Large Language Models in Military Planning and War-gaming: A Literature Review

Stephen Moore
st690445@ucf.edu
University of Central Florida
Orlando, Florida, USA

Figure 1: Italian Army Maj. Stefano Catania (right) and U.S. Army Maj. Keith Weaver discuss potential locations of their troops during a game of "Land power" as U.S. Army Maj. Colin Bair (back left) observes at the U.S. Army Command and Staff College, Fort Leavenworth, KS. 2018. [10]

Abstract

This literature review explores the potential of Large Language Models (LLMs) as a solution to address the war-gaming capability gap in military planning. We examine the key features of LLMs and review recent research in two critical areas: documenting critical events in simulations and controlling macro actions for strategic decision-making. By synthesizing these findings, we evaluate the feasibility of integrating LLMs into military planning processes to enhance course of action (COA) analysis and war-gaming capabilities.

CAP-6671 '24, August 19 – December 05, 2024, Orlando, FL

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM
https://doi.org/XXXXXXX.XXXXXXX

CCS Concepts

• **Computing methodologies** → **Interest point and salient region detections**; **Simulation types and techniques**; **Neural networks**; • **Applied computing** → **Military**.

Keywords

War-Gaming, Course of Action (COA) Analysis, Large Language Model (LLM)

ACM Reference Format:
Stephen Moore. 2018. Large Language Models in Military Planning and War-gaming: A Literature Review. In *Proceedings of Intelligent Systems: Robots, Agents, and Humans (CAP-6671 '24).* ACM, New York, NY, USA, 5 pages. https://doi.org/XXXXXXX.XXXXXXX

1 Introduction

Within the planning for military operations, course of action (COA) analysis, also known as war-gaming, is a critical step for military planners because it identifies vulnerabilities and critical decisions within each COA. This process is often executed in an analog medium in which the planners form two sides, one for the friendly forces and the other for the enemy, and then conduct a turn-based

simulation of each COA on a depiction of the area of operations. Through each turn of the COA, one side states their action and then waits for the opposing side to state their counteraction. Each turn's action and counteraction are recorded and adjudicated by a designated planner using a correlation of forces and means (COFM) calculator that predicts casualty and consumption rates given the type of operation. Although it is a critical step, the effectiveness of this analysis is heavily dependent on the experience and expertise of the planners involved, particularly in representing opposing forces.

The challenges in COA analysis are further compounded by the anticipated nature of future conflicts, specifically large-scale combat operations (LSCO). Unlike recent counter-insurgency operations, LSCO is expected to result in higher casualty rates and more frequent destruction of command posts. This scenario presents a significant risk of inexperienced personnel being thrust into critical planning roles, potentially leading to poorly devised operations.

This literature review aims to explore the potential of Large Language Models (LLMs) as a solution to address the anticipated war-gaming capability gap in military planning. I will examine the key features of LLMs and review recent research in two critical areas: documenting critical events in simulations and controlling macro actions for strategic decision-making. By synthesizing these findings, I seek to evaluate the feasibility of integrating LLMs into the military planning process to enhance war-gaming capabilities.

2 Features of LLMs

Before delving into current research, it is crucial to establish a baseline understanding of LLMs. In essence, an LLM is a machine learning model trained to generate text-based responses from text-based prompts. While there were predecessors capable of similar tasks, LLMs stand out due to three distinguishing features: size, architecture, and versatility [3].

2.1 Size

The scale of LLMs, both in terms of training data and model parameters, sets them apart from the earlier models. For instance, the BERT model, an early breakthrough in language modeling, was trained on approximately 3.3 billion words (about 16 GB of text data), resulting in 110-340 million parameters [2]. In contrast, GPT-2, one of the first widely recognized LLMs, used a dataset nearly triple this size (about 40 GB of text data), with 117 million to 1.5 billion parameters [11]. Subsequent LLM releases have continued this trend of growth in both dataset size and parameter count. The increased size of LLMs contributes significantly to their performance and capabilities. According to Brown et al. [1], larger models with more parameters can capture and utilize a broader range of knowledge and patterns from the training data. This allows them to generate more contextually appropriate and diverse responses to prompts, as well as demonstrate improved performance across a wide range of tasks without task-specific fine-tuning.

