**SELF-SUPERVISED TIME-AWARE TRANSFORMER FOR UNSUPERVISED DETECTION OF OVERTRAINING AND INJURY PRECURSORS IN TENNIS ATHLETES**

**Shobharani Polasa**, Weihao Qu, Jay Wang, Ling Zheng  
Department of Computer Science and Software Engineering, Monmouth University, West Long Branch, USA  
Email: {s1365603, wqu, jwang, lzheng}@monmouth.edu

**Abstract**— Early detection of overtraining and injury precursors in tennis athletes remains a critical but under-addressed challenge, primarily because conventional injury-prediction models rely on labeled clinical or self-reported injury events—labels that are often sparse, delayed, or inconsistently recorded. To overcome this limitation, we present a **self-supervised Time-Aware Transformer** framework that operates entirely without injury labels, enabling continuous, real-time readiness monitoring.

In our approach, we first **use** three complementary data streams into a single **weekly multimodal token** per athlete:

1. **Daily wellness questionnaires** capturing subjective measures (muscle soreness, mental stress, nutrition, hydration, pain location) aggregated into weekly statistics;
2. **Weekly vertical jump assessments** (Sparta protocol) quantifying explosive lower-body performance and neuromuscular fatigue through mean jump height, peak jump height, and within-week drop percentages;
3. **Continuous WHOOP wearable metrics**, including heart-rate variability (HRV), resting heart rate, respiratory rate, sleep stages (deep, light, REM), sleep efficiency, cumulative sleep debt, and daily strain scores.

Over a four-week window, each athlete thus contributes a sequence of **four** weekly tokens (36 sequences total across nine collegiate players), which our Transformer ingests. To train without labels, we combine two **self-supervised objectives**:

* **Masked Feature Reconstruction (MFR):** Randomly mask 15% of each weekly token’s features and require the model to reconstruct the missing values, forcing it to learn intra-week correlations (e.g., how reduced HRV correlates with increased soreness).
* **Next-Week Prediction (NWP):** Task the model with forecasting the entire next week’s token from the preceding weeks’ hidden states, compelling it to capture temporal dynamics such as recovery trends and fatigue accumulation.

After training converges, we extract the learned **[CLS]** embedding (512-dimensional) for each four-week sequence—a distilled representation of an athlete’s recent multimodal behavior. We then fit a **Gaussian Mixture Model (GMM)** with four full-covariance components to the embeddings of presumed “normal” weeks (weeks 1–3). Weeks whose embeddings lie in the low‐probability tail of this GMM distribution are **flagged as anomalies**.

We validate our pipeline on nine held-out anomalous weeks—identified independently by coaches through elevated soreness (≥8/10), >15% drop in jump height, sleep efficiency below 75%, or ≥10% performance rating dip. Our unsupervised method achieves **Precision = 0.80**, **Recall = 0.76**, and **F₁ = 0.78**, significantly outperforming baseline unsupervised pipelines: **PCA + GMM** (F₁ = 0.58) and **LSTM-Autoencoder + GMM** (F₁ = 0.69).

By eliminating the need for any injury labels, our fully unsupervised framework offers a scalable, real-time solution for coaches and sports scientists to detect early warning signs of overtraining or injury, enabling timely interventions and personalized load management across diverse athletic populations.

**Index Terms**—Transformer, self-supervised learning, unsupervised anomaly detection, sports analytics, tennis readiness.

**I. INTRODUCTION**

Tennis is a sport characterized by its high physical demands—rapid changes of direction, explosive serves and groundstrokes, and prolonged rallies under varying environmental conditions. To maintain peak performance while minimizing injury risk, coaches and sports scientists routinely collect a range of data: objective physiological measurements (e.g., heart-rate variability, sleep patterns), performance tests (e.g., vertical jump height), and subjective self-reports (e.g., muscle soreness, perceived stress). Each of these modalities offers a window into an athlete’s readiness and recovery status, yet when considered in isolation they provide an incomplete picture. For example, elevated heart-rate variability may signal good recovery, but if paired with high soreness scores it might instead indicate compensatory stress patterns or latent inflammation.

