 2: A design of an optimal MLO containing a lecture video. The video is temporally segmented and annotated.

This is the focus of the research presented in this report. The problem identified here is which keywords should be used to annotate the lecture video [4]. The keywords used should be able to describe the content of the video accurately. This video annotation is generally done manually, but that is not an ideal approach due to a number of reasons. Manual annotation is limited to a restricted set of words, subjective to the annotator, and very labor intensive; making an automated alternative desirable. Hence, the objective of this research is to develop methods that can both select the most appropriate words for lecture video annotation in an automated fashion.

1.2 Methodology

This research explores the concepts of cross document annotation, in which the annotation of a certain media file does not originate solely from that single file. The annotation takes into consideration multiple files that function as surrogate documents to extract potential keywords for annotation. This is logical since e-learning systems often present media learning objects together with other learning objects together in a certain topic. Additionally the notion of semantic annotation also comes in play. The idea is to have a semantic network which can model relationships between words. These relationships can be exploited in order to determine which words are more suitable or less suitable to be used for annotation.

This work attempts to create a system which is fully automatic, in order to avoid the limitations of manual annotation. The keywords are extracted and then selected accordingly to obtain the optimal set of keywords to use in annotation, while consistently maintaining an automatic nature. These methods will use very limited human input, built using a dataset of freely available lecture videos. The results of this automatic system will then be evaluated against responses obtained from a survey. An experiment will be carried out on-line with experienced e-learning system users as participants. This survey aims to obtain the words that users would actually use as keywords. The automated methods will then be assesed as to how well they predict these words.

2 Relevant Literature Review

This chapter aims to illustrate the development of research in this field. It will show progressively the different approaches that have been done in the past years and the improvements that have been achieved. All the work explained in this chapter initiated the idea of the work in this thesis and serves as the literature review which was done as the first step of this research.

2.1 Multimedia Retrieval

In its initial implementations, information retrieval (IR) was defined as finding material (usually documents) of an unstructured nature (usually text) that satisfies an information need from within large collections (usually stored on computers) [5]. In order to be able to extract the information subset which caters an individual need, various methods and processes are needed to select and refine the information. These techniques employed in this process are called information retrieval techniques.

The sources in which the requested information is sought need to be organized in such a way so that the search can be done effectively and efficiently. In the presence of this index, the search is conducted on an index and then the information accordingly from the corresponding index retrieved [6].

In the retrieval of the stored multimedia data, a user can choose between 3 methods, as follows [7].

1. First the user can simply browse the collection manually and retrieve the desired object. This method is undoubtedly not desirable due to its manual nature.
2. The second method is query by keyword, in which a user can pose terms as queries and the system will process it to retrieve relevant content.
3. The last is the query by content method in which the user will use a sample multimedia object as the query and the system shall retrieve similar multimedia objects.

Many users are still most accustomed to the query by keyword method, which is probably due to the fact that this method is commonly used in search engines on the web. Users of a system will more likely tend to express the need of information in the form of text because of familiarity and also the difficulties of composing a multimedia object as a query.

Therefore, multimedia databases require a method to be able to search and retrieve multimedia content according to a textual term. The multimedia content must therefore be annotated and tagged with certain textual terms that describes its content correctly. From here arises the question about which words would be the best suitable to associate with a multimedia object. These terms are commonly known as keywords [4].

One type of multimedia content is video. The process of labeling the videos with correct keywords is an important task. Manual video annotation with keywords by an observer is very labor intensive. Furthermore this method also introduces subjectivity due to the biased judgment of the human observer. An automated method of labeling and annotating videos is thus urgent.

2.2 E-Learning Systems

Advances in technology and the widespread integration of the Internet into society has induced many digital alternatives to every day tasks. One of which is e-learning, an alternative to the conventional classroom educational system. These digital e-learning systems provide a platform for the exchange of educational content, just as it is done in conventional teaching.

E-learning systems can be done in a synchronous and asynchronous method. Synchronous e-learning requires the teacher and student to access the system at the same time. The system then will facilitate the exchange of information using chatting, discussion or conferencing features. Although this approach does not require the participants to be in the same place, it still has limitations because they need to synchronize the time of access. This is not much different then the conventional classroom, since the information exchange is still done directly from teacher to student, with the system acting as an intermediary [8].

The asynchronous approach is probably the most appealing characteristic of e-learning. Using this approach, the teacher and student are not required to use the system at the same time. The knowledge exchange is done with the help of certain digital objects which hold pedagogical value. This enables students and teachers to be able to organize their time between study and other priorities such as work ad family. Students can access the content at any preferable time and study with the help of these digital objects.

The digital objects used to convey instructional content plays a very important role in the learning process. Actually, even in the execution of conventional education, additional material is usually distributed to students as well, although it does not necessarily come in the digital form. Many instructors assist the students by providing a copy of the lecture slides. Nowadays, lecture slides are usually created digitally, so a digital form of slides can easily be incorporated to become a digital object to distribute in an on-line e-learning system. These digital objects that contain pedagogical value are commonly known as learning objects [9].

2.2.1 Learning Objects

Learning objects are therefore very important in the implementation of distance learning. Since the conventional method of teaching is no longer done, the education process relies heavily on the quality of learning objects provided for the students. In order to achieve a good educational outcome, it is very important to have good learning objects.

The form of learning objects are not limited to only the lecture slides as mentioned in the previous example. The possibilities range from the lecture slides to a digital copy of the course textbook. Although these examples are clearly very useful to aid students at a distance, they are not the only type of objects that are useful.

2.2.2 Media Learning Objects

While learning objects are any type of object with pedagogical value, media learning objects (MLO) are, in turn, the learning objects in the form of multimedia, such as videos [10]. Although lecture slides or books may provide a rich source of knowledge, this type of content would not include the explanations given orally as well as the discussions and questions that arise during a face to face lecture. For this reason, a video of the whole lecture session would provide a large educational value [9]. This thesis will focus on these lecture videos used for instructional or educational purposes.

The retrieval techniques needed for these videos make use of certain annotations and indexes built on these lecture videos. The methods are similar to those used in generic multimedia database indexing, although the unique nature of lecture videos must be considered. The main issue is that lecture videos have little if any structure in the temporal domain in regard to its content, making many of the usual methods of search and retrieval in multimedia difficult to implement.

Structuring an effective MLO from a video is challenging. First of all, the whole duration of a video itself must be restructured in order to be able to organize the knowledge content. Students usually have a very specific information need while searching for material, and it is preferable to be able to find that particular content without having to search through a whole video manually. Therefore videos are usually segmented into shorter shots.

A lecture video is often only a continuous shot of the lecturer giving the lecture with little means to determine structure. In comparison to movies and news videos, in which the transitions between shots is quite clear with the change of scene, lecture videos have nothing of the kind.

Movies have clear shot transitions, due to camera perspectives or scene changes (Figure 3), and news videos also have distinguishable scenes that can indicate different shots (Figure 4).

Meanwhile, most lecture videos are captured in a non-professional manner, and does not go through editing and processing afterwards [11]. As a comparison, a screen shot of a lecture video is shown in Figure 5. This visual does not change much throughout the duration of the video.

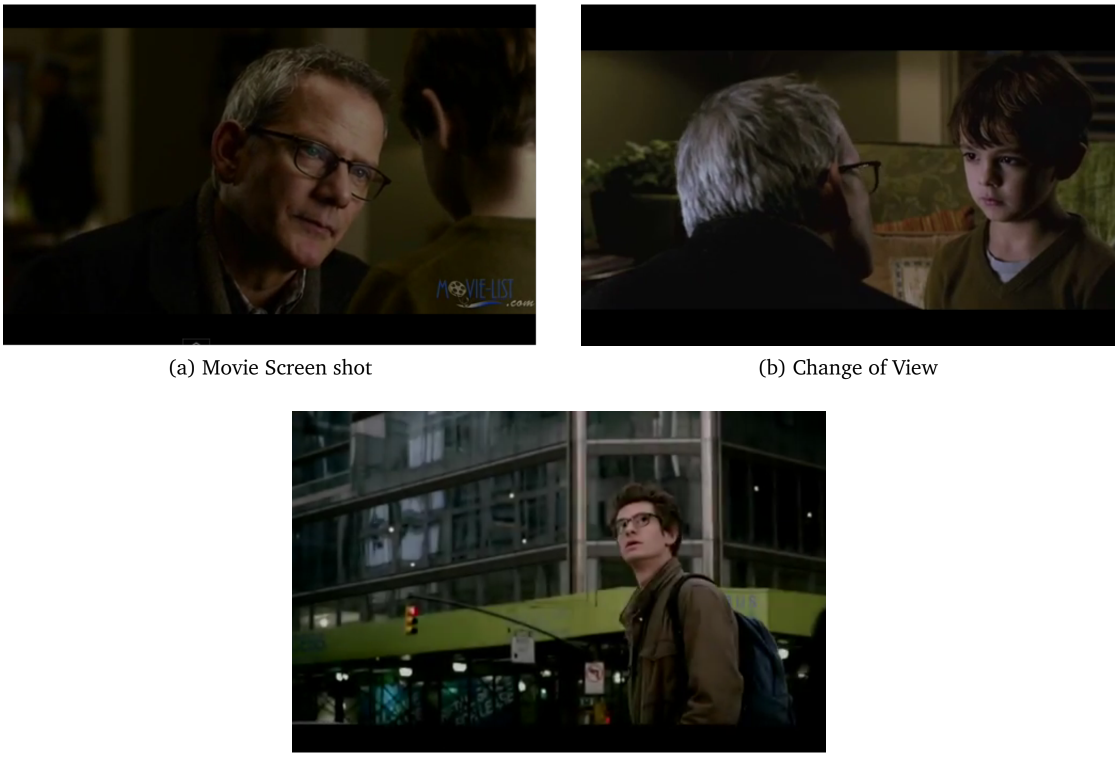
Some attempts to structure a lecture video and determine the shots have been done. This usually depends on what type of teaching aids the lecturer is using. Different teaching aids provide different types of lecture content, such as blackboards, slides, white boards, or other types of teaching aids.

Blackboard content is quite commonly used in many lectures, and various works have been carried out on this type of data. In the research of [12], the lecture videos used were those that used a blackboard in the class. The work in [12] processes the video with temporal segmentation by detecting the time at which the lecturer erases the content of the blackboard. This erasing action is assumed to be the indicator of the end of one segment and the start of the next. Many different approaches exist as alternatives, and each is specific to the type of content area.

Blackboard-based lectures are now slowing getting replaced with digital teaching aids, such as slide shows. This setup is used in the approach in [13], where the lecture is assumed to use the aid of power point slide shows. The processing is then very much different from the blackboard processing. Here the temporal segmentation is done according to slide transition.

The drawback of the implementation done in [13] is that it needs 2 video streams, one of the lecturer in front of the class, and the other stream is focused only on the slide show. The slide show stream is used to detect slide changes, which are used as shot boundaries. This approach is obviously not applicable to many lecture video setups that only have a single stream of video.

Due to these multiple possibilities of setup for a lecture video, the approaches developed up until now are usually specific to a specific setup. These specific solutions are not generally



(a) Movie Screen shot

(b) Change of View

(c) Change of Scene

Figure 3: Different shots of a movie. Clear transitions can be seen between the shots.

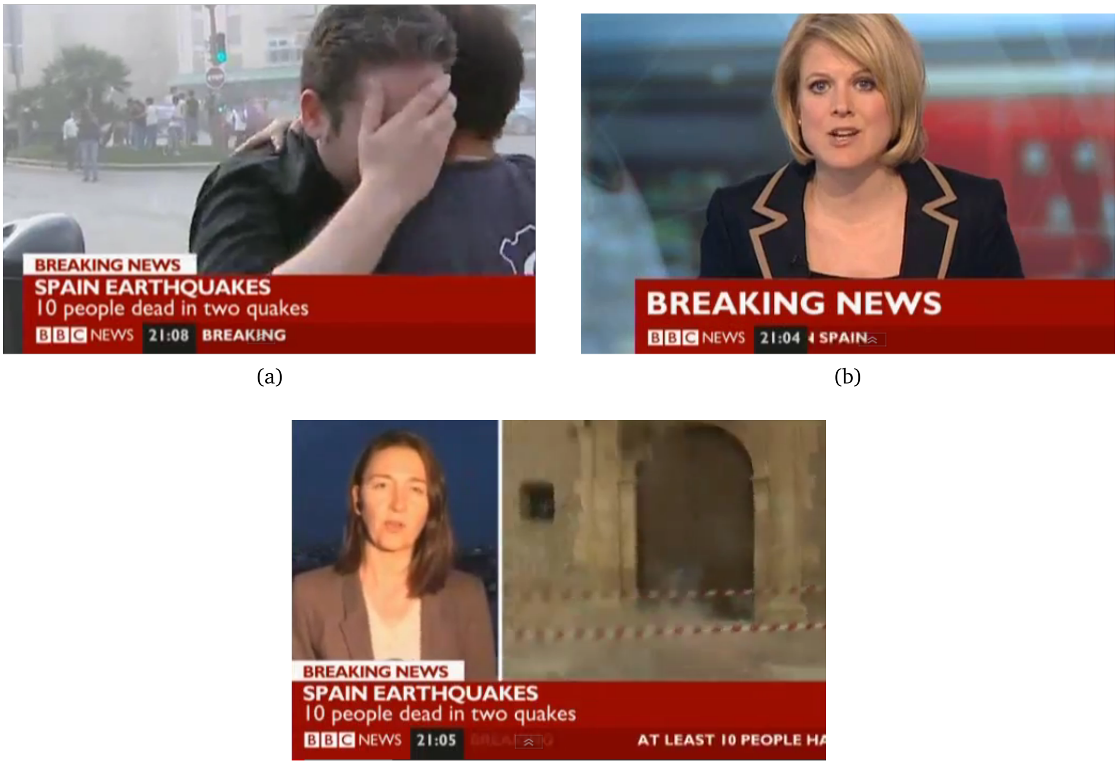
applicable to all lecture videos, but as there are no defined standards on lecture video capture, it seems unrealistic to establish one solution that can solve all types of lecture video.

2.3 Video Annotation

As mentioned in chapter 1, it is common in multimedia databases that users pose queries to the system for relevant content using textual terms. In order to be able to use these textual terms to retrieve the correlating video segments, there is a need to annotate the video with text as well. Initial works in this field include many different classification methods in which each image or frame will correspond exclusively to a concept; or even multi-level classification in which each image or frame is associated with multiple concepts [14].

Video annotation can indeed be considered as a form of video classification but is more commonly done on the temporal video segments. While video classification will classify videos to predefined categories according to the video's features, video annotation is defined as allocating video shots or segments to different semantic concepts, not categories, such as indoors, person, boat, etc. A video segment can be annotated with multiple concepts or words, not just a predefined category as done in classification [15].

Due to the idea of annotation using multiple concepts, video annotation can be divided into



(a) (b)

(c)

Figure 4: Different shots of a news video. Clear transitions can be seen between the shots.

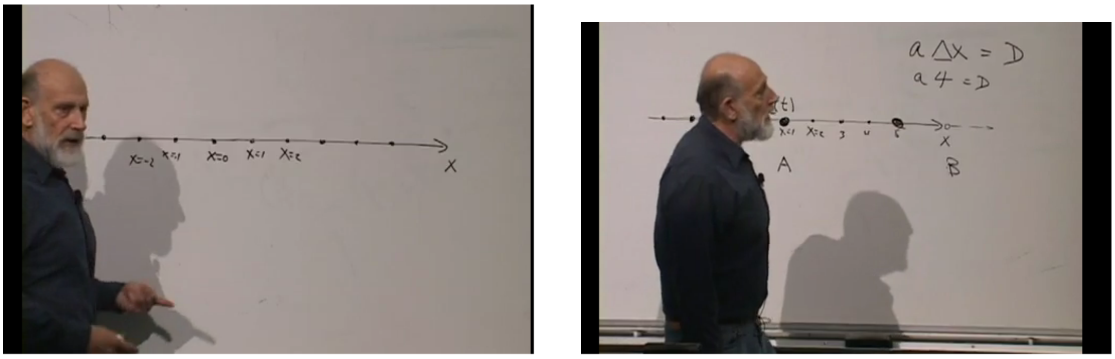
3 methods :

1. Isolated concept based annotation. This method trains a certain classifier for each individual concept, and each of these binary classifiers determine whether or not the corresponding concept is applicable for annotation. The obvious limitation of this approach is that each concept is treated independently and the relation between concepts is not modeled.

2. Context based annotation. This approach attempts to lift the binary classifiers to higher level concepts. This method also acknowledges the relation between concepts. Detected events or entities are associated with higher level concepts using higher level semantics. This approach also requires a certain ontology or semantic network to infer these associations and relations.

3. Integration Based Annotation. This method models both the individual concepts and their relations.

In wide domain videos, the annotation is done by extracting features of the video scene to recognize certain concepts or objects. However in lecture videos, association between features and concepts are not easily feasible due to the lack of visual content it contains. Lecture videos contain the lecturer with the teaching aids which do not have much correlation to meaningful content of the video itself.



(a) (b)

Figure 5: Different shots of a lecture video. No clear transitions can be seen between the shots.

Therefore in order to determine which words, in fact, would be suitable to be used as keywords to annotate a lecture video, there needs to be a source of potential keywords. Keeping in mind the unique nature of lecture videos, potential keywords are more often extracted from text in the lecture content of the video (slides or blackboard) [13].

This process usually start with attempting to isolate the textual content of the lecture, such as blackboards. As the lecturer is frequently writing, a preliminary step is sometimes needed to do background and foreground segmentation, in which the moving lecturer in front is separated from the background which contains the blackboard. Approaches such as [16] and [12] explain methods in which this segmentation can be done.

After obtaining the content areas, the area as processed to extract potential words or concepts. This can be done using classifiers to predefined concepts, or also optical character recognition (OCR). OCR attempts to identify the characters on the content area and hopefully exract text and words.

This is a very tedious and complicated task due to many reasons. First of many is that in the case of blackboard content, there is no structure of lecture content at all. Slide shows, or electronic visual aids made using computers, has an advantage of having text is a typeface which is computer-generated. Many attempts such as [13] and [17] have been researched, but correctly identifying and extracting words from an image frame is still a big challenge.

2.3.1 Cross Document Annotation

In current e-learning systems, certain organization and categorization is apparent in the structure of the learning content. It is very seldom that an MLO is delivered on its own. Media learning objects in e-learning systems are commonly organized together with other types of learning objects within the same topic or lecture[10].

This organization will imply the relatedness of these documents, so that external files can be used to annotate the MLO. These external documents can vary from the lecture slides, transcripts, or other material such as video subtitles or transcripts. These documents can be used in the annotation of the video, consequently enabling cross document annotation [18].

Supporting documents can possibly be in textual form. This is a great alternative to the video processing with OCR that does not guarantee a correct text extraction. These documents provide a source of potential keywords for the MLO.

In previous publications, the annotation of a certain media would be focused on that single media as an individual source. Considering the nature of the related content in an e-learning system, it seems to be such a waste to concentrate solely on a single media itself. Therefore, the next step is to include the additional documents aside from the video itself when doing the video annotation.

2.3.2 Semantic Annotation

The idea of a semantic annotation can enrich the annotation massively. Semantics is a branch in linguistics that focuses on the study of meaning. Conventional video annotation maps features to concepts. This approach relies on statistical classifiers and learning methods. By introducing a semantic processor, it will attempt to lift the meaning of the mapped concepts from the simplest form of low-level features to a higher semantic level of understanding.

Real objects and concepts in this world don't usually exist on their own, but instead they have relationships between themselves, creating a semantic network of concepts. Using this so-called commonsense understanding, it is possible to correlate concepts and create a richer and enhanced set of concepts.

Many researches have been carried out utilizing this semantics combined with visual features from the video using wide domain videos such as [19] or [20], as well as news videos such as in [14]. It is important to remember that in lecture videos, there are very little visual features to work with, so the semantic analysis must be adapted accordingly.

The relationships between concepts is essential to this approach. For example the concept of ”water” is strongly related to ”ocean” and ”sea”. This calls for a necessary network of concepts which correctly describes the interrelation between them. A manual ontology can be developed to a certain video, but a general network of concepts is ideal to apply on any type of video.

2.4 Semantic Concepts

In order to be able to understand the implementation of semantic annotation done for this thesis, there are some important semantic concepts that must be covered. These concepts are WordNet, semantic similarity, visualness, and Word Sense Disambiguation (WSD) that will be elaborated in this section.

2.4.1 WordNet

In this research it was decided that the semantic analysis would be carried out using a preexisting semantic network, WordNet. WordNet was developed by Princeton University as a lexical database for the English language. Humans are already familiar with dictionaries as word databases, but dictionaries that exist now are easily understood by humans, but not so much by machines. WordNet attempts to create a word database of English terms understood by humans but structured in such a way that it can be processed by machines [21].

WordNet attempts to establish semantic meaning to words using methods understood by a computer, distinguishing between 4 different parts-of-speech (POS), namely verbs, nouns, adverbs and adjectives. In the WordNet hierarchy, words are grouped into synsets, which is the minimum fundamental unit of WordNet. Words grouped into the same synset are said to have the same meaning [22].

WordNet also demonstrates the relations of synonymy, antonymy, hyponymy, meronymy, troponymy and entailment. This rich relation knowledge provided by WordNet is very useful in any semantic processing system. A visualization of the WordNet structure is shown in Figure 6.

