

# DSCI 560: Data Science Professional Practicum

## Laboratory Assignment 4 Part 2

### Team Details:

Team Name: Team PVC

| Team Member         | USC ID     |
|---------------------|------------|
| Haoran Zhang        | 3502885083 |
| Venkatesh Dharmaraj | 6773159759 |
| Yichen An           | 2780765696 |

### Message Content Abstraction

Doc2Vec: This takes in the CSV file from lab4 part 1 and transforms the keyword extracted for that document into vector forms

```
import pandas as pd
from gensim.models.doc2vec import Doc2Vec, TaggedDocument
from nltk.tokenize import word_tokenize

# Load the CSV file
df = pd.read_csv('posts.csv')

# Assuming 'formatted_post_title' is the column with text data
data = df['formatted_post_title'].tolist()

# Preprocess the documents and create TaggedDocuments, using post_id_tech as tags if needed
tagged_data = [TaggedDocument(words=word_tokenize(doc.lower()), tags=[str(i)]) for i, doc in enumerate(data)]

# Train the Doc2Vec model (adjust parameters as needed)
model = Doc2Vec(vector_size=20, min_count=2, epochs=50)
model.build_vocab(tagged_data)
model.train(tagged_data, total_examples=model.corpus_count, epochs=model.epochs)

# Infer vectors for the original dataset (or any new data)
document_vectors = [model.infer_vector(word_tokenize(doc.lower())) for doc in data]

# Print the document vectors alongside their titles
for i, vec in enumerate(document_vectors):
    print(f"Document: {data[i]}")
    print(f"Vector: {vec}\n")
```

```

Document: In a First, a Prosthetic Limb Can Sense Temperature Like a Living Hand
Vector: [ 0.3332558 -0.17202151 -0.02791681  0.07258766  0.26919416 -0.37276015
-0.11447767  0.44625977  0.06345354  0.13722219  0.06276309  0.01030525
0.10246638  0.05064563  0.08841644  0.18558119  0.59672666 -0.03281841
-0.13928427 -0.02914918]

Document: Alternate qubit design does error correction in hardware. Early-stage technology has the potential to cut qubits needed for useful computers.
Vector: [ -0.3256915  0.21789767 -0.18155953 -0.03502605  0.40092766 -0.40997815
0.23175985  0.43988124  0.01394675 -0.11393522  0.130297 -0.06024748
-0.11098453  0.33694077  0.29514938  0.55408835  0.5083762  0.06292582
-0.43755668 -0.64656985]

Document: Gel and lithium-ion tech could enable 1000-mile EV range on one charge | Researchers achieve EV battery breakthrough with silicon-based materials and gel electrolytes, moving closer to a 1,000-kilometer range on a single charge.
Vector: [ 0.04483001  0.05117815  0.08749112  0.26148008 -0.10640705 -0.48808905
-0.11357064  0.6323668  0.22529991  1.2392393 -0.23424962  0.5085762
-0.24022935 -0.37916362  0.0229969  0.80105174  0.52243483 -0.7136255
0.2907413 -0.0608465 ]

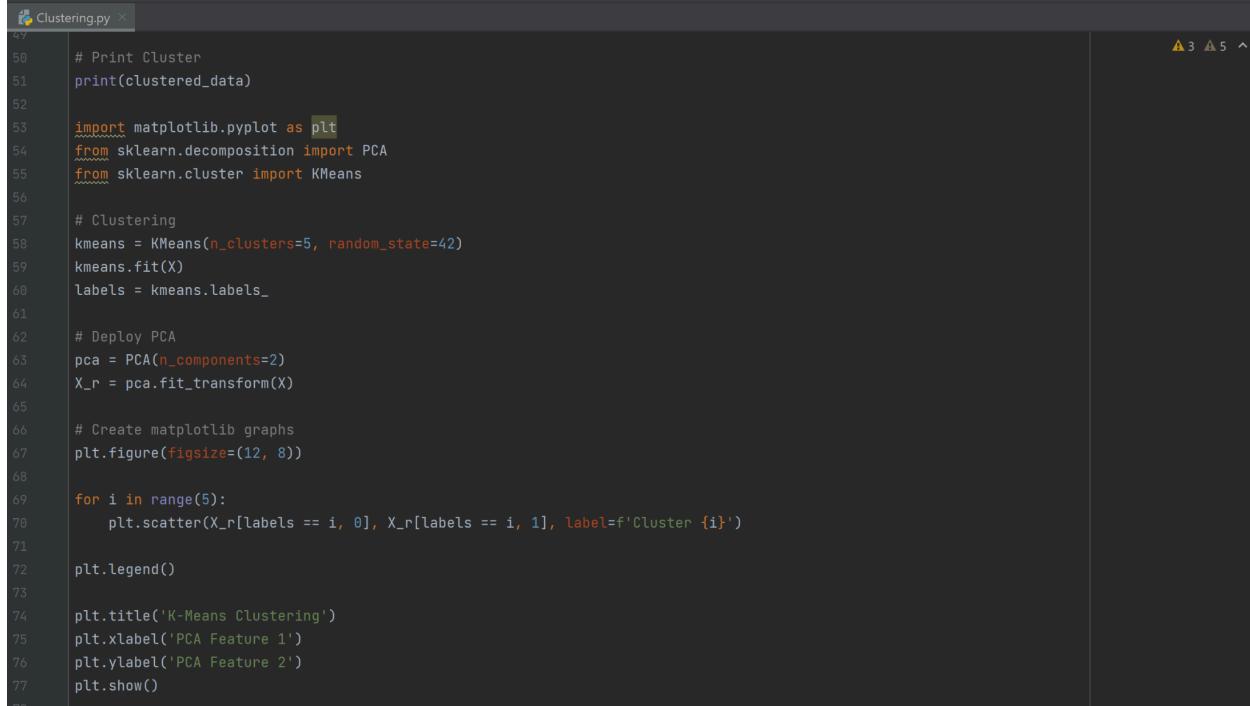
Document: Inhalable sensors could enable early lung cancer detection
Vector: [ -0.13791318  0.0185714 -0.49091372  0.24483877  0.18482071  0.1372347
0.3038938  0.14694436 -0.21829017  0.55681986  0.16870919 -0.22888522
0.11537617 -0.2140774  0.5503922 -0.03265684  0.2816531 -0.48537165
-0.11538043  0.0152991 ]

Document: Scientists develop a low-cost device to make cell therapy safer | A plastic microfluidic chip can remove some risky cells that could potentially become tumors before they are implanted in a patient.
Vector: [ 0.6079858 -0.13758685 -0.517426 -0.07735139 -0.6894494 -0.10371833
-0.35433185  0.3665978 -0.45838818  1.0604633 -0.35693288  0.1846211
0.39967406 -0.04315176  1.1437122  0.26860696  1.0905862 -1.1873078
-0.05681656  0.22932011]

