

---

# ATTENTION IS ALL YOU NEED, FOR SPORTS TRACKING DATA

---

PREPRINT

**Udit Ranasaria**

SumerSports

udit.ranasaria@sumersports.com

**Pavel Vabishchevich**

SumerSports

pavel.vabishchevich@sumersports.com

August 20, 2024

## ABSTRACT

The rapid advancement of spatial tracking technologies in sports has led to an unprecedented surge in high-quality, high-volume data across all levels of play. While this data has catalyzed innovations in sports analytics, current methodologies for team sports often struggle with the inherent challenge of the player-ordering problem. This paper highlights the application of Transformer architectures and Attention to address these challenges in sports analytics. Our approach operates end-to-end on raw player tracking data, processes unordered collections of player vectors, and is inherently designed to learn pairwise spatial interactions between players. The framework satisfies critical criteria for widespread adoption in sports modeling: minimal feature engineering, adaptability across diverse problems, and accessibility in terms of understandability and reproducibility. To demonstrate its effectiveness, we apply our approach to the task of predicting tackling location in the NFL, a problem recently explored in the public domain. Our results show significant improvements over commonly used approaches, particularly in generalizing to diverse game situations. This work aims to catalyze a paradigm shift in sports analytics research methodologies, moving from traditional models to Transformer-based architectures. The potential implications include unlocking new insights into player dynamics, team strategies, and game outcomes across various sports domains, paving the way for more sophisticated deep learning models in sports analytics.<sup>1</sup>

**Keywords** sports analytics · sports tracking data · latent representation learning · multi-agent spatiotemporal modeling

## 1 Introduction

The field of sports analytics has experienced rapid growth, driven by unprecedented advancements in spatial tracking technologies. High-quality, high-volume data acquisition—facilitated by optical systems, computer vision algorithms, and chip-based tracking solutions—is now pervasive across all sports and levels of play. This confluence of rich data with innovative research and sophisticated modeling techniques has precipitated disruptions in multiple domains, including sports science, broadcasting, player evaluation, team building optimization, injury prevention, and tactical strategy formulation. ? comprehensively summarizes the burgeoning corpus of innovative research contributions leveraging sports tracking data, with the most advanced and modern modeling emphasis in Latent Variable Estimation, Event Prediction, and Value Attribution. Despite the sophistication of these methodologies, their potential is often hampered by reliance on traditional approaches to circumvent the *player order problem*. This challenge arises from the dynamic nature of team sports, where player roles and formations are inconsistent and can vary between games. The absence of a persistent intrinsic order of players in different intervals of play or games conflicts with the input structure requirements of many standard machine learning models, necessitating hand-crafted feature preprocessing steps to transform raw tracking data into structured feature representations or the imposition of heuristic-based orderings (?). These hand-designed feature extractors or ordering approaches, while functional, rely on domain expertise and lack generalizability. Moreover, with the proliferation of deep learning, feature engineering has become an anti-pattern with preference given to *end-to-end learning* (??). Advocates for end-to-end learning emphasize that, given sufficient data,

---

<sup>1</sup>Our code is openly available here

models should aim to learn latent features directly from raw inputs, optimizing against training objectives. Consequently, modeling advances in deep learning research typically stem from architectural innovations tailored to the data space rather than feature extraction techniques. Furthermore, we posit that the majority of modeling tasks involving sports player tracking share a fundamental objective: learning pairwise spatial interactions between players. Models adept at capturing these interactions are likely to excel in event prediction and other supervised tasks that necessitate a nuanced understanding of player positioning within the context of sport-specific rules and dynamics. This observation motivates our exploration of novel architectural approaches that can inherently handle unordered sets of player data while capturing complex spatial relationships. To address these challenges, this paper highlights the application of Transformer architectures as a modeling framework for sports analytics. Transformers, originally developed for natural language processing tasks, have shown remarkable capabilities in handling sequential and unordered data across various domains. Our proposed approach satisfies several critical criteria:

1. End-to-end operation on raw player tracking data with minimal feature engineering, ensuring flexibility and adaptability across diverse sports modeling problems.
2. Capability to process unordered collections of player vectors, directly addressing the player-ordering problem.
3. Inherent design for learning pairwise player interactions.
4. Accessibility in terms of understandability, explainability, and reproducibility.

