

Automated Tagging to Enable Fine-Grained Browsing of Lecture Videos

K.Vijaya Kumar
(09305081)

under the guidance of
Prof. Sridhar Iyer

June 28, 2011

Outline

- 1 Introduction
- 2 Motivation
- 3 Example Lecture Video Repositories
- 4 Problem Definition
- 5 Solution Approach
- 6 System Architecture
- 7 Implementation Details
- 8 Experiments and Evaluation Results
- 9 Conclusion and Future Work

Outline

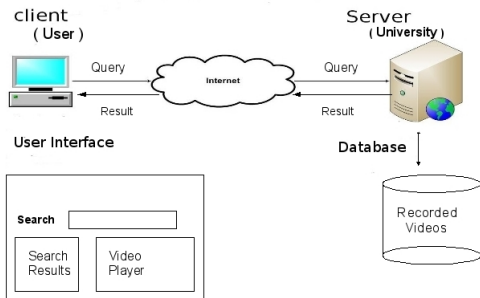
- 1 Introduction
- 2 Motivation
- 3 Example Lecture Video Repositories
- 4 Problem Definition
- 5 Solution Approach
- 6 System Architecture
- 7 Implementation Details
- 8 Experiments and Evaluation Results
- 9 Conclusion and Future Work

Introduction

- Lecture video recordings are widely used in distance learning
- To make best use of the available videos a system called **Browsing System** is required
- Purpose of the browsing system is to provide search facility in the lecture video repository
- **Problem Statement :**
To develop a browsing system which is useful for users to find their required video content easily

Video Browsing System

- It takes keywords from users and gives them lecture videos matching their keywords



Outline

- 1 Introduction
- 2 Motivation**
- 3 Example Lecture Video Repositories
- 4 Problem Definition
- 5 Solution Approach
- 6 System Architecture
- 7 Implementation Details
- 8 Experiments and Evaluation Results
- 9 Conclusion and Future Work

Can we do the same in Lecture Videos ?

- Yes, We can provide the same type of search facility in lecture videos based on their contents

Example Scenarios

- Portion of video where *Matrix Multiplication* is discussed in a programming course lecture
- Searching for a video which discusses *Quick Sort* in a Data Structures course videos
- Finding video results containing *Double Hashing* in lecture video repository

Techniques for Searching in Lecture Videos

- **Meta data based :**
Uses data such as video title, description or comments associated with the video
- **Content based :**
Based on data extracted from lecture videos, which represents contents present within it

How You Tube Searches Videos?

The screenshot shows a YouTube video player for a video titled "TCP/IP 3 way handshake" by the channel "liamsomerville". The video content is a diagram illustrating the 3-Way Handshake process between two hosts, A and B. Host A sends a SYN packet (seq=x) to Host B. Host B receives the SYN packet and responds with a SYN-ACK packet (seq=y, ack=x+1). The diagram uses blue arrows for the SYN packet and a green arrow for the SYN-ACK packet. The video player interface includes a search bar, navigation buttons (Search, Browse, Upload), and a suggestions sidebar on the right. The video has 73,579 views and was uploaded on April 09, 2007.

- Youtube video search is based on meta data associated with videos
- Meta data include video title, description and tags

Outline

- 1 Introduction
- 2 Motivation
- 3 Example Lecture Video Repositories**
- 4 Problem Definition
- 5 Solution Approach
- 6 System Architecture
- 7 Implementation Details
- 8 Experiments and Evaluation Results
- 9 Conclusion and Future Work

Example Lecture Video Repositories

- CDEEP[5] : No search feature
- NPTEL[16] : No search feature
- freelecturevideos.com[8]
- videolectures.net[20]
- Lecture Browser, MIT[13]

Some more

- Academic Earth[1]
- Youtube Edu[23]

Link to list of available educational video repositories is at[15]

Slide Index feature in NPTEL

- Recently launched
- Through a video processing company called videopulp [21]

The screenshot displays the NPTEL website interface for an NPTEL course. The header includes the NPTEL logo and the text "E-LEARNING COURSES FROM THE IITS & IISc". The breadcrumb trail reads "NPTEL >> Courses >> Computer Science and Engineering >> Computer Graphics (Video) >> 1 - Introduction".

Below the breadcrumb trail are three buttons: "Video", "Watch on YouTube", and "Download". The main content area features a video player showing a slide titled "Conceptual framework for interactive graphics". The slide content includes a flowchart with three boxes: "Application model", "Application program", and "Graphics system", connected by arrows. To the right of the "Graphics system" box is an image of a computer monitor, and below it is an image of a keyboard and mouse.

