A stroke of genius: Predicting the next move in badminton

Magnus Ibh   Stella Graßhof   Dan Witzner Hansen

Machine Learning Group, IT University of Copenhagen

{ibhq, stgr, witzner}@itu.dk

Abstract

This paper presents, RallyTemPose, a transformer encoder-decoder model for predicting future badminton strokes based on previous rally actions. The model uses court position, skeleton poses, and player-specific embeddings to learn stroke and player-specific latent representations in a spatiotemporal encoder module. The representations are then used to condition the subsequent strokes in a decoder module through rally-aware fusion blocks, which provide additional relevant strategic and technical considerations to make more informed predictions. RallyTemPose shows improved forecasting accuracy compared to traditional sequential methods on two real-world badminton datasets. The performance boost can also be attributed to the inclusion of improved stroke embeddings extracted from the latent representation of a pre-trained large-language model subjected to detailed text descriptions of stroke descriptions. In the discussion, the latent representations learned by the encoder module show useful properties regarding player analysis and comparisons. The code can be found at: This https url.

1. Introduction

In racket sports, players exchange strokes in a rally until one player fails to successfully return a shot, giving the point to the opponent, see Fig. 1. Predictions about strokes of players, drawing on their history of previous strokes, benefit athletes' training and match preparation and can contribute to an improved viewing experience during live broadcasts [5]. Badminton, characterized by swift shot exchanges and strategic shuttle placement, presents a challenge for deep-learning computer vision-based sports analytics. The probabilistic nature of stroke forecasting in racket sports such as badminton [1] complicates predictive analytics due to the inherent unpredictability of player decisions. At any time, players face multiple viable actions. This unpredictability originates from the dynamic interplay of factors such as the physical state, psychological condition, and tactical approach of the player, which influence the selection of poten-
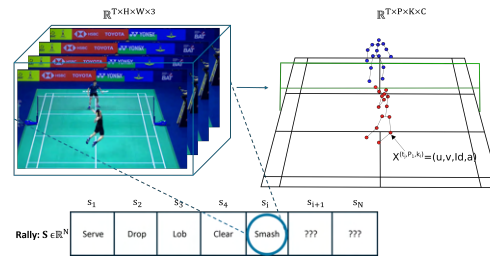

Figure 1. Datastructure overview shows that each stroke/action in a rally, i.e., stroke sequence, is provided the skeleton motion sequence of the stroke for additional context.

tial strokes and strategies. This work aims to design a model capable of incorporating some of the uncertainty-inducing factors into the prediction process.

One approach to reducing the uncertainty associated with subsequent stroke predictions involves incorporating various otherwise uncertain contextual information into the model. These factors include player skeleton data sequences, player identification (ID), and turn-based rally awareness. Player skeleton data sequences contain the movement and positions of a player's joints over time and have shown increasingly promising results for general action recognition [14, 22, 25, 33] and sports applications [8, 18, 19, 34]. The sequences contain the motion of player strokes. This data can reveal patterns in a player's technique and movement that give information about future strokes, allowing models to account for individual players' physical capabilities and limitations. Incorporating player ID as information allows the predictive models to consider historical performance data and personal playing styles. This individual-specific approach recognizes that each player has unique characteristics (strengths, weaknesses, and strategic preferences) influencing their selection of shots in the game. Theoretically, by identifying these characteristics, the model can better predict a player's likely actions in various game situations. Finally, turn-based rally awareness introduces an extra contextual layer to the prediction [31] by specifying the actor and reactor behind each stroke. Including turn-based nuances allows the model to isolate the

individual player's motion and obtain a clear representation of each stroke. The additional context provided by skeleton data sequences, player ID, and turn-based rally awareness attempts to construct a method that moves beyond the basic statistical probability of the rally sequences and instead embraces a more holistic understanding of the game. This approach aims to predict the next stroke and simultaneously capture the underlying (player) process behind stroke selection in racket sports.

This paper builds upon concepts of our previous work [17] and presents a transformer-based model for action forecasting in badminton.[1] The primary contributions of this research include:

1. A skeleton-based spatiotemporal encoder that uses transformer and pooling blocks to learn representations for enhancing next stroke predictions in badminton.
2. An adaptive cross-attention decoder that incorporates contextual stroke descriptors from high-dimensional embeddings of a pre-trained language model (LM).
3. Examples of how the latent variables can be used for match and playstyle analysis.

The following sections will detail the methodology behind this approach, the architecture of the transformer-based model, and the findings from various experimental evaluations.

2. Related Work

2.1. Action forecasting

Prior works have attempted to develop a wide range of neural network models to forecast future action sequences from observed action labels or extracted video features. The paper [11] introduced a method using a recurrent neural network (RNN) - hidden Markov model to classify actions from video frames, followed by a convolutional neural network (CNN) or RNN that predicted the following actions in the sequence. In [20], they employed a variational multi-headed GRU to predict future actions and their duration. They showed that their approach worked for both one-hot action labels and extracted video features. Similarly to our approach, [12] suggested jointly using both frame and annotation features to improve the prediction capacity of their model. [23] employed sequence-to-sequence models using a gated recurrent unit (GRU) encoder-decoder architecture to predict future actions from RGB frames alone. To our knowledge, skeleton data is not commonly used as a modality for in-action sequence forecasting. Instead, action labels are used to condition observed skeleton sequences to generate future skeleton sequences. [13, 24] employs variational autoencoders for this task.