2.2 Architecture

LLMs employ Transformer architectures, which offer several advantages over the previously used Long Short-Term Memory (LSTM) networks [5]. Key benefits include:

- Parallel processing: Transformers can conduct parallel processing due to their self-attention mechanism, which allows them to process all input tokens simultaneously rather than sequentially [13]. This parallel computation is possible because the self-attention operation can be implemented as matrix multiplications, which are highly efficient on modern hardware like GPUs. As a result, Transformers can process long sequences much faster than recurrent architectures like LSTMs.
- Improved handling of long-range dependencies: Transformers excel at handling long-range dependencies because their self-attention mechanism allows each token to directly attend to every other token in the sequence, regardless of their distance [13]. This direct connection enables the model to capture relationships between distant parts of the input without the information degradation that occurs in sequential processing of Recurrent Neural Networks (RNNs) and LSTMs. As a result, Transformers can maintain context over much longer sequences, which is crucial for understanding and generating coherent text.
- Self-attention mechanism: This allows the model to weigh the importance of different parts of the input when producing each part of the output. The self-attention mechanism works by computing attention weights for each token with respect to all other tokens in the input sequence [13]. For each token, the model calculates query, key, and value vectors. The attention weights are then computed by comparing the query of one token with the keys of all tokens, determining how much focus should be placed on other parts of the input when encoding that specific token. These weights are used to create a weighted sum of the value vectors, producing a context-aware representation for each token. This process allows the model to dynamically focus on relevant parts of the input, regardless of their position in the sequence.

2.3 Versatility

A consequence of their size and architecture is the remarkable versatility of LLMs. They can perform a wide range of tasks with minimal fine-tuning or retraining. Recent developments have even expanded LLMs' capabilities to process and produce images by combining the Transformer architectures with various machine learning image and audio processing techniques [3].

2.4 Relevance to Military Planning

The features of LLMs make them particularly promising for application in military planning and war-gaming. Their ability to process and generate large amounts of complex, context-dependent information aligns well with the needs of COA analysis. The versatility of LLMs suggests they could potentially adapt to various scenarios and operational contexts, while their advanced architecture could enable rapid processing of multiple factors simultaneously, mirroring the complex decision-making environment of military operations.

3 Related Work

Current research relevant to the application of LLMs in military planning and war-gaming can be grouped into two main areas: methods for documenting critical events in games and methods for controlling macro actions for agents' strategies within games. These areas correspond closely to the essential requirements for successful COA analysis in military planning.

3.1 Documenting Critical Events

The implementation of LLMs has significantly enhanced the feasibility of documenting critical events in games and simulations. While pre-LLM approaches demonstrated some success, they were limited in their ability to capture the full complexity and context of game events. LLMs, with their vast knowledge base and ability to understand and generate natural language, have expanded the possibilities for more comprehensive and nuanced event documentation. Below is a summary of the surveyed literature that illustrates the limitations of pre-LLM approaches and the possibilities of recent LLM approaches.

3.1.1 Pre-LLM Approaches. Researchers at the Georgia Institute of Technology found that basic machine learning methods were insufficient for capturing adequate explanations of a game's runtime events. In their experiments, the Random Forest and K-Nearest Neighbor algorithms achieved accuracy rates of only 37% and 42% respectively in identifying and describing key game events. These methods struggled particularly with complex or rare events, often misclassifying them or providing overly simplistic descriptions that lacked context. They concluded that automating real-time logging or commentary for games would require deep neural networks [4].

The studies from the University of Kentucky and Japanese universities recognized that achieving human-like performance in event documentation required a deeper understanding of context, which could be provided by incorporating additional modalities such as audio and structured numerical data [8] [7]. They found that multi-modal input improved event identification accuracy by 15-20% compared to single-modality approaches. Both research groups emphasized the critical role of domain experts in identifying and labeling critical events for training data. However, they noted that this process was highly time-consuming and costly, with experts spending an average of 20-30 hours per 100 minutes of game-play footage to accurately label all significant events. To handle the increased data size, these works used sequence-to-sequence deep neural networks, specifically LSTM networks. However, both studies identified limitations in using LSTMs for processing extended sequences of data.

3.1.2 LLM-Based Approaches. Addressing the limitations of previous approaches, researchers at California Polytechnic State University demonstrated the potential of LLMs in generating real-time audio commentary for the game League of Legends [12]. Their approach combined neural networks for visual feature detection, ChatGPT for generating text-based descriptions of key events, and FakeYou for converting text to audio commentary. This study achieved a 77.14% accuracy in identifying key events and received high marks for commentary richness in qualitative evaluations. Given their generally successful initial results, Renella & Eger (2023) echoed the conclusion that domain experts are still required in the pre-processing phase. They noted that the accuracy of their pipeline was limited by the tedious manual processing of training game-play footage, which remained a bottleneck in scaling up the system. The researchers estimated that for every hour of game-play footage used for training, approximately 3-4 hours of expert annotation time was required to achieve optimal performance.