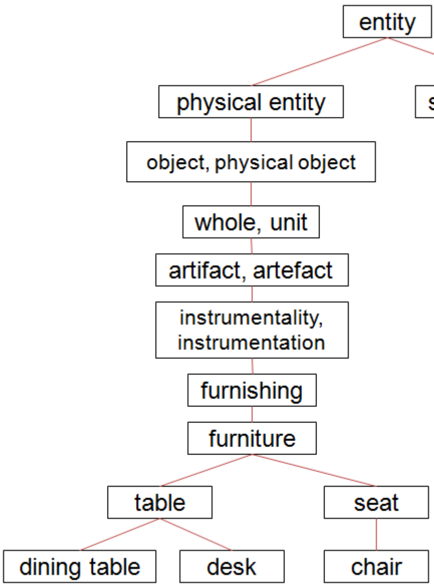


Figure 6: Visualization of the WordNet semantic network as a graph.

WordNet provides a user-friendly graphical user interface (GUI) that can be downloaded and installed freely. Another alternative to access WordNet is by a command-line interface (CLI). These interfaces are mostly used for human access to WordNet. In order to be able to use WordNet in programming a larger application, many implementations of WordNet have been previously developed into usable programming packages. Some of these programming packages are the Java WordNet Library [23], the MIT Java WordNet Interface [24], encapsulated in the Natural Language Toolkit for Python [25], WordNet querying system with SQL [26], and others.

These packages provide a black box to the linguistic characteristics of WordNet to a usable function list which provides all the linguistic features and processes that might be needed in semantical analysis. Most of these ready to use packages to access WordNet are freely available on the web. It is possible to choose the appropriate platform to develop an application on top of WordNet.

2.4.2 Semantic Similarity

Semantic similarity between two words in the dictionary is the measurement on how similar or how related they are. The implementation of semantic similarity can be done using a network of semantic concepts by taking into consideration their relative positions in the hierarchy. The semantic network used in this research is WordNet.

WordNet also categorizes words according to parts-of-speech, namely nouns, verbs, adjectives, and adverbs. The hierarchical ”is-a” relationship is only defined on nouns and verbs, while this relationship is crucial in determining the relation between word, which is in turn is needed in the calculation of similarity. Therefore the similarity measures can only be calculated on nouns and verbs. Additionally, as the ”is-a” relationship does not cross part-of-speech boundaries, it can also only be performed on pairs of the same type [27].

According to this taxonomy, similarity between two synsets can be calculated in two ways. The first method is according to path length, by counting the path between each synset on the tree. The second is by calculating the information content of the least common sub sumer (LCS). The LCS is the most specific term that is a parent of both synsets [28].

Although different similarity measures are available, this research is limited to only the Jiang and Conrath measure which is calculated based on the information content. According to [28] as well, Jiang and Conrath performs the best between the methods utilizing information content. Due to this fact, this research will base all semantic similarity calculations on this method.

2.4.3 Visualness

The concept of visualness is very important to this process. As explained in Section 2.4.2, the relatedness of two words can be measured with semantic similarity. In the issue of video annotation, simple semantic similarity is not sufficient to determine which words are best suitable for the video.

In order to quantify the capability of visual illustration of a word in a certain context or video, a visualness measure was introduced to calculate it. The main aim is to process the words so that the words with the highest visualness would be chosen to be used in the annotation. Different words have different visualness in a certain context. For example, in a sports video, most likely the word ”ball” and ”goal” would be visual in this context, whereas ”number” would not. Inversely, in a mathematics lecture video, ”matrix” would be visual, but ”ball” and ”goal” would not.

Visualness calculations involve a certain set of seed words, which will have the visualness value of 0 and 1, being either visual or not. This basis of seed words is usually determined manually, as well as their visualness value. The first step in visualness calculation is determining a set of n seed synsets, or seed words, denoted by $S = s_1, s_2, ...s_n$. Each seed word will have a determined visualness value $vis(s_i)$ for every $s_i \in S$ of either 1 (visual) or 0 (not visual) ini the video [14].

The visualness value of all other potential words will then be calculated against these seed words, resulting in a value between 0 and 1. The higher the visualness value indicates a more relevant word. Based on those seed words $s_i \in S$ and seed word visualness values $vis(s_i)$, the visualness of each potential keyword w can be calculated.

The visualness formula is as follows:

$$vis(w) = \sum_i vis(s_i) \frac{sim(w, s_i)}{\sum sim(w, s_i)},$$

where $vis(w)$ is the visualness value of a given word w. $vis(s_i)$ is the visualness of seed $s_i \in S$ while S is the set of n seed words. The visualness of these seeds are annotated to either 1 or

0 depending on the presence if it in the video, which is denoted by $vis(s_i)$ for each seed s_i in the set of seeds $S = \{s_1, s_2, ...s_n\}$. The calculation uses semantic similarity $sim(w, s_i)$ between the word w in question and every seed word $s_i \in S$. The value of i ranges from 1 to n iterating through the whole list of seed words $S = \{s_1, s_2, ...s_n\}$.

The main problem is to select the best possible pre-defined set of seed words for a video $S = s_1, s_2, ...s_n$, as well as their corresponding visualness values $vis(s_i)|s_i \in S$.. In [14], this is done manually by human input. This formula must be adapted to the automatic requirements of this research and this adaptation will be explained later in Section 3.5.

2.4.4 Word Sense Disambiguation

Another consideration to pay attention to is the structure of WordNet which provides different senses for each word. Although the access of words through WordNet has already been limited to a certain part of speech (POS), even within the same POS it is possible for a word to have multiple meanings, or senses. It is therefore necessary to determine which sense of a word that is relevant in the context by doing word sense disambiguation (WSD).

As an example, the word ”index” has the POS of both verb and noun, and in the noun form it has 5 senses, as the following.

1. (1) index – (a numerical scale used to compare variables with one another or with some reference number)
2. (1) index, index number, indicant, indicator – (a number or ratio (a value on a scale of measurement) derived from a series of observed facts; can reveal relative changes as a function of time)
3. exponent, power, index – (a mathematical notation indicating the number of times a quantity is multiplied by itself)
4. index – (an alphabetical listing of names and topics along with page numbers where they are discussed)
5. index, index finger, forefinger – (the finger next to the thumb)

Provided the multiple senses of a word, it is then crucial to determine which sense is the sense intended in the context of the video. Many different methods exist, but this research will use specific methods suitable to its needs. This approaches will be explained in Section 3.5.

3 Automatic Semantic Annotation

This research revolves around the development of a system which can take a lecture video as an input and create an efficient media learning object (MLO) as a final product. The system implemented is intended to be an automatic process, in which the involvement of human input is minimal. The system will be explained as an overview, then each sub-system will be elaborated in detail.

3.1 Proposed System Framework

As mentioned in Section 2.2.2, the setup and type of lecture video is important to determine the type of processing. For example, lecture videos which contain the lecturer moving in front of a board needs additional processing to separate the moving foreground from the board with content such as in [16] or [12]. Additionally the type of content area also determines how to proceed with a lecture video, whether it is a blackboard, white board, or a digital slide show [13].

Due to this fact, the lecture videos used in this research were limited to a certain database. This research bypasses the need to do preliminary processing in order to obtain the content areas of the video by using videos which contain only the content on its own. The lecture videos used in this work were taken from the Khan Academy [1] website because they show the content areas directly (examples shown in Figure 9). This database of lecture videos will be explained further in Section 4.1.

The automatic annotation system implemented consists of two main sub-systems. The first of which is the temporal segmentation and the second is semantic keyword selection. The overview of the system framework is shown in Figure 7. The input needed is a full-length lecture video together with the title and the transcript file available from the Khan Academy website. In the semantic keyword selection process, the categories of Khan Academy lectures are also needed as an input.

The final product resulting from this system is a Media Learning Object (MLO) with appropriate annotation along the temporal domain of the video. The full length lecture video will be annotated at each individual lecture segment with a refined set of meaningful keywords. The main source of these keywords is the transcript file with additional semantic processing.

3.2 Temporal Video Segmentation

The temporal segmentation process is the first sub-system needed for this framework. In e-learning applications, it is essential to organize educational content in such a way to enable easy search and retrieval for the student users. Forming efficient MLOs from lecture videos involves organizing the video according to its content to give maximum benefit for students. It is briefly mentioned in Section 2.2.2 that a whole video in its full duration is most likely not the optimum representation as an MLO.

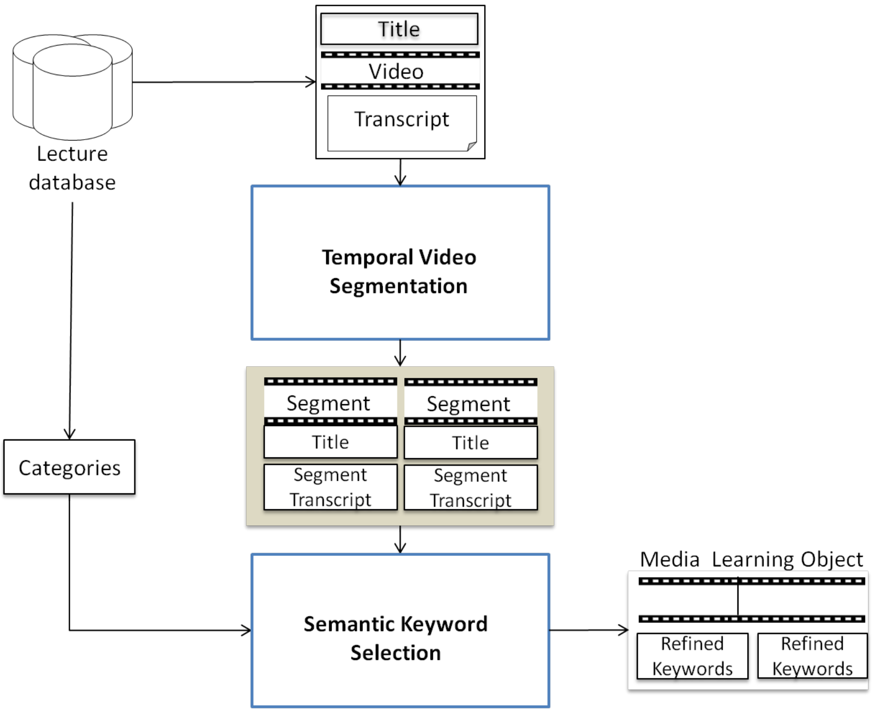


Figure 7: The proposed system framework used in this research.

A full duration video consists of different shots or segments, which on their own represent the smallest unit of information in the video. Each shot is a section of the whole video which is assumed to contain a singular concept, and different shots should have the least similarity possible, resulting into individual segments containing different concepts. This will allow users to access desired video segments without the need to watch the whole video, providing the maximum benefit of the video [10].

Since the system in its later stages will employ the concept of cross document annotation, the transcript of the video plays an important role for the video annotation process. The transcript itself is for the duration of the whole video, whereas it is vital to be able to process the video in segments. Therefore the temporal video segmentation is later followed with transcript segmentation, which ultimately ends with video segments with their corresponding segment transcripts. The process flow is illustrated in Figure 8.

3.2.1 Shot Boundary Detection

Over the years, multiple methods have been developed in the pursuit of a solution for temporal video segmentation. These methods aim to detect abrupt cuts as well as the more subtle transitions. It is very difficult to create a unique solution for all types of video, due to the very different features contained in the videos. Due to the nature of lecture videos as mentioned in Section 2.2.2, these videos are a special case. Since this system was built using the database from Khan Academy, the temporal segmentation implemented is specific to the characteristics of those videos.

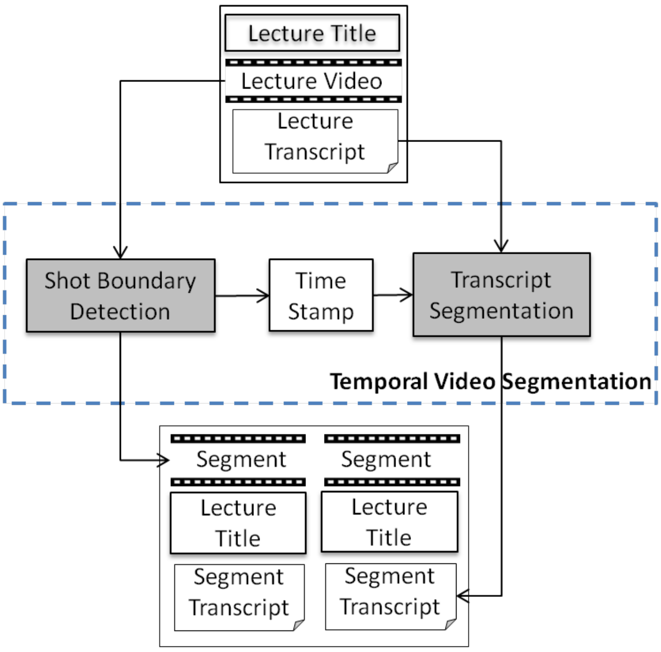


Figure 8: The temporal segmentation sub-system.

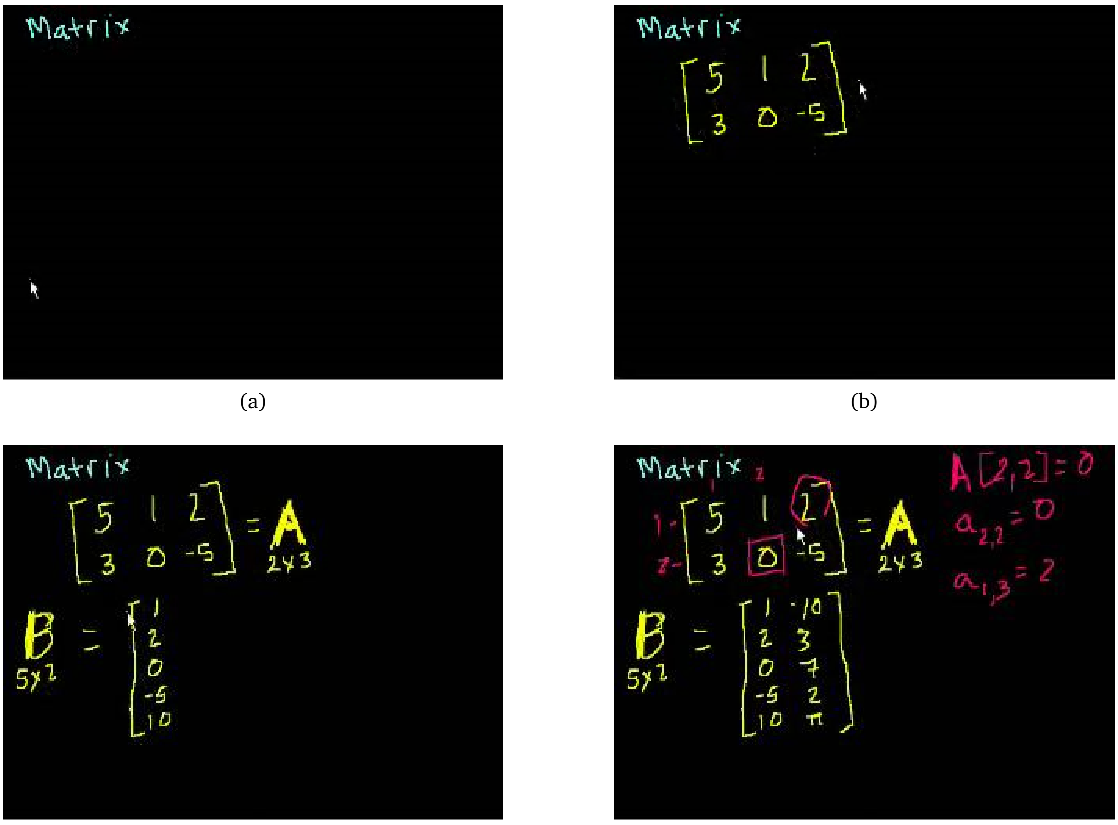
Although videos may come both in the compressed and uncompressed forms, this research deals with uncompressed videos only. Algorithms for uncompressed videos deal with successive frames of the video and treat each frame as an image. A similarity or difference measure is then employed between each pair of successive videos, and large differences are treated as cuts [29].

Among the various comparison methods, standard methods to compare frames in uncompressed videos include pixel, block based, and histogram comparison. In the effort to determine the best suitable method, it is necessary to study the specifics of the dataset in use. An example of a succession of video frames in the dataset used is shown in Figure 9.

The frames show that the content is relevant in the entire frame, indication that the whole frame itself is the region of interest (ROI). Therefore the processing is to be done on the whole frame. Luminance based approaches aim to detect a change in luminance when scenes change in the video, and the lighting conditions change as well. The videos used in this research are constant with no apparent scene change, hence they do not exhibit this change of scene and lighting.

Further examination of the content in Figure 9 also show that the content is in the spatial domain of the frame and is always changing while the lecturer adds content. The relevant content covers the frame entirely, and blocks of the video frame are not entirely relevant, so the block approach is not pursued.

The main aim in detecting shot boundaries is detecting discontinuities between successive frames. The pixel-to-pixel comparison is an effective method to do so. This method is straightforward, comparing the change in pixel values on corresponding pixels. This method is time and resource consuming since so many comparisons must be made, and the videos used in this re-



(a)

(b)

(c)

(d)

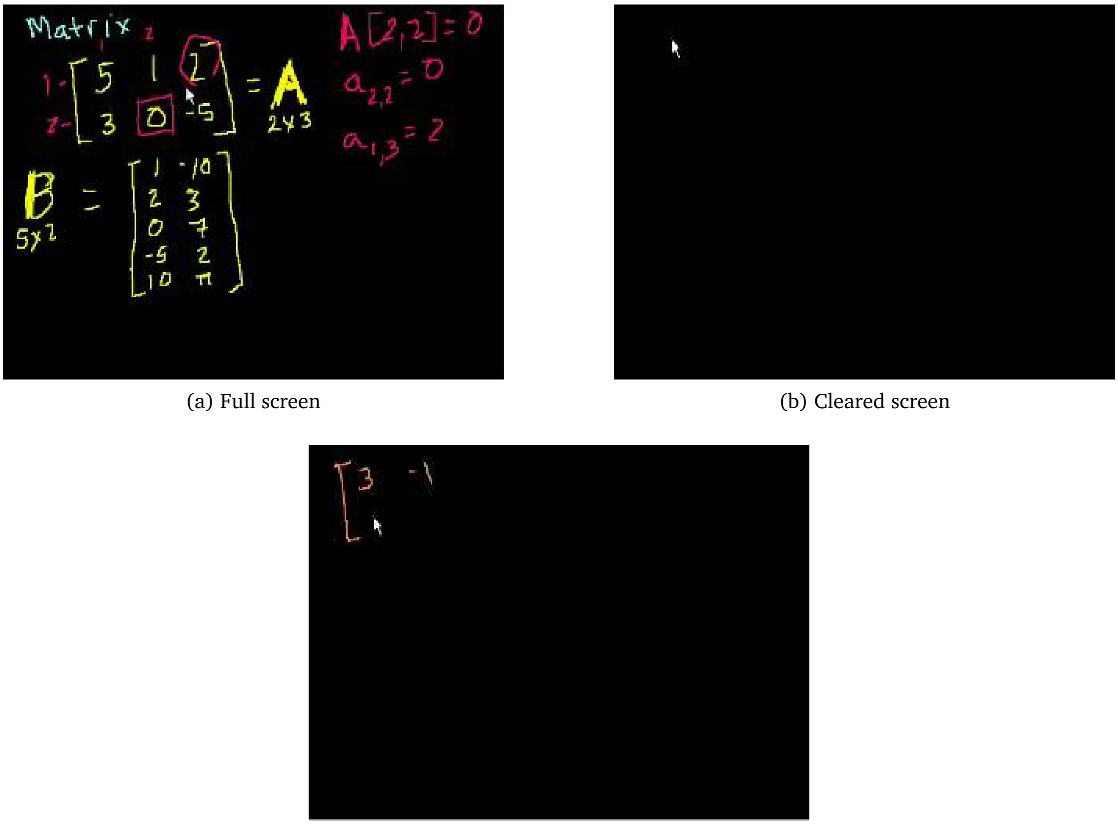
s

Figure 9: Some frames from a video instance from the Khan Academy dataset [1].

search exhibit little change at the pixel level and large portions of the frame is constantly uniform as the background. At certain intervals, the video frames in the used videos are cleared before the lecturer starts a new explanation (Figure 10).

The idea behind histogram comparisons is that similar frames will have similar histograms. The gradual addition of content in this database should also have little effect on the histograms. Meanwhile, the abrupt changes which happen when the lecturer clears the screen for a new explanation can be detected by a drastic change in the histogram. As the colors used in the lecture have no importance in this histogram comparison, the frames are converted to gray scale, and the histograms are built from the gray scale values only. This drastic change is shown in Figure 11.

The histogram values are a 256 dimension long feature and each dimension contains the number of pixels with the corresponding pixel value $H = [h_0, h_1, ...h_2 55]$. The content of the 0-th dimension of the histogram feature is the number of pixels with a gray value of 0, and so forth. These histograms are compared by doing the sum of absolute differences (SAD) between



(a) Full screen

(b) Cleared screen

(c) Beginning of a new segment

Figure 10: An example of a cut in a lecture video.

successive histograms H^t and H^{t+1} as follows:

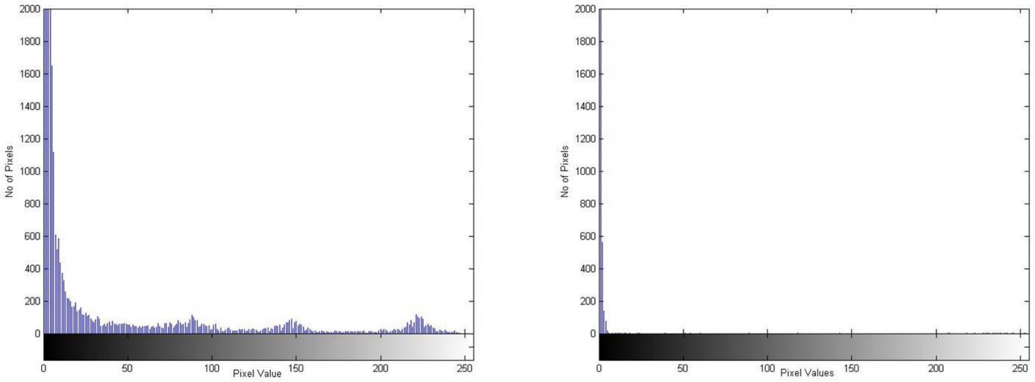
$$SAD(H^t, H^{t+1}) = \sum_{i=0}^{2} 55(abs(h_i^t - h_i^{t+1})),$$

where H^t is the histogram at time t and H^{t+1} is the histogram of the following histogram at time $t+1$.

