

LAB 2 REPORT

Table of Contents

Team Details.....	1
Demo Video.....	1
Rationale.....	2
Domains & Datasets.....	3
Datasets.....	3
API Conversation Data.....	3
Machine Learning Mastery Blogs & Articles.....	3
CMU Machine Learning Course Archives.....	4
Implementation.....	6
Data Extraction using Kaggle API.....	6
Web Scraping Machine Learning Mastery Blogs.....	6
Web Scraping + PDF text extraction: CMU ML Course Archives.....	7
Output.....	9
Kaggle API.....	9
Web Scraping: Machine Learning Mastery Blogs & Articles.....	9
Web Scraping + PDF Data Extraction.....	9
Individual Contributions.....	12

Team Details

Team Name: SSK

Name	USC ID
Shreyansh Baredia	3739756887
Soma Meghana Prathipati	4226812081
Kayvan Shah	1106650685

Demo Video

Youtube Video: <https://youtu.be/R1fHKz2EPxY>

Rationale

The decision to train chatbots focusing on machine learning arises from the desire to create versatile conversational agents that engage users on various topics. As machine learning continues to evolve and have a significant impact, it has become increasingly relevant to current technological trends. Chatbots become proficient at addressing user inquiries ranging from basic principles to advanced methodologies by imparting knowledge about machine learning concepts, algorithms, and applications. This approach aligns with user interests and educational goals, providing a platform for users to seek information, clarification, and guidance on machine learning topics. Chatbots' adaptability to theoretical and practical machine learning enhances their utility. This focus also aligns with integrating chatbots into emerging technologies, ensuring they can seamlessly interact with users exploring or implementing machine learning solutions. With training in machine learning, chatbots' capabilities and responsiveness across diverse user interactions are enhanced.

However, existing chatbots may require more support regarding their depth of understanding, context retention, and ability to provide accurate and up-to-date information. These limitations may be due to a need for more specificity in training data or challenges in capturing nuanced and evolving concepts. Focusing on machine learning in training chatbots allows one to address these limitations and improve overall performance and accuracy.

1. **Depth of Understanding:** Machine learning encompasses many topics, from foundational principles to advanced techniques. Training chatbots with a deep understanding of machine learning allows them to engage in more nuanced conversations, providing detailed explanations and insights into complex concepts that may be challenging for generic chatbots.
2. **Context Retention:** Machine learning involves building models to adapt and learn from data. By incorporating this aspect into chatbot training, there's a potential improvement in the bot's ability to retain context throughout a conversation. This enables more coherent and context-aware responses, enhancing the overall user experience.
3. **Up-to-date Information:** Machine learning is ever-evolving, with continual breakthroughs and fresh discoveries. Concentrating on machine learning during training empowers chatbots to remain abreast of the most recent developments, guaranteeing that they can furnish users with precise and up-to-date information. This is particularly crucial for a field where knowledge evolves swiftly.
4. **Specialized Knowledge:** Machine learning often requires technical knowledge of algorithms, models, and applications. Training chatbots specifically in this domain equips them with the necessary expertise to answer user queries accurately and with a higher level of detail compared to general-purpose chatbots.
5. **Problem-Solving Capability:** Machine learning involves solving complex problems, and training chatbots in this area enhances their problem-solving capabilities. Users seeking assistance in understanding or solving machine learning-related challenges can benefit from a chatbot that is well-versed in the intricacies of the field.

Domains & Datasets


Shortlisted Domain: Machine Learning

Datasets

API Conversation Data

Link-<https://www.kaggle.com/datasets/kreeshrajani/3k-conversations-dataset-for-chatbot>

Description: This dataset is used for research or training natural language processing (NLP) models. The dataset may include various conversations such as casual or formal discussions, interviews, customer service interactions, or social media conversations. It has conversation datasets that train chatbots and virtual assistants to interact with users more human-likely.

3K Conversations Dataset for C			
Data Card		Code (3)	Discussion (1)
		3510 unique values	3512 unique values
0	hi, how are you doing?	i'm fine. how about yourself?	
1	i'm fine. how about yourself?	i'm pretty good. thanks for asking.	
2	i'm pretty good. thanks for asking.	no problem. so how have you been?	

Machine Learning Mastery Blogs & Articles

Link-<https://machinelearningmastery.com/linear-regression-for-machine-learning/>

Description: This web page introduces linear regression for machine learning. It explains what linear regression is, why it belongs to both statistics and machine learning, the many names by which it is known, the representation and learning algorithms used to create a linear regression model, and how to best prepare the data for modeling. The web page also includes examples of calculating and using linear regression in Python and Excel and tips and tricks for improving the model's performance.

Linear Regression Learning the Model

Learning a linear regression model means estimating the values of the coefficients used in the representation with the data that we have available.

In this section, we will take a brief look at four techniques to prepare a linear regression model. This is not enough information to implement them from scratch, but enough to get a flavor of the computation and trade-offs involved.

