Senior Project –Computer Science – 2014

STRUCTURING THE UNSTRUCTURED NOTE

AUTOMATIC ORGANIZING AND FORMATTING FOR LECTURE NOTES

Shiqing He **Advisor: Prof. Kristina Striegnitz**

Note taking is an important practice during lectures. With the rapid development of electronic devices such as smart phones, tablets and laptops, digital note-taking has become an option for many students. In the classroom setting, note takers might fail to record clearly organized notes due to limitations of time and devices. Therefore, users often need to manually organize long paragraphs of notes by dividing them into appropriate sections based on content, or structuring them into lists and tables. The goal of this research is to develop a system that automatically structures unorganized notes. Using topic modeling and automatic summarization, we detect and analyze the "structure" of the notes. We then build a layout based on the structure and format notes into a more readable format.

Raw Text

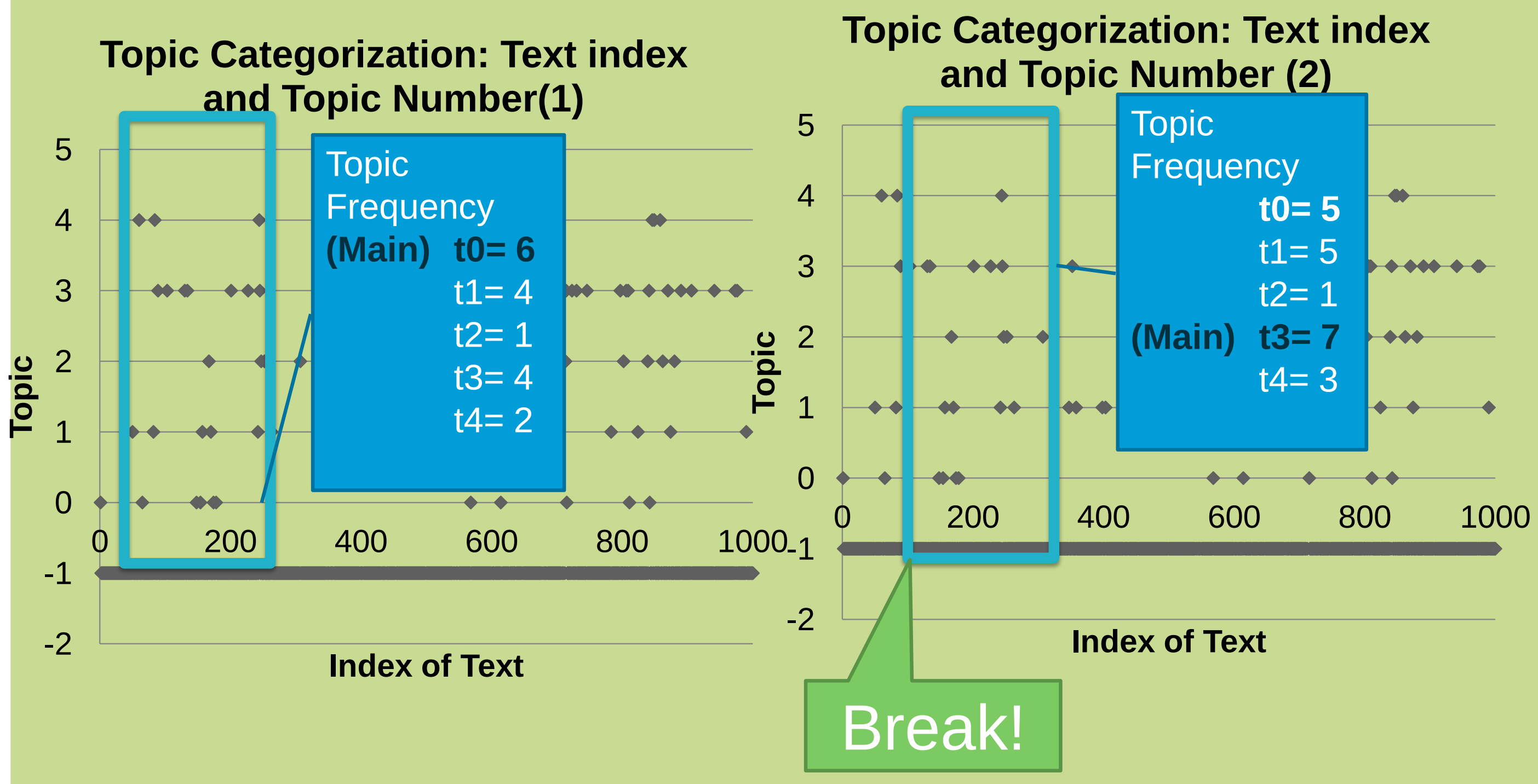
- Raw text =unstructured texts that need to be organized.
- Users can type or record lectures to obtain raw texts.
- For this research, we use online course subtitles from *Coursera* as raw texts.[1]

