Week 5
Project Report

May 31 - June 6, 2021

Eric, Vicens, and Ben

StatCan Covid-19

Canada Emergency Wage Subsidy

Dashboard and Report

Overall Progress

- **1st Week:** Project understanding
- **2nd Week:** Data exploration
- **3rd Week:** Dashboard and report
- **4th Week:** Dashboard and report tweaks and enhancements
- **5th Week:** More minor improvements

Accomplishments

- **CEWS Dashboard**
  - Thematic consistency between Eric and Ben's work
  - Complete, polished, and sent to client
- **CEWS Analysis**
  - Metadata files written and code commented in preparation for sharing

StatCan Covid-19

API Feasibility Study

Report and Model

Overall Progress

- **4th Week:** API research
- **5th Week:** API feasibility report and initial model development

Accomplishments

- **API Report**
  - Feasibility study as an inventory of available APIs (cost, methods of accessing, wrappers, etc.)
    - 15 APIs currently covered
- **Twitter Sentiment Analysis**
  - Initial iteration for a model using Tweepy, TextBlob and NLTK
- **Steam Game Review Collection**
  - Collecting interesting game reviews during covid - 19

Twitter Sentiment Analysis

1. Gathering data

```
# Twitter Authentication
auth = tw.OAuthHandler(consumer_key, consumer_secret)
auth.set_access_token(access_token, access_token_secret)
api = tw.API(auth, wait_on_rate_limit=True)
```

```
# Filtering for tweets in Canada
places = api.geo_search(query='Canada', granularity='country')
place_id = places[0].id
```

```
# Search terms
num_of_tweets = int(input ('Please enter the number of tweets to analyze: '))
begin_date = (input ('Please enter the start date for the analysis (YYYY-MM-DD): '))
```

```
Please enter the number of tweets to analyze:  500
Please enter the start date for the analysis (YYYY-MM-DD):  2020-01-01
```

```
# Getting tweets and details
tweets = tw.Cursor(api.search, q='place:%s' % place_id, lang='en', since_id=begin_date).items(num_of_tweets)
tweets_details = [[tweet.created_at, tweet.text, tweet.user.screen_name, tweet.user.location] for tweet in tweets]
```

```
# Converting to dataframe for analysis
tweets_details_df = pd.DataFrame(tweets_details, columns=['Date', 'Tweet', 'User', 'Location'])
tweets_details_df.head()
```

	Date	Tweet	User	Location
0	2021-06-08 06:00:57	if you see someone blatantly copies another ar...	MaxMayonnaise	Toronto & Shanghai
1	2021-06-08 06:00:44	It's too hot to sleep you could melt grilled c...	rocketfromthe	Halifax, Nova Scotia
2	2021-06-08 06:00:39	DON'T BELIEVED THIS IDIOT CORRUPT EVIL PRIEST...	Alakdan1102Eric	California, USA, Vancouver Can
3	2021-06-08 06:00:35	BAD IDEA😂 he charged at me while I was holding...	maddypatra	Canada
4	2021-06-08 06:00:30	@drsaminaali Well i will follow you now.	cliffgadwa	North Battleford, Saskatchewan

Twitter Sentiment Analysis

2. Pre-processing data

```
# Cleaning tweets
tweets_details_df['CleanTweet'] = tweets_details_df['Tweet'].str.lower()
tweets_details_df['CleanTweet'] = tweets_details_df['CleanTweet'].apply(lambda elem: re.sub(r"(@[A-Za-z0-9]+)|([^0-9A-Za-z \t])|(\w+:\/\/\S+)|^rt|http.+?", "", elem))
tweets_details_df.head()
```

	Date	Tweet	User	Location	CleanTweet
0	2021-06-08 06:00:57	if you see someone blatantly copies another ar...	MaxMayonnaise	Toronto & Shanghai	if you see someone blatantly copies another ar...
1	2021-06-08 06:00:44	It's too hot to sleep you could melt grilled c...	rocketfromthe	Halifax, Nova Scotia	its too hot to sleep you could melt grilled ch...
2	2021-06-08 06:00:39	DON'T BELIEVED THIS IDIOT CORRUPT EVIL PRIEST...	Alakdan1102Eric	California, USA, Vancouver Can	dont believed this idiot corrupt evil priest
3	2021-06-08 06:00:35	BAD IDEA😂 he charged at me while I was holding...	maddypatra	Canada	bad idea he charged at me while i was holding ...
4	2021-06-08 06:00:30	@drsaminaali Well i will follow you now.	cliffgadwa	North Battleford, Saskatchewan	well i will follow you now

