MATES Computer Science

Senior Capstone Project Bi-Weekly Progress Report

Project Title	MorningBread
Team Members	Max Bradshaw, Jake Dorick
Dates Covered by Report	March 9th - March 22nd
Link to Github	https://github.com/maxbshaw17/MorningBread

1. **Summary of Project** (Provide a one paragraph summary of your project. You can largely copy/paste this from one progress report to the next, unless there are significant changes.)

To provide a personalized morning financial report to our users by utilizing a web scraping tool for financial website articles to find the general topics of the articles. In doing so, we are creating and designing a website to give the public a general grasp of financial news for the day, over the course of the day with hourly, highly summarized news headlines/stock tickers. An account system will allow users to follow certain companies and manage tickers.

2. **Summary of Progress this Period** (Provide a high-level, one paragraph overview of what was accomplished this progress period collectively by the team.)

Max -

I implemented the NLP text clustering using the DBSCAN density-based algorithm. I also worked on the SQL injection code and cleaned up the scraping methods again, as well as completely rewriting the yahoo scraping function again (in progress). Finally, I structured the back-end databases and created a test database for Jake to work off of for now.

Jake -

Creating the background for what is to come to be dynamically-added soon was my main priority. The design of how the articles and tickers will be laid out on the front page was my top priority. The structure of my code is bound to change as the JavaScript is implemented, allowing divs to be dynamically-added as they are fetched from the API, but the CSS will remain the same.

3. **Detailed Progress this Period, separated by Team Member** (Provide detailed information on the progress that you made in the reporting weeks. Include screenshots of code, your game or website, etc. Each team member should have a separate subsection covering their accomplishments. Not including screenshots, this section should be 1-2 pages.)

Max -

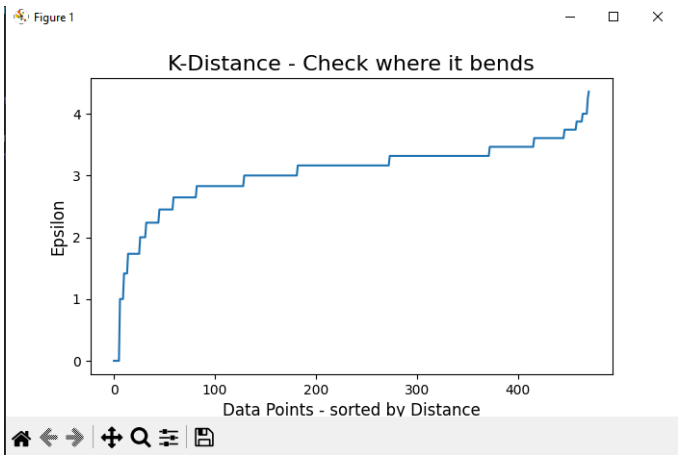
- Wrote functions for ML so I can call in the file that I will be running automatically
    - This took the longest as I tried several models before choosing this one

```python
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.cluster import DBSCAN
from sklearn.neighbors import NearestNeighbors
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import string
from nltk.tokenize import word_tokenize, sent_tokenize
from nltk.corpus import stopwords
from nltk.stem import PorterStemmer

default_stemmer = PorterStemmer()
default_stopwords = stopwords.words('english') # or any other list of your choice

# check for repeats - need to fix on sql side
def remove_repeats(headlines):
    text_no_repeats = []
    for sent in headlines:
        if sent not in text_no_repeats:
            text_no_repeats.append(sent)
    return (text_no_repeats)

def clean_text(text, ): ...
    return text

def clean_text_list(text_list):
    text_list = remove_repeats(text_list)
    return list(map(lambda text: clean_text(text), text_list))

def knn_plot(text_array):
    neigh = NearestNeighbors(n_neighbors=2)
    nbrs = neigh.fit(text_array)
    distances, indices = nbrs.kneighbors(text_array)
    distances = np.sort(distances, axis=0)
    distances = distances[:,1]

    plt.figure(figsize=(7,4))
    plt.plot(distances)
    plt.title('K-Distance - Check where it bends',fontsize=16)
    plt.xlabel('Data Points - sorted by Distance',fontsize=12)
    plt.ylabel('Epsilon',fontsize=12)
    plt.show()

def text_vectorizer(text_list):
    text = remove_repeats(text_list)
    text_cleaned = clean_text_list(text_list)

    cv = CountVectorizer()
    x = cv.fit_transform(text_cleaned)
    return x.toarray(), text

def fit_dbscan_text(text_array, text, ep=1, min_s=2):
    dbscan_opt=DBSCAN(eps = ep,min_samples = min_s)
    dbscan_opt.fit(text_array)
    df_data = {"group": dbscan_opt.labels_, "sentence" : text}
    df = pd.DataFrame(data=df_data)

    return df
```