Detect Topics Within the Text

- Topic Modeling =a suite of algorithms that discovers thematic topics within texts.[2]
- We use *Mallet,* a Java-based statistical natural language processing package to generate topic models for raw text inputs. [3]
  - For example, when we run Mallet on an hour- long lecture subtitle, we detect five topics, each shown as a word cloud:
    - 0 processing android device ve run ll running javascript java ios thing simple devices sketch screen ip address mode work
    - 1 width ve speed make symmetry line screen ll idea brush point green map brushes application divided distance mapping simple
    - 2 sound time ll things desktop week start app bit yeah people lot show sounds music good ve kind make
    - 3 mouse color ve draw position drawing line colors red numbers program rectangle screen green blue code basic point lines
    - 4 sound audio maxim play player ve sketch file ll code folder data environment create beat device files store speed

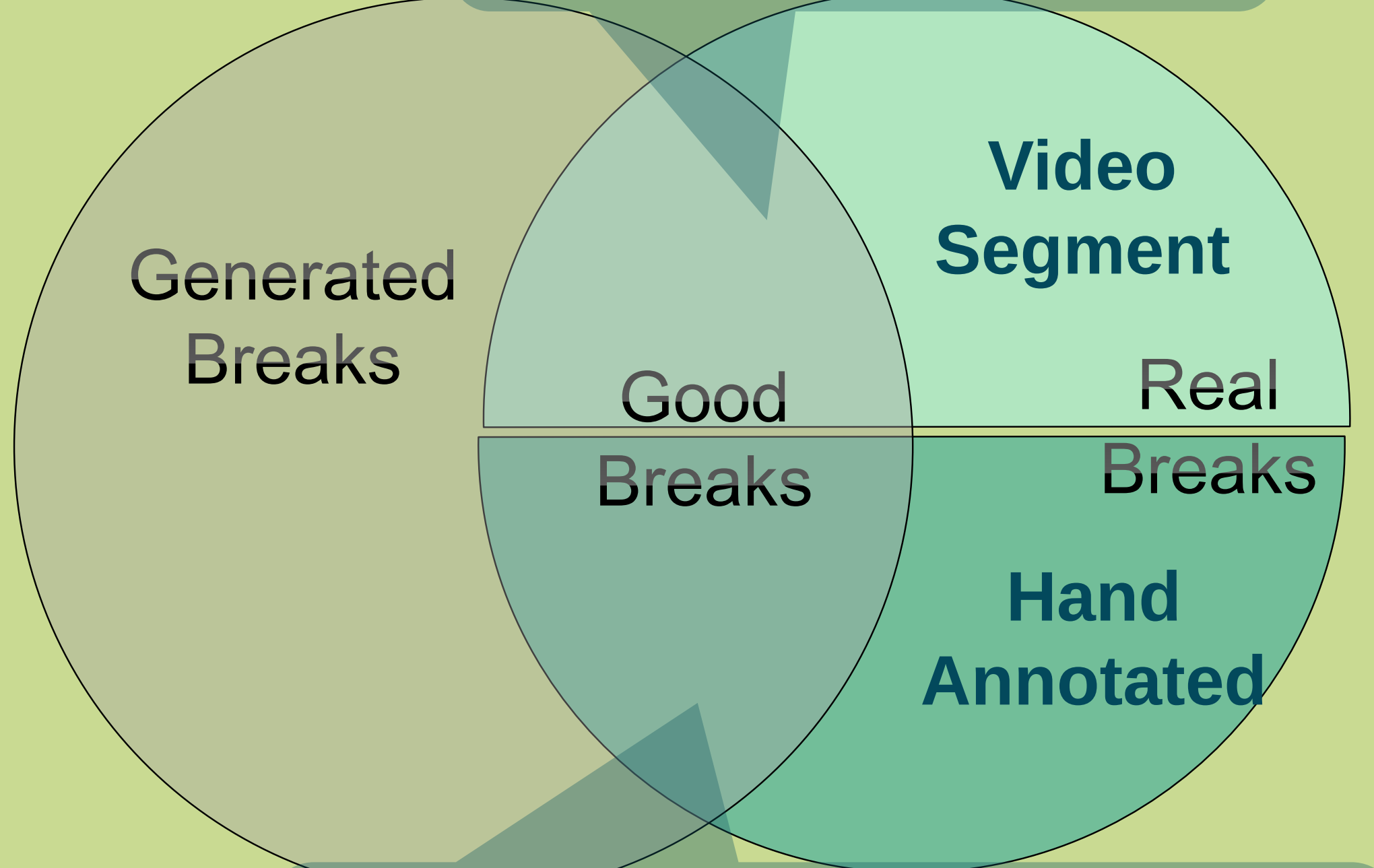
Break text into topic segments

- In the raw text, we categorize each word into different topics. A word belongs to topic -1 if it does not belong to any topic that is detected by our topic model.
- By calculating each topics' frequency of appearance , we determine the main topic over a fixed number of texts.
- When the main topic shifts, we generate breaks that separate the text into topic segments.
  - For example:



- Sample Evaluation:

Approximately 121 words away from real breaks .