There are many more techniques because the model is so well studied. Take note of Ordinary Least Squares because it is the most common method used in general. Also take note of Gradient Descent as it is the most common technique taught in machine learning classes.

1. Simple Linear Regression

With simple linear regression when we have a single input, we can use statistics to estimate the coefficients.

This requires that you calculate statistical properties from the data such as means, standard deviations, correlations and covariance. All of the data must be available to traverse and calculate statistics.

Snapshot of the blog

CMU Machine Learning Course Archives

Link: <https://www.cs.cmu.edu/~ninamf/courses/601sp15/lectures.shtml>

Description: The page includes information about the course schedule, recitations, Piazza webpage, course description, prerequisites, textbooks, grading, and auditing policy.

Scraping this public data can be helpful for a teaching assistant assisting bot in several ways:

- **Automated Information Retrieval:** The bot can retrieve up-to-date information about the course schedule, recitation timings, and any changes in the syllabus. This ensures that the teaching assistant is always informed.
- **Answering Student Queries:** By scraping data from the Piazza webpage, the bot can assist students with frequently asked questions, discussions, and announcements related to the course.
- **Assignment and Project Updates:** The bot can monitor and provide updates on homework assignments, midterm schedules, and details about the final project. This helps students stay organized.
- **Prerequisite Review:** The bot can provide review materials or resources for students who need to catch up on prerequisites. It can also schedule recitation sessions for reviewing basic concepts.
- **Textbook Recommendations:** The bot can recommend relevant readings or resources to students based on the textbooks mentioned. This ensures that students have access to additional learning materials.

- Grading Information: The bot can keep track of grading components, such as midterm percentages, homework weightage, and the final project. It can provide reminders and updates about grading milestones.
- Auditing Policy Awareness: The bot can inform students about the auditing policy and any constraints related to course attendance. This ensures that students are aware of the class limitations.


By automating these aspects, a teaching assistant bot can enhance communication, streamline information access, and provide valuable support to students and the teaching staff.

Machine Learning

10-601, Spring 2015

Carnegie Mellon University

Tom Mitchell and **Maria-Florina Balcan**



[Home](#)
[People](#)
[Lectures](#)
[Recitations](#)
[Homeworks](#)
[Project](#)
[Previous material](#)

This is a tentative schedule and is subject to change.
Please note that Youtube takes some time to process videos before they become available.

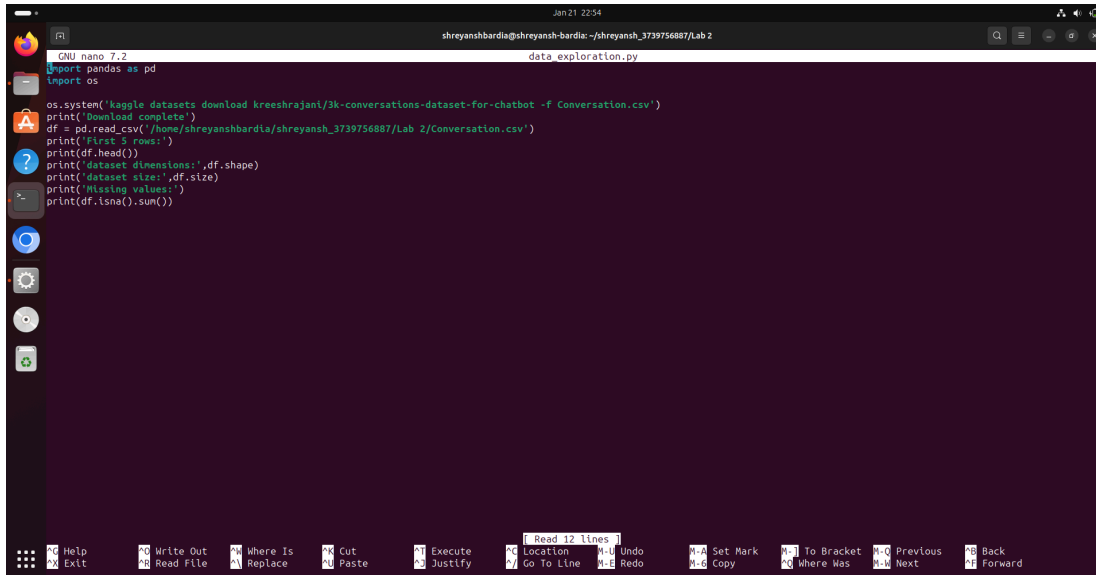
Date	Lecture	Topics	Readings and useful links	Handouts
Jan 12	Intro to ML Decision Trees	<ul style="list-style-type: none"> Machine learning examples Well defined machine learning problem Decision tree learning 	Mitchell: Ch 3 Bishop: Ch 14.4 The Discipline of Machine Learning	Slides Video
Jan 14	Decision Tree learning Review of Probability	<ul style="list-style-type: none"> The big picture Overfitting Random variables and probabilities 	Mitchell: Ch 3 Andrew Moore's Basic Probability Tutorial	Slides Annotated Slides Video
Jan 21	Probability and Estimation	<ul style="list-style-type: none"> Bayes rule MLE MAP 	Mitchell: Estimating Probabilities	Slides Annotated Slides Video

We only focus on scraping the schedule and course-related material, such as syllabus and slides, from the lecture table.