- Added a duplicate removal call to the database

```
def remove_dupes_sql(table, column):
  try:
    command = f"""DELETE FROM
  {table}
WHERE
  id IN (
    SELECT
      id
    FROM
      (
        SELECT
          id,
          ROW_NUMBER() OVER (
            PARTITION BY {column}
            ORDER BY
              {column}
          ) AS row_num
        FROM
          {table}
      ) t
    WHERE
      row_num > 1
  );"""
```

- Because DBSCAN is based on the multidimensional distance between each vectorized text, I have to fine-tune the distance parameter
- I made a quick function that plots the number of groups vs each epsilon (distance) input
    - 2.5 seems to be the sweet spot



- Script to run ML and example of groupings

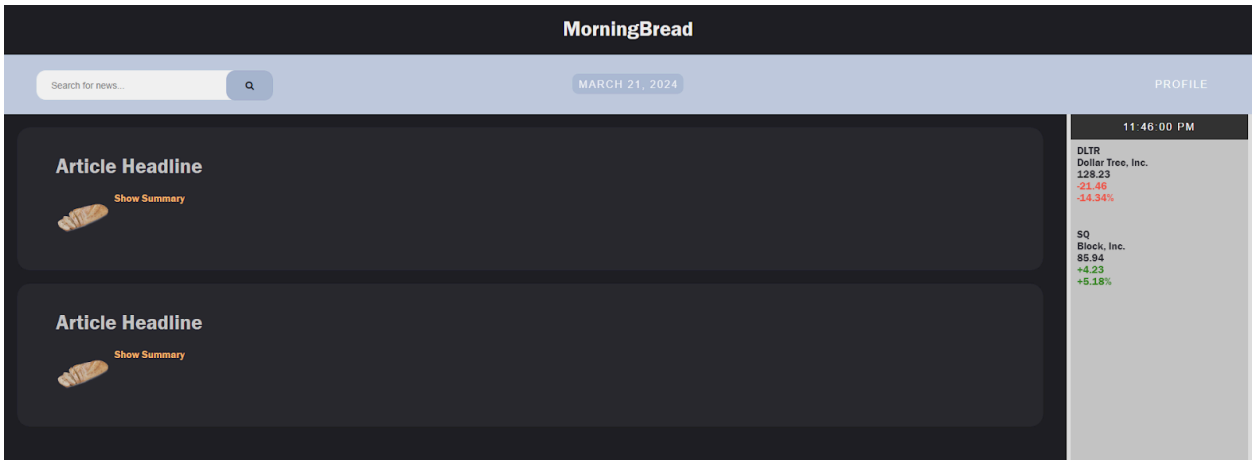
```python
from db_update.db_connector import *
from nlp_algo.nlp_functions import *

headlines = get_all_headlines()

array, sents = text_vectorizer(headlines)


print(knn_plot(array))


fit_df = fit_dbscan_text(array, sents, ep = 2.5, min_s=2)

print(fit_df.sort_values(by='group', ascending=False).head(80).to_string())

#for i in fit_df.sort_values(by='group', ascending=False).index:
#    if fit_df['group'][i] != -1:
#        print(f"{fit_df['group'][i]}, {fit_df['sentence'][i]}")
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

	group	sentence
467	24	OPEC leaves forecast for 2024 oil-demand growth unchanged at 2.2 million barrels a day
388	24	OPEC keeps 2024 oil-demand growth forecast at 2.2 million barrels a day
386	24	OPEC leaves forecast for oil-demand growth unchanged
385	23	The Moneyist on one reader's 'indecent proposal' dilemma
458	23	The Moneyist: 'Is this an indecent proposal?
399	22	Elon Musk questioned by Don Lemon over diversity and content moderation
378	22	Elon Musk discusses ketamine use, questioned by Don Lemon over diversity and content moderation
418	21	Astera Boosts IPO Size, Price Range as AI Stock Rally Churns On
376	21	Astera Labs Boosts IPO Size, Price Range as AI Stocks Rally
375	20	Stock market today: Nasdaq soars to lead stocks higher, Nvidia spurs AI hopes
396	20	Nasdaq soars to lead stocks higher as Nvidia spurs AI hopes
407	19	Nasdaq says issues impacting connectivity resolved
366	19	Nasdaq Says It Is Investigating Issues With Connectivity
401	18	Kuehne in talks on 100-mln-eur loan for Signa Prime, source says
361	18	Signa Prime, Kuehne in Talks on Emergency Loan to Buy Time
381	17	LPP’s stock rebounds 30% after Polish clothing maker denies faking its exit from Russia
348	17	LPP’s stock rebounds 30% after Polish clothes seller denies faking its exit from Russia
