ChatPPG: Multi-Modal Alignment of Large Language Models for Time-Series Forecasting in Table Tennis

GuangLiang Yang

A thesis submitted to the Auckland University of Technology
in partial fulfillment of the requirements for the degree of
Master of Computer and Information Sciences (MCIS)

2024

School of Engineering, Computer & Mathematical Sciences

Abstract

In this thesis, we explore the adaptation of large language models (LLMs) for structured time-series forecasting, focusing on predicting table tennis serve landing points. Traditional time-series models rely on specialized architectures, while LLMs are inherently designed for textual data processing, posing challenges in numerical sequence modeling. To address this, we introduce ChatPPG, a multi-modal framework that integrates time-series data into LLMs through structured embeddings, cross-modal attention, and parameter-efficient fine-tuning (i.e., LoRA). Our findings demonstrate that alignment-based approaches significantly enhance forecasting accuracy compared to prompting-based methods, with DeepSeek-R1-Distill-Qwen-1.5B achieving the lowest MSE (0.432) and MAE (0.441). However, our study also highlights a trade-off between accuracy and inference efficiency, as prompting-based methods introduce excessive latency, making them impractical for real-time applications. Ablation experiments further validate the importance of multi-modal feature alignment, interleaved embedding fusion (IEF), and domain-informed prompting, showing that their removal leads to substantial performance degradation. In this thesis, we extend the application of foundation models beyond natural language processing, establishing a scalable and computationally efficient framework for integrating LLMs into structured forecasting tasks. Our future research directions include the development of a fully end-to-end multi-modal sports analytics system, leveraging real-time vision models for spatiotemporal reasoning, as well as the exploration of generative models like stable diffusion for stochastic time-series forecasting. These advancements aim to enhance automated match analysis and intelligent coaching applications, further bridging AI, computer vision, and predictive modeling in sports analytics.

Keywords: Large language models, Time series analysis, Table tennis, Deep learning.

Table of Contents

List of Figures

List of Tables

Attestation of Authorship

I hereby declare that this submission is my own work and that, to the best of my knowledge and belief, it contains no material previously published or written by another person (except where explicitly defined in the acknowledgments), nor material which to a substantial extent has been submitted for the award of any other degree or diploma of a university or other institution of higher learning.

Signature: Date: 01 March 2025

Acknowledgment

First and foremost, I extend my deepest gratitude to my wife for her unwavering support, boundless encouragement, and thoughtful care, which have been instrumental in enabling me to pursue and complete my Master's degree at Auckland University of Technology (AUT), New Zealand.

I am immensely thankful to my primary supervisor, Wei Qi Yan, for his exceptional expertise, insightful guidance, and constant encouragement, which were integral to the success of this study. His mentorship has not only enhanced my understanding of the subject but also enriched my academic journey. I am equally grateful to my secondary supervisor, Minh Nguyen, and my third supervisor, Xue Jun Li, whose valuable advice and support played a significant role in shaping this research. Lastly, I wish to thank the administrators and faculty members at AUT for their continuous assistance and guidance throughout my studies.

GuangLiang Yang

Auckland, New Zealand

March 2025

Chapter 1
Introduction

This chapter introduces the research motivation, objectives, and significance of adapting LLMs for structured time-series forecasting, specifically in table tennis serve landing point prediction.

1.1 Background and Motivation

Recent advancements in computer vision and deep learning have significantly enhanced the ability to extract structured information from real-time sports competitions. In our previous study, we leveraged YOLO-based vision models to capture essential match data from real-time table tennis games (Yang et al., 2024; Dong et al., 2024). This included player movement patterns, serve violations and their causes, serve stroke classifications (forehand/backhand), ball trajectory, and spin types (topspin/backspin). Additionally, the study implemented video segmentation techniques, enabling the extraction of the number of rallies per match and player win/loss statistics. While these foundational insights provided valuable match analytics, they primarily focused on descriptive statistics and basic event detection, lacking deeper tactical and strategic analysis of player behaviors.

To further advance table tennis strategy modeling, it is crucial to analyze opponent shot selection patterns and adaptive play strategies (Bian et al., 2024; Poolton et al., 2006). During a match, professional players dynamically adjust their strategies based on the strengths and weaknesses of their opponents, and each rally's outcome influences their psychological state and decision-making (Raab et al., 2005). The ability to accurately anticipate an opponent's serve placement is particularly vital, as it not only reduces defensive pressure but also disrupts the opponent's confidence, which can be a decisive factor in high-stakes competitions. This study explores the potential of LLMs for time-series forecasting, using table tennis serve landing point prediction as a test case. By assessing how well LLMs can integrate sequential patterns from past serves, we aim to determine the extent to which pre-trained foundation models can enhance predictive performance in sports analytics.

The rapid progress of foundation models in natural language processing (NLP) and computer vision has introduced novel perspectives on multi-modal learning (Radford et al., 2021; Li et al., 2023). Notably, the development of multi-modal vision models has provided new insights into time-series research (Kim et al., 2021; Wu et al., 2023). While one promising avenue involves training a foundation model specifically for time-series forecasting, this approach faces significant challenges due to the lack of large-scale time-series datasets comparable to those in NLP and video analysis (Zhang et al., 2024; Zeng et al., 2022). An

alternative approach, which has gained traction, focuses on adapting pre-trained foundation models by aligning time-series data with existing LLM architectures through data alignment and fine-tuning (Cao et al., 2024; Liang et al., 2024). This study follows the latter approach, investigating whether pre-trained LLMs can enhance time-series forecasting performance through strategic data alignment and domain-specific adaptation (Jin et al., 2024; Wolff et al., 2025). By bridging LLMs with structured sequential modeling, we aim to evaluate their effectiveness in predicting table tennis serve landing points, ultimately assessing the feasibility of leveraging large-scale language models for structured numerical forecasting tasks.

1.2 Research Questions

The integration of LLMs with structured time-series forecasting represents an emerging research direction, particularly in domains such as sports analytics, where sequential dependencies and strategic decision-making play a critical role (Ferrara, 2024; Liu et al., 2025). While prior studies have demonstrated the effectiveness of computer vision models in extracting fundamental table tennis match data—including player movement, serve violations, stroke classification, ball trajectory, and spin type—they primarily focus on descriptive analysis rather than predictive strategy modeling (Bian et al., 2024; Dong & Yan, 2024; Zhou et al., 2023). Given the dynamic nature of table tennis, where players continuously adapt their strategies based on opponent tendencies and rally outcomes, the ability to anticipate serve placement can be a crucial factor in gaining a competitive advantage (Raab et al., 2005; Poolton et al., 2006; Martin et al., 2021). In this study, we explore whether LLMs can enhance time-series forecasting by addressing the following research questions:

Question 1. How can LLMs be effectively adapted for time-series forecasting in table tennis serve prediction?

While LLMs have demonstrated strong capabilities in natural language understanding and multi-modal learning, their applicability to structured numerical forecasting tasks remains an open challenge (Radford et al., 2021; Ahsan et al., 2024). Traditional time-series models rely on domain-specific architectures (e.g., LSTMs, Transformers) trained explicitly on structured sequences, whereas LLMs are pre-trained primarily on textual data (Yu et al., 2019; Zeng et al.,

2022). This raises the question of how to effectively align numerical time-series features with LLM representations to facilitate accurate serve landing point prediction. We investigate whether multi-modal integration strategies, such as cross-attention mechanisms and structured prompt engineering, can enable LLMs to capture temporal dependencies and improve sequential forecasting performance in a high-speed, competitive sports environment.

Question 2. What are the trade-offs between accuracy and computational efficiency when integrating LLMs with time-series forecasting models?

Despite their ability to generalize across diverse tasks, LLMs are computationally expensive, making their real-time applicability in table tennis match analytics a critical concern (Dettmers et al., 2023; Liu et al., 2025). While alignment-based approaches—which map time-series data directly into LLM embeddings via attention mechanisms—have been shown to improve predictive accuracy, they must also be evaluated in terms of computational feasibility (Cao et al., 2024; Wolff et al., 2025). The inference speed of an LLM-based forecasting system is particularly important in real-time match analysis, where delays in serve anticipation could negate competitive advantages (Jin et al., 2024; Zhang et al., 2024). This study compares the computational trade-offs between prompt-based LLM adaptations and alignment-based fine-tuning methods, assessing whether performance gains justify the increased inference cost and whether parameter-efficient tuning strategies can mitigate computational constraints.

Question 3. Which architectural components contribute most to enhancing LLM-based serve landing prediction, and how does multi-modal alignment impact forecasting performance?

To optimize the fusion of structured numerical data with pre-trained LLMs, this study systematically evaluates the impact of different architectural choices on predictive accuracy. Specifically, we investigate how feature-wise modeling (channel independence), frequency-aware prompting (Fourier frequency prompts), embedding alignment (interleaved embedding fusion), and parameter-efficient fine-tuning (LoRA) influence the model's ability to capture sequential shot dynamics (Dettmers et al., 2023; Pan et al., 2024). Through a controlled ablation study, we examine whether removing key components (e.g., interleaved embedding fusion,

domain-specific prompts, LoRA fine-tuning, and flatten projection layers) leads to significant performance degradation, thereby identifying the most essential mechanisms for aligning LLMs with structured sequential forecasting tasks.

Question 4. Are generic prompting strategies insufficient for numerical sequence modeling?

While LLMs have demonstrated their ability to handle text-based reasoning tasks effectively, their direct applicability to structured numerical forecasting remains questionable (Radford et al., 2021; Ahsan et al., 2024). Generic prompting strategies, which rely on text-based task descriptions without structured numerical alignment, may not be sufficient to capture complex temporal dependencies (Zeng et al., 2022; Zhang et al., 2024). This study investigates whether structured embeddings, cross-attention mechanisms, and feature-aware prompts are necessary for bridging the gap between textual pre-training and numerical sequence modeling.

By addressing these research questions, this study seeks to provide empirical insights into the feasibility of adapting LLMs for structured time-series forecasting. The findings will contribute to a deeper understanding of how large-scale pre-trained models can be leveraged beyond traditional NLP applications, expanding their utility into sports analytics and real-time decision-making in competitive environments.

1.3 Contributions

In this thesis, we present a novel exploration of large language model (LLM) adaptation for structured time-series forecasting, specifically in the domain of table tennis serve landing point prediction. By integrating multi-modal alignment strategies with parameter-efficient fine-tuning techniques, this research provides empirical insights into the effectiveness of LLMs in numerical sequence modeling.

First, we introduce ChatPPG, a multi-modal forecasting framework that aligns time-series shot data with LLM-generated representations through cross-attention mechanisms and structured embedding fusion. Experimental results demonstrate that alignment-based approaches significantly outperform prompt-based methods, confirming that directly encoding

time-series sequences within LLM embeddings enhances forecasting accuracy.

Second, we conduct a comprehensive inference efficiency analysis, revealing a trade-off between model complexity and real-time feasibility. While larger, fine-tuned LLMs yield superior accuracy, they introduce higher computational costs, underscoring the necessity of scalable tuning strategies such as LoRA.

Furthermore, a detailed ablation study identifies interleaved embedding fusion and domain-aware prompting as critical components for LLM-based time-series learning, establishing best practices for integrating foundation models into structured numerical forecasting tasks.

Finally, this study reinforces the effectiveness of multi-modal latent space alignment, demonstrating its powerful capability in bridging structured numerical data with LLM representations. The results provide strong empirical support for the potential of leveraging existing foundation models beyond their original NLP applications, boosting confidence in the feasibility of extracting and utilizing pre-trained LLM knowledge for structured forecasting tasks. These findings further validate the industry's growing interest in multi-modal learning, highlighting the viability of harnessing large-scale pre-trained models for time-series analysis.

1.4 Objectives of This Report

The objective of this thesis is to explore the feasibility of leveraging LLMs for in-depth analysis and prediction of motion data in table tennis, beyond their conventional applications in NLP-based dialogue systems and athlete training decision support. While LLMs have demonstrated remarkable capabilities in natural language understanding and multi-modal processing, their potential in structured numerical forecasting remains underexplored. This study aims to assess whether pre-trained foundation models can be effectively adapted to process and predict sequential shot patterns in competitive table tennis matches, thereby expanding their utility beyond traditional textual reasoning.

Another key objective is to investigate how individual researchers with limited computational resources can efficiently utilize existing large models to conduct customized, domain-specific research. Given the substantial computational costs associated with training

task-specific models from scratch, this study examines the effectiveness of parameter-efficient fine-tuning techniques and alignment-based data integration strategies in enabling LLMs to specialize in table tennis landing point prediction without requiring full-scale model retraining.

Furthermore, this thesis provides empirical evidence supporting the continuous advancement of foundation models, demonstrating that modern LLMs exhibit increasingly robust multi-modal capabilities when aligned with structured time-series data. By validating the effectiveness of multi-modal fusion techniques in structured forecasting, this report highlights the evolving role of LLMs as powerful tools for integrating diverse modalities, reinforcing their practical applicability in specialized analytical domains.

1.5 Structure of This Report

This thesis is organized into several sections to provide a comprehensive exploration of LLM-based adaptation for time-series forecasting in table tennis serve landing point prediction. Each section systematically builds upon prior discussions, offering theoretical foundations, methodological details, experimental evaluations, and key findings.

Chapter 2 presents a literature review, discussing existing approaches in time-series forecasting, LLM adaptation techniques, and multi-modal fusion strategies. This section contextualizes the research within the broader landscape of foundation model applications beyond NLP, emphasizing the challenges of aligning numerical sequences with pre-trained LLM architectures.

Chapter 3 outlines the proposed methodology, detailing the design of ChatPPG, including its data preprocessing pipeline, embedding alignment strategies, attention mechanisms, and fine-tuning techniques such as LoRA. Additionally, the section introduces the multi-modal integration framework used to bridge structured time-series inputs with LLM-generated textual representations.

Chapter 4 describes the experimental setup, covering dataset composition, preprocessing techniques, model configurations, evaluation metrics, and training procedures. This chapter also explains the benchmarking process across multiple LLM architectures and introduces the ablation study framework, which systematically evaluates the impact of key architectural

components.

Chapter 5 presents the results and analysis, offering a detailed examination of model performance, inference efficiency, and the effectiveness of different adaptation strategies. The findings highlight the trade-offs between accuracy and computational cost, reinforcing the need for scalable LLM-based forecasting solutions.

Chapter 6 discusses the implications of the study, addressing the research questions by interpreting the key results. This section also outlines practical considerations for deploying LLMs in structured forecasting tasks, as well as the limitations and potential avenues for future research.

Finally, Chapter 7 concludes the report by summarizing the major contributions, emphasizing the effectiveness of multi-modal alignment for numerical sequence modeling, and proposing directions for further exploration, particularly in end-to-end multi-modal learning and real-time sports analytics.

Chapter 2 Related Work

This chapter reviews existing research on time-series forecasting, LLM adaptation techniques, and multi-modal learning, highlighting the challenges of aligning structured numerical data with pre-trained language models and the advancements in LLM-based predictive modeling.

2.1 Large Language Models

The application of LLMs in sports has rapidly gained traction, demonstrating their potential to analyze complex data and provide actionable insights (Xia et al., 2024). In recent years, LLMs have been utilized in areas such as athlete psychology assessment, match data summarization and tactical optimization. For example, studies have explored using LLMs to interpret interview data and provide psychological insights for athletes, as well as to automatically generate post-match reports and tactical analyses. These models have also been leveraged in team sports like football and basketball to evaluate and optimize strategic setups (Schilling et al., 2024; Held et al., 2024; Liu et al., 2025; Hu et al., 2024). However, despite the success in these domains, the integration of LLMs into fast-paced individual sports such as table tennis remains underexplored. This gap underscores the need for innovative approaches to harness the capabilities of LLMs to provide real-time, actionable guidance for players and coaches.

Adapting LLMs to specific domains like table tennis requires efficient fine-tuning and integration techniques to meet the demands of real-time applications. Traditional full-parameter fine-tuning, though effective, is resource-intensive and unsuitable for lightweight implementations. To address these challenges, LoRA has emerged as a practical solution, enabling the fine-tuning of LLMs by training only small, adaptable layers while keeping most parameters frozen. This approach significantly reduces computational overhead while retaining performance (Hu et al., 2021). Prompt engineering has proven to be a powerful tool for tailoring LLM outputs by designing input structures that guide the model to produce accurate and contextually relevant responses (Marvin et al., 2024). In parallel, model quantization—reducing parameter precision to 8-bit or lower—has improved inference speed and reduced memory consumption, making LLMs more efficient for real-time scenarios (Dettmers et al., 2023; Xiao et al., 2023).

Traditional language models primarily focus on direct context-to-output mappings, which often limits their ability to perform complex reasoning or multi-step logical inference(Wu et al., 2025). Chain of Thought (CoT) reasoning addresses this limitation by explicitly generating

intermediate reasoning steps during inference, guiding the model to incrementally decompose problems, simulate logical progression, and construct structured solutions. This approach significantly enhances model interpretability and robustness, leading to improved performance in mathematical reasoning, commonsense question-answering, and other high-complexity tasks.