On the right side of the video player is a "Browse Video by Topic" sidebar. It contains a list of video topics, with "Introduction to COMPUTER GRAPHICS" repeated multiple times. Below the list, there are sections for "Typical applications areas are" (with expandable options for "GUI - Graphical User Interface", "Typical applications areas are", and "Entertainment") and "Various application packages and standards are available:" (with expandable options for "GKS - Graphics Kernel System" and "GKS - Graphics Kernel System"). At the bottom of the sidebar, it says "Powered by videopulp".

At the bottom of the video player, there is a "Contents in this course" section with a link to "Introduction (54:08)".

freevideolectures.com

- Provides Google custom search to index textual data
- Topic Looked for : **Double Hashing**

The screenshot shows the homepage of FreeVideoLectures.com. At the top left is the logo with the tagline "Bringing Free Education to All". To the right is a Google Custom Search box with a "Search" button. Further right are links for "Blog", "Signup", and "Login". Below the search bar are links for "Home" and "Courses list". A main banner features the text "740+ Online Courses, 18000+ Videos from Top 20+ Universities on 35+ Categories" and "Download in MP4, FLV, 3GP, MP3 and Torrents". Below this is a promotional banner for "Study Places" with the text "Want to Study Abroad? Top Universities Calling... Meet Expert Counsellors in your city" and a "FREE Counselling Register NOW" button. A "Subjects" section lists various fields of study such as Anatomy, Physiology, Astronomy, Aerospace, Bio Sciences, Bio Technology, Biology, Business Management, Calculus, Chemistry, Civil Engineering, Communication Skills, Computer Science, Data Structures, Economics, Electrical Engineering, Electronics, Entrepreneurship, Genetics, Health Sciences, History, IC Design, Languages, Law, Literature, Mathematics, Mechanical, Medicine, Networking, Other Courses, Philosophy, Physics, Programming, Psychology, Signals Systems, Social Sciences, VLSI and ASIC Design, and Web Designing. A red "NEW" badge indicates "50 Online Courses Conducted during Spring/Winter 2010". A "Universities" section is partially visible at the bottom.

freevideolectures.com

- Keyword : **double hashing**
- Result : Your search - double hashing - **did not match any documents.**



Your search - **double hashing** - did not match any documents.

Suggestions:

- Make sure all words are spelled correctly.
- Try different keywords.
- Try more general keywords.
- Try fewer keywords.

Ads by Google

[Video Encryption](#)

www.cobham.com/gms

Cobham Surveillance, GMS Products Encrypted wireless video links

[Video Encryption](#)

www.aftindia.in

Anti-piracy/copy protection solution for video, audio, software, data.

[USB encryption](#)

raakhs.blogspot.com

and removable media control req. tools and lessons learned

freevideolectures.com

- Keyword : hashing
- Result : 6 video results



Ads by Google

[Video Encryption](#)

[www.cobham.com/gms](#) Cobham Surveillance, GMS Products Encrypted wireless video links

[Video Encryption](#)

[www.allmedia.in](#) Antipiracy/copy protection solution for video, audio, software, data.

[USB encryption](#)

[raikhs.blogspot.com](#) and removable media control req. tools and lessons learned

Results 1 - 6 for hashing with Safesearch on. (0.14 seconds)

[Hashing Video Lecture IIT Delhi NPTEL Course ,Videos, Video ...](#)

Hashing NPTEL Video Online Course, Classes, video Tutorials, IIT Delhi Video Lectures, Naveen Garg Audio Lectures, free Online Courses, Online Classes, ...
[freevideolectures.com/Course/2279/Data-Structures-And.../5](#)

[Hashing, Hash Functions | MIT OCW Online videos, Erik Demaine ...](#)

Hashing, Hash Functions Online Videos, Classes, Tutorials, MIT OCW Video Lectures, Erik Demaine Audio Lectures, free Online Courses, Online Classes, ...
[freevideolectures.com/Course/1941/Introduction-to-Algorithms/7](#)

[Universal Hashing, Perfect Hashing | MIT OCW Online videos, Erik ...](#)

Universal Hashing, Perfect Hashing Online Videos, Classes, Tutorials, MIT OCW Video Lectures, Erik Demaine Audio Lectures, free Online Courses, ...
[freevideolectures.com/Course/1941/Introduction-to-Algorithms/8](#)