Our objective is to catalyze a paradigm shift in sports analytics research methodologies. We anticipate a transition from traditional models like XGBoost and MLPs, commonly employed in competitions such as the NFL Big Data Bowl, towards variants and extensions of Transformer architectures. This shift has the potential to unlock new insights in player dynamics, team strategies, and game outcomes, ultimately advancing the field of sports analytics and its applications across various domains.

## 2 Methods

### 2.1 Sports Tracking Data

Table 1: Example of a Multi-Agent Tracking Frame from the NFL

frame_id	event	nfl_player_id	team	x	y	s	o	dir
15	handoff	34452	OFF	26.87	27.9	3.0	274.91	242.98
15	handoff	40089	OFF	35.19	34.03	0.51	246.43	84.06
15	handoff	42368	DEF	40.37	25.44	6.15	304.19	297.75
⋮								
15	handoff	54948	DEF	35.33	31.36	2.67	272.22	261.64

Sports tracking data provides rich spatiotemporal information about player movements and game events. This data typically includes:

- A frame number or time column to track the temporal progression of events
- A unique player identifier for each athlete on the field
- An indicator for team affiliation (e.g., offense or defense)
- An event stream that annotates specific "actions" occurring at given moments in the game
- Feature columns representing spatial properties of each player, such as position and velocity

Table ?? presents a sample of data from the 2024 NFL Big Data Bowl, illustrating these key components.

In this paper, we focus on modeling the static *multi-entity* aspect of tracking data that exists within a single frame (or timestamp). We treat each tracking frame at a given timestamp as a unique, independent training sample, disconnected from the frames temporally surrounding it. This approach allows us to focus the discussion around the spatial relationships between players at specific moments, which is crucial for many predictive tasks in sports analytics.

## 2.2 Task Formulation

To facilitate a comparative discussion of various modeling approaches, we first establish a general mathematical framework for learning from sports tracking data. Let  $P = p_1, p_2, \dots, p_K$  represent the set of  $K$  players participating in a particular frame. Similarly, let  $V = v_1, v_2, \dots, v_K$  be the set of feature vectors such that each  $v_k \in \mathbb{R}^d$  captures all relevant spatial (e.g., position and velocity) and characteristic (e.g., height and weight) features for the player  $p_k$ . Crucially, both  $P$  and  $V$  are *unordered* sets, reflecting the absence of intrinsic order of players in most team sports. We define our supervised learning task over  $V$  as follows: Let  $y \in \mathcal{Y}$  be the objective label we aim to predict, where  $\mathcal{Y}$  is the set of possible outcomes. This could represent various tasks such as predicting future events or outcomes. We define a model  $f : (\mathbb{R}^d)^K \rightarrow \mathcal{Y}$  as a function that maps the set of feature vectors  $V$  to the label space  $\mathcal{Y}$ . Formally,

$$\hat{y} = f(V) = f(v_1, v_2, \dots, v_K) \quad (1)$$

where  $\hat{y}$  is the predicted label. To train this model, we define a loss function  $\mathcal{L} : \mathcal{Y} \times \mathcal{Y} \rightarrow \mathbb{R}$  that measures the discrepancy between the true label  $y$  and the predicted label  $\hat{y}$ . The optimization problem can then be formulated as:

$$f^* = \arg \min_f \mathbb{E}_{(V,y) \sim \mathcal{D}} [\mathcal{L}(y, f(V))] \quad (2)$$

where  $\mathcal{D}$  represents the underlying data distribution from which our training samples are drawn. This formulation describes a general framework for modeling on unordered sets of player data. In the following sections, we will discuss various prior approaches to ascribing a model to  $f$ , highlighting their strengths and limitations in handling the player-ordering problem. This will set the stage for our proposed Transformer-based approach, which we argue offers a more flexible and effective solution to learning from unordered sports tracking data.

## 2.3 Decomposition Approaches in Prior Work

As discussed in the Introduction??, often sports researchers identify this player ordering issue and then decompose the modeling as:

$$f(V) = g(\Phi(V)) \quad (3)$$

where  $\Phi : (\mathbb{R}^d)^K \rightarrow \mathbb{R}^m$  is a feature extraction or player ordering process that maps the unordered set of player feature vectors to an ordered fixed-dimensional representation, and  $g : \mathbb{R}^m \rightarrow \mathcal{Y}$  is typically implemented using MLPs or gradient boosted tree models. Applying a fixed ordering over  $V$  based on domain heuristics is still considered a special case of  $\Phi$  where:  $\Phi_{\text{fixed}} : (\mathbb{R}^d)^K \rightarrow \mathbb{R}^{d \cdot K}$ .

