Developing an Artificial Intelligence Agent to Predict Offensive Play Calling In Canadian Football

Noah LeFrancois[a]

[a]Department of Physics, McGill University, 3600 Rue Université, Montréal, QC, H3A 2T8, Canada

This manuscript was compiled on August 26, 2020

This work presents an ensemble classification method for predicting the offensive play selection of an opponent in order to assist the decision-making of defensive football coaches. By combining multiple prediction models, this method achieves more consistent predictive accuracy than the individual models themselves. A streamlined user interface is employed to allow efficient in-game use of this application by football coaches and not just scientists. The training data input process is designed for easy use of the data formats already used by college and professional teams, and new plays seen by the model throughout a game can easily be added to that data base. This prediction method can offer strategic insight to defensive coordinators in a number of situations including game planning and live use in the coaches' booth. Additional suggestions for the modification of this method for use by offensive coordinators in predicting defensive play calling are also made, displaying the widespread potential implications of this assistive coaching technology.

Football Play Prediction | Sports Analytics | Classification Learning

The use of statistical methods to inform decision making in sports has grown rapidly in the 21st century as vast quantities of data has become available alongside the development of flexible, open-source machine learning code libraries. Football is a prime candidate for the application of these methods due to the discrete nature of its play sequence which is separated into offensive and defensive plays which can be easily categorized by strategy (whether the play called was a dropback pass, a play-action pass, a run-pass option, etc.).

Past investigations into the application of machine learning to offensive play calling have mostly focused on the selection of classification algorithms which are well-suited to the problem and the optimization of their parameters. They focus on exploring the mathematics underlying the problem, while treating the direct applicability to live support of coaches' decision making as a secondary concern.

As a number of works have already thoroughly explored these more theoretical aspects of play prediction, my work has integrated their algorithm-selection insight into a framework tailored to the intended end-user: professional and college coaches.

In most professional and college leagues across Canada and the United States, the play-by-play situational data necessary for training a classification learning algorithm is already being collected either in-game or from post-game film review for game planning purposes. By taking this existing data and creating a framework which minimizes the reformatting necessary for use by a coaching staff, an effective and user-friendly application has been developed which can provide live play predictions in the coaches' booth on gameday.

An intuitive user interface which allows users with no programming knowledge to quickly input the situational data of an upcoming play and receive a prediction is also essential to making this resource directly useful to a defensive coordinator.

The remainder of this paper is organized as follows: Section 1 will discuss previous work on the problem of statistical football play prediction. Section 2 will outline the data preprocessing, classification models, and user interface employed by this application. Section 3 will discuss the model performance, some important usage recommendations, and directions for future work. Finally, in Section 4 some concluding statements will be made on the importance of this work and its applications.

1. Previous Work

A number of investigations have been conducted into the selection of classification algorithms for determining whether the offence is going to call a run or a pass on the next play. They have primarily used data scraped from NFL play-by-play records, which constitute a large data set of 10 seasons, with 256 total regular season games played by the 32-team league each season. A downside to this data is that it contains limited play result categories of use for informing the decisions of defensive coordinators; all past work I have encountered is designed to predict run/pass data with no major investigations into more detailed predictions such as specific play type or

Significance Statement

One of the primary challenges of defensive play calling is understanding the offensive play calling strategies of each unique opponent in any given situation and predicting their actions in order to call an appropriate defensive scheme to stop them. This project aims to provide an accurate and easy-to-use prediction application capable of providing this valuable information to defensive coordinators in a live game environment. Prior investigations into the use of machine learning to make simple run/pass predictions have shown promising results, but have received mixed feedback from coaches who want more specific predictions and need a simple user interface which requires no programming knowledge to operate. This project focuses on using the play-by-play data already available to Canadian USports coaches in combination with an efficient user interface which makes more specific play type predictions to cater to the needs of the live in-game coaches' booth.

[1]To whom correspondence should be addressed. E-mail: noah.lefrancois@mail.mcgill.ca

zone thrown to.

(1) compared four models and determined that the Gradient Boosting Machine (GBM) and Random Forest (RF) models perform significantly better compared to both logistic regression and linear discriminant analysis. They hypothesized that this was "most likely because the latter two models attempt to find a linear decision boundary which does not need seem to lend itself well to this particular problem", and also implemented an ensemble model which combined the predictions of both the GBM and RF. (2) also used a GBM to predict run/pass data, discussing which features had the largest effect on outcome and how the accuracy of this model varied across all 32 NFL teams.

(3) tested a multi-layer perceptron algorithm for run/pass data and focused on a thorough investigation of how the model accuracy differs across situations, such as changing down & distance or field position.

