Monitoring Public Sentiment of NFL Draft Picks via Machine Learning Techniques

Jonathan Wiseman

Elizabethtown College

(717)857-4069

wisemanj@etown.edu

**ABSTRACT**

Sentiment analysis is a topic in natural language processing that seeks to automatically extract positive and negative polarity from text data. Its applications are diverse, ranging from marketing and sales to forum moderation to gauging public opinion. One particularly interesting application area can be found in professional sports: fans share a huge volume of opinions, predictions, and reactions online that can be used to gauge public opinion on specific entities. This paper explores the application of machine learning based sentiment analysis on a hand-labeled social media dataset focused on reacting to National Football League draft picks. The resulting model provides information for more detailed analysis, including attitude towards drafted players, comparison between fan reactions and on-field performance, and comparison between drafted players based on the language used to describe them. Additionally, a labeled dataset for sentiment analysis on professional football will be created for further use.

# INTRODUCTION

The National Football League is the world’s most profitable sports league, achieving over $13 billion in revenue in 2017 alone [5]. The league continues to expand and evolve, drawing in viewers from around the world and regularly updating safety standards, rules and regulations, and even team locations. Accompanying this expansion has been an interest in applying analytics to data generated by NFL players, coaches, and outside observers. In 2019, the NFL hosted its inaugural Big Data Bowl, challenging college and independent teams to make use of its databases to generate valuable insights about the game and its players [12]. The spirit of the Big Data Bowl reflects a growing interest in using the techniques of data analysis and machine learning to generate insights that stretch across many different sports areas.

The application areas of data analysis on professional football are diverse, ranging from a Sabermetrics-like approach to predicting game and player performances to suggesting rule and safety changes to market analysis of commercial placement and fan engagement. A large media empire has grown up around professional football, with injury reports, game predictions, and assorted player and coach news providing constant coverage on all aspects of the game.

Sentiment analysis is a field spanning the disciplines of natural language processing, machine learning, information retrieval, and text mining [15]. Its primary aim is the automatic extraction of standpoint, view, and mood of an author [15]. The most common use of sentiment analysis is determining the polarity (i.e. positive or negative) of a particular sample of text. This can be of particular use in marketing research, where companies seek to gauge public opinion of their products; other application areas include monitoring of online forums, automatically assessing product reviews, and as additional information for search engines [3].

Sentiment analysis can be performed in one of two ways. The first is a grammatical approach based on the linguistic features of text, such as descripting adjectives and adverbs, negation words (i.e. “not”), intensifiers (i.e. “very,” “extremely”), case, and tense [3]. This approach involves the creation of a carefully crafted lexicon that accurately captures the sentiment of words specific or important to a particular domain; for example, a lexicon crafted for determining sentiment in sports articles would have to assign sentiment to words like “interception” and “fumble.” The second approach involves the use of machine learning algorithms to create models that can predict the sentiment of a given text.

A general problem with sentiment analysis is its inability to generalize across multiple domains; for example, a lexicon or model crafted for use in the movie reviews domain will not generalize well to the sports domain [3]. This makes the crafting of specific lexicons time-consuming and requiring a significant amount of domain knowledge. The machine learning approach runs into similar problems: supervised classification requires carefully labeled datasets, which are often not publicly available or are based on implicit ratings (for example, movie and product reviews are standardized on a five “star” scale that gives text data implicit ratings). Either approach requires a significant investment in either crafting a lexicon or acquiring a significant dataset that captures the nuances of a given application field.

Any potential use of sentiment analysis on NFL articles must be performed with a specific goal in mind. Using news articles to predict the outcome of NFL games is problematic. For one, most articles are not specific to one aspect of the game: there are injury reports; news and updates on trades, signings and draft prospects; articles about players’ personal lives; and news about retired players and coaches that are no longer active in the game. Each of these areas requires a specific lexicon, and it is doubtful that each is useful in predicting the outcome of a specific game. Secondly, each article deals with multiple players and topics, such that extracting entity-based sentiment is difficult. For example, one sentence in an article might deal with an offensive and a defensive player. This makes sentiment analysis difficult, since phrasal extraction is a difficult area of natural language processing [3]; additionally, this requires a model that is capable of orienting sentiment-bearing words to specific players (i.e. an interception is bad for an offensive player but good for a defensive player).

