Knowledge-Based Systems 300 (2024) 112219

Contents lists available at ScienceDirect

Knowledge-Based Systems

journal homepage: www.elsevier.com/locate/knosys

LLM-Commentator: Novel fine-tuning strategies of large language models for automatic commentary generation using football event data

Alec Cook, Oktay Karakuş [*]

School of Computer Science and Informatics, Cardiff University, Abacws, Senghennydd Road, Cardiff, CF24 4AG, UK

ARTICLE INFO

Keywords:
Large language models
Natural language processing
Football
Commentary generation
Fine-tuning

ABSTRACT

Real-time commentary on football matches is a challenging task that requires precise and coherent descriptions of events as they unfold. Traditional methods often fall short in providing timely and accurate insights into the game. This study aims to explore the utilisation of innovative Large language model (LLM) techniques to develop an adept language model – dubbed LLM-Commentator – that can generate (near-) real-time commentary on football matches. The goal is to demonstrate that open-source language models, when fine-tuned with domain-specific data on consumer-grade hardware, can accurately depict football events from raw match data. Three distinct training strategies are employed to fine-tune the language models, addressing various challenges encountered in generating real-time football commentary. The study evaluates the efficacy of these models in producing coherent and accurate descriptions of unseen football events. Among the three strategies proposed, the Mixed Immediately Model emerges as particularly efficient in learning and adeptly handling challenging workloads. This suggests a promising future for simultaneous multi-task learning with compact, open-source language models in the context of real-time sports commentary. Additionally, the study highlights the practicality of utilising consumer-grade hardware for fine-tuning language models with specialised knowledge. The findings underscore the importance of customising training approaches and ensuring well-balanced datasets when fine-tuning language models for specific tasks. Moreover, they serve as a practical guide for broader accessibility to large language models and significantly contribute to the application of NLP in sports journalism, enabling more insightful and engaging real-time commentary on football matches.

1. Introduction

Football has emerged as the most popular sport in today's globalised world [1]. It surpasses other sports in terms of viewership, participation, following, and cultural importance. The allure of football lies not just in the moments of individual or team glory that are woven into the fabric of the game, but in the overarching narratives of triumph, ecstasy, sorrow, and even tragedy. These moments and stories are brought to life for those not present by the power of commentary.

Iconic moments in football history are often remembered by how they were commented on. From Martin Tyler's emotional outburst when Manchester City won the 2011/12 Premier League, to Kenneth Wolstenholme's narration of England winning the 1966 World Cup, the commentary is intrinsically linked with the events that are being described.

Commentary and technology have shared a historical connection since the early days of radio, evolving from the commentator's voice as the sole link between fans and distant matches to the seamless integration of visual and audio experiences in television broadcasts [2]. In the contemporary digital era, there is a growing demand for football content, with the European football market expanding by 50% in the last decade [3], and the global market projected to grow by 4% annually over the next five years [4].

The changing audience habits, characterised by a shift towards digital consumption over traditional live TV, present a formidable challenge for clubs, especially those operating in lower tiers with limited resources to undergo a digital transformation [5]. However, the recent triumph of Wrexham AFC underscores the potential benefits for smaller clubs in expanding their audience reach through consistent content creation, resulting in amplified stadium attendance, heightened social media interactions, and increased participation in match commentary [6]. In this context, digital interaction with the fans (via automated commentary or social media posts) offers an efficient and cost-effective means for clubs to engage fans [7], and enhance their viewing experience amidst the evolving landscape of digital sports consumption and computational sports journalism. By leveraging automated commentary technology, small clubs and limited-resource teams can bridge the gap

[*] Corresponding author.
E-mail address: karakuso@cardiff.ac.uk (O. Karakuş).

https://doi.org/10.1016/j.knosys.2024.112219
Received 23 January 2024; Received in revised form 17 May 2024; Accepted 7 July 2024
Available online 22 July 2024

between digital and traditional media platforms, thereby empowering themselves to thrive in an increasingly digital-centric environment while maximising fan engagement and brand exposure.

The auto-generation of football commentary presents several significant challenges, with coherence and accuracy being at the forefront. Ensuring that the generated commentary flows naturally and accurately depicts the events of the match is crucial for maintaining the engagement and trust of the audience. However, achieving coherence and accuracy in automated commentary is inherently challenging due to the complexity of natural language and the nuanced nature of football events. In social chat-bot development research, a comparable issue has been encountered historically, wherein the rapid evolution of individualised traits among diverse users has hindered the progress of research in this area. This ongoing challenge remains a prominent focus in the field of text generation research [8].

Domain specificity poses another formidable challenge in the auto-generation of football commentary. Football matches are rich in domain-specific knowledge, including player names, team tactics, match statistics, and historical context. To generate commentary that resonates with football fans, automated systems must possess a deep understanding of the sport and its intricacies. Incorporating this domain-specific knowledge into the commentary generation process is essential for producing commentary that is relevant, insightful, and engaging. Handling diverse events in real-time adds further complexity to the auto-generation of football commentary. Football matches are dynamic and unpredictable, with a wide range of events occurring throughout, including goals, fouls, substitutions, tactical changes, and momentum shifts. Generating coherent and accurate commentary that captures the significance of each event requires robust algorithms capable of detecting, analysing, and contextualising diverse types of events in real time. Moreover, different events may require different linguistic expressions and levels of detail, further complicating the commentary generation process.

In the last couple of years, large language models (LLMs) have demonstrated remarkable capabilities in understanding and generating natural language, making them a promising solution for addressing the challenges of auto-generating football commentary. Their ability to process vast amounts of text data and learn intricate language patterns enables them to capture the nuances of football events, maintain coherence, and provide accurate insights. Recent studies, such as Radford et al. [9] and Brown et al. [10], showcase the potential of large language models in various natural language understanding and generation tasks, suggesting their applicability in the context of football commentary generation.

Particularly, a large language model (LLM) is an artificial neural network that has been designed to understand, generate, and manipulate human language [11]. LLMs are characterised by their:

1. **Architecture:** Most modern LLMs are built upon the transformer architecture, which significantly enhances language processing capabilities [12].
2. **Scale:** GPT-3 was trained on 175 billion parameters [13].
3. **Transfer learning capability:** LLMs can be fine-tuned to fit a specific task, demonstrating remarkable versatility [14].

The release of ChatGPT in November 2022 marked a significant advancement in LLM sophistication and was followed by the widespread availability of Meta's LLaMA in February 2023 [15]. LLaMA's impact is noteworthy on multiple fronts: firstly, Meta's 13B model outperforms GPT-3 in reasoning tasks with fewer parameters [15]. Secondly, its smaller models demonstrate the ability to efficiently operate on a single consumer-grade GPU, enhancing accessibility to advanced LLMs [16]. Lastly, the open-source nature of LLaMA, freely available for download and experimentation, has produced innovation in the field.

This surge in creative application is evident in the enthusiast community's rapid expansion of the technology's uses. For instance, HuggingFace, a popular machine learning platform, now hosts nearly 16,000 text generation models, of which 4000 are derivatives of LLaMA [17], highlighting the exponential growth in the usage and experimentation of large language models.

When considering the task of generating commentary, LLMs offer a promising solution, yet they are not without their challenges. Obtaining domain-specific data of high quality for fine-tuning LLMs can be difficult, especially given the limited size and diversity of football commentary datasets. Furthermore, identifying the most suitable fine-tuning strategies for LLMs in the context of football commentary generation necessitates extensive experimentation and optimisation. Addressing these issues entails significant demands for data, computational resources, and financial investment.

The potential implications of effectively fine-tuning a large language model (LLM) for generating football commentary using consumer-grade hardware are extensive and multifaceted. Introducing an AI-driven football commentator could substantially augment the accessibility of the sport within the sports industry. Prior efforts have explored fine-tuning LLMs to support multilingual capabilities [18], and supported computer vision machine learning research for example on Norwegian sign language [19]. While the former study primarily focuses on English-language commentary, the outlined methodology holds promise for potential adaptation to minority, lesser-known, or even endangered languages, thereby extending the global reach of the sport.

Smaller football clubs stand to benefit. Consider Chelmsford City FC, playing in the English National League South division. An examination of their official website shows that supporters currently have no way of following live text commentary for their team, a predicament shared by most fans in the depths of the English league system. Enhancing content offerings and match coverage has consistently proven to amplify the growth of smaller clubs [20]. The primary reason why these clubs do not offer live match coverage is mainly attributed to financial constraints, as most clubs at this level typically operate with an annual budget shortfall of nearly £700,000 [21]. An automated football commentator presents a viable, cost-effective solution for these clubs to enhance fan engagement and expand their audience.

The successful fine-tuning of an LLM on consumer-grade hardware demonstrated in the context of football commentary generation, extends beyond benefits for the sports industry. This achievement holds broader implications for AI research and development, particularly in democratising access to LLMs. While this is not the first attempt to fine-tune LLMs without relying on supercomputers or cloud infrastructure, this paper aims to serve as a comprehensive guide for enthusiasts, independent researchers, and smaller organisations interested in creating their models. By showcasing the feasibility of locally training sophisticated LLMs on a single GPU accessible to all consumers, this paper encourages grassroots innovation and emphasises the potential for broader applications beyond football commentary generation. The primary excitement lies in the democratisation of LLM usage and training, which could spark widespread curiosity about the diverse capabilities of locally trained models. The goal is to inspire a wave of innovation and exploration in the field of AI.

This paper presents a novel approach, termed *LLM-Commentator*, in the field of computer science research. Our innovation involves fine-tuning an open-source LLM on standard consumer-grade hardware to develop an AI football commentator. Our primary objective is to create a model capable of providing accurate match event narrations from raw sporting data, thereby enabling clubs to engage with fans worldwide at reduced costs and facilitating comprehensive match analyses irrespective of geographical limitations. Additionally, by leveraging consumer-grade hardware, we aim to not only pioneer a cost-effective AI-driven sports commentary solution but also fully explore the immense potential within the realm of LLMs in this emerging era. Specifically, we introduce three novel fine-tuning models – Layered (LM), Mixed Immediately (MIM), and Mixed Sequentially (MSM) – to fine-tune the Llama 7B LLM using two raw football commentary datasets. Each of these models addresses significant challenges in the literature surrounding fine-tuning LLMs, such as catastrophic forgetting. We evaluate the performance of these models across various scenarios using error metrics such as Precision, Recall, and Rouge Score. Among the proposed strategies, the Mixed Immediately Model demonstrates notable efficiency in learning with a 0.91 F_1 score whilst preventing catastrophic forgetting and effectively managing complex tasks involved in generating automated football commentary data.

The novel contributions of this paper include:

- Introduction of the *LLM-Commentator*, a pioneering approach to AI football commentary through fine-tuning of open-source LLMs on consumer-grade hardware.
- Advancement of previous methodologies by proposing innovative fine-tuning strategies, including LM, MIM, and MSM models, aimed at enhancing the base model's behaviour, capabilities, and learning rate.
- With the application of the suggested fine-tuning methodologies, the *LLM-Commentator* can be made available as an AI football commentary tool on standard consumer-grade hardware.
- Stand-alone testing of each fine-tuning approach to comprehensively understand its impact on the model's performance.
- Comprehensive comparative analysis of the outcomes from the fine-tuning processes, providing insights into the effectiveness of each strategy in crafting an adept football commentary model.

The rest of the paper is organised as follows: The paper begins with a thorough examination of existing literature, with a particular focus on previous attempts to create an AI football commentator and recent advancements in the field, outlined in Section 2. It then proceeds to explore the methodologies utilised for fine-tuning, covering topics such as data preprocessing in Section 3, whilst detailing fine-tuning strategies, encountered challenges, and testing procedures in Section 4. Subsequently, the paper presents the results derived from the training and testing of the proposed fine-tuning models in Section 5. The subsequent discussion in Section 6 analyzes the findings, providing an assessment of the various fine-tuning strategies employed. Finally, Section 7 concludes the paper by summarising its key points.

2. Related works

2.1. AI commentators

The first attempts to create an automated football commentator took place at the Robot World Cup (RoboCup) in 1997. Created to spur innovative research in robotics and AI, the RoboCup held football competitions in three formats: a simulation league, a small-sized robot league, and a medium-sized robot league [22]. Of particular interest to this research was the scientific challenge award introduced in the 1998 RoboCup. At the 1998 RoboCup, three different research groups introduced three automated football commentators [22]. These systems were MIKE [23] and Rocco [24], two data-to-text models, and Byrne, an animated talking head model [25].

MIKE relies on being fed a continuous stream of high-quality data every 100 ms from a server that records the robot participants [23]. This data is interpreted by ’analyser modules’ to detect specific game events, such as passes and shots. These modules process each event as a ‘proposition’, characterised by a tag and an attribute. To illustrate, a pass executed by player 5 is denoted as ’PASS 5’, where ‘PASS’ is the tag, and ‘5’ is the attribute. MIKE's commentary generation is contingent upon an expansive inventory of remarks corresponding to different events. Comments are assigned an importance score which deprecates over time [26]. MIKE then chooses events with the highest scores to output. MIKE was a ground-breaking model for its time but is now extremely outdated in its methods and processes. Its reliance on a game log to provide data to recognise events may not be feasible for human games. Output is a very simple description of basic play (see Fig. 1(a)).

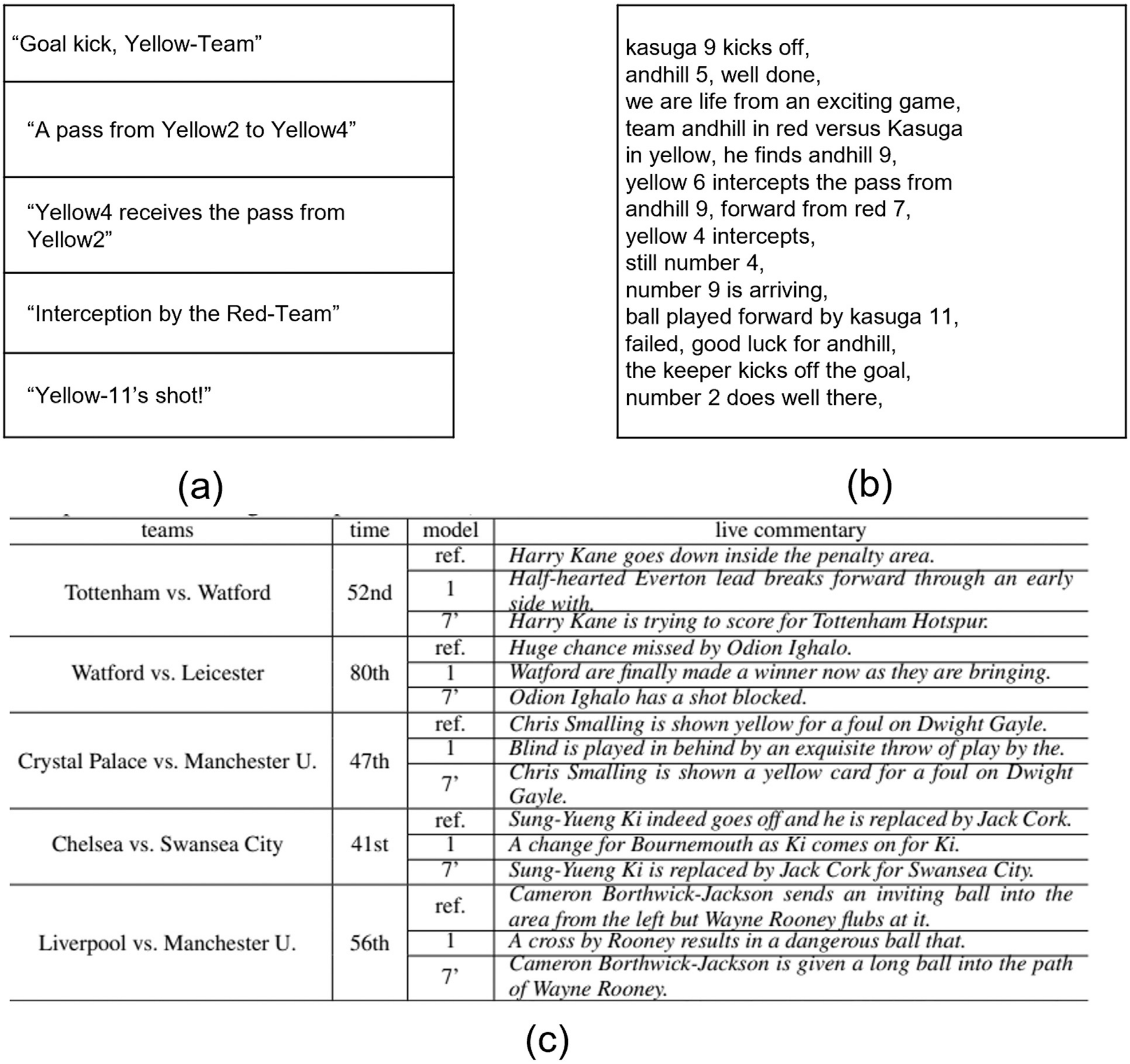
Similar to MIKE, Rocco receives granular game data, like player locations and ball orientation, directly from a server [24]. Rocco's ability to transform this raw data into a 2D geometric representation of the ongoing game set it apart at the time. Movements observed within this geometric model equate to predefined definitions, allowing Rocco to recognise specific game events. The system then categorises these events based on a combination of their importance and the elapsed time since they occurred. Language generation in Rocco is rooted in a template-based system. Depending on the nature and context of the game event, specific templates, comprising strings and variables, are chosen. These templates are populated with the most up-to-date game data, ensuring the commentary is current and relevant. The template selection process weighs the event type, specificity, and desired commentary length. While Rocco was groundbreaking for its era, its template-driven approach to language generation is now considered archaic (see Fig. 1(b)).