Implementation

Data Extraction using Kaggle API

In this script, we have used the Kaggle API to get the dataset from the Kaggle website. Once the dataset is downloaded, we read it using pandas, and then we display the first 5 rows, the dimensions of the dataset, the size of the dataset, and the missing values.

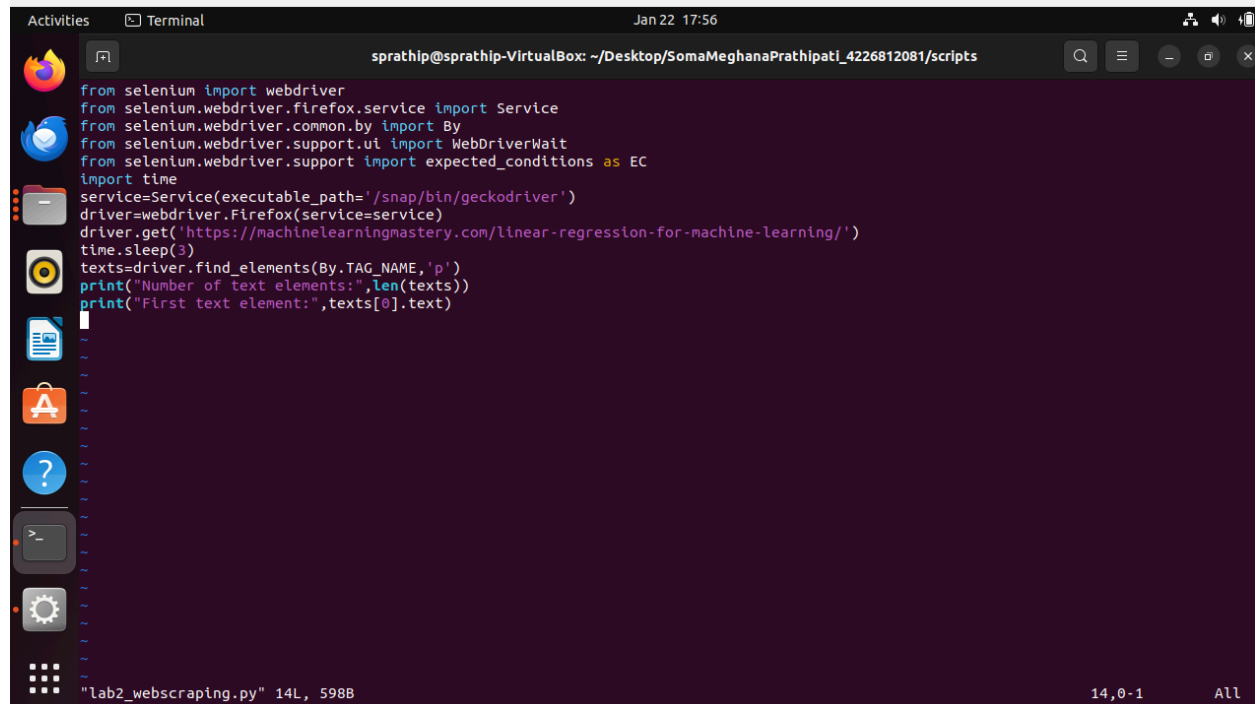


```
GNU nano 7.2 data_exploration.py
import pandas as pd
import os

os.system('kaggle datasets download kreeshrajan/3k-conversations-dataset-for-chatbot -f Conversation.csv')
print('Download complete')
df = pd.read_csv('/home/shreyanshbardia/shreyansh_3739756887/Lab 2/Conversation.csv')
print('Read 5 rows:')
print(df.head())
print('dataset dimensions:', df.shape)
print('dataset size:', df.size)
print('Missing values:')
print(df.isna().sum())
```

Web Scraping Machine Learning Mastery Blogs

The below Python code utilizes Selenium to automate web scraping. It opens a webpage, waits for 3 seconds, extracts all paragraph elements, and then prints the count and text of the first paragraph about linear regression in machine learning from the specified URL.



```
from selenium import webdriver
from selenium.webdriver.firefox.service import Service
from selenium.webdriver.common.by import By
from selenium.webdriver.support.ui import WebDriverWait
from selenium.webdriver.support import expected_conditions as EC
import time
service=Service(executable_path='/snap/bin/geckodriver')
driver=webdriver.Firefox(service=service)
driver.get('https://machinelearningmastery.com/linear-regression-for-machine-learning/')
time.sleep(3)
texts=driver.find_elements(By.TAG_NAME,'p')
print("Number of text elements:",len(texts))
print("First text element:",texts[0].text)
```

The screenshot shows a terminal window with a dark background and light-colored text. The code is a Selenium WebDriver script that sets up a Firefox driver, navigates to a specific URL, waits for 3 seconds, and then finds all paragraph elements on the page. It prints the total number of text elements and the text of the first element. The terminal window title is 'sprathip@sprathip-VirtualBox: ~/Desktop/SomaMeghanaPrathipati_4226812081/scripts'. The bottom status bar shows the file path 'lab2_web scraping.py' with line 14 and column 5988, and a cursor position of 14,0-1.