339	16	Tesla’s stock rallies with price of Model Y midsize SUV set to increase in April
351	16	Tesla’s stock rallies as prices for Model Y EVs will increase in April
242	15	What 100 years of history shows about periods of extreme market concentration, according to Goldman Sachs
236	15	Here’s what 100 years of history shows about periods of extreme market concentration, according to Goldman Sachs
323	14	U.S. adds more-than-expected 275,000 jobs in February; unemployment rate at 3.9%
250	14	US economy adds 275,000 jobs, unemployment rate rises
315	14	February jobs report: US economy adds 275,000 jobs, unemployment rate hits 3.9%
233	14	U.S. economy adds 275,000 jobs, stronger than expected
228	13	VIEW Feb US payrolls show labor market healthy but not overly tight
308	13	Feb US payrolls show labor market healthy but not overly tight
225	12	Stock market today: US futures rise after February jobs report
231	12	Stock market today: US futures steady after jobs report
207	11	U.S. trade deficit widens more than 5% to its highest level in nine months
198	11	US trade deficit widens in January on imports
192	10	Market hopes are high that big central banks will cut rates around mid-year

-
- You can see that sentences with similar topics are grouped together

Jake -



- The layout for the articles (and adding the areas for the tickers) has been changed
- JavaScript is implemented in both the articles section and the tickers
    - The articles section has a “Show Summary” button that will give the article’s information when clicked
    - This requires reading the button’s status based off of its original status when the page is first loaded

- Above shows the general layout the articles will have for website (probably going to remove the goofy bread icon)
- The tickers on the right are scrolling, they will also be dynamically-added, but that is my second priority
- What’s currently there is just an example of what I hope to implement after I get the articles onto the site
- Time function because that’s important for stocks, as so I’m told

```python
from flask import Flask, jsonify
import mysql.connector

app = Flask(__name__)

# Connect to the MySQL database (Not yet created)
articles_db = mysql.connector.connect(
    host = "mysql-2ed0e70f-morningbread.a.aivencloud.com",
    user = "avnadmin",
    password = "AVNS_-1y1cgAxePfkqdPTpji",
    port = 25747,
    database = "morningbread",
)

c = articles_db.cursor()

@app.route('/articles_tickers_api/articles_api', methods=['GET'])
def get_articles():
    # Query the database for article headlines and summaries
    c.execute("SELECT headline, summary FROM articles")
    articles = c.fetchall()