Chen and Mooney use a RoboCup engine to simulate matches from which their model receives data on events within the game [27]. Unlike previous systems, their model determines which events to commentate on based on the probability of whether a human commentator would choose to highlight them, a method termed ’strategic generation’. This is the first football commentary model to use machine-learning techniques to train a language model with a dataset of human-produced commentary. The model learns to attach meaning representations to human commentary of events through a combination of KRISPER [28] and WASP [29] learning algorithms. When confronted with a similar event during a match, the model uses these learned associations to generate natural language descriptions. However, a notable limitation is its lack of contextual understanding, which the authors acknowledge results in repetitive and occasionally oversimplified commentary. When compared against human commentators using various evaluation metrics, the model did not surpass human performance [27].

The model proposed by Taniguchi et al. [30] is another important milestone, as it focuses on being able to describe key events during a game, eschewing the play-by-play approach of previous models. The model preprocesses events into 70 primary categories such as ‘pass’ and ’foul,’ and further classifies these into 298 subcategories like ’through ball’ and ’long ball’. Events are assigned an attention score, which the model uses to decide which moments warrant commentary. A specialised gate mechanism further refines the model's understanding of the sequential nature of these events [31]. The model was trained using Chainer, a deep learning framework, by utilising a corpus of 13,662 different pieces of commentary [32]. Templates with placeholders are generated, which the model then populates with contextually appropriate information [30]. The framework the model was built on is now defunct, as the developer has shifted their focus to the PyTorch framework [33]. The model is also prone to generating incorrect or nonsensical descriptions of events (see Fig. 1(c)). To date though, this remains the most impressive attempt to create an automated football commentary language model.

Apart from the models reviewed above to the best of our knowledge, there are no other significant attempts to produce a model capable of generating football commentary based on match events. This gap is particularly notable in the context of recent advancements in LLMs and neural networks. Existing models, once pioneering, now appear obsolete due to their reliance on older technologies and limited processing capabilities, resulting in commentary that often lacks depth, contextual relevance, and accuracy.

This paper aims to bridge the research gap by building a model that leverages advanced NLP frameworks, modern computing power, and refined fine-tuning methods. These advancements enable a model to provide full match coverage with accurate, meaningful, and informative descriptions of each event in a football match in a superior way to these past projects. By doing so, this paper seeks to elevate the standard of AI-generated sports commentary and align it with the recent evolution in AI technology



“Goal kick, Yellow-Team”

“A pass from Yellow2 to Yellow4”

“Yellow4 receives the pass from Yellow2”

“Interception by the Red-Team”

“Yellow-11’s shot!”

(a)

kasuga 9 kicks off,
andhill 5, well done,
we are life from an exciting game,
team andhill in red versus Kasuga
in yellow, he finds andhill 9,
yellow 6 intercepts the pass from
andhill 9, forward from red 7,
yellow 4 intercepts,
still number 4,
number 9 is arriving,
ball played forward by kasuga 11,
failed, good luck for andhill,
the keeper kicks off the goal,
number 2 does well there,

(b)

teams	time	model	live commentary
Tottenham vs. Watford	52nd	ref.	*Harry Kane goes down inside the penalty area.*
		1	*Half-hearted Everton lead breaks forward through an early side with.*
		7’	*Harry Kane is trying to score for Tottenham Hotspur.*
Watford vs. Leicester	80th	ref.	*Huge chance missed by Odion Ighalo.*
		1	*Watford are finally made a winner now as they are bringing.*
		7’	*Odion Ighalo has a shot blocked.*
Crystal Palace vs. Manchester U.	47th	ref.	*Chris Smalling is shown yellow for a foul on Dwight Gayle.*
		1	*Blind is played in behind by an exquisite throw of play by the.*
		7’	*Chris Smalling is shown a yellow card for a foul on Dwight Gayle.*
Chelsea vs. Swansea City	41st	ref.	*Sung-Yueng Ki indeed goes off and he is replaced by Jack Cork.*
		1	*A change for Bournemouth as Ki comes on for Ki.*
		7’	*Sung-Yueng Ki is replaced by Jack Cork for Swansea City.*
Liverpool vs. Manchester U.	56th	ref.	*Cameron Borthwick-Jackson sends an inviting ball into the area from the left but Wayne Rooney flubs at it.*
		1	*A cross by Rooney results in a dangerous ball that.*
		7’	*Cameron Borthwick-Jackson is given a long ball into the path of Wayne Rooney.*

(c)

Fig. 1. AI commentator examples. (a) Examples of MIKE’s descriptions of game events [23]. (b) The Rocco system. Examples of language generation are given [24]. (c) Examples of Taniguchi et al.’s model’s output. Compare human-produced commentary (ref.) with commentary produced by the first (1) and seventh (7’) iteration of the model [30].

2.2. Fine tuning

In 2011 and 2012, breakthroughs in deep learning training emerged with DanNet and AlexNet [34], the latter being a pioneer in fully leveraging the computational power of modern GPUs [35]. Initially impactful in computer vision, these models paved the way for the wider adoption of deep learning in NLP by addressing computationally challenging problems. Their success inspired the development of more sophisticated and efficient language models in NLP. The deep learning revolution was fuelled further with the introduction of the ‘Transformer’ architecture [12]. Transformers revolutionised NLP and other sequence-based tasks by enabling language models to train in a parallel, and therefore more efficient, manner due to a ‘self-attention’ mechanism. This mechanism allows a model to weigh the importance of different parts of the text, which strengthens its understanding of context and the relationships between words [36]. Unlike previous architectures that processed tasks sequentially, transformers can perform multiple computations simultaneously. This parallel nature allows for faster data throughput, meaning more data can be processed in less time. This alleviated bottleneck issues that plagued earlier models and contributed to much quicker training and fine-tuning runs.

Transfer learning, a concept borrowed from cognitive science [37], became a central strategy in the training of language models after the introduction of transformer architecture. This technique involves refining a neural network on a specific task after pre-training on a general task [38]. ULMFiT, released in 2018, was pivotal in demonstrating how pre-trained language models could be fine-tuned for specific tasks with minimal data by effectively applying the principle of transfer learning [39]. In the same vein, BERT [40], released in 2018, and GPT-2 [9], released in 2019, both set new performance benchmarks by utilising these new advances in model architecture, training techniques, and transfer learning. These methods have become the standard for modern large language models.

Current state-of-the-art models now employ sophisticated techniques like few-shot and zero-shot learning, which can be seen as extensions of transfer learning [41]. These techniques enable language models, such as GPT-3, to learn a new task from a small number of examples or none at all [10]. Additionally, meta-learning techniques are being explored, where models ‘learn how to learn’ by teaching themselves better learning algorithms [42].

This paper leverages breakthroughs in research by adopting a transformer architecture as the model’s backbone and utilising specialised libraries and frameworks designed to maximise its potential. The fine-tuning strategies employed involve transfer-learning techniques by refining a pre-trained, generalist base model for the specific task of generating football commentary. Additionally, the paper delves into few-shot and zero-shot learning, pushing the boundaries of AI efficiency by achieving significant outcomes with minimal data, computing resources, and training time. This exploration highlights the feasibility of locally fine-tuning an LLM on domain knowledge, contributing to the democratisation of advanced AI technologies.

2.3. LLMs

The last few years have seen dramatic improvements in the accessibility and variability of LLMs. The memory requirements and resource demands of deploying and running large language models have shrunk rapidly, lighting a spark in NLP experimentation and innovation.

In June 2022, Meta AI introduced OPT (Open Pre-Trained Transformers), a series of models designed to allow the research community to freely experiment with LLMs [43]. This initiative departed from the conventional practice of interacting with models through APIs, providing researchers with hands-on experience with the raw code of an LLM. OPT's release facilitated insights into the challenges of LLM training, fostering a collaborative response to longstanding issues and emphasising the importance of community engagement in addressing problems like loss spikes, hardware failures, and mid-training adjustments [44]. In a similar spirit, BigScience released BLOOM in July 2022, extending public access to LLMs even further [45]. BLOOM not only provided open access to anyone, researcher or not but also offered direct access to the model through their HuggingFace page [45]. Additionally, BLOOM stood out by being trained in 46 different human languages, contributing to the broader accessibility of LLM technology for widespread research and experimentation.

This increasing trend of releasing more open, inclusive, and accessible language models reached a crescendo with the publishing of LLaMA by Meta in February 2023. Comprising four different-sized models, LLaMA was lightweight enough to run on consumer-grade hardware [46], sophisticated enough to go toe-to-toe with GPT-3 [15], and shapeable enough to be fine-tuned quickly and effectively [47]. The release of LLaMA ignited a huge upswing in interest and subsequent experimentation with LLMs with many models being released in a short amount of time that built upon LLaMA's foundations.

Showcasing the potential of the LLaMA models, Alpaca, a fine-tuned version of the LLaMA 7B model, was released a month later and proved itself able to compete with the strongest proprietary models from OpenAI [48]. Alpaca was fine-tuned using 52,000 instruction-following examples that were generated by OpenAI's Text-davinci-003 [48]. The total development process cost around 600 dollars and took a couple of weeks. The success of Alpaca has set a template for the current deluge of fine-tuned, lightweight, and capable language models that are freely available to anyone with an internet connection.

These recent developments have ushered in a new era of accessibility and flexibility in natural language processing. The recent democratisation of LLMs, or the ability to operate these models locally, has been instrumental in this paper. Without it, the feasibility of locally running a sophisticated LLM, let alone training one to generate football commentary, would have been a far more daunting, if not an impractical endeavour. The ability to not only access, but also modify the architecture, weights, and parameters of modern LLMs has provided an unprecedented opportunity to tailor a model that can understand and articulate the flow of football matches.

2.4. Fine-tuning LLMs

Progress in deep-learning techniques and neural network understanding led to the increased sophistication of modern language models. The openness of organisations like BigScience and Meta put these advancements in the hands of the public. However, training or fine-tuning a model still requires prohibitively large computational resources. For instance, a 7B model using float32 precision data types, where 32 bits or 4 bytes are used to store one number, would necessitate a 28 GB GPU for fine-tuning [49]. Whilst most people could now run inference on a language model locally, these hardware requirements put the customisation of modern language models out of reach.

That is until LoRA (Low-rank adaptation), a method stemming from the PEFT (parameter efficient fine-tuning [50]) paradigm, was introduced. LoRA focuses on updating a select subset of a pre-trained model's parameters to achieve the same result as training on all of the model's parameters [51]. In essence, the bulk of the original model remains unmodified and new information is added separately to the core store of knowledge.

This approach offers three primary advantages. Firstly, it reduces the hardware memory requirements by approximately one-third [52]. Secondly, models fine-tuned with LoRA are less susceptible to catastrophic forgetting, a phenomenon where models forget previous information when learning new knowledge [53] since the foundational knowledge embedded in the model remains intact. Finally, whilst the hardware requirements are greatly downgraded, this technique retains the effectiveness of a full fine-tuning of all of a model's weights and parameters [54].

Building upon LoRA, QLoRA (Quantised Low-Rank Adaptation) further minimises the memory demands for fine-tuning LLMs [55]. LLMs use various precision data types for storage. These precision types usually store values in 32 and 16 bits respectively and are what makes training a large language model prohibitively costly in terms of computing power. An important trend in neural networks is quantisation, or fitting more into less, where the memory needs of an LLM is reduced by converting high-precision data types (float32) into low-precision data types (float16) [56]. Quantisation allows less-capable GPUs and CPUs to run the model, and even fine-tune it, in its quantised state [57]. The trade-off has traditionally been that you get a worse model, as some of the model's original performance is lost in the quantisation process [58], and there is a drop in the efficacy of fine-tuning [59].

The QLoRA method allows for both quantisation of a model so that it can fit on a consumer setup and an efficient and effective fine-tuning of a quantised model. It does this by first quantising a model from float32 to a new experimental data type: NormalFloat, or nf4, which is a 4-bit precision data type, allowing it to be stored and run on less capable GPUs [60]. Next, LoRA training is performed by converting a model's parameters back to float 32-bit precision. This is done because to train LoRA adapters in float32, when training is most effective, the model's parameters must also be de-quantised to float32. Essentially, QLoRA de-quantises the 4-bit elements of a model only when they are needed for training, and then re-quantises these elements once they are fine-tuned [61].

The result is a significant decrease in the memory needed to fine-tune an LLM whilst retaining the effectiveness of non-quantised model fine-tuning. The authors of the QLoRA method backed this up by unveiling another camelid-themed language model: Guanaco. Guanaco is a family of LLMs built upon the foundations of LLaMA, the biggest of which is a 65B parameter model that was fine-tuned on a single 48 GB GPU. The same model trained without QLoRA would have required 780 GB of GPU memory [55].

LoRA and QLoRA have effectively dismantled the final barrier that once restricted the customisation and deployment of sophisticated LLMs to those with access to high-end computational resources. By drastically reducing the memory and hardware requirements needed to fine-tune a large language model, these techniques enable the capability to fine-tune an LLM locally, thus serving as the crucial final piece in realising the feasibility of this paper. These methods enable anybody to now easily use, train, and deploy their own fully customised

LLM by just using their local machine. We make full use of these modern techniques by employing QLoRA to conduct the fine-tuning of the three models, demonstrating the practicality and accessibility of advanced NLP models in a more resource-constrained environment.

3. Data

When fine-tuning large language models, the calibre and volume of the dataset are of paramount importance. The efficacy and accuracy of the final model are intrinsically linked to the quality of the data it is trained on [62]. For this paper, the selection of datasets was guided by three essential criteria to ensure their utility in achieving the paper's goals.

First, it must include enough examples of all the events that could take place in a football game, or at least enough events to give a satisfactory level of coverage. Without this, the final model could not hope to be used in real-world applications. Key events include goals, attempts, cards, and fouls, as they make up the majority of a football game's narrative.

Second, the dataset must include the raw details of each event. This includes specifics such as the pitch location, player names, and the nature of each event. Training the model on data with this level of granularity is crucial for it to grasp the intricate context surrounding each event. Understanding these fine details is imperative for the model to accurately interpret and process the dynamics of a football match.

Finally, this granular event data must be coupled with corresponding natural language descriptions of these events. By doing so, the model is trained to draw meaningful connections between the raw data of match events and their narrative descriptions. This enables the model to generate its own accurate and contextually relevant descriptions when confronted with new, previously unseen football events.

Despite extensive research, just two datasets were found that match these criteria:

1. **An English football dataset:** Sourced from the blog of Chris Love, a data scientist, this dataset exclusively covers the top four leagues of the English football system during the 2015/2016 season [63]. It was created by web-scraping BBC football commentary and pairing the commentary with event data from Opta, a large sports analytics company.
2. **A European football dataset:** Located on Kaggle, this dataset contains data from five major European leagues spanning the seasons from 2011/12 to 2016/17 [64]. Similar to the English dataset, it provides a play-by-play account of matches, providing the details of hundreds of thousands of events and their corresponding descriptions.

Events in the datasets are separated into seven discrete categories, with a varied quantity of each presented in Table 1. There are several other categories included in the datasets, such as penalties, handballs, and own goals. However, the number of occurrences for these events is extremely small compared to the categories shown in Table 1, and there is a risk that the models would not be able to effectively learn from the small number of samples. Ultimately, these events are excluded from the training process as a balance needs to be struck between comprehensive event coverage, time to train the models, challenges presented by data sparsity, and added value.

Both datasets required extensive pre-processing. While they captured similar football match events, their representation in the datasets significantly differed. The English dataset provided information in a direct, descriptive format. In contrast, the European dataset used coded values, necessitating its interpretation with an accompanying reference (see Table 2).

Fortunately, the datasets shared more similarities than they had differences. Both broke match events down into discrete categories, both provided a granular examination of these events and crucially, both included a relevant commentary of the event. Furthermore, commentary patterns in both datasets were strikingly consistent. Having a unified linguistic pattern throughout the entirety of the training data makes it much easier for the model to learn to replicate these patterns when generating its commentary. These patterns, encompassing 'what' happened, 'who' was involved, and 'how' it happened, can be seen with this description of a goal:

```
Goal! Morton 1, Falkirk 1. Peter Macdonald (Morton)
header from very close range to the top left corner.
Assisted by Ross Forbes with a cross following a corner.
```

All of these qualities bode well for the fine-tuning. The model is able to learn from repeatable patterns, it has all the individual pieces of information needed to construct its commentary, and it will have a lot of examples.

