**SteamIndex Documentation**

Idea:

To aggregate player reports based on coach, team, reporter comments to garner some sort of ‘score’ for the player. This score can be a variety of things. Depends on what you are looking at it for. ‘Player is clear RB2’ means scores goes up, ‘Player is playing third on depth chart’ goes down. Use a combination of AI/ML learning tactics to classify these player reports as good, bad etc… Further develop to incorporate ideas of DFS with ‘Steam’ on players. Player A being projected for 5% ownership but so many people talking about him means steam index goes up. Could also have a history where if place A is hyping him up then more then likely his ownership will go up compared to other spots. Something similar to Coachspeak Index in the way that it gathers coach comments to give player takes based on their index score. “Understand the mood of the market, the sentiment of the masses, to make informed decisions”. Can also have a pool of most common words used towards a player. “Workload, increase etc…”

Machine learning: Aspect of AI where algorithms are trained to learn from data and make predictions or decisions without being explicitly programmed. Train of massive sets of data.

Natural Language Processing (NLP): allows AI to understand, interpret and generate human language. Bridge between the ML algorithm and the raw textual data. What allows the AI to understand that when a company says stock x is soaring, it implies positive sentiment.

Sentiment analysis: process of determining whether a piece of text is positive, negative or neutral. Maybe in our case its ‘Steam Up’, ‘Neutral’, ‘Steam Down’. Or bullish, neutral, bearish. Can also determine the strength of the sentiment. Tell the difference between a mildly positive piece of news and a hugely positive one.

All this information is aggregated and synthesized into an overall market (or steam in our case) score.

Then this score can be used by people to gauge the overall sentiment towards a particular asset (in our case player, team).

Insights into Early Sentiment Stuff

Very rarely going to see full negative sentiment towards a player. The most you will see is neutral based because obviously coaches don’t want to throw their players under the bus. So, when seeing the neutral sentiment it is more indicative of ‘negative’ or downward trends towards the player.

Also, if there is negative context that doesn’t apply to the player, the sentiment score will pick it up as being negative. EX: The top pass catcher at 49ers rookie minicamp wasnt first-round WR Ricky Pearsall, but fourth-round RB Isaac Guerendo, who has a background as a receiver. This will get picked up as slightly negative/neutral by the sentiment because of the negative connotation towards Pearsall.

Will be most useful when you can simply take the press conferences in audio form and have it summarized for you. Able to get the videos from the teams youtube channels.

Storing the actual sentiment scores in the DB isn’t too useful. The sentiment one day for a quote versus the next wouldn’t change. The stored sentiment is of use when you want to filter by players or actually look into the scores. So, it doesn’t matter to much about storing sentiment over time, the more important thing is storing information in the quotes table. Then can build some sort of filtering system that will use the sentiment data when we want to look at it.

The primary factor affecting cost is the length and complexity of the input and output. Cleaning up the text might slightly increase processing time, but it is not a substantial change compared to the original prompt.

Barriers to Entry:

Finding reporters for each team. You don’t want the algorithm to incorporate false or misleading information. So, when searching for tweets, reports want to make sure it is from a qualified department. Need high quality data.

Coachspeak Index looks like it used videos posted to team websites. Offensive coordinators, coaches talking about players. He then grabs the video and the short summary of the player description. Then you could honestly, grab the video, get it automatically transcribed then feed that transcription as data. Major point of his page is devoted towards the press conferences. As season goes on more and more videos will be available. This is kind of the first point of entry. Either use the coach speaks videos or find them myself via the team websites.

Ideas into how to accomplish this. Best tactics for gathering information. How to gather information.

All the twitter scraping APIs seem to be not working due to new twitter rules. So, as I found out before you need a paid twitter subscription to be able to read tweets. In the meantime, can just build out a method to do this analysis with about 50-100 tweets. Maybe, use coach speak index’s tweet data to build this out. Honestly, could use a lot of what coach speak does. Retrieve videos (manually to start with but could automate later), then have the transcriptions made for them, automatically uploaded into a file. Then use that data to run the analysis.