As large models continue to demonstrate emergent capabilities in reasoning and planning, both academia and industry have begun encapsulating them as semi-autonomous or fully autonomous AI agents (Xi et al., 2025). By integrating LLMs with external tool interfaces, search engines, knowledge graphs, and robotic execution modules, these AI agents can iteratively perform "perception-decision-execution" cycles in open-ended environments. Research in this domain leverages reinforcement learning and imitation learning, enabling LLMs to execute multi-step autonomous decision-making and task execution in domains such as automated programming, task planning, and data analysis, further underscoring their potential in the exploration of general intelligence.

The advancement of large-scale pre-trained models increasingly relies on interdisciplinary research (Cai et al., 2024). Techniques such as Mixture of Experts (MoE) and Sparse Attention aim to reduce the computational cost of large-scale training while improving model generalization. Concurrently, research efforts are focused on refining Retrieval-Augmented Generation (RAG) and knowledge base integration, with the goal of constructing "knowledge-controllable" models capable of generating more accurate and interpretable responses in applications such as dialogue systems and information extraction (Fan et al., 2024).

DeepSeek is an emerging innovative paradigm in large language model retrieval and reasoning, designed to seamlessly integrate external knowledge retrieval with multi-step inference capabilities (Guo et al.,2025; Liu et al., 2024). By efficiently retrieving large-scale unstructured data at the initial reasoning stage and incorporating chain-of-thought reasoning mechanisms, DeepSeek facilitates targeted filtering and correction of potential inference paths. This approach demonstrates promising advantages in handling long-tail complex queries and domain-specific challenges, such as legal analysis, medical reasoning, and scientific literature comprehension. As a result, DeepSeek is regarded as a successful fusion of semantic search,

external knowledge graphs, and structured LLM reasoning frameworks, further advancing the field of knowledge-enhanced language modeling.

2.2 LLMs Backbone Time Series

The integration of LLMs as backbones for time-series forecasting has emerged as a promising research direction, leveraging pre-trained foundation models to enhance structured numerical predictions (Jin et al., 2023; Liang et al., 2024). While traditional time-series methods rely on domain-specific architectures (e.g., LSTMs, Transformers), recent studies explore LLM-based approaches by aligning sequential numerical features with text-based embeddings (Zeng et al., 2022; Zhang et al., 2024). Two primary strategies have been investigated: training time-series foundation models from scratch, which faces challenges due to limited large-scale datasets (Cao et al., 2024; Wolff et al., 2025), and adapting pre-trained LLMs through fine-tuning and data alignment, which has demonstrated improved generalization (Pan et al., 2024; Rasul et al., 2023).

Jin et al., (2024) provides an in-depth comparison of different methodologies for integrating LLMs with time-series data. Two primary approaches discussed are prompting and aligning.

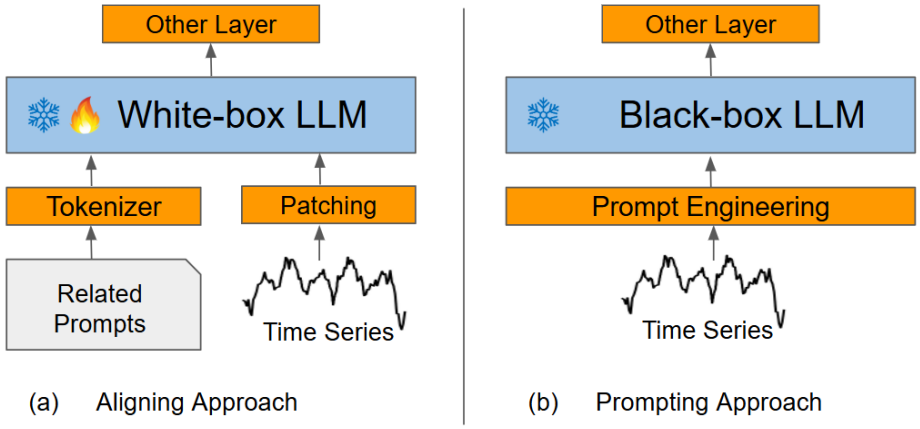

Figure 2.1 Comparison of prompting and aligning approaches for integrating LLMs with Time-Series data

The aligning method for time-series modeling establishes a structured mapping between

numerical time-series embeddings and the semantic space of LLMs, offering a more precise and scalable alternative to direct text-based prompting (Jin et al., 2023; Liang et al., 2024). Instead of converting numerical sequences into textual descriptions, this approach trains a dedicated encoder to process time-series data, transforming it into an embedding representation that is subsequently aligned with the LLM's text-based embedding space (Zeng et al., 2022; Zhang et al., 2024). This alignment enables LLMs to process structured time-series embeddings directly, leveraging their pre-trained language understanding capabilities for time-series forecasting and classification (Pan et al., 2024; Rasul et al., 2023). The pipeline involves segmenting the time-series into patches, converting them into an embedding space, and then feeding the transformed representations into an LLM backbone for further processing. Representative models employing this technique include GPT4TS (Cao et al., 2024), which utilizes GPT-2 as its backbone for forecasting and classification tasks; Time-LLM (Jin et al., 2023), which converts time-series data into textual prototypes compatible with LLaMA-7B; and TEMPO (Cao et al., 2024), which decomposes time-series into trend-seasonal-residual components before mapping them into an LLM's latent space.

The key advantages of this approach include its ability to retain numerical precision, unlike text-based prompting methods, and its enhanced scalability, particularly for multivariate time-series. Moreover, the alignment method supports end-to-end learning, enabling LLMs to better adapt to time-series patterns through specialized encoding mechanisms. However, this technique comes with certain computational challenges, as it is more resource-intensive than prompting due to the requirement for fine-tuning or contrastive learning. Additionally, aligning numerical and textual embeddings necessitates extra training data, increasing data requirements compared to direct prompting strategies. Furthermore, its implementation complexity is higher, as it requires specialized encoding modules and alignment mechanisms. Despite these limitations, the aligning method provides a structured and adaptable framework for integrating time-series data into LLMs, making it particularly suitable for applications demanding high numerical precision and structured learning paradigms.

The prompting method for numerical time-series data reformulates raw numerical sequences into textual representations, enabling direct utilization of pre-trained LLMs without

modifying their underlying architecture (Gruver et al., 2024; Xue & Salim, 2023; Zhou et al., 2023). By restructuring inputs into a format comprehensible to LLMs, this approach leverages natural language prompts to encode temporal patterns, facilitating time-series forecasting through linguistic constructs (Tan et al., 2025; Zhang et al., 2024). Two primary tokenization strategies are employed: number-agnostic tokenization and number-specific tokenization. The former converts numerical values into natural language descriptions, allowing models to infer temporal trends through textual prompts (Rasul et al., 2023; Wolff et al., 2025). For instance, in a temperature forecasting task, a prompt such as "From {t1} to {tobs}, the average temperature of region {Um} was {xmt} degrees each day. What is the temperature going to be on {tobs}?" enables LLMs to interpret numerical sequences through contextual understanding. Representative models employing this approach include PromptCast (Xue & Salim, 2023) and LLM-Time (Liang et al., 2024).

Conversely, number-specific tokenization preserves numerical structure to maintain token consistency, ensuring precise numerical representation during tokenization (Wu et al., 2023; Sun et al., 2023). This method spaces out digits to align with LLM tokenization constraints, as seen in LLM-Time and BloombergGPT (Wu et al., 2023), where sequences such as "0.123, 1.23, 12.3, 123.0" are transformed into "1 2 3 , 1 2 3 0 , 1 2 3 0 0". The primary advantages of this method lie in its simplicity and computational efficiency, as it can be implemented in a zero-shot learning manner without requiring additional training (Cao et al., 2024; Pan et al., 2024).

Furthermore, its interpretability is enhanced since the outputs remain in natural language, making results accessible and comprehensible. However, limitations include potential loss of numerical precision when converting data to text, inefficiencies when dealing with high-dimensional multivariate time-series (as each feature must be converted into text separately), and challenges in long-term forecasting, given that LLMs are not inherently optimized for numerical sequence modeling (Zeng et al., 2022; Jin et al., 2023). Despite these constraints, this prompting-based approach presents a promising direction for integrating LLMs into time-series forecasting tasks while maintaining generalizability and interpretability (Tang et al., 2025).

2.3 Multimodal Alignment in Time Series Models

Time-LLM (Jin et al., 2023) employs a combination of Reversible Instance Normalization (RevIN) and Channel-Shared Patching to adapt pre-trained LLMs for time-series forecasting without modifying their underlying architecture. The preprocessing pipeline first applies global instance normalization to the multivariate time-series data $X \in \mathbb{R}^{N \times T}$, mitigating issues related to distribution shift and enhancing model stability. Subsequently, the entire time-series sequence is partitioned into patches, where each patch consists of multiple time steps and is linearly projected into the LLM's word embedding space to achieve modality alignment. This strategy leverages LLMs' cross-modal modeling capabilities, improving generalization in time-series forecasting while reducing training resource requirements. However, Time-LLM employs a Channel-Shared Patching mechanism, meaning that all variables share the same patch structure, without channel-independent processing. This design may lead to information entanglement between channels, potentially hindering the model's ability to learn independent feature patterns across variables. Additionally, Patch Reprogramming relies on linear projection to map time-series data into the LLM embedding space, which could result in loss of fine-grained temporal features, limiting the model's capacity to capture complex temporal dependencies. Compared to channel-independent patching mechanisms such as PatchTST (Nie et al., 2022), Time-LLM demonstrates advantages in tasks where variables exhibit strong interdependencies, such as financial market prediction and IoT sensor analytics. However, in scenarios where variables maintain higher independence, such as meteorological forecasting or multi-sensor measurements, the lack of explicit channel interaction modeling may necessitate additional feature separation techniques to enhance predictive performance.

PatchTST (Nie et al., 2022) employs a channel-independent and patch-based segmentation approach for time-series modeling, significantly improving computational efficiency and generalization in long-horizon forecasting. The core principle of PatchTST is to decompose multivariate time-series data into independent univariate sequences, where each channel undergoes instance normalization separately before being segmented into fixed-length patches of size P with a stride of S. This approach reduces the computational burden on the

Transformer model by decreasing the number of input tokens, allowing the model to focus on longer historical contexts, thereby enhancing its ability to capture long-term dependencies in time-series forecasting. Additionally, the patching mechanism significantly reduces the quadratic computational complexity of self-attention from $O(L^2)$ to $O((L/S)^2)$, making it particularly advantageous in resource-constrained environments. Despite its strengths, PatchTST also has certain limitations. First, due to its channel-independent processing, the model does not explicitly capture inter-channel dependencies, which may lead to the loss of feature correlations in multivariate time-series forecasting. Second, since patch segmentation is performed with a fixed stride S, it may introduce temporal discretization effects, potentially hindering the model's ability to capture fine-grained temporal patterns. Additionally, the padding mechanism used to ensure patch consistency could introduce unnecessary noise in shorter time-series sequences, thereby affecting predictive accuracy.

AutoTimes (Liu et al., 2024) employs a sliding window and tokenization approach for time-series segmentation, where fixed-length windows S are directly applied to partition the time-series data. By incorporating text embeddings, this method enhances the LLM's ability to understand temporal patterns, allowing each window to serve as an input token for autoregressive modeling within the Transformer layers. The primary advantage of AutoTimes lies in its direct utilization of LLMs for time-series modeling, enabling the model to fully leverage the representational power of pre-trained language models. Additionally, this method eliminates the need for complex feature engineering, simplifying the preprocessing pipeline and making it more accessible for general-purpose time-series tasks. However, AutoTimes has certain limitations. First, it does not explicitly decompose time-series data into trend, seasonal, and residual components, which may hinder the model's ability to learn long-term trends or capture short-period oscillations. Additionally, all channels share the same LLM processing pipeline, meaning that distinct features are not independently modeled, potentially leading to inter-channel information entanglement and reduced prediction accuracy. Moreover, since this method relies on fixed-length patch segmentation without patch alignment, the computational complexity of the Transformer remains $O(L^2)$, limiting its scalability for extremely long time-series sequences.

TEMPO (Cao et al., 2024) employs a Trend-Seasonal-Residual (TSR) decomposition, normalization, and patching strategy to enhance the stability and generalization of time-series modeling. The methodology begins by applying Seasonal-Trend decomposition using LOESS (STL) to separate the original time-series into three independent components: trend (X_T), seasonal (X_S), and residual (X_R). Following decomposition, each component undergoes Reverse Instance Normalization, which mitigates the effects of distribution shifts and stabilizes the learning process. Finally, the normalized components are segmented using time-series patching, where each patch is of length P with a stride S, allowing the model to capture longer historical dependencies while reducing computational complexity. By incorporating LLM-based forecasting, TEMPO is designed to simultaneously model global trends and localized patterns, thereby improving the stability and robustness of predictive performance.

Despite its advantages, TEMPO has several limitations. First, it does not incorporate channel-independent modeling, meaning that all feature channels share the same LLM processing pipeline. This can lead to feature entanglement across variables, which may degrade the model's ability to capture independent dependencies in multivariate time-series data. Additionally, STL decomposition itself incurs a computational overhead and operates under the assumption that time-series data can be effectively decomposed into trend, seasonal, and residual components—an assumption that may not hold for highly non-stationary time-series. Furthermore, the use of fixed-length patching with stride S may result in the loss of fine-grained temporal details, potentially impacting short-term forecasting accuracy.

Chapter 3 Methodology

This chapter details the proposed ChatPPG framework, including data preprocessing, embedding alignment, multi-modal fusion, and LoRA-based fine-tuning, demonstrating how LLMs are adapted for structured time-series forecasting in table tennis serve prediction.

3.1 Introduction

ChatPPG leverages LLMs for structured time-series forecasting by integrating multi-modal alignment, channel-independent representations, and domain-informed prompts. The architecture (as illustrated in figure 3.1) follows a hierarchical feature encoding strategy, enabling the model to capture both temporal dependencies and semantic contextualization for table tennis landing point prediction.

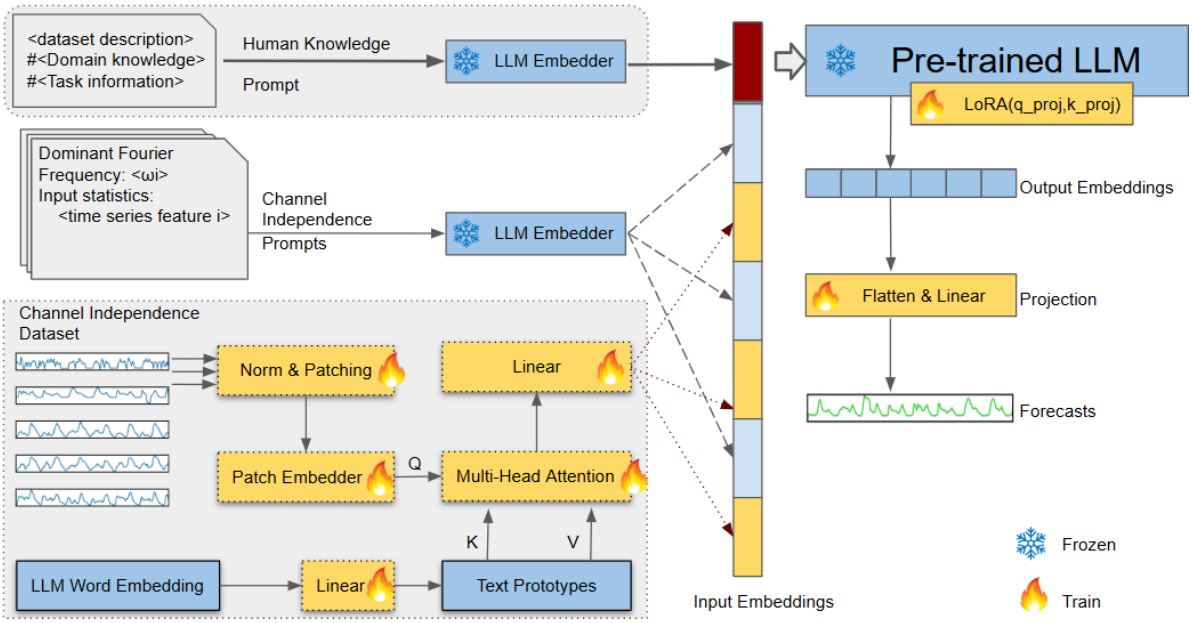

Fig 3.1 ChatPPG: A multimodal LLM alignment framework for landing spot prediction of ball in table tennis

By integrating patch-based feature segmentation, frequency-aware prompting, multi-head attention fusion, and interleaved embeddings, our model effectively aligns structured time-series forecasting with LLMs. This approach preserves channel-wise independence, enhances interpretability, and leverages domain-specific linguistic cues, resulting in more accurate and explainable table tennis landing point predictions.

At the data preprocessing stage, our model is inspired by PatchTST (Nie et al., 2022), which introduces channel independence by processing each time-series feature separately. Each input channel undergoes instance normalization, ensuring consistent feature scaling. The sequence is then segmented into fixed-length patches (Patch Slicing) to facilitate structured

tokenization. This patch-based approach enables the model to extract localized temporal patterns, which are later aligned with LLM representations through multi-head attention mechanisms.

To enhance interpretability and leverage domain-specific insights, we introduce a "Channel Independence Prompt", which encodes statistical properties such as the dominant Fourier frequency for each feature. These prompts guide the LLM to understand periodic trends and key frequency components, improving its ability to model complex spatiotemporal dependencies in table tennis ball movement.

