Personalized Cycling Training Plan Generator Using Strava Activity Data

Rhishabh Hattarki
Arizona State University
Arizona, USA
rhattark@asu.edu

***Abstract*-Cycling, a sport embraced by enthusiasts ranging from casual riders to competitive athletes, demands personalized training plans for performance enhancement. This paper presents an innovative data-driven methodology that harnesses historical Strava activity data to craft individualized cycling training plans. Our system executes a two-step data mining pipeline, integrating ride data clustering and personalized plan classification. The resulting tailored plans cater to individual cyclist needs, obviating the necessity for personal coaching. This research contributes a scalable and accessible solution to the realm of personalized training plans, serving a diverse spectrum of cyclists seeking performance optimization.**

I. Introduction

A. Background:

Cycling, a widely embraced sport for both leisure and competition, has spurred a quest for refined strategies to elevate performance. The emergence of modern athlete-centric platforms, exemplified by Strava, opens new avenues for leveraging personal activity data to intricately tailor training plans for individuals. This project aims to delve into the intricate realm of performance enhancement, more specifically in cycling by harnessing the wealth of information offered by platforms like Strava, providing a foundation for the creation of highly personalized and effective training regimens.

B. Problem:

The challenge arises when cyclists, seeking performance improvement, encounter the complexity of formulating tailored training plans. For those without access to personal coaching, navigating this intricacy can be particularly daunting. This paper addresses this challenge by proposing a data-driven approach to generate personalized cycling training plans.

C. Importance:

The value of personalized training plans extends beyond novice guidance, presenting seasoned cyclists with diverse perspectives to enhance their existing knowledge base. By tailoring recommendations to individual needs, this project not only serves as a valuable resource for those new to cycling but also acts as a strategic tool for experienced practitioners, offering them nuanced approaches to optimize and augment their performance trajectories.

D. Existing Literature:

While the concept of personalized training plans is not novel, existing approaches often require extensive manual input or lack the specificity required for individual athletes. Our work builds upon this foundation, introducing a data-

driven methodology that leverages historical Strava activity data for enhanced accessibility. There is limited research available on automated training plan generators and more information is provided in the Related Work section.

E. System Overview:

Our proposed system encompasses a two-step data mining pipeline crafted to discern and capitalize on discernible training patterns within a cyclist's historical activities. In the initial step, we employ ride data clustering to categorize rides into distinct groups, with a particular emphasis on identifying recurring patterns within training blocks, typically spanning a 4-week duration [1]. These clusters serve to organize and label the previously unlabeled data based on the specific type of training undertaken during each block.

Subsequently, in the second step, we leverage classification techniques on the labeled data. This involves training a classification model capable of categorizing rides based on their training block labels. The model is then adept at generating tailored training plans, factoring in user-defined parameters such as training duration and frequency. By first clustering and labeling the data, our system effectively transforms previously unstructured ride information into a labeled dataset, enabling the subsequent classification model to provide personalized training recommendations aligned with the user's specific input criteria.

F. Data Collection:

The primary data source for this study is the cyclist's own Strava activity data, spanning over seven years. Strava's developer APIs provide a seamless means of accessing and extracting the rich dataset required for the data-driven training plan generator [2]. More specifically the activities and zone APIs were used to construct the dataset that this project is based on.

G. Components of the ML System:

The machine learning system is comprised of two core components, each employing distinct data mining techniques. The first component focuses on ride data clustering, utilizing the k-means clustering algorithm to discern inherent patterns within historical cycling activities. Additionally, Agglomerative Clustering and MiniBatchKmeans were explored as alternative clustering methods.

The second component centers on classification, wherein the identified patterns from clustering are transformed into personalized training plans. Support Vector Machine serves as the primary classification algorithm in this context. Furthermore, the efficacy of alternative classification algorithms, including Decision Tree, Random Forest Classifier, Logistic Regression, and K-nearest neighbors, was evaluated through systematic testing.

H. Experimental Results:

In subsequent sections, we present the methodology employed in detail, delineating the intricacies of the data mining processes and the system's architecture. Additionally, we provide experimental results, offering insights into the effectiveness and practicality of our proposed approach.

II. Definitions and Problem Statement

A. Definitions

1) Data: In the context of this study, the term "data" pertains to the extensive compilation of cycling activities recorded on the Strava platform by the user. This dataset encompasses a diverse set of parameters, including but not limited to ride duration, distance covered, elevation gain, heart rate, and various other relevant metrics. Initially comprising 56 features, the dataset was expanded to 72 features through the inclusion of zone data, and subsequently streamlined to 6 features during the feature extraction process specifically designed for clustering purposes. Later the classification was done using 2 features.