The plot of the SAD against the temporal frames is shown in Figure 12. By applying a threshold value, any SAD of a frame surpassing that threshold value is determined as the shot boundary. This threshold is determined empirically against a number of videos from the database in use.

3.2.2 Transcript Segmentation

To enable the next step of annotation which utilizes the transcript, the full-length transcript must also be segmented accordingly. This is possible because the transcript file provided from the Khan Academy website is a subtitle file which contains the transcript with the corresponding time stamps at which they occur. This is called a subtitle item.



(a)

(b)

Figure 11: Histograms of two successive frames in a lecture video.

A single subtitle item contains 3 components:

1. Start time
2. End time
3. Text

This process goes through the text file and constructs a list of subtitle items which are ordered sequentially. As the shot boundary detection process in Section 3.2.1 gives an output of time stamps at which the shot boundaries have been detected, this time stamp is used to determine where to cut the transcript. The program goes through the list of subtitle items to find the subtitle item with the shot boundary time stamp. The transcript is then segmented at those cuts. The result of this process is multiple segment transcript files corresponding to each video segment.

3.3 Semantic Keyword Selection

The second sub-system in the system is the Semantic Keyword Selection sub-system. This part is essentially the determination of the words to be used to annotate the video. AS previously mentioned in Section 2.3, lecture videos present with a challenge due to the lack of visual cues and objects. The system attempts to utilize the concepts of cross document annotation and semantic annotation to overcome this challenge.

This sub-system is broken down into 2 steps, first is the keyword extraction, then the keyword refinement (Figure 13). The refinement is done using semantic analysis on the initial list of extracted keywords. This semantic analysis itself is done using WordNet [22], more specifically using the commercial package in the Java programming environment Java WordNet Library [23] and the MIT Java WordNet Interface [24].

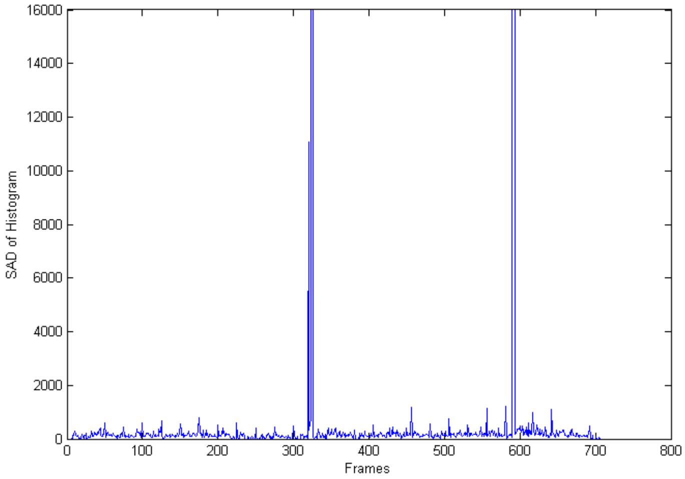


Figure 12: Sum of absolute difference between two successive frames.

3.3.1 Keyword Extraction

The main source of keywords is the transcript file of the lecture. Assuming all the words from the transcript are potentially suitable to be annotate the video, the text of the transcript is the processed to obtain the initial list of potential keywords. The process is illustrated by Figure 14.

Transcript Processing

The processes applied to the transcript in order to extract the potential list of keywords are the following:

1. Tokenization. The transcript is tokenized to obtain a list of individual words. This is done by firstly removing the punctuation in the text then separating each word according to white spaces.
2. Removing the stop words from the text. Stop words are very commonly used words in the English dictionary which provide very little value of meaning to context. Examples are words such as “a”, “the”, and others of the same kind [5].
3. Bringing each word to its singular form. Since the access of WordNet is only indexed according to the singular form of the word, the MIT Java WordNet Interface (MIT JWI) used in this research [24] provides the function to stem the word to the original form. This is valid for plural forms of a word. Using this function, all plural forms are replaced with the singular form.
4. Removing multiple word occurrences, so that each word will be present in the list only once.

At the end of this process the final result is an initial list of all potential keywords from the lecture transcript. For the ease of understanding, this list will be denoted as $W = \{w_1, w_2, w_3, ...\}$.

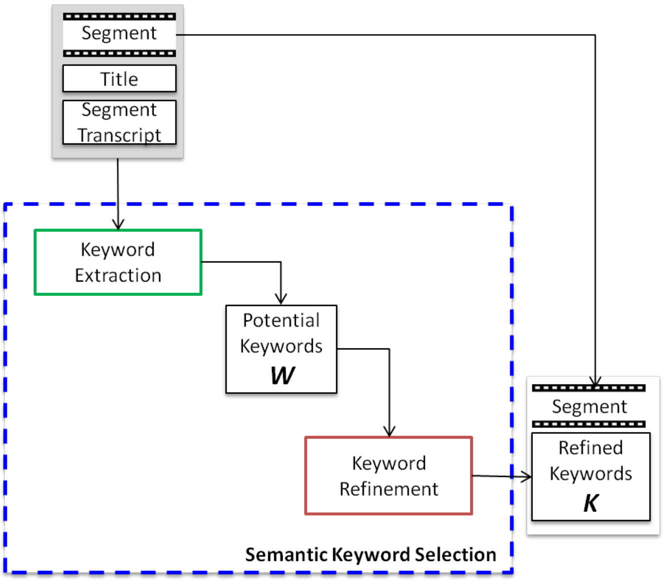


Figure 13: Semantic keyword selection sub-system.

Part of Speech Processing

In the field of semantic processing numerous publications such as [28] and [30] exist, and the focus is on the use of semantic similarity methods used for numerous semantic applications. It is commonly argued in this field of research that language semantics are mostly captured by nouns. This is also arguable in the case of semantics on visual media, as the first things noticeable are usually things or entities in the scene. Nouns denote physical word entities, and hence are most often used in semantic applications.

The list of words in obtained from the transcript analysis process are next analyzed to determine if the words can be *lemmatized* to a noun form. A lemma is the most basic form of a word [5]. It is possible to use the MIT JWI functions to lemmatize the word according to part of speech. Going through the list of words W, each word is lemmatized according to the noun POS, so the word will be taken if it proves to have a lemma which serves as a noun, or else it will be discarded, resulting into a list $N = n_1, n_2, ...n_n$.

The MIT JWI also provides the possibility to obtain derivation forms of a word. For example, the noun calculation can be derivated related to the verb calculate, and so on. This property is established in the WordNet structure by having pointers between words that have this relation.

As an expansion to the usual processing of nouns, this research tried to utilize all information possible. Due to that, this research also expanded the keyword domain to the other POS as well. The list W is processed next to identify all other POS, either verb, adverbs, or adjectives $O = o_1, o_2, ..o_m$. The system then searches for noun derivations of these non-noun words in O. If a noun form exists for any non-noun o_i, the noun derivation of o_i will be added into the list W.

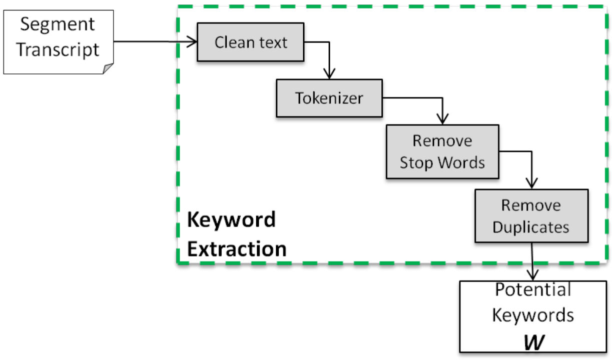


Figure 14: Keyword extraction block diagram.

Compound Words

An additional processing is done to detect the occurrences of compound words in the transcript, such as ”arithmetic mean” or ”black hole”. These phrases are commonly known and also exist in WordNet, so to use them as keywords for annotation makes sense. The processing is done by concatenating successive words, and checking if they exist in WordNet. If the phrase is contained in WordNet, it is then added to the list W.

3.3.2 Keyword Refinement

In this section the initial list of potential keywords W is put through a semantic analysis process in order to score each word on how relevant or meaningful it is in consideration to the video. The main objective of the semantic analysis is to be able to refine the list of keywords to a subset of highly relevant keywords.

The words in the word list W are processed by scoring them through a semantic analysis process. The higher the score, the more meaningful the word is in respect to the video. This research uses 4 different scoring methods to score these words, one of which is implemented from a well-known IR measure, namely TF-IDF. Additionally 3 other keyword scoring methods have been developed. All four of these methods will be explained in detail in Section 3.5. The list of keywords is then sorted according to the descending scores. A ranking cutoff is then employed in order to obtain the highest n words in the list to form the refined list of keywords K which will be used to annotate the video segment in question.

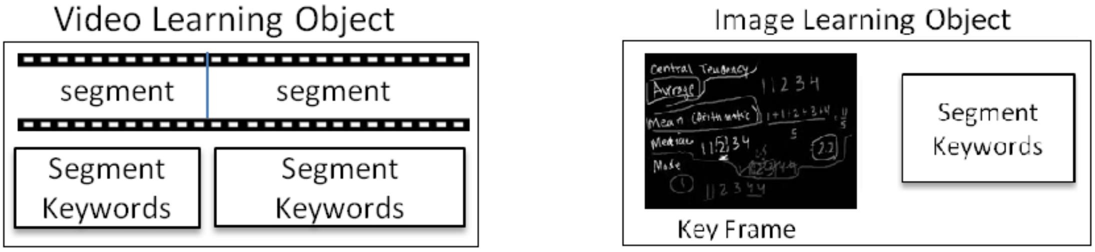
3.4 Proposed Media Learning Objects

Finally the formation of the MLO is done as the output of this whole system. The media learning object can be in two forms:

1. Video Learning Object (VLO) This VLO contains the video with the annotations along the temporal domain of the video. The full length video will not be physically temporally segmented, but rather the time stamp at which the shot boundaries occur are indicated. The refined keyword list K of each video segment is associated to the corresponding video segment time.

2. Image Learning Object (ILO) Recalling the temporal segmentation done in Section 3.2, the shot boundaries are detected by identifying a cleared screen. The last frame before a transition is therefore most commonly a screen full of content. This frame full of content can serve as the key frame representing that previous video segment. That still frame or image can be annotated with the refined keyword list K of that video segment, forming an ILO.

The resulting MLOs are shown in Figure 15.



(a)

(b)

Figure 15: Proposed design of MLO.

3.5 Proposed Objective Scoring Methods

As previously explained in Section 3.3, the initial list of potential keywords W are processed with a semantic analysis which assigns a certain score to the words. The system developed is an automatic system with minimum human input, and the process is done purely by the program, hence this scoring will be referred to as objective scoring of keywords. This research attempts to find the most accurate scoring possible in order to be able to get the best suitable words for the video annotation. This section will explain the various methods implemented in the attempt to score the potential keywords.

3.5.1 Term Frequency - Inverse Document Frequency (TF-IDF)

A popular method of scoring terms in the field of Information Retrieval is TF-IDF. This measure is purely statistical and does not consider anything about semantics. This measure shows how important a certain word is in a document within an entire corpus. Similarly, in the video segment, it is possible to compute the importance of a word in a segment within an entire video. The TF-IDF measure is defined as follows:

$$\mathrm{TF}-\mathrm{IDF}(w)=\frac{f_s(w)}{f_v(w)},$$

where w is a given word in W, $f_s(w)$is the frequency of occurrence of w in a segment and $f_v(w)$ is its frequency of occurrence in the entire video.

As this method does not take into account any semantic features, the words obtained are only based on frequency, and although important words are indeed used far much more times than less important words, many words are used often without any semantic relation to the topic and video, and hence are not very useful to be used for annotation. The process is illustrated in figure 16.

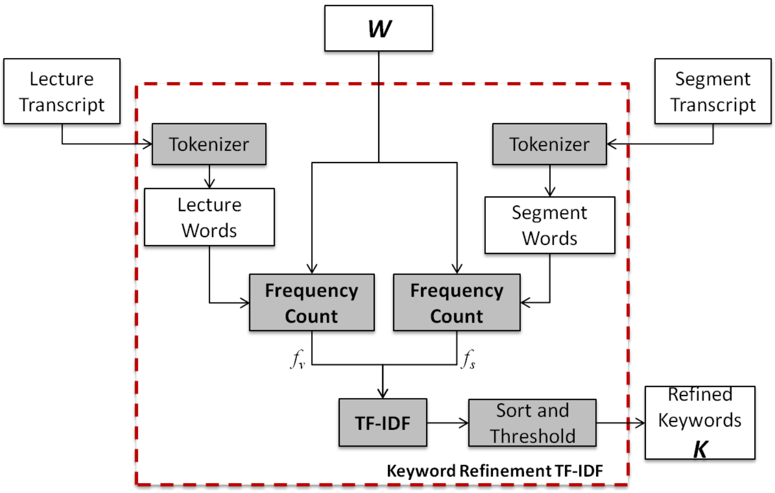


Figure 16: TF-IDF keyword refinement process.

3.5.2 Visualness with Disambiguation by Category (VDC)

This is the first approach taken in this research which attempts to utilize semantic relations between words. As mentioned previously in Section 2.4.3, semantic similarity is not used here directly, but is used in the calculation of visualness. Although the visualness concept is already defined, it must be adapted slightly in order to be able to accommodate the data and videos used in this research.

Visualness Adaptation

Referring back to Section 2.4.3, for the visualness of each candidate word is relative to the initial visual seeds. Initial works such as in [14] determine the initial seed concepts as well as each of their visualness values in each frame manually. This thesis attempts to avoid this manual determination to maintain a fully automatic system.

The idea of having a set of seed words is that the seed words are the main concepts that can be present at any given video. Continuously, these seed words shall be determined a visualness value of either 0 or 1. For example, if the seed word list were the following:

$$S = 'disease','medicine','math','physics','vector','astronomy'$$

In the case of a lecture video about cancer, these visualness values would be $vis(disease) = 1$, $vis(medicine) = 1$, $vis(math) = 0$, $vis(physics) = 1$, $vis(vector) = 0$, and $vis(astronomy) = 1$. A different example in a lecture video about planets, $vis(astronomy)$ would be 1 and all the others will be 0.

If the formula in Section 2.4.3 is revisited, it is clear that the formula of a given word w is equivalent to the sum of the semantic similarities between w and all seeds visual in the frame $(s_i|vis(s_i) = 1)$ divided by the sum of the similarities between w and all possible seeds $(S =$

$\{s_1, s_2, ...s_i\}$).

$$vis(w) = \frac{sum_i(sim(w, s_i))}{sum_j(sim(w, s_j))}$$

$$where\, s_i = \{s_i \,|\, s_i \in S \wedge vis(s_i) = 1\}\, and\, s_j = \{s_j \,|\, s_j \in S\}$$

It is crucial to determine which seed words should be used, because they will highly influence the visualness calculation. In previous research such as [31] and [14], these seeds as well as their visualness values are manually determined. In order to maintain the automatic property of the annotation, the seed concepts used are the categories of the lecture videos. The Khan Academy organizes lecture videos into categories, which can also be extracted from the lecture video page. These categories can be used as the global concepts for seeds, keeping in mind that naturally the category where the lecture falls into will have the visualness value of 1, and the other categories 0. The categories are shown in Table 1.

arithmetic	calculus	physics	currency
algebra	linear algebra	astronomy	microeconomics
geometry	biology	cosmology	macroeconomics
trigonometry	chemistry	computer science	history
probability	medicine	economics	civics
statistics	health	banking	art history

Table 1: The 24 possible Khan Academy lecture video categories [1].

Therefore for any lecture video it will have these 24 words as seeds. A lecture video within the category of algebra will have the seed visualness $vis(algebra) = 1$ and all other words will have the visualness of 0 as shown in Table 2.

Seed Word	**Visualness**	**Seed Word**	**Visualness**
arithmetic	0	physics	0
algebra	1	astronomy	0
geometry	0	cosmology	0
trigonometry	0	computer science	0
probability	0	economics	0
statistics	0	banking	0
calculus	0	currency	0
linear algebra	0	microeconomics	0
biology	0	macroeconomics	0
chemistry	0	history	0
medicine	0	civics	0
health	0	art history	0

Table 2: Example of seed words and corresponding visualness values from categories.

To add the value of semantic meaning to the video in question, additional words in the favor of the context of the video are needed. After further exploration on the Khan Academy lecture video page, more information could be extracted from the title of the video. For example, a video in the category of algebra entitled ”Linear Equations in Mathematics: Solving the Inequality” could benefit from those additional words from the title to add he context of the video. The title is processed in the same manner as the transcript as explained in 3.3.1 and the resulting words are added to the seed list for that particular video, with the visualness of the additional seeds set to 1. For the given example, the words ”equation”, ”mathematics”, ”solution” (from POS processing as explained in Section 3.3.1), and ”inequality” would be added, resulting into the seed words as in Table 3.

Seed Word	**Visualness**	**Seed Word**	**Visualness**
arithmetic	0	physics	0
algebra	1	astronomy	0
geometry	0	cosmology	0
trigonometry	0	computer science	0
probability	0	economics	0
statistics	0	banking	0
calculus	0	currency	0
linear algebra	0	microeconomics	0
biology	0	macroeconomics	0
chemistry	0	history	0
medicine	0	civics	0
health	0	art history	0
equation	1	mathematics	1
solution	1	inequality	1

Table 3: Example of seed words and corresponding visualness values from categories with additional words from the lecture title.

Seed Word Disambiguation by Category

Another problem is to disambiguate the seed words. It is important to remember the word sense disambiguation problem explained in Section 2.4.4. The seed word list S of any given lecture video is comprised of all the categories available, each set to its first sense. Hence for the set $S^t = \{s_1, s_2, ...s_i\}$ at the time t after the addition of categories and before the addition of the title. The initial senses for these categories will be $Se^t = \{Se(s_i) = 1 | \forall s_i \in S^t\}$. As these words are pre-defined before the annotation process, this will not effect the automatic nature of the annotation.

The additional words from the title of the video, however, are processed with the video. No manual determination is done on these words in order to maintain minimum user input. In order to determine the sense of these words, the similarity is calculated between each sense of the word with the pre-defined category of the lecture. The sense with the highest similarity value will be set as the sense of the seed word.

For each additional seed word u, before being added to the previous set of seeds S^t, an analysis must be done. Referring back to the category to which the lecture belongs s_cat, which is the only seed in S^t which as a visualness of 1 ($s_cat = s_i \in S^t | vis(s_i) = 1$). A sense determination is then done according to the following, for each additional seed word u with j senses $u^1, u^2, ..u^j$.

$$Se(u) = i | max(sim(u^i, s_cat)) \wedge i < j$$

$$S^{t+1} = S^{t} \cap u \; and \; Se^{t+1} = Se^{t} \cap Se(u)$$

In other words, semantic similarity is calculated between all senses of $u = u_1, u_2, ...u_n$ and the category of the lecture. The sense i which has the highest similarity to the category is chosen as the appropriate sense. The final result of this step would be 2 lists, one is the final list of seed words $S^{final} = s_1, s_2, ...s_n$ and corresponding senses $Se^{final} = Se(s_1), Se(s_2)...Se(s_n)$.

Considering a certain lecture video in the field of algebra once more, having the seeds as shown in Table 3. The lecture of the video is assumed to be ”Linear Equations in Mathematics: Solving the Inequality”. In the process of determining the sense of the word ”equation”, this particular word has 3 different senses. Therefore by calculating the semantic similarity of each sense against the category ”algebra”, $sim(algebra, equation_1)$, $sim(algebra, equation_2)$, and $sim(algebra, equation_3)$, the highest value would be for $sim(algebra, equation_1)$, and the sense 1 will be assigned to the word equation.

Refinement of *W*

Therefore, each potential keyword w_i in $W = \{w_1, w_2, w_3, ...\}$ from the transcript is scored by calculating visualness of each word according to the visualness formula, against the final seed words S^{final}. The sense of each candidate keyword w_i is not determined by context. The similarity value for w_i used for visualness calculation is then the maximum value of similarity of all the possible senses of w_i. If any one of the senses of a potential keyword has high visualness in that context, the keyword is assumed to be semantically relevant in that sense.

The final results then would be a list of the visualness values of all keywords in W according to the seed set S and seed senses Se, obtaining the list $Vis(w_i)$. After sorting and ranking every word w_i of W according to $Vis(w_i)$, the list is cut off at a certaiin rank to determine the top n words, forming the refined keyword list K. The process is shown in Figure 17, with the ”disambiguator block’ using the disambiguation by category method explained in this section.

3.5.3 Visualness with Lesk Disambiguation (VLD)

The next scoring method developed still uses the concept of visualness as in the previous VDC method. Likewise, the process of the seed word selection is also done in the same way as in VDC, as explained in Section 3.5.2. The difference between these 2 methods lie in the different word sense disambiguation (WSD).