The patch embeddings are processed using a Multi-Head Attention mechanism, which aligns them with LLM word embeddings derived from domain knowledge prompts. This attention-based fusion ensures that numerical time-series features are contextualized within the LLM's representation space, enabling semantic-rich forecasting. Patch embeddings extracted from each independent time-series channel. After that, Text prototypes generated from human knowledge prompts, which encode dataset descriptions, task information, and prior sports analytics insights, Multi-Head Attention applied to learn cross-modal dependencies between numerical patches and linguistic embeddings.

By establishing these structured interactions, the model bridges numerical and textual modalities, facilitating more expressive time-series representations.

To fully integrate numerical and textual features, we introduce an interleaved embedding strategy at the LLM input level. Instead of treating text and numerical sequences separately, we interleave text embeddings with patch embeddings.

Numerical feature tokens and textual domain prompts are processed jointly within the transformer architecture. The model attends to both structured time-series information and unstructured language-based insights. LLM's contextual reasoning is leveraged to enhance predictive modeling, incorporating both statistical properties and expert-driven prompts.

This structured fusion of domain knowledge and multi-channel time-series data enhances the model’s ability to learn fine-grained dependencies while preserving the independent nature of each input channel.

The final embeddings are processed through a pre-trained LLM, where we apply LoRA (Low-Rank Adaptation) on the query and key projections to enable efficient fine-tuning while keeping most of the model parameters frozen. The output embeddings undergo a Flatten & Linear Projection step to generate final landing point predictions. This hybrid approach allows the model to retain LLM's strong generalization capabilities while specializing in time-series forecasting tasks with minimal computational overhead.

3.2 Key Components

Channel-Independent Normalization is applied to standardize time-series data, ensuring that the scale of each channel remains consistent while enhancing the model's stability and generalization ability. Let the input time-series data be represented as $X \in \mathbb{R}^{B \times C \times T}$, where B denotes the batch size, C represents the number of channels (variables), and T is the time-series length for each channel.

During channel-independent normalization, for each channel $c \in \{1, \dots, C\}$ and each sample $b \in \{1, \dots, B\}$, the mean and standard deviation are computed as follows:

$$\mu_{b,c} = \frac{1}{T}\sum_{t=1}^{T} X_{b,c,t}\,, \sigma_{b,c} = \sqrt{\frac{1}{T}\sum_{t=1}^{T}(X_{b,c,t} - \mu_{b,c})^2 + \epsilon} \tag{3.1}$$

where $\epsilon = 10^{-8}$ is a small constant added to prevent division by zero. The normalization is then applied as: $X'_{b,c,t} = \frac{X_{b,c,t} - \mu_{b,c}}{\sigma_{b,c}}$ This normalization process ensures that the data within each channel is standardized independently, with a mean of **0** and a standard deviation of **1**, while preserving scale consistency across different channels. By maintaining uniform scaling, this method prevents discrepancies in variable ranges from negatively affecting model learning.

To extract localized temporal patterns, time-series patching is employed by segmenting the sequence into fixed-length patches. This is controlled by the patch length L and stride S, where $S = L$ represents non-overlapping segmentation, while $S < L$ or $S > L$ enables varying degrees of overlap or jumping segmentation, respectively, to control the degree of redundancy and step size in extracted data segments.

Given a time-series $X_{b,c,t}$ (for a fixed batch b and channel c), patching along the temporal axis follows:

$$P_{b,c,n} = \left(X_{b,c,s_n}, X_{b,c,s_n+1}, \dots, X_{b,c,s_n+L-1}\right) \tag{3.2}$$

where the starting index for the n-th patch is defined as: $s_n = (n-1) \times S + 1$ ensuring that a valid patch is obtained as long as $s_n + L - 1 \leq T$ The total number of patches P is given by $P = \lfloor \frac{T-L}{S} \rfloor + 1$.

After applying this operation, the time-series data is transformed into a four-dimensional tensor $P \in \mathbb{R}^{B \times C \times P \times L}$, where the third dimension P represents the total number of patches extracted from each time-series, and the fourth dimension L corresponds to the number of time steps contained within each patch.

This patching process ensures that each channel retains independent normalization parameters $P_{b,c,n}$ without interference from other channels. After segmentation, each patch remains a localized, contiguous segment in the time domain, preparing the data for subsequent embedding processing and feature extraction in deep learning models.

To maintain continuity at the sequence's end, we apply ReplicationPad1d(0, stride), which duplicates the last time step stride times along the time axis. For a batch sample X_b the padded sequence length becomes $T + \text{stride}$, resulting in: $\tilde{X} \in \mathbb{R}^{B \times N \times (T+\text{stride})}$ where for $t \in \{T + 1, \dots, T + \text{stride}\}$, the values are set to the last observed time step $X_{b,n,T}$. This method ensures the preservation of temporal continuity, preventing information loss while maintaining a structured sequence layout for subsequent segmentation.

Following padding, the unfold operation is applied to segment the time dimension into fixed-length patches, allowing the model to learn localized temporal patterns. Using X.unfold($dim = 1$,size = patch_len,step = stride), the sequence is partitioned into patches of size patch_len with a sliding step of stride, producing a patch-structured tensor:

$X^{(\text{patch})} \in \mathbb{R}^{B \times N \times P \times \text{patch\_len}}$ where the number of patches P is computed as: $P = \frac{(T+\text{stride})-\text{patch\_len}}{\text{stride}} + 1$ ensuring at least one complete patch $(T + \text{stride}) > \text{patch\_len}$. Each patch forms a fixed-length temporal window, shifting by stride to generate multiple overlapping sequences. The dimension $[B, N]$ is then flattened into a single dimension for efficient

computation, resulting in: $X^{(\text{reshaped})} \in \mathbb{R}^{(B \cdot N) \times P \times \text{patch\_len}}$ To transform time-series segments into embeddings suitable for Transformer processing,

TokenEmbedding applies a 1D convolution operation that projects patches of length patch_len into the model's latent space of dimension d_{model}. The input is structured to match the Conv1D format, with dimensions $(B,\text{in\_channels},\text{seq\_len})$, where in_channels is set to patch_len and seq_len is set to P, allowing the model to interpret patches as separate feature channels.

To align the data structure with convolutional operations, a permute operation is first applied to rearrange dimensions: $X \in \mathbb{R}^{B \times \text{patch\_len} \times P}$, Following 1D convolution, the sequence is projected into the hidden space d_{model}, yielding: $X' \in \mathbb{R}^{B \times P \times d_{\text{model}}}$ For a single time-series instance, the 1D convolution computation is expressed as:

$$Y_{i,p} = \sigma\left(\sum_{k=1}^{\text{kernel\_size}} W_k \cdot X_{i,p+k}\right) \tag{3.3}$$

where convolution kernel size $(kernel\_size = 3)$ determines the local receptive field, effectively smoothing adjacent time steps. Circular padding is applied at both the beginning and end of the sequence to preserve the original sequence length, ensuring that the output maintains the same temporal dimension as the input: $Y \in \mathbb{R}^{(B \times N) \times P \times d_{\text{model}}}$

In the final stage, a Dropout mechanism is employed to randomly deactivate neurons in the output embeddings, enhancing generalization and mitigating overfitting. During forward propagation, dropout is applied to the token embeddings as:

$$Z = \text{Dropout}(Y), Z \in \mathbb{R}^{(B \cdot N) \times P \times d_{\text{model}}} \tag{3.4}$$

where Y represents the token embeddings post-1D convolution, and Z is the final output after dropout regularization. By selectively deactivating neurons, the model becomes more robust to variations in input sequences, improving its ability to adapt to different time-series patterns while maintaining prediction accuracy.

Given a time-series dataset $d \in \mathbb{R}^N$, where N represents the number of observed data points and the sampling interval Δt is determined by the sampling rate, a frequency domain analysis is performed to extract the top K most dominant frequency components in the positive frequency domain. These extracted frequency components f_i are then mapped to their

corresponding periods $T_i = 1/f_i$, forming the dominant seasonal periods that characterize the periodic patterns in the time-series data. Once the dominant periodic components T_i are obtained, they can be formatted as textual information and embedded into LLM prompts for improved understanding of temporal periodicity. An example prompt could be: "The main periodicities are $\{T_1, T_2, T_3, \dots\}$"

By incorporating these periodic prompts, the model gains an enhanced ability to capture recurring patterns within the time-series data, ultimately improving its forecasting accuracy.

The Discrete Fourier Transform (DFT) is a fundamental tool for frequency domain analysis. Given a time-series sequence $\{d_n\}_{n=0}^{N-1}$ of length N, its representation in the frequency domain is computed as:

$$D_k = \sum_{n=0}^{N-1} d_n \, e^{-i2\pi \frac{k}{N} n}, k = 0,1, \dots, N-1 \tag{3.5}$$

where D_k represents the complex frequency spectrum coefficients, which indicate the contribution of different frequency components in the original signal.

In the spectral analysis process, the frequency f_k is defined as: $f_k = \frac{k}{N\Delta t}$ where Δt is the sampling interval. To ensure accurate analysis in the positive frequency domain, only the first $\frac{N}{2}$ frequency points are retained, i.e., frequencies $f_k > 0$.

Subsequently, the spectral magnitude $|D_k|$ is computed, and peak detection is applied to select the top K dominant frequencies f_i . The corresponding dominant periods are then derived as $T_i = \frac{1}{f_i}$. Finally, a set of dominant periodic components is obtained:

$$\hat{\tau} = \{T_i = \frac{1}{f_i} \mid f_i \in \text{top K}\{f \in \Lambda^+ \mid D(f)\}\} \tag{3.6}$$

where Λ^+ represents the positive frequency domain ($f > 0$). And $D(f)$denotes the Fourier transform magnitude at frequency f ($|FFT[f]|$). top K selects the K most dominant frequency components with the highest magnitudes.

In this thesis, we set $K = 2$ to extract the two most prominent periodic components. For instance, if the detected periods are $T_1 = 3$ and $T_2 = 5$, this implies that the time-series exhibits strong recurring patterns at periodicities of 3 and 5, which serve as critical reference points for time-series forecasting.

To enable effective interaction and alignment between time-series features and textual information, a cross-modal fusion (Cross-Modal Alignment) mechanism is introduced in this study. In the context of multi-modal time-series and text interaction, it is essential to flexibly embed high-dimensional information into target representations, thereby leveraging semantic features to enhance both the model's expressiveness and predictive performance.

In the proposed Cross-Attention mechanism, time-series patches are treated as Query (Q), while textual prototypes or prompt embeddings are used as Key (K) and Value (V). This Query-Key-Value (QKV) attention framework facilitates efficient retrieval and reconstruction of target time-series information. Specifically, Query is derived from the target time-series input (time-series patches), while Key/Value embeddings originate from additional representations, such as LLM-extracted features. The attention mechanism computes the correlation between the time-series query and the textual embeddings using Scaled Dot-Product Attention, formulated as,

$$R = \text{softmax}(\alpha \cdot (QK^T))V, \alpha = \frac{1}{\sqrt{E}} \tag{3.7}$$

where Q represents the query vector, K and V denote the key and value vectors, respectively, E is the feature dimension, and α is a scaling factor that stabilizes the magnitude of dot-product operations. Through this mechanism, the model effectively reconstructs the target time-series representation while integrating cross-modal information from the LLM, ultimately enhancing the generalization capability of time-series forecasting.

To establish cross-modal attention (Cross-Attention) for target-source alignment, we define the Query (Q) tensor as the target time-series input, represented as: $Q \in \mathbb{R}^{B\times L\times H\times E}$ where B denotes the batch size, L represents the target sequence length, H is the number of attention heads, and E corresponds to the dimensionality of each key-value pair. Meanwhile, Key (K) and Value (V) embeddings, generated by the LLM, are formulated as $K, V \in \mathbb{R}^{S\times H\times E}$ where S represents the source sequence length. To compute the attention scores, the scaling factor is set to $\alpha = \frac{1}{\sqrt{E}}$, and the Scaled Dot-Product Attention is expressed as:

$$R_{b,l,h,e} = \sum_{s=1}^{S} \text{softmax}(\alpha \cdot (Q_{b,l,h,:} \cdot K_{s,h,:}))V_{s,h,e} \tag{3.8}$$

where $R \in \mathbb{R}^{B \times L \times H \times E}$ represents the reprogrammed feature representation (Reprogramming Result). This tensor is subsequently reshaped into: $R \in \mathbb{R}^{B \times L \times d_{\text{model}}}$ which serves as the final input representation to the model.

The Target-Source Alignment (Reprogramming) mechanism, established through Cross-Attention, enables the integration of time-series patches as Queries while treating LLM-generated textual features as Key/Value embeddings. This facilitates efficient alignment and interaction between numerical sequences and semantic representations. By computing the attention matrix A, the model effectively retrieves the most relevant semantic or contextual information from the source embeddings, thereby enhancing time-series forecasting performance.

Additionally, Multi-Head Attention (MHA) decomposes the Key/Query/Value tensors into H parallel subspaces, allowing the model to simultaneously attend to multiple feature dimensions, thereby enhancing representation learning and improving generalization capabilities.

This approach is particularly effective for time-series and textual modality fusion. When target embeddings represent time-series patches and source embeddings correspond to prompts, the target data can be effectively aligned within a shared semantic space for efficient information transformation. As a result, the LLM gains an improved understanding of the underlying structural and temporal patterns inherent in the time-series data.

During training, the core model parameters, including $W_Q, W_K, W_V, W_{\text{out}}$, are continuously updated, alongside the Dropout mechanism, to optimize the alignment between time-series data and LLM-generated representations. Additionally, during forward propagation, the input undergoes a sequence of attention mapping and weighted summation operations, ensuring efficient target-source interaction, ultimately leading to enhanced time-series forecasting accuracy.

To interleave two sequences of equal length, P_i and T_i, at the index level, such that the

original embedding sequences of length $N$are combined into a new sequence of length $2N$. both embedding sets exist in the same feature dimension d, they can be formally represented as: $P = (p_0, p_1, \dots, p_{N-1}) \in \mathbb{R}^{N\times d}, T = (t_0, t_1, \dots, t_{N-1}) \in \mathbb{R}^{N\times d}$ Following the interleaving operation, the resulting sequence $Z \in \mathbb{R}^{(2N)\times d}$ must satisfy,

$$Z_{2i} = p_i, Z_{2i+1} = t_i, i = 0, \dots, N-1. \tag{3.9}$$

This operation performs slot-wise fusion solely along the indexing dimension without requiring additional linear transformations or trainable parameters. In practical implementation, an empty tensor of shape $[2N, d]$ is pre-allocated, where even-indexed positions are assigned values from P, and odd-indexed positions are assigned values from T, thereby efficiently completing the interleaving process.

Once fed into an LLM, this interleaved sequence of length $2N$ serves as the model input, allowing the self-attention mechanism to facilitate fine-grained interactions between numerical and textual features at the lowest embedding level.

To adapt LLMs for table tennis landing point prediction while avoiding full fine-tuning, this study employs Low-Rank Adaptation (LoRA) to efficiently fine-tune the model. LoRA enables the model to specialize in the target task while significantly reducing computational overhead by selectively adjusting key attention projections.

In a standard Transformer self-attention mechanism, the Query (Q), Key (K), and Value (V) matrices are obtained through linear transformations of the input data X,

$Q = XW_Q, K = XW_K, V = XW_V$ where $X \in \mathbb{R}^{B\times T\times d}$ represents the input data (batch size B, time steps T, feature dimension d), and $W_Q, W_K, W_V \in \mathbb{R}^{d\times d}$ are the full-rank parameter matrices used in standard self-attention, with a computational complexity of $O(d^2)$. LoRA introduces low-rank decomposition in the Query and Key projection layers (Q_{proj}, K_{proj}), effectively reducing computation costs while preserving the expressiveness of the Transformer structure. Instead of updating the full-rank matrices, LoRA decomposes the weight matrices into base weights and low-rank incremental components, formulated as:

$W_Q = W_Q + \Delta W_Q, W_K = W_K + \Delta W_K$, where the low-rank updates ΔW_Q and ΔW_K are parameterized as: $\Delta W_Q = A_Q B_Q, \Delta W_K = A_K B_K$ with A and B being low-rank matrices:

$A_Q \in \mathbb{R}^{d\times r}, B_Q \in \mathbb{R}^{r\times d}$ and $A_K \in \mathbb{R}^{d\times r}, B_K \in \mathbb{R}^{r\times d}$ By employing low-rank structure, LoRA reduces the computational complexity from $O(d^2)$ to $O(dr)$, where $r \ll d$, significantly improving efficiency while maintaining model representation capability.

In the context of LLM adaptation for table tennis analysis, LoRA is applied to Query Projection (Q_{proj}) and Key Projection (K_{proj}), enabling efficient fine-tuning without modifying the overall model structure. Instead of updating full LLM weights, LoRA factorizes the parameter updates into low-rank matrices, significantly reducing computational demands while allowing efficient domain adaptation for table tennis trajectory forecasting.