Table 1
Data set quantitative details for each event.

Event	English dataset	European dataset	Total
Attempts	23,473	204,694	**228,167**
Goals	3,311	24,446	**27,757**
Substitutions	6,496	51,738	**58,234**
Offsides	2,325	43,476	**45,801**
Cards	4,188	41,163	**45,351**
Corners	13,100	91,204	**104,304**
Fouls	25,304	232,925	**258,229**
Total	**78,197**	**689,646**	**767,843**

3.1. Preprocessing

To enhance the learning capabilities of the LLM for generating commentary, the datasets from the two sources had to undergo standardisation to address challenges arising from discrepancies and errors. Decisions were made on retaining essential information and maintaining consistency in event descriptions across the training dataset. The European dataset, due to its larger size, served as the gold standard, shaping the style of the English dataset. To meet the requirements of the QLoRA training, each event in the datasets was converted into a JSONL object and grouped together with other similar events. Each JSONL object comprises a prompt, containing granular event details, and a reference, containing a natural language description of the event. The primary objective during fine-tuning is for the LLM to learn how to synthesise the various pieces of information in the prompt to generate accurate event descriptions.

The process of pre-processing the datasets to construct the prompts is thus: (1) filter each row of the dataset depending on what type of event it describes, (2) save each cell in a row to a variable. For example, the cell containing the player name is saved to a 'Player' variable. Most of the data cleaning takes place in this step, (3) insert these variables into a pre-prepared JSONL object. This moulds the raw information in the dataset into a format more suitable for training. These objects were different for each event, as each event had different categories of information, and (4) save the JSONL object into an event-specific file and continue filtering the rows in the source dataset.

There are several points to mention about this JSONL object; **Structure:** inspired by the QLoRA fine-tuning method this JSONL format was chosen due to its success with the Guanaco chatbot models [65]. **Training style:** The model trains on raw, continuous text. It learns to understand context, cues, and relationships within a flowing textual space. This is different from methodologies such as sentiment analysis where a sentence would have a corresponding sentiment label (positive, negative, neutral) as its distinct output. **Human and Assistant:** A "Human" is designated to provide instructions on what to do and an "Assistant" to provide a concluding description of the event data that has come before, the model learns a flow. **Prefix symbols:** Each piece

Table 2
An attempt event in (left) the English, (right) the European datasets.

English Dataset		European Dataset	
Column name	Example value	Column name	Example value
Half	2	id_odsp	UFot0hit/
Home Team	Crawley Town	id_event	UFot0hit1
Away Team	Peterborough	sort_order	1
Time	76:10:00	time	2
Extra Time	*(blank)*	text	Attempt missed. Mladen Petric (Hamburg) left-footed shot from the left side of the box is high and wide to the left. Assisted by Gokhan Tore.
Mins	76	event_type	1
Secs	10	event_type2	12
Decimal Mins	76.17	side	2
Decimal + Injury	76	event_team	Hamburg SV
Mins + Injury	76	opponent	Borussia Dortmund
Injury Secs	0	player	Mladen Petric
Event	Attempt missed. Gwion Edwards (Crawley Town) left footed shot from a difficult angle and long range on the left misses to the right.	player2 (assister)	Gokhan Tore
Special Event	*(blank)*	player_in	N/A
What	Attempt	player_out	N/A
Event Team	Crawley Town	shot_place	6
Vs Team	Peterborough	shot_outcome	2
Player	Gwion Edwards	is_goal	0
Result	missed	location	9
Pitch Position	a difficult angle and long range on the left	bodypart	2
Goal Placement	to the right	assist_method	1
Goal Following	*(blank)*	situation	1
Assist Method	*(blank)*	fast_break	0
Assist Following	*(blank)*		
Foot	left footed		
Shot Close	No		
Reason	*(blank)*		
Card Type	*(blank)*		
Home Score	2		
Away Score	0		
Home Players on	11		
Away Players on	11		
Match id	33 768 514		
League	League Cup		
Date	11/08/2015		
Pitch Position x/y	102.37/808.99		

of event data has been flagged with a '###' prefix to aid the model in recognising the start and end of different pieces of information.

Post-fine-tuning, the model's ability to interpret and respond to similar prompts will be the key indicator of its learning efficacy. When presented with a new prompt, structured in the same format but containing information about an unseen match event, the model will be expected to fill in the '### Assistant: ' section itself. This is where the model's capability to generate a coherent and contextually accurate description of the event will be demonstrated. Essentially, the model's response in this section serves as its interpretation and description of the new event. This stage of the process also gives the chance to fix any mistakes or inconsistencies in the data. The ultimate goal is to have a dataset with a homogeneous format to give the LLM the best possible prospects for learning.

Sometimes errors in the data were small and easily fixed. Errors in this category included repetition of words in the example pieces of commentary, extra spaces between words that were not needed, and important information in the wrong cells in the dataset. Other times, the errors were embedded deep in the dataset's shortcomings and were difficult to root out. For example, the European dataset did not have separate columns for the home and away team's scores. This posed a significant challenge, especially given that the sample commentaries often included the score in their event descriptions. Without addressing this inconsistency, the model would lack the context for generating accurate score information and might even resort to fabricating scores in its outputs

The solution to this, and other major problems, was to make heavy use of regular expressions. In this case, a regular expression was used to extract the home and away team from the example commentary. These were then included in the prompts in a more beneficial manner for the LLM as separate score variables. The importance of clean training data cannot be overstated. It took many rounds of pre-processing and a few worthless models to get the data into a state where fine-tuning could take place in a healthy, structured, and consistent way. All of the code related to pre-processing the data and creating the training datasets can be found here: https://github.com/Iron-Chef/MascotAI/tree/main/data_processing.

4. Proposed methodology

The primary aim of this paper is to create a language model capable of generating natural language descriptions of football match events that surpass the capabilities of past models. To achieve this, this paper employs a novel approach: training three distinct models, each built upon the same base model, but utilising bespoke fine-tuning methodologies. This strategy is designed to provide a comprehensive understanding of how language models adapt to varied training environments and approaches, thereby identifying the most efficient and effective fine-tuning methodologies.

This paper also adopts a mixed-methods approach, combining both quantitative and qualitative research methods [66]. This dual approach is integral to the objectives, as it allows for a holistic evaluation: quantitatively assessing the performance capabilities of the different models through measurable metrics, and qualitatively evaluating their real-world applicability and effectiveness through human interaction and subjective judgement.

The above-mentioned objectives are combined under the umbrella name - *LLM-Commentator* - which is the final proposed model of this paper. LLM-Commentator consists of the stages given below:

1. First, a base model that is lightweight, capable, and customisable will be selected.
    - **Lightweight**: The entirety of LLM-Commentator's training operations are confined to consumer-grade hardware and the choice of a base model needs to reflect this. An OpenLlama 7B model was chosen for several reasons. Training rounds are quicker, which means results can be seen sooner. A smaller model also allows for parallel tasks to be done on the same computer due to the reduced strain placed on the GPU. Finally, given the industry's shift towards smaller and more efficient LLMs, the focus on a 7B model aimed to provide insights into the capabilities and potentialities of contemporary compact models.
    - **Competent**: The model should already have a high level of base training, empirically validated through academic testing. This maximises the effect of transfer learning during fine-tuning.
    - **Tried and tested**: Only models with a well-documented academic and practical track record were considered. This strategy ensured not only the reliability of our base but also opened up a reservoir of pre-existing research and resources.
2. The next step involves sourcing and preparing training data that accurately represents real-world football match scenarios. The data is meticulously cleaned and processed to ensure uniformity and applicability, setting the stage for effective model training.
3. The model trains on this data, using it to generalise the task of generating descriptions of events in a football. It learns, essentially, to replicate the patterns and styles seen in the training data.
4. The fine-tuning process is iterative, taking place over multiple rounds where the model's generative capabilities are gradually enhanced and its understanding of diverse football events is expanded. Each round involves adjusting the model's parameters and introducing new events for the model to learn how to describe.
5. Throughout this process, rigorous evaluations and testing are conducted, both during and post-fine-tuning to monitor performance improvements and the cumulative impact of the fine-tuning. This helps analyse not only the model's technical performance but also its practical utility in real-world scenarios, adhering to the mixed-methods approach of this study.

The above procedure is repeated three times, with the final goal being the production of three distinct models, all trained to accomplish the same task. The process differs for each model in the duration of fine-tuning, the composition and proportion of training data used, and the underlying theoretical approaches. These differences are designed to explore various aspects of model training and performance, providing a comprehensive understanding of the most effective strategies for generating natural language descriptions of football matches on consumer-grade hardware.

Considering all factors, the base model selected for this paper is OpenLLaMA 7B, a fully open-source version of the original LLaMA model produced by Meta. This model satisfies the criteria for being sufficiently lightweight for consumer-grade hardware, while still maintaining a high level of performance, as verified in academic studies [15]. There is also an extensive body of research and analysis available on this model, with many recent LLMs using LLaMA as its base. Importantly, this version of the model being open-sourced aligns with this paper's aim of opening up research and experimentation with LLMs. We are especially thankful for being granted researcher access by Meta, which gave us the rights to utilise the official model weights.

In fine-tuning, the primary objective is to expose a base model to ample training data, shaping it to produce specific patterns, styles, tones, or domain-specific knowledge [67]. This adjustment involves modifying the model's internal parameters, namely its weights and biases, thereby influencing the recognition of patterns within a neural network. Weights determine signal strength between neurons, impacting the model's pattern recognition abilities [68], while biases offer flexibility in neuron activation, aiding the approximation of complex functions for natural language generation [69]. The fine-tuned weights and biases represent learned inclinations that guide the model in generating natural language descriptions. It is essential to note that the model operates on probabilistic predictions and lacks true intelligence. Instead, anticipates the most likely sequence of words [70]. Fine-tuning aims to enhance the model's likelihood of predicting the correct continuation of a prompt, ultimately shaping its descriptions of events, such as those in football. In this paper, we propose three models for fine-tuning LLMs: (1) The Layered Model (LM), (2) The Mixed Sequentially Model (MSM), and (3) The Mixed Immediately (MIM) model.

The decision to employ three fine-tuning models – LM, MSM, and MIM – for football commentary generation stems from considerations regarding (1) the characteristics of raw football data and (2) the training aspects of large language models (LLMs). Firstly, raw football data encompasses various events and shares similarities with natural language characteristics, such as the processing of emotions. Each event within this data necessitates careful consideration during the training process, particularly in the fine-tuning of LLMs, as undertaken in this study. When deliberating on potential approaches to fine-tuning LLMs for raw football data, we categorised potential options into three main strategies: (i) fine-tuning the selected LLM model by sequentially assigning each event to different layers (LM), (ii) treating all events as a collective batch and repetitively fine-tuning the model multiple times (MIM), and (iii) organising events in a cascaded combination across several layers (MSM). It is worth noting that alternative approaches

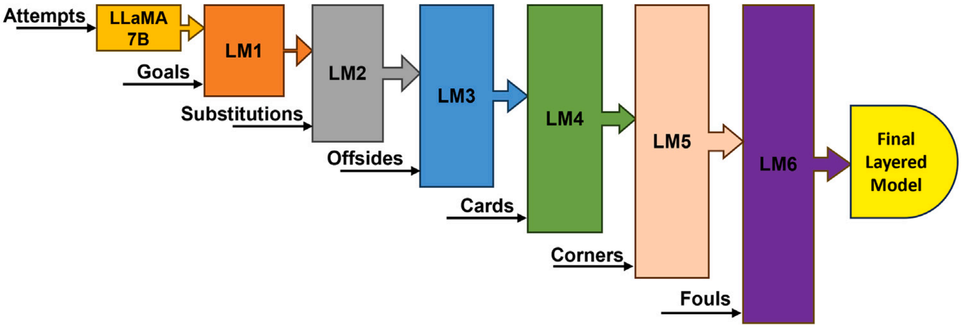


Fig. 2. The layered model (LM) training process flow diagram.

to fine-tuning LLMs may exist beyond the three options proposed in this paper. However, the authors contend that these three proposed options adequately fulfil the objectives of this study, and readers are encouraged to explore and expand upon these strategies further.

4.1. The layered model (LM)

4.1.1. Hypothesis

An LLM acquires proficiency in a layered fashion, achieving mastery in discrete areas sequentially. It should focus on learning one type of event at a time, thereby reducing the cognitive load and potential confusion that might arise from simultaneous multi-event learning. Mastery of one event is used as the foundation for learning the next event, applying the concept of transfer learning.

4.1.2. Experimentation

The model undergoes a series of fine-tuning rounds, each dedicated to learning how to describe a single event type. Training rounds utilise all relevant data, before progressing to the next round. The training continues in this manner until the model has been trained on all events.

4.1.3. The LM details

The Layered Model employs a sequential training approach, where the base model is fine-tuned on the event datasets constructed during the pre-processing stage. The model is trained exclusively on one event dataset at a time, developing a separate level of proficiency in each domain. By the end of the fine-tuning process, this separately learned proficiency will form a unified level of competency that will allow the model to generate descriptions for all events in a football match.

The datasets contain the maximum number of events in their category to expose the model to all the possible scenarios that could take place. For instance, the attempts events dataset contains 228,167 samples, and the model will be trained on every single one. Upon completion of a training round, this refined model serves as the new base for learning the next event dataset. In effect, domain-specific knowledge will be 'layered' onto the base model in sequential training iterations. This layering process is repeated until the model has been fine-tuned on all event types.

This strategy is predicated on the hypothesis that expertise can be compartmentalised and that a deep understanding of individual elements can lead to comprehensive mastery when combined. This is akin to how a human might learn a new topic, with Ericsson and Charness, two psychologists, describing this technique in human experts as "deliberate practice" [71]. By isolating each event type during training, the model is expected to develop a robust understanding of the different types of information needed to describe each event before integrating them into a cohesive whole. The training process is depicted in Fig. 2.

4.1.4. The LM - problems

This approach, however, is highly susceptible to catastrophic forgetting. To counter this, performance will be closely monitored. After every iteration the model will be tested on the new knowledge it has learnt, and all of the previous knowledge gained from previous iterations. Should forgetting occur, 'rehearsal' rounds, as first described by Robins, will be introduced, wherein the model revisits earlier data to reinforce past learning [72].

The discrepancy in dataset sizes and the complexity of events introduces a novel challenge as we approach the upcoming rehearsal rounds. One striking observation is the considerable contrast between the size of the fouls dataset and that of the goals dataset, alongside the notable difference in the complexity of the samples within each. This scenario exemplifies what is commonly referred to as an "imbalanced dataset", a prevalent issue in data science, where crucial data points often tend to be underrepresented [73].

Table 3 illustrates a comparative analysis delineating the distinctions between two sample prompts drawn from each dataset. Notably, the goal prompt manifests a richer content structure, encompassing 16 distinct pieces of information, in contrast to the 10 pieces found within a foul prompt. Moreover, the goal prompt exhibits a significantly more intricate narrative description.

An inherent risk lies in the potential inefficacy of rehearsal rounds should all available data be indiscriminately utilised for every event. Such an approach may inadvertently result in the model disproportionately favouring simpler, more prevalent events, thereby hindering its ability to discern the nuances of complex events effectively. A complexity scoring system has been implemented to address this concern wherein each event type is assigned a respective complexity score. Specifically, attempts and goals are assigned a score of 3, cards receive a score of 2, while the remaining four event types are assigned scores of 1 each.

The complexity score of an event is determined based on several factors, including the number of possible permutations it can undergo, the volume of individual data points associated with it, and the requisite level of detail needed in the commentary to depict it accurately. These complexity scores play a pivotal role in determining the proportional representation of each event type within the datasets earmarked for rehearsal rounds. This strategic allocation ensures that the model receives a balanced exposure to both simple and complex events, thereby fostering a more comprehensive learning experience.

For instance, a rehearsal round following the third fine-tuning round would include attempts, goals, and substitution events in a 3:3:1 ratio, reflecting their complexity scores. The underlying principle is that the model may require more exposure to complex events like goals to achieve proficiency when compared to simpler events like fouls. This methodology is conceptually similar to the way stratified [74] and weighted [75] sampling techniques are used to address data imbalances, with SMOTE being a famous example of an attempt to solve the problem of imbalanced datasets [76].

4.1.5. The LM - success

The success of this strategy is measured by the model's ability to generate accurate and contextually relevant commentary for each type

Table 3
Example prompts: (a) Foul, (b) Goal.

```
{"text":
"### Human: Below is a series of pieces of
information describing an event in a soccer match
paired with an output that describes the event
based on the pieces of information. Acting as an
expert soccer commentator, describe this event in
an informative and engaging manner.
### Event: Foul.
### Time: 39.
### Event Team: Napoli.
### Opponent Team: Sassuolo.
### Player: marek hamsik.
### Assistant: Foul by Marek Hamsik (Napoli)."}
```

(a)

```
{"text":
"### Human: Below is a series of pieces of
information describing an event in a soccer match
paired with an output that describes the event
based on the pieces of information. Acting as an
expert soccer commentator, describe this event in
an informative and engaging manner.
### Event: Goal.
### Time: 34.
### Home Team: Caen.
### Away Team: Valenciennes.
### Event Team: Caen.
### Opponent Team: Valenciennes.
### Player: gregory proment.
### Assist by: nicolas seube.
### Assist method: Pass.
### Pitch Position: Outside the box.
### Goal Placement: Top right corner.
### Goal Following: Open play.
### Foot: right foot.
### Score: Caen 1, Valenciennes 0.
### Assistant: Goal!  Caen 1, Valenciennes 0.
Gregory Proment (Caen) right-footed shot from
outside the box to the top right corner. Assisted
by Nicolas Seube."}
```

(b)

of event after the training process. It should be able to accurately predict what the description should be when presented with a prompt containing the details of an event in the football game.