Web Scraping + PDF text extraction: CMU ML Course Archives

This script is designed to scrape information from a course webpage and process PDF files related to the course. Here's a summary:

1. Scraping Course Information:
 - a. The script starts by scraping information from a specific course webpage at Carnegie Mellon University.
 - b. It collects details such as lecture names, topics, readings, and handouts from the course schedule table.
 - c. The scraped data is stored in a Pandas DataFrame and saved as a CSV file.
2. Processing PDF Files:
 - a. The script reads the CSV file containing course information.
 - b. For each handout that corresponds to slides, it downloads the associated PDF file and extracts text from each page.
 - c. Text content from each page is saved in separate text files within a folder named after the lecture title.
 - d. The PDF text extraction is performed using the PyMuPDF (fitz) library.
3. Logging and Output:
 - a. Logging statements provide information about the progress and any encountered errors.
 - b. The final CSV file containing course details is saved, and text files from PDFs are organized in a designated folder structure.
4. Organization:
 - a. The script utilizes modular functions for better code organization.

- b. It uses libraries like BeautifulSoup for web scraping, HTTP requests, fitz for PDF processing, and Pandas for data handling.

```
def read_pdf_from_url(url, output_folder):
    try:
        # Create the output folder if it doesn't exist
        if os.path.exists(output_folder):
            shutil.rmtree(output_folder)

        os.makedirs(output_folder, exist_ok=True)

        # Download the PDF file
        pdf_data = requests.get(url).content

        # Open the PDF document
        pdf_document = fitz.open(stream=pdf_data, filetype="pdf")

        # Iterate through pages
        for page_number in range(pdf_document.page_count)[:5]:
            # Get text content of the page
            page = pdf_document[page_number]
            text = page.get_text()

            # Create a text file for each page
            output_file_path = os.path.join(output_folder, f"page_{page_number + 1}.txt")
            with open(output_file_path, "w", encoding="utf-8") as output_file:
                output_file.write(text)

        logging.info(f"Text files for each page created in `{os.path.relpath(output_folder)}`")

        # Close the PDF document
        pdf_document.close()

    except Exception as e:
        logging.exception(f"Error: {e}")

def extract_text_from_pdfs(data):
    for row in tqdm(data.to_dict(orient="records")[:5]):
        title = row["lecture"].replace(" ", "_").lower()
        handouts = [item for item in eval(row["handouts"]) if item["name"] == "Slides"]

        output_dir = os.path.join(Path.PDF_FILE_DIR, title)
        for h in handouts:
            read_pdf_from_url(h["link"], output_dir)
```


Output

Kaggle API

```

shreyanshbardia@shreyansh-bardia:~/shreyansh_3739756887/Lab 2$ python3 data_exploration.py
Warning: Your Kaggle API key is readable by other users on this system! To fix this, you can run 'chmod 600 /home/shreyanshbardia/.kaggle/kaggle.json'
Conversation.csv: Skipping, found more recently modified local copy (use --force to force download)
Download complete
First 5 rows:
  Unnamed: 0      question      answer
0          0      hi, how are you doing?      i'm fine, how about yourself?
1          1      i'm fine, how about yourself?      i'm pretty good, thanks for asking.
2          2      i'm pretty good, thanks for asking.      no problem, so how have you been?
3          3      no problem, so how have you been?      i've been great, what about you?
4          4      i've been great, what about you?      i've been good, i'm in school right now.

dataset dimensions: (3725, 3)
dataset size: 11175
Missing values:
  Unnamed: 0      0
  question      0
  answer      0
dtype: int64
shreyanshbardia@shreyansh-bardia:~/shreyansh_3739756887/Lab 2$

```

Web Scraping: Machine Learning Mastery Blogs & Articles

```

sprathip@sprathip-VirtualBox: ~/Desktop/SomaMeghanaPrathipati_4226812081/scripts$ python3 lab2_webscrapping.py
Number of text elements: 299
First text element: Linear regression is perhaps one of the most well known and well understood algorithms in statistics and machine
sprathip@sprathip-VirtualBox: ~/Desktop/SomaMeghanaPrathipati_4226812081/scripts$

```

Web Scraping + PDF Data Extraction

On script execution:

1. Few records are printed in the stdout
2. The first five rows of the **ml_course.csv** are displayed. This file contains the data extracted from the lecture tables from the course website.
3. Subsequently, the data frame info is displayed, giving information about the columns in the dataset.

```

shreyansh@shreyansh-bardia:~/shreyansh_3739756887/Lab 2$ python3 data_exploration.py
Warning: Your Kaggle API key is readable by other users on this system! To fix this, you can run 'chmod 600 /home/shreyanshbardia/.kaggle/kaggle.json'
Conversation.csv: Skipping, found more recently modified local copy (use --force to force download)
Download complete
First 5 rows:
  Unnamed: 0      question      answer
0          0      hi, how are you doing?      i'm fine, how about yourself?
1          1      i'm fine, how about yourself?      i'm pretty good, thanks for asking.
2          2      i'm pretty good, thanks for asking.      no problem, so how have you been?
3          3      no problem, so how have you been?      i've been great, what about you?
4          4      i've been great, what about you?      i've been good, i'm in school right now.

dataset dimensions: (3725, 3)
dataset size: 11175
Missing values:
  Unnamed: 0      0
  question      0
  answer      0
dtype: int64
shreyanshbardia@shreyansh-bardia:~/shreyansh_3739756887/Lab 2$

```

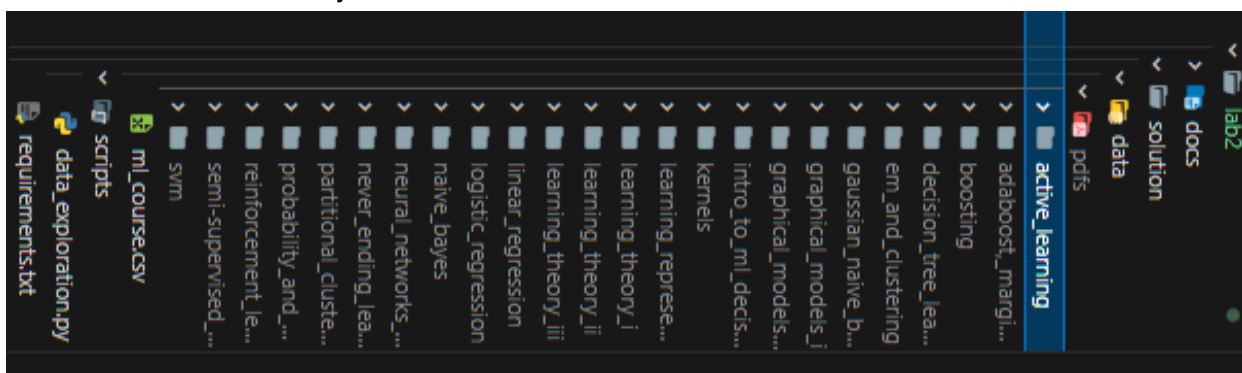
Below is the snapshot of the `ml_course.csv` file.