By keeping most LLM parameters frozen and only adjusting key attention components, LoRA ensures training stability and prevents large-scale parameter shifts that may cause overfitting to specific player data, which is particularly important when modeling high-speed movements in table tennis matches. Unlike full fine-tuning, LoRA modifies only the attention heads, allowing seamless adaptation across different LLM architectures while preserving robustness across diverse table tennis datasets (e.g., professional matches vs. amateur training data). The updated LoRA-based Query and Key projections are:

$$\begin{aligned} Q_{\text{LoRA}} &= X(W_Q + \frac{\alpha}{r} A_Q B_Q) \\ K_{\text{LoRA}} &= X(W_K + \frac{\alpha}{r} A_K B_K) \end{aligned} \tag{3.10}$$

where $r = 8$ is the rank of the low-rank matrices, controlling the degree of LoRA-based adaptation, and $\alpha = 32$ is a scaling factor that regulates LoRA's impact. LoRA only modifies the Query (Q_{proj}) and Key (K_{proj}) projections, leaving the Value (V_{proj}) projection unchanged.

By integrating LoRA fine-tuning, the LLM efficiently learns and adapts to ball trajectory patterns and tactical strategies in table tennis matches. LoRA facilitates effective learning of diverse shot styles, spin types, and player-specific tactics, enhancing the model's ability to

predict ball landing positions and strategic shot placement. For instance, in table tennis serve prediction, LoRA enables the model to learn serve trajectory variations across different players, improving the ability to forecast ball placement under different game conditions.

This approach not only optimizes LLM adaptation for table tennis analytics but also enhances its generalization capabilities across different match conditions, allowing the model to efficiently capture individual player strategies and provide more precise tactical insights for competitive play.

Table tennis datasets contain complex motion dynamics, incorporating diverse features such as spin types (Topspin, Backspin), stroke types (Forehand, Backhand), and spatial coordinates (Ball Position X, Y). By fine-tuning Q_{proj} and K_{proj} via LoRA, the model learns critical relationships between shot characteristics and match outcomes, enabling more accurate predictions of successful shot placements and scoring probabilities.

To convert high-dimensional embeddings into numerical time-series forecasts, corresponding to single-step or multi-step prediction with multi-channel outputs, the final time-series semantic vectors extracted from the LLM's output layer are first flattened and then processed through a linear projection layer, generating the final numerical predictions.

In time-series modeling, the input data X is structured as $X \in \mathbb{R}^{B\times\cdots\times n_f}$, where B represents the batch size, nfn_fnf is the size of the last feature dimension, and n_f, $\cdots$ denotes potential intermediate dimensions. Through the FlattenHead mechanism, the data transformation process is defined as follows,

$$\hat{Y} = \text{Dropout}(\text{Linear}(\text{Flatten}_2(X))). \tag{3.11}$$

where $Flatten_2$ represents an operation that unfolds the penultimate dimension and concatenates it with the last feature dimension n_f, effectively restructuring the tensor. The Linear layer performs a mapping operation that projects the data from $\mathbb{R}^{n_f}$ into $\mathbb{R}^{\text{target\_window}}$, ensuring that the output conforms to the expected prediction window. Additionally, a Dropout mechanism is applied to the output, randomly deactivating a fraction of neurons to mitigate overfitting and enhance the model's generalization capabilities.

This projection ensures that the flattened input is transformed into a structured representation of shape $B \times \text{target\_window}$, making it suitable for regression-based forecasting. Consequently, the model effectively learns target window dependencies, enabling robust time-series forecasting by leveraging LLM-driven embeddings for structured numerical prediction.

3.3 Dataset

The dataset used in this study comprises multi-dimensional time-series information, capturing key aspects of temporal attributes, technical action classification, winning probability, and spatial positioning. This dataset is designed to facilitate the analysis of player shot patterns and their influencing factors in table tennis matches, providing a quantitative foundation for optimizing match strategies.

The dataset consists of six primary features, as detailed in Table 3.2.1. These features collectively enable a comprehensive understanding of ball landing positions, shot types, and point-winning dynamics, contributing to the development of predictive models for strategic decision-making in competitive play.

Table 3.1 Table tennis dataset feature description

Feature	Description	Data Type	Value Range
Time	Records the match timestamp for time-series analysis.	Datetime	YYYY-MM-DD
Topspin/Backspin	Indicates the type of ball spin, where 1 represents Topspin and 0 represents Backspin.	Binary	{0,1}
Forehand/Backhand	Indicates the type of stroke, where 1 represents Forehand and 0 represents Backhand.	Binary	{0,1}
Winning in First Three	Binary indicator of whether the player won the point within the first three	Binary	{0,1}

Strokes	strokes.		
Ball Position X	Records the ball's horizontal landing position on the table (unit: cm).	Integer	[0, 152]
Ball Position Y	Records the ball's vertical landing position on the table (unit: cm).	Integer	[0, 140]

To ensure robust model training and evaluation, the dataset is partitioned into training (70%), validation (10%), and test (20%) sets. The training set is used for model learning, the validation set is employed for hyperparameter tuning and generalization assessment, and the test set is reserved for final performance evaluation. This partitioning strategy ensures that the model learns under a well-balanced data distribution while maintaining an independent evaluation set for assessing its ability to generalize to unseen samples.

.

Please act as an expert in data analysis and sports science to help me analyze a dataset related to table tennis matches.

This dataset contains multiple dimensions of information, including temporal attributes, technical action classification, winning probability, and spatial positioning of the ball. Based on the detailed background knowledge provided below, please understand and analyze the characteristics of this data.

1. Technical Background of Table Tennis

Table tennis is a fast-paced competitive sport, where each rally typically involves serves, returns, and continuous attacks. Below are key tactical concepts:

1) First Three Strokes Strategy:
The first three strokes in a rally (serve, return, and third-ball attack) are critical in determining the winner of a point. Scoring within the first three strokes indicates strong tactical execution.

2) Spin Control:
Topspin shots are mainly used for attacking, Backspin shots are mainly used for defense and control. The variation in spin influences match tempo and the nature of exchanges.

3) Shot Placement Strategy:
Analyzing the X and Y coordinates of the ball's landing position helps identify whether a player prefers straight shots (small X variation) or cross-court shots (large X variation). It also helps determine if a player targets the opponent's baseline (large Y values) or prefers short placements (small Y values).

2. Your Task
Based on the above background knowledge, please analyze this dataset, identify patterns in the data, and predict the next landing position of the ball.

Fig 3.2 Human knowledge prompt in ChatPPG

3.4 Evaluations

In this study, we employ Mean Squared Error (MSE) and Mean Absolute Error (MAE) as the primary evaluation metrics to assess the accuracy of our model in predicting the spatial position (x, y coordinates) of the table tennis ball landing points. These error metrics quantify the deviation between the predicted and actual values, providing insight into the model's precision and robustness.

MSE measures the mean squared difference between the predicted and actual coordinates, emphasizing larger errors due to the squaring operation. It is formally defined as:

$$MSE = \frac{1}{N}\sum_{i=1}^{N}[(\hat{x_i} - x_i)^2 + (\hat{y_i} - y_i)^2] \tag{3.12}$$

where $(\hat{x_i}, \hat{y_i})$ represents the model-predicted coordinates, (x_i, y_i) denotes the ground truth coordinates, and N is the total number of samples. Since MSE incorporates the squared error, it is more sensitive to large deviations, making it particularly suitable for identifying and penalizing significant prediction errors.

In contrast, MAE calculates the average Euclidean distance between the predicted and actual values, providing a more interpretative measure of the model's average prediction deviation. It is defined as:

$$MAE = \frac{1}{N}\sum_{i=1}^{N}\sqrt{(\hat{x_i} - x_i)^2 + (\hat{y_i} - y_i)^2} \tag{3.13}$$

Unlike MSE, MAE applies absolute differences instead of squared values, ensuring that large errors do not disproportionately influence the overall metric. Consequently, MAE is more robust to outliers and is better suited for assessing the model's overall prediction stability.

In this study, we use both MSE and MAE to provide a comprehensive evaluation of the model's performance in table tennis landing point prediction. MSE serves as a strict error

measurement, highlighting cases where the model exhibits large deviations, whereas MAE provides a more intuitive interpretation of average prediction errors. By combining these two metrics, we ensure that the model is assessed from both extreme error sensitivity and general predictive accuracy perspectives, thereby guaranteeing its stability and generalization ability across different match scenarios.

ChatPPG is evaluated across four different large language model (LLM) backbones, each varying in size, architecture, and pre-training methodology. These include GPT-2 (125M parameters), Llama-2-7B (7B parameters), Llama-3.2-1B (1.2B parameters), and DeepSeek-R1-Distill-Qwen-1.5B (1.5B parameters). The selection of these models allows for a comparative analysis of ChatPPG's performance across different model capacities, ranging from lightweight architectures (GPT-2) to more advanced instruction-tuned models (DeepSeek-R1-Distill-Qwen-1.5B). This evaluation framework ensures that the effectiveness of ChatPPG is assessed under varying computational constraints and generalization capabilities, providing insights into the scalability and adaptability of LLM-based forecasting for table tennis landing point prediction.

To optimize the model's ability to predict table tennis ball landing points, we configure a set of hyperparameters that facilitate the alignment of time-series data with LLM-generated text representations using Multi-Head Attention (MHA) for multi-modal fusion. These hyperparameters govern critical aspects of attention mechanisms, sequence modeling, optimization, and training stability, ensuring effective learning from both structured numerical inputs (ball landing positions) and unstructured textual data (LLM-provided tactical insights). The details of these hyperparameters are presented in Table 3.3.4.

Table 3.2 ChatPPG hyperparameter settings

Hyperparameter	Description	Value
d_{model}	Query projection hidden dimension	16
n_{heads}	Number of heads in Multi-Head Attention (used for aligning time-series and LLM text representations)	8
d_{ff}	FlattenHead layer dimension	32
Dropout	Dropout rate to prevent overfitting	0.1

seq_{len}	Length of input sequence (i.e., number of past timesteps observed)	8
$label_{\text{len}}$	Start token length for sequence prediction	2
$pred_{\text{len}}$	Length of the forecasted sequence (future timesteps to predict)	4
Factor	Attention mechanism scaling factor	1
Train epochs	Number of training epochs	10
Batch size	Training batch size	32
Eval batch size	Evaluation batch size	8
Learning rate	Initial learning rate	0.0001
Learning rate adjustment	Learning rate adjustment strategy	Exponential Decay
Patience	Early stopping patience (number of epochs)	10

Our model utilizes $d_{\text{model}} = 16$, which determines the size of the hidden representations in the query projection layer. The Multi-Head Attention Mechanism is employed with 8 heads, where its primary function is to align time-series data with LLM-generated text representations. Instead of just capturing dependencies within numerical sequences, MHA enables effective multi-modal fusion between structured data (previous ball landing positions) and LLM-provided semantic cues. This cross-modal attention mechanism allows the model to extract contextualized landing point predictions by integrating historical shot patterns with language-based game strategies.

The FlattenHead layer, with a dimension of 32, processes the transformed representations from MHA, ensuring that both time-series features and text-based insights are effectively incorporated into the predictive framework. A dropout rate of 0.1 is applied to reduce overfitting and enhance generalization.

For sequence modeling, we define a past sequence length of 8 timesteps seq_{len}, meaning the model observes the previous 8 strokes before making a prediction. Within this, 2 timesteps $label_{\text{len}}$ act as the start tokens, guiding the model's learning process, while 4 timesteps

$pred_{\text{len}}$ are forecasted as future landing points. The attention mechanism scaling factor is set to 1, ensuring a stable weight distribution across inputs.

In the training process, we utilize a batch size of 32 for training and an evaluation batch size of 8. The model is trained for 10 epochs, with an initial learning rate of 0.0001, which is dynamically adjusted using an exponential decay strategy lr_{adj}. This strategy ensures that the learning rate adapts as training progresses, balancing optimization efficiency and model stability while preventing premature convergence or oscillations. we define the learning rate schedule as follows:

$$\eta_t = \begin{cases} \eta_0, & t < 3 \\ \eta_0 \cdot 0.9^{(\frac{t-3}{1})}, & t \geq 3 \end{cases} \tag{3.14}$$

where $\eta_t$represents the learning rate at epoch t, η_0 is the initial learning rate 0.0001. During the first three epochs, the model maintains a fixed learning rate $\eta_0$to allow stable convergence. After epoch 3, the learning rate decays exponentially at a rate of 0.9 per epoch $\eta_t = \eta_0 \cdot 0.9^{(t-3)}$, This results in a gradual reduction of the learning rate, ensuring that early training phases prioritize exploration, while later phases focus on fine-tuning and convergence.

To further enhance training stability and prevent overfitting, we employ an early stopping mechanism with a patience of 10 epochs. This means that if no improvement is observed in the validation performance for 10 consecutive epochs, training is automatically halted, preventing unnecessary computation and avoiding overfitting to the training data. This combination of exponential decay learning rate adjustment and early stopping ensures that our model achieves efficient learning, robust generalization, and stable convergence for table tennis landing point prediction.

This multi-modal learning setup enhances the model's predictive power by integrating structured time-series data with LLM-generated textual information, enabling the system to capture both spatial and strategic insights for table tennis landing point prediction.

All experiments were conducted on a single workstation equipped with 10 NVIDIA RTX 3080 GPUs, leveraging DeepSpeed and Hugging Face Accelerate for efficient multi-GPU distributed training. DeepSpeed optimizations enable memory-efficient LoRA fine-tuning,

while Accelerate seamlessly distributes computations across GPUs, ensuring scalability without significant latency overhead.

This setup allows us to train LLM-based models at scale, while ensuring efficient gradient synchronization and parallelization across GPUs for optimized throughput.

Chapter 4 Results

This chapter presents the experimental findings, evaluating the performance of ChatPPG across different LLMs, analyzing prediction accuracy, inference efficiency, and architectural ablations, and highlighting the trade-offs between model complexity and real-time applicability.

4.1 Analysis of LLM Adaptation

In this thesis, evaluates the performance of different LLMs in predicting table tennis ball landing points using two different approaches: alignment-based approaches (ChatPPG, Time-LLM, TEMPO, Autotimes) and a prompting-based approach (LLM-Time). The models were fine-tuned to adapt different LLM backbones—GPT-2, Llama-2-7B, Llama-3.2-1B, and DeepSeek-R1-Distill-Qwen-1.5B—and their prediction accuracy was assessed using Mean Squared Error (MSE) and Mean Absolute Error (MAE). The results are summarized in Table 4.1 and visualized in the MSE and MAE comparison plots.

Table 4.1 Different LLMs with time series model

Model	GPT-2 MSE/MAE	Llama-2-7b MSE/MAE	Llama-3.2-1B MSE/MAE	DeepSeek-R1-Distill-Qwen-1.5B MSE/MAE
ChatPPG(Ours)	**0.512/0.522**	**0.503/0.514**	**0.475/0.493**	**0.432/0.441**
Time-LLM	0.562/0.577	0.549/0.563	0.524/0.531	0.472/0.485
TEMPO	0.568/0.582	0.558/0.569	0.500/0.512	**0.429/0.438**
Autotimes	0.523/0.551	0.510/0.522	0.475/0.493	0.444/0.472
LLM-Time	0.708/0.715	0.682/0.706	0.644/0.671	0.571/0.584