Videos:

<https://www.youtube.com/watch?v=w1Bg1gO7L-M>

<https://youtube.com/watch?v=cRHt-S-HpIs>

<https://www.youtube.com/watch?v=UpFqcfEyoLE>

<https://www.youtube.com/watch?v=QpzMWQvxXWk>

<https://www.youtube.com/watch?v=sUtthdcPyhc>

AI stuff: <https://www.youtube.com/@robmulla>

Possible named sentiment:

<https://towardsdatascience.com/sentiment-analysis-of-the-new-york-giants-2020-draft-6f095e2363af>

Resources:

<https://twitter.com/32BeatWriters/status/1787669315568722336> - beat writers subscription could be a decent option to find all news related pieces. Can sort by team based news and player based. Have a lot of retweets as well.

<https://theathletic.com/5489798/2024/05/13/rookie-minicamp-scoop-city/> - good reports

<https://12ft.io/> - bypass ads

Done:

* Able to get a Roberta model sentiment analysis run on some data. Provided with negative, neutral and positive scores for each.
* Manually enter player names beside the player quotes. Then run the model which will aggregate all the sentiment scores for each player and store in the excel file. Color added to determine sentiment level.
  + -100 = all negative sentiment 0 = neutral 100 = all positive sentiment
* Created a mysql database. Created a table to store the quotes which are retrieved from the csv file. Also, stored the sentiment scores in the database under the ‘sentiment’ table, as well as the overall sentiment towards a player (which considers all three values)
* Setup the open AI API and can start using gpt 4o. $5 currently in, with more money added it seems you can move up tiers. Unsure of the limits apart of this tier.
* When prompted correctly the AI is able to distinguish the questions asked and the player takes accordingly, along with occasionally getting the player names correct.
  + Can generate summaries for each player as well
* Test with a much larger video to see how it performs. Performed well was able to accurately get all the questions and answers with longer videos.
* Found out that nfl teams have youtube channels where all their videos get posted (still wondering about post game press conferences, seems like teams post them)
* Storing the summaries (in the DB) to be used in the sentiment along with the date stored. As well as running the sentiment on these stored summaries.
* Stored all player names (using Underdog ADP and only players on teams currently) under the allPlayers table
  + Can add in defensive players in season (so like if a key defensive player is out or trending downwards it can affect things)
* Can retrieve the last 10 uploaded videos (within a 24 hour span) for a specific youtube channel (only the videos section for now) and save them in .mp3 format. Also, it will skip over previously downloaded videos within the 24 hour period by saving the IDs in a json file.
* After AI aggregates quotes by player, the function now gets rid of any non-offensive players (i.e. players not included in the allplayers table) and only inserts those.
* Audio downloader now runs through each teams youtube channel to download new videos posted in the last 24 hours (that are not already downloaded)
  + Automated retrieval from all the nfl youtube channels including the live videos (function checks for both)
* Generates transcriptions for each audio file in a folder by going through each team
* Break down larger audio files into smaller ones then combined all the transcriptions into one to be transcribed (using garbage collection to be able to delete the chunks that are generated, could cause issues later)
  + Also, deletes full video after getting the transcription for it (but keeps the id saved within the json file to prevent redownloads)
* Added possible name context to the prompt by providing it with a list of player summaries to create so it only creates summaries for players provided in the allPlayers list.
  + Can continuously update this to improve on what it outputs.
  + Not verified fix: fixed error with the presumed names getting added to the database table (that open AI suggests) and not the actual name from the underdog rankings.
* Converted the jupyter notebook to intellji idea to be able to make more progress going forward (especially GPU usage)
  + OpenAI already leverages its GPU on its own side so don’t have to worry about that
  + Seems like 2/3 of the time is spent towards transcribing. However, this transcription doesn’t use gpt 4o, so it is something that could be improved on when gpt 4o allows it
* Generate quotes gives a better outline of what exactly the quote said
  + One version is commented out (has questions included in the answer)
  + Other version has question separate
* Pre install of flask complete
* Download Youtube video function and transcribe audio is called with a button click
  + Only one at a time
* Show all the quotes with new columns added on the webpage. Can fix how it looks afterwards.
* Initial filtering and a more presentable quotes page is now displayed.
  + Upgrade filtering for multiple teams/players etc…
* Added date stamps to start of name of downloaded videos (so I can see if a download is duplicated or not, mainly for live and non-live press conferences)
* Can delete videos from website
* Able to pull videos from team websites using yt-dlp (press conferences tab for the eagles and bengals as they were the 2 teams that don’t upload to youtube)
* Delete quotes
  + Change to real time deletion (future)
* Changed some of the AI prompt to take into account more useful information
  + Can keep experiment with this and use chat gpt to help formulate a better prompt
  + <https://community.openai.com/t/openai-sentiment-analysis/599661/2>
* Accurate coach names (based on file name title)
* Sentiment runs and stores the sentiment beside in the quotes table in the db (it also won’t rerun any sentiment already stored)
  + AI determine if information is valuable or not as well as possibly running sentiment? With respect to player charts and such as well-being involved in the prompts. Can work on this further. $0.29 to run about 45 quote sentiments and 40,000 tokens
  + Sentiment also works over a given date range
* Created a simple UI to use to be able to load in reports from other sites instead of having to scrape them. Easier and can allow for more reports to come in
* Click on player name which links to all quotes about that player (i.e. redirects to quote page with name already filtered)
  + Fix issue with refresh and it should remove filter at top
* Updated sentiment page to link to reports as well and overall sentiment takes into account report sentiment
  + Added in date features as well so if sentiment selected over date range it redirects to only quotes or reports within that range
* News reports with the athletic page. Leg up has a bot for this
* Added ability to sort by columns when clicking on a table header
* Targets/fades based on previous weeks’ worth of reporting
* Possibly fixed issues with “Live” videos going on (added 'match\_filter': '!is\_live')
* Implemented the whisper-1 model locally which doesn’t use open ai’s paid services
  + See if this is still effective in generating transcripts
    - Will reduce costs but is a tad bit slower then open ai’s version
* Number of quotes/reports for a player beside their name in targets and fades section
* Added seahawks press pass to list