Despite the wealth of possible indicators, actual injury events are relatively infrequent, and when they do occur the official labels—typically clinical diagnoses or self-reports—can be delayed by days or even weeks. This latency means supervised machine-learning models trained on injury labels often suffer from severe class imbalance, sparse positive examples, and noisy ground truth. Consequently, supervised approaches frequently overfit to the few labeled incidents, generalize poorly to new athletes, or fail to capture more subtle “pre-injury” states that manifest days before an acute injury.

In contrast, self-supervised learning offers a promising alternative: rather than requiring labeled injury events, models can learn the underlying structure of “normal” training and recovery patterns from abundant unlabeled data. Transformer architectures—with their self-attention mechanism—are particularly well suited to this task. Originally developed for natural language, Transformers can flexibly capture long-range dependencies across sequential data. By training on proxy tasks such as reconstructing randomly masked inputs or forecasting future time points, a Transformer learns rich representations of temporal dynamics without any explicit injury labels.

In this work, we adapt the Transformer to the sports-analytics domain with two key innovations. First, we introduce a Time-Aware Transformer that processes sequences of weekly multimodal tokens, each token fusing subjective wellness questionnaires, weekly vertical jump metrics, and continuous wearable signals (heart-rate variability, sleep efficiency, strain score). Second, we employ a combined Masked Feature Reconstruction objective—wherein the model must predict withheld features within each week—and a Next-Week Prediction objective—wherein it must forecast the subsequent week’s full token. Together, these proxy tasks encourage the network to internalize both intra-week interdependencies (e.g., how soreness and sleep interact) and inter-week trends (e.g., cumulative fatigue or recovery patterns).

Once trained, the Transformer’s learned representations can be used for unsupervised anomaly detection: weeks whose encoded embeddings fall outside the typical distribution of “normal” weeks are flagged as potential overtraining or injury precursors. By alerting coaches to these anomalous weeks—characterized by high soreness, significant drop in jump height, poor sleep, or performance dips—this pipeline enables real-time, label-free monitoring. Coaches can then proactively adjust training load or prescribe recovery protocols, ultimately helping to reduce injury incidence and extend athletes’ competitive longevity.

**II. RELATED WORK**

**A. Transformers in Time Series**

The Transformer architecture was first introduced by Vaswani *et al.* in 2017 for natural language processing, where it replaced recurrent and convolutional modules with a **self-attention** mechanism that directly models pairwise interactions across all positions in an input sequence [2]. In contrast to RNNs—whose sequential nature limits parallelization and can struggle with long-range dependencies—Transformers compute attention weights in parallel, allowing each token to attend to any other token regardless of distance.

Extending Transformers to time-series data requires two key adaptations:

1. **Positional Encodings:** Unlike words in a sentence, numeric feature vectors for different time steps lack an inherent order. To address this, researchers add learned or fixed positional encodings that inject temporal order into the model. For example, sinusoids of varying frequencies or trainable embeddings are summed with each time-step’s feature projection, so the model knows which week or day each token corresponds to.
2. **Masked Reconstruction:** Inspired by BERT-style pretraining in NLP, masked reconstruction tasks mask out portions of the input sequence—either entire time steps or individual feature dimensions—and train the model to reconstruct the missing values. This objective encourages the Transformer to learn the underlying correlations in multivariate temporal data, such as how a drop in sleep quality might co-occur with elevated resting heart rate or rising muscle soreness.

Several studies have shown that these adaptations allow Transformers to outperform traditional RNNs and convolution-based models on benchmarks such as energy consumption forecasting, traffic prediction, and multivariate sensor data modeling. By capturing both local and global temporal patterns in parallel, Time-Aware Transformers can learn richer, more flexible representations of sequential data.

**B. Self-Supervised Anomaly Detection**

Anomaly detection in time series traditionally relies on reconstruction- or prediction-based autoencoders: a model is trained on “normal” data, then high reconstruction or forecasting error flags potential anomalies. Recently, **self-supervised Transformers** have been explored for anomaly detection by repurposing masked reconstruction as a proxy task. For instance, **AnomalyBERT** [4] randomly masks entire time-steps in a multivariate sequence and trains a Transformer to reconstruct the masked data from its context. Weeks or days with unusually high reconstruction error are then marked as anomalies.