Seed Word Disambiguation: Lesk Algorithm

The formal definition of WSD is the computational identification of meaning of words in a certain context. WSD can be considered as a classification task in which each word occurrence is assigned to its most appropriate sense based on evidence provided by the context and knowledge. Up until now, many methods of performing WSD have been investigated. To avoid going too deep into the linguistic and lexical aspect, in the interest of this research the usage of WordNet is preserved. The approach to WSD brought to use here, therefore, is the knowledge based WSD using WordNet as the knowledge base.

A well-developed way to do this is by calculating the overlap of sense definitions, with a very popular algorithm named the Lesk Algorithm. The Lesk algorithm attempts to correlate word senses with one another using the words contained in the words respective glosses. A gloss is the definition of a word which WordNet provides perfectly, nevertheless multiple glosses are present for multiple senses of a word [32].

Given any two-word context (a, b) and their senses $Se(a) = sa_1, sa_2, ..sa_n$ and $Se(b) = sb_1, sb_2, ..sb_3$, the score of overlap would be the following formula for each pair of $sa_i \in Se(a)$ and $sb_j \in Se(b)$. As a reminder, $gloss(sa_i)$ is the set of words in the gloss or definition of the word sa in its i-th sense [33].

$$\text{Lesk}(sa_i, sb_j) = ||gloss(sa_i) \cap gloss(sb_j), ||$$

with || denoting the number of overlapping elements of $gloss(sa_i)$ and $gloss(sb_j)$.

The correct assignment of senses would be sense i for word a and sense j for word b where $\text{Lesk}(su_i, sv_j)$ is maximum. But it is a drawback that this method requires calculation of every possible pair of senses. The context in this research application is often not simply between 2 words but more, so the process then gets exponentially large.

Due to that, a Simplified Lesk Algorithm is available and is what was implemented for this research. This simplified version does not take into consideration the whole context at one time, but instead it focuses on a target word specifically. Take a certain seed word s, which is found in a context of n different words $context(s) = c_1, c_2, ..c_n$. The word s has m multiple senses $Se(s) = s_1, s_2, ..s_m$. Therefore the score is done between the glosses of each sense of the word $s_i \in Se(s)$ and the context. The assigned sense of s is the i^th that maximizes the Lesk value [33].

$$\text{Lesk}(s) = ||context(s) \cap gloss(s_i)||,$$

with || denoting the number of overlapping elements of $context(s)$ and $gloss(s_i)$.

In this application, the seed words are disambiguated within the context of all the seed words with the value of 1. Once more, taking as an example, a lecture video in the field of algebra once more, having the seeds as shown in Table 3. Once again the title of this lecture is ”Linear Equations in Mathematics: Solving the Inequality”. As mentioned previously, the word ”equation” has 3 different senses, which are:

1. equation – (a mathematical statement that two expressions are equal)
2. equality, equivalence, equation, par – (a state of being essentially equal or equivalent; equally balanced; "on a par with the best")
3. equation, equating – (the act of regarding as equal)

Calculating the number of overlapping words between the context (the context is all the seeds with the visualness value of 1). The context will be:

$$context = algebra, equation, mathematics, inequality$$

Compared to the 3 different glosses or definitions of the word ”equation”, $equation_1$ overlaps with the context on the word ”mathematics”. This results into the Lesk score of $\text{Lesk}(equation_1) = 1$. The other two senses $equation_2$ and $equation_3$ each have the Lesk score of 0. Therefore, the maximum is sense 1, and sense 1 is assigned for the word ”equation”.

Refinement of W

Identical to the process in Section 3.5.2, the final result of the WSD process is the final list of seed words $S^{final} = s_1, s_2, ...s_n$ and corresponding senses $Se^{final} = Se(s_1), Se(s_2)...Se(s_n)$. The scoring method then proceeds to use these seed words S and the senses Se in the visualness calculation, obtaining the list $Vis(w_i)$. After sorting and rank cutoff, the system obtains the final refined keyword list K. The process of VLD can also be illustrated by Figure 17, with the difference from VDC only apparent in the ”disambiguator” block. For the VLD method, this block uses the Simplified Lesk Algorithm.

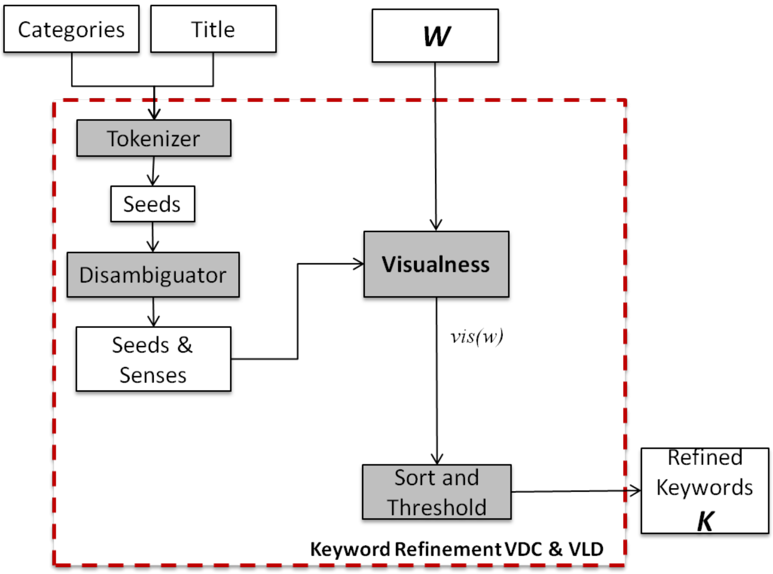


Figure 17: VDC and VLC keyword refinement process.

3.5.4 Lesk Visualness and Disambiguation with Frequency of Occurrence (LVD-F)

This last scoring method builds onto the previously developed VLD method. Of the two different WSD methods, the Simplified Lesk algorithm is maintained to perform the word sense disambiguation due to the good performance and reliable results according to various resources such as [33] and [34].

The visualness calculation is the same as the previously explained methods, but with a significant difference. Since the Lesk concept was very successful and logical in its implementation, it is also applicable to use the Lesk algorithm concept of gloss overlaps in similarity measurements as well.

Adapted Lesk Similarity

The Adapted Lesk metric uses the gloss of a word as a representation of its semantic meaning and concept. This measure measures relatedness of two words by scoring the overlap of their glosses. Not only does this metric use the direct gloss of the word, but it also takes the gloss of related words. These related words are found by the hypernym, hyponym, holonym, meronym, or troponym relation. The overlap of these extended glosses contribute to a certain score.

Additionally, the Adapted Lesk metric does not only consider overlapping words as explained in Section 3.5.3, but also overlapping sequences of words. Each overlap found between two glosses contributes a score equal to the square of the number of words in the overlap. This method was used in the hopes that the results would improve, although as a trade off, the computation time needed for this method also increased. As pairwise comparisons must be performed between glosses, the time needed is much higher [35].

Ultimately the word list W is scored using the visualness formula but using the adapted lesk measure as the similarity function. The final result is the visualness values $\mathrm{Vis}(w_i)$.

Frequency of Occurence

As an addition to the semantic visualness score, this method also takes into consideration the statistical value from the word's occurrence. The frequency of each word is calculated for the video segment relative to the total number of words in the segment. This returns the list of relative frequency values $\mathrm{Freq}(w)$ for all words w in W.

Combined LVD-F Score

The final score of this LVD-F method is a combination of both the visualness values and the frequency values of each word. The idea is that a word with a certain semantic value but occurs often with get favored over words with low frequency of occurrence. The final score is calculated as the following:

$$\mathrm{LVD} - \mathrm{F}(w) = (\alpha)\mathrm{Vis}(w) + (1 - \alpha)\mathrm{Freq}(w)$$

The optimal α value for the forumla is determined empirically over the dataset in use.

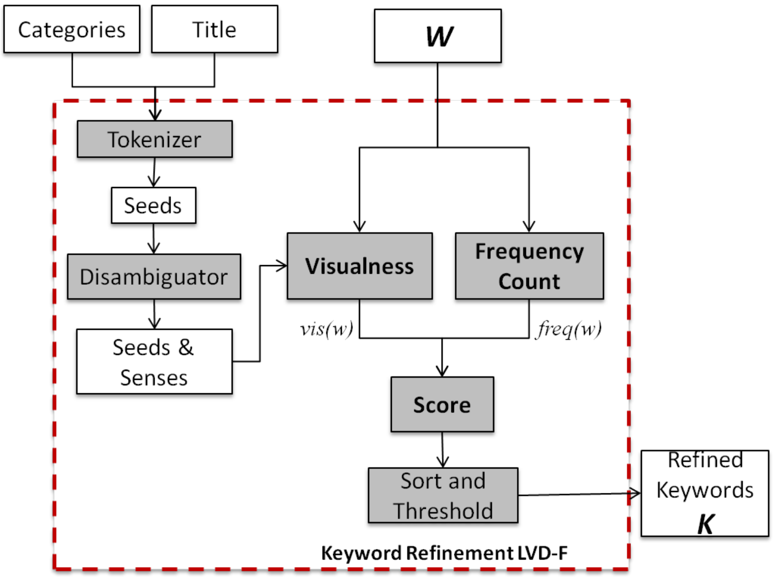


Figure 18: LVD-F keyword refinement process.

Refinement of W

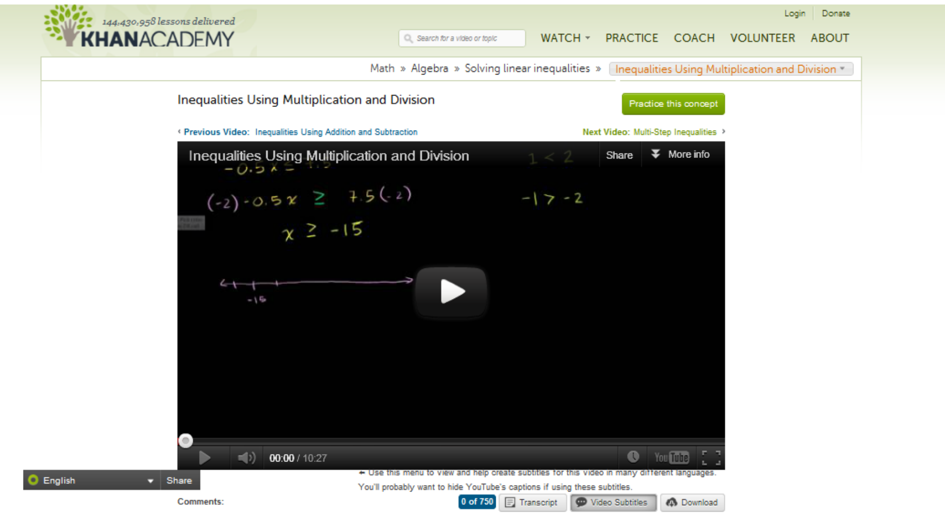
Upon obtaining the list of the visualness values of all keywords in W according to the seed set S and seed senses Se, namely $Vis(w_i)$, the list is processed as follows. Sorting and ranking is performed in every word w_i of W according to $Vis(w_i)$, then the list is cut off at a certaiin rank to determine the top n words, forming the refined keyword list K. The process is shown in Figure 18.

4 Experiments

The automatic semantic annotation system explained previously in Chapter 3 was developed and implemented on some videos from the Khan Academy website. The implementation of the system was done using both Matlab and Java SE 7 platforms. The shot boundary detection was implemented in Matlab, whereas the remaining processes starting from transcript segmentation to the final keyword scoring is done in Java.

4.1 Lecture Video Dataset

As explained previously in Section 3.1, the lecture videos taken from the Khan Academy website, with widespread topics of common knowledge. The Khan Academy is a non-profit organization which has a collection of more than 3000 videos containing educational lectures in topics ranging from mathematics to art history [1]. A screen shot of a lecture in this database is shown in Figure 19.


144,430,958 lessons delivered
KHANACADEMY
Login
Donate
Search for a video or topic
WATCH
PRACTICE
COACH
VOLUNTEER
ABOUT
Math » Algebra » Solving linear inequalities » Inequalities Using Multiplication and Division
Inequalities Using Multiplication and Division
Practice this concept
Previous Video: Inequalities Using Addition and Subtraction
Next Video: Multi-Step Inequalities
Inequalities Using Multiplication and Division
Share
More info
(-2)-0.5x ≥ 7.5(-2)
x ≥ -15
-1 > -2
-15
00:00 / 10:27
YouTube
English
Share
Comments:
0 of 750
Transcript
Video Subtitles
Download

← Use this menu to view and help create subtitles for this video in many different languages.
You'll probably want to hide YouTube's captions if using these subtitles.

Figure 19: A screenshot of a lecture video from the Khan Academy [1].

The decision to opt for an on-line repository of lecture videos was mainly due to the lack of a standardized database for lecture videos. The Khan Academy lectures also had the advantage of containing the content of the lectures directly without the lecturer in the visual field, as well as the availability of transcripts and titles of the lectures which provide the additional documents needed for cross document annotation. The last reason is that the Khan Academy website also provides a hierarchical category structure in which the lecture videos are organized. This categorization is needed in the later process of the video.

The experiments were carried out using 5 different lecture videos covering a variety of topics. The lectures were selected randomly, but deliberately basic so that any random person would be able to follow them. These 5 lectures are the following:

1. Lecture 1: Statistics
2. Lecture 2: Physics
3. Lecture 3: Linear Algebra
4. Lecture 4: Astronomy
5. Lecture 5: Biology

Each of these full-length lectures were put through the temporal segmentation sub-system, which cut the videos into segments if a cut was detected. The segmentation resulted into 11 video segments as shown in Table 4. The temporal segmentation subsystem obtains the cut boundaries using the method explained in Section3.2, if detected. Obviously if cuts are not present the full length video is not segmented and assumed to be a single segment, such as what happens with Lecture 2. The resulting segmentation for this particular lecture only results in 1 segment, Segment 3, which is the duration of the full-length video.

Video	No Of Segments	Part	Segment Label	Duration
Statistics	2	Statistics part 1	Segment 1	0:00 - 8:45
		Statistics part 2	Segment 2	8:45 - 12:30
Physics	1	Physics part 1	Segment 3	0:00 - 8:33
Linear Algebra	3	Linear Algebra part 1	Segment 4	0:00 - 5:22
		Linear Algebra part 2	Segment 5	5:22 - 9:50
		Linear Algebra part 3	Segment 6	9:50 - 11:41
Astronomy	2	Astronomy part 1	Segment 7	0:00 - 3:54
		Astronomy part 1	Segment 8	3:54 - 10:52
Biology	3	Biology part 1	Segment 9	0:00 - 8:13
		Biology part 2	Segment 10	8:13 - 13:38
		Biology part 3	Segment 11	13:38 - 18:20

Table 4: Resulting segments of temporal video segmentation.

4.2 Experiment Design

Each of these video segments along with their corresponding segment transcripts are processed individually into the semantic keyword selection sub-system, resulting into a list of keywords sorted by their descending scores. The final keywords used to annotate each segment is determined by a rank cutoff system, so that only the top ranking words are taken to annotate that segment.

The semantic keyword selection is done using all 4 of the scoring methods described in Section 3.5, 3 of which are the novel semantic scoring methods developed in this research. This scoring is done with a fully automatic process by the computer, therefore this score will be referred to as the objective score of keywords.

In order to measure the performance of these objective scoring methods, a ground truth must be developed. The aim of the automatic system is to determine the best suitable words to annotate the video, in the pursuit of enabling an effective search and retrieval mechanism on the videos. Therefore it seems logical to determine which words do users actually use while posing queries for videos.

4.2.1 On-Line Survey

A subjective survey was then carried out, by publishing on-line questionnaires. This survey was distributed to many people, in the aim of determining the words that are really used by e-learning system users. The survey was published with Google Docs on-line forms, asking participants to view a video segment, and selecting words that they would use if they were searching for that segment.

The survey was limited to people who have had experience using an e-learning system. Responses from unexperienced users were discarded. The demographics of the participants are shown as follows. Figure 20 shows distribution of participants by country of residence, Figure 21 by age, Figure 22 by gender, and 23 by highest level of education.

By participating in the survey, participants were asked to watch a segment of a lecture video. For the sake of simplicity, the survey was carried out on the first 10 segments only. 10 surveys were carried out, 1 for each of the first 10 segments (Segment 1 - Segment 10) in Table 4.

For these 10 surveys, 15 participants were obtained for each survey. In order to make the results comparable to the objective scores, the list of potential words *W* from the keyword extraction process are given to the respondents in random order. The participants are asked to limit their word choices to the words in the provided list only.

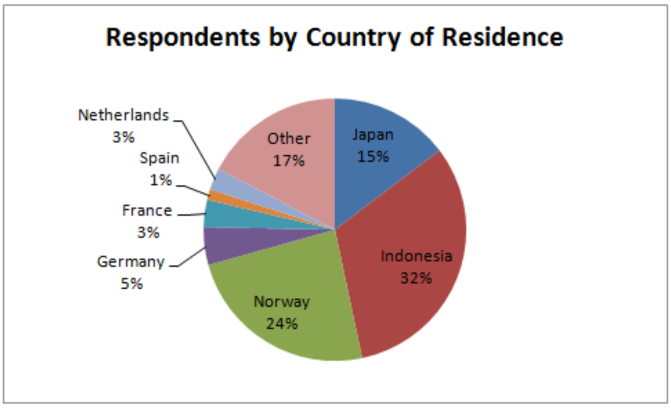


Figure 20: Participant distribution based on country of residence.

The participants are then asked to imagine if they were users of an e-learning system with these videos as MLOs. They are then asked to list the top 5 words they would use to retrieve that video segment in question. Therefore, for each participants, there will be a list of 5 top words they would use. In order to obtain a single list of top words, the first step needed is to recap the frequencies of the word selection according to the position. The recap is done to obtain a final count of the chosen words, and how many times it comes up in the responses as word 1, word 2, and so on. 'Position 1' is the number of times the word comes up as the first choice in the survey,

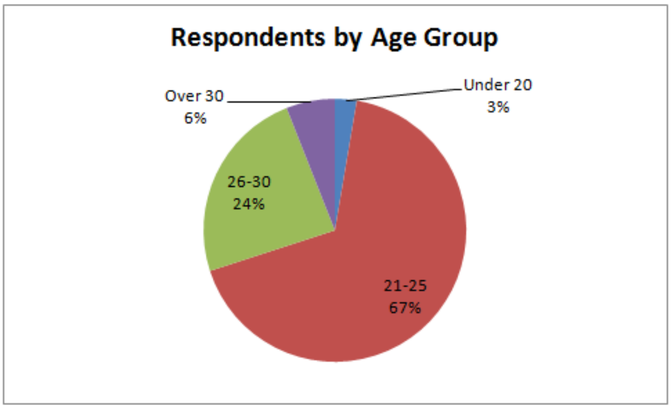


Figure 21: Participant distribution based on age.

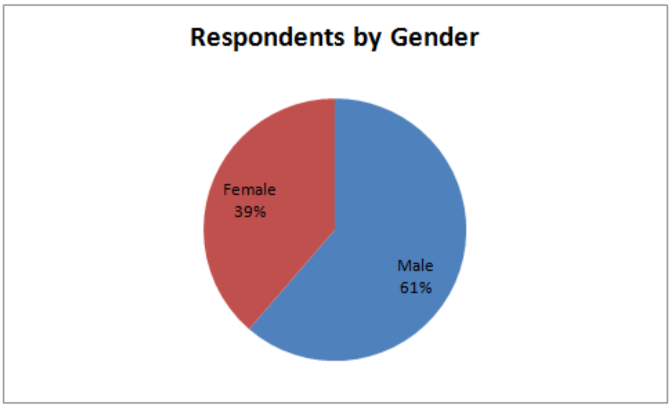


Figure 22: Participant distribution based on gender

’Position 2’ is the number of times the word comes up as a second choice, and so forth. The result of the recap is shown in Table 5.

4.2.2 Borda Count

From the frequency recap, it is necessary to be able to get a score for each word. A total frequency can be calculated, but it is not desired, since the result of the surveys rank the word of choice starting from Position 1 to Position 5. In this step, the Borda Count is employed. The Borda Count is an election method used to determine a winner from a voting where voters rank the candidates in order of preference. The same is valid in this survey choice, in which participants rank the words of choice in order of preference [36].

The Borda Count scoring method is used in different wasy for different applications, but in this case the Borda count is used in the simplest way. For the choice of ranking n choices, each $i - th$ choice is scored by $(n - i)$. Therefore words occurring in Position 1 will get scored by $(n - 1)$, the highest score, and the $n - th$ choice will be scored by $(n - n)$ or 0.

In the particular case of this survey, the responses ranked the words in 5 ranks $(n = 5)$. Each

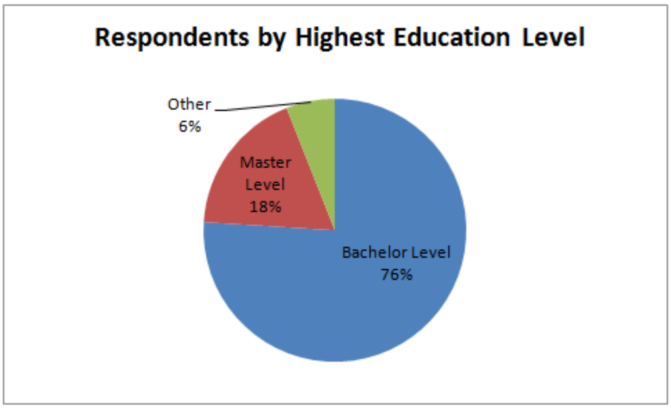


Figure 23: Participant distribution based on highest level of education

word in position 1 will be scored by 4, position 2 by 3, and so on, respectively. The resulting scores for each word is computed as follows:

$$\mathrm{Borda}(w) = \sum_{i=1}^{n} ((n-i) * \mathrm{freq}_i(w)),$$

where $\mathrm{Borda}(w)$ of a given word w is calculated by a total sum of the weights of the frequencies $\mathrm{freq}_i(w)$. $\mathrm{freq}_i(w)$ is the frequency of word w chosen at Position i. n is the total number of possible positions, in our case $n = 5$.