Todo:

* Better way to show sentiment scores (make it a more relevant page)
* Create page for DFS work (i.e. shows projections and stuff)
  + Add in missing information
  + Get data ready for pre-season week 1
  + Bought fantasydata membership which can help fill in some snap count data and other stuff
  + Keep adding comments about second stringers
  + Look at Cody’s picture to get more information (store playing time notes etc..)
* Figure out a way for the AI to only put out needed information from quotes. No need for all these access quotes that it is currently giving out
* Work on how the AI interprets this a bit. More interested in usage then just saying someone looks good
* Issue with players with ‘ in their names not properly formatted and thus not being inserted into db (possibly fixed?)
* Web app
* Have AI leave out non important information

Future Work

* Load 10 quotes at a time instead of all
* Filter out insightful videos from good ones when downloading
  + Could be something like check the title for ‘press’, ‘conference’, ‘coach’, etc…
* Garbage collection being used to delete chunks (lower priority for now but could cause an issue)
* More weight can be applied to comments from more legitimate sources.
* Most used words for a certain player (workload, increase, etc…)
* Have time stamps of when the quote was said which can then link to the part in the actual video (or audio file) where that was shared, to get more context
* See how much more expensive or resources it would take to get video based transcription that can take into account more of the context of what is going on
* Warning of throttling issue with videos
* Get to a point that I have some sort of online interface where I can view player sentiments and the associating quotes that generated that, the date of those reports, who said it etc…. Massive database containing insight on the player.
* Streamlined process

Issues

End Goal

1. Get text into files
   1. Via video transcriptions, tweets, etc…
2. Model should be able to classify the text into names for the players and provide sentiment scores for each player
   1. Along with popular used words for the player
3. Generate document containing this information
4. Streamline entire process