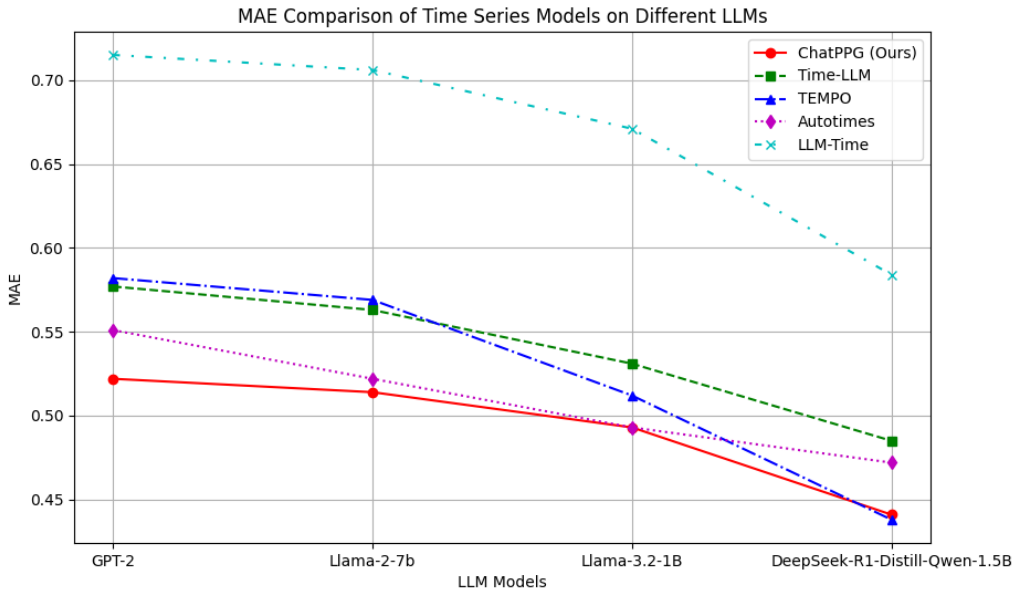

Fig 4.1 MAE comparison of time series models on different LLMs

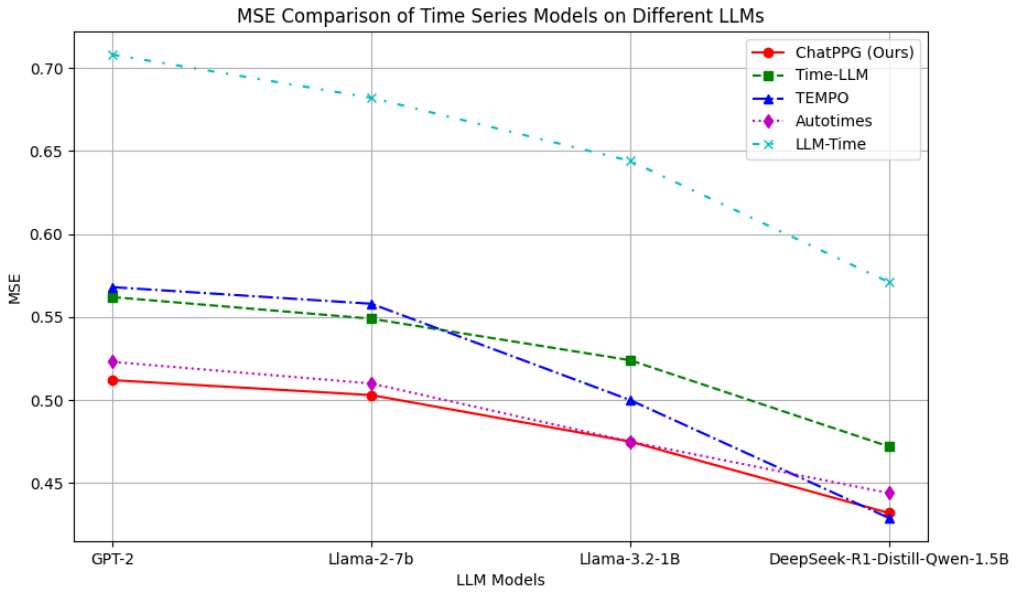

Fig 4.2 MSE comparison of time series models on different LLMs

From Table 4.1, it is evident that ChatPPG consistently achieves the lowest MSE and MAE across all LLMs, demonstrating its effectiveness in capturing spatiotemporal dependencies for table tennis landing point prediction. The best performance is observed when ChatPPG is adapted with DeepSeek-R1-Distill-Qwen-1.5B (MSE = 0.432, MAE = 0.441), which is highlighted in blue, reinforcing the importance of aligning structured time-series data with

LLM-generated text representations to enhance predictive accuracy. Across all evaluated models, alignment-based approaches, including ChatPPG, Time-LLM, TEMPO, and Autotimes, consistently outperform the prompting-based approach (LLM-Time), confirming that direct integration of numerical embeddings into LLMs is more effective than relying on textual prompting alone. The LLM-Time model exhibits significantly higher errors (MSE > 0.64, MAE > 0.67), suggesting that prompting alone is insufficient for fine-grained time-series forecasting, as LLMs struggle to infer structured temporal relationships purely from textual input. Among the alignment-based models, TEMPO, when paired with DeepSeek-R1-Distill-Qwen-1.5B, achieves a competitive performance (MSE = 0.429, MAE = 0.438), slightly surpassing ChatPPG in MSE but not in MAE, indicating that different LLM architectures may specialize in minimizing different aspects of forecasting error. Additionally, Llama-3.2-1B demonstrates superior predictive accuracy compared to GPT-2 and Llama-2-7B, suggesting that newer LLM architectures, particularly those optimized for instruction tuning, contribute positively to time-series learning by improving the model's ability to contextualize structured numerical data within the LLM representation space.

DeepSeek-R1-Distill-Qwen-1.5B consistently outperforms other LLMs across all models, suggesting that larger, instruction-tuned LLMs exhibit superior generalization capabilities in time-series forecasting tasks. Furthermore, alignment-based approaches consistently outperform the prompting-based approach (LLM-Time), confirming that directly integrating time-series representations into LLM embeddings via attention mechanisms enhances predictive accuracy. A strong correlation is observed between MSE and MAE trends, where models that achieve lower MSE also tend to exhibit lower MAE, indicating that minimizing squared errors effectively reduces absolute deviations as well. Among the alignment-based models, Autotimes and Time-LLM show comparable performance across all LLMs, yet they consistently lag behind ChatPPG and TEMPO, suggesting that the additional temporal structure modeling incorporated in ChatPPG contributes to improved accuracy. In contrast, GPT-2 struggles the most across all models, highlighting its inherent limitations in handling structured numerical data for complex time-series forecasting tasks, particularly when compared to more recent, instruction-tuned LLM architectures.

The results demonstrate that alignment-based approaches are superior for time-series prediction in sports analytics, as mapping numerical data into LLM embeddings via structured attention mechanisms leads to better generalization than purely prompt-based methods. Among the evaluated models, larger, fine-tuned LLMs exhibit the most significant improvements, with DeepSeek-R1-Distill-Qwen-1.5B consistently achieving the best results, indicating that instruction-tuned architectures enhance predictive performance in structured numerical forecasting tasks. Additionally, ChatPPG shows strong adaptability across different LLM architectures, reinforcing its effectiveness as a method for integrating time-series forecasting with pre-trained LLMs while maintaining generalization across varying model capacities. In contrast, LLM-Time, which relies solely on prompting, struggles to learn temporal dependencies, confirming that structured input representations are essential for effective time-series forecasting and that directly encoding time-series data within the LLM representation space yields significant performance gains. Overall, the findings highlight that aligning time-series data with LLM-generated text embeddings significantly enhances prediction accuracy compared to prompt-based methods. The strong performance of ChatPPG and TEMPO, particularly when paired with larger, fine-tuned LLMs such as DeepSeek-R1-Distill-Qwen-1.5B, underscores the importance of structured representation learning and LLM adaptation in sports analytics, providing a pathway for more accurate and interpretable predictions in table tennis trajectory modeling.

4.2 Inference Performance Comparison

In addition to evaluating accuracy, it is crucial to analyze inference efficiency, as real-time decision-making is an essential requirement in table tennis analytics. Table 4.2 and the inference time comparison plots illustrate the computational efficiency of different models across multiple LLM backbones (GPT-2, Llama-2-7B, Llama-3.2-1B, and DeepSeek-R1-Distill-Qwen-1.5B) when deployed on dual RTX 3080 GPUs.

Table 4.2 Inference time comparison (ms) across different LLMs on dual RTX 3080 GPUs

Model	GPT-2	Llama-2-7b	Llama-3.2-1B	DeepSeek-R1-Distill-

				Qwen-1.5B
ChatPPG(Ours)	**31.2**	**37.4**	**53.8**	**121.6**
Time-LLM	**29.6**	**35.8**	**52.3**	**120.3**
TEMPO	31.2	37.9	54.1	122.5
Autotimes	31.1	37.2	53.9	121.7
LLM-Time	1362	4757	3673	3976

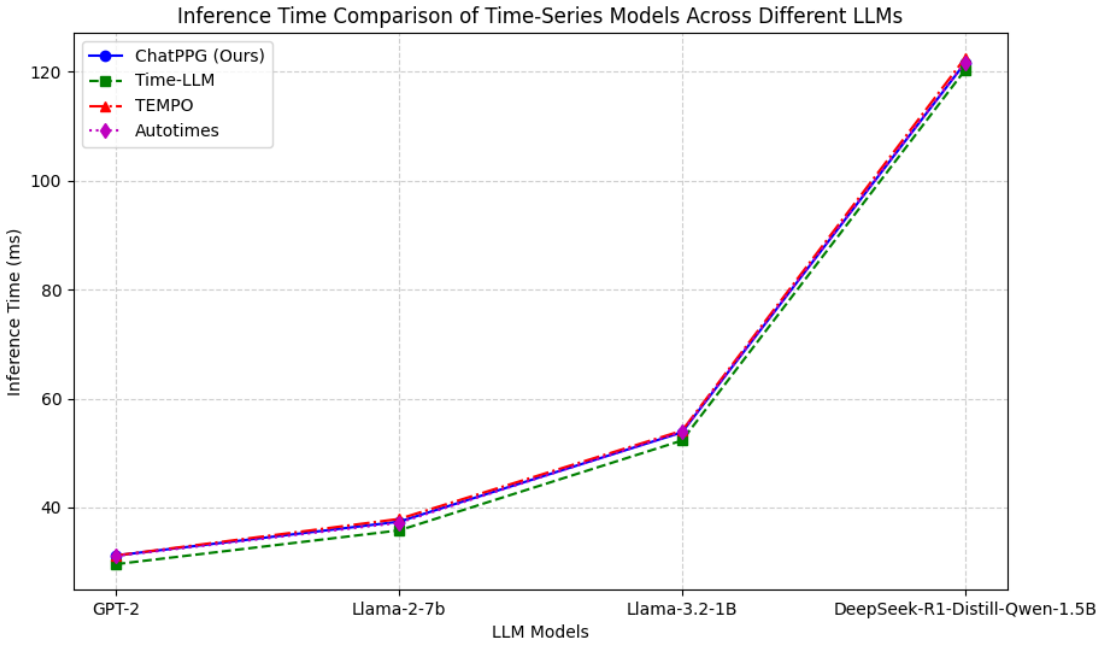

Fig 4.3 Inference time comparison of time series models across different LLMs

The inference time analysis reveals that ChatPPG achieves competitive computational efficiency while maintaining strong prediction accuracy. When evaluated with GPT-2, it achieves an inference time of 31.2 ms, while with Llama-3.2-1B, it reaches 53.8 ms. The DeepSeek-R1-Distill-Qwen-1.5B variant exhibits a longer inference time of 121.6 ms, which is expected given the model's larger architecture and increased computational complexity. Time-LLM emerges as the fastest model, with inference times of 29.6 ms (GPT-2), 35.8 ms (Llama-2-7B), and 52.3 ms (Llama-3.2-1B), outperforming other alignment-based approaches in terms of computational speed. However, its accuracy is consistently lower than that of ChatPPG, indicating a trade-off between computational efficiency and predictive performance.

Both TEMPO and Autotimes demonstrate inference times similar to ChatPPG, suggesting that their attention-based alignment mechanisms introduce comparable computational overhead. In contrast, LLM-Time (a prompt-based approach) exhibits exceptionally high inference times, reaching 1362 ms with GPT-2, 4757 ms with Llama-2-7B, 3673 ms with Llama-3.2-1B, and 3976 ms with DeepSeek-R1-Distill-Qwen-1.5B. This excessive latency is attributed to the inherent inefficiency of LLMs when processing numerical time-series data purely through textual prompting, reinforcing that alignment-based methods offer significantly greater computational efficiency while maintaining superior accuracy.

The results highlight a trade-off between predictive accuracy and inference speed, particularly when evaluating models for real-time table tennis landing point prediction. ChatPPG effectively balances inference speed and predictive accuracy, making it a strong candidate for real-time applications where both performance and computational efficiency are critical. In contrast, Time-LLM achieves the lowest inference latency, demonstrating superior speed across all LLMs; however, this comes at the cost of slightly higher MSE and MAE, suggesting that it is better suited for speed-sensitive applications where minor accuracy trade-offs are acceptable. Both TEMPO and Autotimes exhibit efficiency comparable to ChatPPG while maintaining predictive performance, but neither model significantly outperforms ChatPPG in accuracy or inference time, reinforcing the latter's advantage in balancing both aspects. On the other hand, LLM-Time is highly inefficient, exhibiting exceptionally high inference latency, making it impractical for real-time inference tasks despite leveraging LLM prompting for time-series forecasting. These findings confirm that alignment-based methods provide a more computationally efficient and scalable approach for structured numerical forecasting, whereas prompting alone is insufficient for time-sensitive applications.

The inference results confirm that alignment-based approaches (ChatPPG, Time-LLM, TEMPO, and Autotimes) significantly outperform the prompting-based approach (LLM-Time) in both accuracy and computational efficiency. While Time-LLM is the fastest, ChatPPG provides the best balance between accuracy and inference speed, making it the most viable solution for real-time table tennis landing point prediction.

4.3 Ablation Experiment

To assess the contribution of each component in our proposed LLM-aligned time-series forecasting model, we conduct an ablation study by systematically removing key components and evaluating their impact on performance. This helps isolate the effects of LoRA fine-tuning, domain prompts, channel independence, and embedding structures in the overall prediction architecture. The following configurations are examined:

1) w/o LoRA

Removes LoRA fine-tuning, keeping the pre-trained LLM entirely frozen. This tests the impact of task-specific adaptation on the LLM's ability to process table tennis landing point sequences.

2) w/o Freq Pro

Eliminates the Dominant Fourier Frequency prompts from the Channel Independence Prompts, preventing the model from leveraging frequency-based summarization of the input time-series data.

3) w/o HumKnow

Removes human knowledge prompts, including dataset descriptions and general task information. This tests the necessity of explicit dataset understanding provided to the LLM.

4) w/o IChannel

Disables channel independence, meaning all features are treated as a single fused sequence. Additionally, it removes the Dominant Fourier Frequency prompt, evaluating the contribution of per-channel processing versus global feature fusion.

5) w/o FltProj

Removes the Flatten & Linear Projection layer, relying solely on linear projection of the LLM output embeddings into forecasted values. This assesses whether flattening structured outputs improves sequence forecasting.

6) w/o IEF

Disables interleaved embedding fusion (IEF), meaning prompt embeddings and patch embeddings are grouped separately instead of being interleaved. This tests whether alternating prompts with patches benefits multi-modal representation learning.

By systematically removing these components and comparing performance degradation, this ablation study provides insights into the importance of multi-modal alignment, frequency-based prompts, embedding structures, and LLM fine-tuning in table tennis landing point forecasting.

MSE and MAE values across different LLM backbones (GPT-2, Llama-2-7B, Llama-3.2-1B, DeepSeek-R1-Distill-Qwen-1.5B) are summarized in Table 4.3, while the corresponding MAE and MSE comparison plots provide a visual representation of performance degradation.

Table 4.3 Ablation experiment on ChatPPG

Model	GPT-2 MSE/MAE	Llama-2-7b MSE/MAE	Llama-3.2-1B MSE/MAE	DeepSeek-R1-Distill-Qwen-1.5B MSE/MAE
ChatPPG	**0.512/0.522**	**0.503/0.514**	**0.475/0.493**	**0.432/0.441**
w/o LoRA	0.520/0.541	0.525/0.533	0.491/0.510	0.442/0.457
w/o Freq Pro	0.529/0.541	0.518/0.529	0.482/0.501	0.456/0.468
w/o HumKnow	0.591/0.598	0.581/0.593	0.553/0.569	0.511/0.527
w/o IChannel	0.585/0.594	0.577/0.585	0.541/0.560	0.503/0.512
w/o FltProj	0.548/0.557	0.531/0.547	0.502/0.522	0.462/0.470
w/o IEF	0.610/0.618	0.607/0.613	0.572/0.591	0.527/0.539

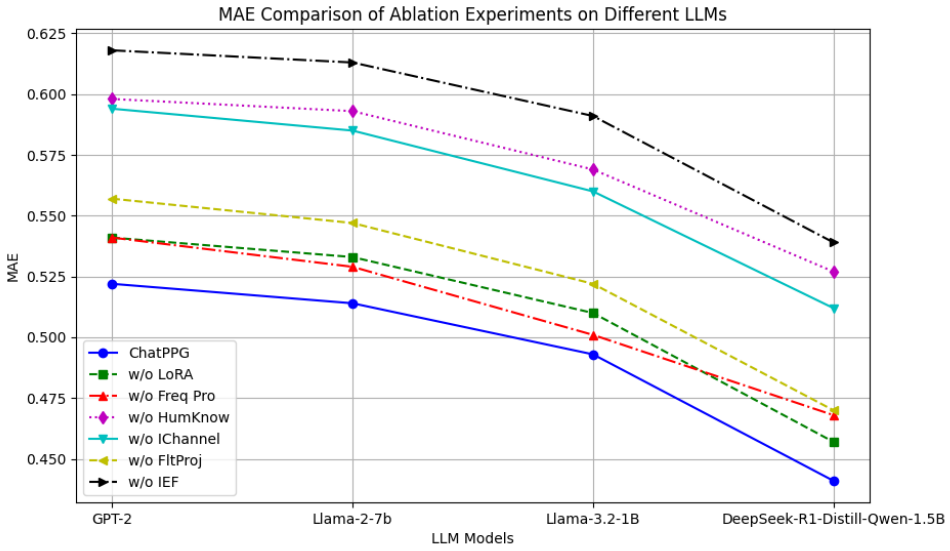

Fig 4.4 MAE comparison of ablation experiments on different LLMs

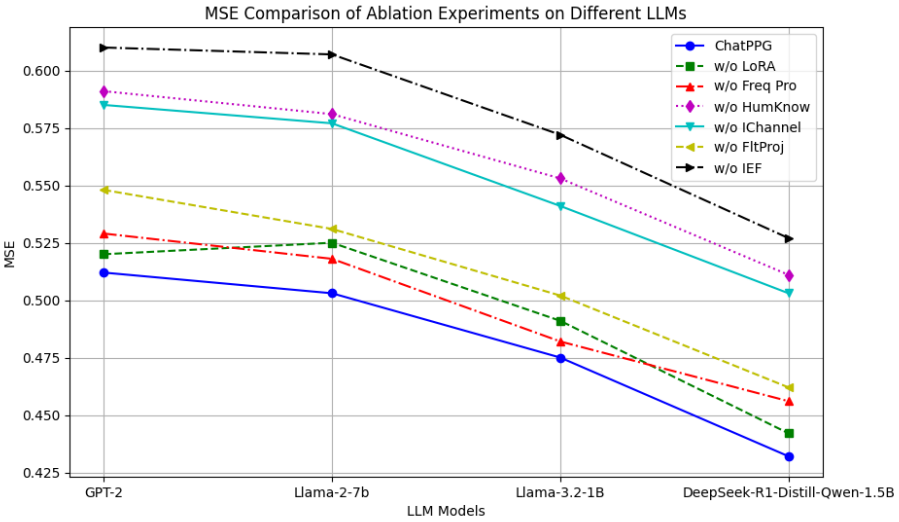

Fig 4.5 MSE comparison of ablation experiments on different LLMs

The ablation study reveals the contributions of individual model components to overall prediction accuracy. Removing LoRA (w/o LoRA) results in a slight increase in MSE and MAE across all LLMs, but the degradation is not as severe as other ablations. This suggests that while LoRA fine-tuning improves accuracy, the core model remains functional without it, albeit with reduced adaptability to domain-specific data. In contrast, removing frequency prompts (w/o Freq Pro) leads to noticeable performance degradation, particularly for DeepSeek-R1-Distill-Qwen-1.5B, where MAE increases from 0.441 to 0.468, indicating that Fourier-based

frequency prompts effectively encode temporal dependencies and enhance forecasting accuracy. The elimination of human knowledge prompts (w/o HumKnow) results in significant accuracy loss, with both MSE and MAE deteriorating across all LLMs, most notably in GPT-2 (MAE: 0.598, MSE: 0.591) and DeepSeek-R1-Distill-Qwen-1.5B (MAE: 0.527, MSE: 0.511). This highlights the critical role of explicit dataset descriptions and task-related prompts in helping LLMs contextualize structured time-series data.