(4) tested a hidden Markov model for predicting run/pass data and also discussed how the model accuracy varied across all 32 NFL teams. They found that accounting for the time-series structure of the data using this model improved their accuracy compared to other studies which they referenced.

(5) tackles an inverse problem from that which is the focus of this paper: predicting the expected value of prospective offensive plays in order to help offensive coordinators choose the best play call.

2. Methods

A. Data Pre-Processing.

Raw play-by-play situational data is stored by teams in a comma-separated values (CSV) file. The sports video analysis website Hudl used by most football teams at the high school and university level allows this data to be downloaded in the form of an Excel file (see Fig. 1) which can easily be converted to a CSV file. The situational information necessary to describe each play is:

· Offensive and Defensive teams
· Quarter
· Score Differential
· Situation
· Number of drives in game
· Number of first downs in drive
· Number of plays in drive
· Down and distance
· Field Position
· Offensive Personnel

QTR	SCORE DIFF. (O)	SITUATION (O)	DRIVE #	DRIVE PLAY #	1ST DN #	DN	DIST	D&D	YARD LN	Field Zone
1	0	OPENERS 1ST	1	4	1	2	2	2&2-	-52	Open Field (-40 to 40)
1	0		3	5	2	1	10	1&10	-52	Open Field (-40 to 40)
1	0		3	6	2	2	13	2&7+	-52	Open Field (-40 to 40)
1	0		4	1	0	0	10	0&10	-51	Open Field (-40 to 40)
1	0		2	2	1	1	10	1&10	-50	Open Field (-40 to 40)

Fig. 1. An example displaying the raw situational data in an Excel format before being exported to a CSV file.

Each play is also tagged with the play-calling outcome, or "play category", such as "run","PA pass", "dropback", etc.

If the down and distance are available separately in the raw data, they need to be grouped into intervals such as 2&3-, 2&4-7, 2&8+. Similarly the field position should be grouped into intervals such as "Open Field (-40 to 40)" and "Goal Line (5 to 0)".

The situation information identifies plays in the opening drive of a half, plays where the offensive team is losing in the final 2 minutes of a half ("2 minute drill") or winning in the final 4 minutes of a half ("4 minute drill"). All plays that do not fall into one of these situations is tagged with the label "REG".

Since the classification models will not accept strings as input, any information stored as strings needs to be relabelled using a dictionary mapping before training the model.

Finally, the data is separated into a training set and a testing set.

B. Classification Model.

The present model uses three classifiers from the sci-kit learn library as explored by (1) and (2): GBM, RF, and Extra Tress (ET) which is similar to the RF model. The results of these three classifiers are combined using the Voting Classifier (VC) module which outputs a weighted average of its input predictions.

The concept of combining multiple classification models is called an ensemble method. Ensemble methods are used to improve the robustness of a model when applied to new data sets compared to a single classifier. Robust performance on new data is an important consideration for our model since it is intended for application to any Canadian football league or conference and there is likely to be significant variation in offensive play calling across these data sets.

Training multiple classifiers does require increased initialization time for our application compared to the use of a single classifier, and opening the application takes approximately 60 seconds as a result. However, after the application is opened and the model has been trained, the time required to make a play prediction is nearly instantaneous. Each time that new plays are added to the model a short re-training period of approximately 10 seconds is required. This time may be slightly too long to make the feature usable after each play, but it could easily be used between offensive possessions which are usually between two and ten plays in length. This feature will allow the model to adapt to the opponent as a game goes on and more information is available.

When asked to predict a play, each classifier as well as the ensemble classifier outputs a list containing the probabilities of each possible outcome. In order to measure their performance, the rate at which their most likely outcome was the correct outcome was calculated, as well as the rate at which one of their top three most likely outcomes was correct. This gives a better sense of model accuracy for the application than a simple measure of whether their most likely outcome was correct since coaches tend to prefer to be prepared for a range of the most likely outcomes.

C. User Interface.

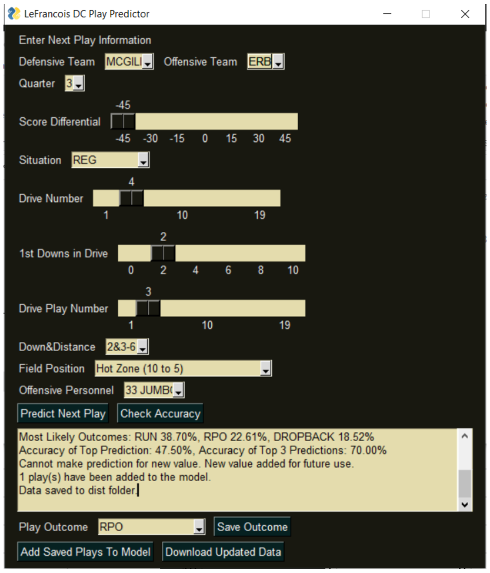

Fig. 2. User Interface for offensive play predictor. Predictions and other output messages are displayed in the text box at the bottom of the interface, alongside various buttons for controlling the application functions.