The resulting scores are then sorted to obtain the top words, shown in Table 6. This scoring method is done from the subjective responses of participants, therefore it will be referred to as the subjective scoring. These subjective scoring results will later be used to compare the results of the objective methods and determine their performance.

Segment 1 : Statistics					
Word	**Position 1**	**Position 2**	**Position 3**	**Position 4**	**Position 5**
Statistics	12	0	0	0	0
Average	0	2	6	2	0
Mean	2	2	2	2	2
Arithmetic	0	4	2	0	0
Tendency	0	2	2	0	3
Median	0	1	1	2	3
Data	0	1	0	1	0
Mode	1	0	0	0	2
Example	0	1	0	1	0
Numbers	0	1	0	0	0
Definition	0	0	0	3	0
Central	0	0	0	3	0
Computation	0	1	0	0	0
People	0	0	1	0	0
Inference	0	0	1	0	0
Tool	0	0	0	1	0
Measure	0	0	0	0	1
Sample	0	0	0	0	1
Video	0	0	0	0	1
Formula	0	0	0	0	2

Table 5: Recapitulation of the responses from on-line survey for a lecture video segment about statistics (Segment 1).

Segment 1 : Statistics			
Word	**Borda Count**	**Word**	**Borda Count**
Statistics	48	Definition	3
Average	20	Central	3
Mean	20	Computation	3
Arithmetic	16	People	2
Tendency	10	Inference	2
Median	7	Tool	1
Data	4	measure	0
Mode	4	Sample	0
Example	4	Video	0
Numbers	3	Formula	0

Table 6: Borda count of the responses from on-line survey for Segment 1.

5 Results and Analysis

In this chapter, the comparison between the subjective and objective scores will be shown. The aim is for the objective score to be able to predict and replicate the results of the subjective score. It is desired that the objective scores employed in the system are able to score the words correctly in order to obtain the same final list of refined keywords that was obtained using subjective scores.

The performance of objective methods were evaluated based on two criteria. First being how well the objective methods score the top subjective words. This is done by examining the obtained objective scores. The second evaluation criteria was to check if the top words scored by the objective methods are accurate. A precision and recall calculation was used for this purpose.

5.1 Performance of Objective Scores

The first evaluation is to investigate how well the objective methods score the subjective words. This is done by taking the subjective words as a ground truth. The words are sorted based on this subjective scores then the top n words are selected. For this evaluation the top 10 words are taken. It is assumed that these are the refined keywords in K that would be the optimal list to use for video annotation. The objective scores of these 10 words are then compared. The goal is to have the objective methods give a high score to these words. For the top 10 subjective words for each of the video segments, all 4 objective scores are shown. It is important to remember that the objective scores have a range from 0 to 1, 1 indicating high importance and 0 indicating the opposite.

The following results that are shown here are of 5 different video segments, one each for every subject covered as mentioned in Section 4.1.The results are shown in Table 7 for statistics, Table 8 for physics, Table 9 for linear algebra, Table 10 for astronomy, and Table 11 for biology.

The aim of this comparison is to see how well the objective methods score the top 10 subjective words. The hope is that these methods are able to give a high score to this ground truth. It is apparent that the objective scores give a relatively good score towards the top subjective words.

In Table 7, VDC gives a high score to a few of the words, but the rest are scored rather poorly. The word ”median” even scores 0. This is possibly because of the disambiguation by category method used which assigned the wrong sense to ”median”. Therefore when using the Lesk disambiguation method, VLD assigns a high score to ”median”. In the general case, VLD scores the words slightly better. We can observe that VLD gives high scores to a larger number of the words. Other than that, generally all the other scores are still in the higher range.

The LVD-F scores behave a bit differently. The scores are relatively good, with most of the scores having a value above 0.6. Nevertheless, many of the values are lower than the VLD scores. This probably is the result of the additional frequency score $\mathtt{freq}(w)$ being added to the visualness score vis in the LVD-F method. This frequency score influences the overall resulting score of LVD-F.

Segment 1 : Statistics				
Subjective Result	**Objective Scores**			
Top 10 Words	**TF-IDF**	**VDC**	**VLD**	**LVD-F**
Statistics	0.82	1.00	1.00	0.85
Average	0.73	1.00	1.00	1.00
Mean	0.00	0.55	1.00	0.64
Arithmetic	0.00	0.26	0.95	0.29
Tendency	0.61	0.07	0.62	0.20
Median	0.64	0.00	1.00	0.18
Data	1.00	0.06	0.59	0.23
Mode	0.84	0.06	0.05	0.29
Example	0.00	0.11	0.91	0.63
Numbers	0.77	0.06	0.58	0.66

Table 7: Objective scores of proposed methods for Segment 1.

Segment 3 : Physics				
Subjective Result	**Objective Scores**			
Top 10 Words	**TF-IDF**	**VDC**	**VLD**	**LVD-F**
Vector	1.00	1.00	1.00	0.97
Scalar	1.00	1.00	1.00	1.00
Distance	1.00	0.39	0.76	0.34
Physics	1.00	1.00	1.00	0.76
Magnitude	1.00	0.37	0.77	0.36
Displacement	1.00	0.44	0.77	0.3
Direction	1.00	0.41	0.77	0.58
Definition	1.00	0.37	0.80	0.23
Move	1.00	0.49	0.87	0.28
Speed	1.00	0.42	0.77	0.26

Table 8: Objective scores of proposed methods for a lecture video segment about physics (Segment 3).

Alternately the other results in Table 8 also show the same trend. In this video, the same trend can be observed. VDC once more scores some words high, but the others relatively low. VLD performs well as it scores most of the words consistently high. LVD-F scores are high for some but low for the others.

The TF-IDF scores are high for all instances with a score of 1, but this is due to the fact that the physics lecture was not temporally segmented. A cut was not detected in this lecture, hence the segment was equal to the whole video. Therefore the frequency of occurrence of any word w in the segment $f_s(w)$ is equal to the frequency in the whole video $f_v(w)$, always resulting in a TF-IDF score of 1. This makes the TF-IDF score irrelevant in lectures that can not be temporally segmented.

In the remaining lecture videos in Tables 9, 10, and 11, the same phenomenon can be observed . VLD still consistenly gives the highest scores. Once again, the LVD-F method does not give high scores as VLD, due to the influence of the freuqency score. It is relevant though to

Segment 4 : Linear Algebra				
Subjective Result	**Objective Scores**			
Top 10 Words	**TF-IDF**	**VDC**	**VLD**	**LVD-F**
Matrix	0.57	1.00	1.00	1.00
Row	0.61	0.66	0.96	0.96
Mathematics	1.00	1.00	1.00	0.99
Linear algebra	0.00	1.00	1.00	0.55
Element	0.20	0.54	0.815	0.22
Number	0.14	1.00	0.99	0.13
Column	0.54	0.92	0.93	0.33
Introduction	0.00	1.00	1.00	0.55
Algebra	0.75	1.00	1.00	0.55
Representation	1.00	0.53	0.74	0.43

Table 9: Objective scores of proposed methods for a lecture video segment about linear algebra (Segment 4).

Segment 7 : Astronomy				
Subjective Result	**Objective Scores**			
Top 10 Words	**TF-IDF**	**VDC**	**VLD**	**LVD-F**
Big bang	0.54	1.00	1.00	1.00
Universe	0.47	0.35	0.70	0.74
Astronomy	0.00	1.00	1.00	0.70
Theory	0.40	0.35	0.72	0.27
Explosion	1.00	0.46	1.00	0.65
Cosmology	0.00	1.00	1.00	0.7
Space	0.58	0.33	0.76	0.68
Edge	0.54	0.35	0.76	0.52
Problem	0.00	0.33	0.76	0.18
Bang	0.54	1.00	1.00	1.00

Table 10: Objective scores of proposed methods for a lecture video segment about astronomy (Segment 7).

point out here that even a low scored word such as ”column” with 0.33 LVD-F score in table 9, is actually ranked high when the LVD-F scores are sorted. This will be retrieved as a top word. This will be shown in more detail with the calculation of precision and recall in Section 5.2.

5.2 Precision and Recall

The next evaluation will be calculating precision and recall of each objective score and each lecture. The subjective refined keyword list K remains the same, taking the top 10 words from the subjective scores. But in this comparison, the top 15 words of each objective score are also taken as the objective refined keyword list. From these two lists, the precision and recall of the objective methods are calculated.

Reviewing the definitions, precision is the fraction of retrieved documents that are relevant to the search, and recall is the fraction of relevant documents that are successfully retrieved [5].

Segment 10 : Biology				
Subjective Result	**Objective Scores**			
Top 10 Words	**TF-IDF**	**VDC**	**VLD**	**LVD-F**
Bacteria	0.27	1.00	1.00	1.00
Cell	0.25	0.47	0.92	0.19
Mutation	1.00	0.44	0.44	0.29
Conjugation	1.00	0.44	0.44	0.14
Reproduction	1.00	0.42	0.43	0.46
Nucleus	0.37	0.42	0.42	0.22
Prokaryote	0.00	0.00	1.00	0.3
Fission	1.00	0.42	0.42	0.23
Biology	0.00	1.00	1.00	0.56
DNA	0.61	0.00	0.00	0.59

Table 11: Objective scores of proposed methods for a lecture video segment about biology (Segment 10).

Precision and recall values are calculated as follows:

$$Precision = \frac{retrieved \cap relevant}{retrieved}$$

$$Recall = \frac{retrieved \cap relevant}{relevant}$$

A recapitulation of the words retrieved by each method is shown in Table 12 for the statistics lecture, Table 13 for the physics lecture, Table 14 for the linear algebra lecture, Table 15 for the astronomy lecture, and Table 16 for the biology lecture. The correctly retrieved words are underlined, and the precision and recall values are calculated accordingly. Once again the detailed results are shown for one video segment for each subject.

It is observable that the overall result of recall is better than the precision for every type of lecture. Precision and recall values are dependent on the lecture video in question, therefore they will fluctuate on different lecture videos. It is also important to know the are of expertise of the respondents, because it will influence the subjective words. Overall the LVD-F method constantly achieves the highest precision and recall values. The recapitulation of the precision for every segment is shown in Table 17, and the recall is shown in Table 18.

In both videos shown here, the precision and recall of LVD-F is always better than the other two methods. This shows that the top ranked words by LVD-F is a rather good estimation of the words that a subjective user would choose. Revisiting the results in Tables 7 - 11, it is mentioned that the LVD-F scores are not very high in some cases. Even so, after the ranking and cutoff process, these words are still retrieved in the top 15. Therefore, LVD-F is able to achieve the best precision and recall score. The precision is still relatively low for most of the lectures, but the recall is able to reach high values up to 0.9, for the Statistics Lecture. Overall the average of all segments also shows that the LVD-F method obtains the highest average precision and recall. LVD-F obtains the highest average of 0.47 for precision and 0.71 recall.

Segment 1 : Statistics				
Top 10	**Objective Words**			
Subjective Words	**TF-IDF**	**VDC**	**VLD**	**LVD-F**
Statistics	abstract	statistics	statistics	average
Average	add	average	average	statistics
Mean	ambiguity	mean	mean	numbers
Arithmetic	answer	data	median	median
Tendency	basic	arithmetic	arithmetic	set
Median	bunch	time	harmonic	number
Data	clarity	representative	mode	datum
Mode	classified	example	datum	mode
Example	close	sample	representative	arithmetic
Numbers	compare	case	general	data
	computation	particular	time	tendency
	context	thinking	add	central
	couple	computation	example	plus
	cover	sense	particular	middle
	data	inference	sample	median
Precision	0.06	0.40	0.46	0.60
Recall	0.10	0.60	0.70	0.90

Table 12: Precision and recall of the top words for Segment 1.

Segment 3 : Physics				
Top 10	**Objective Words**			
Subjective Words	**TF-IDF**	**VDC**	**VLD**	**LVD-F**
vector	amount	physics	physics	scalar
scalar	basic	introduction	introduction	vector
distance	bit	vector	vector	meter
physics	block	scalar	scalar	question
magnitude	brick	start	quantity	physics
displacement	bunch	quantity	amount	introduction
direction	business	amount	idea	right
definition	calculation	move	kind	direction
move	call	movement	like	second
speed	change	travel	make	quantity
	clear	right	right	know
	color	change	color	distance
	deal	kind	version	magnitude
	dealing	idea	way	change
	definition	pick	well	talking
Precision	0.06	0.27	0.20	0.40
Recall	0.10	0.40	0.30	0.60

Table 13: Precision and recall of the top words for Segment 3.

Segment 4 : Linear Algebra				
Top 10	**Objective Words**			
Subjective Words	**TF-IDF**	**VDC**	**VLD**	**LVD-F**
matrix	back	algebra	algebra	matrix
row	back burner	introduction	introduction	row
mathematics	bit	matrix	matrix	algebra
linear algebra	bold face	linear algebra	linear algebra	introduction
element	bunch	mathematics	book	linear algebra
number	burner	table	number	column
column	cabinet	column	table	table
introduction	capital letter	row	row	determinant
algebra	class	file	writing	equation
representation	colored	pick	column	mean
	comma	means	put	notation
	computer	going	file	number
	computer graphics	find	well	back
	concept	class	letter	call
	coordinate	draw	determinant	face
Precision	0.00	0.46	0.46	0.46
Recall	0.00	0.70	0.70	0.70

Table 14: Precision and recall of the top words for Segment 4.

Segment 7 : Astronomy				
Top 10	**Objective Words**			
Subjective Words	**TF-IDF**	**VDC**	**VLD**	**LVD-F**
big bang	answer	cosmology	cosmology	bang
universe	case	astronomy	astronomy	big bang
astronomy	change	bang	bang	universe
theory	couple	introduction	introduction	cosmology
explosion	earth	big bang	big bang	astronomy
cosmology	explosion	start	explosion	introduction
space	idea	change	bite	explosion
edge	infinite	explosion	start	space
problem	latitude	going	going	question
bang	longitude	example	line	area
	mass	draw	talk	edge
	matter	case	bit	right
	nice	try	draw	sphere
	product	best	change	now
	side	bit	case	planet
Precision	0.06	0.33	0.30	0.53
Recall	0.10	0.50	0.50	0.80

Table 15: Precision and recall of the top words for Segment 7.

Segment 10 : Biology				
Top 10	Objective Words			
Subjective Words	TF-IDF	VDC	VLD	LVD-F
bacteria	beginning	biology	biology	bacteria
cell	binary	bacteria	bacteria	dna
mutation	bit	introduction	introduction	question
conjugation	call	beginning	prokaryote	biology
reproduction	cell wall	move	well	introduction
nucleus	circular	movement	simple	reproduction
prokaryote	cleavage	motion	piece	pilus
fission	conjugation	transfer	cell	motion
biology	connection	mix	primitive	kind
dna	deal	combination	tell	get
	deep	mixing	stepchild	plasmid
	end	means	begin	antibiotic
	evolution	draw	zygote	mitosis
	extra	spread	gamete	prokaryote
	fission	mechanism	transfer	mutation
Precision	0.13	0.13	0.26	0.40
Recall	0.20	0.20	0.40	0.60

Table 16: Precision and recall of the top words for Segment 10.

	Segment			Precision			
Lecture	Part	Segment No	Details	TF-IDF	VDC	VLD	LVD-F
Statistics	Part 1	Segment 1	Table 7	0.66	0.40	0.46	0.60
	Part 2	Segment 2	Not shown	0.00	0.26	0.40	0.53
Physics	Part 1	Segment 3	Table 8	0.06	0.27	0.20	0.40
Linear Algebra	Part 1	Segment 4	Table 9	0.00	0.46	0.46	0.46
	Part 2	Segment 5	Not shown	0.13	0.33	0.40	0.53
	Part 3	Segment 6	Not shown	0.13	0.33	0.46	0.53
Astronomy	Part 1	Segment 7	Table 10	0.06	0.33	0.30	0.53
	Part 2	Segment 8	Not shown	0.00	0.13	0.20	0.43
Biology	Part 1	Segment 9	Not shown	0.00	0.20	0.33	0.33
	Part 2	Segment 10	Table 11	0.13	0.13	0.26	0.40
Average Precision				0.12	0.28	0.35	0.47

Table 17: Precision of each method in all lecture video segments.

	Segment			Recall			
Lecture	**Part**	**Segment No**	**Details**	**TF-IDF**	**VDC**	**VLD**	**LVD-F**
Statistics	Part 1	Segment 1	Table 7	0.10	0.60	0.70	0.90
	Part 2	Segment 2	Not shown	0.10	0.50	0.80	0.90
Physics	Part 1	Segment 3	Table 8	0.10	0.40	0.30	0.60
Linear Algebra	Part 1	Segment 4	Table 9	0.00	0.70	0.70	0.70
	Part 2	Segment 5	Not shown	0.20	0.50	0.60	0.80
	Part 3	Segment 6	Not shown	0.20	0.50	0.70	0.80
Astronomy	Part 1	Segment 7	Table 10	0.10	0.50	0.50	0.80
	Part 2	Segment 8	Not shown	0.00	0.20	0.30	0.50
Biology	Part 1	Segment 9	Not shown	0.00	0.30	0.50	0.50
	Part 2	Segment 10	Table 11	0.20	0.20	0.40	0.60
	Average Recall			0.10	0.44	0.55	0.71

Table 18: Recall of each method in all lecture video segments.

6 Conclusion and Future Work

After the completion of the research presented in this report, there are some conclusions to be drawn. The work done in this thesis has provided some results and insight on semantic annotation done automatically. The system presented in this report must still be tested to different videos and databases to determine its performance. Ultimately the goal is to create an automatic method to do annotation with semantic analysis for lecture videos, and the system developed in this work shows the potential that can be done.

Additionally, many improvements can also still be done on this subject. Therefore this chapter will also list some potential topics for future work in this field. A lot of additional features and methods can be employed to improve the system in this report. The prospect of continuous research in this area seems to be promising in the future.

6.1 Conclusions

The conclusions that were drawn after the completion of this work and the development of this system are explained in ths section.

1. Three (3) objective scoring methods were developed in this research. These semantic scoring methods are able to score the initial potential keyword list well to obtain a top list of words. These top words correspond to the words used by a human user. The calculation of visualness is a reliable method of calculation, but there is the need to determine the best suitable seed words.
2. VLD method assigns the highest scores to the top ranking words obtained in the subjective experiment. As shown in Section 5.1, the aim of these objective methods are to obtain high scores for subjective top words. VLD is the most consistent of the developed methods in giving a high score to the subjective methods.
3. Among the 3 developed methods, LVD-F obtains the highest precision and recall. This is shown in Section 5.2 where it is clear that LVD-F is consistently able to retrieve the subjective words with the highest precision and recall.
4. Semantic annotation is necessary in this application. It is shown that the simple statistical methods on their own (such as TF-IDF) are not able to determine which keywords are most meaningful from a potential list. Therefore it is necessary to employ additional semantic analysis to enable a proper keyword refinement process.
5. WordNet is a very robust and reliable semantic network to use. The development of WordNet is an ongoing research itself, and the development is hoped to continue to grow. WordNet has also been developed in different languages, so the implementation can be done in different languages as well. Many developed platforms and programming packages are available freely for use, which gives a large advantage to any further research in this field. WordNet is also

the most popular semantic network in a large number of research.

6. Cross document annotation is proven to be a viable solution for automatic video annotation. Visual objects and concepts relatively difficult to detect from the lecture video. Considering that in real e-learning systems, MLOs are indeed very commonly accompanied by other related documents, cross document annotation is a potential approach to be pursued.
7. For the particular dataset of videos used in this research, the temporal segmentation performs very well, detecting the particular cut boundaries nicely. The method should also work well on any type of lecture video that uses the same type of content area.

6.2 Contributions

The work completed in this thesis can be used for further work in various areas. The contributions this thesis has to offer can be described in the following points.

1. This research has successfully implemented a fully automatic calculation of visualness for lecture videos. Previous research used wide-domain videos which in their nature are quite different from lecture videos. The initial implementations of visualness also involve manual input from users, as for the implementation in this research is automatic.
2. In this report, a novel framework for automatic annotation of lecture videos is used. This framework processes lecture videos until they are formed into efficient media learning objects.
3. There are 3 different novel objective methods developed in this research based on visualness similarity and word sense disambiguation. Among these 3 methods, the proposed LVD-F method consistently outperforms the other 3 objective scoring methods.

6.3 Future Work

The work done in this thesis can still be expanded in so many ways. The semantic processing done here was not done on the linguistic level, as many WordNet function were used as a black box. Additionally, the system itself can use various improvements, as well as different approaches to perfect the automatic system.

The ultimate goal in this field is to be able to automatically create a media learning object which contains of the video, segmented temporally, and appropriate keywords to annotate along this temporal domain. It is ideal to be able to create this MLO in a fully automatic fashion, and this system is a start towards that direction. Further works that may be able to improve this solution are:

1. Objective Scoring Methods. There is still a lot of potential in further development of scoring methods, utilizing semantic and statistical properties. The proposed scoring methods described in this report can also still be tuned in order to obtain the best scores possible.
2. Cross Document Annotation The source of additional text documents does not have to be limited to transcripts. In a real e-learning system, alternate resources should be available together with the video, and can be used for cross document annotation. The system can be expanded to include these different documents for a broader and richer keyword source.

3. Audio Processing If the scope of this research is expanded, an additional source that can be used is the audio of the video. Speech analysis can be performed to determine spoken words. This can be an additional source for keywords.