```

## Clustering Messages

The inferred vectors are then used as input to a clustering algorithm, such as K-means. K-means clustering aims to partition the set of vectors into K clusters, with each document assigned to the cluster with the nearest mean vector. This clustering step groups together documents with similar meanings based on their vector representations.



```

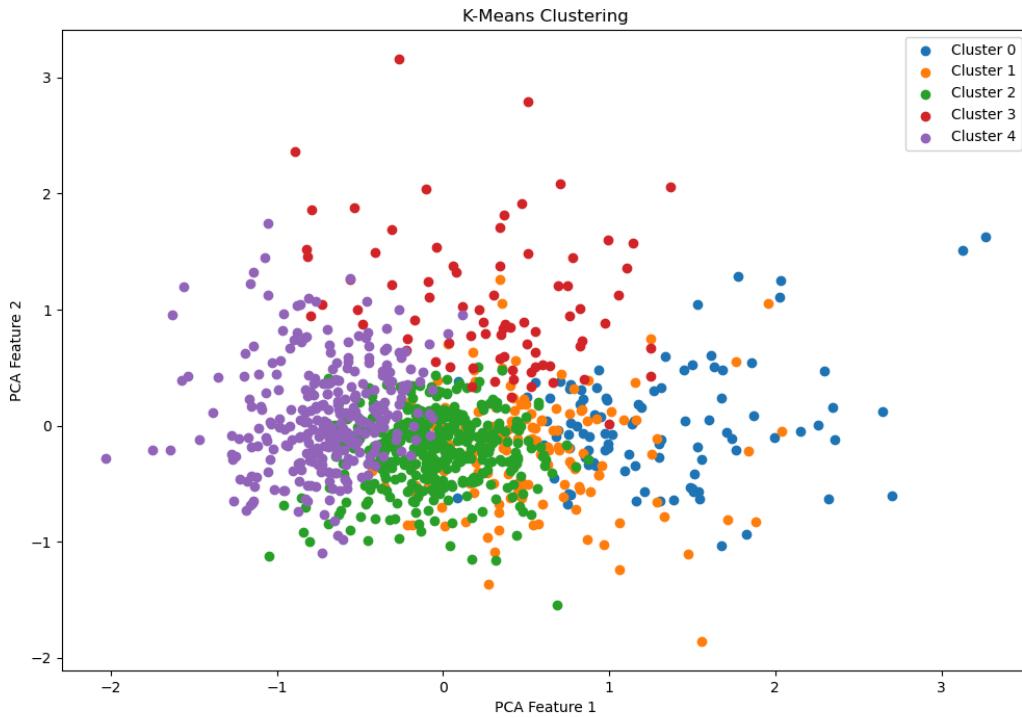
# Clustering.py
47
48 # Print Cluster
49 print(clustered_data)
50
51
52
53 import matplotlib.pyplot as plt
54 from sklearn.decomposition import PCA
55 from sklearn.cluster import KMeans
56
57 # Clustering
58 kmeans = KMeans(n_clusters=5, random_state=42)
59 kmeans.fit(X)
60 labels = kmeans.labels_
61
62 # Deploy PCA
63 pca = PCA(n_components=2)
64 X_r = pca.fit_transform(X)
65
66 # Create matplotlib graphs
67 plt.figure(figsize=(12, 8))
68
69 for i in range(5):
70     plt.scatter(X_r[labels == i, 0], X_r[labels == i, 1], label=f'Cluster {i}')
71
72 plt.legend()
73
74 plt.title('K-Means Clustering')
75 plt.xlabel('PCA Feature 1')
76 plt.ylabel('PCA Feature 2')
77 plt.show()

```

After clustering, the results can be analyzed to understand the common themes or topics within each cluster. Each cluster should ideally represent a collection of documents that are semantically similar to each other.

|     |   | document | cluster |
|-----|---|----------|---------|
| 0   | This ultrasound sticker senses changing stiffn... |          | 0       |
| 1   | In a First, a Prosthetic Limb Can Sense Temper... |          | 2       |
| 2   | Alternate qubit design does error correction i... |          | 4       |
| 3   | Scientists develop a low-cost device to make c... |          | 0       |
| 4   | Gel and lithium-ion tech could enable 1000-mil... |          | 0       |
| ..  |   | ...      | ...     |
| 992 | The clock speed wars are back as Intel brags a... |          | 2       |
| 993 | Nuclear fusion reactor in Korea reaches 100 mi... |          | 4       |
| 994 | Twitter whistleblower cites security flaws bef... |          | 2       |
| 995 | Two atomic clocks have been quantum entangled ... |          | 3       |
| 996 | Nuclear fusion reactor in Korea reaches 100 mi... |          | 4       |

For better interpretability, the high-dimensional vectors can be visualized in two or three dimensions using techniques like t-SNE (t-distributed Stochastic Neighbor Embedding). This helps in understanding the distribution and separation of the different clusters.



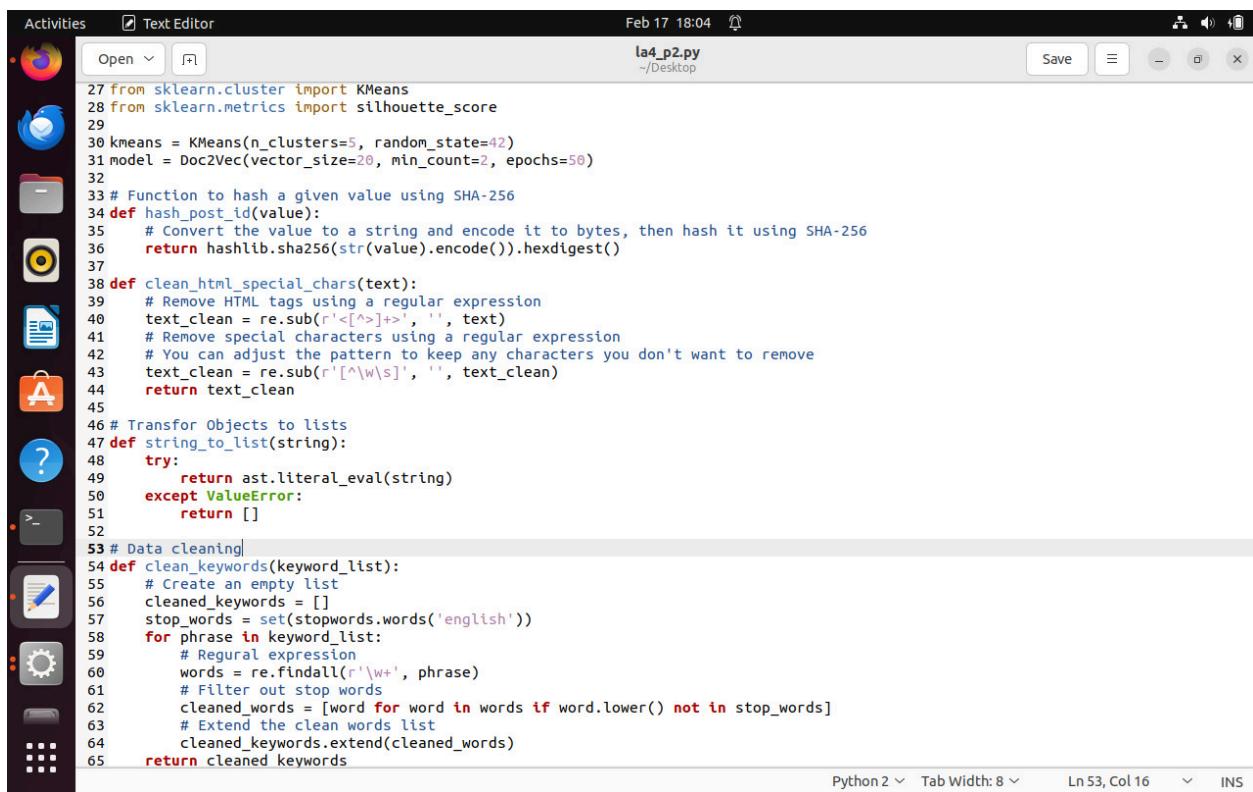
*Evaluation of the Algorithm:* The K-means clustering algorithm is been evaluated based on the [silhouette score](#).

## Automation

- The scripts of Part I of the assignment and the clustering algorithm are combined as a single compiled script. The logic for automation to perform the mentioned tasks is also added to the script.
- To execute the script, run the following command:
  - `python3 la4_p2.py <interval_minutes>`
  - `<interval_minutes>` can be any integer
- The script will fetch, preprocess, cluster, and store the data in the database periodically based on the ‘interval\_minutes’ value the user gives.
- *Storing Data in the Database:* The script will check whether the post already exists in the database using the `post_id` value. If exists, It updates the existing record. Else, it creates a new record in the database table.