4.2. The Mixed Sequentially Model (MSM)

4.2.1. Hypothesis

Sequential training of LLMs on domain-specific tasks often leads to catastrophic forgetting. The use of imbalanced training datasets that prioritise complex and crucial data types might offer a mitigation strategy. Consistent re-exposure to previous knowledge, coupled with the introduction of new information, could help retain previously learned information. By training LLMs on incrementally imbalanced datasets that combine old and new knowledge, there is a potential for the model to become proficient in describing a wide range of events without dedicated rehearsal rounds.

4.2.2. Experimentation

The proposed fine-tuning approach involves sequentially introducing event types across multiple training rounds. In each round, the training datasets will consist of a mix of previously learned events and new events. This strategy aims to maintain a balanced focus on learning both high-complexity and low-complexity events, utilising the concept of complexity scores to guide the training process.

4.2.3. The MSM details

The Mixed Sequentially Model employs a sequential training process with a greater emphasis on knowledge retention. This model iteratively layers event data, similar to the Layered Model, but with a key modification: each iteration integrates a blend of previously learned events with the new event data.

The overarching goal of this strategy is to curtail catastrophic forgetting. By interleaving old data with new, the model is continually prompted to recall and reinforce prior knowledge, potentially eliminating the necessity for rehearsal rounds. This method draws inspiration from spaced repetition, a cognitive science principle known to enhance long-term retention [77].

The same issue is present in this strategy as with the previously described rehearsal concept; if all available data is used, the model may be overwhelmed with the sheer number of low-complexity events (corners, fouls, etc.) and will be unable to grasp the intricacies of high-complexity events (goals and attempts).

To address this, the training datasets for this model are all constructed as imbalanced datasets. The concept of using complexity scores to determine the specific ratio of events relative to each other has been taken further, with every training dataset having a purposefully disproportionate representation of events. The intention behind this approach is to ensure that the model spends an appropriate amount of time learning about each type of event, based on its complexity and importance. This should help the model develop a more balanced understanding of both simple and complex events.

Fig. 3 details the distribution of event data across fine-tuning rounds. New events are sequentially mixed in the training dataset, but the proportion of each event to the whole, as decided by its complexity score, is always maintained.

4.2.4. The MSM - problems

Despite its preventative design, this strategy may encounter limitations. Overfitting may occur due to the model being exposed to the same event for potentially seven iterations. To minimise this risk, and to give the model the best chance to learn the nuances of each event, each training dataset will contain randomly selected samples from the master dataset. For example, the attempt events used in the second fine-tuning round will not necessarily be the same events used in the first round.

Moreover, the model may under-fit to foul events, as it will only be exposed to foul training samples once in the final fine-tuning round, and at a greatly reduced number compared to the Layered Model. If this happens, an extra fine-tuning round may be inserted at the end to give the model a chance to fully grasp the ability to describe fouls.

4.2.5. The MSM - Success

Success for this model would be demonstrated by its ability to generate accurate descriptions for each event it has been trained on, without the need for rehearsal rounds during its training process. If successful, this strategy will prove more efficient than the Layered Model due to its reduced data consumption and successful mitigation of catastrophic forgetting. This aligns with the paper's objectives of determining optimal approaches to fine-tuning language models on consumer-grade hardware.

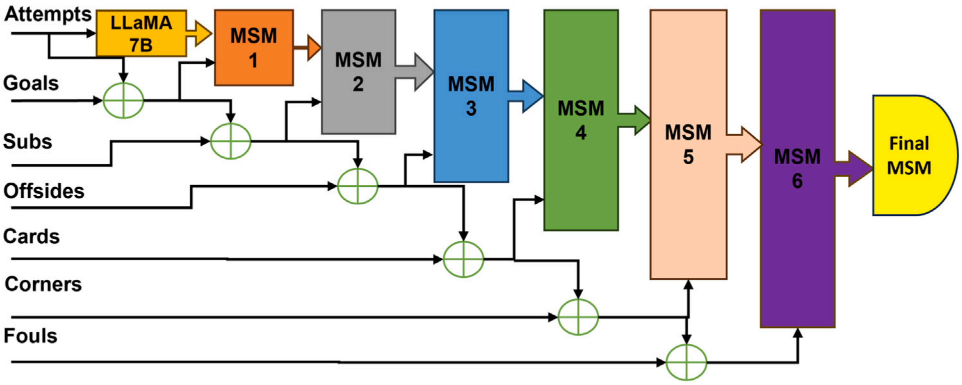


Fig. 3. The MSM training process flow diagram.

4.3. The Mixed Immediately Model (MIM)

4.3.1. Hypothesis

A sequential training approach may not be necessary for LLMs when dealing with similar events. The inherent pattern recognition capabilities of a sophisticated model might enable it to concurrently learn different but similar event types, given sufficient training time. This hypothesis is based on the premise that the events used to train the models in this paper are more similar than different, and the model could effectively differentiate between these events without the need for isolated, sequential training.

4.3.2. Experimentation

The training process is built on the continued use of imbalanced datasets and complexity scores. In each round, the model will train on a dataset that comprises a blend of all event types from the outset. The events will maintain their complexity-based ratios, ensuring that the model is consistently challenged to comprehend low and high-complexity events proportionally throughout its training. This approach will test the model's capacity for simultaneous multi-event learning and its ability to generalise across a diverse spectrum of training data.

4.3.3. The MIM details

The underpinning principle of this strategy is inspired by the theory of distributed practice, which suggests that learning is more effective when exposure to information occurs in multiple, spaced-out sessions [78]. The knowledge needed to be learnt is presented as a single corpus that is repeatedly looked at until understanding is achieved. This approach is designed to reduce the total number of iterations required for the model to assimilate the knowledge by pushing the base model to learn as many different types of events as possible in as quick a time as possible.

Fig. 4 illustrates the composition of the training data in each iteration of fine-tuning. To give the model as best a chance as possible at learning the nuances of each event type, five rounds of fine-tuning are planned. An evaluation takes place at the end of the fifth round to determine if the model requires more training time.

This model may exhibit a dip in performance relative to its counterparts during the early stages, which can be attributed to the inherent complexity of learning multiple event types simultaneously. Nonetheless, it is hoped that the model will eventually achieve performance parity with the Layered Model and Mixed Sequentially Model in fewer iterations.

In direct contrast to the Mixed Sequentially Model, the Mixed Immediately Model proactively tackles the potential drawbacks of integrating events at later stages of the training process. Integrating all events from the beginning ensures equitable learning opportunities for each event category, potentially leading to a more uniform and thorough proficiency in the event description. Also, unlike the Layered Model, there is a negligible chance of catastrophic forgetting in this model's fine-tuning. This is because the base-model weights will be updated with all of the different event types at the same time, leaving no opportunity for older knowledge to be erased and replaced with new information.

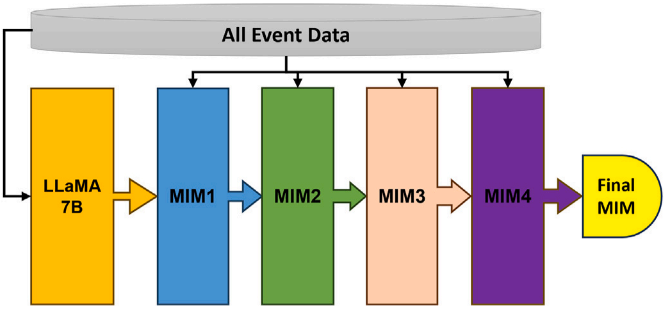


Fig. 4. The MIM training process flow diagram.

4.3.4. The MIM - problems

One of the inherent challenges for the model will be its ability to effectively differentiate between similar football events. Given the model's exposure to a diverse array of events simultaneously, there is a risk that it might struggle to distinguish between events that are closely related or have overlapping characteristics. For instance, differentiating between a 'goal' and an 'attempt' could be challenging, as both involve shots at the goal but with different outcomes.

It is hoped that the differences in each event type's prompts (due to the different pieces of information being needed) will limit this risk. Performance will have to be closely examined between fine-tuning rounds to see how the model is evolving. Adjustments, such as tweaking the proportions of the events, may be made to the training dataset if the model is found to be struggling with the simultaneous learning of seven different event types.

Additionally, this model has a heightened risk of over-fitting the training data as it will be exposed to all of the different types of events numerous times. To address this, the datasets in each round will consist of the same ratio of events, but with different samples randomly picked from the master datasets in each round. This ensures exposure to the many possible combinations of information that could be present in one event and reinforces the model's ability to generalise across different data points.

4.3.5. The MIM - success

Success for this model is defined by its ability to achieve performance parity with the previous two models in generating accurate descriptions of football events. This achievement would be particularly significant if realised within a shorter time frame, as it would demonstrate greater efficiency in the training process than the other two models. Moreover, accomplishing this without any negative impacts from the simultaneous training on multiple event types would underscore the effectiveness of this approach. If this model meets these benchmarks, it will substantially enrich our understanding of optimal fine-tuning practices for modern, locally-run language models, and will especially benefit our understanding of how to manage imbalanced training datasets. It will also give a greater understanding of the potential for fine-tuning small, modern LLMs on a limited amount of data.

5. Experimental analysis & results

LLM-Commentator with its three candidate fine-tuning models was experimentally tested under the England and European data sets. To evaluate the LLM-Commentator models' learning progression, each iteration of the different models underwent a comprehensive testing regime post-training. This was done by building a suite of tests that measure the ability of the model to generate descriptions of unseen events from real football matches. These events were siphoned off the training dataset randomly at the start of the process, ensuring a representative mix of event types to accurately assess model performance across various scenarios.

Each event had its own test dataset of 100 different examples. This size was chosen as it provides a statistically significant sample to confidently assess the model's performance, balancing thoroughness with the practicality of testing duration. To ensure valid results, all hyperparameters were kept the same during training which are given as:

- **Precision:** 4-bit NormalFloat (NF4)
- **Epochs:** 1
- **Batch size:** 16
- **Weight Decay:** 0
- **Warm-up ratio:** 0.03
- **Learning rate:** 0.0002

Test datasets were created by splitting the event samples into two: a prompt and a reference. A regular expression was used to delineate the end of the event information (the prompt) and the start of the event description (the reference). This method ensures that the model generates descriptions based solely on the event information, without being influenced by existing or past descriptions.

Each iteration, after being archived in a HuggingFace repository for the sake of reproducibility and transparency [79], was systematically presented with prompts from the test datasets. The iteration's task was to generate an accurate description of the event. This then was quantitatively and qualitatively compared against the reference description. Analysis was done with a range of tests that provided a multifaceted view of the model's capabilities:

- **Direct comparison:** This binary test is a direct comparison between the generated text and the reference text. If the iteration generates a description of an event that is identical to the reference description, one point is awarded. If they are different, no score is given. Whilst this metric has obvious limitations, it offers an immediate and straightforward approximation of each model's capabilities.
- **Rouge Score:** This is a collection of metrics that evaluate text summarising quality by measuring the overlap of individual words, word pairs, and the longest common sequence between the generated and reference descriptions [80]. Rouge scores help display whether the generated description contains the same essential points as in the reference. It does this by providing three scores: Rouge-1 (R1), Rouge-2 (R2) and Rouge-L (RL) where unigrams (individual words), bigrams (word pairs) and the longest common subsequences are awarded, respectively.

At the end of each testing round, the three rouges scores for all one hundred prompts were averaged out. This gave one Rouge score for each iteration per test dataset. The final iterations of each model were subjected to advanced testing to evaluate their performance in generating accurate and relevant football commentary. This determines which model can best generate natural language descriptions of events in a football match, thus signifying which training approach produced the most capable model. This testing focused on three metrics: precision, recall, and an F1 score, which are widely held standards in measuring NLP models [81].

All experiments will be run in a system with the following specifications: CPU: Intel i9-13900K, GPU: NVIDIA GeFORCE RTX 4090 (24 GB VRAM), and RAM: DDR-5 (64 GB). Python was used as the primary programming language due to its extensive support for NLP libraries and ease of integration. All data pre-processing and model testing was done using Python 3.11. QLoRA was used as the primary fine-tuning method. Qlora considerably reduced the memory requirements needed for fine-tuning, thus making this paper possible. QLoRA is in turn dependent on the following libraries: transformers [82], bitsandbytes [83], accelerate [84], peft [85], PyTorch [86] and CUDA [87].

5.1. The LM test results

The quantitative performance results for the LLM-Commentator with the LM fine-tuning are given in Table A.6 and depicted in Fig. 5. Examining the results in Fig. 5 for DC (solid lines with different colours), shows that comparison scores are generally high immediately following training for the first time on new data. For example, the third iteration (Subs), generated 98 perfect descriptions of substitution events after being exposed to substitution training data just once. Rehearsal rounds (indicated in the table by the R:x group of iterations) proved to be a necessity as the LM suffered from bouts of catastrophic forgetting at regular intervals. This is most evident in the fifth iteration's (Offsides) drastic drop in performance when describing a substitution event — plummeting from a score of 95 in the previous iteration to a score of just 6. Implementing a rehearsal round after this iteration proved successful in re-instilling the model with the capability to describe substitution events. Performances on past knowledge usually dipped in each sequential iteration before rising again due to the effect of a rehearsal round.

The base model initially learned how to describe attempt events well, but this was followed by a significant decline in performance over the next two iterations. This trend of worsening performance over time was reversed with the inclusion of rehearsal rounds. The majority of the volatility in the model scores occurred in earlier iterations. A noticeable stabilisation in scores across different datasets was observed after the second rehearsal round. The first two rehearsal rounds had a more significant impact compared to the third and fourth, suggesting a diminishing return in the effectiveness of rehearsal rounds over time. The final iteration's (R:4) scores across the different datasets were high. The final model scored extremely well on low-complexity events and moderately well on higher-complexity events, indicating the success criteria for this strategy were met.

Examining Rouge scores in Fig. 5 (dashed-RL, dash-dotted-R2 and dotted-R1 lines with different colours), we conclude that the Rouge scores indicate a generally high level of competency across all iterations, and present a more nuanced picture of performance than the comparison tests. For instance, whilst the comparison test showed a 46-point drop in the ability to describe 'Attempt' events from the first to the third iteration (from 64 to 18), the R1 and RL scores only dropped from 0.97 to 0.92. This indicates that, despite some variations from the reference descriptions, the iterations consistently captured the essential words and structure of the description in its generated commentary, and the drop in performance is not as bad as first suggested by the direct comparison (DC) test scores.

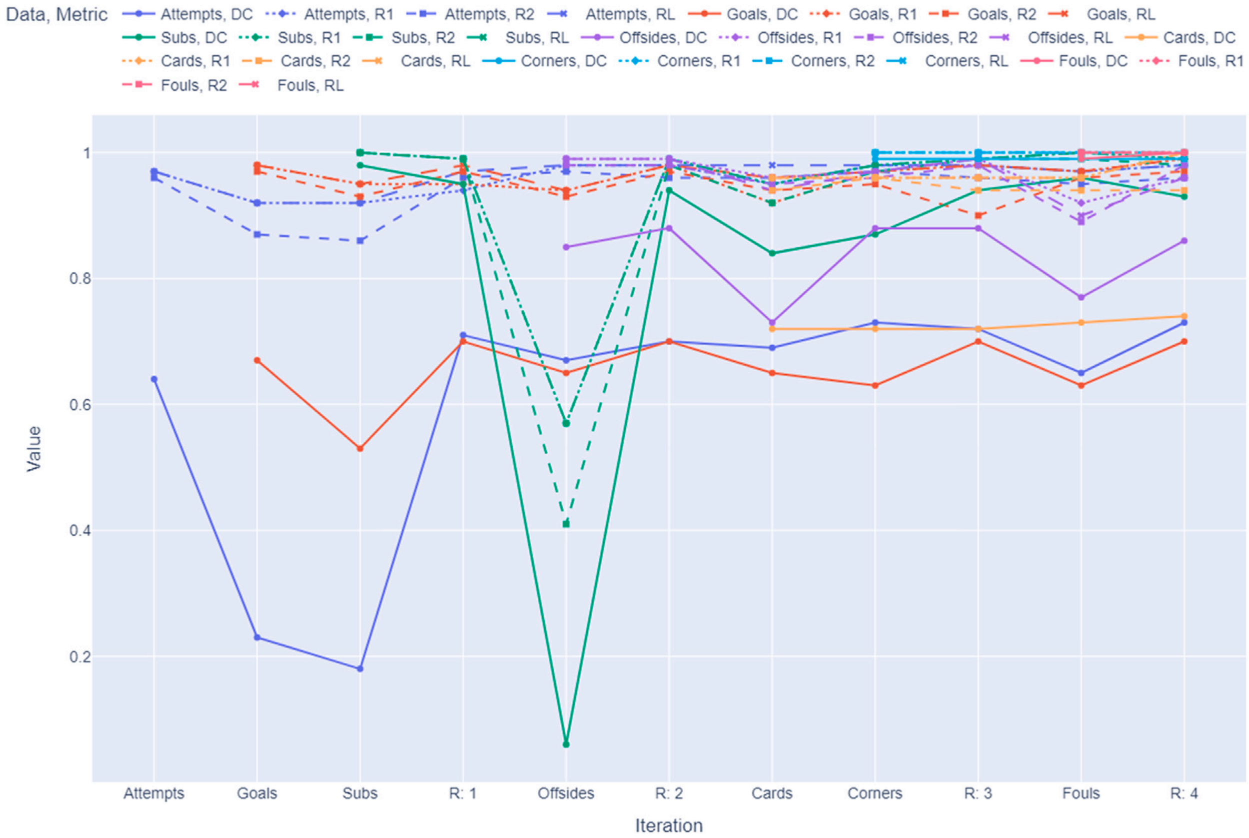


Fig. 5. The LM results in terms of DC, R1, R2, and RL. (For more details see Table A.6).