Key advantages of Transformer-based self-supervision include:

* **Contextual Awareness:** Self-attention integrates information from all time steps, allowing anomalies that subtly disrupt long-range patterns to be detected.
* **Feature-Level Masking:** Masking individual features—rather than whole steps—enables the model to learn cross-feature dependencies (e.g., how unusual heart-rate patterns coincide with changes in activity strain).
* **No Labels Required:** Self-supervision eliminates the need for ground-truth anomaly labels, which are often scarce in real-world deployments.

Empirical results across domains (network intrusion, industrial sensor monitoring, and healthcare time series) show that Transformer-based masked reconstruction outperforms convolutional and recurrent autoencoders, particularly when anomalies manifest as slight drifts or complex multivariate shifts rather than gross outliers.

**C. Sports Injury Prediction**

Traditional sports-injury prediction models adopt a **supervised** paradigm, requiring labeled injury events to train classifiers or regressors. The **PART** framework [1], for example, fuses data from WHOOP wearables, daily self-report questionnaires, weekly vertical-jump tests, and match-play video analysis into a composite **Athlete Readiness Score (ARS)**. It then trains an ensemble of XGBoost regressors and classifiers, multilayer perceptrons (MLPs), and long short-term memory (LSTM) networks to predict both overall readiness and injury risk for specific body regions.

While PART and similar pipelines achieve strong predictive performance when supplied with high-quality injury labels, they face several limitations:

1. **Label Scarcity & Bias:** Injuries are often underreported, and clinical confirmation can lag behind initial symptom onset. This leads to noisy or delayed labels that compromise model reliability.
2. **Generalizability:** Supervised models trained on one team or season may not transfer well to different athlete populations, sports, or training regimens.
3. **Data Collection Burden:** Gathering match-video, self-reports, and wearable metrics at scale requires considerable infrastructure and athlete compliance.

Our proposed approach removes the dependence on any injury or readiness labels by leveraging **self-supervised** Transformer pretraining followed by **unsupervised** density modeling. This allows continuous, label-free monitoring: instead of explicitly learning a mapping to discrete injury outcomes, the model learns the structure of “normal” training and flags deviations that correspond to early signs of overtraining or injury. Coaches and sports scientists can then intervene based on anomaly alerts, rather than waiting for injury labels to accrue.

**III. DATA PROCESSING AND WEEKLY TOKENIZATION**

**A. Participants**

We recruited **nine Division I** collegiate tennis athletes (5 male, 4 female; mean age 20.3 ± 1.5 years) from Monmouth University’s men’s and women’s teams. Each athlete participated in our study over a **16-week** period spanning three distinct phases:

1. **Pre-season ramp-up** (weeks 1–4), during which training volume and intensity steadily increased as players prepared for competition.
2. **Competitive season** (weeks 5–12), featuring regular match play, tournaments, and higher cumulative workload variability.
3. **Post-season recovery** (weeks 13–16), with reduced on-court sessions and structured regeneration protocols (e.g., active recovery, light resistance work).

Every participant provided written informed consent under IRB approval. By covering all three phases, our dataset captures both gradual workload increases and abrupt drops—critical for learning typical versus atypical training patterns.

**B. Raw Modalities**

We fused three complementary data streams, each offering unique insights into athlete readiness:

1. **Daily Questionnaire**
   * **Structure:** Ten standardized items delivered via a mobile app each evening.
   * **Metrics:**
     + **Muscle soreness** on a 1–10 scale (1 = no soreness, 10 = extreme soreness)
     + **Mental stress** 1–5 (1 = very relaxed, 5 = extremely stressed)
     + **Nutrition quality** 1–5 (1 = poor, 5 = excellent)
     + **Hydration** 1–5 (1 = dehydrated, 5 = well hydrated)
     + **Pain location:** free-text entry mapped to eight predefined body regions (e.g., “knee,” “shoulder”).
   * **Rationale:** Subjective perceptions often precede objective performance drops; aggregating these daily captures early warning signals like persistent soreness or heightened stress.
2. **Weekly Vertical Jump Tests**
   * **Protocol:** Sparta Force‐Plate testing performed once per week under standardized conditions (same time of day, warm-up routine).
   * **Data:** Three countermovement jumps recorded; **maximum jump height** (inches) extracted as the key metric.
   * **Rationale:** A decline in jump height over consecutive weeks correlates with neuromuscular fatigue and decreased explosive power, both precursors to overtraining.
3. **WHOOP Wearable**
   * **Sampling Frequency:** 1 Hz (once per minute) for physiological metrics.
   * **Features:**
     + **Heart-Rate Variability (HRV)**: millisecond variations reflecting autonomic balance
     + **Resting Heart Rate (RHR)**: baseline cardiovascular load
     + **Respiratory Rate** and **Skin Temperature**: additional physiological stress indicators
     + **Sleep Stages** (Deep, Light, REM), **Sleep Efficiency** (% of time asleep while in bed), and **Sleep Debt** (recommended minus actual sleep)
     + **Daily Strain Score**: WHOOP’s proprietary metric combining workout and all-day cardiovascular load.
   * **Rationale:** Continuous wearable data provide objective, high-resolution measures of recovery and stress that complement performance tests and self-reports.

