Bryan Huici
ENC 3241: Project #3
Office of Undergraduate Research Proposal
November 16, 2025

Project Title

Safeguarding Personal Intelligence: Efficient Private Knowledge Management Systems for Resource-Constrained On-Device Large Language Models

Project Objective or Aim

This research investigates efficient methods for managing private knowledge in on-device large language models (LLMs) deployed on embedded systems while maintaining robust security and fault tolerance. The primary research questions are: (1) How can we optimize knowledge graph-based retrieval augmented generation (RAG) systems to operate efficiently within the memory and computational constraints of embedded devices? (2) What security vulnerabilities emerge when personal knowledge is stored and accessed locally, and how can we mitigate attacks such as bit-flip manipulation on volatile and non-volatile memory? (3) How do soft errors and intentional memory corruptions affect the accuracy of personalized LLM responses, and what fault-tolerant mechanisms can preserve response quality? This work aims to develop a lightweight, secure framework that enables personalized LLM interactions on resource-constrained devices without compromising user privacy or system reliability.

Project Background and Significance

Large language models have revolutionized natural language processing, but their deployment on personal devices faces critical challenges in balancing personalization, privacy, security, and resource efficiency. Cloud-based LLMs expose sensitive user data to privacy risks when personal information is transmitted for personalization, while on-device deployment faces severe memory constraints—modern smartphones typically have 6-12 GB of DRAM, limiting viable models to sub-billion parameter sizes.

Recent advances in retrieval augmented generation (RAG) combined with knowledge graphs (KGs) offer promising solutions for LLM personalization. RAG systems enhance LLM reliability by grounding responses in factual, up-to-date information, while KGs provide structured, dynamically updateable representations of personal data from calendars, conversations, and user interactions. However, existing approaches primarily target cloud deployment and fail to address the unique constraints and vulnerabilities of embedded systems.

The significance of this research is threefold. First, it addresses the growing need for privacy-preserving AI by keeping personal data on-device, preventing sensitive information

aggregation by cloud LLM providers. Second, it tackles the technical challenge of operating sophisticated personalization systems within embedded system resource constraints through memory-efficient KG compression and optimized RAG retrieval. Third, it pioneers security analysis of on-device personalized LLMs, examining novel attack vectors targeting the knowledge management layer through memory corruption techniques including bit-flips in DRAM and flash storage.

The embedded systems context introduces unique challenges beyond traditional LLM deployment. Mobile devices exhibit hierarchical memory structures with DRAM (6-12 GB), last-level cache (8-32 MB), and flash storage (~100 GB), each with different access speeds and vulnerability profiles. KV cache management becomes critical—for a 7B parameter model processing 3K tokens, the KV cache can exceed 10 GB, approaching or exceeding available DRAM. Furthermore, embedded systems are particularly susceptible to soft errors from cosmic radiation and voltage fluctuations, with DRAM bit-flip rates increasing as process technology scales down. These reliability concerns compound when adversaries can exploit Rowhammer-style attacks to intentionally corrupt memory.

This research employs a theoretical framework grounded in systems security, fault-tolerant computing, and information retrieval. We utilize threat modeling to identify attack surfaces in the knowledge management pipeline, particularly focusing on the unique vulnerabilities introduced when private knowledge graphs reside in attacker-accessible local memory. We apply error detection and correction theory to develop resilient data structures that can withstand both natural soft errors and intentional bit manipulation. Compression theory guides our optimization of KG storage and KV cache transmission, informed by recent findings that token-wise locality and layer-wise sensitivity enable 3-5× compression with minimal quality loss. The intersection of these frameworks enables holistic analysis of the privacy-security-efficiency tradeoff inherent in on-device personalized LLMs, advancing both theoretical understanding and practical deployment of trustworthy edge AI systems.

Research Methods

This research employs a multi-phase experimental methodology combining system development, security analysis, and performance evaluation on real embedded hardware.

Phase 1: Baseline System Development (Weeks 1-3)
We will implement a foundational on-device RAG system using open-source LLMs (Llama-2 7B, MobileLLM 350M) on embedded platforms including Raspberry Pi 4, NVIDIA Jetson Nano, and Gem5 simulator. The system will integrate knowledge graphs constructed from synthetic personal data (calendar events, conversations, contact information) using Neo4j graph database and implement vector embedding-based retrieval using sentence-BERT for contextual matching. This phase establishes performance baselines for inference latency, memory consumption, and response quality measured using ROUGE and BLEU scores.

Phase 2: Memory Optimization (Weeks 4-6)
We will develop and evaluate compression techniques for knowledge graph storage and KV cache management. Building on CacheGen's insights about layer-wise sensitivity and token-wise locality in KV caches, we will implement quantization strategies that preserve response quality while reducing memory footprint by 3-4×. For knowledge graphs, we will explore pruning strategies that retain high-importance entities while compressing less-critical information. Evaluation metrics include compression ratio, decompression latency, and impact on retrieval accuracy.