The drop in performance of the third iteration's ability to describe Attempt events is most pronounced in its R2 score, falling from 0.96 in the first iteration to 0.86 in the third. This suggests that catastrophic forgetting had the biggest impact on the third iteration's ability to generate the correct bigrams (word pairs) when describing attempts. Overall, R2 scores were lower than the other two metrics, suggesting there is more of a challenge for the model in accurately generating the correct word pairs in its descriptions

Relatively higher R1 and RL scores were common when the model tested on low-complexity events (such as fouls and corners). This could be attributed to the simpler linguistic structure and less varied content of these events. The Rouge scores also corroborate the beneficial impact of rehearsal rounds on performance when recalling formerly learned knowledge. Rouge scores for iterations of the model that have just completed a rehearsal round are higher across the board than iterations that came before the rehearsal. Most importantly, the Rouge scores show that when the model incorrectly replicated the reference description, it was due to small margins of error rather than large mistakes.

5.2. The MSM test results

The quantitative performance results for the LLM-Commentator with the MSM fine-tuning are given in Table A.7 and depicted in Fig. 6. The results in Fig. 6 for DC (solid lines with different colours), suggest that the Mixed Sequentially Model has a capacity for a quicker pace of learning compared with the Layered Model. This is particularly evident in its handling of new event types and the performance parity in most events between both models' final iterations. The model's ability to quickly grasp and accurately describe new data is most apparent in the sixth (corners) and seventh (fouls) iterations. After just a single round of exposure, these iterations were able to generate descriptions of corner and foul events that matched the reference descriptions with an accuracy of 98% and 100%, respectively.

However, the model's performance when describing high-complexity events (attempts, goals, cards), presents a more varied picture. The first iteration, for instance, struggled to accurately describe attempt events, with less than 50% of its outputs matching the reference descriptions. This is 14% lower than the performance of the first iteration of the Layered Model when describing attempts. There was a noticeable plateau in performance across these high-complexity events, with no significant improvements observed between earlier and later iterations. A particularly concerning observation is the dramatic drop in the final iteration's ability to describe goal events, where the accuracy fell to less than half of that in previous iterations. This indicates a potential systemic issue in how this model processes and learns from goal event data, as the final iteration could not generate descriptions of goal events that earlier iterations could. This is surprising, as the final iteration, which should be the strongest model, generated a drastically worse description of the goal event than all previous iterations. It incorrectly described the score, the position of the player when they scored, and the placement of the ball in the net.

Examining Rouge scores in Fig. 6 (dashed-RL, dash-dotted-R2 and dotted-R1 lines with different colours), we conclude that The Rouge scores for the MSM reveal a consistent level of proficiency across the iterations, with softened declines in performance rather than sudden drops. This suggests that even when the model's outputs deviated from the reference descriptions, they remained closely aligned in terms of linguistic structure and content. Several near-perfect scores when testing on low-complexity events demonstrate the model's adeptness at learning and handling simpler events.

R1 scores were consistently high across all datasets. This indicates a strong ability of each iteration to correctly generate the right vocabulary for each event. This is most evidenced by the perfect scores gained by later iterations when describing corners and fouls.

The R2 scores also reflect a high level of proficiency across the iterations. However, a slight decline in scores for high-complexity

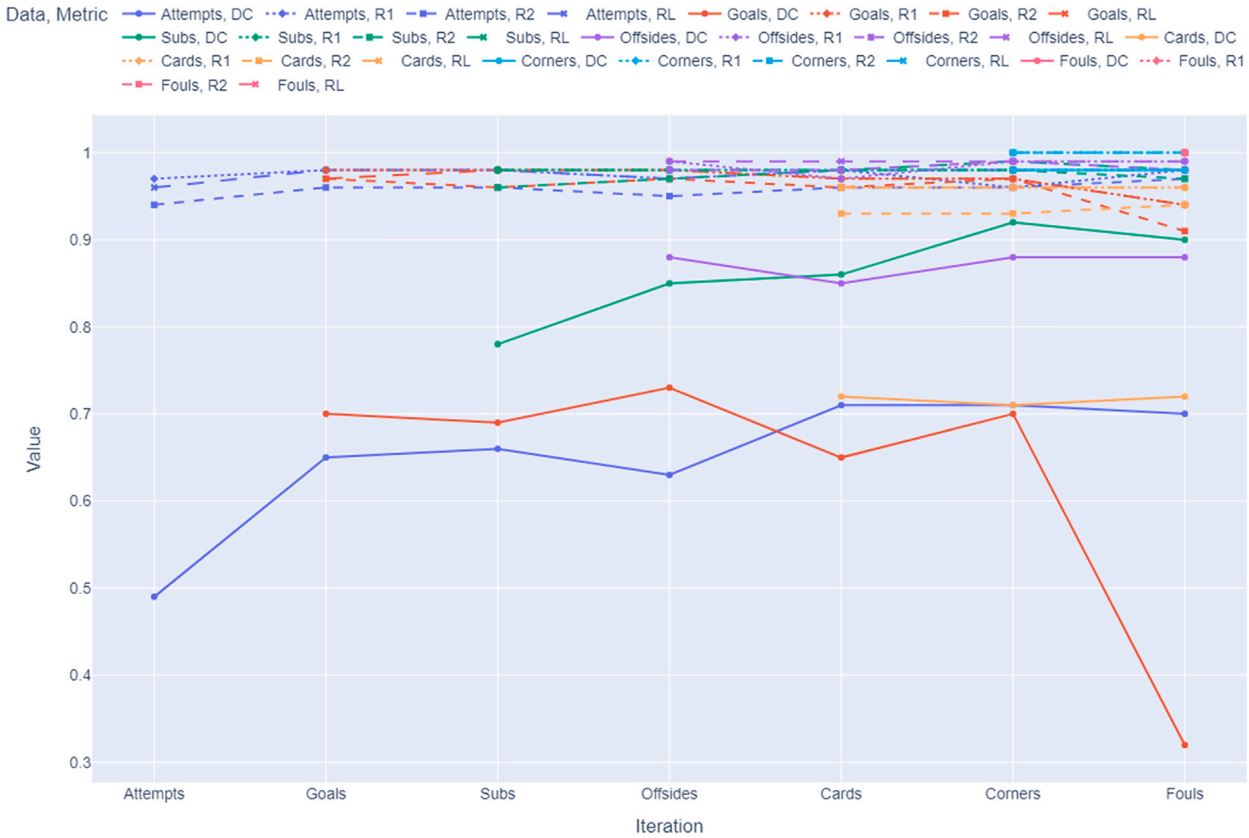


Fig. 6. The MSM results in terms of DC, R1, R2, and RL. (For more details see Table A.7).

events suggests challenges in maintaining the same level of accuracy when dealing with more intricate descriptions. For example, R2 is the only metric that shows that the final iteration struggled to describe goal events, receiving an average score of 0.91 when describing 100 events. This suggests that the final iteration was commonly incorrectly sequencing the correct word pairs together, generating descriptions that were in a different lexical order than the reference description.

RL scores follow the same trend of high competency across the different iterations as R1. This shows that most iterations are excellent at getting both the details of the description, and the larger grammatical structure and syntax identical to the reference description

5.3. The MIM test results

The quantitative performance results for the LLM-Commentator with the MIM fine-tuning are given in Table A.8 and depicted in Fig. 7. This model was expected to perform poorly in earlier iterations and slowly improve over each round of fine-tuning. Examining the results in Fig. 7 for DC (solid lines with different colours), we can see that the model performed strongly from the very first iteration. Whilst the first iteration did struggle to describe attempt and foul events, test scores for these two events climbed in every subsequent iteration. The poor performance of the first iteration on foul events can probably be attributed to underfitting, as the generated descriptions show signs of the model being confused as to which event to describe.

Performances for all iterations started high and remained high. There are no symptoms of overfitting or catastrophic forgetting as seen in the results of the other two models. However, the model did exhibit signs of a plateau in learning, particularly when dealing with high-complexity events like goals. The performance in these areas remained stable across iterations, exhibiting little improvement or decline. This observation could indicate a ceiling in the model's learning capacity for these types of events.

Overall, the results of the Mixed Immediately Model vindicate its training strategy. It reached parity in performance across the different events in a shorter amount of time and with a much more straightforward training approach. The final iteration of the model is on par with the Layered and Mixed Sequentially model, despite using vastly fewer data than the former and going through fewer fine-tuning rounds than both. The comparison scores indicate that the model coped very well with the demands of simulating multi-event learning.

Examining Rouge scores in Fig. 7 (dashed-RL, dash-dotted-R2 and dotted-R1 lines with different colours), we conclude that the Rouge scores corroborate the results from the comparison tests. On the whole, the model outperformed expectations with its strong performances from the first iteration, and performances following the first iteration remained very stable. For most test datasets, the model maintained high R1 scores from the first iteration, with scores generally ranging from 0.96 to 1.0. This consistency is a strong indicator of each iteration's ability to correctly capture the target vocabulary for each event during training. The R2 and RL scores were also high, suggesting that the model was not only choosing the correct words but also placing them in an order that closely mirrors the reference texts.

A key observation from the Rouge scores is the disparity in the first and second iterations' performance when describing foul events. The performance of the first iteration on fouls was the lowest out of any event for all three scores. However, the second iteration scored perfectly on each Rouge test when describing foul events. In the span of just one training round, the model went from performing worst when describing foul events to describing them perfectly. The Rouge scores collectively suggest that the model is highly effective in generating text that is lexically similar to the reference material and maintains the original texts' structural and sequential integrity. That it did so from the very first fine-tuning round is a testament to the effectiveness of the training strategy.

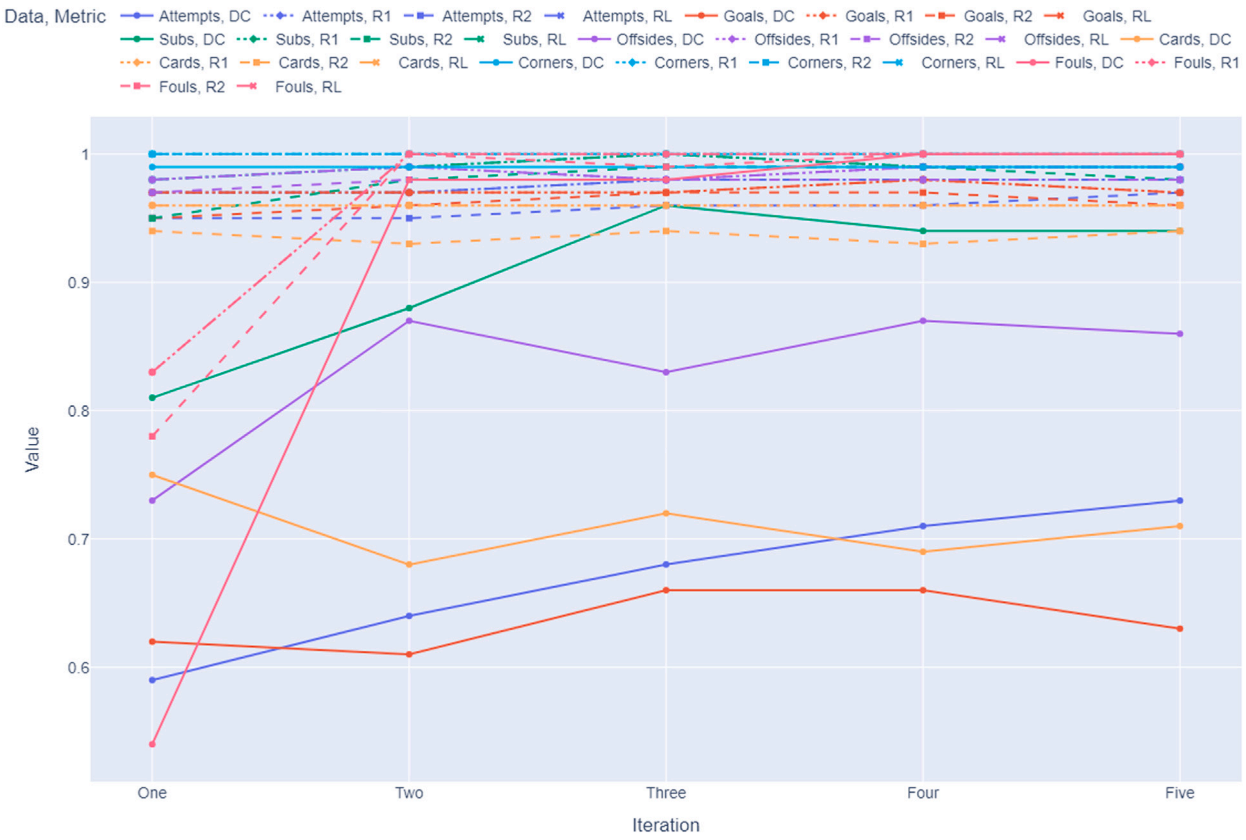


Fig. 7. The MIM results in terms of DC, R1, R2, and RL. (For more details see Table A.8).

Table 4
Model comparison after the final iteration.

Model	Acronym	TP	FP	FN	Precision	Recall	F1
Layered	LM	**0.61**	**0.11**	**0.00**	**0.84**	**1.00**	**0.92**
Mixed Sequentially	MSM	0.54	0.18	**0.00**	0.75	**1.00**	0.85
Mixed Immediately	MIM	0.60	0.12	**0.00**	0.83	**1.00**	0.91

5.4. Final comparison

This subsection presents a comparison study of all three proposed fine-tuning techniques for commentary generation via the LLM-Commentary method in terms of precision, recall and F1 scores. The total number of true positives, false positives, false negatives, the precision, recall, and F1 score for each of the final iterations are given in Table 4.

Examining Rouge scores in Table 4, we conclude that the Layered Model achieved the highest F1 score of 0.92, indicating a strong balance between its precision and recall capability. This was followed closely by the Mixed Immediately Model with a score of 0.91. The Mixed Sequentially performed the worst, with an F1 score of 0.85. This was due to its lower precision score when compared against the other two models. All models had a perfect recall score, with none of them failing to generate a description of a single event.

Furthermore, a comprehensive evaluation of the fine-tuning models is conducted, considering their advantages and disadvantages, as well as the findings derived from the experimental investigation described earlier. These findings are summarised in Table 5. The performance analysis indicates that the initial fine-tuning model based on LM exhibits superior performance with a decreased computational burden due to the implementation of deliberate practice. Nonetheless, it is prone to experiencing catastrophic forgetting of multiple events, as anticipated. This issue is addressed by incorporating rehearsal rounds, resulting in a greater number of iterations and mitigating the aforementioned challenge.

The use of the MSM, as opposed to the LM, aims to prevent catastrophic forgetting through a step-by-step training method. While the MSM eliminates the need for rehearsal rounds, it has led to overfitting in certain instances and underfitting in events trained later in the sequence. Consequently, this has resulted in lower performance metrics compared to the LM, despite requiring fewer iterations to train. Currently, the MIM has surpassed the MSM in effectiveness. By employing a non-isolated event training approach, the MIM achieves faster learning and performance better than MSM and is comparable to the simpler LM model. Unlike the LM, the MIM does not exhibit catastrophic forgetting, although it shows signs of a learning plateau, particularly in complex events such as Goals.

ROUGE scores demonstrate a more consistent yet detailed representation of results across all three proposed fine-tuning models. In contrast to direct comparison metrics, the R1, R2, and RL scores exhibited smaller fluctuations, while also showcasing improvements in fine-tuning for uni-grams, bi-grams, and L-grams. Despite achieving comparable performance levels for single-word predictions and overall sentence reconstructions, challenges arose specifically with word pairs (bi-grams) as indicated by the R2 scores. This suggests that the models are more susceptible to combining incorrect word pairs while maintaining the overall structural integrity. An illustration of this challenge is provided below:

```
Goal! Stuttgart 3, Schalke 0. Shinji Okazaki
(Stuttgart) left-footed shot
from outside the box to the top left (right) corner.
Assisted by Zdravko Kuzmanovic following a set-piece
situation.
```

Table 5
Overall comparison between the proposed fine-tuning models.