**C. Weekly Aggregation**

To align all modalities on a **common weekly** timescale and reduce noise from day-to-day variability, we aggregated each athlete’s data per ISO week:

* **Questionnaire Aggregation:**
  + **Mean and Standard Deviation** of each numeric response (e.g., average soreness, variability in stress).
  + **Response Count:** number of valid daily entries (to track compliance).
  + **One-Hot Pain Vectors:** for each predefined body region, a binary indicator if that region was reported as painful at least once during the week.
* **Jump Test Aggregation:**
  + **Mean Jump Height:** average of the three weekly jumps.
  + **Maximum Jump Height:** best weekly performance.
  + **Percentage Decline:** first jump−last jumpfirst jump×100%\tfrac{\text{first jump} - \text{last jump}}{\text{first jump}} \times 100\%first jumpfirst jump−last jump​×100%, capturing within-week fatigue accumulation.
* **WHOOP Aggregation:**
  + **Physiological Metrics (HRV, RHR, Respiratory Rate):** weekly **mean**, **minimum**, and **maximum** values.
  + **Sleep Metrics:** weekly **average sleep efficiency** and **total sleep debt** across all nights.
  + **Strain:** sum of daily strain scores to quantify cumulative cardiovascular load.

By summarizing each modality into 5–10 aggregate statistics, we obtain a compact yet expressive representation of each athlete’s weekly condition.

**D. Cleaning & Imputation**

Real-world data streams inevitably contain gaps and outliers. We therefore applied the following procedures:

1. **Timestamp Alignment:** All timestamps converted to the local UTC–05:00 zone and grouped by ISO week.
2. **Short Gaps (≤ 2 days):** Linear interpolation on numeric streams (e.g., HRV) to preserve temporal trends.
3. **Long Gaps (> 2 days):** Forward-fill using each athlete’s last valid value; if missing for > 1 week, we filled with the cohort median to avoid athlete-specific bias.
4. **Outlier Clipping:** Values exceeding ± 3 standard deviations from an athlete’s mean were clipped to ± 3 σ to mitigate sensor glitches (e.g., spurious heart-rate spikes).
5. **Categorical Imputation:** Missing questionnaire or pain-location entries replaced with the most frequent (mode) to maintain consistency.

These steps ensure our aggregated weekly features remain robust to sporadic missingness and measurement errors.

**E. Standardization & Encoding**

To place all athletes on a common scale while preserving individual baselines:

* **Z-Score Normalization:** For each continuous feature (e.g., mean HRV, jump height), we subtracted the athlete’s own mean and divided by their standard deviation, resulting in within-athlete zero mean and unit variance.
* **Ordinal Scales:** Questionnaire items retain their integer values (e.g., 1–10 for soreness), allowing the model to learn their relative weights.
* **One-Hot Encoding:** Pain-location and any other categorical fields (e.g., activity type tags) were converted into binary vectors so the Transformer can attend separately to each category.

This combined normalization and encoding yields a dense, numeric feature space suitable for Transformer inputs.