[Introduction to Algorithms Video Lectures by Prof. Erik Demaine ...](#)

Topics covered include: sorting; search trees; heaps; and hashing; divide-and-conquer; dynamic programming; amortized analysis; graph algorithms; ...
[freevideolectures.com/Course/1941/Introduction-to-Algorithms](#)

Google
Custom Search

freevideolectures.com

First video

- Duration - 61:22
- Found at - 42:32

The screenshot shows a YouTube video player interface. The video title is "Data Structures And Algorithms" and the channel is "IIT Delhi NPTEL Course, Prof. Naveen Garg". The video is titled "Lecture 5: Hashing". The video content is a slide titled "Double Hashing" with the following text:

Uses two hash functions, h_1 , h_2

- $h_1(k)$ is the position in the table where we first check for key k
- $h_2(k)$ determines the offset we use when searching for k
- In linear probing $h_2(k)$ is always 1.

DoubleHashingInsert(k)
if (table is full) error
probe = $h_1(k)$; offset = $h_2(k)$
while (table[probe] occupied)
 probe = (probe+offset) mod m
table[probe] = k

The video player shows a progress bar at 42:11 / 61:22. The video has 1332 views and 0 likes. The video is in HD 1080p quality.

videolectures.net

- Provides free online access to lecture video recordings of various universities
- Has hyper links to slide change timings

Lecture popularity: ★★★★★ You need to login to cast your vote.

We are currently conducting a short survey. We value your feedback, and would appreciate if you took a few moments to respond to some questions. [Click here to take the survey.](#)

Description:

"Today starts a two-lecture sequence on the topic of hashing, which is a really great technique that shows up in a lot of places. So we're going to introduce it through a problem that comes up often in compilers called the symbol table problem. And the idea is that we have a table S holding n records where each record, just to be a little more general here, is made of several fields and a pointer to the next record in the table."

Probing strategies

Double hashing

Given two ordinary hash functions $h_1(k)$ double hashing uses the hash function

$$h(k, i) = (h_1(k) + i \cdot h_2(k)) \bmod m.$$

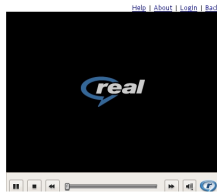
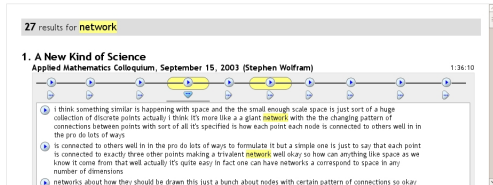
This method generally produces excellent results but $h_1(k)$ must be relatively prime to m , $h_2(k)$ is to make m a power of 2 and design h_2 produce only odd numbers.

October 3, 2005 Copyright © 2005-7 by John D. DeRose and Nathan F. Liu

Slides	Relevant topic
30:55	Division method (1)
37:17	Division method (2)
38:55	Multiplication method
42:13	Multiplication method example
50:01	Resolving collisions by open addressing
55:31	Example of open addressing (1)
56:33	Example of open addressing (2)
56:48	Example of open addressing (3)
58:01	Example of open addressing (4)
58:54	Probing strategies (1)
61:21	Probing strategies (2)
64:19	Analysis of open addressing - Theorem

Lecture Browser

- Provides free on line access to lecture videos available in MIT Open Course ware
- Has Content based Search feature and highlights relevant segments of each video



Our System User Interface

CS101 Lecture:09 Character Arrays and Pointers

Slide Index

- overview (00:00:24)
- arrays (00:01:22)
- arrays (00:03:35)
- arrays to store marks info for (00:05:42)
- accessing array elements (00:06:31)
- accessing totalmarks (00:07:56)
- reading roll and various marks (00:09:18)
- accessing array elements (00:11:11)
- **matrix multiplication** (00:13:37)
- **matrix multiplication** (00:16:15)
- **matrix multiplication** (00:18:18)

localhost:8000/Browser/GetVideoServlet?id=CS101A2009L964=matrix multiplication#

Matrix Multiplication

A matrix, after all, is a 2-dimensional array.
Given

$m \times n$ matrix A,
 $n \times p$ matrix B,
AB is a $m \times p$ matrix.