2) Prediction Target: In this study, the "prediction target" is defined as the personalized training plan created for an individual cyclist, derived from their historical Strava activity data. To simplify the initial product, the prediction target was distilled into a combination of average_watts and suffer_score. The resulting binary representation of classes adopts a format such as 0_1, where the first bit signifies average_watts and the second bit signifies suffer score. A value of 0 represents low, while 1 represents high for each respective metric.

3) Variables or Concepts in the Data: Key variables or concepts within the dataset include ride moving_time, timestamp, average_speed, average_watts, and suffer_score [3]. These variables serve as foundational elements for discerning patterns and constructing a personalized training plan tailored to the cyclist. Moving time corresponds to the total duration of riding during the activity, while the timestamp denotes the start time of the activity. Average speed represents the overall average speed of the entire activity, calculated based on the moving time. Average watts quantify the cyclist's average power output in watts throughout the activity. Suffer score, a metric generated by Strava, reflects the degree of exertion or training stress experienced by the cyclist during the activity, providing a nuanced measure of their physical strain.

B. Problem Statement

1) Given: The presented scenario revolves around cyclists expressing a need for personalized training plans to optimize their performance. The dataset at our disposal comprises historical Strava activity data, encapsulating intricate details of individual rides. This dataset forms the bedrock for the development of customized training plans. It is presupposed that a substantial volume of data, spanning a minimum of one year, is available for analysis, inclusive of GPS and power data.

2) Objective: The primary objective is to develop a data-driven system capable of generating personalized cycling training plans. The system should leverage the available Strava activity data to identify patterns, cluster relevant training features, and subsequently produce a tailored regimen that optimally suits the individual cyclist's needs. The simplified model should be able to provide whether the average power should be high or low and [4]whether it should be a hard or easy. This will be output in the format given in prediction target.

3) Constraints: The key constraints include the reliance on historical Strava activity data for a specific user, necessitating the use of Strava's developer APIs for data access. Additionally, the system requires considerable amount of data, at least a year’s worth, like mentioned in the given section. It also requires that the cyclist has uploaded the activities with power data, which can be collected using a power meter. It also prefers having GPS data as opposed to manual entries.

III. Overview of Proposed System

A. Data Mining Pipeline

1) Strava Authentication: The initial phase of our data mining pipeline initiates with Strava authentication. Users authorize access to their Strava accounts, enabling the system to fetch their historical activity data through Strava's developer APIs *[4]*. During the authentication process, a code is generated, facilitating the retrieval of an access token by calling a subsequent API. This access token, functioning as a Bearer token, grants the system continued access to Strava data through various APIs.

2) *Data Collection via Strava API:* Following authentication, the system employs Strava's API to gather an extensive dataset of the user's cycling activities. The key APIs utilized for this purpose are "list athlete activities" and "get activity zones" *[3] [5]*. These APIs enable the retrieval of essential data, encompassing, among other parameters, distance, moving time, average watts, and various zones pertinent to the cycling activities.

3) Data Preprocessing and Cleaning: Raw data collected from Strava may contain inconsistencies, outliers, or missing values. In this step, we perform thorough data preprocessing and cleaning. This involves handling missing values, standardizing data formats to ensure uniformity across all features. The goal is to prepare a clean and reliable dataset for subsequent analysis.

4) *Feature Extraction:* In the feature extraction phase, we identify pertinent attributes from the preprocessed data that play a significant role in characterizing cycling activities. These relevant features, such as ride duration/moving time, average watts, and suffer score, are selected to reduce dimensionality while retaining essential information crucial for subsequent clustering and classification processes. Further details on this process are elaborated in the technical details section below.

5) *Clustering:* The clustering phase uses unsupervised learning techniques to group similar cycling activities. This process allows the system to help create personalized training plans by dividing cycling activity into blocks that typically span four weeks. These blocks contain a mix of easy and challenging rides over several weeks. Essentially, this process transforms unclassified data into labeled datasets, each of which is assigned specific training characteristics. 3 different clustering techniques were tested to get the best results. More information about this process is provided in the Technical Details section.