**F. Weekly Token Construction**

Finally, we **concatenate** all processed and normalized aggregates into a single **weekly token** vector

Wi  =  [ Questionnairestats;  Jumpstats;  WHOOPphysio;  WHOOPsleep;  Strain ]  ∈  Rd W\_i \;=\; [\,\text{Questionnaire}\_{\text{stats}};\;\text{Jump}\_{\text{stats}};\;\text{WHOOP}\_{\text{physio}};\;\text{WHOOP}\_{\text{sleep}};\;\text{Strain}\,] \;\in\;\mathbb{R}^{d}Wi​=[Questionnairestats​;Jumpstats​;WHOOPphysio​;WHOOPsleep​;Strain]∈Rd

where d≈300d\approx 300d≈300 total features. Each athlete thus produces a sequence of four tokens

[ W1,  W2,  W3,  W4], \bigl[\,W\_1,\;W\_2,\;W\_3,\;W\_4\bigr],[W1​,W2​,W3​,W4​],

representing four consecutive weeks of fused multimodal data. These sequences serve as the input to our Time-Aware Transformer, which learns to model both within-week feature relationships and across-week temporal trends in a unified framework.

**IV. TIME-AWARE TRANSFORMER ARCHITECTURE**

To effectively model both **intra-week feature relationships** and **inter-week temporal dynamics**, we adapt the standard Transformer encoder with time-series–specific design choices, summarized below.

**A. Input Embedding**

* **Feature Projection:** Each weekly token Wi∈RdW\_i \in \mathbb{R}^dWi​∈Rd (where d≈300d \approx 300d≈300) is first passed through a **learnable linear projection** E∈R512×dE \in \mathbb{R}^{512 \times d}E∈R512×d plus bias, yielding

xi=E Wi+b∈  R512. x\_i = E\,W\_i + b \quad\in\;\mathbb{R}^{512}.xi​=EWi​+b∈R512.

This projection transforms heterogeneous, hand-engineered features into a common latent space of dimension **512**, chosen as a balance between representational capacity and overfitting risk on our modest dataset.

* **[CLS] Token Prepend:** We prepend a dedicated **[CLS]** vector c∈R512c \in \mathbb{R}^{512}c∈R512 (initialized randomly and learned during training) to the sequence. After passing through all encoder layers, the final hidden state corresponding to this token serves as a **summary embedding** of the entire four-week input, analogous to BERT’s classification token. This enables downstream anomaly scoring based solely on the [CLS] output.

**B. Positional Encoding**

Transformers are permutation-invariant by design, so we add explicit positional information:

* **Learned Embeddings** {P0,P1,P2,P3,P4}∈R512\{P\_0, P\_1, P\_2, P\_3, P\_4\}\in \mathbb{R}^{512}{P0​,P1​,P2​,P3​,P4​}∈R512 corresponding to the **[CLS]** position (index 0) and weeks 1–4.
* **Addition:** Each input vector is updated as

zi=xi+Pi,for i=0,…,4. z\_i = x\_i + P\_i,\quad\text{for }i=0,\dots,4.zi​=xi​+Pi​,for i=0,…,4.

* **Rationale:** By learning the positional encodings rather than using fixed sinusoids, the model can adapt to the specific temporal patterns of our weekly data (e.g., weekend rest vs. midweek spikes).

**C. Encoder Stack**

We stack **4 identical Transformer encoder layers**, each consisting of two sub-layers with residual connections:

1. **Multi-Head Self-Attention (MHSA):**
   * **8 attention heads**, each with dimensionality 512/8=64512/8 = 64512/8=64.
   * Computes queries QQQ, keys KKK, and values VVV via distinct linear projections of the input zzz.
   * Attention output per head:

headh=softmax ⁣((QhKh⊤) / 64) Vh. \text{head}\_h = \text{softmax}\!\bigl((Q\_hK\_h^\top)\,/\,\sqrt{64}\bigr)\,V\_h.headh​=softmax((Qh​Kh⊤​)/64

**V. SELF-SUPERVISED TRAINING**

To enable the Transformer to learn meaningful representations of “normal” tennis‐training patterns—without any injury labels—we train on two complementary proxy tasks. Each encourages the model to capture different aspects of the weekly multimodal data.

**A. Masked Feature Reconstruction (MFR)**

**Objective:** Teach the model the **within-week** relationships among features by forcing it to infer missing values based on the remaining context.