Phase 3: Security Analysis and Attack Implementation (Weeks 7-9)
We will systematically analyze security vulnerabilities by implementing controlled bit-flip attacks on volatile memory (DRAM) and non-volatile storage (flash). Using tools like Rowhammer for DRAM manipulation and direct memory access for flash corruption, we will target: (1) KG entity representations to cause misidentification, (2) embedding vectors to corrupt retrieval results, (3) KV cache values to induce hallucinations, and (4) model weights to degrade overall performance. We will measure attack success rates and characterize impact on output quality.

Phase 4: Fault-Tolerance Mechanisms (Weeks 10-11)
Based on attack analysis, we will design and implement protective mechanisms including error-correcting codes for critical KG nodes, checksums for embedding integrity verification, redundant storage for high-importance entities, and anomaly detection algorithms to identify corrupted retrievals. We will evaluate the overhead-protection tradeoff for each technique.

Phase 5: Integration and Evaluation (Weeks 12-13)
Final integration testing will assess the complete system across multiple dimensions: response quality (using question-answering benchmarks), latency (prefill and decoding time), memory efficiency (peak and average usage), energy consumption (measured on Jetson Nano), and security resilience (successful attack mitigation rate). We will compare our approach against baseline cloud-based personalization and unprotected on-device systems.

Expected Outcome

This research will produce several significant deliverables that advance the state-of-the-art in secure on-device AI systems. The primary technical deliverable is a fully functional prototype system demonstrating efficient, secure private knowledge management for on-device LLMs, with source code, datasets, and documentation released as open-source software on GitHub under an MIT license. This will include implementation for multiple embedded platforms, comprehensive API documentation, and example applications demonstrating calendar-based personalization and conversational context management.

We anticipate submitting our findings to top-tier venues in embedded systems and security, specifically targeting the ACM International Conference on Embedded Software (EMSOFT) or IEEE/ACM Design Automation Conference (DAC), with a goal of submission in Fall 2025.

Additionally, we will prepare a poster presentation for the UCF Office of Undergraduate Research Symposium in Fall 2025, making our work accessible to the broader UCF community.

For knowledge dissemination to practitioners, we will prepare a technical white paper detailing best practices for secure on-device LLM deployment, targeted at embedded systems developers and mobile application engineers. This document will provide practical guidelines on balancing personalization quality with security requirements and resource constraints.

The expected contributions to the field include: (1) the first comprehensive security analysis of knowledge graph-based RAG systems on embedded platforms, identifying novel attack vectors and quantifying their impact; (2) memory-efficient compression techniques specifically designed for on-device knowledge management, achieving 3-5× reduction in storage requirements while maintaining retrieval quality; (3) fault-tolerant mechanisms that provide robust protection against both intentional attacks and natural soft errors with minimal performance overhead (<10% latency increase); and (4) empirical benchmarks establishing baselines for personalized LLM performance on resource-constrained devices.

For the UCF community, this research addresses timely concerns about AI privacy and security while demonstrating the university's leadership in embedded AI systems. The open-source release will enable other undergraduate researchers to build upon our work, fostering collaborative innovation. Furthermore, the techniques developed have direct applications to emerging UCF initiatives in healthcare IoT, smart campus technologies, and assistive devices—all domains where privacy-preserving on-device intelligence is crucial. By demonstrating that sophisticated AI personalization can operate securely within resource constraints, this work helps democratize access to personalized AI while protecting user privacy, contributing to more equitable and trustworthy AI deployment.

Literature Review

[1] Prahlad, D., Lee, C., Kim, D., & Kim, H. (2025). Personalizing large language models using retrieval augmented generation and knowledge graph. In Companion Proceedings of the ACM Web Conference 2025 (WWW Companion '25). ACM. https://doi.org/10.1145/3701716.3715473

[2] Baek, J., Chandrasekaran, N., Cucerzan, S., Herring, A., & Jauhar, S. K. (2024). Knowledge-augmented large language models for personalized contextual query suggestion. In Proceedings of the ACM Web Conference 2024 (WWW '24). ACM. https://doi.org/10.1145/3589334.3645404

[3] Liu, Z., Zhao, C., Iandola, F., Lai, C., Tian, Y., Fedorov, I., Xiong, Y., Chang, E., Shi, Y., Krishnamoorthi, R., Lai, L., & Chandra, V. (2024). MobileLLM: Optimizing sub-billion parameter language models for on-device use cases. In Proceedings of the 41st International Conference on Machine Learning (ICML 2024). PMLR.