If $C = AB$, then

$$C[i][j] = \sum_k A[i][k] * B[k][j]$$

[Lectures List](#)

Transcript

going back look at the next student total marks but that's and that is valuable skilled statement is nothing going to be added this is a stack architecture exists on elements in this particular case we have x is ten first in the log one one in the middle legal another one follow what i queries in it and then it out i couldn't that column gordon also if i wanted but invalid intentional printing attention to **matrix** we call or and the difference because of that number example you all agree that such a die system be multiplied with so consequently if i then as a n by n **matrix** a another **matrix** b which is n by p **matrix multiplication** of a might be **matrix** of entity this this simplest out a most specific every that every element of c is defined by the summation as indicated in the form binary with this that seek i j to system of constant eventually which it so for a given value of i and j which values given that it does some **matrix** the minute i mention is n so given that if someone to n so then i sum up all the **matrix** element on indefinitely that we will one that middle how do i element in a c plus plus program second element and element **matrix** if for example these n by an n then the number of rows and columns in that **matrix** that indexed by zero to n

Features in Lecture Video Repositories

Repository	Search	Navigation Features
CDEEP	No	No
NPTEL	No	No
freelecturevideos.com	Meta data	No
videlectures.net	Meta data	Slide Index (Manual)
Lecture Browser, MIT	Content	Speech Transcript
Our System	Content	Speech Transcript Slide Index (Automated)

Table: Lecture Video Repositories Comparison

Problems with existing systems

freevidelectures.com

- No indication of where exactly searched keywords occur within the video
- Takes more time to find required information

videlectuers.net

- Uses manual process for Synchronization of the slides

Why can't we use lecture browser?

- Can not be applied directly to our lecture videos.
- Requires speech recognition engine adaptation for non native english speakers
- Not an open source tool
- Their speech recognition engine is also not publicly available

How our system is different

- Provides automatic synchronization of slides.
- Improved user interface with more navigation features.
It combines features in videolectures.net and lecture browser
- Open source application by integrating available speech recognition and text search engines
- Tune Sphinx speech recognition engine to recognize and transcribe Indian accents (English)

Outline

- 1 Introduction
- 2 Motivation
- 3 Example Lecture Video Repositories
- 4 Problem Definition**
- 5 Solution Approach
- 6 System Architecture
- 7 Implementation Details
- 8 Experiments and Evaluation Results
- 9 Conclusion and Future Work

- Input: **keywords**

- Output :

List of videos matching the keywords

In each video portions where the keywords occur in the speech are highlighted

When user clicks on a particular portion video starts playing in the media player

Along with the media player user interface also shows slide index and speech transcript

- Scope of the project : Only deals with lecture videos which are in **English** and related **Computer Science** domain.
Reason : Speech Recognition Engine

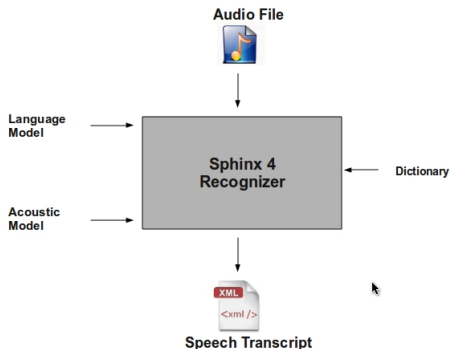
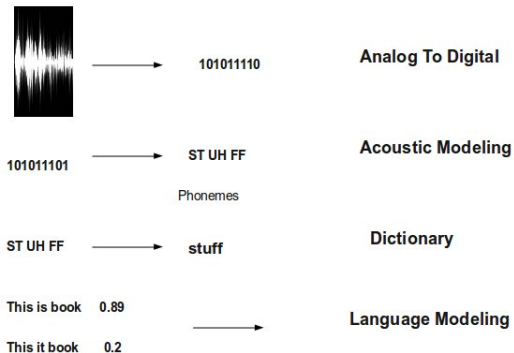


Figure: Sphinx 4 Recognizer

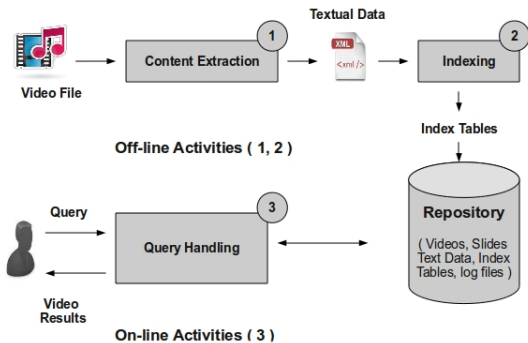
Steps in Speech Recognition



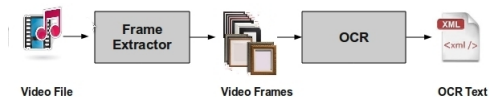
Outline

- 1 Introduction
- 2 Motivation
- 3 Example Lecture Video Repositories
- 4 Problem Definition
- 5 Solution Approach**
- 6 System Architecture
- 7 Implementation Details
- 8 Experiments and Evaluation Results
- 9 Conclusion and Future Work