Twitter Sentiment Analysis

3. Modeling

```
# Tweets Sentiment Scores
blob = [TextBlob(tweet, analyzer = NaiveBayesAnalyzer()) for tweet in tweets_details_df['CleanTweet']]
tweets_details_df['Polarity'] = [tweet.polarity for tweet in blob]
tweets_details_df['Subjectivity'] = [tweet.subjectivity for tweet in blob]
tweets_details_df.head()
```

	Date	Tweet	User	Location	CleanTweet	Polarity	Subjectivity
0	2021-06-08 06:00:57	if you see someone blatantly copies another ar...	MaxMayonnaise	Toronto & Shanghai	if you see someone blatantly copies another ar...	-0.500000	0.500000
1	2021-06-08 06:00:44	It's too hot to sleep you could melt grilled c...	rocketfromthe	Halifax, Nova Scotia	its too hot to sleep you could melt grilled ch...	0.250000	0.850000
2	2021-06-08 06:00:39	DON'T BELIEVED THIS IDIOT CORRUPT EVIL PRIEST...	Alakdan1102Eric	California, USA, Vancouver Can	dont believed this idiot corrupt evil priest	-0.766667	0.933333
3	2021-06-08 06:00:35	BAD IDEA😂 he charged at me while I was holding...	maddypatra	Canada	bad idea he charged at me while i was holding ...	-0.700000	0.666667
4	2021-06-08 06:00:30	@drsaminaali Well i will follow you now.	cliffgadwa	North Battleford, Saskatchewan	well i will follow you now	0.000000	0.000000

```
# Tweets Sentiment Classification
tweets_text = tweets_details_df['Tweet'].tolist()
blob = TextBlob("".join(tweets_text), analyzer = NaiveBayesAnalyzer())
tweets_sentiment = pd.DataFrame(columns = ['Tweet', 'Polarity', 'Subjectivity', 'Classification', 'PositiveScore', 'NegativeScore'])
for tweet in blob.sentences:
    tweets_sentiment.loc[len(tweets_sentiment)] = [str(tweet), tweet.polarity, tweet.subjectivity, tweet.sentiment.classification, tweet.sentiment.p_pos, tweet.sentiment.p_neg]
tweets_sentiment.head()
```

	Tweet	Polarity	Subjectivity	Classification	PositiveScore	NegativeScore
0	if you see someone blatantly copies another ar...	-0.5100	0.83000	neg	0.032937	0.967063
1	https://t.co/AuMsfAHP5VBAD IDEA😂 he charged at...	0.0000	0.00000	pos	0.961992	0.038008
2	https://t.co/aq52pJvLJrit's cute how my friend...	0.5625	0.75000	pos	0.933133	0.066867
3	@SaneeahQ Yeah.....it's like....black coffee a...	0.3000	0.40000	neg	0.161880	0.838120
4	@MainKamlee I think I'll be on the same boat 🚣...	0.1000	0.35625	neg	0.027243	0.972757

Steam Game Review Collection

Motivation: Plague Inc. game review.

API used:

- **steamSpy:**

  A fast way to get game ID.

- **Valve API:**

  Collecting game reviews.

Link to repo:

https://github.com/ubco-mds-2020-labs/w2020-data599-capstone-projects-statistics-canada-covid-19/blob/main/code/API/Ben_SteamAPI.ipynb

API Research Report

- 15 APIs currently covered
  - Reddit, Facebook/Instagram, Foursquare, Yelp, TicketMaster, ...
- 7 sections for each API:
  - About
  - Use Case
  - Functionality
  - Capabilities
  - Access
  - Drawbacks
  - Conclusion

Planning and Action Items

- **CEWS Report**
  - Report editing and proofreading
- **CEWS Dashboard**
  - Tweaks
- **API Feasibility Study**
  - Report editing and improved coverage
  - Segment and assess tweets by location and interest categories
  - Analyze and visualize sentiment model

Roadblocks

- CEWS report feedback pending from StatCan
- CEWS dashboard redo
- Twitter API limitations
- Most APIs studied have limited potential for StatCan’s goals

Deviation From Initial Plan

- More focus put on the development of prototype API software
  - “Study whatever APIs interest you.”