Several notable examples in the literature follow this decomposition paradigm:

- Both ? and ? propose novel frameworks to decompose complex sports like soccer and American football into continuous time value-based metrics. Both papers invest heavily in deriving "a wide set of spatio-temporal features" to feed individual models that comprise the frameworks.
- ?, in building a model to infer ball location from tracking data, identify that "it is impossible to find a correct ordering for the individual players to be represented in the feature matrix" and implement a segment-based role assignment algorithm to fix an order.
- ? and ? employ deep imitation learning for "ghosting" and team-strategy evaluation in soccer and American football, respectively. To featurize a tracking frame as input into recurrent nets, they rely on a role-based assignment step to "impose ordering on the training input".
- ? built a conditional variational auto-encoder model capable of synthetically generating basketball player trajectories conditioned on identity and context. They separately develop an algorithm to solve the "significant challenge in encoding multi-agent trajectories is the presence of permutation disorder".
- ? innovates with convolutional network filtering over the time dimension but uses an anchor-based sorting scheme to avoid "implicitly enforc[ing] an order among this set of players".

This collection, while not exhaustive, illustrates a common pattern in advanced sports research: encountering the player-ordering problem and addressing it through various feature engineering or ordering schemes. These approaches, while effective for specific tasks, often lack generalizability and may not fully capture the complex spatial relationships inherent in sports tracking data.

Given the trends in deep learning towards end-to-end solutions, we argue that an approach capable of discovering latent features directly from the data, without the need for explicit feature engineering or ordering, could potentially outperform these methods across a wider range of tasks.

## 2.4 Generalized Transformer Model Architecture

The Transformer architecture, introduced by ?, revolutionized Natural Language Processing by introducing a self-attention mechanism that enables learning from direct pairwise interactions between all elements in a sequence, regardless of their order. This approach effectively addresses the challenge of modeling long-range dependencies, a limitation of commonly used recurrent models.<sup>2</sup> Crucially, the overall Transformer architecture maintains permutation equivariance: any permutation of the input sequence results in the same permutation of the output embeddings, without affecting the values of the embeddings in any way. This property is *exactly* what is needed to directly learn over the unordered set of player feature vectors end-to-end while capturing player interaction relationships. We define our Transformer-based model as:

$$f(V) = g(\text{TransformerEncoder}(V)) \quad (4)$$

where the TransformerEncoder function can be expressed as:

$$\text{TransformerEncoder}(V) = \text{LayerNorm}(V + \text{FFN}(\text{LayerNorm}(V + \text{MultiHead}(V, V, V)))) \quad (5)$$

The function  $g$  is a problem-specific "decoder" pooling + MLP layer that maps the Transformer Encoder's learned salient latent player embeddings to the desired label space  $\mathcal{Y}$ . The pooling operation is necessary for problems where when we need to eventually aggregate information across all latent player embeddings for a single shared prediction. In Figure ??, we visualize this architecture at a high level, demonstrating its generalizability across problems in Sports Tracking. The key advantage of this architecture lies in its self-attention mechanism. As visualized in the center of Figure ??, each player vector is updated with information from every other player in a parameterized manner. This allows the model to learn to identify important patterns for the objectives, creating rich embeddings that capture complex spatial relationships between players. This approach addresses the limitations of previous methods in several ways:

1. It operates directly on raw player vectors, eliminating the need for feature engineering or fixed ordering schemes.
2. The permutation equivariance property naturally handles the player-ordering problem.
3. The self-attention mechanism allows for learning complex, long-range spatial relationships between players.
4. The architecture is flexible and can be adapted to various sports and analytical tasks with minimal modifications.

## 2.5 Prior Equivariant Solutions

This paper is not the first to develop general-purpose Deep Learning frameworks over unordered player tracking in sports. Here, we compare several notable approaches to our proposed Transformer-based method:

### 2.5.1 DeepSets

? targeted the problem of proposing a canonical end-to-end modeling of raw trajectory data for learning latent representations generally across problems and sports. This paper, released before Transformers demonstrated widespread modeling success outside of the NLP domain, used an equivariant architecture \textit{DeepSets} \citep{zaheer2018deepsets} that relies on global pooling of element-wise transformations.