2.2. Data analytical sports applications

Action recognition tasks fill up the majority of sports-focused research in the field of computer vision. Here, convolutional neural networks (CNN) have been used for feature extraction on RGB images [26]. Classification algorithms such as Support Vector Machines then use the extracted features to make predictions. Transformer models have also gained traction for spot application tasks. In [3], a Vision Transformer (ViT) [9] is used as the backbone to do group activity recognition (GAR) in Volleyball and basketball.

Skeleton data, as opposed to image data, has proven effective for the analysis and recognition of activities in various sports, including Tai Chi [8, 10, 30] and fencing [21, 34]. Skeleton-based Temporal convolutional networks (TCN) have seen use for action recognition in table tennis [18], where TCNs performed better than LSTM models. In badminton, [19] used skeleton data and shuttle trajectory data in a GRU model to perform binary hit detection. They further improved the detection rate by using badminton-specific rules. Specifically for stroke prediction [31] employed a transformer-based player and position-aware model that used prior stroke types and shuttle placement to predict future position and type of strokes. Instead of the shuttle placement, our work uses the players' skeleton and ground motion to provide a dynamic understanding of each stroke as the basis for predicting the subsequent strokes in the sequence.

3. Task formulation

In action forecasting for racket sports such as badminton, the strokes are the central actions. A stroke is the motion of a player preparing to hit the shuttle until shortly after contact between the racket and the shuttle. The exchange of strokes between players, called a rally, continues until one player fails to return the opposing player's stroke. The scientific objective is to predict the next stroke within a rally based on previously executed strokes while also considering the actual motion of players by incorporating 2D skeleton pose data. A pose $K_j^{(i)}$ within a stroke s_i (ith stroke in the sequence) captures the spatial configuration of the player's body at a given time frame j, represented by a set of keypoints that denote the 2D image positions of the body joints. Additionally, the sequence $G = g_1^{(i)}, .., g_j^{(i)}, .. g_T^{(i)}$, representing the 2D positions of the players' feet on the ground plane for each frame, is sampled and structured as $g_j^{(i)} \in \mathbb{R}^{T \times 2}$, as an additional data source. A rally S is denoted $S = [s_1, \ldots, s_N]$, where s_i is the ith stroke within the rally. Each stroke is described by a sequence of poses $K_1^{(i)}, \ldots, K_T^{(i)}$, with T representing the duration of a stroke sequence and N the number of strokes in a rally.

The goal is to predict the subsequent stroke s_{i+1} in the rally sequence S and show that leveraging both the historical sequence of strokes and motion provided by the 2D skeleton poses improves the prediction rate.

4. Model

This section describes the concepts of the autoregressive stroke prediction model, RallyTemPose. The main contribution of the model is that the encoder module takes the skeleton data and player ground condition as additional data and computes an embedded representation that conditions the rally sequence to predict the next stroke in the rally,

$$p(s_{i+1}|s_{1:i}, K_{1:i}, G_{1:i}, I) = \text{Dec}(s_{i:1}, \text{Enc}(K_{1:i}, G_{1:i}, Id)). \tag{1}$$

The overview of RallyTempose can be seen in Fig. 2.

4.1. Encoder

The encoder consists of a linear projection layer that embeds the raw data frames of player positions and skeleton poses into tokens. A learnable joint encoding (JE) is added to the tokenized data to provide information about the joint arrangement of the skeleton data. The Spatial Transform (ST) then applies a pose-wise transformer mechanism focusing on spatial relationships between keypoints in the player's movements. The ST is followed by a Grouped Pooling Block (GPB), which aggregates information, reduces dimensionality, and focuses on the relevant features of the players' movements. The Temporal Transformer (TT) focuses on the temporal dynamics, processing the pose movements over time. An important detail is that for the ST and TT blocks, both the inter-player (cross-attention) and intra-player (self-attention) attention is computed; see Fig. 3 for a visual depiction. The temporal transformer is followed by another GPB that pools over the embedded temporal representation. The final step produces (see Fig. 2) the three latent variables: a stroke representation z_s, a player 1 representation z_1, and a player 2 representation z_2. z_s merges the processed representations of both players for each stroke, providing complete context for each stroke. The player representations, on the other hand, are limited to information about one specific player.

Transformer Block:
In the transformer block, see Fig. 2, the layer normalized input is first subjected to the multi-headed self-attention (MHSA) mechanism that computes attention scores after being masked with either casual or padding mask (hides padded or future token from getting attention).

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V, \tag{2}$$

Q, K, and V represent Queries, Keys, and Values, all being learned linear projections of the embedded representation vectors. The transformer block employs multi-head attention by splitting Q, K, and V into multiple heads h for parallel processing: $\text{MultiHead}(Q, K, V) = \text{Concat}(\text{head}_1, ..., \text{head}_h)W^O$. Following this, a fully connected network (FC) applies nonlinear transformations to each position independently:

$$\text{FC}(x) = \text{GELU}(\text{Norm}(x)W_1 + b_1)W_2, \tag{3}$$

where both the MHSA and FC block have residual connections, GELU activations, first proposed in [16], and Norm refers to a layer normalization.