4. Optical Character Recognition (OCR)
   OCR attempts to recognize the text on the content area. This report mentions OCR very briefly, but does not go into detail. OCR is a very hard task and its research is constantly ongoing. OCR becomes even more difficult considering that the content area in this database is handwriting. OCR may prove to be more robust on content areas where the text is type written (such as slides).
   If a successful OCR can be performed on the visual field of the video, extracting potential text, this additional text can be use in the semantic processing. In the system proposed in this thesis, this additional text can be used as the seed concepts in the semantic scoring methods.

5. System Integration
   The system implemented in this work is modular, with each separate sub-system implemented individually. This enables future work to use each sub-system independently for different applications. However, it is also a potential continuation of this system to incorporate each sub-system into a single working application, completing the automatic process entirely.

Bibliography

[1] The Khan Academy. Khan Academy. `http://www.khanacademy.com`.

[2] Blanken, H. M., Vries, A. P. d., Blok, H. E., & Feng, L. 2007. *Multimedia Retrieval (Data-Centric Systems and Applications)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA.

[3] El Saddik, A., Ghavam, A., Fischer, S., & Steinmetz, R. 2000. Metadata for smart multimedia learning objects. In *Proceedings of the Australasian conference on Computing education*, ACSE '00, 87–94, New York, NY, USA. ACM.

[4] Filho, C. A. F. P. & Santos, C. A. S. 2010. A new approach for video indexing and retrieval based on visual features. *Journal of Information and Data Management (JIDM)*, 1(2), 293–308.

[5] Manning, C. D., Raghavan, P., & Schutze, H. 2008. *Introduction to Information Retrieval*. Cambridge University Press, Cambridge, UK.

[6] Baeza-Yates, R. & Ribeiro-Neto, B. 1999. *Modern Information Retrieval*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA.

[7] Shih, T. K., ed. 2002. *Distributed multimedia databases: techniques & applications*. IGI Publishing, Hershey, PA, USA.

[8] Hrastinski, S. 2008. Asynchronus and synchronous e-learning. *Educause Quarterly*, 4, 51–56.

[9] Chandra, S. 2011. Experiences in personal lecture video capture. *IEEE Transactions on Learning Technologies*, 4, 261–274.

[10] Imran, A. S. 2009. Interactive media learning object in distance and blended education. In *Proceedings of the 17th ACM international conference on Multimedia*, MM '09, 1139–1140, New York, NY, USA. ACM.

[11] Lin, M., Chau, M., Cao, J., & Nunamaker Jr., J. 2005. Automated video segmentation for lecture videos: A linguistics-based approach. *International Journal of Technology and Human Interaction (IJTHI)*, 1(2), 27–45.

[12] Halvorsen, M. R. Content based lecture video indexing. Master's thesis, Høgskolen i Gjøvik, 2007.

[13] Yang, H., Siebert, M., Luhne, P., Sack, H., & Meinel, C. 2011. Automatic lecture video indexing using video OCR technology. In *Proceedings of the 2011 IEEE International Symposium on Multimedia*, ISM '11, 111–116, Washington, DC, USA. IEEE Computer Society.

[14] Qiu, Y., Guan, G., Wang, Z., & Feng, D. 2010. Improving news video annotation with semantic context. In *Proceedings of the 2010 International Conference on Digital Image Computing: Techniques and Applications*, DICTA '10, 214–219, Washington, DC, USA. IEEE Computer Society.

[15] Hu, W., Xie, N., Li, L., Zeng, X., & Maybank, S. 2011. A survey on visual content-based video indexing and retrieval. *IEEE Transactions on Systems, Man, and Cybernetics Part C: Applications and Reviews*, 41(6), 797–819.

[16] Imran, A. S. & Cheikh, F. A. 2011. Blackboard content classification for lecture videos. In *Proceedings of The International Conference on Image Processing (ICIP)s*, 2989–2992.

[17] Das, D., Chen, D., & Hauptmann, A. G. 2008. Improving multimedia retrieval with a video ocr. In *Proceedings of SPIE*, volume 6820, 68200B–68200B–12.

[18] Chakravarthy, A. 2006. Cross-media document annotation and enrichment. In *Proceedings of the 1st Semantic Authoring and Annotation Workshop (SAAW2006)*.

[19] Wang, F., Lu, W., Liu, J., Shah, M., & Xu, D. 2008. Automatic video annotation with adaptive number of key words. In *Proceedings of The International Conference on Pattern Recognition (ICPR 2008)*, 1–4. IEEE.

[20] Altadmri, A. & Ahmed, A. 2009. Automatic semantic video annotation in wide domain videos based on similarity and commonsense knowledgebases. *IEEE International Conference on Signal and Image Processing Applications (ICSIPA)*, 74 – 79.

[21] Princeton University. 2010. About WordNet. `http://wordnet.princeton.edu`.

[22] Miller, G. A. 1995. Wordnet: A lexical database for english. *Communications of the ACM*, 38(11), 39–41.

[23] Didion, J. 2004. JWNL (Java WordNet Library). `http://sourceforge.net/projects/jwordnet/`.

[24] CSAIL MIT. The MIT Java Wordnet Interface. `http://projects.csail.mit.edu/jwi/`.

[25] Bird, S. 2006. Nltk: the natural language toolkit. In *Proceedings of the COLING/ACL on Interactive presentation sessions*, COLING-ACL '06, 69–72, Stroudsburg, PA, USA. Association for Computational Linguistics.

[26] Bou, B. 2010. WordNet Sql. `http://wnsql.sourceforge.net/`.

[27] Pedersen, T., Patwardhan, S., & Michelizzi, J. 2004. Wordnet::similarity: measuring the relatedness of concepts. In *Demonstration Papers at HLT-NAACL 2004*, HLT-NAACL–Demonstrations '04, 38–41, Stroudsburg, PA, USA. Association for Computational Linguistics.

[28] Li, H., Tian, Y., Ye, B., & Cai, Q. oct. 2010. Comparison of current semantic similarity methods in wordnet. In *Proceedings of The 2010 International Conference on Computer Application and System Modeling (ICCASM)*, volume 4, V4–408 –V4–411.

[29] Koprinska, I. & Carrato, S. 2001. Temporal video segmentation: A survey. In *Signal Processing: Image Communication*, 477–500.

[30] Kilinc, D. & Alpkocak, A. September 2011. An expansion and reranking approach for annotation-based image retrieval from web. *Expert Systems with Applications*, 38(10), 13121–13127.

[31] Deschacht, K., Moens, M., & Law, I. C. F. 2007. Text analysis for automatic image annotation. In *Proceedings of the 45 th Annual Meeting of the Association for Computational Linguistics. East Stroudsburg: ACL*.

[32] Navigli, R. February 2009. Word sense disambiguation: A survey. *ACM Computing Surveys*, 41(2), 10:1–10:69.

[33] Agirre, E. & Edmonds, P. 2007. *Word Sense Disambiguation: Algorithms and Applications*. Springer Publishing Company, Incorporated, 1st edition.

[34] Banerjee, S. & Pedersen, T. 2002. An adapted lesk algorithm for word sense disambiguation using wordnet. In *Proceedings of the Third International Conference on Computational Linguistics and Intelligent Text Processing*, CICLing '02, 136–145, London, UK, UK. Springer-Verlag.

[35] Castillo, J. J. 2010. A semantic oriented approach to textual entailment using wordnet-based measures. In *Proceedings of the 9th Mexican international conference on Advances in artificial intelligence: Part I*, MICAI'10, 44–55. Springer-Verlag.

[36] Young, P. 1995. Optimal voting rules. *The Journal of Economic Perspectives*, 9(1), 51–64.

A Program Documentation

Program Documentation

Automatic Semantic Annotation for Media Learning Objects

Laksmita Rahadianti

5/7/2012

Contents

1 Introduction

This document will explain the necessary information needed in order to execute or build upon the program developed for the thesis. The program executes the processes explained in the master thesis main document [1]. The program will be able to analyze the appropriate keywords to be used in annotating lecture videos that fulfill certain requirements.

1.1 Platforms

The program was mostly developed on the Java Standard Edition version 7. It is necessary to install the Java Development Kit (JDK) in order to be able to develop Java programs. The necessary installers, documentation, demo, and samples can be obtained from [2]. The implementation was done using an integrated development environment (IDE). This choice was taken in order to be able to import the necessary libraries that would be needed in the development. The particular IDE used was Eclipse [3].

Some parts of the system were implemented on the Matlab platform, specifically Matlab R2010b. This was mostly done in order to be able to use image and video processing functions. The program is realized in adjoining m-files that are executable.

1.2 Necessary Libraries or Plug-Ins

For the Matlab-based sections of the program, no additional plug-ins are needed. The Matlab program used must be at least versions Matlab2008b and onwards. No additional configurations or libraries are needed.

For the Java-based sections, on the other hand, require certain additional libraries. It is advisable to use Eclipse although the program will run just fine with any IDE. The further explanation of the program will assume the use of Eclipse. In order of the program to run, it is necessary to use the following libaries:

1. The MIT Java WordNet Interface (MTIJWI) [4]. This is a Java library implemented by the Massachusetts Institute of Technology as an interfacing tool for Java programs and the WordNet database files. This library is needed to be able to query and access WordNet functions such as glossaries, antonyms, synonyms, derivations, etc.

2. Java WordNet::Similarity (JWS) [5]. This is the Java implementation provided by David Hope at Sussex University to encapsulate the original Perl-based version of WordNet semantic similarity metrics [6]. This library provides us with the implementation and functions of various semantic similarity metrics.

Both of the packages mentioned are available at [4, 5] in the form of a Java Archive (.jar) file. This jar file needs to be included in the project.

1.3 System Division

The entire process is shown in Figure 1. It is clear that the entire system can be divided into two main sub-systems, mainly the Temporal Video Segmentation sub-system as well as the Semantic Keyword Selection sub-system. A detailed explanation of this system is described in the thesis [1]. This document will only explain the program functions.

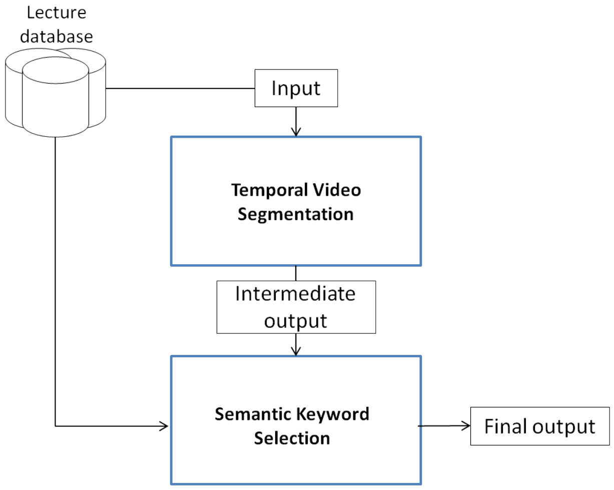


Figure 1: Overall design of system.

1.3.1 Temporal Video Segmentation

This sub-system needed 2 different functions, namely video processing and text processing, therefore both Matlab and Java were used. The sub-system can be visualized in Figure 2. There are 2 main components to this sub-system, namely video shot boundary detection and transcript segmentation. Each component is implemented individually on a different platform, creating Module 1: Video Shot Boundary Detection and Module 2: Transcript Segmentation.

1.3.2 Semantic Keyword Selection

The semantic keyword selection subsystem is not divided into any components, and implemented as a whole. Therefore this whole sub-system is also

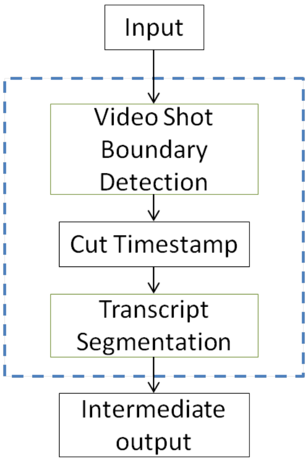


Figure 2: Temporal video segmentation sub-system.

referred to as Module 3: Semantic Keyword Selection. This module is also implemented in Java, and designed to be run on Eclipse.

2 Modules

This section will describe in detail each module of the program. Each module function as well as its setup will be explained.

2.1 Module 1. Video Shot Boundary Detection

This module is implemented on Matlab and can be run on any Matlab installation versions 2008b and onwards. After the Matlab console is running, it is necessary to change the Matlab directory to the folder where the program is. The program can then be run from that active folder. Some additional input are needed for the program to run. These files are needed inputs, namely `stopwords.dat` and `categories.dat`.

The input to this module is the video itself. The video should can be in any video format, but the .avi format is recommended. The input files should be put in the same folder, entitled `lecture`*i*`.avi`, with i being the number of the video. As an example, the video can be `lecture1.avi`. This is to enable more than one video to be processed in sequence.

The program can be invoked in two ways. The first is to process each video individually, by running the $[\texttt{min}, \texttt{singleSec}] = \texttt{segmentVideo}(\texttt{filename}, \texttt{n})$ command in the Matlab command prompt. The input arguments are the name of the video file, and the interval of frames to be processed (`n`). If `n` is set to 1, the program will process every successive frame, if `n` is 10, the program will process every 10 frames, and so forth. The return argument `min` is the time stamp in the `min` : `sec` format at which the cut occurs, and `singleSec` is the second at which it occurs.

Additionally multiple videos can be processed sequentially, using the command $\texttt{ProcessVideos}(\texttt{a}, \texttt{b})$, with `a` being the starting index i of the first video to be processed and `b` is the ending one. So if the processing was to be done on videos 2 to 9, the command would be $\texttt{ProcessVideos}(2, 9)$. The results are then written directly on result files $\texttt{lecture}i\texttt{.seg}$ which contains the timestamps of all cuts, if any exist. These outputs are needed for the next module.

2.2 Module 2. Transcript Segmentation

This module is implemented in Java, and is recommended to be run on Eclipse. After the Eclipse program is running, the project should be imported into the Eclipse workspace. This module's folder includes the `.project` file which is the Eclipse project configuration file.

This module does not need any additional Java plug-ins or libraries, but it is crucial to remember the input files. The resulting `.seg` files from the video shot boundary detection should be included by copying them into this module's folder.

The program is run by simply running the file `SegmentTranscript.java` from the console. The program must be edited on this file to determine which lectures are to be processed. The line 9 should be edited accordingly to accommodate the lecture numbers to be processed. The program gives an output of the segmented transcripts in `.trans` files. The lectures will be divided into multiple transcript files according to the cuts detected, for example the $\texttt{lecture4}_1\texttt{.trans}$ file, which is the transcript file of lecture 4, segment 1. These files are the intermediate output and will be needed later in the semantic keyword selection module.

2.3 Module 3. Semantic Keyword Selection

The semantic keyword selection module is also implemented in Java, and designed to be run on Eclipse. The project should be imported into the Eclipse workspace just as the previous module in 2.2 through the `.project` Eclipse project configuration file.

There are 4 different methods implemented for this module, namely Term Frequency Inverse Document Frequency (TF-IDF), Visualness with Disambiguation by Category (VDC), Visualness with Lesk Disambiguation (VLD), and Lesk Visualness and Disambiguation with Frequency of Occurence (LVD-F). Each method is implemented into its own project, but is executed in the same fashion. Depending on which method is desired, the corresponding `.project` file should be loaded.

This module needs both the libraries mentioned in 1.2 to be able to run. It is necessary to add both the MTIJWI and JWS `.jar` files to the build path in the project properties. This is configurable in Eclipse, and the files are `mti.jwi.jar` and `sussex.jws.jar`, respectively.

Default input files are needed for this module, which are `stopwords.dat` and `categories.dat`. The `stopwords.dat` contains the stop words needed for the keyword extraction and `categories.dat` contains a list of categories. The categories are needed for the semantic analysis. These categories are a list of all the possible categories in the video database, and this list should separate each category with a simple comma (,).

The input to this module is the intermediate output resulted in the entire temporal video subsystem in Section 2.2, which was the segmented transcript `.trans` files. These files must be copied into this semantic keyword selection module's folder.

The program is run the program at the `MainKeywordProcess.java` file. The program must be edited on this file to determine which lectures are to be processed. The line 14 should be edited accordingly to accommodate the lecture numbers to be processed. This program will result into `.csv` files. These files will contain the words extracted and their corresponding scores. These files are the final output of the program

References

[1] Rahadianti, L. Automatic semantic annotation for media learning objects. Master's thesis, Hogskolen i Gjovik, 2012.

[2] Oracle. 2012. Java standard edition 7. `http://www.oracle.com/technetwork/java/index.html`.

[3] Eclipse Foundation. 2012. Eclipse. `http://www.eclipse.org/`.

[4] CSAIL MIT. The MIT Java Wordnet Interface. `http://projects.csail.mit.edu/jwi/`.

[5] Hope, D. 2008. Java WordNet:: Similarity. `http://www.sussex.ac.uk/Users/drh21/`.

[6] Pedersen, T. WordNet::Similarity. `http://wn-similarity.sourceforge.net/`.

B Accepted Academic Paper

The following academic paper was written and submitted as a four page technical paper to The Sixth IEEE International Conference on Semantic Computing (ICSC) in Palermo, Italy. The paper was written in the IEEE academic paper format and submitted on May 17th 2012. On July 9th 2012 this paper has been accepted for publication and presentation at the conference.

This academic paper covers the first two objective scoring methods based on WordNet semantic similarity and visualness presented in Section 3.5, which are Visualness with Disambiguation by Category (VDC) and Visualness with Lesk Disambiguation (VLD).

The website of the conference can be accessed at **http://icsc2012.pa.icar.cnr.it/**.

Semantic Tags for Lecture Videos

Ali Shariq Imran, Laksmita Rahadianti, Faouzi Alaya Cheikh, Sule Yildirim Yayilgan
Gjøvik University College
P.O.Box-191, N-2802, Gjøvik, Norway
ali.imran@hig.no, laksmita.rahadianti@hig.no

***Abstract*—In an effort to understand the development of effective multi-media learning objects (MLO), we propose a framework to extract and associate semantic tags to temporally segmented instructional videos. These tags serve for the purpose of efficient indexing and retrieval system. We create these semantic tags from potential keywords extracted from the lecture transcript. The keywords undergo a series of refinement process to select few but meaningful set of tags. Finally, the tags are associated with video segments. Each video segment represents a key idea or a topic. We also evaluated the objective keyword selection criteria to subjective test with some interesting results.**

I. Introduction

Recent advances in the e-learning technologies coupled with the significant increase in the Internet growth have led to the widespread use and availability of digital lecture videos [1]. Most of these videos are distributed through various learning management systems and through online education portals. These lecture videos contains a large set of instructional content. Usually the instructional content contains ubiquitous information about the lecture subject. This, however, poses a great challenge to search and retrieve the right content from such a huge amount of data [2], and so does their efficient indexing.

Among many other retrieval techniques, users are most commonly accustomed to employ text-based queries. These textual queries must be matched somehow to the non-textual lecture videos for retrieval purpose. Therefore, it is necessary to annotate the videos with textual keywords which are highly relevant to the video content [3]. Generally, lecture videos do not contain rich visual content. Keywords are often extracted from text in the lecture content of the video (slides or blackboard) [4]. Nevertheless, in the distribution of learning objects in e-learning systems, lecture videos are rarely delivered on their own. They are bundled with lecture transcripts and additional notes. Therefore, it is logical to include these additional documents while doing video annotation; using them as sources of potential keywords and introducing cross document annotation [5].

Furthermore, additional semantic annotation can enrich the annotation massively. Semantics is a branch in linguistics that focuses on the study of meaning. Concepts in this world do not usually exist on their own, but instead they have relationships among themselves, creating a semantic network of concepts. Using this so-called commonsense understanding, it is possible to correlate words and create an enhanced set of words in correlation to the video. This has been done previously in news videos such as in [6] and [7]. In this research, the semantic analysis was carried out using a preexisting semantic network called WordNet [8].

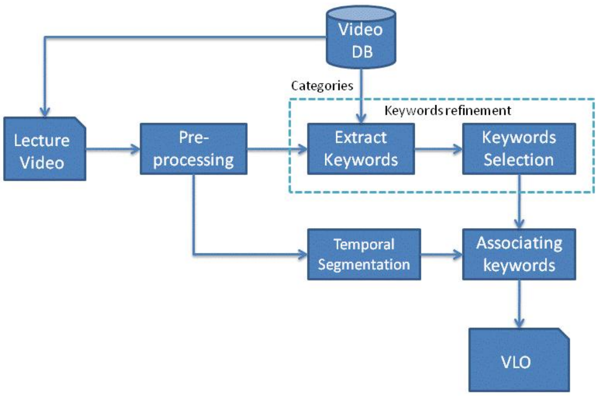


Fig. 1. Proposed Framework Design

The rest of the paper is organized as follows. Section 2 describes our proposed framework design. In Section 3, we present the experimental setup. Section 4 highlights the experimental and objective results followed by conclusion in Section 5.

II. Framework Design

The proposed framework consists of 5 modules as shown in Figure 1. The input to the framework system is a raw instructional video. The pre-processing module prepares the raw video for further processing by removing noise, localizing, enhancing and extracting text portions, and removing any portion of a video with no pedagogical value. The video is then segmented temporally. Candidate keywords are extracted both from the text regions and any additional material provided as part of the raw video. The additional material can be lecture slides, lecture notes and/or transcripts. The keywords then go through a set of refinement processes (i.e. selection criteria) to obtain best suitable words describing the context of the video. We used two different techniques and compared their results to select the best suitable keywords. Automatically selected keywords were evaluated against the keywords selected by the viewers. In the end, the potential keywords are then refined to a meaningful set of tags and associated to video learning objects (VLO).