Fig. 2 shows the application's user interface. A number of drop-down boxes and sliders are provided to collect situational data from the user. These are easy and fast to update between plays; once set they remain the same until changed. This avoids asking the user to re-enter data that doesn't change frequently throughout the game, such as opponent or quarter, and is well-suited to the incremental changes seen in data columns such as drive number and number of 1st downs in drive.

Once the inputs have been set up, the top predicted outcomes can be printed using the "Predict Next Play" button; afterwards the outcome of the play can be recorded and saved to be added to the model later.

Possible input values for the drop-down menus are read from the data set to make sure that all inputs the model has seen are available. In the case of a previously unseen value, such as a new opponent or formation, this value can be typed into the box and the new play will be added to the model. After this, the new value will appear as an option in the drop-down menu and can be used to make predictions on future plays. This feature will allow coaches to start using our application even with a very small initial data set; although the accuracy may not be optimal initially in this case it will improve as more data is added.

At the end of a game, a CSV file containing all of the new plays seen during the session can be saved using the "Download Updated Data" button. The new data can then be combined with the original data for future use. In this way, the data set can be expanded throughout a season in order to progressively improve predictive performance.

3. Results and Discussion

The performance of the individual classifier models as well as the combined VC was measured while varying the number of plays available for training. Fig. 3 displays the model accuracy for the top choice as well as the top 3 choices, while Fig. 4 displays the time necessary to train the model. Each measurement was averaged over 10 different runs, each using a different partitioning of the available plays into training and testing data sets, in order to reduce statistical fluctuations. This method could be further improved with a larger base data set which would allow for multiple data sets of each size to be tested as well. It is important to note that this training time measurement was performed on a 2015 student laptop. While reasonable times can clearly be achieved using such a machine, a newer and more powerful computer would allow the use of much larger data sets before the training time becomes an issue for in-game usage.

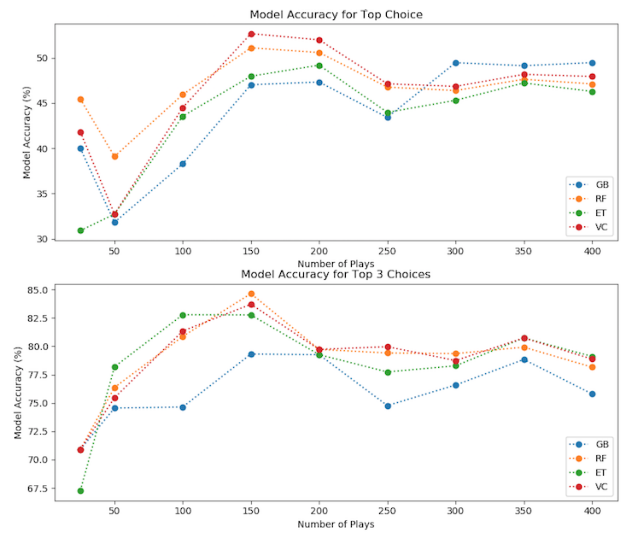

Fig. 3. Measuring prediction accuracy for classification models.

While each individual model has some areas where it performs better than the other two, no one model is dominant over the whole range of both accuracy plots. The combined VC model has more consistently strong performance than any of the individual models because it takes a weighted average, and in some intervals it actually performs better than any of its individual parts.

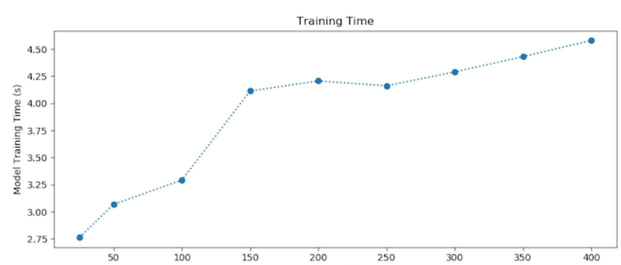

Fig. 4. Measuring training time for classification models.

The training time necessary for the combined model increases roughly linearly with the data set size. This indicates that a trade-off between accuracy and training time needs to be balanced if a user with a large data set wants to add new data to the model in-game. Based on the sample size of Fig. 3 (one team's plays over 9 games) it appears that there may be diminishing returns in accuracy with increasing data set size. While further testing needs to be done with larger data sets (such as full league data over one season) to corroborate this trend, this suggests that multi-year data for the full league would not be necessary for optimized performance of our application when taking training time into account.