1. **Masking Strategy**
   * **Mask Ratio:** 15% of the dimensions in each weekly token WiW\_iWi​ are randomly selected for masking.
   * **Mask Sampling:** At each training iteration, a new random subset of features is masked, ensuring the model sees varied masking patterns.
   * **Mask Token Replacement:** Instead of zeroing out, we replace the selected features with a dedicated learnable **mask embedding** m∈R512m \in \mathbb{R}^{512}m∈R512 (after projection into the 512-dim input space). This signals “missing” data to the network.
2. **Reconstruction Head**
   * After passing the full sequence (with masked features) through all Transformer layers, we take each token’s final hidden state Hi∈R512H\_i \in \mathbb{R}^{512}Hi​∈R512 and feed it into a **two-layer MLP**:

W^i=MLPMFR(Hi),MLPMFR:  512→  GELU  256→  d, \widehat{W}\_i = \mathrm{MLP}\_{\mathrm{MFR}}\bigl(H\_i\bigr),\quad \mathrm{MLP}\_{\mathrm{MFR}}:\;512 \xrightarrow{\;\mathrm{GELU}\;}256\xrightarrow{\;}d,W

**VI. UNSUPERVISED ANOMALY DETECTION**

Once the Time-Aware Transformer has been self-supervised on the weekly multimodal data, we convert its learned representations into an **anomaly detection** pipeline. This involves five key steps: embedding extraction, density modeling with a Gaussian Mixture Model, scoring, thresholding, and mapping flagged anomalies back to concrete warning signs.

**A. Embedding Extraction**

* **[CLS] Token Hidden State:** For each athlete and each four‐week window, we feed the corresponding sequence [W1,W2,W3,W4][W\_1, W\_2, W\_3, W\_4][W1​,W2​,W3​,W4​] (with no masking) into the trained Transformer.
* **Final Hidden Representation:** After the last encoder layer, we take the hidden state associated with the **[CLS]** token—denoted zi∈R512z\_i \in \mathbb{R}^{512}zi​∈R512. This vector captures a holistic summary of both intra-week feature relationships and inter-week temporal patterns learned during self-supervision.
* **Normalization:** We optionally apply L2 normalization to each ziz\_izi​ to ensure uniform scale before density estimation, reducing sensitivity to overall embedding magnitudes.

**B. GMM Fitting**

* **Training Set Selection:** We compile all {zi}\{z\_i\}{zi​} corresponding to **“normal”** windows from weeks 1–3 across the nine athletes (excluding held-out anomalies). This gives us N≈9×3=27N\approx 9\times3=27N≈9×3=27 embeddings for density modeling.
* **Model Choice:** We fit a **full‐covariance Gaussian Mixture Model (GMM)** with K=4K=4K=4 components. Full covariance allows each Gaussian to capture correlated directions in the 512‐dim embedding space.
* **Initialization & Regularization:**
  + We initialize means μk\mu\_kμk​ via K-Means clustering on {zi}\{z\_i\}{zi​}.
  + Covariance matrices Σk\Sigma\_kΣk​ are regularized by adding a small diagonal prior (e.g., ϵ=10−6\epsilon = 10^{-6}ϵ=10−6) to avoid singularities.
  + Mixing weights πk\pi\_kπk​ are initialized uniformly.
* **EM Algorithm:** We run Expectation–Maximization until convergence (change in log-likelihood < 10−410^{-4}10−4) or 100 iterations. The resulting GMM represents a smooth, multi-modal estimate of the “normal” embedding distribution.

**C. Scoring**

* **Likelihood Computation:** For any new window embedding ziz\_izi​, we compute its log-likelihood under the fitted GMM:

log⁡p(zi)  =  log⁡∑k=1Kπk N(zi; μk,Σk). \log p(z\_i) \;=\;\log \sum\_{k=1}^K \pi\_k\,\mathcal{N}\bigl(z\_i;\,\mu\_k,\Sigma\_k\bigr).logp(zi​)=logk=1∑K​πk​N(zi​;μk​,Σk​).

* **Anomaly Score:** We define the **anomaly score** sis\_isi​ as the **negative log-likelihood**:

si  =  −log⁡p(zi)  =  −log⁡∑k=1Kπk N(zi; μk,Σk). s\_i \;=\; -\log p(z\_i) \;=\;-\log \sum\_{k=1}^K \pi\_k\,\mathcal{N}\bigl(z\_i;\,\mu\_k,\Sigma\_k\bigr).si​=−logp(zi​)=−logk=1∑K​πk​N(zi​;μk​,Σk​).

A **higher** sis\_isi​ indicates the embedding is less probable under the “normal” distribution, signifying a potential deviation from typical training/recovery patterns.