The modules are explained as follows:

A. Pre-processing Module

The purpose of pre-processing module is to prepare raw video for spatial and temporal segmentation. Depending upon the type of the input lecture video, we apply different processing steps. For example, for typical instructor-led lecture video with traditional handwritten text on blackboard, we separate the background content regions (i.e. blackboard) from the foreground objects (i.e. instructor). This is to get a clear picture of the background text for further processing. We do this by creating a foreground model of the moving object in a video frame and then removing the foreground object from the original frame [9]. This process gives us the background content frame without any instructor in it. We then apply a series of morphological filtering to remove left over noise as a result of foreground/background segmentation and to enhance text quality. Later, we apply the labeling procedure to localize and extract the text. For smart board and power point presentation videos, we directly process the frames for text localization and extraction.

B. Temporal Video Segmentation

Temporal video segmentation is carried out in order to create efficient learning material such as VLO [1]. The idea behind segmenting a video in time domain is to create an educational content of high pedagogical value. We do this by identifying and removing inactive scenes from the instructional video i.e. where there is nothing significant happening. For unscripted instructor-led lecture videos, we analyse the motion pattern of foreground object (i.e. the instructor) to classify different key states. We classify 4 states as writing, erasing, speaking and idle [10]. An idle state is one where the instructor is neither speaking nor doing any other critical activity. We truncate the lecture video by removing idle states. Next, we segment the video based on change in content or topic. For this we compare vertical projection of the text regions among two frames. A change in content is detected if the difference between two projections exceeds a specified threshold.

C. Keyword Extraction Module

The keywords to be used in the proposed framework is done with the cross document annotation principle. This means that the initial list of potential keywords is taken from an associated textual document. We make use of the accompanying subtitle (transcript) file of the video. After the potential keyword list is extracted from the subtitle, additional text processing is done to obtain a refined set of potential keywords. The text preprocessing comprises of the following steps.

1) Tokenizing the transcript to a list of individual words.
2) Removing the stop words from the text [11].
3) Bringing each word to its singluar form.
4) Removing multiple word occurrence.

The english language contains 4 different parts of speech (POS): nouns, verbs, adverbs and adjectives. WordNet does not enable operations between words of different POS, therefore, it is necessary to distinguish them from one another. It is commonly argued in this field of research that language semantics are mostly captured by nouns. Thus, nouns are often used in semantic applications [12]. This holds true for visual media as well, as the first things that are noticeable are usually things or entities in the scene. Therefore, the first step is to extract all the nouns from the words we obtain from the subtitles, and create a noun list N.

In our implementation, we expanded the keyword domain to the verb POS. If nouns are used due to the property of denoting physical world entities, the verbs denote the relation and interaction of these entities. Hence, this research also goes through the initial keyword list, searching for verb form lemmas, and resulting into a verb list V. A lemma is the most basic form of a word [11] . Adverbs and adjectives are are usually excluded in semantic analysis applications because they do not follow the same hierarchy in WordNet as the verbs and nouns. In an attempt to utilize all the available information, including adverbs and adjectives, we employ the derivation property. For example, the adjective “happy” can be derived to the noun happiness. These noun derivations are then added to N. As a last step, this derivation is also performed on the nouns in N to obtain verb derivations and add them to V and vice versa.

D. Keyword Selection Module

The keyword selection module makes use of both semantic similarity measures and visualness. Semantic similarity between two words is a measurement of how similar or how related they are. Similarity measures can only be calculated on nouns and verbs, which is the reason behind the POS processing done in the Keyword Extraction Module explained in section II-C. Many similarity measures are available and this research does not go deeper in this area. We utilize commonly known similarity measures that are already available [13].

Visualness is used to quantify the capability of visual illustration of a word [6]. The process of calculating visualness begins with determining a set of seed words over the whole set of videos. Then, for each video the visualness of these seeds are annotated to either 1 or 0, indicating whether or not the seed is visible in the video. This is denoted by $vis(s_i)$ for each seed s_i in the set of seeds $S = \{s_1, s_2, ...s_i\}$. Based on these seed words and their visualness values, together with the similarity measure of choice $sim(word_1, word_2)$, the visualness of each potential keyword w can be calculated as:

$$vis(w) = \sum_i vis(s_i) \frac{sim(w, s_i)}{\sum sim(w, s_i)}$$

After revisiting the formula, it is clear that the formula of any given word w is equivalent to the sum of the similarities between w and all seeds present in the frame ($\{\forall\, s_i \mid vis(s_i) = 1\}$) divided by the sum of the similarities between w and all possible seeds ($S = \{s_1, s_2, ...s_i\}$) as:

$$vis(w) = \frac{sum_i(sim(w, s_i))}{sum_j(sim(w, s_j))}$$

$$where\, s_i = \{s_i \mid s_i \in S \wedge vis(s_i) = 1\}\, and\, s_j = \{s_j \mid s_j \in S\}$$

Using these concepts we calculate the visualness of each potential keyword from the transcript. We propose two approaches to calculate visualness of keywords in an automatic fashion. The first one uses disambiguation by category and the second one uses Lesk algorithm disambiguation.

1) Visualness with Disambiguation by Category (VDC): We adopted the visualness described in [6] to our needs by using categories and title as seeds. To avoid the manual determination of the visualness value of these seeds, all title words were assumed to be visual in the video, having a visualness of 1. In addition to this, we utilize the categorization of the lecture videos. The database comprises of videos from different categories, and each video has it own category. These categories can be used as the global concepts for seeds.

As for the visualness values, we maintain the automatic nature of the process and avoid any manual processing. With categories as seed words, the category a lecture belongs to will have the visualness value of 1 while, the other categories are all set to 0. Due to the POS processing, we work with 2 lists of potential keywords, the noun list N and the verb list V. Similarly, the seed words are also put through POS processing, resulting into verb seeds Sv and noun seeds Sn. The visualness calculation is then done on each POS separately.

We further addressed the issue of multiple senses in a word. For example, the word ”cancer” can be used as a disease or it can be used as a zodiac sign. It is therefore necessary to determine which sense of a word is relevant in the context. A word disambiguator is needed in order to determine the correct senses. Both seed word lists Sn and Sv consisting of available categories are set to its first sense. As these words are predefined before the annotation process, this will not affect the automatic nature of the annotation. The additional words from the title of the video, however, requires sense determination. To establish this automatically, similarity is calculated between each sense of the word with the pre-defined category of the lecture. The sense with the highest similarity value will be set as the sense of the seed word. For each seed word $s \in Sn$ or $s \in Sv$ that originated from the title, its sense $Se(s)$ is determined by finding the sense i that maximises the similarity sim to the category as:

$$Se(s) = i|max(sim(category, s_i))$$

Using this modified interpretation of seed words and a disambiguated-by-category seed list, the visualness according to this method could be computed. VDC is illustrated in Figure 2, in which the *disambiguator* process employs the disambiguation by category method.

2) Visualness with Lesk algorithm Disambiguation (VLD): In this method the principle on the seed word determination and the POS processing is done in the same way as VDC. The main difference here is in the sense disambiguation process. In the ongoing research of Word Sense Disambiguation (WSD), a popular solution is the Lesk algorithm. The Lesk algorithm attempts to correlate word sense pairs using the words contained in the words respective glosses. A gloss is a definition of a given word.

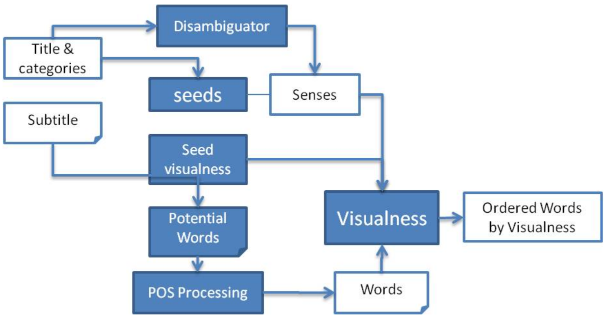


Fig. 2. Visualness Calculation

For the sake of simplicity, our application employs the simplified Lesk algorithm. Take any word w, which is found in a context of n different words $context(w) = c_1, c_2, ..c_n$. The word w still has m multiple senses $Se(w) = sw_1, sw_2, ..sw_m$. Therefore, the score is calculated for each senses of a word $sw_i \in Se(w)$ and the its' context [14]. The assigned sense of w is the i that maximizes the Lesk value given by the following formula.

$$Lesk(sw_i) = |context(w) \cap gloss(sw_i)|$$

In order to disambiguate the lecture title, the whole title serves as the context. In Figure 2, VLC uses the Lesk algorithm in the *disambiguator* process block.

III. Experimental Setup

To evaluate the keyword selection criteria we conducted an online survey based experiment. We prepared a list of keywords whereby asking viewers to select 5 best possible keywords describing the video segment. This list of candidate keywords were extracted from the lecture transcript. Transcripts were processed as described in Section II-C to generate possible candidate keywords. We used 5 videos for this experiment and each video was 2-3 minutes in duration. To create these video segments, we used the lecture videos available at Khan Academy website [15]. It is a non-profit organization which has a collection of more than 3000 videos containing educational lectures in topics ranging from mathematics to art history. The reason for opting online website was the lack of standardised database of lecture videos and the availability of the lecture transcript. It also served us greatly due to the hierarchal category structure in which the videos are organized. We then compared the subjective results obtained from the experiment to the objective results obtained by keyword selection algorithms. To do the comparison, we computed the frequency of occurrence of subjective results.

IV. Results

We computed the frequency of occurrence of each word from 18 participants. We then selected the top 5 words that best describe a particular video segment. These words are

Word	Subjective Score	Objective Score	
	Frequency of Occurrence	VDC	VLD
binary	17	0.99	0.99
base	15	0.28	0.29
number	10	0.99	0.99
digit	9	0.27	0.27
digitize	7	0	0

TABLE I
LECTURE VIDEO 1

Word	Subjective Score	Objective Score	
	Frequency of Occurrence	VDC	VLD
cell	17	0.99	0.99
nucleus	13	0.99	0.42
biology	11	0.99	0.99
membrane	11	0.39	0.405
DNA	10	0	0

TABLE II
LECTURE VIDEO 2

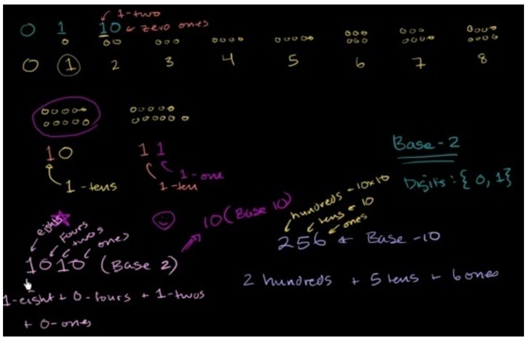


Fig. 3. Screen shot of lecture video 1.

sorted in order of highest frequency count from subjective score as shown in Table I and II. We then refer back to the automatic results from VDC and VLD to analyze the objective scores that the objective metrics assign to these 5 words. The objective scores are in a range of 0 - 1, where 1 indicates a high relevance word and 0 a low relevance word.

The words selected by the subjects reflect the video under observation as can be seen in Figure 3. The score obtained by the objective metrics are similar to those selected by the subjects for the two videos except for 'digitize', 'digit' and 'base' word for video 1. This is due to the fact that these words get low similarity value based on visualness when compared to the ground truth values.

In general, the two approaches give equal importance to the words selected by the observers with slight variation. For instance, for the second video VDC is slightly better in case of "nucleus" while VLD puts "membrane" at higher level than VDC. From these initial results it is safe to deduce that the top 5 words picked up by the automatic methods are very close to the ones chosen by the human subjects. However, the performance of these metrics varies depending upon the POS words extracted from lecture video and the initial seed words.

V. CONCLUSION

In this paper, we propose a framework design to associate keywords extracted from lecture video and it's transcript for efficient indexing and retrieval purpose. We used the title of video and the categories as ground truth for selecting potential keywords. We then choose the best possible words using similarity measure and disambiguation criteria as visualness between different senses of the words. We proposed two metrics to do this objectively. The words that best describe the video segments are selected as tags. These tags are then associated to temporally segmented video in which they appear. We also performed subjective experiment and compared the results to those obtained by objective metrics. We are now in a process of embedding an optical character recognition module to extract keywords from within the content of the video automatically. As a future work, we plan to develop a metric that select words as close as possible to those of human observers.

REFERENCES

[1] Ali Shariq Imran, "Interactive media learning object in distance and blended education," in *Proceedings of the 17th ACM international conference on Multimedia*, New York, NY, USA, 2009, MM '09, pp. 1139–1140, ACM.

[2] Henk M. Blanken, Arjen P. de Vries, Henk Ernst Blok, and Ling Feng, *Multimedia Retrieval (Data-Centric Systems and Applications)*, Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2007.

[3] Carlos A. F. Pimentel Filho and Celso A. S. Santos, "A new approach for video indexing and retrieval based on visual features," *JIDM*, vol. 1, no. 2, pp. 293–308, 2010.

[4] Haojin Yang, Maria Siebert, Patrick Luhne, Harald Sack, and Christoph Meinel, "Automatic lecture video indexing using video OCR technology," in *Proceedings of the 2011 IEEE International Symposium on Multimedia*, Washington, DC, USA, 2011, ISM '11, pp. 111–116, IEEE Computer Society.

[5] Fabio Ciravegna Ajay Chakravarthy and Vitaveska Lanfranchi, "Cross-media document annotation and enrichment," in *Proceedings of the 1st Semantic Authoring and Annotation Workshop (SAAW2006)*, 2006.

[6] Yu Qiu, Genliang Guan, ZhiyongWang, and Dagan Feng, "Improving news video annotation with semantic context," in *Proceedings of the 2010 International Conference on Digital Image Computing: Techniques and Applications*, Washington, DC, USA, 2010, DICTA '10, pp. 214–219, IEEE Computer Society.

[7] A. Altadmri and A. Ahmed, "Automatic semantic video annotation in wide domain videos based on similarity and commonsense knowledgebases," *IEEE International Conference on Signal and Image Processing Applications (ICSIPA)*, pp. 74 – 79, 2009.

[8] George A. Miller, "Wordnet: A lexical database for english.," *Commun. ACM*, vol. 38, no. 11, pp. 39–41, 1995.

[9] Ali Shariq Imran and Faouzi Alaya Cheikh, "Blackboard content classification for lecture videos," in *ICIP*, 2011, pp. 2989–2992.

[10] Ali Shariq Imran and Faouzi Alaya Cheikh, "Multimodal activity recognition in lecture videos," http://www.ansatt.hig.no/alii/BMVCunderreview.pdf.

[11] Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schutze, *Introduction to Information Retrieval*, Cambridge University Press, Cambridge, UK, 2008.

[12] Haisheng Li, Yun Tian, Ben Ye, and Qiang Cai, "Comparison of current semantic similarity methods in wordnet," in *2010 International Conference on Computer Application and System Modeling (ICCASM)*, oct. 2010, vol. 4, pp. V4–408 –V4–411.

[13] Ted Pedersen, Siddharth Patwardhan, and Jason Michelizzi, "Wordnet: : Similarity - measuring the relatedness of concepts.," in *AAAI*, 2004, pp. 1024–1025.

[14] Eneko Agirre and Philip Edmonds, *Word Sense Disambiguation: Algorithms and Applications*, Springer Publishing Company, Incorporated, 1st edition, 2007.

[15] The Khan Academy, "Khan Academy," http://khanacademy.org.

C Submitted Academic Paper

The following academic paper was written and submitted as a twelve page full length paper to The 11th Asian Conference on Computer Vision (ACCV) in Daejeon, Korea. This academic paper is submitted in the Lecture Notes on Computer Science (LNCS) academic paper format, which is a one-column format, and submitted on July 1st 2012. The paper is currently still under review of the committee and the review process is hoped to provide additional insight for this research. The paper review for this conference is double blind, so the submission was done anonymously. Apart from Laksmita Rahadianti, this paper was jointly authored by Ali Imran Shariq, Faouzi Alaya Cheikh, and Sule Yildirim Yayilgan.

The paper covers the majority of the thesis work, focusing heavily on the developed semantic scoring methods (VDC, VLD and LVD-F) in Section 3.5 and their comparison against the traditional TF-IDF method. The experiments and results of the thesis work as shown in Section 4 and Section 5.

The website of the conference can be accessed at **http://www.accv2012.org/main/**.

Semantic Keyword Selection for Automatic Video Annotation

Anonymous ACCV 2012 submission

Paper ID **

Abstract. Choosing the right keywords to best describe digital media content is crucial especially for indexing and retrieval purposes. Generally these keywords are selected manually, which is in general restrictive to a set of words, subjective to the annotation, and labor intensive. In this paper, we therefore propose automatic keyword selection methods for annotating video. We specifically used lecture videos and surrogate documents e.g. transcripts to extract potential candidate keywords. These potential keywords are then filtered based on a set of seed words to select a few but meaningful set of keywords. The seed words are extracted from the title of the video and subject category. We propose three new objective methods to select top ranking keywords based on visual similarity and word sense disambiguation (WSD). These selected keywords are then compared to the subjectively selected keywords obtained experimentally. The proposed ranking methods are also compared to traditional term frequency inverse document frequency (TF-IDF) based method. The obtained results shows that the words selected by the proposed objective methods correlate highly with those selected by a set of viewers. In general, the three proposed methods perform better than the traditional TF-IDF method, and the LVD-F method is able to obtain the highest precision and recall of all.

1 Introduction

Nowadays, the amount of publicly accessible information has grown, mostly due to the integration of the world wide web and modern computer technology in everyday life. This has not only brought multimedia as a new communication form, but quite possibly the first original communication form of the computer age. Quite often this information is in the form of raw data that can be used to obtain useful information. Normally, the data is not used directly, but must be organized and processed in such a way that it is meaningful to the users. Not long ago, the data was commonly presented in the form of text, recently this can come in many different forms including audio and video files. In any large collection of data, retrieving relevant information from the set is a necessity. Traditional information retrieval techniques can not be applied directly to these multimedia files due to their non-textual nature. This calls for reliable techniques for the analysis, search, and management of multimedia data, as well as distributed system architectures in which these techniques can be embedded to effectively help users find relevant data effectively.

We focus our attention in selecting semantic keywords useful for describing multimedia videos for efficient indexing and retrieval. In many applications, it is desirable for a user to be able to post a query to the system in order to retrieve relevant videos. Although many query methods are available [1], users are still accustomed to use textual queries in the form of a set keywords. This calls for the video to be annotated with keywords that describe the video content accurately. Thus, the focus of our research is determining which keywords are the most suitable annotations. Selecting appropriate keywords is deemed necessary in order to achieve this. Keyword selection takes into consideration a list of words, and determines which of these words are the best to select. Previous research done on keyword selection [2, 3], used keyword selection to increase profits in search-based advertising services. [4], utilized keyword selection to enhance search and retrieval of large bodies of text. An adaptive approach [5], processed keywords semantically to obtain an adapted list of selected keywords.

Many approaches use the visual track of the video to extract objects, which are then used as keywords such as in [6, 7]. These are not feasible for videos which are not rich in visual content such as lecture videos, speeches or interviews. This is why alternate methods are needed. One of which is cross document annotation [8], in which we no longer consider the video on its own, but additionally we take into consideration related documents which may potentially be in textual form. Extending the context to more than single media source provides a richer source of potential keywords.

In addition, we also employ additional semantic annotation. Concepts in the real world do not occur on their own. Real-world concepts have relationships, such as ”boat” and ”captain” which are obviously related. It is then necessary to have semantic networks which model these relations clearly. Many approaches embrace this idea in order to merge both visual features and semantics on a wide range of videos [9, 10]. This so-called commonsense understanding can correlate these concepts and enhance the annotation to a higher semantic level.

The rest of the paper is organized as follows. Section 12 describes our proposed methods in detail. In Section 3, we present the experimental results and their analysis. Section 4 concludes our paper.

2 Proposed Methods

We used the cross document annotation principle to semantically select top ranked keywords for video annotation. These keywords are selected automatically using visual similarity and word sense disambiguation. The proposed methods are part of the multi-media learning objects (MLO) [11]. Figure 1 shows a framework. Input to this framework is a lecture video with accompanying text material for cross document annotation. The additional text sources may include audio transcript, title of the video and video categories. Since we used lecture video database, the video categories are the different educational subjects under which a particular video is listed. This is feasible due to the fact that in today’s e-learning systems and online digital video repositories, lecture videos are often

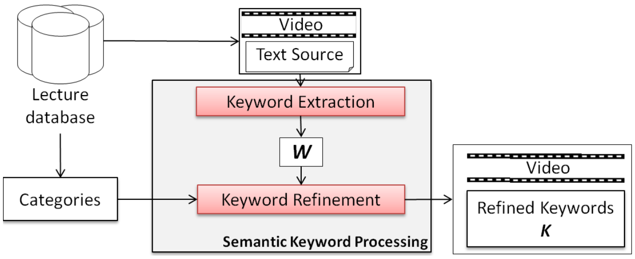


Fig. 1. The system framework.

bundled together with additional sources of text, such as lecture transcripts, course slides, or e-books as in TedTalks and Khan Academy [12].

The first step is to obtain an initial list of potential keywords which are extracted from the additional text source. This additional text source is processed in the following steps to obtain a clean set of meaningful keywords.

1. The text is cleaned by removing all punctuation and capitalization.
2. A tokenizer is used to separate the text into individual words.
3. We remove all stop words. Stop words are words in the English dictionary such as ”and” or ”is” which don’t deliver any significant meaning on their own. [13].
4. Finally we remove all duplicate occurrences of each word.