It is clear that any sentiment model based on football text must be directed and purposeful. One particularly useful application area is determining public sentiment around NFL draft picks. The NFL draft is an annual event in which college football players are selected by professional teams for short-term “rookie” contracts [8]; it is the primary mechanism by which college talent enters the NFL. This task is useful for several key reasons. For one, high-valued draft picks (i.e. those selected in the early rounds of the draft) are expected to be polished, capable players. Although rookie contracts are generally inexpensive compared to those for veteran players [8], teams wish to avoid selecting players whose draft stock does not translate well into actual on-field performance. Creating a model to process text data related to draft picks is a useful analytics tool for gauging expert and public opinion towards a player’s potential. Secondly, gauging sentiment towards a player is useful from a marketing perspective. The off-field (and sometimes on-field) actions of a player influence fans’ perspectives of players and their willingness to engage with the franchises to which they belong. For example, the impact of on-field protests by NFL players such as Colin Kaepernick on NFL revenues is examined in [6]; for an example of a player’s actions harming team reputation, see the example of Antonio Brown in [7].

To gauge public reactions to NFL draft picks, I propose *DraftSense*, a machine learning approach to sentiment analysis on text relating to draft picks after they are made. The key design goals of DraftSense are:

**Comprehensive:** the ability to collect a large volume of data

**Specific:** collecting data specific to NFL draft picks

**Accurate:** accurately predict sentiment to summarize the public’s reactions to NFL draft picks

# BACKGROUND

This section gives a more detailed view of the key aspects necessary to create DraftSense. The topics are:

* Aggregated Forecasts
* Approaches to Sentiment Analysis
* Document Representation
* Dataset

## Aggregated Forecasts

It has been consistently observed that aggregating a number of individual forecasts performs better over time than relying on a single forecast [16].

In [16], this principle was applied in the sphere of politics and international events by the Good Judgment Project and tested over time in the U.S. Intelligence Advanced Research Projects Activity’s Aggregative Contingent Estimation program [16]. The team made probabilistic judgements about specific events (i.e. Greece leaving the Eurozone) by framing them as yes-no questions and presenting them to a poll of 2400 Americans from myriad demographics and professions [16]. The team employed various aggregation techniques ranging from simple averaging to log-odds extremizing of weighted averages [16]. Overall, their methods outperformed U.S. intelligence community predictions by about 30%, even when intelligence officials were given access to classified material [16].

In [9], a lexical approach to sentiment analysis was applied to text data from news, blog, and other web sources. The authors constructed *Lydia* and used it to produce a betting paradigm for NFL games. The favorability of a team is derived from its daily positive and negative mentions in the text dataset [9]. Utilizing sentiment alone, the authors achieved 60% prediction accuracy for the 2006-2008 seasons [9]. The authors found that combining sentiment, statistical performance prior to games, and home field advantage produced the most robust model; however, the authors note that the sentiment model only produced significant improvements over the second half of the NFL season [9].

## Approaches to Sentiment Analysis

[3] provides an overview of sentiment analysis approaches, including both symbolic/lexical and machine learning models. The authors note the difficulty of manually crafting rules and lexicons for specific domains [3]. There are a number of methods for applying lexicons to derive sentiment: bag-of-words with unigrams and bigrams, sentence-level application, and phrase-level analysis [3].

[15] also provides an overview of commonly used machine learning algorithms, including Support Vector Machines, Naïve Bayes Classifiers, and Maximum Entropy. These supervised methods require a fully labeled dataset; unsupervised and weakly-supervised methods do exist but focus on combining the machine learning and lexical approaches [15].

[15] also discusses feature selection for machine learning based sentiment analysis. Using IMDb’s movie review database, seven machine learning classifiers were trained while varying feature selection and feature count. Five feature selection methods (information gain, gain ratio, CHI statistics, Relief-F, and document frequency) and seven machine learning models (K-nearest neighbors, Naïve Bayes, Winnow, Maximum Entropy, and Support Vector Machines) were tested for performance; documents were represented using the Vector Space model [15]. In total, the authors noted that gain ratio feature selection used with Support Vector Machines achieved the highest test accuracy at 90.15%.

## Document Representation

Many machine learning models require data input in numeric vectors, which presents a challenge for text-based datasets. There are two major approaches to representing text: bag-of-words and embedding.

Bag-of-words entails representing each document as a sparse matrix consisting of binary attributes (with features being words: a 0 denotes the absence of a word, a 1 indicates the presence of a word) for each word in the collection’s vocabulary. TF-IDF vectorization is an extension of the standard bag-of-words approach to document vectorization: rather than using static counts of words for each document, each term’s term-document inverse-document frequency measure is used. Broadly speaking, the tf-idf is a measure of a word’s importance to a specific document. This approach suffers from sparsity and huge dimensionality, particularly in documents with non-standard language (such as social media posts or blogs). Additionally, the bag-of-words model loses word order and ignore semantics [11].