	LM	MSM	MIM
Reason to use	Learn an event at a time	Consistent re-exposure to previous knowledge	Sequential training but no isolated events
	Reduce cognitive load	Previous knowledge + new event	Needs for fewer iterations
	Remove multi-task learning problems	Reduced catastrophic forgetting risk	The classical ML training procedure
	Deliberate practice [71]	No need for rehearsal rounds	No catastrophic forgetting
			No need for rehearsal rounds
Drawbacks	Catastrophic forgetting	Overfitting	Struggle to distinguish between closely related events
	Needs for rehearsal rounds [72]	Needs for extra fine-tuning rounds	Risk of overfitting
	Longer to fine-tune	Lastly added event under-fits	
	Weaker for imbalanced data		
Findings	Catastrophic forgetting experienced	Faster learning (simple events - corners) compared to LM	Poor performance at earlier iterations
	Rehearsal rounds proved necessity	Lower performance (complex events - attempts) in earlier stages	Struggle to discriminate fouls and attempts
	Single words and general structure saved	Un-explained drop in Goals performance at the last iteration	Signs of a plateau in learning (Goals)
			Similar performance to LM and MSM with fewer data and iterations
	R scores showed nuanced performance visualisation		

For instance, in a commentary of this nature, the anticipated result is the identification of the "left corner" as the accurate area of the goal. However, across several iterations, all models consistently label it as the "right corner", leading to fluctuating R2 values, while R1 and RL are not significantly impacted by this discrepancy.

6. Discussions

6.1. The LM - discussion

The Layered Model learning strategy, designed to instil domain-specific knowledge in a series of discrete training rounds, demonstrated superior performance in generating accurate and contextually relevant commentary for various types of events. Test results indicated that the Layered Model outperformed the other two models in most comparison and ROUGE tests, particularly excelling in recall and precision in its final iteration. However, a significant drawback emerged as this training method proved highly susceptible to catastrophic forgetting. The model tended to forget how to describe events, particularly evident during the fine-tuning process with the third iteration. The incorporation of a rehearsal round mitigated some issues, but catastrophic forgetting persisted with the introduction of new events, highlighting a challenge that warrants further consideration in model refinement.

Rehearsal rounds prove effective in reducing catastrophic forgetting in neural network iterations, but their impact appears to be transient, possibly due to limitations in the base model's architecture for retaining long-term knowledge across diverse domains. To address this, implementing adaptive learning techniques, as discussed in Gururangan et al.'s study [88], where models focus training on areas they are prone to forgetting, could offer a more lasting solution. Notably, the comparison test, though simple, highlights catastrophic forgetting more strongly than the Layered Model's ROUGE scores, suggesting a need for more comprehensive evaluation methods. The persistence of catastrophic forgetting in fine-tuning, a longstanding issue in neural network research, challenges the expectation that advancements in model architecture over decades would reduce its impact. Catastrophic forgetting was first documented by McCloskey and Cohen in 1989 and remains an obstacle. [89].

6.2. The MSM - discussion

The Mixed Sequentially Model, designed to mitigate catastrophic forgetting, successfully integrated old knowledge with new, achieving comparable performance to the Layered Model with fewer training rounds. This success stemmed from the model's ability to balance the data points in the training datasets, thus preventing the exclusive influence of recent data on weights and avoiding catastrophic forgetting. The continuous adjustment of weights accommodated both old and new information, aligning with the model's fine-tuning strategy. Additionally, the model's proficiency in generating descriptions of low-complexity events, despite exposure to fewer samples (see Fig. 8(a)), underscored the effectiveness of categorising events by complexity and addressing dataset imbalances. Notably, for generating simplistic linguistic patterns, the study found that the quantity of training samples did not significantly impact the Language Model's performance.

The Layered Model encountered foul events 267,229 times during training, while the Mixed Sequentially model experienced foul events only 9000 times. Despite the significant difference in exposure, the Mixed Sequentially model consistently generated flawless descriptions of foul events, aligning with the 'Few-shot' approach to language model fine-tuning. This suggests that for certain tasks, proficiency can be achieved with a limited, high-quality dataset and minimal training rounds [10]. However, the Mixed Sequentially model's subpar performance in describing goal events prompts further investigation. Unlike other event types, the model, from the second iteration onwards, consistently trained on the same 27,000 goal event samples. This potentially led the model to overfit the training data. Using the samples in every fine-tuning round was a necessity due to the limited size of the goal events dataset (27,757 goal event samples – only 3.6% of all data – see Fig. 8(b)) when compared to the total number of events.

This overfitting resulted in a diminished ability to describe goal events in the final iteration, emphasising the importance of a diverse and balanced training dataset. Although the success criteria were not fully met, the model successfully avoided catastrophic forgetting. The overfitting issue observed in the final iteration was primarily attributed to the constrained training data. Recommendations for future work include exploring strategies to increase sample diversity or employing data augmentation techniques to mitigate overfitting, particularly when

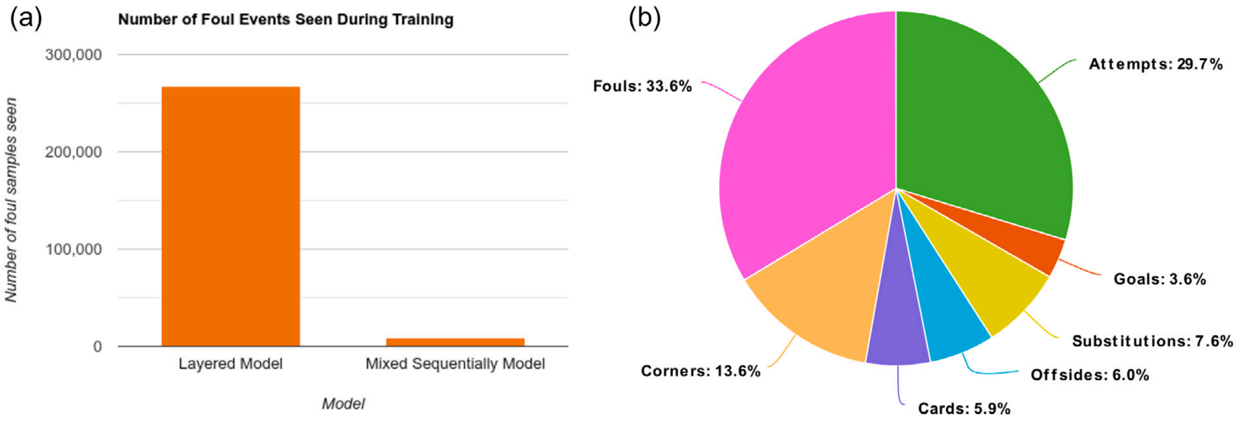


Fig. 8. The MSM discussion supportive plots. (a) The total number of foul events the Layered Model and Mixed Sequentially Model were exposed to during training, (b) The proportion of each event type in the source dataset. The small number of goal events meant that the same events were reused many times during training, potentially leading the model to over-fit.

dealing with limited data for specific event types. Despite the noted limitation, this does not entirely dismiss the fine-tuning strategy. Its potential may be realised with a more expansive training dataset.

6.3. The MIM - discussion

The fine-tuning strategy for the model was an experimental initiative aimed at dynamically pushing the base model's limits, with limited expectations of success. The model, exposed to all event types simultaneously, demonstrated a quick adaptability and learning capacity under chaotic fine-tuning conditions. Despite slightly lower scores in describing attempts or goal events compared to a layered model in the first iteration, the gap was surprisingly narrow, challenging the assumption that sequential, isolated training rounds are necessary for teaching a language model to generate football commentary. The results suggest that grouping similar tasks, as in multi-task learning, allows the model to develop a comprehensive understanding of training data, and to leverage its knowledge across different event types. The significance of complexity scores in distinguishing high and low-complexity events may be overstated, suggesting that modern language models may not heavily differentiate between them in the context of football commentary.

Another takeaway from this training strategy is the general trend of plateauing in the model's performances when describing high-complexity events. Whilst scores were good, they did not match the heights that the model set when describing low-complexity events. This suggests one of three things. There may be a limit to the effectiveness of the training strategy, there may be a limit to the capabilities of the base model, or, perhaps the most likely factor, the training data is not as clean as was previously thought.

Combining back through the 767,843-row dataset does reveal some anomalies that were not caught in the pre-processing stage. For example, the card training dataset was found to contain erroneous reference descriptions:

```
### Assistant: Booking. Booking      Ashley Westwood
(Aston Villa) is shown the yellow card.
```

When looking through the descriptions of events the model generated for the tests, it was found to have repeated these anomalous linguistic patterns. This suggests that unclean data may have played a prominent role in the model's performance plateaus. An example of this is given below:

```
EXPECTED ANSWER: . Ben Mee (Burnley) is shown the yellow
card for a bad foul.

PREDICTED ANSWER: . Booking      Ben Mee (Burnley) is
shown the yellow card for a bad foul.
```

Because of the small number of such anomalous training samples, this finding does not completely explain why the model's abilities to describe certain events plateaued. But it did play a role, and it is interesting to see that the model is sensitive enough to recall patterns it was exposed to during training, even if was only for a small number of times.

In the final iteration, the Mixed Sequentially Model did not exhibit overfitting on goal events, in contrast to previous instances, potentially due to a shorter training period. For further investigation, the model could be exposed to two additional fine-tuning rounds to assess its susceptibility to overfitting on goal event data. Despite this, the model successfully met its predetermined criteria by achieving performance parity with the other models in fewer training rounds. The simultaneous multi-event learning emphasis in the Mixed Immediately Model, which is shown here to be effective, presents broader applications for LLM training, particularly in scenarios requiring simultaneous learning of diverse, yet related, data, such as in multi-lingual translation models or interdisciplinary subject understanding.

6.4. Collective discussion of fine-tuning models

All models achieved a perfect recall score, which reveals some key insights. Primarily, it underscores the models' capability to comprehensively cover the total sum of events in a football match. This suggests a high level of effectiveness in providing exhaustive coverage of match occurrences for all models.

Additionally, the perfect recall score implies a training dataset characterised by both variety and diversity. The imperfections of the dataset have been examined in previous sections, but, despite its flaws, it equipped the models with the necessary breadth to accurately describe a wide array of potential events in a football match.

Also, the recall scores demonstrate the models' high sensitivity, as they successfully detected and generated descriptions for every single event they were trained on. However, this brings to light a critical limitation: the models' proficiency in detecting events is confined to those included in their training data. Given the unpredictable nature of football, the models' performance in response to untrained or unforeseen events remains uncertain. This limitation underscores the need for further model training and adaptation to encompass a broader range of possible match events.

7. Conclusions

7.1. Summary & remarks

In conclusion, this study delves into NLP research by exploring its application in generating live commentary for football matches. Three

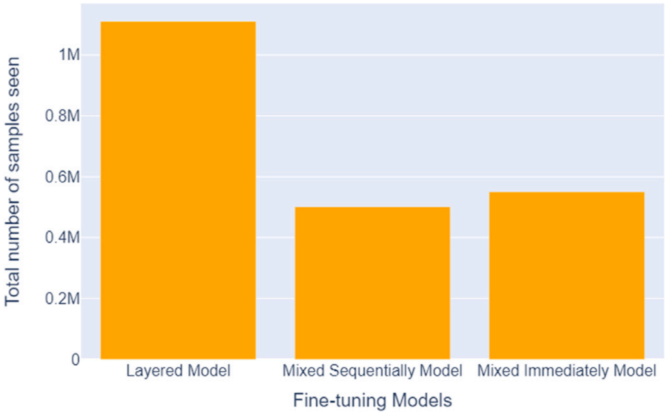


Fig. 9. The total number of samples each model was exposed to during the fine-tuning process. The Layered Model was exposed to more samples than the other two models combined.

distinct training strategies – LM, MSM, and MIM – were employed, each tailored to address unique challenges encountered in the fine-tuning process, such as catastrophic forgetting, overfitting, and underfitting. These strategies varied in the composition and proportions of their training datasets. The evaluation of these models centred on their ability to generate coherent and precise descriptions of unseen football events. Notably, the MIM approach demonstrated remarkable efficiency in learning and adeptness in managing a challenging workload. This highlights the promising potential of simultaneous multi-task learning when utilising compact, open-source language models.

Furthermore, the study underscores the practicality of leveraging consumer-grade hardware for fine-tuning language models with specialised knowledge. It emphasises the importance of adopting customised training approaches and ensuring well-balanced datasets. These findings not only offer practical insights into the utilisation of the new wave of language models but also contribute significantly to NLP applications in sports journalism. They pave the way for broader access to large language models and serve as a valuable resource for future endeavours in this domain.

In summary, the remarks drawn from the deployment of three different training strategies in this paper are as follows:

1. The LM is the most proficient in generating descriptions of in-game football events, primarily due to its slightly superior precision compared to the other two models. This advantage in precision is likely attributable to the model's exposure to a significantly larger training sample during its fine-tuning process (see Fig. 9), encompassing over one million events. This allowed it to be exposed to the full range of possible event permutations multiple times during its fine-tuning.

   (a) The LM fine-tuning strategy produced the most capable final iteration, doing so at a significant cost in terms of data and time resources.
   (b) The LM took eleven rounds of training to reach proficiency, compared to the MSM's seven and the MIM's five.
   (c) The LM was exposed to vastly more samples in each category to end up winning by the slimmest of margins. The amount of data it was exposed to is not proportional to the scale of its victory in the final evaluation.
   (d) A single training round for the LM took a considerably longer amount of time than the other two models due to the much larger amount of data used in each round.
   (e) The LM was continuously plagued with catastrophic forgetting that impeded the progress of the fine-tuning process.

2. The MIM fine-tuning strategy produced the most efficient final results, achieving almost parity with considerably less resource expenditure.
3. The MSM fine-tuning strategy produced the worst final results, highlighting the importance of data diversity in model training.
4. Our study underscores the practicality of employing consumer-grade hardware for fine-tuning language models in sports journalism through various avenues. Firstly, by leveraging readily available and cost-effective hardware, such as standard laptops or desktops, our approach ensures accessibility to a wider range of users within the sports media landscape, including individual journalists and smaller organisations. Secondly, we demonstrate that despite potential hardware limitations, careful optimisation of model architectures and training strategies enables competitive performance levels comparable to more resource-intensive setups. These findings underscore the feasibility and scalability of our approach, offering tangible benefits in terms of efficiency, affordability, and real-world applicability for enhancing sports journalism workflows.

7.2. *Limitations*

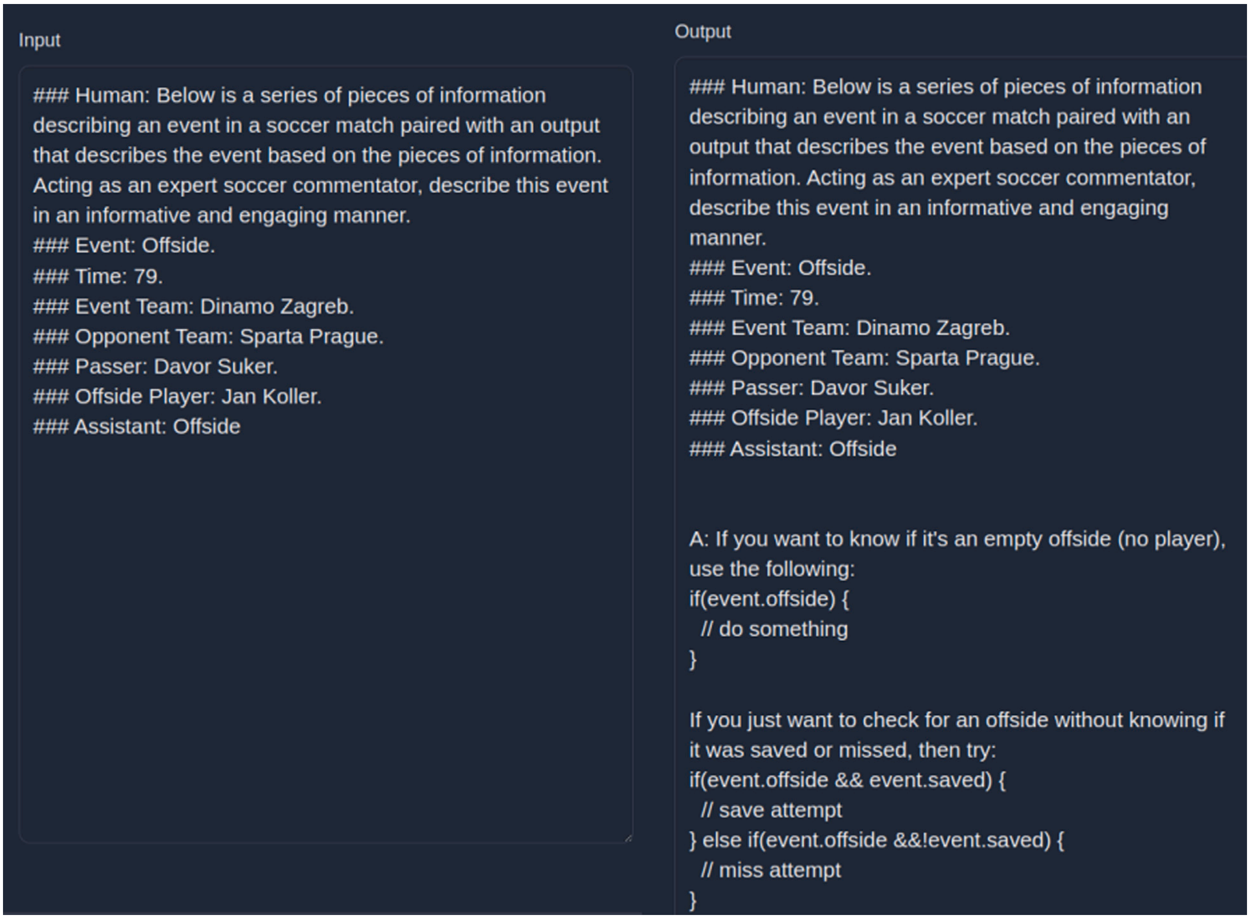
All three models developed in the study share similar limitations and shortcomings. The initial objective of the paper was to refine a Language Learning Model (LLM) to generate football commentary in a manner reminiscent of human commentators. However, it became apparent early in the study that this goal was unattainable due to the severe shortage of appropriate data. There exists a notable absence of datasets that combine raw match data with instances of human commentator descriptions. Without sufficient training data, it is impossible to instruct an LLM to replicate the emotive and vivid descriptions characteristic of human commentators. Consequently, while the models are capable of generating accurate commentary, they fall short of producing engaging commentary.