The potential keywords that we obtained as a result can either be a noun, verb, adverb or adjectives. These are different parts-of-speech (POS) of a language. It is commonly argued that the language semantics are mostly captured by nouns. This holds true for visual media as well, as the first things that are noticeable are usually objects or entities in a scene, and since nouns denote physical word entities, they are often used in semantic applications [14]. The other POS are then usually discarded. However, rather than discarding these POS we utilize them to extract further relevant nouns that could be used as potential keywords. For example, the noun happiness can be derived from the adjective ”happy”. All verbs, adjectives, and adverbs are then used in similar fashion, and their noun derivations (if they exist) are added to the noun list.

After this process, we obtain a clean set of potential keywords W from the text, from which to select the best meaningful words. We do this by semantically separating the meaningful keywords from the rest. To do this, we feed the potential keywords W into keyword refinement as shown in Figure 1.

We propose three objective scoring methods based on visualness [7] and disambiguation [15] concepts for semantic analysis. These methods are designed to automatically select keywords extracted from a lecture video, using semantic relation between words. The semantic relation comes from the visualness. Visualness measures the ability of a word to accurately describe the context. This visualness measure involves similarity measures between a given candidate word and visual seed words. The visual seeds are the ground truth against which the visualness of a candidate word is determined. Previously, as in [7], the seed words as well as their visualness were calculated manually in each frame. We however, extract these seed words automatically from the title of the video and the category in which a video is listed. For example, consider the seed words extracted from a video listed under "algebra" category entitled "Linear Equations: Solving the Inequality". The category of this video (”algebra”) would be given a visualness value of 1 while for all other category seed words it is set to 0. Additional seed words extracted from title will be "'Linear', "Equations" and "'Inequality', are also set to the value of 1.

Visualness employs the semantic similarity metric between words. Semantic similarity measures how related two words are. There are many different variations of semantic similarity measures that can be used to calculate the visual score [14]. We used the WordNet [16] to calculate the visualness of any given word $w \in W$ according to the following formula:

$$vis(w) = \sum_i vis(s_i) \frac{sim(w, s_i)}{\sum sim(w, s_i)},$$

where $i = 1, 2, ...n$ for the whole set of n seed words $s_1, s_2, ...s_n$, and $sim(w, s_i)$ is the semantic similarity between the word w and a seed word s_i. The visualness of a given seed word s_i is denoted by $vis(s_i)$. The resulting visualness score $vis(w)$ is in the range of 0 to 1.

In English language, words can have dual meanings. For example, the word "cancer" can either refer to a disease or it can be used in a sense of zodiac sign. It is therefore, necessary to understand in which context a particular word is used. This can be addressed using word sense disambiguation (WSD) [15]. In our implementation, it is crucial to determine which sense is intended in the context of the video. We use two WSD techniques in our methods, first is the disambiguation by category, and the second is Lesk disambiguation [17].

In semantic keyword processing, we calculate the visualness of each potential keyword extracted from the transcript. We propose three objective scoring methods to calculate visualness of keywords in an automatic fashion. Both first and second method use same visualness calculation. The difference is in the WSD process. The former utilizes category while the later uses Lesk algorithm to disambiguate different senses of a word. We call the first method as *Visualness with Disambiguation by Category (VDC)* and the second one as *Visualness with Lesk Disambiguation (VLD)*. The third method attempts to factor in the frequency of occurrence of a word in addition to its semantic score. Therefore, this third approach is a hybrid combination of visualness with Term Frequency - Inverse Document Frequency (TF-IDF) [13]. We call this method as *Lesk Visualness and Disambiguation with Frequency of Occurence (LVD-F)*. These objective scoring methods are explained in more detail in the remaining of this section.

2.1 Visualness with Disambiguation by Category (VDC)

This method uses visualness as explained previously. The seed words are taken from the categories and the title of the video. These seeds are processed through

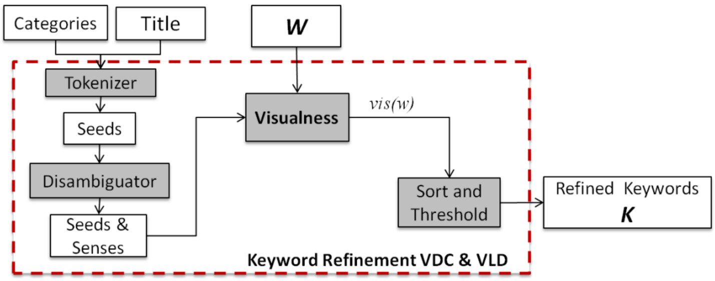


Fig. 2. Block diagram of the VDC and VLD scoring methods.

a ”disambiguator” block to perform the WSD process. In this VDC scoring method, the WSD is performed by utilizing the hierarchical representation of the subject category of a particular video. We use this category to determine the sense of every seed word. Semantic similarity is calculated between all possible senses of seed word s and the subject category of the corresponding lecture video. The sense i which has the highest similarity to the category is chosen as the appropriate sense. For example, a lecture video is about ”treating breast cancer” and it is listed under the ”medicine” subject category. Lets say the word ”cancer” is one of the seed words which has dual meaning; either cancer as disease or cancer as zodiac sign. Then by calculating the semantic similarity of these multiple senses of seed word ”cancer” to subject category ”medicine”, will yield high similarity score between medicine and sense of cancer as a disease.

The rest of the seed words are processed in similar fashion. These seed words with right senses along with the potential keyword list W from the transcript are then put through visualness calculations. Each word $w \in W$ is then processed to calculate its visualness value according to the seeds s with their corresponding senses as shown in Figure 2. At the end, we get the visual score of all the potential keywords. These scores are then sorted in descending order. By using rank cutoff (threshold) we get top ‘n’ ranked words from the list. This is the final refined keywords list K.

2.2 Visualness with Lesk Disambiguation (VLD)

This second method is the Visualness with Lesk Disambiguation. This method follows the same processes as VDC. The visualness is calculated in the same way as in VDC. The only significant difference here is the algorithm used to perform WSD. Instead of using subject category to disambiguate different senses of a word, we used the Lesk algorithm [17]. The Lesk algorithm attempts to correlate word senses with one another using the words contained in the words respective glosses. A ’gloss’ is a definition of a word. For each pair of words in this context, we calculate the number of overlapping words in each word sense gloss. This however requires a large number of pair comparisons.

As a less complicated alternative, we implemented the simplified Lesk Algorithm [15]. First we take into consideration the seed word s. The whole set of all seed words is taken as the context i.e. $context(s) = c_1, c_2, ..c_n$. The seed word s still has m multiple senses i.e. $s_1, s_2, ..s_m$. WordNet provides us with the gloss of each sense, $gloss(s_i)$ for every possible sense i of s. Therefore, the score is calculated by counting the number of overlapping words between the gloss of each sense s_i and the word's context. The assigned sense of s is the i^th where s_i maximizes the Lesk value given as:

$$Lesk(s_i) = |context(s) \cap gloss(s_i)|,$$

with || denoting the number of overlapping elements of $context(s)$ and $gloss(s_i)$.

The potential keyword list W is now processed in the same way as in the previous method, i.e. by calculating the visualness value for each word w according to the seeds s with their corresponding senses. The process is the same as shown in Figure 2, but with the "disambiguator" block using the Lesk Algorithm. The obtained visual scores are then sorted in descending order. We obtain the top 'n' ranked words by thresholding the sorted list. These top 'n' ranked words are the final refined set of keywords.

2.3 Lesk Visualness and Disambiguation with Frequency of Occurance (LVD-F)

The third proposed method is Lesk Visualness and Disambiguation with Frequency of Occurrence. LVD-F is built on top of VLD. The visualness calculation is carried out in the same way as for the other two methods. This method also uses the simplified Lesk algorithm for the WSD. In addition to visualness score $vis(w)$ of every word $w \in W$, we also calculate the frequency of occurrence of the word in the transcript document. We calculate this additional relative frequency of every word w from the initial text source, obtaining a frequency score $freq(w)$. Therefore for each word $w \in W$ we now have two different scores, as $vis(w)$ and $freq(w)$. We combine these two scores to obtain an overall score, according to:

$$LVD - F(w) = (\alpha)vis(w) + (1 - \alpha)freq(w),$$

where α is a weighting coefficient whose optimal value is determined empirically over the dataset in use. In our experiments we used the value of 0.5. This process of LVD-F is shown in Figure 3. Finally, the refined keyword list K is obtained in similar fashion as in the two previous methods, by selecting top 'n' ranked words from the sorted list.

3 Experimental Results

To compare the objective scores to those obtained manually, we conducted an on-line survey based experiment. The experiment was carried out using 5 lecture

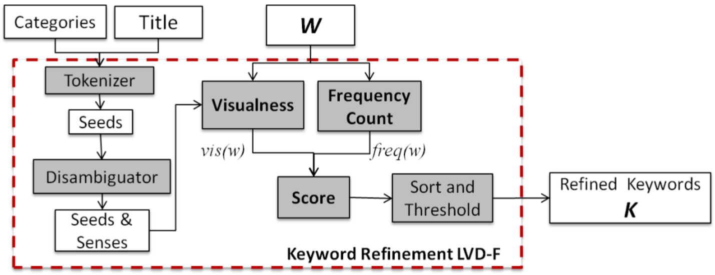


Fig. 3. Block diagram of the LVD-F scoring methods.

videos taken from the Khan Academy website [12]. The lecture videos consists of K-12 subjects including statistics, physics, linear algebra, astronomy, and biology – among many others. To reduce the size of output video, we segment the lecture video temporally into smaller segments based on visual content changes. These visual changes are detected as cuts in the video. We do this by computing a gray scale histogram of each frame, and taking the sum of absolute difference between consecutive frames. As a result of temporal segmentation, we obtained smaller segments of video consisting of 3-5 minutes in duration.

In our experiment we used lecture transcripts as the external text source, so the corresponding transcripts were also segmented accordingly based on timestamps. These transcripts were processed as described in section 2 to extract a list of potential keywords W. A subjective survey was then carried out by publishing on-line questionnaire to 15 participants. These participants were asked to view a video segment and select top 5 keywords from the given list W, in order of preference, that they would use if they were searching for the given segment. The questionnaires were handed out to expert viewers who have had experience using e-learning systems for search and retrieval purpose.

From subjective survey of each segment, we obtained the word ranking and its frequency count as shown in Table 1. ''Pos(n)" is the position of the selected keyword. It shows how many times a particular keyword is selected at n^{th} position, e.g. the word "Statistics" is chosen by 12 participants as their 1^{st} choice to describe the video. To come up with a single score for each selected word, we employed the Borda Count as:

$$Borda(w) = \sum_{i=1}^{n}((n-i) * freq_i(w)),$$

where $Borda(w)$ of a given word w is calculated by a total sum of the weights of the frequencies $freq_i(w)$, $freq_i(w)$ is the frequency of word w chosen at Position i, and n is the total number of possible positions, in our case $n = 5$.

A Borda count (BC) is an election method used to determine a winner from a voting where voters rank the candidates in order of preference [18]. The resulting scores from the Borda count are then sorted to obtain the top 'n' words, giving us

Word	Pos 1	Pos 2	Pos 3	Pos 4	Pos 5
Statistics	12	0	0	0	0
Numbers	1	0	0	0	0
Central	1	0	0	0	0
Average	0	2	6	2	0
Arithmetic	0	4	2	0	0
Computation	0	1	0	0	0
Mode	0	2	0	0	0
Mean	0	2	2	2	**2**
Tendency	0	0	2	0	0
Data	0	1	0	1	0
Example	0	2	0	1	0
Median	0	2	1	2	3

Table 1. Frequency recapitulation of responses.

Word	BC	Word	BC	Word	BC	Word	BC
Statistics	48	Mean	12	Mode	6	Numbers	4
Average	20	Median	10	Data	4	Central	4
Arithmetic	16	Example	7	Tendency	4	Computation	3

Table 2. Borda count of responses.

the refined list of the highest scoring words. For our experiment, we set $n = 10$. This gives us the top 10 words as shown in Table 2

We evaluated the performance of objective methods on two criteria. First being how well the objective methods score the top subjective words. For this, we consider the subjective keyword scores as ground truth. Thereby, taking 10 top keywords selected by viewers for each video segment as the subjective words. We then evaluate the score of these keywords obtained using the objective methods. This comparison is shown in Table 3 and Table 4 for two video segments under observation. The final score is in the range of 0 - 1. Where 0 indicate a word of low relevance and 1 a high relevance word to the video. Segment 1 is extracted from a lecture video about statistics and the segment 2 is from a physics lecture about vectors and scalars.

From these two tables, it can be seen that the objective methods give relatively good score to top subjective words. For 1^{st} video segment depicted in Table 3, VDC scored rather poorly except for top couple of words. The word ”median” even scores 0. This is most likely due to the disambiguation process of VDC which incorrectly assigns ”median” to the wrong sense, since VLD uses a different disambiguation method, it is therefore able to assign the correct sense to the word, hence obtaining a relatively high score as 1.

Comparatively, VLD scores the words slightly better. We can see that VLD gives a maximum score of 1 to a larger number of the words, and the other scores are still in the high range. The LVD-F scores are on a lower side except for top two subjective words. This probably is the result of the additional frequency score

Subjective Result	**Objective Scores**		
Top 10 Words	**VDC**	**VLD**	**LVD-F**
Statistics	1.00	1.00	0.85
Average	1.00	1.00	1.00
Arithmetic mean	0.26	0.95	0.29
Mean	0.55	1.00	0.64
Median	0.00	1.00	0.18
Example	0.11	0.91	0.63
Mode	0.06	0.05	0.29
Data	0.06	0.59	0.23
Tendency	0.07	0.62	0.20
Numbers	0.06	0.58	0.66

Table 3. Objective scores of top words in Segment 1.

Subjective Result	**Objective Scores**		
Top 10 Words	**VDC**	**VLD**	**LVD-F**
Vector	1.00	1.00	0.97
Scalar	1.00	1.00	1.00
Distance	0.39	0.76	0.34
Physics	1.00	1.00	0.76
Magnitude	0.37	0.77	0.36
Displacement	0.44	0.77	0.3
Direction	0.41	0.77	0.58
Definition	0.37	0.80	0.23
Move	0.499	0.87	0.28
Speed	0.42	0.77	0.26

Table 4. Objective scores of top words in Segment 2.

being added to the visualness score. Some words with high visualness values may not be used very often, resulting into a low frequency value, and ultimately a lower LVD-F score. Table 4 shows a similar trend for second video segment. VDC once again scores some words high, but the others relatively low. VLD performs well as it scores most of the words consistently high. LVD-F scores are once again influenced by the frequency.

The second evaluation criteria was to check if the top words scored by the objective methods are accurate. We once again consider the top 10 subjective results as the ground truth, and we compare these words with the top words obtained by the objective scores. We use the top 15 words as the refined keyword list K, and according to the subjective results we compute the precision and recall. The resulting precision and recall for the segments 1 and 2 are shown in Table 5 and Table 6. The correctly retrieved words are highlighted in the table.

We can see that the overall result of recall is better than the precision. Precision and recall values is dependent on the lecture video in question. In both videos shown here, the precision and recall of LVD-F is always better than the other two methods. This shows that the top words ranked by LVD-F is a rather

Top 10	**Objective Words**		
Subjective Words	**VDC**	**VLD**	**LVD-F**
Statistics	statistics	statistics	average
Average	average	average	statistics
Arithmetic	mean	mean	numbers
Mean	data	median	mean
Median	arithmetic	arithmetic	set
Example	time	harmonic	number
Mode	representative	mode	datum
Data	example	datum	mode
Tendency	sample	representative	arithmetic
Numbers	case	general	data
	particular	time	tendency
	thinking	add	central
	computation	example	plus
	sense	particular	middle
	inference	sample	median
Precision	0.4	0.46	0.6
Recall	0.6	0.7	0.9

Table 5. Precision and recall of Segment 1.

good estimation of the words that a user could choose. If we revisit the results of the scores in Table 3 and Table 4, we mentioned that the LVD-F scores are not very high in some cases. Even so, after the ranking and cutoff process, these words are still retrieved in the top 15. Therefore, LVD-F is able to achieve the best precision and recall score.

We further compared the scores of our developed scoring methods to the traditional TF-IDF method [13]. TF-IDF considers only the frequency of occurrence of a word and is widely used in various other information retrieval applications. We show the comparison for Segment 1 in Table 7. It can be seen that our developed methods outperform the TF-IDF which scores 0 for many words.

4 Conclusion

We develop keyword selection methods to automatically annotate lecture videos. These methods are designed for digital media with surrogate documents. We specifically used lecture videos with accompanying transcripts to test the validity of these methods. We used the title of video and the categories as ground truth for selecting potential keywords. The potential keywords were extracted from lecture transcript. We then score each potential keyword using a similarity measure and disambiguation criteria between different senses of words. We proposed three methods to do this objectively. The first and second method both employ visualness to describe the semantic value of the words. While the first method uses disambiguation by category, the second one uses Lesk algorithm dis-

Top 10	**Objective Words**		
Subjective Words	**VDC**	**VLD**	**LVD-F**
Vector	physics	physics	scalar
Scalar	introduction	introduction	vector
Distance	vector	vector	meter
Physics	scalar	scalar	question
Magnitude	start	quantity	physics
Displacement	quantity	amount	introduction
Direction	amount	idea	right
Definition	move	kind	direction
move	movement	like	second
Speed	travel	make	quantity
	right	right	know
	change	color	distance
	kind	version	magnitude
	idea	way	change
	pick	well	talking
Precision	0.26	0.2	0.4
Recall	0.4	0.3	0.6

Table 6. Precision and recall of Segment 2.

ambiguation for WSD. The third method is a hybrid combination of visualness with TF-IDF.

In order to evaluate the performance of the objectives methods, we performed subjective experiments. We then compared the results to those obtained by objective metrics. Keywords selected automatically by our proposed methods are very similar to the ones selected by viewers. The obtained results showed that the automatic solutions to semantic keyword selection are plausible. As a future work, we plan to incorporate speech and video analysis for extracting potential keywords.

References

1. Shih, T., ed.: Distributed multimedia databases: techniques & applications. IGI Publishing, Hershey, PA, USA (2002)
2. Rusmevichientong, P., Williamson, D.P.: An adaptive algorithm for selecting profitable keywords for search-based advertising services. In: In EC 06: Proceedings of the 7th ACM conference on Electronic commerce, ACM Press (2006) 260–269
3. Kiritchenko, S., Jiline, M.: Keyword optimization in sponsored search via feature selection. Journal of Machine Learning Research - Proceedings Track **4** (2008) 122–134
4. Azcarraga, A.P., Yap, T.N., Tan, J., Chua, T.: Evaluating keyword selection methods for websom text archives. IEEE Trans. on Knowl. and Data Eng. **16** (2004) 380–383
5. Wang, F., Lu, W., Liu, J., Shah, M., Xu, D.: Automatic video annotation with adaptive number of key words. In: ICPR, IEEE (2008) 1–4

Subjective Result	**Objective Scores**			
Top 10 Words	**TF-IDF**	**VDC**	**VLD**	**LVD-F**
Statistics	0.82	1.00	1.00	0.85
Average	0.73	1.00	1.00	1.00
Arithmetic mean	0.00	0.26	0.95	0.29
Mean	0.00	0.55	1.00	0.64
Median	0.64	0.00	1.00	0.18
Example	0.00	0.11	0.91	0.63
Mode	0.84	0.06	0.05	0.29
Data	1.00	0.06	0.59	0.23
Tendency	0.61	0.07	0.62	0.20
Numbers	0.77	0.06	0.58	0.66

Table 7. Comparison of scores to TF-IDF in Segment 1.

6. Yang, H., Siebert, M., Luhne, P., Sack, H., Meinel, C.: Automatic lecture video indexing using video OCR technology. In: Proceedings of the 2011 IEEE International Symposium on Multimedia. ISM '11, Washington, DC, USA, IEEE Computer Society (2011) 111–116
7. Qiu, Y., Guan, G., Zhiyong, W., Feng, D.: Improving news video annotation with semantic context. In: Proceedings of the 2010 International Conference on Digital Image Computing: Techniques and Applications. DICTA '10, Washington, DC, USA, IEEE Computer Society (2010) 214–219
8. Chakravarthy, A., Ciravegna, F., Lanfranchi, V.: Cross-media document annotation and enrichment. In: Proceedings of the 1st Semantic Authoring and Annotation Workshop (SAAW2006). (2006)
9. Altadmri, A., Ahmed, A.: Automatic semantic video annotation in wide domain videos based on similarity and commonsense knowledgebases. IEEE International Conference on Signal and Image Processing Applications (ICSIPA) (2009) 74 – 79
10. Kilinc, D., Alpkocak, A.: An expansion and reranking approach for annotation-based image retrieval from web. Expert Syst. Appl. **38** (2011) 13121–13127
11. Imran, A.: Interactive media learning object in distance and blended education. In: Proceedings of the 17th ACM international conference on Multimedia. MM '09, New York, NY, USA, ACM (2009) 1139–1140
12. The Khan Academy: Khan Academy (2012) http://www.khanacademy.org/.
13. Manning, C.D., Raghavan, P., Schutze, H.: Introduction to Information Retrieval. Cambridge University Press, Cambridge, UK (2008)
14. Li, H., Tian, Y., Ye, B., Cai, Q.: Comparison of current semantic similarity methods in wordnet. In: Computer Application and System Modeling (ICCASM), 2010 International Conference on. Volume 4. (2010) V4–408 –V4–411
15. Agirre, E., Edmonds, P.: Word Sense Disambiguation: Algorithms and Applications. 1st edn. Springer Publishing Company, Incorporated (2007)
16. Miller, G.A.: Wordnet: A lexical database for english. Commun. ACM **38** (1995) 39–41
17. Navigli, R.: Word sense disambiguation: A survey. ACM Comput. Surv. **41** (2009) 10:1–10:69
18. Young, P.: Optimal voting rules. The Journal of Economic Perspectives **9** (1995) 51–64