    # Convert the articles to a list of dictionaries
    article_list = [{'headline': row[0], 'summary': row[1]} for row in articles]

    return jsonify(article_list)

if __name__ == '__main__':
    app.run(debug=True)
```

- I have used a similar concept of the Flask API for my research project (PneumoVision), so I was able to recycle a lot of this code
- articles_db connects to the MySQL database to begin fetching the information that will be added to the website
- I have a similar file to the one above for the tickers despite there being no ticker data in the database yet…

```python
from flask import Flask, jsonify
import mysql.connector

app = Flask(__name__)

# Connect to the MySQL database (Not yet created)
tickers_db = mysql.connector.connect(
    host = "mysql-2ed0e70f-morningbread.a.aivencloud.com",
    user = "avnadmin",
    password = "AVNS_-1y1cgAxePfkqdPTpji",
    port = 25747,
    database = "morningbread"
)

c = tickers_db.cursor()

@app.route('/articles_tickers_api/tickers_api', methods=['GET'])
def get_tickers():
    # Query the database for the ticker's symbol, name, change, and percent change
    c.execute("SELECT symbol, name, change, percent_change FROM tickers")
    tickers = c.fetchall()

    # Convert the tickers to a list of dictionaries
    tickers_list = [{'headline': row[0], 'summary': row[1]} for row in tickers]

    return jsonify(tickers_list)

if __name__ == '__main__':
    app.run(debug=True)
```

- Only difference is variable names and what the cursor is selecting for in the database
- An external JavaScript file (all of my current JavaScript is in the HTML files [probably going to want to change]) was created to call the API and summon it into the website

```javascript
fetch('/articles_tickers_api/articles_api')
  .then(response => response.json())
  .then(articles => {
    const dynamicArticlesContainer = document.getElementById('dynamic-articles');