Although the models can accurately describe events they were trained on, they are entirely incapable of describing events they have not encountered. Football matches can be unpredictable, with occurrences such as pitch invasions, match abandonment, or a manager's expulsion, all of which the models would be unable to depict. Once again, this limitation stems from a lack of relevant data, extending even to relatively common events like own goals. An illustrative example of this scenario is provided below:

```
{"input":
"### Event: Pitch invasion.
### Time: 89.
### Home Team: Liverpool.
### Away Team: Man Utd.
### Game Abandoned?: Yes
### Assistant: Pitch invasion"}

{"output":
"### Assistant: Pitch invasion by Liverpool. Man Utd
won the toss and chose to play."}
```

The LLM-based automated commentary systems can better adapt to the unpredictability of football matches via careful data-curation perspectives. For example, techniques like *Human-in-the-loop Systems* (integrating human supervision or feedback mechanisms), or *Data generation/augmentation* (artificially introducing instances of pitch invasions, match abandonment, or managerial expulsions into the training data) can be options to provide more comprehensive and realistic commentary, even in the presence of rare events.



```
### Human: Below is a series of pieces of information
describing an event in a soccer match paired with an output
that describes the event based on the pieces of information.
Acting as an expert soccer commentator, describe this event
in an informative and engaging manner.
### Event: Offside.
### Time: 79.
### Event Team: Dinamo Zagreb.
### Opponent Team: Sparta Prague.
### Passer: Davor Suker.
### Offside Player: Jan Koller.
### Assistant: Offside
```

```
### Human: Below is a series of pieces of information
describing an event in a soccer match paired with an
output that describes the event based on the pieces of
information. Acting as an expert soccer commentator,
describe this event in an informative and engaging
manner.
### Event: Offside.
### Time: 79.
### Event Team: Dinamo Zagreb.
### Opponent Team: Sparta Prague.
### Passer: Davor Suker.
### Offside Player: Jan Koller.
### Assistant: Offside


A: If you want to know if it's an empty offside (no player),
use the following:
if(event.offside) {
  // do something
}

If you just want to check for an offside without knowing if
it was saved or missed, then try:
if(event.offside && event.saved) {
  // save attempt
} else if(event.offside &&!event.saved) {
  // miss attempt
}
```

Fig. 10. A model writes conditional logic that can check for hypothetical offsides in Java when prompted to describe an offside event.

Bender et al. famously characterised LLMs as “stochastic parrots”, suggesting that they generate natural language based on probability rather than meaning (Bender et al. 2021). This study confirms this observation to a large extent. Despite maintaining consistent conditions, each model generated by this project consistently produces perplexing output, even when it would typically generate accurate descriptions. While the three models developed in this project do not exhibit a higher likelihood of generating nonsensical descriptions compared to other LLMs, there remains the possibility that these models might opt to write Java code instead of describing an offside event, as illustrated in Fig. 10.

In addition to the limitations already discussed, it is important to acknowledge the challenge of real-time implementation of the aimed automated commentary system. Despite the potential benefits of providing real-time commentary, particularly for engaging fans during live football matches, our current system faces constraints that hinder its immediate applicability in this context. One significant barrier is the difficulty in obtaining real-time data from sources such as Opta, especially for minor league teams, which were the primary focus of our study. The lack of access to timely and accurate data poses a substantial obstacle to achieving real-time functionality, limiting the practical applicability of our system in live match scenarios. As such, while our system demonstrates promising capabilities in generating automated commentary from historical data, its real-time implementation remains a goal for future research efforts.

7.3. Further research questions for proposed models

The research conducted in this project opens several avenues for further investigation, each with the potential to advance our understanding and capabilities in NLP and AI language model training. The fine-tuning strategies used in this project still have the potential to be explored.

Optimising the Layered Model Strategy: The LM strategy demands a lot of data. Future research could investigate the effects of capping the number of training samples in a Layered Model. This would address questions such as: Can a similar level of proficiency be achieved with less data? What is the optimal balance between the quantity of training data and model performance?

Enhancing the Mixed Sequentially Model Strategy: The MSM showed promise, but its performance could potentially be improved with a richer dataset. The overfitting suffered as its final iteration crippled the performance level of this model in the final evaluation. Research could focus on expanding the variety and number of goal events in the source datasets, or methods to achieve better fine-tuning results when faced with a scarcity of data. Key questions include: How does the diversity and volume of data impact the model’s ability to learn complex tasks? Is there a threshold for the number of events that optimises learning without leading to overfitting?

Scaling the Mixed Immediately Model Strategy: The most captivating field of study regarding the MIM lies in its ability to engage in simultaneous multi-event learning. Subsequent research endeavours could explore augmenting the number of events employed in each training iteration. This prompts inquiries such as: What is the upper limit of events that can be inclusively integrated without diminishing the model’s efficacy? How does the model’s learning process adjust to a heightened diversity of concurrent inputs?

7.4. Future directions

The findings in this paper have broader implications beyond football commentary generation. For instance, in real-world scenarios like

Table A.6
The LM quantitative performance results.

	Attempts				*Goals*				*Subs*				*Offsides*				*Cards*				*Corners*				*Fouls*			
	DC	*R1*	*R2*	*RL*	*DC*	*R1*	*R2*	*RL*	*DC*	*R1*	*R2*	*RL*	*DC*	*R1*	*R2*	*RL*	*DC*	*R1*	*R2*	*RL*	*DC*	*R1*	*R2*	*RL*	*DC*	*R1*	*R2*	*RL*
Attempts	64	0.97	0.96	0.97	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Goals	23	0.92	0.87	0.92	67	0.98	0.97	0.98	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Subs	18	0.92	0.86	0.92	53	0.95	0.93	0.95	98	1.00	1.00	1.00	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
R: 1	71	0.94	0.96	0.97	70	0.95	0.97	0.98	95	0.99	0.99	0.99	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Offsides	67	0.98	0.97	0.98	65	0.94	0.93	0.94	6	0.57	0.41	0.57	85	0.99	0.98	0.99	–	–	–	–	–	–	–	–	–	–	–	–
R: 2	70	0.98	0.96	0.98	70	0.98	0.97	0.98	94	0.99	0.98	0.99	88	0.99	0.98	0.99	–	–	–	–	–	–	–	–	–	–	–	–
Cards	69	0.95	0.96	0.98	65	0.92	0.94	0.96	84	0.95	0.92	0.95	73	0.96	0.95	0.94	72	0.96	0.94	0.96	–	–	–	–	–	–	–	–
Corners	73	0.98	0.97	0.98	63	0.97	0.95	0.97	87	0.98	0.97	0.98	88	0.97	0.96	0.97	72	0.96	0.96	0.96	99	1.00	1.00	1.00	–	–	–	–
R: 3	72	0.98	0.96	0.98	70	0.98	0.90	0.98	94	0.99	0.99	0.99	88	0.99	0.98	0.99	72	0.96	0.94	0.96	99	1.00	1.00	1.00	–	–	–	–
Fouls	65	0.97	0.95	0.97	63	0.97	0.96	0.97	96	1.00	0.99	1.00	77	0.92	0.89	0.90	73	0.96	0.94	0.96	99	1.00	1.00	1.00	99	1.00	1.00	1.00
R: 4	73	0.98	0.96	0.98	70	0.99	0.97	0.99	93	0.99	0.98	0.99	86	0.96	0.98	0.96	74	1.00	0.94	1.00	99	1.00	1.00	1.00	100	1.00	1.00	1.00

Table A.7
The MSM quantitative performance results.

	Attempts				*Goals*				*Subs*				*Offsides*				*Cards*				*Corners*				*Fouls*			
	DC	*R1*	*R2*	*RL*	*DC*	*R1*	*R2*	*RL*	*DC*	*R1*	*R2*	*RL*	*DC*	*R1*	*R2*	*RL*	*DC*	*R1*	*R2*	*RL*	*DC*	*R1*	*R2*	*RL*	*DC*	*R1*	*R2*	*RL*
Attempts	49	0.97	0.94	0.96	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Goals	65	0.98	0.96	0.98	70	0.98	0.97	0.97	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Subs	66	0.98	0.96	0.98	69	0.98	0.96	0.98	78	0.98	0.96	0.98	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–	–
Offsides	63	0.97	0.95	0.97	73	0.98	0.97	0.98	85	0.98	0.97	0.98	88	0.99	0.98	0.99	–	–	–	–	–	–	–	–	–	–	–	–
Cards	71	0.98	0.96	0.98	65	0.97	0.96	0.97	86	0.98	0.98	0.98	85	0.97	0.98	0.99	72	0.96	0.93	0.96	–	–	–	–	–	–	–	–
Corners	71	0.96	0.96	0.98	70	0.97	0.97	0.97	92	0.98	0.98	0.99	88	0.99	0.99	0.99	71	0.96	0.93	0.96	98	1.00	1.00	1.00	–	–	–	–
Fouls	70	0.98	0.97	0.98	32	0.94	0.91	0.94	90	0.98	0.97	0.98	88	0.99	0.98	0.99	72	0.96	0.94	0.96	98	1.00	1.00	1.00	100	1.00	1.00	1.00

Table A.8
The MIM quantitative performance results.

	Attempts				*Goals*				*Subs*				*Offsides*				*Cards*				*Corners*				*Fouls*			
	DC	*R1*	*R2*	*RL*	*DC*	*R1*	*R2*	*RL*	*DC*	*R1*	*R2*	*RL*	*DC*	*R1*	*R2*	*RL*	*DC*	*R1*	*R2*	*RL*	*DC*	*R1*	*R2*	*RL*	*DC*	*R1*	*R2*	*RL*
One	59	0.97	0.95	0.97	62	0.97	0.95	0.97	81	0.98	0.95	0.98	73	0.98	0.97	0.98	75	0.96	0.94	0.96	99	1.00	1.00	1.00	54	0.83	0.78	0.83
Two	64	0.97	0.95	0.97	61	0.97	0.96	0.97	88	0.99	0.98	0.99	87	0.99	0.98	0.99	68	0.96	0.93	0.96	99	1.00	1.00	1.00	98	1.00	1.00	1.00
Three	68	0.98	0.96	0.98	66	0.97	0.97	0.97	96	1.00	0.99	1.00	83	0.98	0.98	0.98	72	0.96	0.94	0.96	99	1.00	1.00	1.00	98	1.00	0.99	1.00
Four	71	0.98	0.96	0.98	66	0.98	0.97	0.98	94	0.99	0.99	0.99	87	0.99	0.98	0.99	69	0.96	0.93	0.96	99	1.00	1.00	1.00	100	1.00	1.00	1.00
Five	73	0.98	0.97	0.98	63	0.97	0.96	0.97	94	0.99	0.98	0.99	86	0.99	0.98	0.99	71	0.96	0.94	0.96	99	1.00	1.00	1.00	100	1.00	1.00	1.00

automated news reporting, social media content moderation, or even interactive entertainment, the efficiency and adaptability of LLMs are paramount. The insights gained from this research could guide the development of more resource-efficient models in these areas, balancing the need for precision with practical constraints.

Moreover, this research opens avenues for future exploration in the field of LLMs, particularly in understanding the balance between data diversity, training efficiency, and model accuracy. As LLMs continue to evolve, the lessons learned here could inform researchers and enthusiasts how to fine-tune a model that is capable of handling a wider array of tasks with greater efficiency and effectiveness.

While this paper's quest for the most proficient model for generating football match commentary has yielded valuable insight, it also highlights the dynamic nature of machine learning and the ongoing need to balance various factors for optimal model fine-tuning and performance.

CRediT authorship contribution statement

Alec Cook: Writing – review & editing, Visualization, Validation, Software, Resources, Methodology, Formal analysis, Data curation, Conceptualization. **Oktay Karakuş:** Writing – review & editing, Writing – original draft, Visualization, Validation, Supervision, Project administration, Methodology, Formal analysis, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Appendix. Model quantitative result tables

This appendix presents detailed quantitative performance results of LM, MSM and MIM models in Tables A.6, A.7 and A.8, respectively.

References

[1] B. Milanovic, Globalization and goals: Does soccer show the way? Rev. Int. Political Econ. 12 (5) (2005) 829–850, Publisher: Taylor & Francis, Ltd. URL https://www.jstor.org/stable/25124053.

[2] F. Chisari, When football went global: Televising the 1966 world cup, Hist. Soc. Res. (Historische Sozialforschung) 31 (1 (115)) (2006) 42–54, Publisher: GESIS - Leibniz-Institute for the Social Sciences, Center for Historical Social Research URL https://www.jstor.org/stable/20762101.

[3] Deloitte's Sport Business Group, Annual Review of Football FInance 2023, Tech. Rep., Deloitte, London, 2023, URL https://www2.deloitte.com/content/dam/Deloitte/uk/Documents/sports-business-group/deloitte-uk-annual-review-of-football-finance-2023.pdf.

[4] Mordor Intelligence, Football Market Size & Share Analysis - Growth Trends & Forecasts (2023 -2028), Tech. Rep., Mordor Intelligence, Hyderabad, 2023, URL https://www.mordorintelligence.com/industry-reports/football-market.

[5] Nielsen, Fans are Changing the Game: 2022 Global Sports Marketing Report, Tech. rep, The Nielsen Company, 2022, p. 29, URL https://nielsensports.com/wp-content/uploads/2022/02/Nielsen-Sports-Fans-are-changing-the-game-1.pdf.

[6] C. Henrys, How Wrexham AFC has grown as we mark the second anniversary of the Club's takeover, 2023, URL https://www.wrexhamafc.co.uk/news/2023/february/how-wrexham-afc-has-grown-as-we-mark-the-second-anniversary-of-the-clubs-takeover/.

[7] B. Annamalai, M. Yoshida, S. Varshney, A.A. Pathak, P. Venugopal, Social media content strategy for sport clubs to drive fan engagement, J. Retail. Consum. Serv. 62 (2021) 102648.

[8] A. Ram, R. Prasad, C. Khatri, A. Venkatesh, R. Gabriel, Q. Liu, J. Nunn, B. Hedayatnia, M. Cheng, A. Nagar, et al., Conversational ai: The science behind the alexa prize, 2018, arXiv preprint arXiv:1801.03604.

[9] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, I. Sutskever, et al., Language models are unsupervised multitask learners, OpenAI Blog 1 (8) (2019) 9.

[10] T. Brown, B. Mann, N. Ryder, M. Subbiah, J.D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, et al., Language models are few-shot learners, Adv. Neural Inf. Process. Syst. 33 (2020) 1877–1901.

[11] H. Naveed, A.U. Khan, S. Qiu, M. Saqib, S. Anwar, M. Usman, N. Barnes, A. Mian, A comprehensive overview of large language models, 2023, arXiv preprint arXiv:2307.06435.

[12] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, Ł. Kaiser, I. Polosukhin, Attention is all you need, Adv. Neural Inf. Process. Syst. 30 (2017).

[13] A. Romero, GPT-3 — A Complete Overview, 2021, Medium URL https://towardsdatascience.com/gpt-3-a-complete-overview-190232eb25fd.

[14] J. Yosinski, J. Clune, Y. Bengio, H. Lipson, How transferable are features in deep neural networks? Adv. Neural Inf. Process. Syst. 27 (2014).