Furthermore, disabling channel independence (w/o IChannel) causes substantial performance degradation, as evidenced by MAE increasing to 0.560 on Llama-3.2-1B and 0.512 on DeepSeek-R1-Distill-Qwen-1.5B. This confirms that channel-wise modeling significantly improves representation learning, while treating all features as a single fused sequence reduces the model's ability to capture independent dependencies between features. The removal of the Flatten Projection layer (w/o FltProj) introduces moderate performance degradation, suggesting that flattening structured outputs before linear projection helps preserve sequence integrity and improves feature mapping.

Among all ablations, disabling interleaved embedding fusion (w/o IEF) results in the most severe performance decline, with MAE increasing to 0.618 for GPT-2 and 0.539 for DeepSeek-R1-Distill-Qwen-1.5B. This strongly indicates that alternating prompt embeddings with patch embeddings enhances multi-modal feature alignment, while treating them separately disrupts this synergy, thereby limiting the model's ability to effectively integrate time-series and textual representations. These findings collectively demonstrate that multi-modal alignment, structured representation learning, and frequency-aware prompting are critical components in optimizing LLM-based time-series forecasting for table tennis landing point prediction.

The ablation study highlights the key architectural components that influence the performance of LLM-based table tennis landing point prediction. Among these, Human Knowledge Prompts (HumKnow) and Interleaved Embedding Fusion (IEF) emerge as the most critical elements, as their removal results in the most significant performance degradation, demonstrating their essential role in enabling LLMs to contextualize structured numerical data effectively. Furthermore, Frequency Prompts (Freq Pro) and Channel Independence (IChannel) contribute substantially to predictive accuracy, indicating that domain-specific statistical

insights and feature-wise separation enhance the model's ability to capture distinct temporal dependencies.

While LoRA fine-tuning provides additional performance gains, its removal does not drastically degrade accuracy, suggesting that pre-trained LLM representations remain highly effective even without extensive task-specific adaptation. In contrast, Flatten Projection (FltProj) plays a beneficial but less critical role, as its removal leads to only moderate performance degradation, indicating that while flattening structured outputs enhances representation learning, it is not an essential component of the model. Notably, multi-modal alignment techniques, particularly IEF, prove to be crucial, reinforcing the effectiveness of cross-modal fusion between time-series patches and textual prompts. These findings collectively emphasize that structured numerical embeddings, domain-informed prompting, and modality-aware integration are key to optimizing LLM-based forecasting models for sports analytics.

It confirms that each architectural component contributes uniquely to the overall prediction performance of LLM-based table tennis landing point forecasting. However, the removal of multi-modal alignment strategies, particularly Interleaved Embedding Fusion (IEF) and Human Knowledge Prompts (HumKnow), results in the most severe performance degradation, underscoring their critical role in integrating structured numerical data with LLM-based text representations. These findings emphasize the necessity of feature-wise modeling, domain-aware prompting, and embedding fusion, demonstrating that effectively aligning LLMs with time-series forecasting tasks requires a structured, multi-modal representation approach. The results further reinforce that combining explicit domain knowledge with learned numerical embeddings enhances predictive accuracy, providing a robust framework for applying LLM-based models in sports analytics and structured time-series modeling.

Chapter 5
Analysis and Discussions

This chapter interprets the experimental results, addressing the research questions, examining the impact of multi-modal alignment and fine-tuning strategies, and discussing practical trade-offs, limitations, and implications for LLM-based time-series forecasting.

Question 1: How can LLMs be effectively adapted for time-series forecasting in table tennis serve prediction?

Our results demonstrate that alignment-based approaches significantly outperform prompting-based methods, confirming that LLMs must be explicitly adapted to structured numerical data rather than relying solely on textual input. ChatPPG (ours), Time-LLM, TEMPO, and Autotimes, which integrate time-series embeddings into LLM representation space via attention mechanisms, achieve lower prediction errors and better generalization compared to the prompt-only LLM-Time model. This validates the hypothesis that LLMs require structured data alignment for effective numerical forecasting.

Additionally, our findings indicate that larger, instruction-tuned LLMs improve predictive accuracy, with DeepSeek-R1-Distill-Qwen-1.5B achieving the best performance across all models. This suggests that LLMs pre-trained with multi-modal or instruction-tuned objectives exhibit stronger generalization capabilities for numerical sequence modeling. However, this performance gain comes at the cost of higher inference latency, emphasizing the trade-off between model complexity and real-time applicability.

The ablation study further reinforces that multi-modal feature alignment plays a crucial role in optimizing LLM-based time-series forecasting. Removing structured feature representations, such as human knowledge prompts and interleaved embedding fusion (IEF), results in substantial accuracy degradation, highlighting the necessity of explicit domain adaptation techniques. Interestingly, LoRA fine-tuning contributes additional accuracy gains, but its absence does not significantly degrade performance, suggesting that task-aware feature encoding is more influential than full-scale LLM fine-tuning.

Question 2: What are the trade-offs between accuracy and computational efficiency when integrating LLMs with time-series forecasting models?

While larger LLMs yield superior predictive accuracy, they introduce increased inference latency, raising concerns about their suitability for real-time applications in sports analytics. Our inference efficiency analysis reveals a clear trade-off between model complexity and deployment feasibility.

ChatPPG achieves a balanced trade-off between accuracy and inference speed, making it a strong candidate for real-time deployment. Time-LLM, while computationally efficient, sacrifices predictive performance, making it more suitable for speed-sensitive applications where minor accuracy trade-offs are acceptable. Conversely, LLM-Time exhibits excessively high inference latency, rendering it impractical for real-time sports analytics, despite being a purely prompt-based method.

These findings suggest that future deployments of LLM-based time-series forecasting models should prioritize both accuracy and computational constraints, ensuring that models are efficiently adapted without compromising real-time performance. Approaches such as parameter-efficient fine-tuning and hybrid embedding techniques provide promising avenues for maintaining predictive performance while mitigating computational costs.

Question 3: Which architectural components contribute most to enhancing LLM-based serve landing prediction, and how does multi-modal alignment impact forecasting performance?

The ablation study provides a detailed assessment of the impact of individual architectural components on predictive accuracy. The removal of human knowledge prompts (HumKnow) and interleaved embedding fusion (IEF) leads to the most severe performance degradation, reinforcing that explicit domain knowledge and structured embedding alignment are essential for LLM-based time-series learning. Frequency-aware prompts (Freq Pro) and channel-independent modeling (IChannel) also significantly influence forecasting accuracy, indicating that domain-specific statistical insights and feature-wise separation enhance temporal pattern recognition.

Additionally, while LoRA fine-tuning improves accuracy, its removal does not drastically degrade performance, suggesting that pre-trained LLM representations retain significant predictive capability even without task-specific adaptation. The Flatten Projection (FltProj) component, while beneficial, has a smaller impact than other modifications, suggesting that structured representation learning contributes more to performance than simple linear projection adjustments.

Collectively, these findings underscore the importance of multi-modal alignment techniques in optimizing LLM-driven time-series forecasting. Models that integrate time-series patches with structured textual representations via cross-attention mechanisms demonstrate superior performance, highlighting the necessity of explicit feature encoding, structured domain adaptation, and hybrid numerical-text integration strategies.

Question 4: Are generic prompting strategies insufficient for numerical sequence modeling?

The study provides strong empirical evidence that LLMs can be successfully adapted for structured time-series forecasting, particularly in sports analytics applications. Unlike traditional deep learning models that require domain-specific architectures, LLMs—when combined with feature-aware prompts, structured embeddings, and attention-based numerical alignment—can achieve competitive performance in forecasting tasks.

The results emphasize that generic prompting strategies alone are insufficient for effective numerical forecasting. LLMs must be explicitly structured to process time-series data, leveraging multi-modal learning techniques, cross-attention mechanisms, and structured prompt engineering to bridge the gap between textual pre-training and numerical sequence modeling. These findings contribute to the growing body of research on foundation model adaptation, demonstrating that multi-modal fusion techniques can unlock new possibilities for leveraging LLMs beyond their traditional NLP applications.

Chapter 6 Conclusion and Future Work

This chapter summarizes the key findings, reinforces the effectiveness of LLM-based time-series forecasting, and outlines future research directions, including end-to-end multi-modal integration and computational optimizations for real-time sports analytics.

6.1 Conclusion

In this thesis, we investigate the feasibility of adapting LLMs for structured time-series forecasting, with a focus on table tennis serve landing point prediction. Through a comprehensive evaluation of alignment-based and prompting-based approaches, the results demonstrate that explicit integration of time-series embeddings into LLMs significantly enhances forecasting accuracy. Among the tested models, ChatPPG consistently outperforms alternative approaches, achieving the lowest MSE and MAE across all LLM architectures, with DeepSeek-R1-Distill-Qwen-1.5B yielding the best performance (MSE = 0.432, MAE = 0.441).

The thesis also highlights a trade-off between accuracy and inference efficiency, where alignment-based models provide a balance between predictive performance and computational feasibility, while prompt-based methods (e.g., LLM-Time) exhibit excessive latency, rendering them impractical for real-time applications. Furthermore, the ablation study confirms that multi-modal feature alignment, interleaved embedding fusion (IEF), and domain-informed prompting are essential for optimizing LLM-based time-series forecasting, as their removal leads to substantial performance degradation.

Overall, this research validates the effectiveness of multi-modal alignment techniques in bridging structured numerical data with LLM representations, reinforcing the potential of leveraging pre-trained foundation models beyond their traditional NLP applications. These findings contribute to the growing field of foundation model adaptation, offering a scalable and computationally efficient framework for integrating LLMs into structured forecasting tasks in sports analytics and beyond.

6.2 Future Work

In this thesis, we utilize the data extracted from vision models rather than implementing a fully end-to-end multi-modal framework. Given the advancements in computational power and multi-modal alignment techniques, future research could explore the development of a fully integrated, end-to-end multi-modal model, enabling a comprehensive AI-driven table tennis analytics system. Such a system would seamlessly combine visual data with LLM-based

reasoning, allowing for simultaneous processing of spatiotemporal patterns, strategic decision-making, and predictive modeling, ultimately enhancing automated match analysis and intelligent coaching applications.

Beyond LLM-based approaches, future research will explore the integration of other multi-modal large models, such as Stable Diffusion, for time-series forecasting and analysis. While traditionally used for image generation and spatial representation learning, models like Stable Diffusion have shown promise in capturing complex, structured patterns across domains. Applying such models to time-series forecasting presents an exciting opportunity to leverage generative modeling techniques for predictive analytics, particularly in scenarios where temporal patterns exhibit strong stochasticity or uncertainty.

By investigating how multi-modal architectures can process and align textual, numerical, and visual data for forecasting tasks, we aim to broaden the scope of foundation models beyond NLP and vision, enabling cross-domain learning and adaptive decision-making in time-series analytics.

References

Ahsan, H., McInerney, D. J., Kim, J., Potter, C., Young, G., Amir, S., & Wallace, B. C. (2024). Retrieving evidence from EHRs with LLMs: Possibilities and challenges. Proceedings of Machine Learning Research, 248, 489–505.

Achanta, R. S., Yan, W. Q., & Kankanhalli, M. S. (2006). Modeling intent for home video repurposing. *IEEE MultiMedia*, *13*(1), 46-55.

Al-Sarayreh, M., Reis, M. M., Yan, W. Q., & Klette, R. (2018). Deep spectral-spatial features of snapshot hyperspectral images for red-meat classification. In *International Conference on Image and Vision Computing New Zealand (IVCNZ)* (pp. 1-6). IEEE.

Al-Sarayreh, M., M. Reis, M., Qi Yan, W., & Klette, R. (2018). Detection of red-meat adulteration by deep spectral–spatial features in hyperspectral images. *Journal of Imaging*, *4*(5), 63.

Al-Sarayreh, M., Reis, M. M., Yan, W. Q., & Klette, R. (2019). A sequential CNN approach for foreign object detection in hyperspectral images. In *International Conference on Computer Analysis of Images and Pattern, Salerno, Italy, September 3–5, 2019, Proceedings, Part I 18* (pp. 271-283). Springer International Publishing.

Al-Sarayreh, M., Reis, M. M., Yan, W. Q., & Klette, R. (2020). Potential of deep learning and snapshot hyperspectral imaging for classification of species in meat. *Food Control*, *117*, 107332.

Alcaraz, J. M. L., & Strodthoff, N. (2022). Diffusion-based time series imputation and forecasting with structured state space models. *arXiv preprint arXiv:2208.09399*.

An, N., & Qi Yan, W. (2021). Multitarget tracking using Siamese neural networks. *ACM Transactions on Multimedia Computing Communications and Applications*, *17*(2s), 1-

16.

Barron, J. L., Fleet, D. J., & Beauchemin, S. S. (1994). Performance of optical flow techniques. International Journal of Computer Vision, 12(1), 43–77.

Bian, J., Li, X., Wang, T., Wang, Q., Huang, J., Liu, C., Zhao, J., Lu, F., Dou, D., & Xiong, H. (2024). P2ANet: A large-scale benchmark for dense action detection from table tennis match broadcasting videos. ACM Trans. Multimedia Comput. Commun. Appl., 20(4), 118:1-118:23.

Cai, W., Jiang, J., Wang, F., Tang, J., Kim, S., & Huang, J. (2024). A survey on mixture of experts. arXiv preprint arXiv:2407.06204.

Calandre, J., Péteri, R., Mascarilla, L., & Tremblais, B. (2021). Extraction and analysis of 3D kinematic parameters of table tennis ball from a single camera. International Conference on Pattern Recognition (ICPR), 9468–9475.

Cao, D., Jia, F., Arik, S. O., Pfister, T., Zheng, Y., Ye, W., & Liu, Y. (2023). Tempo: Prompt-based generative pre-trained transformer for time series forecasting. arXiv preprint arXiv:2310.04948.

Cao, D., Ye, W., Zhang, Y., & Liu, Y. (2024). TimeDiT: General-purpose diffusion transformers for time series foundation model. arXiv preprint arXiv:2409.02322.

Cao, X., and Yan, W. (2022) Pose estimation for swimmers in video surveillance. Multimedia Tools and Applications, Springer.

Cao, Y, Yan, W. (2024) Lips reading using deep learning. Optimization, Machine Learning, and Fuzzy Logic: Theory, Algorithms, and Applications (Chapter 17). IGI Global.

Chang, C., Wang, W. Y., Peng, W. C., & Chen, T. F. (2024). Llm4ts: Aligning pre-trained LLMs as data-efficient time-series forecasters. arXiv preprint arXiv:2308.08469.

Chen, W., Wang, F., & Sun, H. (2021). S2TNet: Spatio-temporal transformer networks for

trajectory prediction in autonomous driving. Asian Conference on Machine Learning, 454–469.

Chen, Z., Yan, W. (2024) Real-time pose recognition for billiard player using deep learning. Deep Learning, Reinforcement Learning and the Rise of Intelligent Systems, pp.188-208, Chapter 10, IGI Global.

Cui, W., & Yan, W. Q. (2016). A scheme for face recognition in complex environments. *International Journal of Digital Crime and Forensics (IJDCF)*, *8*(1), 26-36.

Das, A., Kong, W., Sen, R., & Zhou, Y. (2023). A decoder-only foundation model for time-series forecasting. arXiv preprint arXiv:2310.10688.