While DeepSets offers permutation equivariance, it is likely inferior to Transformers in sports modeling scenarios where the set size is small but require deep, long-range pattern recognition. Transformers' self-attention mechanism allows for more nuanced interactions between all players, capturing complex spatial and temporal dependencies crucial in team sports. Moreover, Transformers have become ubiquitous across various domains, leading to extensive research, optimizations, and pre-trained models, which DeepSets lack.

### 2.5.2 Graph Neural Nets

? proposed using graph neural networks (GNNs) as a permutation-equivariant method to model over unordered players where each player represents a node in a fully connected graph.

While GNNs offer a natural representation of players on a field, Transformers present several advantages in the context of sports modeling with fully connected player interactions:

- In a fully connected scenario, the multi-hop message passing of GNNs becomes redundant, as all nodes are directly connected. Transformers, with their self-attention mechanism, can model these direct interactions more efficiently.

<sup>2</sup>While we do not dive into the details of the internal pieces of the Transformer, we note that they have been proven to be highly effective learners in many domains. For a comprehensive understanding of Transformer internals, we recommend many of the excellent public sources on the subject

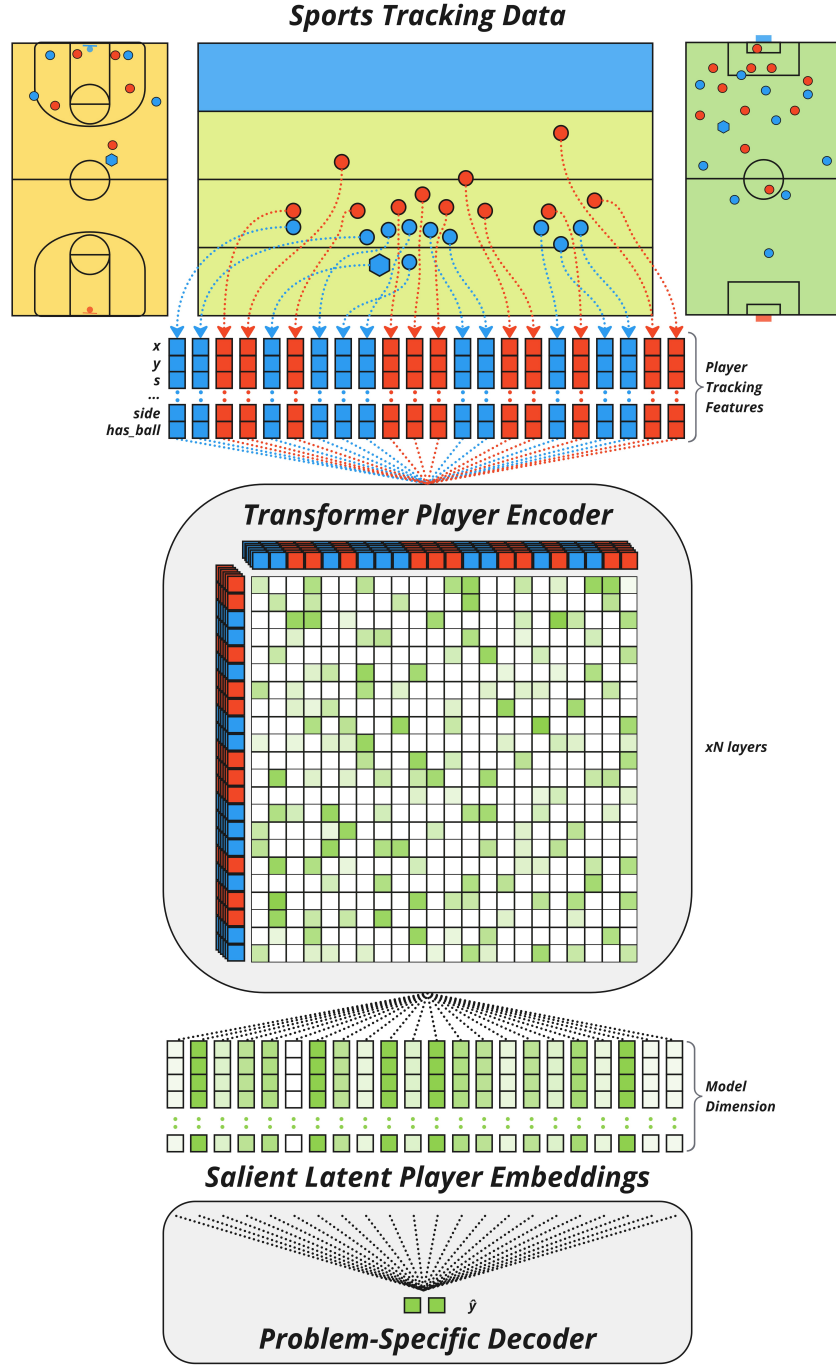


Figure 1: A generalized end-to-end Transformer Encoder modeling solution for sports tracking data analysis. The architecture consists of: (1) an input layer ingesting raw tracking features as unordered player vectors, (2) a Transformer Player Encoder that transforms these into salient player embeddings through repeated Multi-Head Attention Transformer layers, and (3) a problem-specific decoder that pools information (if needed) from the embeddings to learn a final  $\hat{y}$ .

A single head of self-attention is visualized in the center of the figure. This is the key aspect of why this approach fits the multi-agent problem. Each player vector is updated with information from each other player in a weighted manner as the model learns to identify important patterns for the objectives. The process of players attending to each other creates rich embeddings, which we believe is a common pattern across most tracking sports modeling tasks.

- The sparsity benefits typically associated with GNNs are nullified in a fully connected graph, negating one of their key advantages.
- ? demonstrates empirically that Transformers outperform this specific GNN approach.