API

sk-proj-dHtSQFcagTSv5mq1YFdRT3BlbkFJFnRLZYOiwsQeZOKOaLkh

<https://github.com/yt-dlp/yt-dlp>

beatwriters  
JPuM3z5ctgMwNfc

we look thru depth charts, beat reports, and try to basically what each team is going to actually look like in terms of who is getting carries/targets, what type of output each offense will have, scheme changes, etc how does that affect usage and suh

preseason stuff

<https://www.ourlads.com/nfldepthcharts/>

<https://www.footballdb.com/statistics/nfl/player-stats/rushing/2022/preseason>

insert statement

INSERT INTO rb\_wr\_te\_data (number, player, pos, school, orig\_team, nfl\_exp, gms, ratt, ryds, ravg, rypg, rlg, rtd, rec, reyds, reavg, reypg, relg, retd, tar, team) VALUES (22, 'Akers, Cam', 'RB', '', '', 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 'Texans')

INSERT INTO qb\_data (number, player, pos, school, orig\_team, nfl\_exp, gms, att, cmp, pct, yds, ypa, td, td\_pct, inter, ypg, sack, ratt, ryds, ravg, rypg, rlg, rtd, team) VALUES (0, 'Luton, Jake', 'QB', '', '', 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,0,0,0, 'Panthers')

Old prompt

The audio provided is a transcript of a press conference. The speaker will talk about players on the roster and provide insights into their performance, among other topics.

Your task is to extract relevant quotes about the players from the list provided ({allPlayers\_values}).

The list includes the correct names of all players on the team, which should be used as a reference to ensure accuracy when printing out names.

The player names may be mentioned in various forms such as first names, last names, or nicknames. Please refer to the provided list to resolve any ambiguities.

Instructions:

1. Output the information as a CSV with the following headers: name,title,question,quote,speaker.

2. For each player mentioned in the press conference, provide:

- The player's full name as listed in {allPlayers\_values}.

- A concise summary (title) of the major point said about the player in a few words.

- The question that prompted the response. If no specific question prompted the response, write "no question".

- The exact quote about the player. Clean up the text to correct any grammatical errors, spelling mistakes, or incoherent sentences while preserving the original meaning and content. Ensure there are no extra spaces around commas and quotes within the quote field.

- The name of the speaker who made the quote. Oftentimes, the name of the speaker can be extracted from the title of the transcription file: {title}.

3. Combine all quotes about the same player throughout the conference into a single entry. Ensure the information is comprehensive.

4. Focus on extracting quotes that provide useful insights into player performance and status. Examples of useful insights include:

- "getting more playing time"

- "running with the first team"

- "will see increased usage"

- "will lead the backfield"

- "is currently injured"

- "will get more carries"

- "needs a lot of work"

- "is third on the depth chart"

- anything else not closely related to these do not extract quotes for them

5. Ensure each row in the CSV matches the format specified in the headers. Pay attention to:

- No extra spaces around commas.

- Properly formatted quotes with no spaces immediately inside the quotes.

- Correct handling of special characters, such as apostrophes, in player names.

- Ensure that apostrophes within quotes or the question are preserved correctly and do not cause formatting issues.

6. Verify that all lines in the CSV are complete and avoid EOF issues by ensuring each row follows the correct structure and format without missing or extra fields.

7. Do not include the "```csv" symbols at the top or bottom of the output.

Example Output:

name,title,question,quote,speaker

Brock Bowers,Versatile tight end,"What about Brock Bowers? I know he's a tight end, but he's really a weapon. I mean just what do you see when you see him?","Brock will get a lot of work this year and has the ability of being a good tight end. He wants to be a really good blocker. I know that's important to him. And, you know, he's preparing himself just for that.",John Harbaugh

Isaiah Michael,Will be a big part of the offense,"How about Isaiah Michael? We know what he can do over the last couple of years. He's made quite a few wild plays this week. How impressed have you been with his game?","He's talented. He's going to be a big part of what we do and have a big role in the offence.",John Harbaugh

Please ensure the final CSV is properly formatted and contains accurate information.