- When the script is not updating the database in the background, the command line prompt will take keywords or a message as input and find the cluster that matches the input closest to it. The messages from a selected cluster should be displayed along with a graphical representation.
- The user needs to input ‘quit’ to stop the execution.
- In addition to the terminal outputs, the script also generates 3 CSVs:
  - posts.csv - contains scrapped post records
  - processed\_data.csv - contains processed post records
  - clustered\_data.csv - contains post and assigned cluster details

The script is shown in the below screenshots,



```

Activities Text Editor Feb 17 18:04 la4_p2.py ~/Desktop Save
Open ↗
27 from sklearn.cluster import KMeans
28 from sklearn.metrics import silhouette_score
29
30 kmeans = KMeans(n_clusters=5, random_state=42)
31 model = Doc2Vec(vector_size=20, min_count=2, epochs=50)
32
33 # Function to hash a given value using SHA-256
34 def hash_post_id(value):
35     # Convert the value to a string and encode it to bytes, then hash it using SHA-256
36     return hashlib.sha256(str(value).encode()).hexdigest()
37
38 def clean_html_special_chars(text):
39     # Remove HTML tags using a regular expression
40     text_clean = re.sub(r'<[^>]+>', '', text)
41     # Remove special characters using a regular expression
42     # You can adjust the pattern to keep any characters you don't want to remove
43     text_clean = re.sub(r'[^w\s]', '', text_clean)
44     return text_clean
45
46 # Transform Objects to lists
47 def string_to_list(string):
48     try:
49         return ast.literal_eval(string)
50     except ValueError:
51         return []
52
53 # Data cleaning
54 def clean_keywords(keyword_list):
55     # Create an empty list
56     cleaned_keywords = []
57     stop_words = set(stopwords.words('english'))
58     for phrase in keyword_list:
59         # Regular expression
60         words = re.findall(r'\w+', phrase)
61         # Filter out stop words
62         cleaned_words = [word for word in words if word.lower() not in stop_words]
63         # Extend the clean words list
64         cleaned_keywords.extend(cleaned_words)
65     return cleaned_keywords

```

Python 2 ▾ Tab Width: 8 ▾ Ln 53, Col 16 ▾ INS

The screenshot shows a Jupyter Notebook interface with the following details:

- Activities:** Text Editor
- Date and Time:** Feb 17 18:05
- File Path:** la4\_p2.py (~/Desktop)
- Buttons:** Save, Full Screen, Minimize, Close

The code in the notebook is as follows:

```
67 # Function to extract text from image using pytesseract
68 def extract_text_from_image(image_content):
69     try:
70         image = Image.open(BytesIO(image_content))
71         text = pytesseract.image_to_string(image)
72         return text
73     except Exception as e:
74         print(f'Error extracting text from image: {e}')
75         return None
76 # Function to scrape the image and extract text for each post
77 def scrape_and_extract_text(row):
78     post_link = row['post_link']
79     time.sleep(1)
80
81     # Send a request to the post link and parse the HTML content
82     response = requests.get(post_link)
83     soup = BeautifulSoup(response.text, 'html.parser')
84
85     # Find the image with id 'post-image'
86     post_image = soup.find('img', {'id': 'post-image'})
87
88     if post_image:
89         # Get the image source URL
90         image_url = post_image.get('src')
91
92         # Fetch the image content directly
93         image_response = requests.get(image_url)
94         image_content = image_response.content
95         # print(image_content)
96
97         # Extract text from the image content using pytesseract
98         text_from_image = extract_text_from_image(image_content)
99
100        # Return the extracted text
101        return str(text_from_image)
102
103    return None
```

The screenshot shows a Linux desktop environment with a terminal window open. The terminal window has a title bar "Text Editor" and a status bar indicating "Feb 17 18:06". The terminal content is a Python script named "la4\_p2.py" located on the desktop. The script uses Selenium to scroll a webpage and update a database. It includes imports for time and webdriver, defines a function to update the database, and a main loop that scrolls until a time limit is reached.

```
106
107 def update_database(interval_minutes):
108     #while True:
109     #    print("Fetching the data.....")
110
111     # Set up the Selenium WebDriver
112     driver = webdriver.Chrome() # You can configure the driver as needed
113
114     # URL of the webpage
115     url = "https://www.reddit.com/r/tech/"
116
117     # Open the webpage with Selenium
118     driver.get(url)
119
120     # Initialize variables for scrolling
121     scroll_js = "window.scrollTo(0, document.body.scrollHeight);"
122     scroll_interval = 1 # Time interval between scrolls (in seconds)
123     scroll_attempts = 0
124     time_limit = 10 # Time limit for scrolling in seconds
125
126     # Get the start time
127     start_time = time.time()
128
129     # Scroll down the webpage until the time limit is reached
130     while True:
131         driver.execute_script(scroll_js)
132         time.sleep(scroll_interval)
133         scroll_attempts += 1
134
135         # Calculate elapsed time
136         elapsed_time = time.time() - start_time
137
138         # Check if the time limit is reached
139         if elapsed_time >= time_limit:
140             break
141
142     # Get the updated page source after scrolling
143     page_source = driver.page_source
```

Activities Text Editor Feb 17 18:09 la4\_p2.py ~/Desktop

```
144 # Parse the HTML content using BeautifulSoup
145 soup = BeautifulSoup(page_source, "html.parser")
146
147 # Initialize a list to store post names
148 post_names_tech = []
149 post_id_tech = []
150
151 for a in soup.find_all('a', href=True):
152     href = a['href']
153     if href.startswith('/r/tech/comments/'):
154         # Split the URL by "/" and get the last part
155         url_parts = href.split("/")
156         if len(url_parts) > 3:
157             comment_part = url_parts[5] # Get the part after "/r/tech/comments/"
158             id_part = url_parts[4]
159             if comment_part not in post_names_tech:
160                 post_names_tech.append(comment_part)
161                 post_id_tech.append(id_part)
162
163
164 # Return the number of posts
165 num_posts = len(post_names_tech)
166 print("Number of Posts:", num_posts)
167
168 # Close the WebDriver
169 driver.quit()
170 print("Data is been successfully fetched from the Web....")
171 print("Preprocessing data....")
172
173 timestamp_elements = soup.find_all(attrs={"created-timestamp": True})
174
175 # Extract and print the attribute values
176 timestamps = [element["created-timestamp"] for element in timestamp_elements]
177
178 post_links = [f"https://www.reddit.com/r/tech/comments/{post_id}/{name}/" for post_id, name in zip(post_id_tech,
179 post_names_tech)]
180 post_title_elements = soup.find_all('a', id=lambda x: x and x.startswith('post-title-'))
181 post_titles = [element.text.strip() for element in post_title_elements]
```

Python 2 Tab Width: 8 Ln 410, Col 35 INS

Activities Text Editor Feb 17 18:10 la4\_p2.py ~/Desktop

```
183
184     rake_nltk_var = Rake()
185     keyword_lst = []
186     for title in post_titles:
187         rake_nltk_var.extract_keywords_from_text(title)
188         keyword_extracted = rake_nltk_var.get_ranked_phrases()
189         keyword_lst.append(keyword_extracted)
190
191
192     data = [{"post_id_tech": post_id, 'formatted_post_title': title, 'formatted_timestamps': timestamp, 'post_link': link,
193 'keyword': keyword}
194         for post_id, title, timestamp, link, keyword in zip(post_id_tech, post_titles, timestamps, post_links,
195         keyword_lst)]
196
197     # Define CSV file path
198     csv_file = 'posts.csv'
199     # Write data to CSV file
200     with open(csv_file, 'w', newline='', encoding='utf-8') as file:
201         fieldnames = ['post_id_tech', 'formatted_post_title', 'formatted_timestamps', 'post_link', 'keyword']
202         writer = csv.DictWriter(file, fieldnames=fieldnames)
203         writer.writeheader()
204         writer.writerows(data)
205
206
207     #print(f"CSV file '{csv_file}' has been created with the data.")
208
209
210     df = pd.read_csv('posts.csv')
211     df['post_id_tech'] = df['post_id_tech'].apply(hash_post_id)
212     df['formatted_post_title'] = df['formatted_post_title'].apply(clean_html_special_chars)
213
214
215     # Apply the function to each row and create a new column 'extracted_text'
216     df['extracted_text'] = df.apply(scrape_and_extract_text, axis=1)
217     #df['formatted_extracted_text'] = df['extracted_text'].apply(clean_html_special_chars)
218     # Create a program and attach shaders
219     df['keyword'] = df['keyword'].apply(string_to_list)
220
221     # Apply cleaning function to keyword column
```