    // Loop through the articles and display them on the page
    articles.forEach(article => {
      const articleElement = document.createElement('div');
      articleElement.classList.add('article-preview'); // Add a class for styling
      articleElement.innerHTML = `
        <h1>${article.headline}</h1>
        <p>${article.summary}</p>
      `;
      dynamicArticlesContainer.appendChild(articleElement);
    });
  })
  .catch(error => console.error('Error fetching articles:', error));
```

4. **Difficulties Encountered this Progress Period** (Provide detailed information on the difficulties and issues that you encountered in the reporting weeks. Discuss mitigation strategies for how you got around or plan to get around these issues.)

Max -

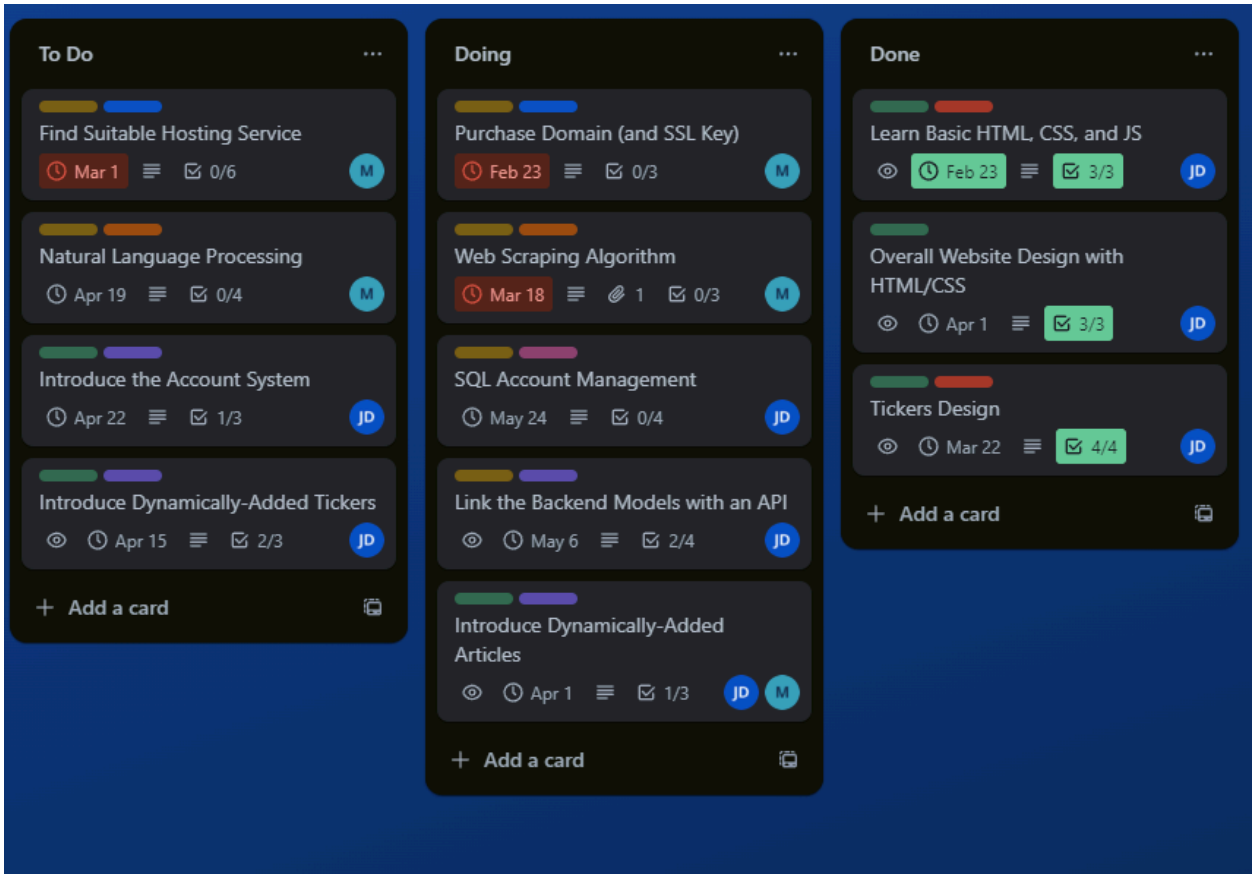
- Database stuff was difficult before I had mysqlworkbench
- Lots of random version issues
- It was tricky choosing which model I would use

Jake -

- Styling decisions have become difficult to determine as that is no longer in the foreground of my time
- Understanding the interactions between the API and how the dynamically-added divs are going to build on top of one another
- Sign up form is still going all the way to the bottom of the page instead of cutting off like the profile page despite them having the exact same CSS? It just looks a bit weird but it is by no means a top priority

5. **Updated Trello Board and Discussion** (Provide screenshot of and link to updated Trello board. Discuss any changes made to board since last progress report and why.)

https://trello.com/b/sSuMPdYn/morningbread



I (Jake) have been updating the tasks and marking things as done when they are done, as well as adding more detail to the cards that already exist.

6. **Tasks to Be Worked on in Next Progress Period** (Discuss the tasks to be worked on in the following two weeks. Discuss who is working on each.)

Max -

- Finally finish the yahoo scrape function (almost there for real this time) and clean up scraping
- Automate upload to SQL
    - Upload articles automatically
    - Delete old article on upload
- Use chat gpt to combine grouped headlines into one

Jake -

- If everything goes smoothly, articles stored in the MySQL database will be presented on the front page of our website
- Ensuring hourly updates is the next task as I'm not even sure if Max's database updates at all (I need to talk to him, I know)
- Keeping all aesthetics of the website when information is added

- I really want to make an accordian-styled About Us page, but I know that shouldn't be worthy of my attention right now
    - I'll probably do it over the weekend/Spring Break just for fun

7. **Additional Information** (Provide any additional information that you want to provide in this section; for example, one of your teammates is going away next week, your Github account is gone, etc. It could be good news as well.)

Max -

I feel like I am very ahead of where I thought I would be. I think it is likely that I finish my original goals in the next two sprints.

Jake -

No comments. My research project didn't win anything at the science fair, but I think the skills in API that I gained from doing it will translate well to MorningBread.