Incremental Results

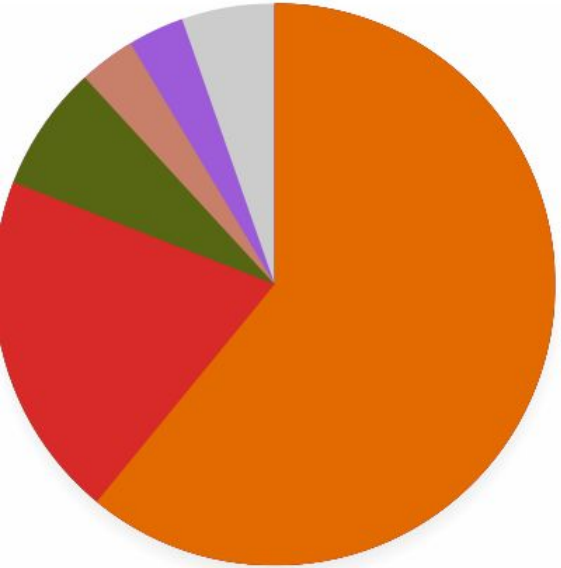
- **1st Week:** Project understanding
- **2nd Week:** Data exploration
- **3rd Week:** Dashboard and report
- **4th Week:** Dashboard and report tweaks
- **5th Week:** First draft API report and Twitter API demo
- **Remaining time:** API project and report/dashboard tweaks

Summary of Individual Work

Eric

05/31/2021 – 06/06/2021

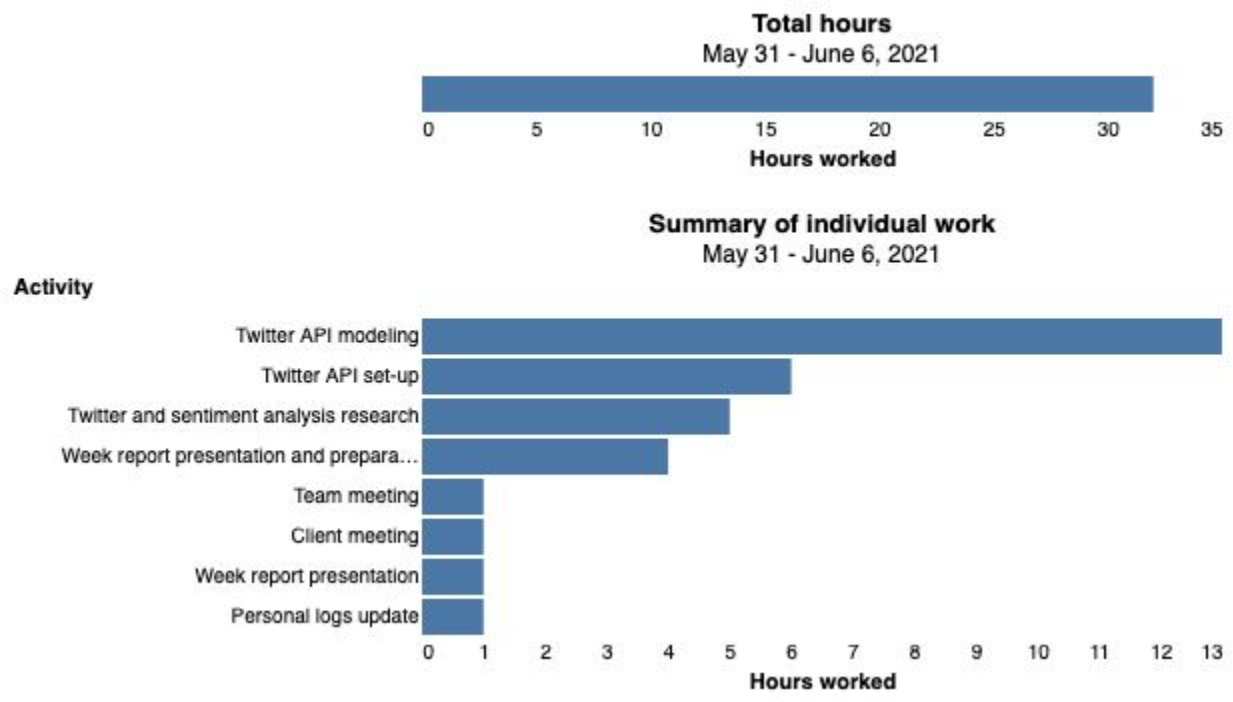
TOTAL HOURS: 30:54:00



TIME ENTRY	DURATION
API Report	18:49:00
Midterm Presentation	6:12:00
Dealing with plagiarism	2:13:00
Dashboard	1:00:00
Dashboard Polish	1:00:00
Other time entries	1:40:00