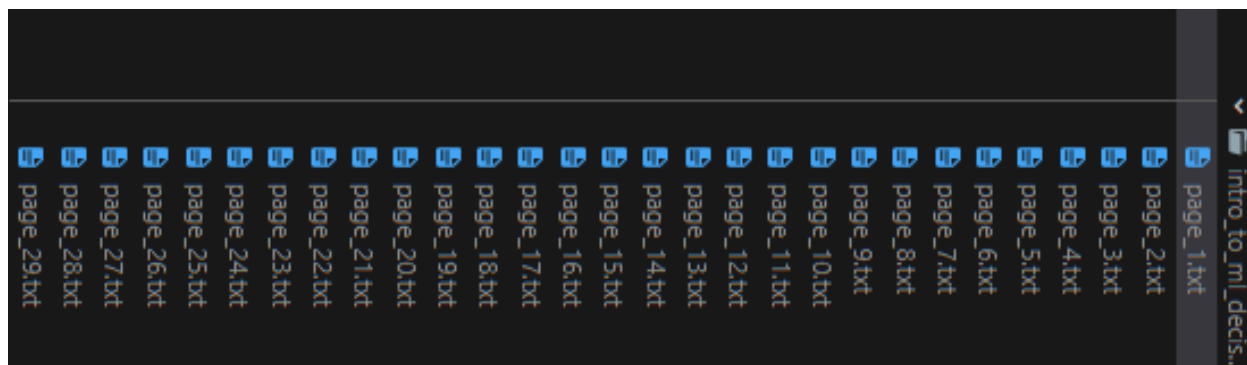
```
data_exploration.py  ml_course.csv  requirements.txt
lab2 > solution > data > ml_course.csv > data
1 lecture,topics,readings,handouts
2 Intro to ML Decision Trees,["(name: 'Machine Learning examples'), (name: 'Well defined machine learning problem'), (name: 'Decision tree learning')"],["(name: 'The Discipline of Machine Learning', 'link: 'http://www.cs.cmu.edu/77
3 Decision tree learning Review of Probability,["(name: 'The big picture'), (name: 'Overfitting'), (name: 'Random variables and probabilities')"],["(name: 'Andrew Moore's Basic Probability Tutorial', 'link: 'http://www.autonlab.o
4 Probability and Estimation,["(name: 'Bayes rule'), (name: 'MLE'), (name: 'MAP')"],["(name: 'Estimating Probabilities', 'link: 'http://www.cs.cmu.edu/77etom/mlbook/Joint_MLE_MAP.pdf'), (name: 'Mitchell:', 'link: None)],["(name:
5 Naive Bayes,["(name: 'Conditional Independence'), (name: 'Naive Bayes: why and how?')"],["(name: 'Naive Bayes and Logistic Regression', 'link: 'http://www.cs.cmu.edu/77etom/mlbook/NaiveBayesLogReg.pdf'), (name: 'Mitchell:', 'link: No
6 Gaussian Naive Bayes,["(name: 'Gaussian Bayes classifiers'), (name: 'K-means Classification'), (name: 'Brain image classification'), (name: 'Form of decision surfaces')"],["(name: 'Naive Bayes and Logistic Regression', 'link:
7 Logistic Regression,["(name: 'Naive Bayes - the big picture'), (name: 'Logistic Regression: Maximizing conditional likelihood'), (name: 'Gradient descent as a general learning/optimization method')"],["(name: 'Naive Bayes and Logis
8 Linear Regression,["(name: 'Generative/discriminative models'), (name: 'Minimizing squared error and maximizing data likelihood'), (name: 'Regularization'), (name: 'Bias-variance decomposition')"],["(name: 'Slides', 'link:
9 Learning Theory I,["(name: 'Distributional Learning'), (name: 'PAC and Statistical Learning Theory'), (name: 'Sample Complexity')"],["(name: 'Notes on Generalization Guarantees', 'link: 'sc-2015.pdf'), (name: 'Mitchell: Ch 7',
10 Learning Theory II,["(name: 'Sample Complexity'), (name: 'Shattering and VC Dimension'), (name: 'Sauer's Lemma')"],["(name: 'Notes on Generalization Guarantees', 'link: 'sc-2015.pdf'), (name: 'Mitchell: Ch 7', 'link: None)],["(name:
11 Learning Theory III,["(name: 'Rademacher Complexity'), (name: 'Overfitting and Regularization')"],["(name: 'Slides', 'link: 'https://www.cs.cmu.edu/~ninam/courses/601sp15/slides/10_theory3_2-16-2015.pdf'), (name: 'Video', 'link:
12 Graphical Models I,["(name: 'Bayes Nets'), (name: 'Representing joint distributions with conditional independence assumptions')"],["(name: 'Bishop chapter 8, through 8.2', 'link: None)],["(name: 'Slides', 'link: 'https://www.cs.c
13 Graphical Models II,["(name: 'Inference'), (name: 'Learning from fully observed data'), (name: 'Learning from partially observed data')"],["(name: 'Annotated Slides', 'link: 'https://www.cs.cmu.edu/~ninam/courses/601sp15/slides
14 Graphical Models III,["(name: 'EM'), (name: 'Semi-supervised learning')"],["(name: 'Bishop Chapter 8 Mitchell Chapter 6', 'link: None)],["(name: 'Slides', 'link: 'https://www.cs.cmu.edu/~ninam/courses/601sp15/slides/13_Grphd3
15 EM and Clustering,["(name: 'Mixtures of Gaussian clustering'), (name: 'K-means clustering')"],["(name: 'Bishop Chapter 8 Mitchell Chapter 6', 'link: None)],["(name: 'Slides', 'link: 'https://www.cs.cmu.edu/~ninam/courses/601sp15
16 Boosting,["(name: 'Weak vs Strong (PAC) Learning'), (name: 'Boosting Accuracy'), (name: 'AdaBoost')"],["(name: 'The Boosting Approach to Machine Learning: An Overview', 'link: 'https://www.cs.princeton.edu/picasso/mats/schapire00
17 'AdaBoost, Margins, Perceptron',["(name: 'AdaBoost: Generalization Guarantees (margin and margins based)'), (name: 'Geometric Margins and Perceptron')"],["(name: 'Notes on Perceptron-notes.pdf'), (name: 'Slides', 'link:
18 'Kernels,["(name: 'Geometric Margins'), (name: 'Kernels: Kernelizing a Learning Algorithm'), (name: 'Kernelized Perceptron')"],["(name: 'Bishop 6.1 and 6.2', 'link: None)],["(name: 'Slides', 'link: 'https://www.cs.cmu.edu/~nin
19 SVMs,["(name: 'Geometric Margins'), (name: 'SVM: Primal and Dual Forms'), (name: 'Kernelizing SVM'), (name: 'Semi-supervised SVM')"],["(name: 'Notes on SVM by Andrew Ng', 'link: 'http://cs22
20 Semi-supervised Learning,["(name: 'Transductive SVM'), (name: 'Co-training and Multi-view Learning'), (name: 'Graph-based Methods')"],["(name: 'Semi-supervised Learning' in Encyclopedia of Machine Learning', 'link: 'http://pag
21 Active Learning,["(name: 'Batch Active Learning'), (name: 'Selective Sampling and Active Learning'), (name: 'Smoothing Bias')"],["(name: 'Two Faces of Active Learning', 'link: 'http://csweb.ucsd.edu/~dasgupta/papers/twofaces.pdf'), (name:
22 Partitional Clustering Hierarchical Clustering,["(name: 'K-means, Lloyd's method, k-means++'), (name: 'Agglomerative Clustering')"],["(name: 'Center Based Clustering: A Foundational Perspective', 'link: 'http://www.cs.cmu.edu/~ml
23 Learning Representations Dimensionality Reduction,["(name: 'Principal Component Analysis'), (name: 'Kernel Principal Component Analysis')"],["(name: 'Bishop 12.1, 12.3', 'link: None)],["(name: 'Slides', 'link: 'https://www.cs.c
24 Never Ending Learning,["(name: 'Slides', 'link: 'https://www.cs.cmu.edu/~ninam/courses/601sp15/slides/23_neel_4-13-2015.pdf'), (name: 'Video', 'link: 'http://youtu.be/0sP4l486ag')"],["(name: 'Slides', 'link: 'https://www.cs.c
25 Neural Networks Deep Learning,["(name: 'Mitchell, Chapter 4', 'link: None)],["(name: 'Slides', 'link: 'https://www.cs.cmu.edu/~ninam/courses/601sp15/slides/24_neel_4-13-2015.pdf'), (name: 'Video', 'link: 'http://youtu.be/0sP4l486ag')"],["(name: 'Slides', 'link: 'https://www.cs.c
26 Reinforcement Learning,["(name: 'Markov Decision Processes'), (name: 'Value Iteration'), (name: 'Q-learning')"],["(name: 'Kaelbling, et al., Reinforcement Learning: A Survey', 'link: 'https://www.jair.org/media/301/live-301-1562-
27 Deep Learning Differential Privacy Discussion on the Future of ML,["(name: 'Slides (Privacy)', 'link: 'https://www.cs.cmu.edu/~ninam/courses/601sp15/slides/26_privacy_4-22-2015.pdf'), (name: 'Slides (Deep Nets)', 'link: 'http
28 Course review,["(name: 'Slides', 'link: 'https://www.cs.cmu.edu/~ninam/courses/601sp15/slides/26_privacy_4-22-2015.pdf'), (name: 'Slides (Deep Nets)', 'link: 'http
29
30
```