6) Classification: After the clustering phase, the system proceeds to employ classification algorithms for the purpose of categorizing cycling activities into distinct training plans. This classification step enhances the system's capacity to customize training plans according to individual preferences and performance objectives. As part of this simplified model, the classifier identifies a training label, representing a composite of average watts and suffer score. The labels adhere to a specific format, such as <average_watts>_<suffer_score> (e.g., 1_0, where 1 denotes high and 0 denotes

low). The primary model used was Support Vector Machine, and this was tested against 4 different baseline classification methods. More information about this is provided in the Technical Details.

7) *Training Plan Generation:* Utilizing the established classification model, the generation of a training plan requires user input specifying the duration of the plan and the frequency of rides. The model, upon receiving these inputs, proceeds to classify and provides the resultant training plan encoded within a specific label. For instance, if the output label is 1_0, it signifies an interpretation where average watts are designated as 1 and suffer score as 0. In practical terms, this implies that the training plan should emphasize high power output while minimizing training stress.

IV. Technical Details of Proposed System

A. Data Exploration and Feature Extraction

We initiated our exploration by gaining a comprehensive understanding of the dataset. Notably, we plotted the average speed over time, revealing a pattern where initial years exhibited lower speeds, followed by a steady increase around 2018, and subsequent reduction associated with a decrease in training load.

With 56 initial features and the addition of zone data, our dataset expanded to 72 features. However, recognizing the necessity for feature reduction, we applied a two-fold approach. First, features such as activity id and user id, with all unique values, were removed. Subsequently, columns like city, state, and country were excluded to prevent model dependency on location, considering the unlikelihood of users providing or receiving such information.

Further refinement involved removing Boolean values like has heartrate, private, and visible, as they could be deduced from other columns. Standardization was then applied to ensure uniform units, for example, converting average speed from meters/second to kilometers/hour for better interpretability.

Even after these steps, we were left with 10 columns. To identify and eliminate redundant features, we employed a correlation matrix, resulting in a streamlined set of 6 features crucial for the subsequent clustering phase. The Fig 2 shows the correlation matrix where moving time, total elevation gain, kilojoules and distance are correlated, so all but one were removed.

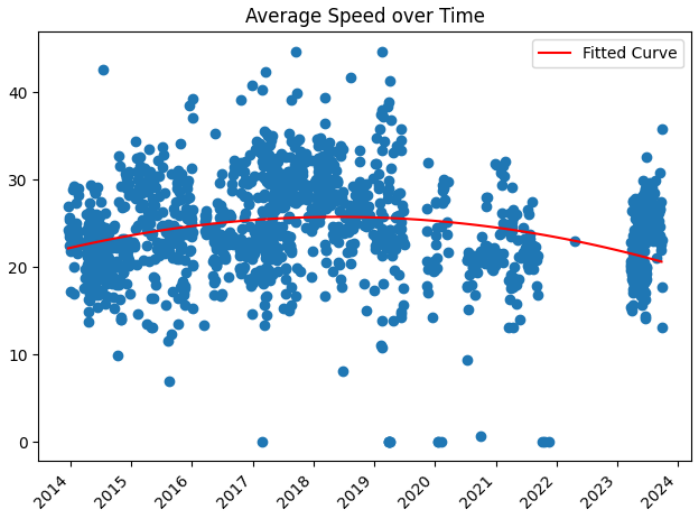

Figure 1. Plot of changing average speed of the cyclist over time from 2014 to 2023