Embedding entails the representation of words, sentences, or documents as a fixed-length feature vector. Original work in embeddings focused on mapping words to a vector space such that “semantically similar words have similar vector representations” [11]. However, word vectors on their own are fairly limited in the context of larger samples of text. A number of approaches have been used, such as averaging the vector-representations of words in a document either simply or in specific orders given by grammatical parse trees [11].

A number of algorithms have been proposed to move beyond word embeddings in order to represent longer samples of text. In [11], *Paragraph Vector* learns vector representations of variable-length pieces of text. The authors treat the problem as a classification task: word vectors contribute to a prediction about the next word in the sentence [11]. This allows vectors to represent semantics and context [11].

In [13], *Sent2Vec* builds on the work of [11] to increase speed and lower complexity. Sent2Vec can be seen as an extension ofParagraph Vector [13]. The authors note that Sent2Vec achieves greater performance with long pieces of text, including entire documents [13].

In [4], *BERT* expands on previous models by allowing tokens to be interpreted in both left-to-right and right-to-left contexts. The authors note BERT’s increased ability to create cross-domain embeddings, reducing the need for “heavily-engineered task-specific architectures” [4].

## Dataset

In [3], the authors note that sentiment analysis is a domain-specific problem. Thus, sentiment analysis models require robust datasets for specific domains. For supervised approaches, this includes labeled sentences, documents, or phrases. Text source is also important: a model for analyzing social media posts will look for different linguistic features than one that analyzes academic writings. However, [9] notes that mainstream news serves as a good proxy for polarity between varied text sources, indicating some degree of agreement between all of them. In other words, a subject spoken about positively in social media posts will also be talked about positively in mainstream news sources.

There exist no substantial publicly available datasets for the proposed task. Training on traditional benchmark datasets such as the IMDb dataset, the Toronto Book Corpus, or Wikipedia’s available corpus is possible; however, none of these datasets represents the domain of NFL football or sports in general. As such, the creation of a sentiment-analysis dataset for professional American football is a major goal of this project. See the Design section below for a discussion of problems in obtaining a comprehensive dataset.

# DESIGN

A picture containing text, map

Description automatically generatedThis section presents an overview of DraftSense design. Figure 1 displays the general workflow of the system in practice:

**Figure 1: DraftSense Overview**

The process can be broken down into five primary stages:

1. Data collection
2. Data preprocessing
3. Sentence embedding or feature selection
4. Model creation
5. Analysis

## Data collection

All data will be collected from public Reddit posts created by r/nfl’s u/NFL\_Mod, an automated moderator that creates important subreddit threads (such as game threads, draft picks, game reactions, and more). A total of twelve players have been selected to create a sentiment analysis dataset and analyze:

1. Baker Mayfield
2. Mitchell Trubisky
3. Daniel Jones
4. Kyler Murray
5. Lamar Jackson
6. Dwayne Haskins
7. Deshaun Watson
8. Sam Darnold
9. Deshone Kizer
10. Josh Rosen
11. Josh Allen
12. Patrick Mahomes II

In order to maximize specificity of language, each of the players chosen is a quarterback drafted between 2017-2019. For each official r/NFL draft thread, the every top-level and second-level comment will be scraped using Reddit’s official PRAW API; the documentation for PRAW is available at [2].

The acquisition of an appropriate sentiment analysis dataset for sports has been a major challenge in proposing this project. Initially, 24000 articles were scraped from ESPN to use in game prediction. However, upon closer inspection most of these articles are not particularly useful (as explained in the Introduction). Additionally, ESPN locks most of its game and draft predictions behind its subscription-only ESPN Plus (formerly ESPN Insider); this makes scraping subscription-only articles difficult and prevents their release as a publicly available dataset (for example, consider the ethical impacts of sharing these authors’ articles as presented in [10]). Searching articles on sports websites such as nfl.com, CBS, NBC, and Fox are not archived in an easily accessible way, preventing any meaningful scraping of past articles.

Without a comprehensive and domain-specific dataset for NFL sentiment analysis, the aims of this project cannot be meaningfully met. As such, the production of a publicly available dataset for sports sentiment analysis is a major goal. The data collected will be hand-labeled, either by the author or through crowdsourcing platforms such as Amazon Mechanical Turk [1]. I have found that, on average, it is possible to label about 1000 comments per hour, making the task of labeling each comment feasible, if time-consuming.

Each comment will be labeled as either positive or negative regarding the player drafted, with extraneous or joke comments labeled as “other.” For example, there are a number of jokes posted about draftees that are neither positive nor negative; some comments deal with other topics as well and should not be treated as draft reactions. In total, I estimate there will be about 10000 labeled comments by the end of the project, although the distribution of positive, negative, and junk comments is not currently known.