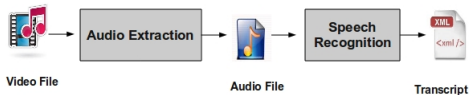
Solution Approach



Content Extraction



(a) Optical Character Recognition



(b) Speech Recognition

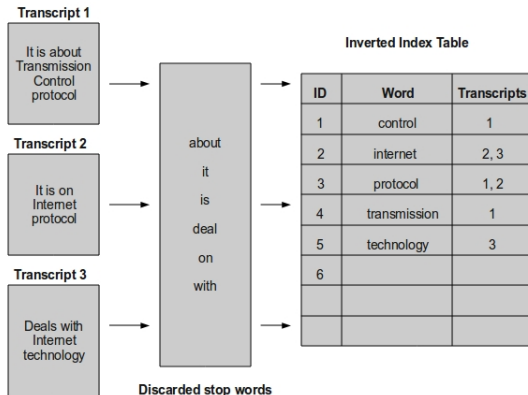
Speech Recognition Engines

- Sphinx 4 [18]
- Hmm Tool Kit (HTK) [9]

Reasons for choosing Sphinx

- Provides Java API(Application Programmable Interface)s, so it can be integrated easily into any application
- CMU Sphinx provides support for various tools useful in speech recognition
- Has easy configuration management where we need to set various parameters related to speech recognition
- Supporting tools are available for generation of acoustic and language models
- Completely written in java, it is highly modular and platform independent

Indexing & Query Handling



Text Search Engines

- Lucene[3], Indri[10]
- Xapian[22], Zettair[24]

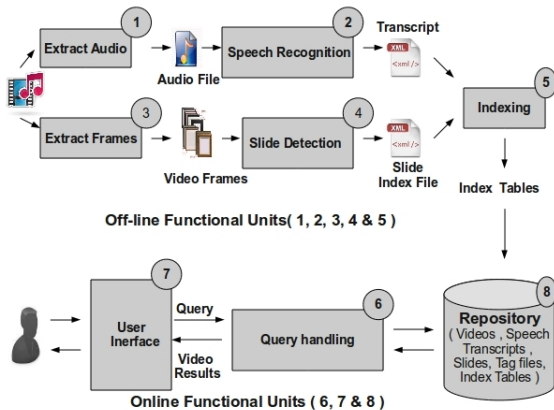
Reasons for choosing Lucene

- It creates index of smaller size and search time is also very less[17]
- Supports ranked searching : best results returned first
- Can handle many powerful query types: phrase queries, wild card queries, range queries and more
- Mostly used text search engine. List of more than 150 applications and websites that are using Lucene to provide search facility[14]

Outline

- 1 Introduction
- 2 Motivation
- 3 Example Lecture Video Repositories
- 4 Problem Definition
- 5 Solution Approach
- 6 System Architecture**
- 7 Implementation Details
- 8 Experiments and Evaluation Results
- 9 Conclusion and Future Work

System Components



Outline

- 1 Introduction
- 2 Motivation
- 3 Example Lecture Video Repositories
- 4 Problem Definition
- 5 Solution Approach
- 6 System Architecture
- 7 Implementation Details**
- 8 Experiments and Evaluation Results
- 9 Conclusion and Future Work

Audio Extraction

- Input : Video file
- Output : Audio file
- Command line tools provided by FFmpeg [7]
- Running *ffmpeg* :
\$ *ffmpeg -i CS101_L10_Strings.mp4 -ar 16000 -ac*
1 *CS101_L10_Strings.wav*

Speech Recognition

- Input : Audio file
- Output : Time aligned transcript in XML format
- Open source Java library for Sphinx-4 Speech Recognizer from CMU Sphinx [18]
- Requires language model, acoustic model and a pronunciation dictionary

Language model creation

- Large amount of text corpus related to the domain of speech recognition is required
- CMU SLM Toolkit [6] is useful for creating language model from the text corpus