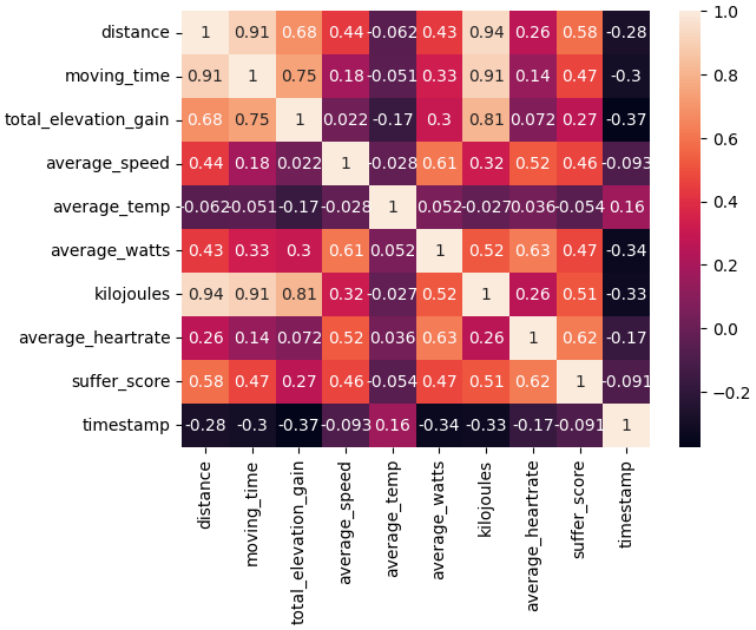

Figure 2. Correlation matrix of features, used to understand redundant features

B. Clustering

The objective of clustering was to categorize rides into distinct training plans for subsequent classification. Commencing the clustering process, we applied k-means clustering to the 6 extracted features. To determine the optimal k value, we initially plotted an elbow curve. The curve indicated that the k value began to stabilize around k = 20. However, given the seven-year span of the data, limiting clusters to 20 would result in grouping training plans over a few months, contrary to the typical 4-week duration of a training block.

To address this, we calculated the number of blocks within the 7-year dataset and utilized this count to create an equivalent number of clusters. This approach ensures that each cluster aligns with the duration of a standard training block. Figure 4 illustrates the clustering outcomes using this technique.

Furthermore, the exploration of alternative clustering techniques aimed to assess their efficacy compared to KMeans. Despite the evaluation of AgglomerativeClustering and MiniBatchKMeans, the silhouette scores consistently favored KMeans as the superior performer. Given that the silhouette score serves as a metric for assessing cluster separation, the consistently higher score of KMeans signifies its proficiency in generating more distinct and well-separated clusters compared to the alternatives [6].

TABLE I
COMPARISON OF CLUSTERING ALGORITHMS

Algorithm	Silhoutte Score
KMeans	0.24232654443637372
AgglomerativeClustering	0.2398740972133002
MiniBatchKMeans	0.23041524809378003