3.2 Controlling Macro Actions for Strategic Decision-Making

Two recent studies have demonstrated the potential of LLMs in generating strategic decisions in gaming environments, which could have significant implications for military planning simulations. These two studies differ significantly in their approach to decision-making timescales. Ma et al. (2023) focused on real-time decision-making in StarCraft II, which requires rapid, continuous action generation [9]. In contrast, Hu et al. (2024) explored turn-based decision-making in Pokémon, allowing for more deliberate, discrete action choices [6]. The computational requirements of each method largely dictated their approach. Real-time decision-making necessitates more efficient, streamlined processing to meet the game's time constraints, often requiring techniques like action pruning or hierarchical decision-making. Turn-based games, however, allow for more computationally intensive approaches, enabling deeper analysis of game states and potential future outcomes.

3.2.1 Real-Time Decision-Making. Ma et al. (2023) focused on using LLMs to make real-time strategic decisions in the game StarCraft II. Their approach involved converting sequences of game frames into text-based inputs for the LLM, which then translated this information into macro actions or strategies for the agent. With minimal fine-tuning, the LLM-based system developed effective strategies, and with further refinement, it performed comparably to highly ranked human players. Despite the promising results, Ma et al. (2023) acknowledged several limitations in their work. Even with implemented frame-skipping techniques, the computational requirements for real-time decision-making remained substantial. The researchers reported that their system required a high-end GPU to process game states and generate actions within the 22.4ms time frame allowed by StarCraft II's game engine. This high computational demand poses challenges for practical implementation, especially in resource-constrained environments.

3.2.2 Turn-Based Decision-Making. Hu et al. (2024) applied a similar approach to the turn-based game Pokémon. Their method stored text-based game state information in a historical turn log, which the LLM analyzed to learn optimal policies. They introduced an "In Context Reinforcement Learning" approach, where the agent's actions were refined based on their impact on the game state. This method demonstrated the ability to generate human-like battle strategies. Additionally, Hu et al. (2024) demonstrated how their turn-based approach significantly reduced computing requirements compared to real-time decision-making systems. By leveraging the discrete nature of Pokémon battles, their system could spend up to 30 seconds deliberating on each turn without impacting game-play. This allowed for more thorough analysis of game states and potential strategies, while still maintaining responsiveness. The researchers

reported that their system could run effectively on consumer-grade hardware, making it more accessible for practical applications.

4 Tools for LLM Research in Military Planning

This section provides an overview of key tools that can be utilized in future research on applying LLMs to military planning and war-gaming simulations.

4.1 FFmpeg

FFmpeg is a powerful, open-source software suite for handling multimedia data. In the context of this research, it can be used to break video footage into individual frames, which is crucial for analyzing and annotating game states or simulation outputs.

Access: https://ffmpeg.org/

4.2 LabelImg

LabelImg is a graphical image annotation tool written in Python. It's particularly useful for annotating bounding boxes over regions of interest in images, which is essential for training object detection models. This tool can be employed to label key elements in game or simulation frames.

Access: https://github.com/tzutalin/labelImg

4.3 FakeYou

FakeYou is a text-to-speech platform that offers a wide range of voices and styles. In the context of LLM-based war-gaming, it can be used to generate audio commentary or voice responses for AI agents, enhancing the realism of simulations.

Access: https://fakeyou.com/

4.4 Large Language Models

Several large language models can be considered for use in military planning and war-gaming research:

- **ChatGPT:** Developed by OpenAI, known for its strong general language understanding and generation capabilities. **Access:** https://openai.com/chatgpt
- **Claude:** Created by Anthropic, noted for its ability to handle complex instructions and maintain context over long conversations. **Access:** https://www.anthropic.com/
- **LLaMA:** Meta's open-source language model, which can be fine-tuned for specific tasks. **Access:** https://github.com/facebookresearch/llama

The choice of LLM would depend on specific research requirements, computational resources, and the need for customization.

4.5 Gymnasium

Gymnasium (formerly OpenAI Gym) is a toolkit for developing and comparing reinforcement learning algorithms. It provides a standardized set of environments and a common interface for RL tasks. In the context of military planning, it could be used to create simulated war-gaming environments for training and evaluating LLM-based agents.

Access: https://gymnasium.farama.org/

These tools, when used in combination, can provide a robust framework for researching the application of LLMs in military planning and war-gaming simulations. They cover various aspects of the research pipeline, from data preparation and annotation to model implementation and environment simulation.

5 Discussion and Synthesis

The reviewed literature demonstrates significant progress in two key areas essential for enhancing military planning and war-gaming through LLMs:

(1) Event Documentation: LLMs have shown the ability to process complex, multi-modal data and generate meaningful descriptions of critical events in real-time. This capability could greatly enhance the recording and analysis of key decision points and outcomes during COA war-gaming sessions.
(2) Strategic Decision-Making: Both real-time and turn-based applications of LLMs in strategy games have demonstrated the potential to generate effective, human-like decisions. This suggests that LLMs could potentially simulate opposing forces or even assist in generating and evaluating friendly force actions during COA analysis.