## Data Preprocessing

An important step in utilizing text data is preprocessing. This includes removing hyperlinks and URLs, emojis, standardizing expressions (for example, shortening “noooooo” to “no”), and possibly removing non-alphabetic characters (like punctuation, braces, and parentheses). However, it is possible that leaving in punctuation will increase the robustness of text available for evaluation.

Performing lemmatization to standardize word forms is an important step in text data preprocessing, but it is questionable if it is valuable here. For producing sentence embeddings, it is not essential to perform lemmatization; however, for bag-of-words models, lemmatization helps to reduce the dimensionality of the ensuing term-document matrix [3]. The final model will either use sentence embeddings or bag-of-words representations.

## Sentence Embedding or Feature Selection

Following the data preprocessing stage, each comment will be represented as either a numeric vector as created by sentence embedding techniques or a bag-of-words term-document matrix.

As discussed in the Background above, there exist several publicly available sentence embedding algorithms, such as Gensim’s implementation in [14], Google’s BERT in [4], and Sent2Vec in [13]. The numeric representations obtained from these algorithms can be directly fed into machine learning models.

To mitigate non-standard language, feature selection in the form of maximizing gain ratio and information gain as performed in [15] will be used. Additionally, the number of features selected will be varied.

## Model Creation

After each comment is represented as fixed-length numeric vector, it can be used in machine learning tasks. DraftSense will experiment with several models in order to maximize prediction accuracy, including Support Vector Machine, Logistic Regression, Naïve Bayes Classifier, and Neural Networks.

Sentiment analysis is presented here as a binary classification task: predicting whether a comment is speaking positively or negatively about a draft pick. Models with sufficient accuracy will be incorporated into the final DraftSense pipeline.

## Analysis

The working sentiment analysis model will be applied to r/nfl threads to examine three areas:

1. Positive and negative counts for each quarterback
2. Comparison between sentiment and real-world performance
3. Comparing quarterback similarity based on sentence embeddings

### Positive and Negative Counts for Each Quarterback

Producing simple positive and negative counts for each quarterback is the primary goal of DraftSense. These raw counts will show the general public consensus towards a draft pick. Additionally, raw positive and negative comment counts will allow direct comparison between drafted quarterbacks.

Reddit’s comments also carry additional information that might enrich raw counts. Each comment carries with it a score, which represents the number of users who agree with a comment or find it useful. For example, a negative comment with high score might represent an insightful and more valuable critique of a player than many positive comments with low score. However, it is not immediately clear how to incorporate comment score into DraftSense, since there is no official, established consensus about a draft pick; that is, it is impossible to determine the meaningful impact of weighting comments because there is no set outcome to achieve.

### Comparison between Sentiment and Real-World Performance

One particularly interesting application of DraftSense will be the ability to compare the public’s reaction to a draft pick and that player’s actual performance on the field. For example, Patrick Mahomes II received mixed draft grades prior to his selection by the Kansas City Chiefs in 2017. However, he has since won the NFL’s Most Valuable Player Award and is set to compete in Super Bowl LIV in February 2020. Similarly, many players whose selections were viewed positively have struggled with consistent play on the field.

#### Comparing Quarterback Similarity Based on Sentence Embeddings

Training sentence embeddings on comments for each quarterback will allow a comparison of quarterback similarity based on the language used to describe them. This will require some additional training, since each comment will be turned into an embedding. However, treating the body of comments about a quarterback as a document might produce meaningful embeddings. Quarterbacks who are talked about with similar language will be closer together in the vector space, potentially generating interesting insights. For example, quarterbacks who are more run-focused might be talked about in different ways than quarterbacks who are pocket passers. Additionally, quarterback with similar strengths (or weaknesses) might be embedded closer.

# EXPERIMENTS

Choice of sentence embedding algorithm will have a substantial impact on final classification accuracy. In the Background and Design sections, three such algorithms were discussed: Gensim’s implementation of Doc2Vec, Sent2Vec, and Google’s BERT. Each sentence embedding algorithm will be tried in conjunction with each machine learning model. The most accurate embedding algorithm will be chosen for inclusion in the final DraftSense system.

Feature selection similar to that in [15] will be performed on bag-of-words representation of comments. Most importantly, the number of features will be varied with information gain in an attempt to maximize classification accuracy downstream.

Four machine learning models will be trained for sentiment analysis: Support Vector Machine, Logistic Regression, Naïve Bayes Classifier, and Neural Network; the architecture for the Neural Network implementation will be determined for future milestones. Classification accuracy will be used as the metric for each machine learning model.