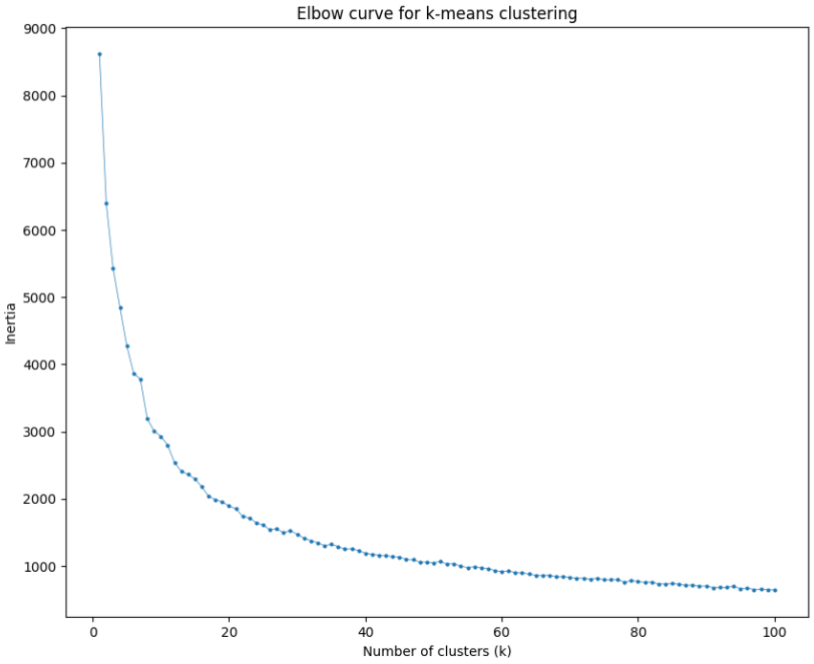

Figure 3. Elbow curve for k-means clustering

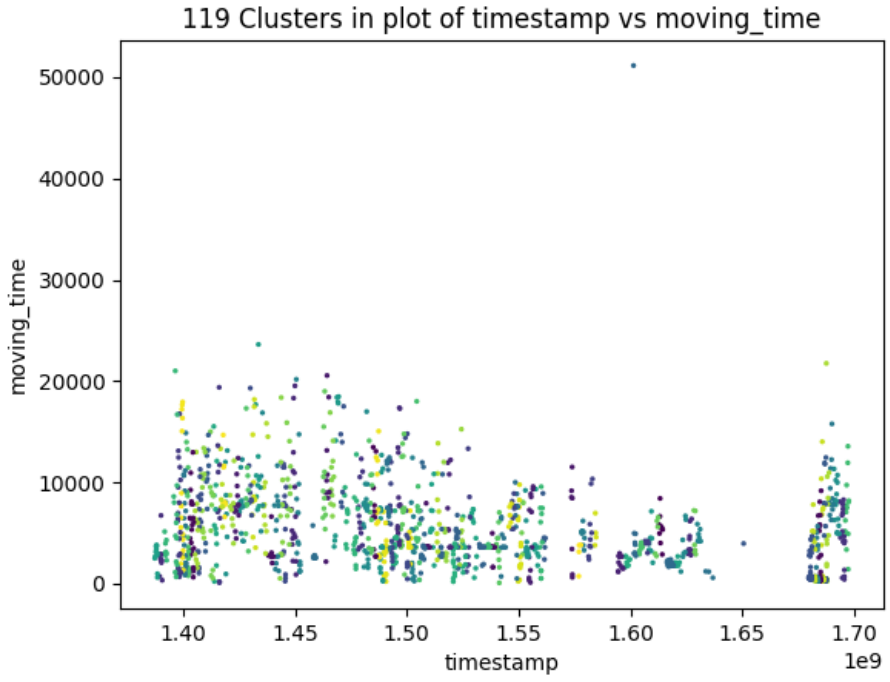

Figure 4. 119 clusters in plot of timestamp vs moving_time

C. Classification

The objective of this phase was to leverage the previously created clusters, assign them labels based on their contained training characteristics, and subsequently train a classification model. The model aimed to take user-inputted duration and frequency and offer a corresponding training plan through classification. To simplify the labeling process while retaining essential training plan information, the labels were formulated as a combination of two continuous columns, average_watts, and suffer_score.

Concatenating these columns would result in 119 unique training plans, making the model essentially a straightforward mapping operation. To facilitate grouping of similar training plans, the continuous values from the two columns were abstracted into binary categories: high and low. Consequently, the label encoding became a representation of both columns, where 1 denoted high and 0 denoted low. For example, a label of 0_1 signified low average watts and high suffer score, indicative of a low power high volume training plan.

This labeled dataset was employed for classification, with the primary classifier being Support Vector Machine. Additionally, four baseline algorithms—Logistic Regression, Decision Tree, Random Forest Classifier, and K-nearest Neighbors—were included for comparative analysis. The models were evaluated based on two metrics, accuracy, and F1 score [7].

In the evaluation, Support Vector Machines demonstrated the highest performance with an accuracy of 52.78%, closely followed by Logistic Regression. In contrast, K-nearest neighbors exhibited the lowest accuracy at 33.33%. The choice of Support Vector Machines as the primary classifier was substantiated by its superior performance in both accuracy and F1 score metrics.