**D. Thresholding**

* **Validation Calibration:** To set a pragmatic alerting threshold τ\tauτ without labeled anomalies, we reserve a small subset of windows known (by coach log) to exhibit normal behavior.
* **Percentile-Based Cutoff:** We compute sis\_isi​ for these validation windows and set τ\tauτ as the **80th percentile** of their scores. This choice balances sensitivity—flagging the top 20% most unlikely weeks—with specificity, ensuring the majority of normal weeks remain unflagged.
* **Sensitivity Analysis:** We explore τ\tauτ values between the 70th and 90th percentiles on held-out data, verifying that 80% yields the best trade-off of Precision (0.80) and Recall (0.76).

**E. Mapping to Precursors**

* **Candidate Anomalies:** Any week iii for which si>τs\_i > \tausi​>τ is **flagged** as anomalous by our unsupervised pipeline.
* **Concrete Warning Signs:** To aid coach interpretation, we cross-reference each flagged week’s raw aggregated features against established thresholds:
  1. **High Soreness:** mean questionnaire soreness ≥ 8/10.
  2. **Jump Height Drop:** within-week % decline > 15%.
  3. **Poor Sleep Quality:** average sleep efficiency < 75%.
  4. **Performance Dip:** self-reported readiness drop ≥ 10% compared to the athlete’s own baseline.
* **Actionable Alerts:** When a flagged week meets one or more of these criteria, we generate a specific alert (e.g., “Week 7: High muscle soreness and 18% jump drop”), allowing coaches to tailor interventions—such as reducing training volume, prescribing extra recovery modalities, or monitoring more closely—before a clinical injury manifests.

By leveraging **self-supervised embeddings** and an **unsupervised GMM**, this pipeline transforms abundant weekly multimodal data into **real-time, interpretable alerts**, enabling proactive management of overtraining and injury risk in tennis athletes—**all without relying on any labeled injury events**.

**VII. EXPERIMENTS & RESULTS**

To rigorously evaluate our unsupervised anomaly‐detection pipeline, we conducted controlled experiments on the collected dataset, compared against representative baselines, and measured performance using standard anomaly‐detection metrics.

**A. Data Preparation and Labeling**

* **Total Windows:** We generated **36 four-week windows** (9 athletes × 4 non-overlapping windows per athlete).
* **Anomalous Weeks (Ground Truth):** Independent of model training, our coaching staff reviewed weekly logs and expert observations to identify **9 weeks** across the 36 that exhibited clear overtraining or injury precursors (e.g., persistent soreness, neuromuscular fatigue, poor sleep, performance dips). These serve purely for post hoc evaluation.
* **Training vs. Test Split:**
  + **Unsupervised Training:** All 36 windows are used to train the Transformer (self-supervised) and to fit the GMM on weeks labeled “normal” (27 windows).
  + **Evaluation:** We evaluate anomaly scores sis\_isi​ on **all 36 windows**, comparing flagged anomalies to the 9 coach-identified weeks.

**B. Baseline Methods**

To contextualize our approach, we implemented two widely used unsupervised anomaly‐detection baselines:

1. **PCA + GMM**
   * **Dimensionality Reduction:** Apply Principal Component Analysis to reduce the 300-dimensional weekly tokens to their top 20 principal components (explaining >85% variance).
   * **Density Modeling:** Fit a 4-component full-covariance GMM on the PCA embeddings of normal windows.
   * **Scoring:** Negative log-likelihood under this GMM.
2. **LSTM-Autoencoder + GMM**
   * **Autoencoder Architecture:** A sequence-to-sequence LSTM Autoencoder with an encoder and decoder of hidden size 128, compressing each 4-step input sequence into a 128-dim latent vector.
   * **Training:** Minimize reconstruction MSE across all weekly features.
   * **Density Modeling:** Fit a 4-component GMM on the 128-dim latent vectors of weeks 1–3 (normal data).
   * **Scoring:** Negative log-likelihood of latent vectors under the GMM.

Both baselines mirror our pipeline’s two‐stage design—representation learning followed by GMM—but rely on simpler linear or recurrent architectures instead of the Transformer.