[15] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar, et al., Llama: Open and efficient foundation language models, 2023, arXiv preprint arXiv:2302.13971.

[16] M. Carlson, Why LLaMa is a big deal, 2023, Hackaday URL https://hackaday.com/2023/03/22/why-llama-is-a-big-deal/.

[17] S. Gao, A.K. Gao, On the origin of LLMs: An evolutionary tree and graph for 15,821 large language models, 2023, arXiv preprint arXiv:2307.09793.

[18] Y. Cui, Z. Yang, X. Yao, Efficient and effective text encoding for Chinese llama and alpaca, 2023, arXiv preprint arXiv:2304.08177.

[19] B. Svendsen, S. Kadry, A dataset for recognition of Norwegian sign language, Int. J. Math. Stat. Comput. Sci. 2 (2024).

[20] M. Bradley, Growing Your Club: Practical & Proven Ideas From Clubs Like Yours, Tech. Rep., The FA, 2023, URL https://www.manchesterfa.com/leagues-and-clubs/marketing/growing-your-club.

[21] P. Buckingham, Fear and losses in the National League: 'Clubs of our size generally lose around £1m a year', 2023, The Athletic URL https://theathletic.com/4341087/2023/03/25/fear-and-losses-in-the-national-league-clubs-of-our-size-generally-lose-around-1m-a-year/.

[22] M. Asada, M.M. Veloso, M. Tambe, I. Noda, H. Kitano, G.K. Kraetzschmar, Overview of robocup-98, AI Mag. 21 (1) (2000) 9.

[23] K. Tanaka, H. Nakashima, I. Noda, K. Hasida, I. Frank, H. Matsubara, MIKE: An automatic commentary system for soccer, in: Proceedings International Conference on Multi Agent Systems (Cat. No. 98EX160), IEEE, 1998, pp. 285–292.

[24] D. Voelz, E. André, G. Herzog, T. Rist, Rocco: A robocup soccer commentator system, in: M. Asada, H. Kitano (Eds.), RoboCup-98: Robot Soccer World Cup II, in: Lecture Notes in Computer Science, Springer, Berlin, Heidelberg, 1999, pp. 50–60, http://dx.doi.org/10.1007/3-540-48422-1_4.

[25] K. Binsted, S. Luke, Character design for soccer commentary, in: G. Goos, J. Hartmanis, J. Van Leeuwen, M. Asada, H. Kitano (Eds.), RoboCup-98: Robot Soccer World Cup II, in: Lecture Notes in Computer Science, vol. 1604, Springer Berlin Heidelberg, Berlin, Heidelberg, 1999, pp. 22–33, http://dx.doi.org/10.1007/3-540-48422-1_2, URL http://link.springer.com/10.1007/3-540-48422-1_2.

[26] K. Tanaka-Ishii, K. Hasida, I. Noda, Reactive content selection in the generation of real-time soccer commentary, in: COLING 1998 Volume 2: The 17th International Conference on Computational Linguistics, 1998.

[27] D.L. Chen, R.J. Mooney, Learning to sportscast: a test of grounded language acquisition, in: Proceedings of the 25th International Conference on Machine Learning, 2008, pp. 128–135.

[28] R.J. Kate, R.J. Mooney, Learning language semantics from ambiguous supervision, in: Proceedings of the 22nd National Conference on Artificial Intelligence - Volume 1, AAAI '07, AAAI Press, Vancouver, British Columbia, Canada, 2007, pp. 895–900.

[29] Y.W. Wong, R. Mooney, Learning synchronous grammars for semantic parsing with lambda calculus, in: Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics, 2007, pp. 960–967.

[30] Y. Taniguchi, Y. Feng, H. Takamura, M. Okumura, Generating live soccer-match commentary from play data, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 33, No. 01, 2019, pp. 7096–7103.

[31] Q. Zhou, N. Yang, F. Wei, M. Zhou, Selective encoding for abstractive sentence summarization, 2017, arXiv preprint arXiv:1704.07073.

[32] S. Tokui, R. Okuta, T. Akiba, Y. Niitani, T. Ogawa, S. Saito, S. Suzuki, K. Uenishi, B. Vogel, H. Yamazaki Vincent, Chainer: A deep learning framework for accelerating the research cycle, in: Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, 2019, pp. 2002–2011.

[33] Preferred Networks Migrates its Deep Learning Research Platform to PyTorch, Preferred Networks, Inc., 2019, URL https://www.preferred.jp/en/news/pr20191205/.

[34] D. Ciresan, U. Meier, J. Masci, L.M. Gambardella, J. Schmid-huber, High performance convolutional neural networks for image classification, in: Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence, pp. 1237–1242.

[35] A. Krizhevsky, I. Sutskever, G.E. Hinton, Imagenet classification with deep convolutional neural networks, Adv. Neural Inf. Process. Syst. 25 (2012).

[36] A. Raganato, J. Tiedemann, An analysis of encoder representations in transformer-based machine translation, in: Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP, The Association for Computational Linguistics, 2018.

[37] A.L. Brown, M.J. Kane, Preschool children can learn to transfer: Learning to learn and learning from example, Cogn. Psychol. 20 (4) (1988) 493–523.

[38] A. Malte, P. Ratadiya, Evolution of transfer learning in natural language processing, 2019, arXiv preprint arXiv:1910.07370.

[39] J. Howard, S. Ruder, Universal language model fine-tuning for text classification, 2018, arXiv preprint arXiv:1801.06146.

[40] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, Bert: Pre-training of deep bidirectional transformers for language understanding, 2018, arXiv preprint arXiv:1810.04805.

[41] M. Yang, A survey on few-shot learning in natural language processing, in: 2021 International Conference on Artificial Intelligence and Electromechanical Automation, AIEA, IEEE, 2021, pp. 294–297.

[42] H.-y. Lee, S.-W. Li, N.T. Vu, Meta learning for natural language processing: A survey, 2022, arXiv preprint arXiv:2205.01500.

[43] S. Zhang, S. Roller, N. Goyal, M. Artetxe, M. Chen, S. Chen, C. Dewan, M. Diab, X. Li, X.V. Lin, et al., Opt: Open pre-trained transformer language models, 2022, arXiv preprint arXiv:2205.01068.

[44] C.R. Wolfe, The History of Open-Source LLMs: Early Days, 2023, Deep (Learning) Focus URL https://cameronrwolfe.substack.com/p/the-history-of-open-source-llms-early,

[45] T.L. Scao, A. Fan, C. Akiki, E. Pavlick, S. Ilić, D. Hesslow, R. Castagné, A.S. Luccioni, F. Yvon, et al., Bloom: A 176b-parameter open-access multilingual language model, 2022, arXiv preprint arXiv:2211.05100.

[46] Geronimo, From Transcripts to AI Chat: An Experiment with the Lex Fridman Podcast, 2023, Medium URL https://medium.com/@geronimo7/from-transcripts-to-ai-chat-an-experiment-with-the-lex-fridman-podcast-3248d216ec16,

[47] C.R. Wolfe, LLaMA: LLMs for Everyone!, 2023, Medium URL https://towardsdatascience.com/llama-llms-for-everyone-724e737835be,

[48] R. Taori, I. Gulrajani, T. Zhang, Y. Dubois, X. Li, C. Guestrin, P. Liang, T.B. Hashimoto, Alpaca: A strong, replicable instruction-following model, vol. 3, (6) Stanford Center for Research on Foundation Models, 2023, p. 7, https://crfm.stanford.edu/2023/03/13/alpaca.html.

[49] E. Beeching, Y. Belkada, L.V. Werra, S. Mangrulkar, L. Tunstall, K. Rasul, Fine-tuning 20B LLMs with RLHF on a 24GB consumer GPU, 2023, Huggingface Blog URL https://huggingface.co/blog/trl-peft.

[50] H. Liu, D. Tam, M. Muqeeth, J. Mohta, T. Huang, M. Bansal, C.A. Raffel, Few-shot parameter-efficient fine-tuning is better and cheaper than in-context learning, Adv. Neural Inf. Process. Syst. 35 (2022) 1950–1965.

[51] E.J. Hu, Y. Shen, P. Wallis, Z. Allen-Zhu, Y. Li, S. Wang, L. Wang, W. Chen, Lora: Low-rank adaptation of large language models, 2021, arXiv preprint arXiv:2106.09685.

[52] B.P. Khushboo Rathi, Llama 2: Efficient Fine-tuning Using Low-Rank Adaptation (LoRA) on Single GPU | Dell Technologies Info Hub, 2023, Dell Technologies URL https://infohub.delltechnologies.com/p/llama-2-efficient-fine-tuning-using-low-rank-adaptation-lora-on-single-gpu/,

[53] R. Kemker, M. McClure, A. Abitino, T. Hayes, C. Kanan, Measuring catastrophic forgetting in neural networks, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 32, No. 1, 2018.

[54] Z. Hu, Y. Lan, L. Wang, W. Xu, E.-P. Lim, R.K.-W. Lee, L. Bing, S. Poria, LLM-adapters: An adapter family for parameter-efficient fine-tuning of large language models, 2023, arXiv preprint arXiv:2304.01933.

[55] T. Dettmers, A. Pagnoni, A. Holtzman, L. Zettlemoyer, Qlora: Efficient finetuning of quantized llms, 2023, arXiv preprint arXiv:2305.14314.

[56] Y. Bondarenko, M. Nagel, T. Blankevoort, Understanding and overcoming the challenges of efficient transformer quantization, 2021, arXiv preprint arXiv:2109.12948.

[57] S.J. Kwon, J. Kim, J. Bae, K.M. Yoo, J.-H. Kim, B. Park, B. Kim, J.-W. Ha, N. Sung, D. Lee, Alphatuning: Quantization-aware parameter-efficient adaptation of large-scale pre-trained language models, 2022, arXiv preprint arXiv:2210.03858.

[58] C. Tao, L. Hou, W. Zhang, L. Shang, X. Jiang, Q. Liu, P. Luo, N. Wong, Compression of generative pre-trained language models via quantization, 2022, arXiv preprint arXiv:2203.10705.

[59] Y. Chai, J. Gkountouras, G.G. Ko, D. Brooks, G.-Y. Wei, INT2. 1: Towards fine-tunable quantized large language models with error correction through low-rank adaptation, 2023, arXiv preprint arXiv:2306.08162.

[60] Y. Belkada, T. Dettmers, A. Pagnoni, S. Gugger, S. Mangrulkar, Making LLMs even more accessible with bitsandbytes, 4-bit quantization and QLoRA, 2023, Huggingface Blog URL https://huggingface.co/blog/4bit-transformers-bitsandbytes.

[61] M. Manohar, Understanding LoRA and QLoRA — The Powerhouses of Efficient Finetuning in Large Language Models, 2023, Medium URL https://medium.com/@gitlostmurali/understanding-lora-and-qlora-the-powerhouses-of-efficient-finetuning-in-large-language-models-7ac1adf6c0cf,

[62] N. Gupta, S. Mujumdar, H. Patel, S. Masuda, N. Panwar, S. Bandyopadhyay, et al., Data quality for machine learning tasks, in: Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining, Virtual Event Singapore, ACM, 2021, (Accessed 2021).

[63] C. Love, BBC Football Commentary Data – Webscraping, 2015, Sciolistic Ramblings URL https://sciolisticramblings.wordpress.com/2015/08/24/bbc-football-commentary-data-webscraping/.

[64] A. Secareanu, Football Events, 2023, Kaggle URL https://www.kaggle.com/datasets/secareanualin/football-events.

[65] J. Cheung, GuanacoDataset, 2023, URL https://huggingface.co/datasets/JosephusCheung/GuanacoDataset.

[66] J. Heigham, R.A. Croker (Eds.), Qualitative Research in Applied Linguistics: A Practical Introduction, Palgrave Macmillan, Houndmills, Basingstoke, Hampshire [England] ; New York, 2009, OCLC ocn317114506.

[67] S. Ruder, An overview of multi-task learning in deep neural networks, 2017, arXiv preprint arXiv:1706.05098.

[68] K. Gurney, An Introduction to Neural Networks, second ed., Routledge, London, 1997.

[69] C.C. Aggarwal, et al., Neural Networks and Deep Learning, vol. 10, (978) Springer, 2018, p. 3.

[70] L. Floridi, AI as agency without intelligence: on ChatGPT, large language models, and other generative models, Philos. Technol. 36 (1) (2023) 15.

[71] K. Ericsson, N. Charness, Expert Performance: Its Structure and Acquisition, Am. Psychol. 49 (1994) 725–747, http://dx.doi.org/10.1037/0003-066X.49.8.725.

[72] A. Robins, Catastrophic Forgetting, Rehearsal and Pseudorehearsal, Connect. Sci. J. Neural Comput. Artif. Intell. Cognit. Res. 7 (2) (1995) 123–146, http://dx.doi.org/10.1080/09540099550039318, URL https://www.tandfonline.com/doi/full/10.1080/09540099550039318.

[73] S. Kotsiantis, D. Kanellopoulos, P. Pintelas, Handling imbalanced datasets: A review, GESTS Int. Trans. Comput. Sci. Eng. 30 (2006).

[74] J. Neyman, On the Two Different Aspects of the Representative Method: the Method of Stratified Sampling and the Method of Purposive Selection, in: S. Kotz, N.L. Johnson (Eds.), Breakthroughs in Statistics: Methodology and Distribution, in: Springer Series in Statistics, Springer, New York, NY, 1992, pp. 123–150, http://dx.doi.org/10.1007/978-1-4612-4380-9_12, URL.

[75] S. Tyagi, S. Mittal, Sampling Approaches for Imbalanced Data Classification Problem in Machine Learning, in: P.K. Singh, A.K. Kar, Y. Singh, M.H. Kolekar, S. Tanwar (Eds.), Proceedings of ICRIC 2019, in: Lecture Notes in Electrical Engineering, Springer International Publishing, Cham, 2020, pp. 209–221, http://dx.doi.org/10.1007/978-3-030-29407-6_17.

[76] N.V. Chawla, K.W. Bowyer, L.O. Hall, W.P. Kegelmeyer, SMOTE: Synthetic minority over-sampling technique, J. Artificial Intelligence Res. 16 (2002) 321–357, http://dx.doi.org/10.1613/jair.953, URL https://www.jair.org/index.php/jair/article/view/10302.

[77] B. Tabibian, U. Upadhyay, A. De, A. Zarezade, B. Schölkopf, M. Gomez-Rodriguez, Enhancing human learning via spaced repetition optimization, Proc. Natl. Acad. Sci. USA 116 (10) (2019) 3988–3993, http://dx.doi.org/10.1073/pnas.1815156116.

[78] N.J. Cepeda, N. Coburn, D. Rohrer, J.T. Wixted, M.C. Mozer, H. Pashler, Optimizing distributed practice: Theoretical analysis and practical implications, Exp. Psychol. 56 (4) (2009) 236–246, http://dx.doi.org/10.1027/1618-3169.56.4.236, URL https://econtent.hogrefe.com/doi/10.1027/1618-3169.56.4.236.

[79] A. Cook, 2023, URL https://huggingface.co/IronChef.

[80] C.-Y. Lin, ROUGE: A package for automatic evaluation of summaries, in: Text Summarization Branches Out, Association for Computational Linguistics, Barcelona, Spain, 2004, pp. 74–81, URL https://aclanthology.org/W04-1013.

[81] C. Goutte, E. Gaussier, A Probabilistic Interpretation of Precision, Recall and F-Score, with Implication for Evaluation, 2005.

[82] HuggingFace, Transformers.

[83] T. Dettmers, bitsandbytes, 2021, URL https://github.com/TimDettmers/bitsandbytes original-date: 2021-06-04T00:10:34Z.

[84] HuggingFace, accelerate, 2020, HuggingFace URL https://github.com/huggingface/accelerate.

[85] HuggingFace, PEFT, 2023, Hugging Face URL https://github.com/huggingface/peft original-date: 2022-11-25T03:51:09Z.

[86] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, S. Chintala, H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché Buc, E. Fox, R. Garnett, PyTorch: An imperative style, high-performance deep learning library, in: Advances in Neural Information Processing Systems, vol. 32, Curran Associates, Inc., 2019, pp. 8024–8035, original-date: 2016-08-13T05:26:41Z URL https://github.com/pytorch/pytorch.

[87] NVIDIA, CUDA Toolkit, NVIDIA URL https://developer.nvidia.com/cuda-toolkit,

[88] S. Gururangan, A. Marasović, S. Swayamdipta, K. Lo, I. Beltagy, D. Downey, N.A. Smith, Don't Stop Pretraining: Adapt Language Models to Domains and Tasks, 2020, arXiv URL http://arxiv.org/abs/2004.10964 [cs].

[89] M. McCloskey, N.J. Cohen, Catastrophic Interference in Connectionist Networks: The Sequential Learning Problem, in: Psychology of Learning and Motivation, vol. 24, Elsevier, 1989, pp. 109–165, http://dx.doi.org/10.1016/S0079-7421(08)60536-8, URL https://linkinghub.elsevier.com/retrieve/pii/S0079742108605368.