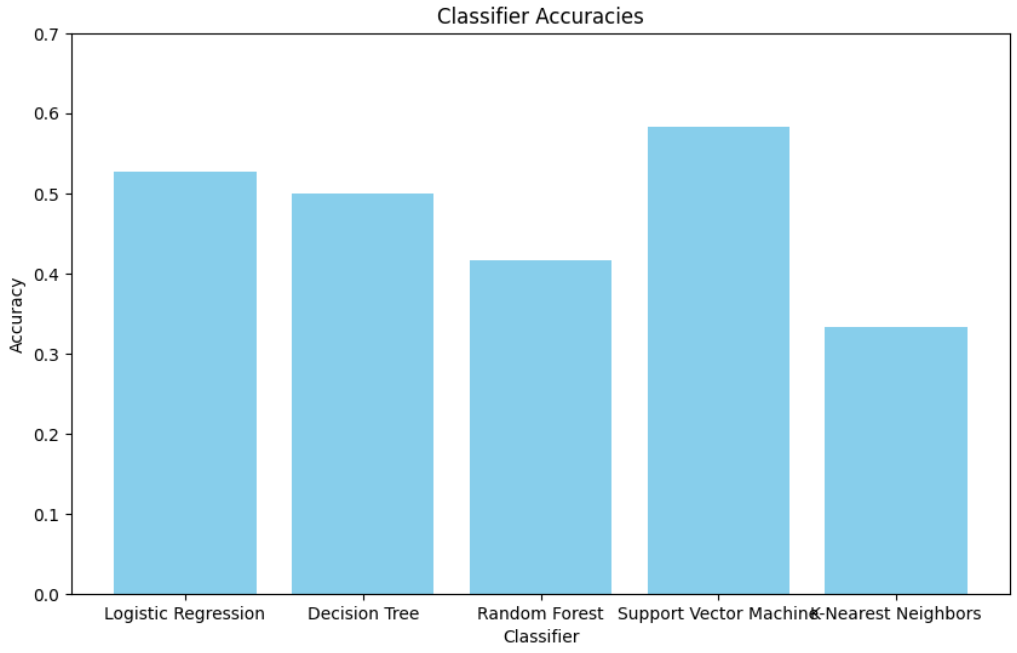

Figure 5. Comparison of Classifiers

V. Experiments

A. Oversampling

Upon recognizing a slight imbalance in class labels within the dataset, an experimental approach was undertaken to assess the impact of oversampling on model performance. The Synthetic Minority Oversampling Technique (SMOTE) was employed to synthetically augment the instances of the minority class, aiming to alleviate the imbalance and potentially enhance model outcomes [8]. Subsequently, the classification process was reiterated across all models.

The outcomes revealed that the oversampling technique led to improved accuracies in the Random Forest and K-Nearest Neighbors models. However, contrary to expectations, this had a divergent effect on the other models, resulting in either marginal improvement or a reduction in accuracy.

B. Data augmentation

To enhance the overall robustness of the training dataset, a comprehensive resampling approach was undertaken. This involved augmenting the entirety of the dataset by introducing small amounts of noise to each new data point, thereby increasing the overall size of the training dataset. Unlike a targeted focus on minority classes, this strategy aimed at introducing diversity across all classes.

Upon empirical evaluation, it was observed that this augmentation methodology did not yield discernible improvements in the performance of Support Vector Machine (SVM), Logistic Regression, or Decision Tree models. However, a contrasting effect was observed with Random Forest and K-Nearest Neighbors, where the augmentation led to enhanced model performance.

TABLE II
CLASSIFICATION EXPERIMENTS

Algorithm	Initial Accuracy	Oversampling accuracy	Augmentation accuracy
Logistic Regression	52.78	38.89	52.78
Decision Tree	50	44.44	50
Random Forest Classifier	41.67	44.44	44.44
Support Vector Machine	58.33	41.67	52.78
K-Nearest Neighbors	33.33	36.11	41.67

VI. Related Work

In the realm of training plan generation, various studies have explored diverse approaches and techniques, offering a spectrum of insights and outcomes. Notably, the work by Rikard Eriksson and Johan Nicander stands out for its utilization of advanced machine learning methods, specifically genetic algorithms, and random tree regressor chains [9]. Their detailed weekly training plans, tailored for swimmers, present an exemplary outcome. However, it's crucial to note that their methodology differs substantially from the more straightforward algorithms employed in the present study, and the data source is distinct, focusing on swimming without reference to Strava-generated data.

Iztok Fister Jr. and Iztok Fister approached training plan generation through swarm intelligence, leveraging GPX/TCX files [10]. While their output, a weekly training plan providing average heart rate and duration per day,

aligns with the goals of this project, disparities emerge in terms of data generation strategy, algorithms utilized, and the specificity of the output.

The work by Tomas Skerik, Lukas Chrpa, Wolfgang Faber, and Mauro Vallati proposed an automated training plan generator applicable to athletes, yet it diverges in its focus on a broader athletic context and doesn't address cycling or incorporate Strava data [11]. Similarly, Alessandro Silacci, Redha Taiar, and Maurizio Caon presented an AI-based training plan generator for road cyclists using Reinforcement Learning [12]. Despite achieving commendable results, their plans exhibited susceptibility to overtraining, and their adjustability throughout the training plan was limited.

The conceptual framework put forth by Laila Zahran, Mohammed El-Beltagy, and Mohamed Saleh introduces adaptability to training plans, a feature not commonly observed in static plans from other studies [13]. Nevertheless, this framework remains generic, lacking a specific focus on cycling and excluding considerations for Strava data.

In the broader landscape, research on training plan generation appears limited, marked by variations in data sources, techniques, output specificity, and overall efficacy. This diversity underscores the evolving nature of this field and the need for tailored approaches to address the nuances of different athletic domains and data types.