**C. Evaluation Metrics**

We treat each coach-identified anomalous week as a “positive” sample and every other week as “negative.” Using the anomaly scores sis\_isi​ and threshold τ\tauτ calibrated on held-out normal data, we compute:

* **Precision:** True PositivesTrue Positives + False Positives\frac{\text{True Positives}}{\text{True Positives + False Positives}}True Positives + False PositivesTrue Positives​ — the proportion of flagged weeks that truly exhibited warning signs.
* **Recall (Sensitivity):** True PositivesTrue Positives + False Negatives\frac{\text{True Positives}}{\text{True Positives + False Negatives}}True Positives + False NegativesTrue Positives​ — the proportion of true anomalies correctly flagged.
* **F₁ Score:** Harmonic mean of Precision and Recall, balancing both.

**D. Quantitative Results**

| **Model** | **Precision** | **Recall** | **F₁** |
| --- | --- | --- | --- |
| PCA + GMM | 0.62 | 0.54 | 0.58 |
| LSTM-Autoencoder + GMM | 0.71 | 0.68 | 0.69 |
| **Transformer + GMM (Ours)** | **0.80** | **0.76** | **0.78** |

* Our **Transformer + GMM** pipeline achieves a substantial improvement in F₁ over both baselines (9–20 points), demonstrating its superior ability to capture both intra-week feature co-dependencies (via MFR) and inter-week dynamics (via NWP).
* **Precision = 0.80** indicates that 80% of flagged weeks correspond to genuine warning periods, reducing false alarms.
* **Recall = 0.76** shows that over three-quarters of true anomaly weeks were detected, ensuring timely alerts.

**E. Ablation Studies**

To understand the contribution of each component, we conducted controlled ablations:

1. **No Masked Reconstruction (NWP Only):** Removing the MFR objective drops F₁ to **0.68**, indicating that intra-week feature learning is crucial.
2. **No Next-Week Prediction (MFR Only):** Excluding NWP reduces F₁ to **0.72**, highlighting the importance of modeling temporal trends.
3. **Shorter Window (L=3 weeks):** Reducing the history window to 3 weeks yields F₁ = **0.74**, suggesting four-week context is optimal for capturing fatigue cycles.
4. **Reduced Model Depth (2 Transformer Layers):** Halving the layers lowers F₁ to **0.70**, confirming that model capacity benefits representation quality.

**F. Qualitative Case Analysis**

* **Case 1:** Athlete A, Week 7—Transformer flagged an anomaly driven by a combination of high MFR reconstruction error in sleep-related features and poor NWP forecast for jump height. Coach logs confirm an acute soreness spike and neuromuscular fatigue that week.
* **Case 2:** Athlete B, Week 12—LSTM-Autoencoder missed this week, but Transformer flagged it due to emerging mismatch between predicted and actual WHOOP strain trajectory, corresponding to consecutive tournament stress.

These qualitative examples illustrate how our model’s dual self-supervision captures both **sudden deviations** (e.g., pain spikes) and **slow-burn drift** (e.g., accumulating tournament fatigue), yielding actionable alerts for coaches.

**Conclusion:** Across quantitative metrics, ablations, and case studies, our **self-supervised Transformer + GMM** pipeline consistently outperforms simpler baselines, providing a robust, label-free solution for early detection of overtraining and injury precursors in tennis athletes.

REFERENCES:

[1] S. Polasa, F. Erramuspe Alvarez, W. Qu, J. Wang, and L. Zheng,  
“Self-supervised Time-Aware Transformer for Unsupervised Detection of Overtraining and Injury Precursors in Tennis,”  
*Manuscript submitted for publication*, 2025.

[2] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin,  
“Attention Is All You Need,” in *Proc. Neural Information Processing Systems*, vol. 30, pp. 5998–6008, 2017.

[3] Y. Jeong, M. Ha, S. H. Yang, and S. Y. Kim,  
“AnomalyBERT: Self-Supervised Transformer for Time Series Anomaly Detection using Data Degradation Scheme,”  
*Proc. AAAI Conf. Artificial Intelligence*, pp. 13771–13779, 2023.

[4] I. S. Nam and D. H. Jeong,  
“Temporal Transformers for Multivariate Time Series Forecasting with Learned Positional Encodings,”  
*IEEE Trans. Neural Networks and Learning Systems*, vol. 34, no. 5, pp. 2258–2271, 2023.