Group Pooling Block:
The GPB, shown in Fig. 2, based on [14], but here used in connection with transformer blocks instead of fully connected layers aggregates global and local information in embedded data through global and local max pooling. The pooling module operates on an embedded tensor $X \in \mathbb{R}^{G \times N \times D}$, split into select groups. N, G, and D denote the number of groups, group size, and the feature dimension, respectively. First, a global max pooling operation over the features in all groups with

$$M_d = \text{Gpool}(X)_d = \max_{n,g} X_{n,g,d}, M \in X \tag{4}$$

thus captures the most significant activations across all groups and instances for each feature. Simultaneously, local max pooling (Lpool) is executed by pooling over the group in X to create N features vector with the aggregated D features, yielding

$$Q_{n,d} = \text{Lpool}(X)_{n,d} = \max_g X_{n,g,d}. \tag{5}$$

The locally pooled features are then concatenated with the globally pooled features (expanded to match local dimensions), yielding a tensor $Y \in \mathbb{R}^{N \times 2D}$.

$$Y_{n,d} = \text{Concat}[\text{Lpool}(X)_{n,d}, \text{Gpool}(X)_d]. \tag{6}$$

Lastly, an FC layer maps it back to the feature dimension D.

4.2. Decoder

In the decoder, an embedding layer maps the one-hot encoded stroke sequences into stroke tokens. Subsequently, the turn-based nature of badminton is exploited by adding the specific player representations (z_1 or z_2) of the player performing the actual stroke. Through a self-attention module, the player-embedded stroke sequence is initially encoded. Subsequently, the decoder block (DB) uses cross-attention mechanisms to condition each stroke on the skeleton-based stroke representations z_s from the encoder. The final component, an MLP Head, takes the output from
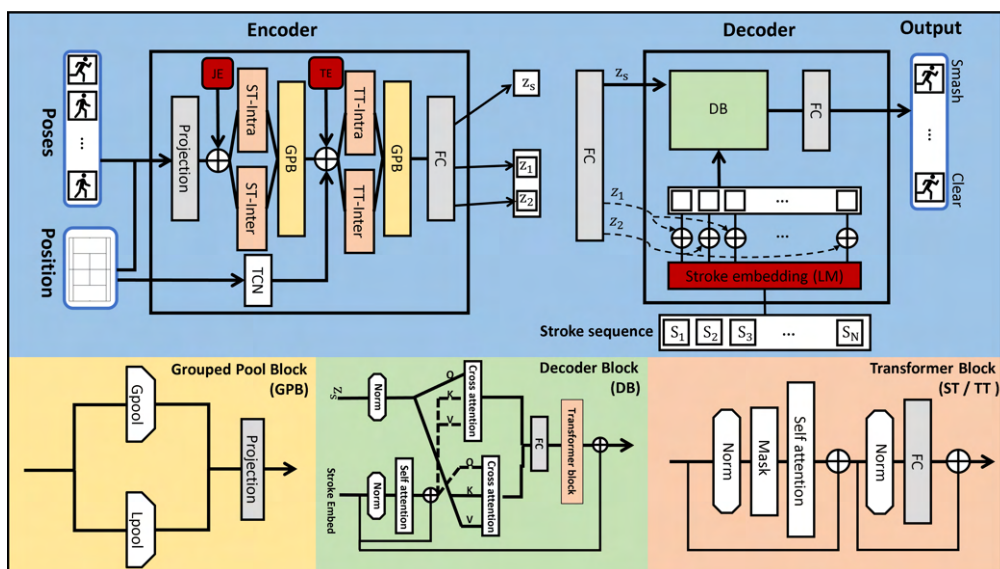
Figure 2. Overview of our approach in, with corresponding components. The abbreviations refer to the following: JE: learned joint encoding added to each pose keypoint, TE: learned temporal encoding added to the frame level tokens in a stroke, ST: spatial transformer, TT: temporal transformer, GPB: group pooling block, FC: fully connected, TCN: temporal convolutional network smoothing over the player ground positions, DB: decoder block.
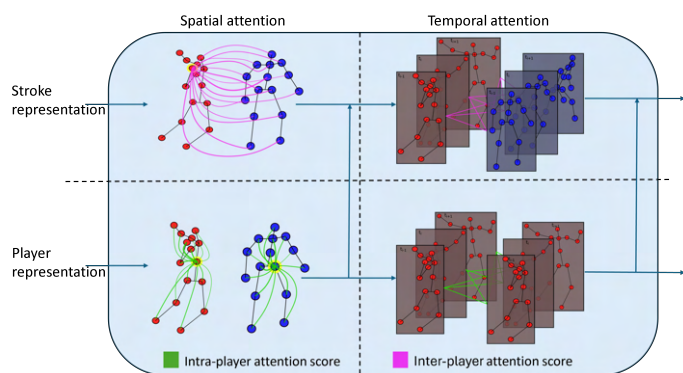

Figure 3. Illustration of the different types of attention present in the encoder module.

the DB to predict the probability of the next stroke in the sequence.