Python 2 Tab Width: 8 Ln 169, Col 30 INS

Activities Text Editor Feb 17 18:11

```
la4_p2.py
~/Desktop
```

Save ⌂ ⌄ ⌅ ⌁

```
218 # Apply cleaning function to keyword column
219 df['keyword'] = df['keyword'].apply(clean_keywords)
220
221 keywords_list = []
222
223 for text in df['extracted_text']:
224     rake_nltk_var.extract_keywords_from_text(str(text))
225     keywords = rake_nltk_var.get_ranked_phrases()
226     keywords_list.append(str(keywords))
227
228 # Add the list of keywords to a new column 'image_keywords'
229 df['image_keywords'] = keywords_list
230
231 df['image_keywords'] = df['image_keywords'].apply(string_to_list)
232 df['image_keywords'] = df['image_keywords'].apply(clean_keywords)
233 # Merge short sentences from the list into a long string, as TfidfVectorizer requires string input
234 df['keyword_joined'] = df['keyword'].apply(lambda x: ' '.join(x)) + ' ' + df['image_keywords'].apply(lambda x: ' '.join(x))
235
236 # Initialize TFIDF vectorizer
237 tfidf = TfidfVectorizer()
238
239 # Applying TFIDF Vectorizer to Keyword_Joined Column
240 tfidf_matrix = tfidf.fit_transform(df['keyword_joined'])
241
242 # Retrieve the index of the maximum TFIDF value for each document
243 max_tfidf_indices = tfidf_matrix.argmax(axis=1)
244
245 # Extract the word corresponding to the maximum TFIDF value of each row from the feature names of the TFIDF vectorizer
246 topics = [tfidf.get_feature_names_out()[max_tfidf_indices[i, 0]] for i in range(tfidf_matrix.shape[0])]
247
248 # Add the extracted keywords to the DataFrame
249 df['topic'] = topics
250
251 # Apply the function to each row and create a new column 'extracted_text'
252 df['extracted_text'] = df.apply(scrape_and_extract_text, axis=1)
253
254
255
```

Python 2 Tab Width: 8 Ln 169, Col 30 INS

Activities Text Editor Feb 17 18:11

```
la4_p2.py
~/Desktop
```

Save ⌂ ⌄ ⌅ ⌁

```
256 df.to_csv('processed_data.csv', index=False)
257
258 print(f'Data has been successfully preprocessed!!!')
259 display_clusters()
260
261 #Insert Records to DB
262
263
264 for index, row in df.iterrows():
265     post_id = row['post_id_tech']
266     select_query = f"SELECT * FROM Posts WHERE post_id = '{post_id}'"
267     cursor.execute(select_query)
268     existing_record = cursor.fetchone()
269     if existing_record:
270         # If the record exists, update it
271         update_query = f"UPDATE Posts SET post_title = '{row['formatted_post_title']}', extract_text_from_image = '{row['extracted_text']}', keywords = '{row['keyword_joined']}' , topic = '{row['topic']}' WHERE post_id = '{post_id}'"
272         cursor.execute(update_query)
273     else:
274         # If the record doesn't exist, insert a new one
275         insert_query = f"INSERT INTO Posts (post_id, post_title, extract_text_from_image, keywords, topic) VALUES ('{post_id}', '{row['formatted_post_title']}',' {row['extracted_text']}',' {row['keyword_joined']}',' {row['topic']}')"
276         cursor.execute(insert_query)
277         #print("New record with post_id {} inserted.".format(post_id))
278
279     # Commit the changes
280     connection.commit()
281
282 print("Values are successfully updated in the database.")
283 time.sleep(60 * interval_minutes) # Convert minutes to seconds
284
285
286 def display_clusters():
287     # Display clusters and messages
288     # Load the CSV file
289     global kmeans, model,X_r
290     df = pd.read_csv('processed_data.csv')
291
292     # Assuming 'formatted_post_title' is the column with text data
```

Python 2 Tab Width: 8 Ln 169, Col 30 INS

Activities Text Editor Feb 17 18:12 \*la4\_p2.py -/Desktop Save

```
291 # Assuming 'formatted_post_title' is the column with text data
292 data = df['formatted_post_title'].tolist()
293
294 # Preprocess the documents and create TaggedDocuments, using post_id_tech as tags if needed
295 tagged_data = [TaggedDocument(words=word_tokenize(doc.lower()), tags=[str(i)]) for i, doc in enumerate(data)]
296
297 # Train the Doc2Vec model (adjust parameters as needed)
298 #model = Doc2Vec(vector_size=20, min_count=2, epochs=50)
299 #model.build_vocab(tagged_data)
300 model.train(tagged_data, total_examples=model.corpus_count, epochs=model.epochs)
301
302 # Infer vectors for the original dataset (or any new data)
303 document_vectors = [model.infer_vector(word_tokenize(doc.lower())) for doc in data]
304
305 # Convert Doc to numpy array
306 X = np.array(document_vectors)
307
308 # Set up k means classifier
309 #kmeans = KMeans(n_clusters=k, random_state=42)
310 kmeans.fit(X)
311
312 # Set up labels
313 labels = kmeans.labels_
314
315 # Attach Labels
316 clustered_data = pd.DataFrame(data, columns=['document'])
317 clustered_data['cluster'] = labels
318 silhouette_avg = silhouette_score(X, labels)
319 print(f"Silhouette Score: {silhouette_avg}")
320
321 # Print Cluster
322 print(clustered_data)
323
324 # Clustering
325 kmeans = KMeans(n_clusters=5, random_state=42)
326 kmeans.fit(X)
327 labels = kmeans.labels_
328
```

Bracket match found on line: 307 Python 2 Tab Width: 8 Ln 307, Col 35 INS

Activities Text Editor Feb 17 18:13 \*la4\_p2.py -/Desktop Save

```
330 # Deploy PCA
331 pca = PCA(n_components=2)
332 X_r = pca.fit_transform(X)
333
334 # Create matplotlib graphs
335 plt.figure(figsize=(12, 8))
336
337 for i in range(5):
338     plt.scatter(X_r[labels == i, 0], X_r[labels == i, 1], label=f'Cluster {i}')
339
340 plt.legend()
341
342 plt.title('K-Means Clustering')
343 plt.xlabel('PCA Feature 1')
344 plt.ylabel('PCA Feature 2')
345 plt.show()
346
347 # # Export csv
348 clustered_data.to_csv('clustered_data.csv', index=False)
349 #return X_r
350
351 def input_thread(interval_minutes):
352     global kmeans, model, X_r
353     try:
354         while True:
355             user_input = input().strip()
356
357             if user_input.lower() == "quit":
358                 if connection.is_connected():
359                     cursor.close()
360                     connection.close()
361                 os._exit(0)
362             else:
363                 # Assume user input is a message or keywords
364                 # Preprocess the input text
365                 #X_r = display_clusters()
366                 labels = kmeans.labels_
367                 clustered_data = pd.read_csv('clustered_data.csv')
368                 cleaned_input = clean_html_special_chars(user_input)
```