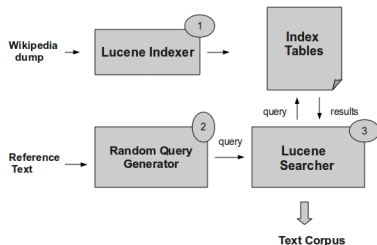


Figure: Framework for creating large amount of text corpus

Language model creation

- Collected text corpus related to Computer Science domain
- Wiki Index : Randomly generated queries consisting of terms from CS and searched in Lucene Indexes
- Text books : Data structures, Algorithms, Computer Networks, DBMS and OS
- Manual Transcriptions : Available in MIT OCW [4]
- Converted PDF files to Text using Java library provided from PDFBox [11]

Acoustic model development

- Requires audio files and corresponding manual transcriptions
- Developing new acoustic modeling takes large amount of time
- Adaptation of acoustic model is an option which requires an existing model
- CMU Sphinx provides WSJ and HUB4 models useful for recognizing US English
- Sphinx Train and Sphinx Base are set of tools useful for development for acoustic model

Acoustic model development

- We have to adapt an acoustic model to match our speakers to get better recognition accuracy
- Time consuming, which requires small audio files each having a sentence and manual transcription of each of the audio file
- Created 150 wav files for adaptation from CS101 lectures of Prof. Deepak Phatak
- Each of the wav file duration is 2 to 5 seconds and gave manual transcriptions for them

Speech Transcript Generation

- Configured the Sphinx-4 recognizer with the created language model and acoustic model
- Transcribed audio files of CS101 lectures and generated time aligned transcripts
- Transcribing of an audio file took approximately double the duration of the file
- The transcription speed can be increased but gives low recognition accuracy

Example Speech Transcript

```
<transcript>
  <tt>
    <text> deals with </text>
    <time> 7 </time>
  </tt>
  <tt>
    <text> searching </text>
    <time> 11 </time>
  </tt>
  <tt>
    <text> of lectures </text>
    <time> 14 </time>
  </tt>
</transcript>
```

Video Frames Extraction

- Input : Video file
- Output : Frames extracted from the video at specified intervals
- *ffmpeg* can be used for the frame extraction
- ```
$ ffmpeg -i CS101_L10_Strings.mp4 -r 1 -f image2 image_%4d.jpeg
```

# Slide Detection

- Input : Video frames of a lecture
- Output : Slides of the lectures along with their title and time of occurrences
- Designed an algorithm based on slide title matching which uses OCR for slide text extraction
- Found an OCR tool called tesseract-ocr [19] which gives better recognition accuracy among available the Open Source tools

# Example frame from a video lecture

## Overview

- Engineering Education
- Research and Critical thinking
- Introduction to the course
- Quiz



## After applying OCR

### Overview

Engineering Education

He\$earchar1&iUrilmu| lthinkirng

lnirucluctivn tc the course

Oui;

# Title Matching algorithm for Slide Detection

**Title    Time**

---

**overview 0104**    → Will be identified as starting of a slide

**overview 0105**

**overview 0106**

overview 0107

overview 0108

overview 0109

overview 0110

**engineering 0135**    → Will be identified as starting of next slide

**engineering 0136**

**engineering 0137**

engineering 0138

engineering 0139

engineering 0140

## Title Matching algorithm for Slide detection

```
while i < titles.length-1
begin
 if !titles[i].equals(prev) && matchesNextTwo(titles,i)
 indices.add(i);
 i = findNextSlide(titles,title[i],i+3)
 if i == -1
 return;
 endif
 prev = titles[i];
 indices.add(i);
 i = i + 2;
endif
i = i + 1;
end
```

## Example Slide Index

```
<slides>
 <slide>
 <title> Overview </title>
 <time> 13 </time>
 </slide>
 <slide>
 <title> Introduction </title>
 <time> 79 </time>
 </slide>
</slides>
```

# Indexing

- Input : Transcript file and Slide index file
- Output : Creates an Index or adds to existing indexes
- Apache Lucene [3] provides Java library for indexing text documents
- Parsed the transcript and slide index file which are in XML format
- Indexed CS101 lectures of Autumn 2009 and created indexes are of size 2.5MB

# Query Handling

- Input : User given queries
- Output : List of lectures matching the query
- Apache Lucene [3] is also include Java classes for searching the indexes
- Technologies : Java Server Pages (JSPs) and Java Servlets
- Web Server : Apache Tomcat/6.0.24 [2]
- Operating System : Ubuntu Lucid Lynx 10.04 LTS

# User Interface

- Created web pages using HTML and Java Script
- Using a freely available version of JW Player [12] for playing videos in the interface

**Browsing System for Lecture Videos**

Search CS 101 lectures of Autumn 2009

[CS101 Lecture:01 Introduction](#)

Slides : overview, engineering, engineering education, critical thinking, journey to an exciting career, attitude issues, the best attitude, course organization, quiz 1, quiz 2, quiz 3, quiz 4, quiz 3, quiz 4.