Decoder Block:
The Decoder Block (DB) combines self-attention, dual cross-attention, and an adaptive fusion mechanism. The block employs layer norms to ensure stability during the forward pass. The target embedded stroke sequence is first subjected to MHSA, after which encoder-to-decoder cross-attention and decoder-to-encoder reverse cross-attention are applied, facilitating stronger incorporation of the encoder representations z_s of the stroke motion. This is further ensured through an adaptive fusion layer that linearly combines the outputs of the dual cross-attention, which a stan-

dard Transformer Block subsequently processes for final refinement.

4.3. Enhanced Stroke Embeddings

Another aspect of our model is its pre-trained Language Model (LM) utilization. Specifically, BERT [7], for embedding various stroke types. Each stroke type is annotated with a description of its characteristics and typical use cases. From these descriptions, a high-dimensional stroke embedding is processed and extracted from the latent layer of a pre-trained BERT model. The LM representation provides more detailed embeddings than those derived from a learned embedding on a comparatively smaller dataset than the one on which the LM model was trained.

4.4. Training

The makeup of a transformer allows for $N-1$ training samples to be created from a N stroke rally S. In each training sample, the last stroke functions as the prediction target of the model, while all prior strokes in the rally serve as the observed sequence. This strategy allows for variable-length training sequences, allowing the model to observe the connection between all possible strokes in a rally during training. Sequence diversity helps the model avoid overfitting. The network is trained using two loss functions. First, we minimize the cross-entropy loss between the target and pre-

dicted strokes:

$$\mathcal{L}_{main}(s_{i+1}, \hat{p}_{i+1}) = -\sum_{j=1}^{C} s_{i+1}^{(j)} \log(\hat{p}_{i+1}^{(j)}), \qquad (7)$$

where C is the number of stroke classes, s_{i+1} is the one-hot encoded target stroke, and $\hat{p}_{i+1}$ is the predicted stroke type probability vector. Second, an auxiliary objective is defined on the output of the encoder's latent stroke variable, z_s. The cross-entropy of the linear projection, $\hat{a}_i$, of the latent stroke variable $z_{s_{(i)}}$ and the corresponding stroke type s_i is minimized as

$$\hat{a}_i = W_{aux}z_{s_{(i)}} + b_{aux}, \qquad (8)$$

$$\mathcal{L}_{aux}(s_i, \hat{a}_i) = -\sum_{j=1}^{C} s_i^{(j)} \log(\hat{a}_i^{(j)}). \qquad (9)$$

Here, both objectives are described for a single stroke denoted by the subscript i, but in practice, the loss is the average of all strokes in a sequence. The total loss is the weighted sum of the two losses

$$\mathcal{L} = \gamma\mathcal{L}_{aux} + \mathcal{L}_{main}, \qquad (10)$$

where γ is a hyperparameter, $\gamma = 0.3$ during experiments.

5. Experiments

5.1. Datasets

ShuttleSet The ShuttleSet [32] dataset contains 42 professional matches from 2018 to 2021, featuring 26 players across men's and women's singles categories. It is composed of more than 3000 rallies and 34000 strokes, with an average rally length of 10 strokes. Domain experts annotated the strokes in the dataset into 10 distinct shot types: *net shot*, *clear*, *push/rush*, *smash*, *defensive shot*, *drive*, *lob*, *dropshot*, *serve*, and *unknown/error*. The number of strokes for each type can be seen in Tab. 1. For model training and testing, the dataset is divided such that 80% of the rallies from each match are used for training, ensuring comprehensive player history, and the remaining 20% for testing.

BadmintonDB The BadmintonDB [2] dataset consists of 9 annotated video data professional men's singles matches. The dataset includes 811 rallies and 9,671 strokes, all featuring the players Kento Momota and Anthony Sinisuka Ginting. The dataset provides annotation of the strokes into 10 distinct types, that follow the recommended coaches' guide of the Badminton World Federation (BWF). The shot types are almost identical to the shuttleset data, see Tab. 1 for the stroke distribution. The same two players play in all matches, hence the 2 complete matches are reserved for testing and the remaining 7 for training.

5.2. Skeleton pose extraction

The pose extraction workflow involves two key stages: adopting techniques from [4, 6] for human detection and pose estimation and utilizing the HRNet framework [27] for precise 2D pose estimation. The presence of non-participants, like spectators, can adversely affect the skeleton data's quality by including poses of irrelevant characters. To tackle this, a homography based on the badminton court dimensions is computed to map detected individuals' feet to the ground plane, ensuring the focus is solely on players. This method also distinguishes between the top and bottom players in each sequence. Missing skeleton data is addressed by linear interpolation between the preceding and future frames. Pose normalization involves centering and scaling to standardize the bounding box diagonal to one.

5.3. Evaluation metrics

In badminton, more than one stroke is often a viable choice, which should be reflected in the evaluation metrics. The performance of the models is judged based on the accuracy (acc) of their prediction, the top-2 accuracy (acc2), and the top-3 accuracy (acc3).