Python 2 Tab Width: 8 Ln 307, Col 35 INS

Activities Text Editor Feb 17 18:14

```

367     clustered_data = pd.read_csv('clustered_data.csv')
368     cleaned_input = clean_html_special_chars(user_input)
369
370     tokenized_input = word_tokenize(cleaned_input.lower())
371     input_vector = model.infer_vector(tokenized_input)
372     input_vector = input_vector.astype(np.float64)
373     kmeans.cluster_centers_ = kmeans.cluster_centers_.astype(np.float64)
374     # Predict the cluster for the input
375     input_cluster = kmeans.predict([input_vector])[0]
376
377
378     # Filter DataFrame to get messages from the selected cluster
379     selected_cluster_data = clustered_data[clustered_data['cluster'] == input_cluster]['document']
380
381     # Display messages from the selected cluster
382     print("Predicted Cluster:", input_cluster)
383     print(f"Messages from Cluster {input_cluster}:\n")
384     for message in selected_cluster_data:
385         print(message)
386
387     # Display graphical representation
388     plt.figure(figsize=(12, 8))
389     for i in range(5):
390         plt.scatter(X_r[labeled == i, 0], X_r[labeled == i, 1], label=f'Cluster {i}')
391         plt.scatter(input_vector[0], input_vector[1], marker='x', s=100, color='red', label='Input')
392     plt.legend()
393     plt.title('K-Means Clustering')
394     plt.xlabel('PCA Feature 1')
395     plt.ylabel('PCA Feature 2')
396     plt.show()
397     print("Input processed successfully!")
398
399 except KeyboardInterrupt:
400     pass
401
402 if __name__ == "__main__":
403     import sys
404     db_credentials = {
405         'host': 'localhost',
406         'user': 'root',
407         'password': 'your_password',
408         'database': 'post_db'
409     }
410
411     # Connect to MySQL database
412     connection = mysql.connector.connect(**db_credentials)
413     cursor = connection.cursor()
414     nltk.download('stopwords')
415     nltk.download('punkt')
416
417     if len(sys.argv) != 2:
418         print("Usage: python filename.py <interval_minutes>")
419         sys.exit(1)
420
421     try:
422         interval_minutes = int(sys.argv[1])
423         print(interval_minutes)
424
425         # Start the background thread for database updates
426         update_thread = threading.Thread(target=update_database, args=(interval_minutes,))
427         update_thread.start()
428
429         # Start the input thread for user commands
430         input_thread = threading.Thread(target=input_thread, args=(interval_minutes,))
431         input_thread.start()
432
433         # Wait for both threads to finish
434         update_thread.join()
435         input_thread.join()
436
437     except ValueError:
438         print("Error: Interval must be a valid integer.")
439         sys.exit(1)
440
441

```

Python 2 Tab Width: 8 Ln 307, Col 35 INS

Activities Text Editor Feb 17 18:15

```

403     import sys
404     db_credentials = {
405         'host': 'localhost',
406         'user': 'root',
407         'password': 'your_password',
408         'database': 'post_db'
409     }
410
411     # Connect to MySQL database
412     connection = mysql.connector.connect(**db_credentials)
413     cursor = connection.cursor()
414     nltk.download('stopwords')
415     nltk.download('punkt')
416
417     if len(sys.argv) != 2:
418         print("Usage: python filename.py <interval_minutes>")
419         sys.exit(1)
420
421     try:
422         interval_minutes = int(sys.argv[1])
423         print(interval_minutes)
424
425         # Start the background thread for database updates
426         update_thread = threading.Thread(target=update_database, args=(interval_minutes,))
427         update_thread.start()
428
429         # Start the input thread for user commands
430         input_thread = threading.Thread(target=input_thread, args=(interval_minutes,))
431         input_thread.start()
432
433         # Wait for both threads to finish
434         update_thread.join()
435         input_thread.join()
436
437     except ValueError:
438         print("Error: Interval must be a valid integer.")
439         sys.exit(1)
440
441

```

Python 2 Tab Width: 8 Ln 307, Col 35 INS

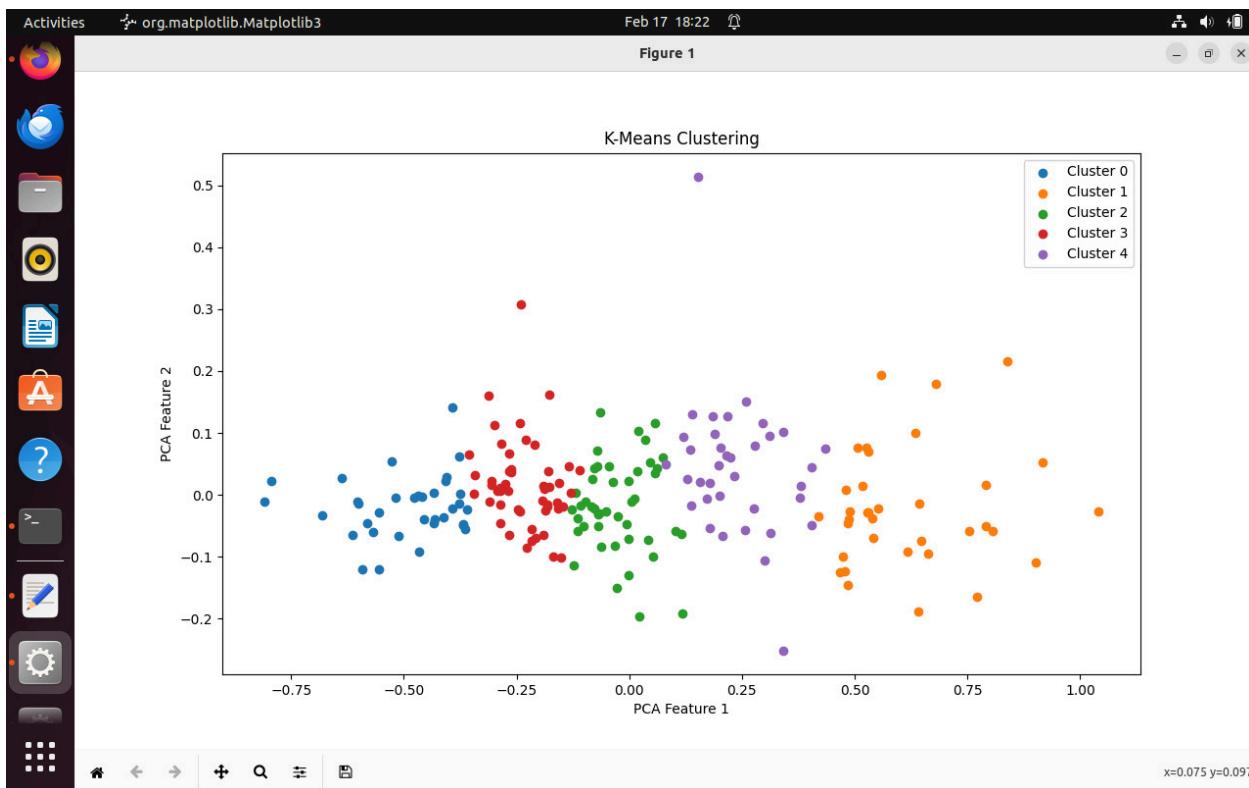
The Linux terminal output of the above script is shown below,

Activities Terminal Feb 17 18:24 vdharmar@Ubuntu2: ~/Desktop

```
vdharmar@Ubuntu2:~/Desktop$ python3 la4_p2.py 4
/home/vdharmar/Desktop/la4_p2.py:4: DeprecationWarning:
Pyarrow will become a required dependency of pandas in the next major release of pandas (pandas 3.0),
(to allow more performant data types, such as the Arrow string type, and better interoperability with other libraries)
but was not found to be installed on your system.
If this would cause problems for you,
please provide us feedback at https://github.com/pandas-dev/pandas/issues/54466

import pandas as pd
[nltk_data] Downloading package stopwords to
[nltk_data]      /home/vdharmar/nltk_data...
[nltk_data]      Package stopwords is already up-to-date!
[nltk_data] Downloading package punkt to /home/vdharmar/nltk_data...
[nltk_data]      Package punkt is already up-to-date!
4
Fetching the data.....
Number of Posts: 203
Data is been successfully fetched from the Web.....
Preprocessing data.....
Data has been successfully preprocessed!!!
Silhouette Score: 0.1739136278629303
document  cluster
0  Microscopic robots could soon float inside you...    2
1  Humanitys remaining timeline It looks more lik...    0
2  Nvidia reveals its Eos supercomputer for AI pr...    4
3  New Solid Electrolyte Matches Liquid Performan...    3
4  New chip opens door to AI computing at light s...    0
...
198 MIT tests new ingestible sensor that records y...    3
199 How tiny hinges bend the infectionspreading sp...    1
200 Running thousands of LLMs on one GPU is now po...    0
201 Scientists 3D print a robotic hand with humanl...    4
202 Swallowable device tracking vital signs inside...    1
[203 rows x 2 columns]
/home/vdharmar/Desktop/la4_p2.py:338: UserWarning: Starting a Matplotlib GUI outside of the main thread will likely fail.
plt.figure(figsize=(12, 8))
/home/vdharmar/Desktop/la4_p2.py:348: UserWarning: Starting a Matplotlib GUI outside of the main thread will likely fail.
plt.show()
Values are successfully updated in the database.
```