Dettmers, T., Pagnoni, A., Holtzman, A., & Zettlemoyer, L. (2023). QLoRA: Efficient finetuning of quantized LLMs. Advances in Neural Information Processing Systems, 36, 10088–10115.

Ding, G., Sener, F., & Yao, A. (2024). Temporal action segmentation: An analysis of modern techniques. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 46(2),* 1011–1030.

Dong, K., & Yan, W. Q. (2024). Player performance analysis in table tennis through human action recognition. Computers, 13(12), 332.

Fan, W., Ding, Y., Ning, L., Wang, S., Li, H., Yin, D., ... & Li, Q. (2024). A survey on rag meeting LLMs: Towards retrieval-augmented large language models. In Proceedings of the ACM SIGKDD Conference on Knowledge Discovery and Data Mining (pp. 6491-6501).

Ferrara, E. (2024). Large language models for wearable sensor-based human activity recognition, health monitoring, and behavioural modelling: A survey of early trends, datasets, and challenges. Sensors, 24(15), Article 15.

Fu, J., Long, Y., Wang, X., & Yin, J. (2024). LLM-driven “coach-athlete” pretraining framework for complex text-to-motion generation. International Joint Conference on Neural Networks (IJCNN), 1–7.

Fu, Y., Nguyen, M., & Yan, W. Q. (2022). Grading methods for fruit freshness based on deep learning. *SN Computer Science*, *3*(4), 264.Gao, S., Koker, T., Queen, O., Hartvigsen, T., Tsiligkaridis, T., & Zitnik, M. (2025). UniTS: A unified multi-task time series model. Advances in Neural Information Processing Systems, 37, 140589-140631.

Gao, X., Nguyen, M., Yan, W. (2021) Face image inpainting based on generative adversarial network. International Conference on Image and Vision Computing New Zealand.

Gao, X., Nguyen, M., Yan, W. (2022) A face image inpainting method based on autoencoder and adversarial generative networks. Pacific-Rim Symposium on Image and Video Technology.

Gao, X., Nguyen, M., Yan, W. (2023) Human face mask detection using YOLOv7+CBAM in deep learning. Handbook of Research on AI and ML for Intelligent Machines and Systems

Gao, X., Nguyen, M., Yan, W. (2023) Enhancement of human face mask detection performance by using ensemble learning models. PSIVT.

Gao, X., Nguyen, M., Yan, W. (2023) A high-accuracy deformable model for human face mask detection. PSIVT.

Gao, X., Nguyen, M., Yan, W. (2024) HFM-YOLO: A novel lightweight and high-speed object detection model. Optimization, Machine Learning, and Fuzzy Logic: Theory, Algorithms, and Applications (Chapter 16). IGI Global.

Gong, S., Li, M., Feng, J., Wu, Z., & Kong, L. (2022). DiffuSeq: Sequence to sequence text generation with diffusion models. *arXiv preprint arXiv:2210.08933*.

Gowdra, N., Sinha, R., MacDonell, S., & Yan, W. (2021). Maximum Categorical Cross Entropy

(MCCE): A noise-robust alternative loss function to mitigate racial bias in Convolutional Neural Networks (CNNs) by reducing overfitting. Pattern Classification.

Gruver, N., Finzi, M., Qiu, S., & Wilson, A. G. (2024). Large language models are zero-shot time series forecasters. Advances in Neural Information Processing Systems, 36.

Gu, D., Nguyen, M., Yan, W. (2016) Cross models for twin recognition. International Journal of Digital Crime and Forensics 8 (4), 26-36.

Guo, D., Yang, D., Zhang, H., Song, J., Zhang, R., Xu, R., ... & He, Y. (2025). DeepSeek-R1: Incentivizing reasoning capability in LLMs via reinforcement learning. arXiv preprint arXiv:2501.12948.

Gupta, D., Bhatti, A., Parmar, S., Dan, C., Liu, Y., Shen, B., & Lee, S. (2024, November). Low-rank adaptation of time series foundational models for out-of-domain modality forecasting. In *International Conference on Multimodal Interaction* (pp. 382-386).

Hegde, N., Vardhan, M., Nathani, D., Rosenzweig, E., Speed, C., Karthikesalingam, A., & Seneviratne, M. (2024). Infusing behavior science into large language models for activity coaching. PLOS Digital Health, 3(4), e0000431.

Held, J., Itani, H., Cioppa, A., Giancola, S., Ghanem, B., & Van Droogenbroeck, M. (2024). X-vars: Introducing explainability in football refereeing with multi-modal large language models. In IEEE/CVF Conference on Computer Vision and Pattern Recognition (pp. 3267-3279).

Herrera, A., Beck, A., Bell, D., Miller, P., Wu, Q., & Yan, W. (2008). Behavior analysis and prediction in image sequences using rough sets. In International Machine Vision and Image Processing Conference (pp. 71-76). IEEE.

Hu, S., Shen, L., Zhang, Y., Chen, Y., & Tao, D. (2024). On transforming reinforcement learning with transformers: The development trajectory. IEEE Transactions on Pattern

Analysis and Machine Intelligence, 46(12), 8580–8599.

Huan, Y., Yan, W. (2025) Semaphore recognition using deep learning. Electronics 14 (2), 286

Hung, C. H. (2018). A study of automatic and real-time table tennis fault serve detection system. Sports, 6(4), 158.

Ji, H., Liu, Z., Yan, W. Q., & Klette, R. (2019). Early diagnosis of Alzheimer's disease using deep learning. In *International Conference on Control and Computer Vision* (pp. 87-91).

Ji, H., Liu, Z., Yan, W. Q., & Klette, R. (2019). Early diagnosis of Alzheimer's disease based on selective kernel network with spatial attention. In *Asian Conference on Pattern Recognition* (pp. 503-515). Cham: Springer International Publishing.

Jiang, P., Ergu, D., Liu, F., Cai, Y., & Ma, B. (2022). A review of YOLO algorithm developments. Procedia Computer Science, 199, 1066–1073.

Jin, M., Koh, H. Y., Wen, Q., Zambon, D., Alippi, C., Webb, G. I., ... & Pan, S. (2024). A survey on graph neural networks for time series: Forecasting, classification, imputation, and anomaly detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Jin, M., Wang, S., Ma, L., Chu, Z., Zhang, J. Y., Shi, X., ... & Wen, Q. (2023). Time-LLM: Time series forecasting by reprogramming large language models. arXiv preprint arXiv:2310.01728.

Jin, M., Wen, Q., Liang, Y., Zhang, C., Xue, S., Wang, X., ... & Xiong, H. (2023). Large models for time series and spatio-temporal data: A survey and outlook. arXiv preprint arXiv:2310.10196.

Jin, M., Zhang, Y., Chen, W., Zhang, K., Liang, Y., Yang, B., ... & Wen, Q. (2024). Position: What can large language models tell us about time series analysis. In International Conference on Machine Learning.

Kieran, D., & Yan, W. (2010). A framework for an event driven video surveillance system. In *IEEE International Conference on Advanced Video and Signal Based Surveillance* (pp. 97-102). IEEE.

Kim, W., Son, B., & Kim, I. (2021). ViLT: Vision-and-language transformer without convolution or region supervision. In International Conference on Machine Learning (pp. 5583-5594).

Klette, R. (2014). Concise Computer Vision: An Introduction into Theory and Algorithms. Springer.

Laadjel, M., Bouridane, A., Kurugollu, F., Nibouche, O., & Yan, W. (2010). Partial palmprint matching using invariant local minutiae descriptors. *Transactions on Data Hiding and Multimedia Security V*, 1-17.

Le, H., Nguyen, M., Yan, W. Q., & Nguyen, H. (2021). Augmented reality and machine learning incorporation using YOLOv3 and ARKit. *Applied Sciences*, *11*(13), 6006.

Li, F., Zhang, Y., Yan, W., Klette, R. (2016) Adaptive and compressive target tracking based on feature point matching. International Conference on Pattern Recognition, (pp.2734-2739).

Li, J., Li, D., Savarese, S., & Hoi, S. (2023). Bootstrapping language-image pre-training with frozen image encoders and large language models. In International Conference on Machine Learning (pp. 19730-19742).

Li, L., Wang, X., Yan, W. (2024) Enhanced multiscale trademark element detection using the improved DETR. Nature Scientific Reports 14, Article number: 29174.

Li, P., Nguyen, M., & Yan, W. Q. (2018). Rotation correction for license plate recognition. In *International Conference on Control, Automation and Robotics (ICCAR)* (pp. 400-404). IEEE.

Li, R., Nguyen, M., & Yan, W. Q. (2017). Morse codes enter using finger gesture recognition.

In *International Conference on Digital Image Computing: Techniques and Applications (DICTA)* (pp. 1-8). IEEE.

Li, X., Thickstun, J., Gulrajani, I., Liang, P. S., & Hashimoto, T. B. (2022). Diffusion-LM improves controllable text generation. *Advances in Neural Information Processing Systems*, *35*, 4328-4343.

Li, Y., Ming, Y., Zhang, Z., Yan, W., & Wang, K. (2021, May). An adaptive ant colony algorithm for autonomous vehicles global path planning. In *IEEE International Conference on Computer Supported Cooperative Work in Design (CSCWD)* (pp. 1117-1122).

Liang, C., Lu, J., Yan, W. (2022) Human action recognition from digital videos based on deep learning. ACM ICCCV.

Liang, C., Yan, W. (2024) Human action recognition based on YOLOv7. Deep Learning, Reinforcement Learning, and the Rise of Intelligent Systems, pp.126-145, Chapter 6, IGI Global.

Liang, Y., Wen, H., Nie, Y., Jiang, Y., Jin, M., Song, D., ... & Wen, Q. (2024). Foundation models for time series analysis: A tutorial and survey. In ACM SIGKDD Conference on Knowledge Discovery and Data Mining (pp. 6555-6565).

Lim, H., Kim, M., Park, S., & Park, N. (2023). Regular time-series generation using SGM. *arXiv preprint arXiv:2301.08518*.

Lin, L., Shi, D., Han, A., & Gao, J. (2024). SpecSTG: A fast spectral diffusion framework for probabilistic spatio-temporal traffic forecasting. *arXiv preprint arXiv:2401.08119*.

Lin, L., Li, Z., Li, R., Li, X., & Gao, J. (2024). Diffusion models for time-series applications: a survey. *Frontiers of Information Technology & Electronic Engineering*, *25*(1), 19-41.

Lin, H. I., Yu, Z., & Huang, Y. C. (2020). Ball tracking and trajectory prediction for table-tennis

robots. Sensors, 20(2), 333.

Liu, A., Feng, B., Xue, B., Wang, B., Wu, B., Lu, C., ... & Piao, Y. (2024). DeepSeek-V3 technical report. arXiv preprint arXiv:2412.19437.

Liu, C., Hayakawa, Y., & Nakashima, A. (2012). An on-line algorithm for measuring the translational and rotational velocities of a table tennis ball. SICE Journal of Control, Measurement, and System Integration, 5(4), 233-241.

Liu, C., & Yan, W. Q. (2020). Gait recognition using deep learning. In Handbook of Research on Multimedia Cyber Security (pp. 214-226). IGI Global.

Liu, M., Huang, H., Feng, H., Sun, L., Du, B., & Fu, Y. (2023). PriSTI: A conditional diffusion framework for spatiotemporal imputation. In *IEEE International Conference on Data Engineering (ICDE)* (pp. 1927-1939). IEEE.

Liu, W., Li, Y., Tomasetto, F., Yan, W., Tan, Z., Liu, J., & Jiang, J. (2022). Non-destructive measurements of Toona sinensis chlorophyll and nitrogen content under drought stress using near infrared spectroscopy. *Frontiers in Plant Science*, *12*, 809828.

Liu, X., Neuyen, M., & Yan, W. Q. (2020). Vehicle-related scene understanding using deep learning. In *Pattern Recognition: ACPR 2019 Workshops* (pp. 61-73). Springer Singapore.

Liu, Y., Nand, P., Hossain, M. A., Nguyen, M., & Yan, W. Q. (2023). Sign language recognition from digital videos using feature pyramid network with detection transformer. Multimedia Tools and Applications, 82(14), 21673-21685.

Liu, Y., Qin, G., Huang, X., Wang, J., & Long, M. (2024). AutoTimes: Autoregressive time series forecasters via large language models. arXiv preprint arXiv:2402.02370.

Liu, Z., Xie, X., He, M., Zhao, W., Wu, Y., Cheng, L., Zhang, H., & Wu, Y. (2025). Smartboard: Visual exploration of team tactics with LLM agent. IEEE Transactions on Visualization and Computer Graphics, 31(1), 23–33.

Lu, J., Shen, J., Yan, W., Boris, B. (2017) An empirical study for human behaviors analysis. International Journal of Digital Crime and Forensics 9 (3), 11-17.

Lu, J., Nguyen, M., Yan, W. (2018) Pedestrian detection using deep learning. IEEE AVSS.

Lu, J., Nguyen, M., Yan, W. (2020) Human behavior recognition using deep learning. International Conference on Image and Vision Computing New Zealand.

Lu, J., Nguyen, M., Yan, W. (2020) Comparative evaluations of human behavior recognition using deep learning. Handbook of Research on Multimedia Cyber Security, 176-189.

Lu, J., Nguyen, M., Yan, W. (2021) Sign language recognition from digital videos using deep learning methods. International Symposium on Geometry and Vision.

Liu, Z., Yan, W. Q., & Yang, M. L. (2018). Image denoising based on a CNN model. In *International Conference on Control, Automation and Robotics (ICCAR)* (pp. 389-393). IEEE.

Liu, X., & Yan, W. Q. (2021). Traffic-light sign recognition using capsule network. *Multimedia Tools and Applications*, *80*(10), 15161-15171.

Liu, X., Yan, W. Q., & Kasabov, N. (2020). Vehicle-related scene segmentation using CapsNets. In *International Conference on Image and Vision Computing New Zealand (IVCNZ)* (pp. 1-6).

Luo, Z., Nguyen, M., Yan, W. (2022) Kayak and sailboat detection based on the improved YOLO with Transformer. ACM ICCCV.

Luo, Z., Nguyen, M., Yan, W. (2021) Sailboat detection based on automated search attention mechanism and deep learning models. International Conference on Image and Vision Computing New Zealand.

Ma, J., Liu, W., Miller, P., & Yan, W. (2009). Event composition with imperfect information for bus surveillance. In *IEEE International Conference on Advanced Video and Signal*

Based Surveillance (pp. 382-387). IEEE.

Martin, P.-E., Benois-Pineau, J., Péteri, R., & Morlier, J. (2021). Three-stream 3D/1D CNN for fine-grained action classification and segmentation in table tennis. International Workshop on Multimedia Content Analysis in Sports, 35–41.

Mehtab, S., & Yan, W. Q. (2022). Flexible neural network for fast and accurate road scene perception. *Multimedia Tools and Applications*, *81*(5), 7169-7181.

Ming, Y., Li, Y., Zhang, Z., & Yan, W. (2021). A survey of path planning algorithms for autonomous vehicles. *SAE International Journal of Commercial Vehicles*, *14*(02-14-01-0007), 97-109.

Nasution, U., Nasution, M. A. H., Habibi, M. I., Tahira, W. L. A., & Ridoh, M. (2024). Analysis of the development of regulations and policies in the world of table tennis: A literature study approach. Journal Coaching Education Sports, 5(1), 25–32.

Nie, Y., Nguyen, N. H., Sinthong, P., & Kalagnanam, J. (2022). A time series is worth 64 words: Long-term forecasting with transformers. arXiv preprint arXiv:2211.14730.

Pan, C., Li, X., & Yan, W. Q. (2018). A learning-based positive feedback approach in salient object detection. In International Conference on Image and Vision Computing New Zealand (IVCNZ) (pp. 1-6).

Pan, C., & Yan, W. Q. (2020). Object detection based on saturation of visual perception. Multimedia Tools and Applications, 79(27), 19925-19944.

Pan, C., Liu, J., Yan, W. Q., Cao, F., He, W., & Zhou, Y. (2021). Salient object detection based on visual perceptual saturation and two-stream hybrid networks. IEEE Transactions on Image Processing, 30, 4773-4787.

Peng, D., Yan, W. (2025) Test-time training with adaptive memory for traffic accident severity prediction. Computers, MDPI.

Pan, Z., Jiang, Y., Garg, S., Schneider, A., Nevmyvaka, Y., & Song, D. (2024). S2IP-LLM: Semantic space informed prompt learning with LLM for time series forecasting. In International Conference on Machine Learning.

Poliakov, A., Marraud, D., Reithler, L., & Chatain, C. (2010). Physics based 3D ball tracking for tennis videos. International Workshop on Content Based Multimedia Indexing (CBMI), 1–6.

Poolton, J. M., Masters, R. S., & Maxwell, J. P. (2006). The influence of analogy learning on decision-making in table tennis: Evidence from behavioral data. Psychology of Sport and Exercise, 7(6), 677-688.

Phong, C. T., & Yan, W. Q. (2014). An overview of penetration testing. *International Journal of Digital Crime and Forensics (IJDCF)*, *6*(4), 50-74.