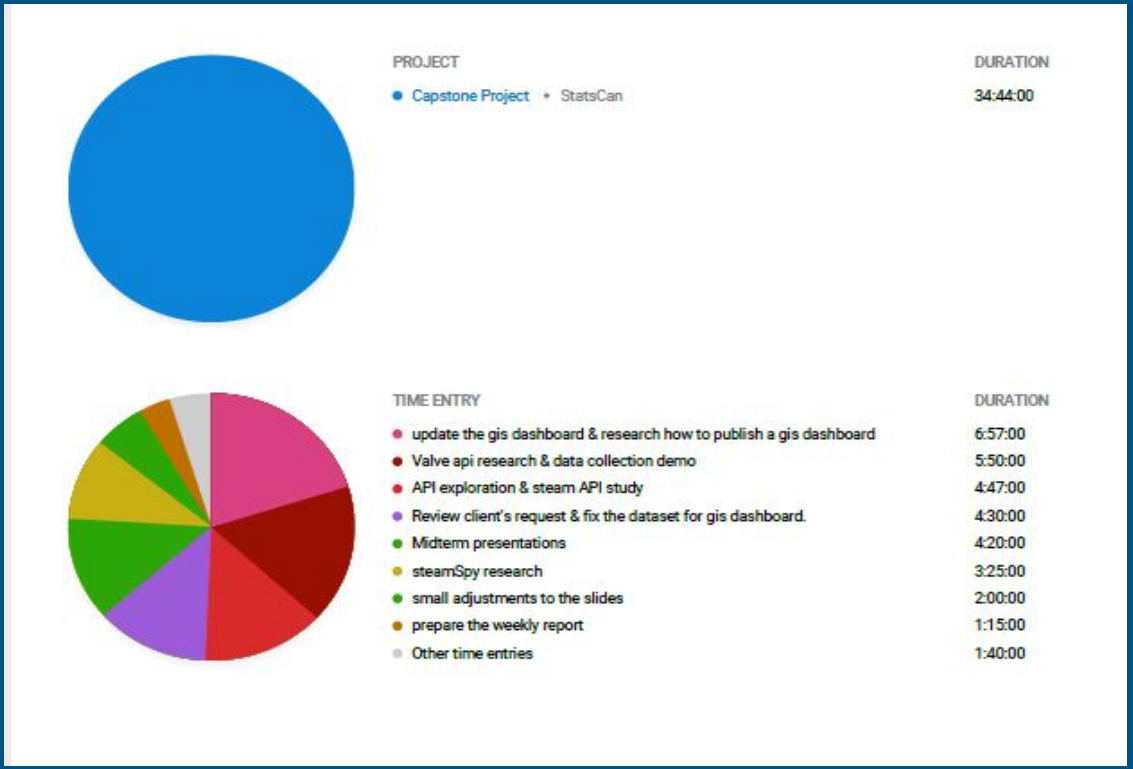
Summary of Individual Work

Vicens



Summary of Individual Work

Ben



PROJECT	DURATION
Capstone Project • StatsCan	34:44:00

TIME ENTRY	DURATION
update the gis dashboard & research how to publish a gis dashboard	6:57:00
Valve api research & data collection demo	5:50:00
API exploration & steam API study	4:47:00
Review client's request & fix the dataset for gis dashboard.	4:30:00
Midterm presentations	4:20:00
steamSpy research	3:25:00
small adjustments to the slides	2:00:00
prepare the weekly report	1:15:00
Other time entries	1:40:00

Client Minutes

https://github.com/ubco-mds-2020-labs/w2020-data599-capstone-projects-statistics-canada-covid-19/blob/main/meeting-logs/meetings_week4.md

https://github.com/ubco-mds-2020-labs/w2020-data599-capstone-projects-statistics-canada-covid-19/blob/main/meeting-logs/meetings_week5.md

Thank you!