---

[CS101 Lecture:02 Introduction to Computers](#)

Slides : overview, buddhu ram computer mr dumbu, dumbu with his assistant's, capabilities of mr dumbu, capabilities of ivr dumbu, dumbu has very poor memory, dumbu's memory device, dumbu's memory device, dumbu can store values in his memory, some peculiarities of memory, some peculiarities of memory, exchanging values with dumbu, exchanging values with dumbu , dumbu computing, dumbu computing, dumbu computing, dumbu computing, dumbu computing, junior dumbu computing, junior dumbu computing, junior dumbu computing, result of dumbu computing, how do we give instructions, cin, how do we give instructions, where does dumbu keep the program , complete program, convert temperature from to f, complete program, convert temperature from to f.

---

[CS101 Lecture:03 More on Computing](#)

Slides : complete program, convert temperature from dc to of, convert temperature from dc to of, convert temperature from dc to of, quiz 1, quiz 2, quiz 3, more on computing, decision making, meet another dumbu, lab sessions , lab sessssssions .

Figure: User Interface of our System

# User Interface


**Browsing System for Lecture Videos**

Search CS 101 lectures of Autumn 2009

[Lectures List](#)

Found 9 video(s) matching "binary search" in 143 milliseconds

[CS101 Lecture:12 Searching, Array Manipulation](#)




Transcript: this approach and the **binary search** dictionary often given element effectively | section... an exam I come out and analysis figure we call this a linear **search** estimated... so imagine that when given sorted and having a linear **search** automatic compact first

Slides : g, similarity with e , similarity with e g, similarity with e , **binary search**, gimmu behaviour of **binary search** ,

.....

[CS101 Lecture:11 Time Complexity and Searching](#)



Transcript: this is not added then this other program width that this value is used algorithm for **binary**... this side finds **binary** by twenty one whenever we acknowledging unsigned byte in the one... instrumentation that identity sop effect in that it is either side of the **search** right

Slides : , fi ndrcil version 1, fmdrcil version 2, behaviour of **binary search** 1,

.....

[Character Arrays, Course Projects](#)




Figure: Search Results for query **binary search**



# User Interface

CS101 Lecture:12 Searching, Array Manipulation

**Slide Index**

- root (00:06:09)
- bisection method for finding root (00:06:44)
- bisection method for finding root (00:06:54)
- bisection method for finding root (00:06:57)
- bisection method for finding root (00:07:36)
- bisection method for finding root (00:07:44)
- similarity with e (00:07:52)
- similarity with e gi (00:07:57)
- similarity with e (00:13:04)
- similarity with e gi (00:13:09)
- similarity with e (00:13:45)
- **binary search** (00:14:04)
- minimum behaviour of binary search (00:14:04)

localhost:8000/BrowserGetVideoServlet?id=CS101A2009L12.6s=binary search#

**Binary search**

```
int lo = 0, hi = n - 1,
mid = (lo + hi) / 2, foundFlag = 0;
While(roll[mid] < givenroll && hi > lo){
//recalculate mid
if (roll[mid] > givenroll){
// roll is towards upper half of the array
hi = mid;
}
else {lo = mid;
mid = (lo + hi) / 2;
}
}
if (roll[mid] == givenroll) foundFlag = 1;
```

that I have to understand my

Lectures List

**Transcript**

are the x z value typically added add a a the yesterday excitation off seventeen evaluate yesterday we have seen observed when given a number numbers intend to start with the first element compare the first element of the identity given a number that does not match the next element next element x element and so on then the vienna and if we find that the first of all the middle number the situation intensity a all values in the having an exam i come out and analysis figure we call this a linear **search** estimated that on another although the algorithm is called that then actually make a comparisons that this as afternoon mine the given number microsoft then is our i didn't impact in that so without so imagine that when given sorted and having a linear **search** automatic compact first is to determine the byte that entity that by program is whenever element in an added and i have mission jason a but then the what's that such information entity in the common processes that have committed these that if the values another ascending order you members to produce anyone after that but use that to some entity and that a descendant that that typical example then given as that of a dictionaries students that event in the dictionary simon with