Qi, J., Nguyen, M., & Yan, W. Q. (2022). Waste classification from digital images using ConvNeXt. In *Pacific-Rim Symposium on Image and Video Technology* (pp. 1-13). Cham: Springer International Publishing.

Qi, J., Nguyen, M., Yan, W. (2023) CISO: Co-iteration semi-supervised learning for visual object detection. Multimedia Tools and Applications.

Qin, Z., & Yan, W. Q. (2021). Traffic-sign recognition using deep learning. In *International Symposium on Geometry and Vision. Revised Selected Papers 1* (pp. 13-25). Springer International Publishing.

Raab, M., Masters, R. S. W., & Maxwell, J. P. (2005). Improving the 'how' and 'what' decisions of elite table tennis players. Human Movement Science, 24(3), 326–344.

Radford, A., Kim, J. W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., ... & Sutskever, I. (2021). Learning transferable visual models from natural language supervision. In International Conference on Machine Learning (pp. 8748-8763).

Rasul, K., Ashok, et al (2023). Lag-Llama: Towards foundation models for time series

forecasting. arXiv preprint arXiv: 2310.08278.

Rasul, K., Seward, C., Schuster, I., & Vollgraf, R. (2021). Autoregressive denoising diffusion models for multivariate probabilistic time series forecasting. In *International Conference on Machine Learning* (pp. 8857-8868). PMLR.

Reis, M. M., Van Beers, R., Al-Sarayreh, M., Shorten, P., Yan, W. Q., Saeys, W., ... & Craigie, C. (2018). Chemometrics and hyperspectral imaging applied to assessment of chemical, textural and structural characteristics of meat. *Meat Science*, *144*, 100-109.

Ren, Y., Nguyen, M., & Yan, W. Q. (2018). Real-time recognition of series seven New Zealand banknotes. *International Journal of Digital Crime and Forensics (IJDCF)*, *10*(3), 50-65.

Schilling, A., Anurathan, J., Mühlberger, J., Gerschner, F., Rössle, M., Theissler, A., & Klaiber, M. (2024). Querying football matches for event data: Towards using large language models. Proceedings of ISACE (Vol. 14794, p. 216). Springer Nature.

Shen, D., Chen, X., Nguyen, M., & Yan, W. Q. (2018, April). Flame detection using deep learning. In *International Conference on Control, Automation and Robotics (ICCAR)* (pp. 416-420). IEEE.

She, L., Zhang, C., Man, X., & Shao, J. (2024). LLMDiff: Diffusion model using frozen LLM transformers for precipitation nowcasting. Sensors, 24(18), 6049.

Shen, H., Kankanhalli, M., Srinivasan, S., Yan, W. (2004) Mosaic-based view enlargement for moving objects in motion pictures. IEEE ICME'04.

Shen, J., Yan, W., Miller, P., & Zhou, H. (2010). Human localization in a cluttered space using multiple cameras. In IEEE International Conference on Advanced Video and Signal Based Surveillance (pp. 85-90). IEEE.

Shen, L., & Kwok, J. (2023). Non-autoregressive conditional diffusion models for time series prediction. In *International Conference on Machine Learning* (pp. 31016-31029).

Shen, Y., & Yan, W. Q. (2018). Blind spot monitoring using deep learning. In *International Conference on Image and Vision Computing New Zealand (IVCNZ)* (pp. 1-5). IEEE.

Sikder, M. F., Ramachandranpillai, R., & Heintz, F. (2023). Transfusion: generating long, high fidelity time series using diffusion models with transformers. *arXiv preprint arXiv:2307.12667*.

Song, C., He, L., Yan, W., Nand, P. (2019) An improved selective facial extraction model for age estimation. International Conference on Image and Vision Computing New Zealand.

Sun, C., Li, H., Li, Y., & Hong, S. (2023). TEST: Text prototype aligned embedding to activate LLM's ability for time series. arXiv preprint arXiv:2308.08241.

Tan, M., Merrill, M., Gupta, V., Althoff, T., & Hartvigsen, T. (2025). Are language models actually useful for time series forecasting?. Advances in Neural Information Processing Systems, 37, 60162-60191.

Tang, H., Zhang, C., Jin, M., Yu, Q., Wang, Z., Jin, X., ... & Du, M. (2025). Time series forecasting with LLMs: Understanding and enhancing model capabilities. ACM SIGKDD Explorations Newsletter, 26(2), 109-118.

Tang, S., Yan, W. (2024) Utilizing RT-DETR model for fruit calorie estimation from digital images. Information 2024, 15(8), 469.

Tran, T.-D. (2024). TTNet: A novel machine learning model for facial emotion detection in online learning systems. SoftwareX, 27, 101787.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł. ukasz, & Polosukhin, I. (2017). Attention is all you need. Advances in Neural Information Processing Systems, 30.

Wang, H., & Yan, W. Q. (2022). Face detection and recognition from distance based on deep learning. In *Aiding Forensic Investigation Through Deep Learning and Machine*

Learning Frameworks (pp. 1-17). IGI Global.

Wang, J., Kankanhalli, M. S., Yan, W., & Jain, R. (2003). Experiential sampling for video surveillance. In *ACM SIGMM International Workshop on Video surveillance* (pp. 77-86).

Wang, J., Yan, W. Q., Kankanhalli, M. S., Jain, R., & Reinders, M. J. (2003). Adaptive monitoring for video surveillance. In *International Conference on Information, Communications and Signal Processing and Pacific Rim Conference on Multimedia. Proceedings of the 2003 Joint* (Vol. 2, pp. 1139-1143). IEEE.

Wang, J., Bacic, B., & Yan, W. Q. (2018). An effective method for plate number recognition. *Multimedia Tools and Applications*, *77*, 1679-1692.

Wang, L., & Yan, W. Q. (2021). Tree leaves detection based on deep learning. In *International Symposium on Geometry and Vision, Revised Selected Papers 1* (pp. 26-38). Springer International Publishing.

Wang, X., Feng, S., & Yan, W. Q. (2019). Human gait recognition based on self-adaptive hidden Markov model. IEEE/ACM Transactions on Computational Biology and Bioinformatics, 18(3), 963-972.

Wang, X., & Yan, W. Q. (2020). Cross-view gait recognition through ensemble learning. Neural Computing and Applications, 32, 7275-7287.

Wang, X., Yan, W. (2019) Gait recognition using multichannel convolutional neural networks. Neural Computing and Applications.

Wang, X., Yan, W. (2019) Multi-perspective gait recognition based on ensemble learning. Springer Neural Computing and Applications.

Wang, X., Yan, W. (2020) Non-local gait feature extraction and human identification. Springer Multimedia Tools and Applications.

Wang, X., Yan, W. (2022) Human identification based on gait manifold. Applied Intelligence

Wang, Z., Wen, Q., Zhang, C., Sun, L., & Wang, Y. (2024). DiffLoad: Uncertainty quantification in electrical load forecasting with the diffusion model. IEEE Transactions on Power Systems.

Wolff, M. L., Yang, S., Torkkola, K., & Mahoney, M. W. (2025). Using pre-trained LLMs for multivariate time series forecasting. arXiv preprint arXiv:2501.06386.

Woo, G., Liu, C., Kumar, A., Xiong, C., Savarese, S., & Sahoo, D. (2024). Unified training of universal time series forecasting transformers. *arXiv preprint arXiv:2402.02592*.

Wu, H., Hu, T., Liu, Y., Zhou, H., Wang, J., & Long, M. (2022). TimesNet: Temporal 2d-variation modeling for general time series analysis. arXiv preprint arXiv:2210.02186.

Wu, J., Yang, S., Zhan, R., Yuan, Y., Chao, L. S., & Wong, D. F. (2025). A survey on LLM-generated text detection: Necessity, methods, and future directions. Computational Linguistics, 1-66.

Wu, S., Irsoy, O., Lu, S., Dabravolski, V., Dredze, M., Gehrmann, S., ... & Mann, G. (2023). BloombergGPT: A large language model for finance. arXiv preprint arXiv:2303.17564.

Xi, Z., Chen, W., Guo, X., He, W., Ding, Y., Hong, B., ... & Gui, T. (2025). The rise and potential of large language model based agents: A survey. Science China Information Sciences, 68(2), 121101.

Xia, Y., Nguyen, M., Yan, W. (2022) A real-time Kiwifruit detection based on improved YOLOv7. International Conference on Image and Vision Computing New Zealand (IVCNZ)

Xia, Y., Nguyen, M., Yan, W. (2023) Kiwifruit counting using KiwiDetector and KiwiTracker. IntelliSys Conference.

Xia, Y., Nguyen, M., Yan, W. (2023) Multiscale Kiwifruit detection from digital images. PSIVT.

Xia, Y., Nguyen, M., Yan, W. (2024) An improved YOLO algorithm for real-time Kiwifruit detection. Optimization, Machine Learning, and Fuzzy Logic: Theory, Algorithms, and Applications (Chapter 9), IGI Global.

Xiang, Y., Yan, W. (2021) Fast-moving coin recognition using deep learning. Springer Multimedia Tools and Applications.

Xiao, B., Nguyen, M., & Yan, W. Q. (2021). Apple ripeness identification using deep learning. In *International Symposium on Geometry and Vision, Revised Selected Papers 1* (pp. 53-67). Springer International Publishing.

Xiao, B., Nguyen, M., & Yan, W. Q. (2023). Fruit ripeness identification using transformers. *Applied Intelligence*, *53*(19), 22488-22499.

Xiao, B., Nguyen, M., & Yan, W. Q. (2024). Fruit ripeness identification using YOLOv8 model. *Multimedia Tools and Applications*, *83*(9), 28039-28056.

Xiao, G., Lin, J., Seznec, M., Wu, H., Demouth, J., & Han, S. (2023). SmoothQuant: Accurate and efficient post-training quantization for large language models. International Conference on Machine Learning, 38087–38099.

Xie, Z., Li, Z., He, X., Xu, L., Wen, X., Zhang, T., ... & Pei, D. (2024). ChatTS: Aligning time series with LLMs via synthetic data for enhanced understanding and Reasoning. arXiv preprint arXiv:2412.03104.

Xing, J., & Yan, W. Q. (2021). Traffic sign recognition using guided image filtering. In *International Symposium on Geometry and Vision* (pp. 85-99). Cham: Springer International Publishing.

Xu, G., Yan, W. (2023) Facial emotion recognition using ensemble learning. Deep Learning, Reinforcement Learning, and the Rise of Intelligent Systems, pp.146-158, Chapter 7, IGI Global.

Xu, T., Li, Z., Yuan, M., Zheng, Z., Zhang, J., & Kuai, X. (2023). Three-dimensional spatiotemporal reconstruction and feature analysis of table tennis movement enhanced by multi-view computer vision. International Conference on Information Technology and Contemporary Sports (TCS), 60–68.

Xue, H., & Salim, F. D. (2023). Utilizing language models for energy load forecasting. In ACM International Conference on Systems for Energy-Efficient Buildings, Cities, and Transportation (pp. 224-227).

Xue, H., & Salim, F. D. (2023). PromptCast: A new prompt-based learning paradigm for time series forecasting. IEEE Transactions on Knowledge and Data Engineering, 36(11), 6851–6864.

Xue, Y. Yan, W. (2023) YOLO models for fresh fruit classification from digital videos. Handbook of Research on AI and ML for Intelligent Machines and Systems, pp. 421-435, Chapter 17, IGI Global.

Yan, W., Kankanhalli, M. (2002) Detection and removal of lighting & shaking artifacts in home videos. ACM International Conference on Multimedia, 107-116.

Yan, W. Q., & Kankanhalli, M. S. (2002). Erasing video logos based on image inpainting. In *IEEE International Conference on Multimedia and Expo* (Vol. 2, pp. 521-524). IEEE.

Yan, W., Kankanhalli, M., Wang, J., Reinders, M. (2003) Experiential sampling for monitoring. ACM SIGMM Workshop on Experiential Telepresence, 70-72.

Yan, W., Kieran, D. F., Rafatirad, S., & Jain, R. (2011). A comprehensive study of visual event computing. *Multimedia Tools and Applications*, *55*, 443-481.

Yan, W. Q. (2019). Introduction to Intelligent Surveillance: Surveillance Data Capture, Transmission, and Analytics. Springer.

Yan, W. Q. (2023). Computational Methods for Deep Learning: Theory, Algorithms, and

Implementations. Springer Nature.

Yan, T., Zhang, H., Zhou, T., Zhan, Y., & Xia, Y. (2021). ScoreGrad: Multivariate probabilistic time series forecasting with continuous energy-based generative models. *arXiv preprint arXiv:2106.10121*.

Yang, B., Yan, W. (2024) Real-time billiard shot stability detection based on YOLOv8. Deep Learning, Reinforcement Learning, and the Rise of Intelligent Systems, pp.159-172, Chapter 8, IGI Global.

Yang, G. L., Nguyen, M., Yan, W. Q., & Li, X. J. (2025). Foul detection for table tennis serves using deep learning. Electronics, 14(1).

Younas, F., Usman, M., & Yan, W. Q. (2023). A deep ensemble learning method for colorectal polyp classification with optimized network parameters. *Applied Intelligence*, *53*(2), 2410-2433.

Yu, Q., & Guo, H. (2022). Sports medicine image modeling for injury prevention in basketball training. Contrast Media & Molecular Imaging, 2022.

Yu, Z. (2021) Deep learning methods for human action recognition. Master's Thesis, Auckland University of Technology, New Zealand.

Yu, Z., & Yan, W. Q. (2020). Human action recognition using deep learning methods. In International Conference on Image and Vision Computing New Zealand (IVCNZ) (pp. 1-6). IEEE.

Yu, Y., Si, X., Hu, C., & Zhang, J. (2019). A review of recurrent neural networks: LSTM cells and network architectures. Neural Computation, 31(7), 1235-1270.

Zeng, A., Chen, M., Zhang, L., & Xu, Q. (2022). Are transformers effective for time series forecasting?. arXiv preprint arXiv:2205.13504.

Zhao, K., & Yan, W. Q. (2021). Fruit detection from digital images using CenterNet.

In *International Symposium on Geometry and Vision, Revised Selected Papers* (pp. 313-326). Springer International Publishing.

Zhao, K., Nguyen, M., Yan. (2024) Evaluating accuracy and efficiency of fruit image generation using generative AI diffusion models for agricultural robotics. IEEE IVCNZ'24.

Zheng, K., Yan, W. Q., & Nand, P. (2017). Video dynamics detection using deep neural networks. *IEEE Transactions on Emerging Topics in Computational Intelligence*, *2*(3), 224-234.

Zhou, H., Nguyen, M., & Yan, W. Q. (2023). Computational analysis of table tennis matches from real-time videos using deep learning. In Pacific-Rim Symposium on Image and Video Technology (pp. 69-81). Singapore: Springer Nature Singapore.

Zhou, H., Zhang, S., Peng, J., Zhang, S., Li, J., Xiong, H., & Zhang, W. (2021). Informer: Beyond efficient transformer for long sequence time-series forecasting. In AAAI Conference on Artificial Intelligence (Vol. 35, No. 12, pp. 11106-11115).

Zhou, T., Ma, Z., Wen, Q., Wang, X., Sun, L., & Jin, R. (2022). FEDformer: Frequency enhanced decomposed transformer for long-term series forecasting. In International Conference on Machine Learning (pp. 27268-27286).

Zhou, T., Niu, P., Sun, L., & Jin, R. (2023). One fits all: Power general time series analysis by pretrained LM. Advances in Neural Information Processing Systems, 36, 43322-43355.

Zhang, L., Shen, L., Zheng, Y., Piao, S., Li, Z., & Tsung, F. (2024). LeMoLE: LLM-enhanced mixture of linear experts for time series forecasting. arXiv preprint arXiv:2412.00053.

Zhang, X., Chowdhury, R. R., Gupta, R. K., & Shang, J. (2024). Large language models for time series: A survey. arXiv preprint arXiv:2402.01801.

Zhang, Y., Sun, P., Jiang, Y., Yu, D., Weng, F., Yuan, Z., ... & Wang, X. (2022). Bytetrack:

Multi-object tracking by associating every detection box. In European Conference on Computer Vision (pp. 1-21). Cham: Springer Nature Switzerland.

Zhang, Y., Yan, W., & Narayanan, A. (2017). A virtual keyboard implementation based on finger recognition. In *International Conference on Image and Vision Computing New Zealand (IVCNZ)* (pp. 1-6). IEEE.

Zhang, Y.-J. (2023). Camera Calibration. In 3D Computer Vision: Principles, Algorithms and Applications (pp. 37–65). Springer Nature.

Zhu, W., Peng, B., Yan, W. (2024) Dual knowledge distillation on multiview pseudo labels for unsupervised person re-identification. IEEE Transactions on Multimedia, 26 (7359 - 7371).

Zhu, W., Peng, B., Yan, W. (2025) Multi-level structural contrastive subspace clustering network. IEEE Signal Processing Letters.

Zhu, Y., Yan, W. (2022) Image-based storytelling using deep learning. ACM ICCCV.

Zhu, Y., & Yan, W. Q. (2022). Traffic sign recognition based on deep learning. *Multimedia Tools and Applications*, *81*(13), 17779-17791.

Zhu, Y., & Yan, W. Q. (2022). Ski fall detection from digital images using deep learning. In International Conference on Control and Computer Vision (pp. 70-78).