The below part shows the extraction of text from the page of the PDF file.

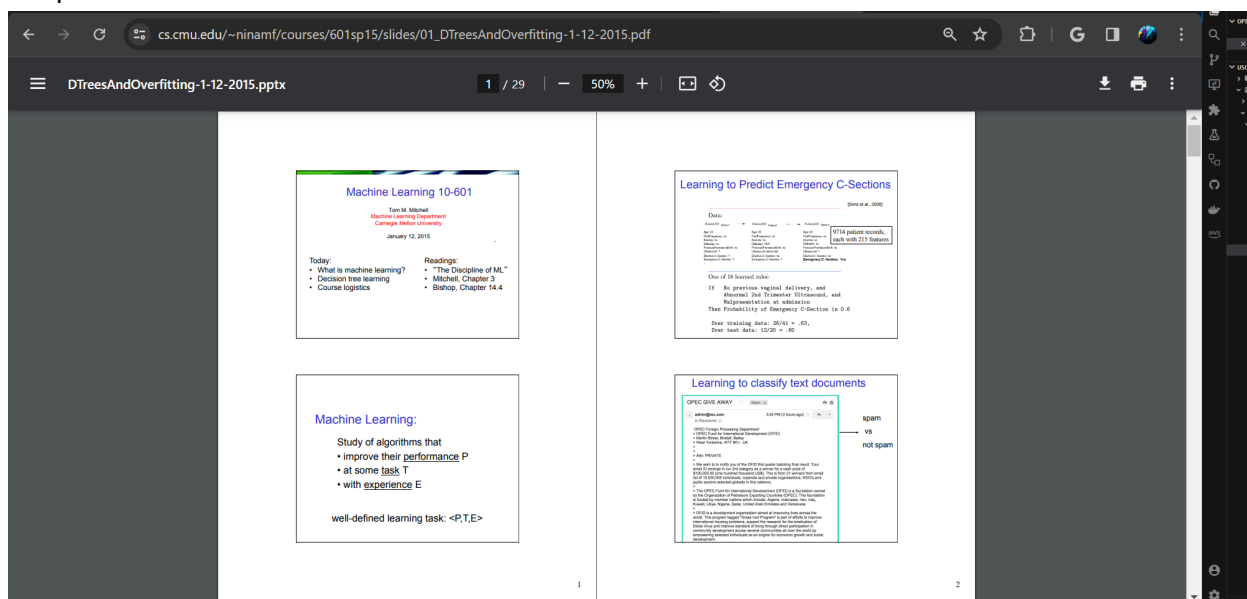
```
INFOroot:Saving course details files to 'lab2\solution\data\ml_course.csv'
INFOroot:Extracting data from PDF files...
0% | 0/27 [00:00:00, 20.30u/t/s]
INFOroot:Text files for each page created in 'lab2\solution\data\pdfs\intro_to_ml_decision_trees'
40% | 1/27 [00:20:08:47, 20.30u/t/s]
70% | 2/27 [00:22:04:01, 9.67u/t/s]
INFOroot:Text files for each page created in 'lab2\solution\data\pdfs\decision_tree_learning_review_of_probability'
110% | 3/27 [00:24:02:29, 6.25u/t/s]
150% | 4/27 [00:27:03:22, 8.80u/t/s]
190% | 5/27 [00:40:02:29, 6.77u/t/s]
INFOroot:Text files for each page created in 'lab2\solution\data\pdfs\gaussian_naive_bayes'
230% | 6/27 [00:50:58:45, 7.86u/t/s]
270% | 7/27 [01:00:02:49, 8.48u/t/s]
310% | 8/27 [01:03:02:11, 6.89u/t/s]
350% | 9/27 [01:06:01:40, 5.57u/t/s]
390% | 10/27 [01:08:01:14, 4.36u/t/s]
430% | 11/27 [01:14:01:17, 4.86u/t/s]
470% | 12/27 [01:24:01:09, 4.94u/t/s]
510% | 13/27 [01:27:08:59, 4.55u/t/s]
550% | 14/27 [01:29:00:45, 3.81u/t/s]
590% | 15/27 [01:31:00:35, 3.27u/t/s]
630% | 16/27 [01:32:00:26, 2.70u/t/s]
670% | 17/27 [01:34:00:21, 2.42u/t/s]
710% | 18/27 [01:38:00:23, 2.95u/t/s]
750% | 19/27 [01:45:00:28, 4.13u/t/s]
790% | 20/27 [01:51:00:27, 4.66u/t/s]
830% | 21/27 [01:53:00:19, 3.89u/t/s]
870% | 22/27 [02:12:00:33, 8.40u/t/s]
910% | 23/27 [02:04:00:26, 9.52u/t/s]
950% | 24/27 [02:25:00:00, 5.40u/t/s]
100% | 25/27 [02:25:00:00, 5.40u/t/s]
```