However, it's important to note that these capabilities have primarily been demonstrated in gaming environments, which, while complex, may not fully capture the intricacies of military operations. The application of these techniques to military planning will likely require significant domain-specific adaptation and rigorous testing to ensure reliability and relevance.

Additionally, the computing resources available to military planners may limit the approach or timeliness of LLM-based strategic decision-making. Real-time applications, as demonstrated by Ma et al. (2023), require significant computational power that may not be readily available in all military planning scenarios. Turn-based approaches, while less computationally intensive, may still face challenges in processing complex military scenarios within acceptable time-frames. Future implementations will need to balance the depth of analysis with the practical constraints of available computing resources and decision-making timelines in military operations.

6 Conclusion and Future Work

The reviewed research illustrates that LLMs have the potential to address the two key requirements for successful COA analysis: documenting critical events and generating strategic decisions. However, these capabilities have largely been demonstrated in isolation within gaming contexts.

Future work should focus on integrating these two aspects into a unified approach specifically tailored for military planning and war-gaming. This could involve developing an LLM-based system capable of:

(1) Analyzing complex military scenarios represented as multi-modal inputs (text, structured data, possibly visual information).
(2) Identifying and explaining critical events within these scenarios.
(3) Generating and evaluating strategic decisions for both friendly and opposing forces.
(4) Providing clear, actionable insights to human planners to support their decision-making process.

Such a system could potentially serve as a powerful tool to address the war-gaming capability gap in military planning, especially in scenarios where experienced personnel are scarce. However, as noted by most of the surveyed research, it would be crucial to conduct extensive validation and testing with domain experts to ensure the system's reliability and relevance to real-world military operations.

Further research should also consider the ethical implications and potential limitations of relying on AI-generated strategies in military planning, as well as the need for maintaining human oversight and decision-making authority in critical situations.

References

[1] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems* 33 (2020), 1877–1901.

[2] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805* (2018).

[3] Roberto Gallotta, Graham Todd, Marvin Zammit, Sam Earle, Antonios Liapis, Julian Togelius, and Georgios N. Yannakakis. 2024. Large Language Models and Games: A Survey and Roadmap. *arXiv preprint arXiv:2402.18659* (2024). https://arxiv.org/abs/2402.18659

[4] Matthew Guzdial, Shukan Shah, and Mark Riedl. 2018. Towards Automated Let's Play Commentary. *arXiv preprint arXiv:1809.09424* (2018). https://arxiv.org/pdf/1809.09424

[5] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation* 9, 8 (1997), 1735–1780.

[6] Sihao Hu, Tiansheng Huang, and Ling Liu. 2024. POKÉLLMON: A Human-Parity Agent for Pokémon Battles with Large Language Models. *arXiv preprint arXiv:2402.01118* (2024). https://arxiv.org/abs/2402.01118

[7] Tatsuya Ishigaki, Goran Topić, Yumi Hamazono, Hiroshi Noji, Ichiro Kobayashi, Yusuke Miyao, and Hiroya Takamura. 2021. Generating Racing Game Commentary from Vision, Language, and Structured Data. *Proceedings of the 14th International Conference on Natural Language Generation* (2021), 103–113. https://doi.org/10.18653/v1/2021.inlg-1.11

[8] Chengxi Li, Sagar Gandhi, and Brent Harrison. 2019. End-to-End Let's Play Commentary Generation Using Multi-Modal Video Representations. In *Proceedings of the 14th International Conference on the Foundations of Digital Games*. 1–7. https://doi.org/10.1145/3337722.3341870

[9] Weiyu Ma, Qihao Mi, Xidong Yan, Yizhe Wu, Rui Lin, Huan Zhang, and Jun Wang. 2023. Large Language Models Play StarCraft II: Benchmarks and A Chain-of-Summarization Approach. *arXiv preprint arXiv:2312.11865* (2023). https://arxiv.org/abs/2312.11865

[10] picture1 2018. USARMY CGSC Landpower Exercise. https://www.army.mil/article/202457/cgsc$_{t}ests_{b}oard_{b}ased_{s}trategy_{g}ame$.

[11] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. *OpenAI blog* 1, 8 (2019), 9.

[12] Noah Renella and Markus Eger. 2023. Towards Automated Video Game Commentary Using Generative AI. *AIIDE Workshop on Experimental Artificial Intelligence in Games* (2023). https://www.exag.org/papers/Towards%20Automated%20Video%20Game%20Commentary%20Using%20Generative%20AI.pdf

[13] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems* 30 (2017).

Received 12 September 2024