### 2.5.3 The Zoo

? is a research group developed a model architecture that won the 2020 NFL Big Data Bowl in Kaggle challenge. This victory, along with the Big Data Bowl's growing prominence, led to The Zoo Architecture (TZA) becoming a de-facto equivariant deep learning approach used by entrants in following competitions. Furthermore, this architecture powers models and stats distributed by the NFL's advanced wing *Next Gen Stats*. In their submission, they identified the importance of designing for the player-order equivariance problem but discovered that their custom equivariant TZA design *outperformed* Transformers with Multi-Head Attention. TZA, as shown in Figure ??, relies on feature engineering vectors pairwise between each offensive and defensive player, and then operating dense layers over each pairwise vector *independently*, with pooling operations to eventually reduce the dimensionality into one final prediction. While not explicitly cited, we find many similarities between TZA and the DeepSet approach. We expect TZA to be

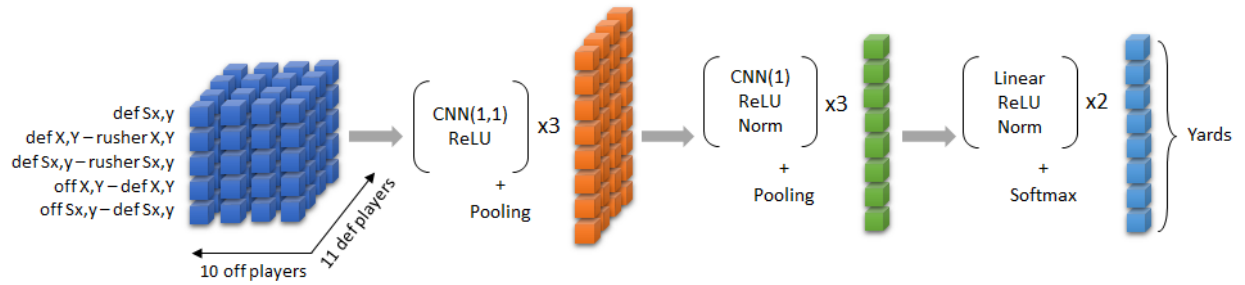


Figure 2: Simplified Structure of The Zoo Architecture that rose to prominence in modeling spatiotemporal NFL tracking data. It was designed to predict a categorical distribution over the number of yards gained on rushing plays using tracking frames at time of handoff. The model relies on manually constructing "interaction" feature vectors pairwise between each offensive and defensive player. The model then essentially treats these "interaction" feature vectors as independent throughout, with no mechanism to learn across player dimension. After applying a few dense layers to each interaction vector, pooling is applied to collect the most salient learned features across the offensive player and then defensive player dimension.

architecturally inferior for similar reasons:

- It is not a true end-to-end approach, relying on manual feature engineering.
- The independent processing of pairwise vectors may limit the model's ability to capture complex, multi-player interactions.
- The reliance on multiple intermediate pooling operations may lead to loss of important spatial information.

Furthermore, in our Experiments (Section ??), we demonstrate this inferiority empirically in a similar, but slightly more general task than what TZA was originally optimized for. We find that while TZA may have achieved impressive performance for that original dataset and specific task, it does not extend generally to other problems, even those that are just slightly different.