# TIMELINE

Completion of the project will be broken down into five stages, with dates as follows:

1. January 27, 2020: Submission of proposal
2. February 17, 2020: Complete dataset collection and cleaning; selection of sentence embedding algorithm/implementation
3. March 16, 2020: Complete dataset labelling; training of initial models
4. April 6, 2020: Final model selection and training; analysis comparing sentiment around quarterbacks; application to other draft threads
5. April 27, 2020: Finalization of pipeline and organization

# REFERENCES

1. Amazon Mechanical Turk. (January 27, 2020). Retrieved January 27, 2020 from <https://www.mturk.com/>.
2. Bryce Boe. PRAW Documntation. (January 27, 2020). Retrieved January 27, 2020 from <https://praw.readthedocs.io/en/latest/index.html>.
3. Erik Boiy, Pieter Hens, Koen Deschacht, and Marie-Francine Moens. 2007. Automatic Sentiment Analysis in On-line Text. In *Proceedings ELPUB 2007 Conference on Electronic Publishing*, 13 - 15 June 2007, Vienna, Austria. 349-360.
4. Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. 24 May 2019. Retrieved from <https://arxiv.org/abs/1810.04805>.
5. Forbes. 2017. Total revenue of all National Football League teams from 2001 to 2018 (I in billion U.S. Dollars). Retrieved January 23, 2020 from <https://www.statista.com/statistics/193457/total-league-revenue-of-the-nfl-since-2005/>.
6. Ahiza Garcia. 2018. Will the NFL Ban on Kneeling Boost Ratings? (May 23, 2018). Retrieved January 27, 2020 from <https://money.cnn.com/2018/05/23/news/nfl-ratings-anthem-protest/index.html>.
7. Paul Gutierrez and Mike Reiss. 2019. Timeline of Antonio Brown’s Season. (Sep. 20, 2019). Retrieved January 27, 2020 from <https://www.espn.com/nfl/story/_/id/27544508/line-antonio-brown-season>.
8. Tadd Haislop. 2019. How does the NFL Draft Work? Rules, Rounds, Eligibility, and More. (April 27, 2019). Retrieved January 27, 2020 from <https://www.sportingnews.com/us/nfl/news/how-does-the-nfl-draft-work-rules-rounds-eligibility-and-more/o431yshp0l431e7543pcrzpg1>.
9. Yancheng Hong and Steven Skiena. 2010. The Wisdom of Bookies? Sentiment Analysis Versus the NFL Point Spread. In *Fourth International AAAI Conference on Weblogs and Social Media*, May 2010, AAAI Publications.
10. Samer Kalaf. Keith Law Will Find You if You Repost his ESPN Insider Articles. (April 6, 2015). Retrieved January 27, 2020 from <https://deadspin.com/keith-law-will-find-you-if-you-repost-his-espn-insider-1695950422>.
11. Quoc Le and Tomas Mikolov. 2014. Distributed Representations of Sentences and Documents. In *Proceedings of the 31st International Conference on International Conference on Machine Learning*, 21 June 2014, Journal of Machine Learning Research, 1189-1196. arXiv: 1405.4053. Retrieved from <https://arxiv.org/abs/1405.4053>.
12. NFL OPS. 2019. The NFL’s Inaugural Big Data Bowl. (February 2019). Retrieved January 23, 2020 from <https://operations.nfl.com/the-game/big-data-bowl/?utm_source=Direct>.
13. Matteo Pagliardini, Prakhar Gupta, and Martin Jaggi. 2018. Unsupervised Learning of Sentence Embeddings using Compositional n-Gram Features. In *Proceedings of NAACL-HLT 2018*, June 1 – 6 2018, New Orleans, Louisiana. Association for Computational Linguistics, 528-540. <https://doi.org/10.18653/v1/N18-1049>.
14. Radim Rehurek. Doc2vec Paragraph Embeddings. (Nov. 1, 2019). Retrieved January 27, 2020 from <https://radimrehurek.com/gensim/models/doc2vec.html>.
15. Anuj Sharma and Shubhamoy Dey. 2012. A Comparative Study of Feature Selection and Machine Learning Techniques for Sentiment Analysis. In *Proceedings of the 2012 ACM Research in Applied Computation Symposium*, 23 – 26 October 2012, San Antonio, Texas. ACM Inc., New York, NY, 1-7. https: doi.org/10.1145/2401603.
16. Philip Tetlock, Barbara Mellers, and Peter Scoblic. 2017. Bringing Probability Judgments into Policy Debates via Forecasting Tournaments. *Science* 355, 6324 (February 3, 2017), 481-483. DOI: <https://doi.org/10.1126/science.aal3147>.