[4] Liu, Y., Li, H., Cheng, Y., Ray, S., Huang, Y., Zhang, Q., Du, K., Yao, J., Lu, S., Ananthanarayanan, G., Maire, M., Hoffmann, H., Holtzman, A., & Jiang, J. (2024). CacheGen: KV cache compression and streaming for fast large language model serving. In ACM SIGCOMM 2024. ACM. https://doi.org/10.1145/3651890.3672274

[5] Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., Yih, W., Rocktäschel, T., Riedel, S., & Kiela, D. (2020). Retrieval-augmented generation for knowledge-intensive NLP tasks. In Proceedings of the 34th International Conference on Neural Information Processing Systems (NIS '20). Article 793.

[6] Pan, S., Luo, L., Wang, Y., Chen, C., Wang, J., & Wu, X. (2024). Unifying large language models and knowledge graphs: A roadmap. IEEE Transactions on Knowledge and Data Engineering.

[7] Qin, R., Xia, J., Jia, Z., Jiang, M., Abbasi, A., Zhou, P., Hu, J., & Shi, Y. (2024). Enabling on-device large language model personalization with self-supervised data selection and synthesis. In Proceedings of the 61st ACM/IEEE Design Automation Conference.

[8] Touvron, H., et al. (2023). Llama 2: Open foundation and fine-tuned chat models. arXiv preprint arXiv:2307.09288.

Preliminary Work and Experience

I joined the Knight Scholar's Research Project in late Summer 2025, working under the mentorship of postdoctoral scholar Dr. Chanhee Lee. Recognizing that most team members lacked prior experience with large language models, our team has been systematically building foundational knowledge by studying "Building a Large Language Model" to understand LLM architecture, training, and deployment principles. This structured approach has provided me with essential theoretical grounding in transformer models, attention mechanisms, and the computational requirements of LLM inference.

My most significant preliminary contribution has been developing a build automation tool for MLC LLM, a machine learning compiler framework designed for efficient LLM deployment. This tool streamlines the compilation process, allowing team members to focus on source code modifications rather than wrestling with complex build configurations. By removing friction from the development workflow, this tool will accelerate our experimentation during the proposed research, particularly when implementing custom optimizations for on-device deployment and testing security modifications to the LLM inference pipeline.

My coursework provides relevant preparation for this project. I am currently enrolled in COP 3502C (Computer Science I), which covers data structures and algorithms in C, which is essential for implementing efficient knowledge graph data structures and understanding memory management. Additionally, I am taking CDA 3103C (Computer Architecture), where I am

learning about memory hierarchies, cache systems, and hardware-level performance optimization, all directly applicable to understanding embedded system constraints and memory-based attack vectors.

I bring three years of programming experience across multiple languages and platforms. Most recently, I won first place at a hackathon where my team built a full-stack application integrating AI agents using both Python and JavaScript, demonstrating my ability to rapidly prototype complex systems and integrate AI capabilities. My web development background has given me strong software engineering skills including version control, testing, and deployment—transferable skills that will support rigorous experimental methodology and reproducible research.

While I am early in my research journey, this combination of ongoing LLM education, systems-level programming skills, practical automation experience, and proven ability to deliver working software under pressure positions me to successfully execute this research project with appropriate mentorship and dedication.

IRB/IACUC Statement

This research does not involve human subjects, as all testing will be conducted on synthetic datasets and simulated user interactions. No surveys, interviews, or collection of personal information from human participants will occur. Similarly, no animal subjects will be used in this research. Therefore, this project does not require IRB or IACUC approval.

Budget

Hardware and Equipment:
- NVIDIA Jetson Nano Developer Kit (4GB): $149.00
- Raspberry Pi 4 (8GB) with accessories: $135.00
- High-speed microSD cards (2x 256GB A2 class): $60.00
- Power measurement equipment (USB power meter): $35.00

Software and Cloud Resources:
- GPU cloud computing credits (for training): $300.00
- Neo4j Enterprise license (educational): $0.00 (free for research)

Components and Peripherals:
- Cooling solutions (heatsinks, fans): $40.00
- Breadboard and fault injection components: $45.00
- JTAG debugger for embedded testing: $180.00

Conference and Dissemination:
- Poster printing for research symposium: $75.00

- Conference registration (if accepted): $450.00

Miscellaneous:
- USB hubs, cables, storage drives: $31.00

TOTAL: $1,500.00

This budget prioritizes hands-on hardware experimentation on multiple embedded platforms to ensure our results generalize across different architectures. The cloud computing credits enable training custom compressed models when embedded hardware is insufficient, while the fault injection components allow rigorous security testing. Conference attendance supports knowledge dissemination to the broader research community, and the poster printing ensures we can share findings with the UCF community at the undergraduate research symposium.