## 3 Experiments

The model code, data, and results for our experiments is available here.

### 3.1 Dataset

We utilized data from the 2024 NFL Big Data Bowl, a public competition hosted on Kaggle<sup>3</sup>. This dataset provides comprehensive tracking data, including the location, speed, and orientation of all 22 players on the field for Weeks 1-9

<sup>3</sup><https://www.kaggle.com/competitions/nfl-big-data-bowl-2024/data>

of the 2022 NFL Season. The dataset’s focus on tackling aligns well with our research goals, as it presents a complex spatial task that requires understanding player interactions. Key characteristics of the dataset include:

- Coverage: 136 games, approximately 2,000 unique plays, and 80,000 frames
- Content: Tracking frames where there is a clear ball-carrier and the defense is focused on tackling
- Features: Player positions, velocities, and orientations for each frame

For our modeling objective, we chose to predict the  $(x, y)$  position of the tackle. This task was selected for several reasons:

1. Alignment with the dataset’s tackling theme
2. Similarity to tasks that previous equivariant architectures (e.g., TZA) were designed for, allowing for meaningful comparisons
3. Generalizability across various football situations (e.g., post-handoff, post-pass catch, during scrambles)
4. Continuous output space, presenting a regression problem rather than a classification task

To standardize the data, we followed common practices in NFL modeling:

- All plays were standardized so that the offense always moves to the right
- We mirrored the data across the y-axis, effectively doubling our training dataset
- Normalize  $x, y$  positions to be relative to an anchor defined as the location of the ball-carrier at the start of a play. This is done primarily to ensure tracking frames look to be drawn from a similar distribution rather than different values based on the yardline of the play, and not for ordering the players in any way.
- We excluded the ball location data as in this dataset the ball was always held by a ball carrier and so was redundant.

For our experiments, we treated each frame as an independent input, although the tackle location labels are unique at the play level. We split the data into training, validation, and test sets with a 70/15/15 ratio based on unique plays. This resulted in:

- Training set: 9,000 unique plays, 750,000 frames
- Validation set: 2,000 unique plays, 150,000 frames
- Test set: 2,000 unique plays, 150,000 frames

## 3.2 Models

To evaluate the effectiveness of the Transformer architecture in sports analytics, we compare it against The Zoo Architecture (TZA), a baseline model that has shown success in similar tasks. To ensure a fair comparison, both models share the same AdamW optimizer, learning rate, and Huber Loss function. The Huber Loss balances L1 Loss for outliers with L2 Loss for predictions close to the target, providing robustness and numerical stability to our training. We conducted a grid search over learning rates, model sizes, and number of layers for both models, while keeping batch size and dropout fixed. This approach allows us to isolate the impact of architectural differences on model performance. The results are from the best model picked over the grid search.

### 3.2.1 Transformer Model Experiment

For the Transformer model in this experiment, we define the set of feature vectors  $V = \{v_1, v_2, \dots, v_K\}$  over the  $K = 22$  players, where each  $v_k \in \mathbb{R}^6$  represents the features of player  $p_k$ . Specifically, each feature vector  $v_k$  is composed of:

$$v_k = [x_k, y_k, vx_k, vy_k, o_k, b_k] \quad (6)$$

where:

- $(x_k, y_k)$  represent the spatial coordinates of player  $p_k$
- $(vx_k, vy_k)$  represent the velocity components of player  $p_k$
- $o_k \in \{0, 1\}$  is a binary indicator for offense (1) or defense (0)
- $b_k \in \{0, 1\}$  is a binary indicator for whether player  $p_k$  is the ball carrier (1) or not (0)

The label space  $\mathcal{Y}$  for our task is the predicted tackle location, defined as  $\mathcal{Y} = \mathbb{R}^2$ , representing the  $(x, y)$  coordinates of the predicted tackle.