The below screenshot shows the graphical representation of the Clusters,

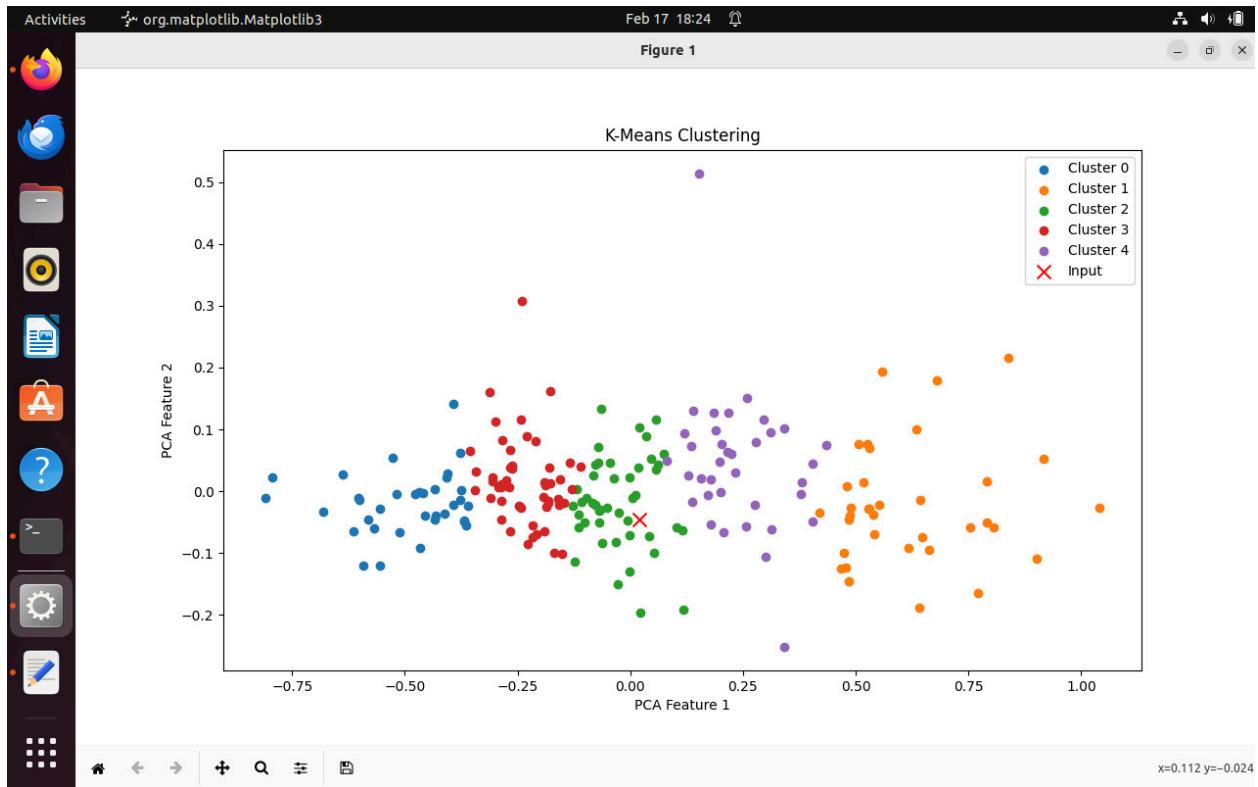


Activities Terminal Feb 17 18:25

```
NLP is awesome!!!
Predicted Cluster: 0
Messages from Cluster 0:

Humanity's remaining timeline It looks more like five years than 50 meet the neoluddites warning of an AI apocalypse
New chip opens door to AI computing at light speed
The existence of a new kind of magnetism has been confirmed
Open AI can now create video from text
NASA Puts NextGen Exoplanet Imaging Technology to the Test
28ton 12megawatt tidal kite is now exporting power to the grid
Shining Laser Light on Glass Creates a Solar Cell New energy and sensor applications could await
A mouse for your mouth New device allows users to scroll with their tongues
Inhalable sensors could enable early lung cancer detection
AI can now master your music and it does shockingly well
CERN unleashes powerful robotdog to inspect nuclear radiation zones
Infrared Sensors Can Now Peer Around Corners Nonlineofsight imaging is no longer restricted to visible wavelengths of light
Can a Brain Implant Treat Addiction Some experts tout deep brain stimulation as a lifeline for people struggling with opioid use others question the hype
The New New Transistor In power electronics aluminum nitride could overtake two powerhouses that only recently bested silicon
Ureasepowered nanobots for radionuclide bladder cancer therapy
Gene Therapy Allows an 11YearOld Boy to Hear for the First Time
Flying Kites Deliver ContainerSize Power Generation Automated windenergy system brings portable renewables offgrid
Experts craft lifesaving robot medics for triage in highrisk places
Mesh Wearables Meld Micro Sensors and LoRa Smarts Say goodbye to boxandstrap wearable tech of old
Biologging method for animal identification using dissolvable microneedle arrays prepared by customisable moulds
How Microsoft found a potential new battery material using AI
ENose Sniffs Out Coffee Varieties Nearly Perfectly Like Shazam but for java the tech can help quantify coffee signatures
10x Stronger Than Kevlar Amorphous Silicon Carbide Could Revolutionize Material Science
Qualcomms firstofitskind SoC can handle infotainment and ADAS on one chip
Airsteered X65 jet from DARPA enters manufacturing phase
The first EV with a lithiumfree sodium battery hits the road in January
This Etongue mimicks human taste and recommends wine with different foods Scientists pave the way for new culinary frontiers
Modernas mRNA cancer vaccine works even better than thought
Doughnut plasma and 100 millionC cores How scientists could soon make nuclear fusion a reality
Texas plant to generate 10 MW of power with desksized turbines
Automatic bike transmission concept is wild and spiky and could be a big shift
New type of geothermal power plant powers data centers in the desert Pilot plant in Nevada uses tech from fracking to generate power in arid landscape
New Neurotech Eschews Electricity for Ultrasound Companies team up to use ultrasoundonchip tech to develop a brain computer interface
```

The below screenshot shows the graphical representation of the clusters and the position of the input,



```

vdharmar@Ubuntu2: ~/Desktop$
New Neurotech Eschews Electricity for Ultrasound Companies team up to use ultrasoundonchip tech to develop a bra
in computer interface
Scientists find way to wipe a cells memory to better reprogram it as a stem cell
Running thousands of LLMs on one GPU is now possible with SLoRA
/home/vdharmar/Desktop/la4_p2.py:391: UserWarning: Starting a Matplotlib GUI outside of the main thread will lik
ely fail.
plt.figure(figsize=(12, 8))
/home/vdharmar/Desktop/la4_p2.py:399: UserWarning: Starting a Matplotlib GUI outside of the main thread will lik
ely fail.
plt.show()
Input processed successfully!
quit
vdharmar@Ubuntu2:~/Desktop$ 
```