5.4. Baselines

No other existing work uses stroke skeleton data to enhance future stroke prediction capabilities. Therefore, our model performance is compared to other sequence and action prediction baselines not explicitly designed for badminton. All model baselines consist of current state-of-the-art concepts for sequence prediction, and thus, while not intentionally designed for badminton stroke prediction, comparing to the baselines allows for a good estimate of the prediction capabilities of our specific model. The following baselines are used for comparison:

- Seq2Seq [28]
- Transformer [29]
- Actionformer [22] + Transformer decoder

5.5. Implementation details

The dimension of embedded representation (d) per head is set to 16, the number of heads (h) in the MHSA is set to 4, and the forward expansion in the inner dimension of feed-forward layers is set to 4 following[15]. A rally's max sequence length (s) is set to 35, and T varies for different rallies. Similarly, the max temporal length of each stroke motion (T) is set to 30. Dropout and Attention dropout are utilized in each MHSA block with a drop rate of 0.3. The models are trained with a batch size of 1 using AdamW with a learning rate set to 10^{-4}. Zero padding is performed for individual stroke motion sequences. Padding the rallies was also tested but did not improve performance.

Table 1. Distribution of the data classes for the two datasets.

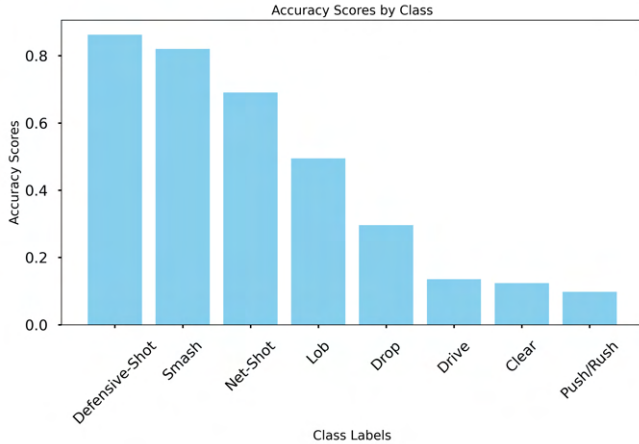|	Net-Shot	Defensive-Shot	Smash	Lob	Clear	Drive	Dropshot	Push/Rush	Serve	Error
ShutSet	6716	3836	3749	4614	2440	1091	2929	3021	2060	1095
BadminDB	1756	1281	1154	1954	596	188	108	715	108	131

Figure 4. Comparisons of class accuracy for the different stroke types in the ShuttleSet dataset.

5.6. Main Experiments

In the comparative analysis of predictive models on the ShuttleSet and BadminDB datsets, our model outperforms the other baseline models in standard and top-3 accuracy. On the ShuttleSet dataset, it achieves an accuracy of 54.3%, a top-2 accuracy of 77.3%, and a top-3 accuracy of 92.5%, indicating its ability to rank the correct outcome within the top three predictions in over 90% of the cases. In the BadminDB dataset, our model achieves an accuracy of 62.8% and a top-3 accuracy of 93.1%. The BadminDB is much smaller than ShuttleSet, which resulted in our model often overfitting. As a result, the much simpler sequential models perform better on BadminDB comparatively, but RallyTempose still slightly outperforms them in the end.

The results show the model's prediction prowess and reflect its ability to select the most logical outcomes. For a given situation, multiple stroke candidates can be perfectly viable simultaneously. The results in both accuracy metrics, especially in the top-3 accuracy, suggest that our model's way of incorporating skeleton-motion and player-specific information improves the prediction logic compared to the baselines in the context of badminton datasets.

Logical misclassifications: In Fig. 4, the specific accuracy and misclassification ratio for all stroke types is plotted. Strokes like the smash are accurately predicted, while strokes like the clear and drives are only correctly predicted

12% and 14% of the time, respectively. However, by examination of the confusion matrix in Fig. 5, most classifications can be attributed to logical reasoning, and all misclassifications belong to sensible groups (Net-shots, push-rush, and lobs), (drives and defensive shots) and, (smash, clears and drops). For example, a clear is predominantly hit from the backcourt on shuttle trajectories and racket swings similar to a smash and, to a lesser degree, a drop. This is consistent with the faulty prediction of clears being smashes and drops, and hence the predictions follow an underlying logic of the game. Similarly, a drive can easily be confused with a defensive reaction shot. Our model can still be improved further. We hypothesize that a deeper strategic understanding of each situation can increase accuracy even more. However, the results indicate that our model, through purely next-stroke action prediction, has developed a rudimentary game understanding.

6. Discussion

6.1. Ablation Study

The impact of our skeleton-based stroke condition on the prediction capability is examined through an ablation study. The relative contribution of 1.) skeleton data, 2.) ground position of the players, and 3.) specific player embedding is determined through six different model variants. In 3 the respective prediction accuracies are shown after removing specific model inputs and their corresponding model components. The results show that the most critical factor is the inclusion of the player ground position, as leaving out this data along with the TCN block leads to a 2.6% drop in performance. The encoder version made up solely of a TCN block achieves a 51.6% accuracy. The player-specific information does not significantly boost the prediction accuracy, however, as shown in the next section, learning player-specific representations allows for introspective player analysis that can be extrapolated from the model. Since including the players' ground positions results in a significant performance boost. A potentially even greater performance increase could be obtained by including 3D skeleton data as well.