The TransformerEncoder model  $f : (\mathbb{R}^6)^{22} \rightarrow (\mathbb{R}^d)^{22}$  maps the set of player feature vectors  $V$  equivariantly into a set of player embeddings of size model dimension  $d$ . Then we apply the task-specific decoder  $g : (\mathbb{R}^d)^{22} \rightarrow \mathbb{R}^2$  as an average pooling over the players followed by an MLP to get the predicted tackle location:

$$\hat{y} = (\hat{x}_{tackle}, \hat{y}_{tackle}) = g(f(V)) = g(f(\{v_1, v_2, \dots, v_{22}\})) \quad (7)$$

### 3.2.2 The Zoo Architecture Model Details

For our TZA, we have to perform a complex pairwise feature interaction process  $\Phi : V \rightarrow (\mathbb{R}^{10})^{10 \cdot 11}$  that converts  $V = \{v_1, v_2, \dots, v_K\}$  over the  $K = 22$  players into 110 interaction vectors. Specifically, each interaction vector  $u_{ij}$  between offensive player  $p_i$  and defensive player  $p_j$  with ball carrier  $p_b$  is composed of:

$$u_{ij} = [vx_j, vy_j, x_j - x_b, y_j - y_b, vx_j - vx_b, vy_j - vy_b, x_i - x_j, y_i - y_j, vx_i - vx_j, vy_i - vy_j] \quad (8)$$

where:

- $(x_k, y_k)$  represent the spatial coordinates of player  $p_k$
- $(vx_k, vy_k)$  represent the velocity components of player  $p_k$
- $p_b$  is the ball carrier

Then TZA model  $g : (\mathbb{R}^{10})^{10 \cdot 11} \rightarrow \mathbb{R}^2$  applies successive MLPs to interaction vectors independently, only pooling across to reduce dimensionality.

## 4 Results

Table 2: Test Set Event-Frame Performance Comparison (Mean Squared Error)

event	n	Transformer	Zoo	% diff
ball snap	1916	66.3	67.1	-0.9
handoff	1776	39.7	40.7	0.2
pass caught	1684	16.0	18.2	10.5
first contact	3156	9.0	12.5	28.0
out of bounds	544	3.7	13.0	64.8
tackle	2994	1.5	9.2	71.3

Our evaluation on a held-out test set reveals that the Transformer model outperforms TZA by 20.4% in our score of Mean Average Error (MAE). A breakdown of performance by specific game events provides further insights into the strengths of each model (Table ??).

Notably, both models perform comparably for early-play events such as ball snap and handoff. TZA’s strong performance in these situations aligns with its original optimization for handoff-related tasks, demonstrating its effectiveness in specific, well-defined scenarios.

However, the Transformer model’s advantage becomes increasingly apparent in later-play events, particularly out of bounds and tackle situations. In these cases, where the prediction target is essentially the ball carrier’s current location, the Transformer shows a superior ability to recognize play-ending situations and adapt to diverse game contexts.

This pattern is continued when we plot score based on temporal distance from the frame of tackle (Figure ??). Figure ?? shows a visual example of the Transformer generalizing to unseen frames in the test set significantly better than TZA is able to.

## 5 Supporting Work

Our discussion thus far has focused on previous work that relied on decomposing the modeling process through feature engineering (Section ??) and approaches that proposed solutions for handling player-order equivariance without utilizing Transformers or Attention mechanisms (Section ??). To provide a comprehensive overview, it is crucial to highlight research that has employed Attention mechanisms for addressing similar challenges in sports analytics:



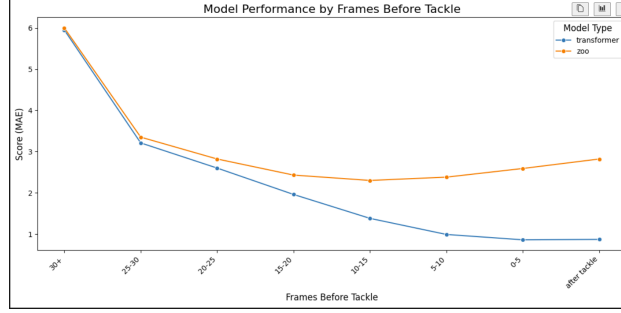


Figure 3: Plot showing how the models perform comparably early in plays but The Zoo Model fails to generalize as we arrive in more varied situations