Precision="No garbage?"= 0.704545
Recall="Got everything?"=0.574074

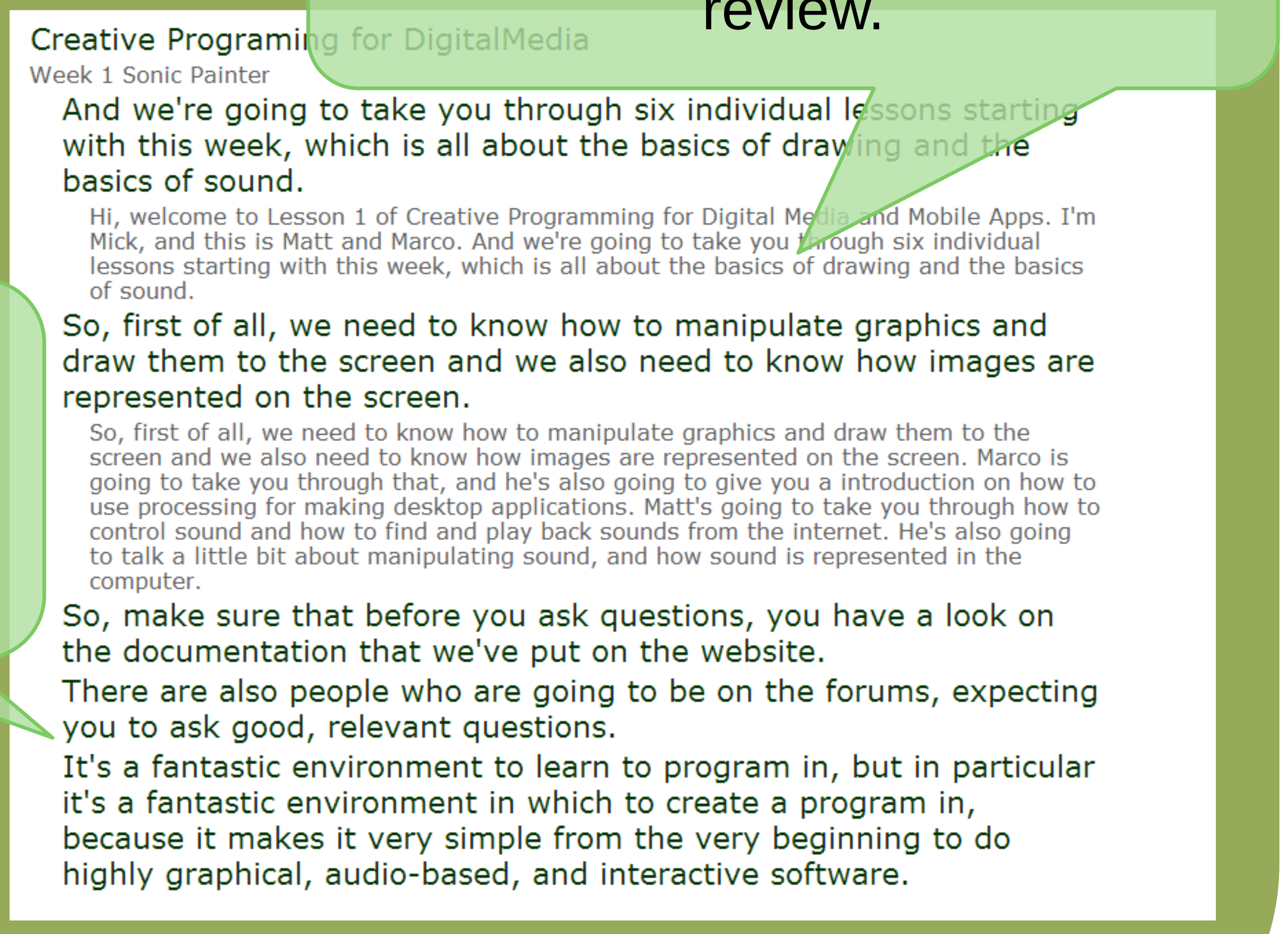
Summarize and Extract Main Sentences

- We summarize each topic segments into short sentences by using sentence extraction. [4]
  - Sentence extraction=identify the most salient sentences of a text.
- Users can decide the length of the sentences to be extracted for each topic segments.

Attach Format

- We format the note into html files.
- For example:

Main sentences are separated from topic segments for easy review.



Creative Programing for DigitalMedia
Week 1 Sonic Painter

And we're going to take you through six individual lessons starting with this week, which is all about the basics of drawing and the basics of sound.

Hi, welcome to Lesson 1 of Creative Programming for Digital Media and Mobile Apps. I'm Mick, and this is Matt and Marco. And we're going to take you through six individual lessons starting with this week, which is all about the basics of drawing and the basics of sound.

So, first of all, we need to know how to manipulate graphics and draw them to the screen and we also need to know how images are represented on the screen.

So, first of all, we need to know how to manipulate graphics and draw them to the screen and we also need to know how images are represented on the screen. Marco is going to take you through that, and he's also going to give you a introduction on how to use processing for making desktop applications. Matt's going to take you through how to control sound and how to find and play back sounds from the internet. He's also going to talk a little bit about manipulating sound, and how sound is represented in the computer.

So, make sure that before you ask questions, you have a look on the documentation that we've put on the website.

There are also people who are going to be on the forums, expecting you to ask good, relevant questions.

It's a fantastic environment to learn to program in, but in particular it's a fantastic environment in which to create a program in, because it makes it very simple from the very beginning to do highly graphical, audio-based, and interactive software.

Click to fold or expand topic segments

Finished Note

Future Study

- Continue testing and evaluating the system.
- Improve precision and recall by adjusting system's settings.
- Keep developing the user interface.

[1] https://www.coursera.org/, all data demonstrated in this poster comes from Creative Programming for Digital Media & Mobile Apps by Dr Marco Gillies, Dr Matthew Yee-King, Dr Mick Grierson
[2]Blei, David M. "Probabilistic Topic Models." *Communications of the ACM* 55.4 (2012): 77. Web
[3]McCallum, Andrew Kachites. "MALLET: A Machine Learning for Language Toolkit." http://mallet.cs.umass.edu. 2002.
[4] https://github.com/YauhenMinsk/SummaryLib *SummarizeLib*, developed by YauhenMinsk.