6.2. Match Analysis Prospects

The model's design allows for player comparison by analyzing the latent variables of the model. Fig. 6 and Fig. 7 show t-SNE plots of the latent variables. In the visualization z_s are colored based on the target stroke they represent,

Table 2. Accuracy (Acc), Top-2 Accuracy (Acc-2), and Top-3 Accuracy (Acc-3) of our models and other baselines on the ShuttleSet and BadminDB datasets.

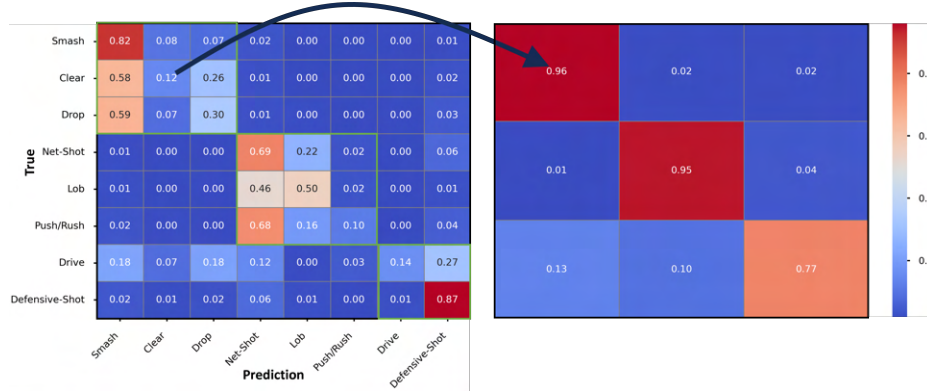| Model	ShuttleSet			BadminDB		
	Acc (%)	Acc-2 (%)	Acc-3 (%)	Acc (%)	Acc-2 (%)	Acc-3 (%)
Seq2Seq (LSTM)	47.9	72.4	83.5	57.3	82.3	86.0
Transformer	49.8	73.9	87.2	61.5	85.4	92.5
POT + Trans Dec	52.1	74.1	91.2	58.4	82.0	91.7
RallyTemPose	54.3	77.3	92.5	62.8	83.5	93.1

Figure 5. To the left is the confusion matrix for the shuttles data, and to the right is the confusion matrix grouped according to logical classes.

Table 3. Ablation Study of RallyTemPose model.

Keypoint	Ground	Player Rep	Accuracy (%)
			48.3
✓			49.2
		✓	46.9
	✓		51.6
	✓	✓	50.1
✓	✓		52.4
✓		✓	51.7
✓	✓	✓	54.3

whereas z_1 and z_2 are colored according to the players they represent. Clear groupings are observed for the different z_s stroke variables and partial groupings of the player variables. This indicates that z_s and, to a lesser degree, z_1 and z_2 stores relevant information about strokes and playing styles respectively. While the specific player embedding does not significantly improve the model's prediction accuracy, it allows for model intrinsic playstyle comparisons.

Player Similarity We can project the playstyle similarity of different players by looking at the cosine similarity of the player-specific latent variable for the other players

in the dataset. The cosine similarity is calculated by random sampling of $N = 1000$, strokes, for each of the pair combinations of players and calculating the average cosine similarity between the latent player variables as

$$\text{Player Sim}_{i,j} = \sum_{n=1}^{N} \frac{\mathbf{z_i^n} \cdot \mathbf{z_j^n}}{\|\mathbf{z_i^n}\| \|\mathbf{z_j^n}\|}. \quad (11)$$

Tab. 4 shows the cosine similarity between the latent variables of players for 5 different players. Observe that there is a notable difference in similarity between the players. On average, the male (first 3 players) and female (last 2 players) have a lower similarity, whereas the same gender similarity scores are higher. However, the player similarity score is also quite low between the 3 males. This, however, is quite sensible since Male 3, known for a unique, endurance-based, hard-to-read playstyle, Male 2, with a very fast-paced style, and Male 1, with a physical and powerful playstyle, are very different players, and the similarity score reflects that. Future work could include categorizing distinct playstyles and attempting to interpret them as defensive, offensive, power, placement, etc.

Play-style analysis In Fig. 8, a bar plot of the average accuracy for each player in the ShuttleSet dataset is shown. There is a notable gap of more than 20% average accuracy
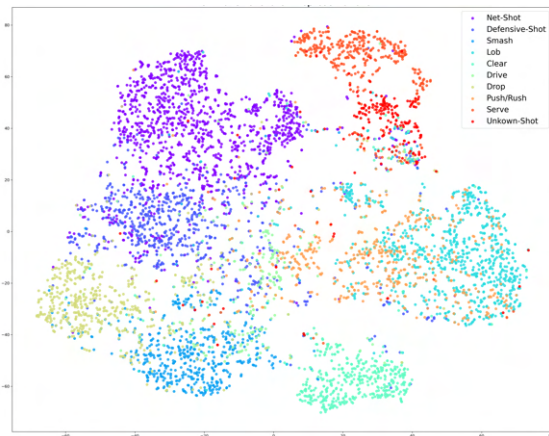
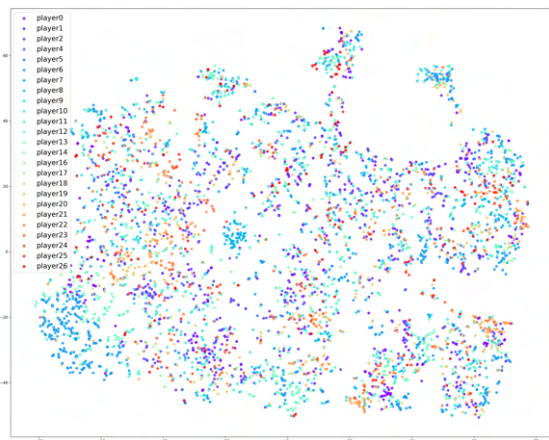Figure 6. t-SNE plot over the latent stroke z_s representation, colored according to the observed stroke types.