The screenshots below show the output of the CSVs generated,

Activities Text Editor Feb 17 18:32

Open Save

posts.csv

1 post\_id,tech,formatted\_post\_title,formatted\_timestamps,post\_link,keyword  
 2 1at99aw,Microscopic robots could soon float inside your liver to fight cancer. Canadian researchers are closing in on a novel approach to treat liver tumours using microrobots in a MRI device.,2024-02-17T18:55:06.569000+0000,https://www.reddit.com/r/tech/comments/1at99aw/microscopic\_robots\_could\_soon\_float\_inside\_your/,[{"microscopic robots could soon float inside", "treat liver tumours using microrobots", "novel approach", "mri device", "fight cancer", "canadian researchers", "liver", "closing"]]  
 3 1asxvnxn,Humanity's remaining timeline? It looks more like five years than 50': meet the neo-luddites warning of an AI apocalypse, 2023-12-08T14:31:02.726000+0000,https://www.reddit.com/r/tech/comments/1asxvnxn/humanitys\_remaining\_timeline\_it\_looks\_more\_like/, [{"humanity", "like five years", "50": "meet", "remaining timeline", "luddites warning", "ai apocalypse", "neo", "looks"}]  
 4 1ataibg,"Nvidia reveals its Eos supercomputer for AI processing sporting 4,608 H100 GPUs | Its the ninth fastest supercomputer in the world",2024-02-17T09:15:38.375000+0000,https://www.reddit.com/r/tech/comments/1ataibg/nvidia\_reveals\_its\_eos\_supercomputer\_for\_ai/,[{"ai processing sporting 4", "608 h100 gpus", "ninth fastest supercomputer", "eos supercomputer", "nvidia reveals", "world"}]  
 5 1asawcw,New Solid Electrolyte Matches Liquid Performance. Better solid electrolytes could result in safer batteries that don't explode.,2024-02-17T19:47:56.082000+0000,https://www.reddit.com/r/tech/comments/1asawcw/new\_solid\_electrolyte\_matches\_liquid\_performance/,[{"new solid electrolyte matches liquid performance", "better solid electrolytes could result", "safer batteries", "solid", "explode"}]  
 6 1asnwmw,New chip opens door to AI computing at light speed,2024-02-16T15:10:43.764000+0000,https://www.reddit.com/r/tech/comments/1asnwmw/new\_chip\_opens\_door\_to\_ai\_computing\_at\_light\_speed/,[{"new chip opens door", "light speed", "ai computing"}]  
 7 1as5xs9,US researchers develop 'unhackable' computer chip that works on light | Computations can be performed at light speed in this chip which is ready for deployment for building AI models immediately.,2024-02-17T00:07:33.274000+0000,https://www.reddit.com/r/tech/comments/1as5xs9/us\_researchers\_develop\_unhackable\_computer\_chip/,[{"building ai models immediately", "us researchers develop", "light speed", "computer chip", "light", "chip", "works", "unhackable", "ready", "performed", "deployment", "computations"}]  
 8 1as7458,"LED glass basketball court to make NBA debut this month, displaying stats and replays | Courts can also display other entertainment content and host potentially any hardcourt sport",2024-02-16T11:02:01.638000+0000,https://www.reddit.com/r/tech/comments/1as7458/led\_glass\_basketball\_court\_to\_make\_nba\_debut\_this/,[{"led glass basketball court", "make nba debut", "host potentially", "hardcourt sport", "entertainment content", "displaying stats", "also display", "replays", "month", "courts"}]  
 9 1arxn30,The existence of a new kind of magnetism has been confirmed,2024-02-16T12:11:52.407000+0000,https://www.reddit.com/r/tech/comments/1arxn30/the\_existence\_of\_a\_new\_kind\_of\_magnetism\_has\_been/,[{"new kind", "magnetism", "existence", "confirmed"}]  
 10 1ardw87,Ultrasound waves spark movement in sleepy sperm by up to 26%,2024-02-16T02:33:40.004000+0000,https://www.reddit.com/r/tech/comments/1ardw87/ultrasound\_waves\_spark\_movement\_in\_sleepy\_sperm/,[{"ultrasound waves spark movement", "sleepy sperm", "26%"}]  
 11 1arxuovo,Open AI can now create video from text,2024-02-15T12:01:22.771000+0000,https://www.reddit.com/r/tech/comments/1arxuovo/open\_ai\_can\_now\_create\_video\_from\_text/,[{"open ai", "create video", "text"}]  
 12 1armyjr,"Surgical robot, MIRA, aces zero-gravity examination in space | SpaceMIRA, a miniature surgical robot, excels in zero gravity, promising revolutionary advances in remote surgery and healthcare accessibility worldwide.",2024-02-16T02:44:39.568000+0000,https://www.reddit.com/r/tech/comments/1armyjr/surgical\_robot\_mira\_aces\_zerogravity\_examination/,[{"promising revolutionary advances", "healthcare accessibility worldwide"}]

CSV Tab Width: 8 Ln 1, Col 1 INS

Activities Text Editor Feb 17 18:32

Open Save

processed\_data.csv

1 post\_id,tech,formatted\_post\_title,formatted\_timestamps,post\_link,keyword,extracted\_text,image\_keywords,keyword\_joined,topic  
 2 e99cfb319cf90abfc79e1e96ddd98bb9fdaf95e89926aff141484fa5a03d2d,Microscopic robots could soon float inside your liver to fight cancer Canadian researchers are closing in on a novel approach to treat liver tumours using microrobots in a MRI device,2024-02-17T18:55:06.569000+0000,https://www.reddit.com/r/tech/comments/1at99aw/microscopic\_robots\_could\_soon\_float\_inside\_your/,[{"microscopic", "robots", "could", "soon", "float", "inside", "treat", "liver", "tumours", "using", "microrobots", "novel", "approach", "mri", "device", "fight", "cancer", "canadian", "researchers", "liver", "closing"}]  
 3 [],microscopic robots could soon float inside treat liver tumours using microrobots novel approach mri device fight cancer canadian researchers liver closing ,liver  
 4 f038f01ab9dad530c1058ebc82e0e43b364a52de84d18a5cb616629b1a31e2c,Humanity's remaining timeline It looks more like five years than 50 meet the neoluddites warning of an AI apocalypse, 2023-12-08T14:31:02.726000+0000,https://www.reddit.com/r/tech/comments/1asxvnxn/humanitys\_remaining\_timeline\_it\_looks\_more\_like/, [{"humanity", "like", "five", "years", "50", "meet", "remaining timeline", "luddites warning", "ai", "apocalypse", "neo", "looks"}], [{"rea", "ta", "apocalypse"}]  
 5 cic376a8746ebedf5ad7e04fff12c04847a70b12da88a56bac87b7bef2661918,Nvidia reveals its Eos supercomputer for AI processing sporting 4608 H100 GPUs Its the ninth fastest supercomputer in the world,2024-02-17T09:15:38.375000+0000,https://www.reddit.com/r/tech/comments/1ataibg/nvidia\_reveals\_its\_eos\_supercomputer\_for\_ai/,[{"ai", "processing", "sporting", "4", "608", "h100", "gpus", "ninth", "fastest", "supercomputer", "eos", "supercomputer", "nvidia", "reveals", "world"}], [{"ai processing sporting 4 608 h100 gpus ninth fastest supercomputer eos supercomputer nvidia reveals world ,supercomputer"}]  
 6 9620d192e43b03dababe1d95d961a19cb0ef22b59f177499a26248e49673,New Solid Electrolyte Matches Liquid Performance Better solid electrolytes could result in safer batteries that dont explode,2024-02-17T19:47:56.082000+0000,https://www.reddit.com/r/tech/comments/1asawcw/new\_solid\_electrolyte\_matches\_liquid\_performance/,[{"new", "solid", "electrolytes", "could", "result", "safer", "batteries", "explode"}]  
 7 [],new solid electrolyte matches liquid performance better solid electrolytes could result safer batteries explode ,solid  
 8 d41bb4e64d83e4350bac0f4114de4237972d0daic97cf5f418ae218e2cceee43b,New chip opens door to AI computing at light speed,2024-02-16T15:10:43.764000+0000,https://www.reddit.com/r/tech/comments/1asnwmw/new\_chip\_opens\_door\_to\_ai\_computing\_at\_light\_speed/,[{"new", "chip", "opens", "door", "light", "speed", "ai", "computing"}], [{"new chip opens door light speed ai computing ,opens"}]  
 10 71ce618af6f73a62dc9e1fcae9681d0ef43f142a8c5e8d7c44c422558dcf76e,US researchers develop unhackable computer chip that works on light | Computations can be performed at light speed in this chip which is ready for deployment for building AI models immediately, 2024-02-17T00:07:33.274000+0000,https://www.reddit.com/r/tech/comments/1as5xs9/us\_researchers\_develop\_unhackable\_computer\_chip/,[{"building", "ai", "models", "immediately", "us", "researchers", "develop", "light", "speed", "computer", "chip", "light", "chip", "works", "unhackable", "ready", "performed", "deployment", "computations"}]  
 11 [],building ai models immediately us researchers develop light speed computer chip light chip works unhackable ready performed deployment computations ,chip  
 12 4930afb71b40a0930330d437f7d4b579c6e56bd3233e6a7063064fb11a53121,LED glass basketball court to make NBA debut this month displaying stats and replays | Courts can also display other entertainment content and host potentially any hardcourt sport,