Figure: playing selected video with the navigation

# Content Repository

- Recorded videos of lectures
- Speech transcripts
- Slide Index files
- Lucene indices

# Outline

- 1 Introduction
- 2 Motivation
- 3 Example Lecture Video Repositories
- 4 Problem Definition
- 5 Solution Approach
- 6 System Architecture
- 7 Implementation Details
- 8 Experiments and Evaluation Results**
- 9 Conclusion and Future Work

## Slide Detection Results

Video	Actual slides	Detected slides	Correctly detected	Duplicates	Recall (%)	Prec. (%)
L_01	14	14	12	0	100	85
L_02	20	20	16	6	100	80
L_03	12	11	11	2	91.6	100
L_04	32	30	26	9	93.7	86.6
L_05	32	30	28	5	93.6	93.3
Total	110	105	93	18	95.4	88.5

Table: Slide Detection results

# Speech Recognition Results

Adaptation files	Words in test files	Matches	Accuracy(%)
0	127	22	13
30	119	43	31
60	124	70	52
90	120	76	59
120	110	69	61
150	123	82	62

Table: Speech Recognition results

# Video Retrieval Results

No.of queries tested	30
Avg Search seconds	0.004
Recall	0.72
Avg Precision	0.91

Table: Search Quality Results

# Outline

- 1 Introduction
- 2 Motivation
- 3 Example Lecture Video Repositories
- 4 Problem Definition
- 5 Solution Approach
- 6 System Architecture
- 7 Implementation Details
- 8 Experiments and Evaluation Results
- 9 Conclusion and Future Work**

## Conclusion and Future Work

- Built a system for providing search facility in CS101 Autumn 2009 lectures
- Speech recognition accuracy can be improved through more adaptation
- Slide Detection method can be improved to reduce duplicate slides
- More lectures can be added to the repository





*Academic Earth.*

<http://academicearth.org/>.



*Apache : An Open Source Web Server.*

<http://tomcat.apache.org/>.



*Apache Lucene.*

<http://lucene.apache.org/java/docs/index.html>.



*Audio/Video Lectures from MIT OCW.*

[http://ocw.mit.edu/courses/audio-video-courses/  
#electrical-engineering-and-computer-science](http://ocw.mit.edu/courses/audio-video-courses/#electrical-engineering-and-computer-science).



*CDEEP , IIT Bombay.*

<http://www.cdeep.iitb.ac.in/>.



*CMU Statistical Language Modeling Toolkit Documentation.*

[http://www.speech.cs.cmu.edu/SLM/toolkit\\_  
documentation.html/](http://www.speech.cs.cmu.edu/SLM/toolkit_documentation.html/).



*FFmpeg.*

<http://www.ffmpeg.org/>.



*freevideolectures.com.*

<http://www.freevideolectures.com/>.



*HTK.*

<http://htk.eng.cam.ac.uk/>.



*Indri.*

<http://www.lemurproject.org/indri/>.



*Java PDF Library.*

<http://pdfbox.apache.org/>.



*JW Player.*

<http://www.longtailvideo.com/players/jw-flv-player/>.



*Lecture Browser , MIT.*

<http://web.sls.csail.mit.edu/lectures/>.



*List of Applications that are using Lucene.*

<http://wiki.apache.org/lucene-java/PoweredBy>.



*List of educational video websites.*

[http://en.wikipedia.org/wiki/List\\_of\\_educational\\_video\\_websites](http://en.wikipedia.org/wiki/List_of_educational_video_websites).



*nptel.*

<http://www.nptel.iitm.ac.in/>.



*Open Source Text Search Engines Evaluation Results.*

<http://wrg.upf.edu/WRG/dctos/Middleton-Baeza.pdf>.



*sphinx.*

<http://www.speech.cs.cmu.edu/>.



*tesseract-ocr.*

<http://code.google.com/p/tesseract-ocr/>.



*videlectures.net.*

<http://www.videlectures.net/>.



*VideoPulp: Official Partners for Slide Index feature in NPTEL.*

<http://www.videopulp.in/>.



*xapian.*

<http://xapian.org/>.



*Youtube Edu.*

<http://www.youtube.com/education?b=400>.



*zettair.*

<http://www.seg.rmit.edu.au/zettair/>.

Thank You