Figure 7. t-SNE plot over the latent player representation (z_1 and z_2), colored according to the target player Id. Note the lack of very distinct groupings of the player variables, which could be explained by the difference/similarity in how players perform certain strokes.

Table 4. Cosine similarity between latent player variables of different classes. (M: male, F: female)

Player sim	M1	M2	M3	F1	F2
M 1	0.61				
M 2	0.43	0.58			
M 3	0.37	0.41	0.67		
F 1	0.21	0.19	0.31	0.71	
F 2	0.23	0.51	0.49	0.57	0.65

between the players, which strokes are predicted the best compared to the player predicted the worst. The prediction accuracy of specific players could be used to indicate how well players can mask their strokes. However, the approach
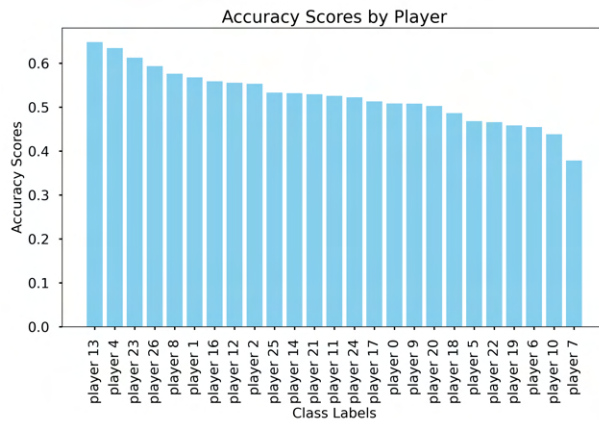

Figure 8. The average accuracy of next stroke predictions for all the players in the dataset.

assumes the model can flawlessly predict straightforward strokes, which is not yet guaranteed. Still, through continuous improvement of the model, this could be a helpful asset for player analysis.

6.3. Future prospects

Looking ahead, we aim to enhance the model's capabilities by incorporating additional variables, such as match outcomes (win/loss), to facilitate more sophisticated tactical analysis. Additionally, expanding the model to predict the skeleton sequence of the predicted strokes would be beneficial not only for sports analysis purposes but also for creating synthetic data in a field where quality annotated datasets are sparse.

7. Conclusion

This research introduced a model specifically designed for stroke prediction in badminton, utilizing an encoder-decoder architecture. The model integrates skeleton data and player-specific information using a spatiotemporal transformer encoder. Our experiments, conducted on two different real-world badminton datasets, show an increase in performance for our approach compared to other forecasting baselines. Furthermore, the extracted latent representations show potential use for player analysis and match preparation.

References

[1] Yazan Abu Farha and Juergen Gall. Uncertainty-aware anticipation of activities. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, pages 0–0, 2019. 1

[2] Kar-Weng Ban, John See, Junaidi Abdullah, and Yuen Peng Loh. Badmintondb: A badminton dataset for player-specific match analysis and prediction. In *Proceedings of the 5th International ACM Workshop on Multimedia Content Analysis in Sports*, page 47–54, New York, NY, USA, 2022. Association for Computing Machinery. 5

[3] Naga VS Raviteja Chappa, Pha Nguyen, Alexander H. Nelson, Han-Seok Seo, Xin Li, Page Daniel Dobbs, and Khoa Luu. Spartan: Self-supervised spatiotemporal transformers approach to group activity recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pages 5158–5168, 2023. 2

[4] Kai Chen, Jiaqi Wang, Jiangmiao Pang, Yuhang Cao, Yu Xiong, Xiaoxiao Li, Shuyang Sun, Wansen Feng, Ziwei Liu, Jiarui Xu, Zheng Zhang, Dazhi Cheng, Chenchen Zhu, Tianheng Cheng, Qijie Zhao, Buyu Li, Xin Lu, Rui Zhu, Yue Wu, Jifeng Dai, Jingdong Wang, Jianping Shi, Wanli Ouyang, Chen Change Loy, and Dahua Lin. MMDetection: Open mmlab detection toolbox and benchmark. *arXiv preprint arXiv:1906.07155*, 2019. 5

[5] Wei-Ta Chu and Samuel Situmeang. Badminton video analysis based on spatiotemporal and stroke features. In *Other Conferences*, pages 448–451, 2017. 1

[6] MMPose Contributors. Openmmlab pose estimation toolbox and benchmark. https://github.com/open-mmlab/mmpose, 2020. 5

[7] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4171–4186. Association for Computational Linguistics, 2019. 4