It is important to note, however, that this training time constraint only affects the addition of new data to the model throughout a game. The time required to make a prediction once a model has been trained is less than one second and therefore any changes in prediction time due to larger data sets should be negligible for the purpose of in-game usage if no re-training is performed with new data.

One situation which could affect the accuracy of our model's predictions is a team changing offensive coordinators or making major playbook changes between seasons. Such a change would likely lead to very different play calling trends, which may not be as well-modelled by previous data. This change in play calling tendency can easily be handled by separating the data for the two coordinators into two different team labels in the raw data so that their different tendencies are each captured instead of averaging between them.

For example, if McGill changed coordinators between the 2019 and 2020 season and we have data for each season (or part of the 2020 season), then we would split the offensive team label from "McGill" into "McGill19" and "McGill20".

Finally, the ability to add new plays to the model actually means that this application could be used self-sufficiently to generate the necessary play-by-play data in the absence of an appropriate database already available to the user's team. While the data collected would be limited to the features specified by our application, this contains the majority of the necessary info for most teams' further analytical study. Any other features a coach may be interested in recording could easily be added as new columns to the data file using Excel, similarly to how data is already typically entered on Hudl.

A. Directions for Future Work.

There are a number of similar problems to which the present approach could be adapted in the future. The simplest would be to predict defensive play calls instead of offensive ones, in order to inform offensive coordinators. This would likely take the form of a prediction of the top three most likely combinations of defensive front and coverage, since this provides a more useful form of information to an offensive coach than either of those two items separately. In order to do so, an additional data pre-processing step would be required in order to take the information in the defensive front column and the coverage column and combine them.

Once this pre-processing is implemented, the interface should remain essentially unchanged since the same information is available pre-snap to both the offense and defence; the only difference is that the offensive personnel is decided by the offensive coordinator so he could get a prediction for each of the different personnel options he is considering.

One challenge with this change is that there is a much greater number of both coverages and fronts than there are offensive play categories, and so there will be significantly fewer data points for any given defensive play than for an offensive play. Initial testing of this concept has indicated accuracy rates of around 20-25% compared to a raw frequency of 15-18% for the most frequent play, and rates for the top three predictions of 45-65% compared to a raw frequency of 45-55%. While this suggests such an approach could be viable given improvements in the model and dimensionality reduction, it is not yet as powerful of a predictive tool as the offensive play prediction method.

Another usage of either the offensive or defensive prediction model is game planning analysis which takes place during the week between games instead of in-game. Coaches could create a mock game script which includes a number of situations they expect to encounter in the upcoming game and produce predictions for the outcome of each situation in order to inform and adjust their game plan. In this case, more input variables such as formation and defensive personnel could be used which we have chosen to exclude from our model since they are not available in-game until late in the play clock; for game planning purposes this consideration of prediction speed is not important. This would allow a more detailed analysis of opponent tendencies for the weekly scouting report in order to inform player intuition instead of only informing the coaches' decision making.

.

4. Conclusion

In this work the usability, accuracy, and robustness of an artificial intelligence agent for predicting offensive play calling and assisting the decision-making process of defensive coordinators has been demonstrated. Focusing on the merging of past theoretical investigations and the specific needs of the live in-game coaches' booth, this project is able to make powerful predictions using widely available data.

The self-sufficiency of this application as a data logging and analysis device, enabled by the ability to add new plays to the model in-game, makes it easily employable for any team with access to a laptop computer on game day and game film of their opponent. While performance and ease-of-implementation will improve with the larger league-wide database of labelled play-by-play data available to college and professional teams on Hudl, this certainly would not prevent a high school coach without that infrastructure from taking advantage of this predictive tool.

We have developed an intuitive, streamlined user interface with in-game use by football coaches in mind, removing the need for any technical programming or mathematical knowledge in order to benefit from the rapid development of machine learning technology in sports analytics.

ACKNOWLEDGMENTS. I would like to thank Julien Deschenes of the McGill Football coaching staff for his feedback on the needs of a coach using this program, providing training data for the building of this model, and input on the user interface visual design.

5. References

1. P. Lee, R. Chen, V. Lakshman (2016) Predicting Offensive Play Types in the National Football League. Stanford University Department of Computer Science.
2. A. Cheema (2019) Predicting NFL Offensive Play-Calling With Python. The Spax.
3. K.L. Ota (2017) Football Play Type Prediction and Tendency Analysis. MIT Department of Electrical Engineering and Computer Science.
4. M. Otting (2020) Predicting play calls in the National Football League using hidden Markov models. Bielefeld University.
5. B. Teich, R. Lutz, V. Kassarnig (2016) NFL Play Prediction. College of Information and Computer Sciences University of Massachusetts Amherst.