Text files are written locally

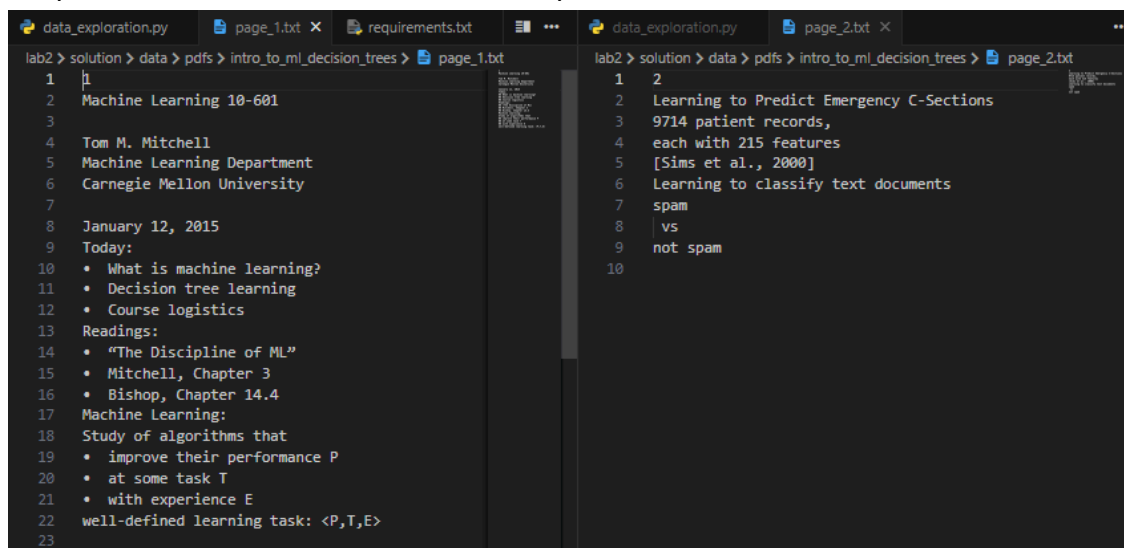




Snapshot of a PDF file



Snapshot of text extracted from the above pdf file



Individual Contributions

- **Shreyansh Baredia:**
 - Wrote the script `data_exploration.py` and downloaded the Conversation data from the Kaggle API.
- **Soma Meghana Prathipati:**
 - Implemented `web_scraping.py` file to fetch the data by scraping the machine learning mastery blogs and displaying the text
- **Kayvan Shah:**
 - Web scraper for extracting data from the CMU ML course archives.
 - Extract text content from PDF file URLs obtained post-scraping.