[8] Lingxiao Dong, Dongmei Li, Shaobin Li, Shanzhen Lan, and Pengcheng Wang. Tai chi action recognition based on structural lstm with attention module. In *Other Conferences*, 2019. 1, 2

[9] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021. 2

[10] Yong Du, Wei Wang, and Liang Wang. Hierarchical recurrent neural network for skeleton based action recognition. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1110–1118, 2015. 2

[11] Yazan Abu Farha, Alexander Richard, and Juergen Gall. When will you do what? - anticipating temporal occurrences of activities. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5343–5352, 2018. 2

[12] Harshala Gammulle, Simon Denman, Sridha Sridharan, and Clinton Fookes. Forecasting future action sequences with neural memory networks. In *British Machine Vision Conference*, 2019. 2

[13] Chuan Guo, Xinxin Zuo, Sen Wang, Shihao Zou, Qingyao Sun, Annan Deng, Minglun Gong, and Li Cheng. Action2motion: Conditioned generation of 3d human motions. In *Proceedings of the 28th ACM International Conference on Multimedia*, pages 2021–2029, 2020. 2

[14] Ryo Hachiuma, Fumiaki Sato, and Taiki Sekii. Unified keypoint-based action recognition framework via structured keypoint pooling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 22962–22971, 2023. 1, 3

[15] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross B. Girshick. Masked autoencoders are scalable vision learners. *CoRR*, abs/2111.06377, 2021. 5

[16] Dan Hendrycks and Kevin Gimpel. Gaussian error linear units (gelus). *arXiv preprint arXiv:1606.08415*, 2016. 3

[17] Magnus Ibh, Stella Graßhof, Dan Witzner, and Pascal Madeleine. TemPose: a new skeleton-based transformer model designed for fine-grained motion recognition in badminton. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 5199–5208, 2023. ISSN: 2160-7516. 2

[18] Kaustubh Milind Kulkarni and Sucheth Shenoy. Table tennis stroke recognition using two-dimensional human pose estimation. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 4571–4579, 2021. 1, 2

[19] Paul Liu and Jui-Hsien Wang. MonoTrack: Shuttle Trajectory Reconstruction From Monocular Badminton Video. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, page 10, 2022. 1, 2

[20] Siyuan Brandon Loh, Debaditya Roy, and Basura Fernando. Long-term action forecasting using multi-headed attention-based variational recurrent neural networks. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 2418–2426, 2022. 2

[21] Filip Malawski and Bogdan Kwolek. Improving multimodal action representation with joint motion history context. *Journal of Visual Communication and Image Representation*, 61: 198–208, 2019. 2

[22] Vittorio Mazzia, Simone Angarano, Francesco Salvetti, Federico Angelini, and Marcello Chiaberge. Action transformer: A self-attention model for short-time pose-based human action recognition. *Pattern Recognition*, 124:108487, 2022. 1, 5

[23] Yan Ng and Basura Fernando. Forecasting future action sequences with attention: A new approach to weakly supervised action forecasting. *IEEE transactions on image pro-*

cessing : a publication of the IEEE Signal Processing Society, PP, 2020. 2

[24] Mathis Petrovich, Michael Black, and Gul Varol. Action-conditioned 3d human motion synthesis with transformer vae. In *ICCV*, pages 10965–10975, 2021. 2

[25] Chiara Plizzari, Marco Cannici, and Matteo Matteucci. Skeleton-based action recognition via spatial and temporal transformer networks. *Computer Vision and Image Understanding*, 208-209:103219, 2021. 1

[26] Nur Azmina Rahmad and Muhammad Amir As'ari. The new convolutional neural network (cnn) local feature extractor for automated badminton action recognition on vision based data. *Journal of Physics Conference Series*, 1529, 2020. 2

[27] Ke Sun, Bin Xiao, Dong Liu, and Jingdong Wang. Deep High-Resolution Representation Learning for Human Pose Estimation. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5686–5696, Long Beach, CA, USA, 2019. IEEE. 5

[28] Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. Sequence to sequence learning with neural networks. In *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2*, page 3104–3112, Cambridge, MA, USA, 2014. MIT Press. 5

[29] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, page 6000–6010, Red Hook, NY, USA, 2017. Curran Associates Inc. 5

[30] Pengcheng Wang and Shaobin Li. Structural-attentioned lstm for action recognition based on skeleton. In *Other Conferences*, 2018. 2

[31] Wei-Yao Wang, Hong-Han Shuai, Kai-Shiang Chang, and Wen-Chih Peng. Shuttlenet: Position-aware fusion of rally progress and player styles for stroke forecasting in badminton. In *AAAI*, pages 4219–4227. AAAI Press, 2022. 1, 2

[32] Wei-Yao Wang, Yung-Chang Huang, Tsi-Ui Ik, and Wen-Chih Peng. Shuttleset: A human-annotated stroke-level singles dataset for badminton tactical analysis. In *KDD*, pages 5126–5136. ACM, 2023. 5

[33] Sijie Yan, Yuanjun Xiong, and Dahua Lin. Spatial temporal graph convolutional networks for skeleton-based action recognition. *AAAI*, 32, 2018. 1

[34] Kevin Zhu, Alexander Wong, and John McPhee. Fencenet: Fine-grained footwork recognition in fencing. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 3588–3597, 2022. 1, 2