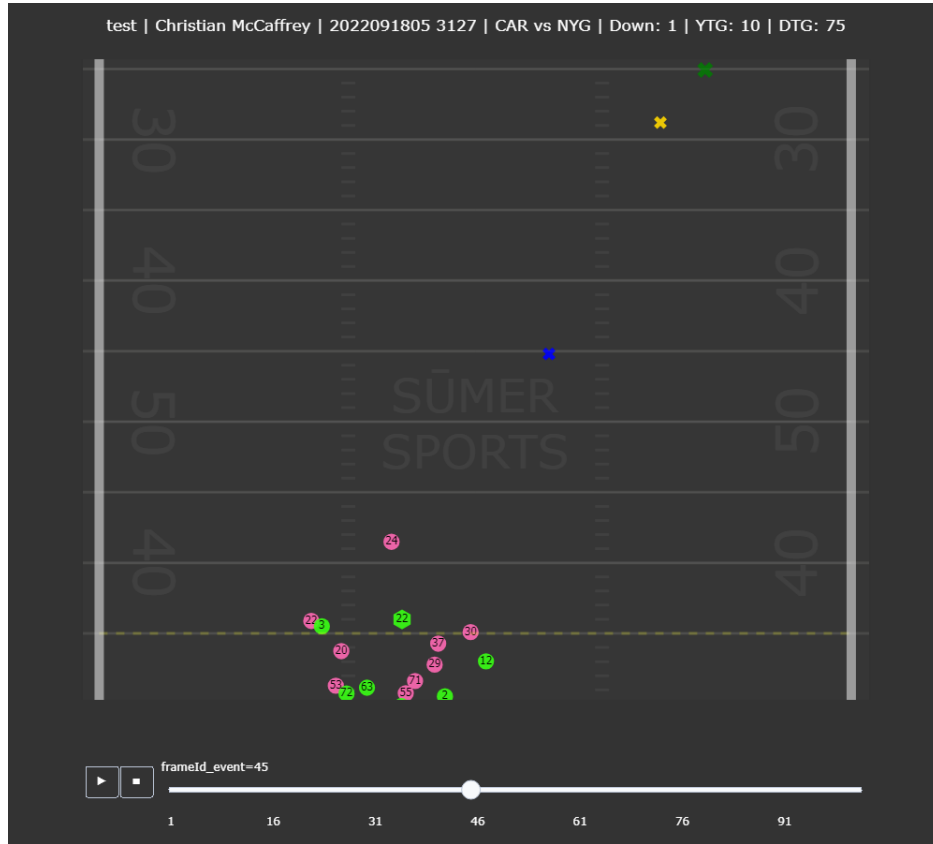


Figure 4: Visualization of a frame from the test set showing improved generalization from the Transformer. The green hexagon (#22) is the ball-carrier, green cross is the true tackle location. Yellow and blue crosses represent predictions from Transformer and Zoo models respectively

- **Baller2Vec:** ? proposed an innovative approach for analyzing basketball trajectories. Their method employed masked Attention applied causally across the time dimension while simultaneously allowing for equivariant learning across the player dimension. While we strongly endorse this work, we posit that its broader scope may have inadvertently obscured the value of Attention mechanisms for static, single-frame use cases, potentially limiting its widespread adoption in sports analytics.
- **TacticAI:** ? addressed a problem similar to the one presented in this paper, focusing on predicting events for corner kicks from static tracking frames. Their approach represents players as nodes in a graph, akin to the method discussed in Section ??, but employs the more advanced Graph Attention v2 (GATv2) mechanism for learning over the graph structure. While GATv2 is a sophisticated modification of the original Attention mechanism tailored for canonical graph representation problems, we argue that its application may be excessive

in the context of sports tracking data. In this domain, the graph is fully connected, contains relatively few nodes, and lacks significant edge features. Consequently, we posit that the application of GATv2 principles in this specific scenario likely reduces to a formulation very similar to the simpler Transformer Attention mechanism proposed in our work.

These studies underscore the utility of Attention mechanisms for modeling sports tracking frames. However, we believe that our paper contributes significantly by providing an easily reproducible approach with a narrower scope, focusing specifically on the application of Transformer-based Attention to static, single-frame sports tracking data.

## 6 Conclusion

In summary, the adoption of transformers in sports data modeling promises to address the limitations of existing methodologies by providing a simple, generalized, and scalable framework. Our methodology demonstrates the superior performance of transformer-based models compared to traditional approaches like The Zoo Architecture, highlighting their potential in capturing complex spatial interactions with minimal feature engineering. This paradigm shift could facilitate more robust and flexible analyses, ultimately advancing the field of sports data science.

### 6.1 Further Work

It was with intention that we kept the experiments and scope of this paper narrow. We wanted to stoke interest in Transformers as a powerful tool to be used in sports modeling, but leave plenty of room for innovation and further work to extend this. For example, while we have made claims about the generalizability of this approach we only had the resources to explore one application in this paper in one sport. We welcome additional effort to rigorously compare it to solutions that are not end-to-end and in other data spaces.