CSV Tab Width: 8 Ln 1, Col 2 INS

Activities Text Editor Feb 17 18:31

clustered\_data.csv ~/Desktop Save

```

1 document,cluster
2 Microscopic robots could soon float inside your liver to fight cancer Canadian researchers are closing in on a novel approach to treat liver tumours using microrobots in a MRI device,2
3 Humanitys remaining timeline It looks more like five years than 50 meet the neoluddites warning of an AI apocalypse,0
4 Nvidia reveals its Eos supercomputer for AI processing sporting 4608 H100 GPUs Its the ninth fastest supercomputer in the world,4
5 New Solid Electrolyte Matches Liquid Performance Better solid electrolytes could result in safer batteries that dont explode,3
6 New chip opens door to AI computing at light speed,0
7 US researchers develop unhackable computer chip that works on light Computations can be performed at light speed in this chip which is ready for deployment for building AI models immediately,4
8 LED glass basketball court to make NBA debut this month displaying stats and replays Courts can also display other entertainment content and host potentially any hardcourt sport,2
9 The existence of a new kind of magnetism has been confirmed,0
10 Ultrasound waves spark movement in sleepy sperm by up to 266,2
11 Open AI can now create video from text,0
12 Surgical robot MIRA aces zero gravity examination in space SpaceMIRA a miniature surgical robot excels in zero gravity promising revolutionary advances in remote surgery and healthcare accessibility worldwide,2
13 Magnetic robotic catheter devised to efficiently treat ischemic strokes The cuttingedge device with helical surface and rotational motion can navigate the vascular system efficiently targeting diseased blood vessels with precision,2
14 NASA Puts NextGen ExoplanetImaging Technology to the Test,0
15 OpenSource Security Chip Released The first commercial chip based on the OpenTitan hardware security design has hit the market,3
16 Replacement cartilage can grow in any shape with 3Dprinted spheroids,3
17 Chinese researchers develop calciumbased battery that lasts 700 cycles Stable in the air at room temperature the calciumoxygen battery can be transformed into flexible fibers for nextgen wearables,2
18 OpenAI CEO warns that societal misalignments could make artificial intelligence dangerous,3
19 28ton 12megawatt tidal kite is now exporting power to the grid,0
20 Shining Laser Light on Glass Creates a Solar Cell New energy and sensor applications could await,0
21 Tokyo scientists create nanoscrolls for nextgen tech Researchers achieved a major breakthrough by crafting nanoscrolls using Janus nanosheets This innovation unlocks doors to exciting possibilities in catalysis optics and clean energy,4
22 Deep Space Station 13 at NASAs Goldstone complex in California part of the agencys Deep Space Network is an experimental antenna that has been retrofitted with an optical terminal In a first this proof of concept received both radio frequency laser signals from deep space at the same time,4
23 A mouse for your mouth New device allows users to scroll with their tongues,0
24 This ultrasound sticker senses changing stiffness of deep internal organs The sticky wearable sensor could help identify early signs of acute liver failure,2
25 In a First a Prosthetic Limb Can Sense Temperature Like a Living Hand,3
26 Alternate qubit design does error correction in hardware Earlystage technology has the potential to cut qubits needed for useful computers,4
27 Scientists develop a lowcost device to make cell therapy safer A plastic microfluidic chip can remove some risky cells that could potentially become tumors before they are implanted in a patient.4

```

CSV Tab Width: 8 Ln 1, Col 1 INS

The below screenshots show the output records stored in the database,

Activities Terminal Feb 17 18:34

vdharmar@Ubuntu2: ~/Desktop

```

mysql> USE posts_db;
ERROR 1049 (42000): Unknown database 'posts_db'
mysql> USE post_db;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A
Database changed
mysql> SELECT * FROM Posts;
+-----+-----+-----+
| id   | post_id          | post_title           |
|      |                  |                      |
|      | keywords          | extract_text_from_image |
|      | topic             |                         |
+-----+-----+-----+
| 244 | e99cfb319cf90abfc79e1e96dd98bb9fdaf95e89926aff141484fa5a603d2d | Microscopic robots could soon float inside your liver to fight cancer Canadian researchers are closing in on a novel approach to treat liver tumours using microrobots in a MRI device |
|      | microscopic robots could soon float inside treat |
|      | liver tumours using microrobots novel approach mri device fight cancer canadian researchers liver closing |
| 245 | f038f01ab9dad530c1058ebc82e0e43b3648a52de84d18a5cb616629b1a31e2c | Humanitys remaining timeline It looks more like five years than 50 meet the neoluddites warning of an AI apocalypse |
|      | None |
|      | humanity like five years 50 meet remaining timeline luddites warning a |
|      | apocalypse neo looks rea ta |
|      | apocalypse |
+-----+-----+-----+

```

Activities Terminal Feb 17 18:34 vdharmar@Ubuntu2: ~/Desktop

```
| mit tests new ingestible sensor records intesti
| breathing | | tiny hinges bend hinges could spreading spikes
| 443 | 9264efb041219a4bfe01a7ed7e5cf02089206fdb81820927c9ec5490b3663b17 | How tiny hinges bend the infectionspreading spikes of a c
oronavirus Disabling those hinges could be a good strategy for designing vaccines and treatments against a broad range of coronaviru
s infections including COVID19 |
including covid good strategy designing vaccines broad range coronavirus infections coronavirus treatments infection disabling 19
| coronavirus | |
| 444 | c804213ea8abf481ba0f0c5154cb989254b34426f02cf6e683f54faa53ccc9d7 | Running thousands of LLMs on one GPU is now possible with
SLoRA |
| running thousands one gpu possible lora llms |
| gpu | |
| 445 | 459187dbc561665ef394498d5e6b3af7c2421a0d170a6175ab08215aab173e28 | Scientists 3D print a robotic hand with humanlike bones a
nd tendons As a layer is printed an optical scan IDs flaws and corrects them in the next layer |
| optical scan ids flaws scientists 3d print robo
tic hand like bones next layer layer tendons printed human corrects |
| layer | |
| 446 | 972a3daf7ce6ab0b5d5f2f71d2228f4ce5daffd0a29356e6236a32e046aca707 | Swallowable device tracking vital signs inside the body i
n human trial The device is part of a growing field of ingestible devices that can perform various functions inside the body |
| swallowable device tracking vital signs inside |
| perform various functions inside ingestible devices human trial growing field device part body body |
| inside | |
+-----+
203 rows in set (0.01 sec)
```