VII. Conclusion

In conclusion, this paper presents a novel approach to generating personalized cycling training plans using historical Strava activity data. Leveraging the wealth of information accessible through Strava's developer APIs, our two-step data mining pipeline involving ride data clustering and classification demonstrates a promising methodology. The clustering phase successfully organizes and labels cycling activities into distinct training blocks, providing a foundation for personalized recommendations. Subsequently, the classification step, led by Support Vector Machine, refines these blocks into specific training plans and classifies them based on user-defined parameters.

Comparing our work with existing literature underscores the uniqueness of our data-driven methodology and its applicability to the cycling domain. While other studies often involve complex machine learning techniques or focus on different sports, our approach stands out for its simplicity, accessibility, and specificity to cycling, making it a valuable tool for cyclists ranging from beginners to seasoned athletes.

While our system demonstrates promising results, it is essential to acknowledge certain limitations. The reliance on historical Strava activity data assumes a minimum of one year's worth of information, and the effectiveness may vary for users with less extensive datasets. Additionally, the need for GPS and power data, preferably collected through a power meter, adds constraints on data requirements. User engagement with Strava, ensuring consistent and accurate activity logging, also influences the system's reliability. Moreover, currently only a summary of the classified training plan can be provided instead of having detailed activities per week for an entire block.

Looking ahead, the proposed system opens avenues for further refinement and expansion. The integration of additional features or the exploration of more advanced machine learning techniques could enhance the system's precision. Additionally, user feedback and iterative improvements can contribute to tailoring the system to diverse cycling preferences and performance goals. A combination of multiple models to classify or predict specific values of a training plan could also be explored.

In essence, our research contributes to the evolving landscape of personalized training plans, offering a scalable and accessible solution for cyclists seeking individualized strategies to enhance their performance. The intersection of data-driven methodologies and the cycling community holds promise for the continual evolution of training plan generators, empowering cyclists of all levels to optimize their riding experiences.

REFERENCES

[1] J. Friel, The Cyclist's Training Bible, Velo Press, 2018.

[2] Strava, "Strava Developers," [Online]. Available: https://developers.strava.com. [Accessed November 2023].

[3] Strava, "Strava Developers List Athlete Activities," [Online]. Available: https://developers.strava.com/docs/reference/#api-Activities-getLoggedInAthleteActivities. [Accessed November 2023].

[4] Strava, "Strava Authentication," [Online]. Available: https://developers.strava.com/docs/getting-started/. [Accessed November 2023].

[5] Strava, "Strava Developers Get Activity Zones," [Online]. Available: https://developers.strava.com/docs/reference/#api-Activities-getZonesByActivityId. [Accessed November 2023].

[6] K. R. Shahapure and C. Nicholas, "Cluster Quality Analysis Using Silhouette Score," in *2020 IEEE 7th International Conference on Data Science and Advanced Analytics (DSAA)*, Sydney, NSW, Australia, 2020.

[7] M. Sokolova, N. Japkowicz and S. Szpakowicz , "Beyond Accuracy, F-Score and ROC: A Family of Discriminant Measures for Performance Evaluation," in *AI 2006: Advances in Artificial Intelligence*, 2006.

[8] A. Fernandez, S. Garcia, F. Herrera and N. Chawla, "SMOTE for Learning from Imbalanced Data: Progress andChallenges, Marking the 15-year Anniversary," *Journal of Artificial Intelligence Research* , vol. 61, pp. 863-905, 2018.

[9] R. Eriksson and J. Nicander, "Automated Generation of Training Programs for Swimmers," Chalmers University of Technology, Gothenburg, Sweden, 2021.

[10] I. Fister Jr. and F. Iztok, "Generating the Training Plans Based on Existing Sports Activities Using Swarm Intelligence," *Nature-Inspired Computing and Optimization. Modeling and Optimization in Science and Technologies,* vol. 10, p. 79–94, 2017.

[11] T. Skerik, L. Chrpa, W. Faber and M. Vallati, "Automated Training Plan Generation for Athletes," *2018 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pp. 3865-3870, 2018.

[12] A. Silacci, R. Taiar and M. Caon, "Towards an AI-Based Tailored Training Planning for Road Cyclists: A Case Study," *Applied Sciences*, vol. 11, no. 1, 2021.

[13] L. Zahran, M. El-Beltagy and M. Saleh, "Generation of Adaptive Training Plans," *International Conference on Advanced Intelligent Systems and Informatics 2019. Advances in Intelligent Systems and Computing,* vol. 1058, p. 673–684, 2019.

GITHUB LINK TO PROJECT

https://github.com/rhattark/CSE572-project