

**Capstone Project Personal Report**

**Project Title:**

**Team Members:**

#### **Personal Contribution in the Project**

* **Tasks and Responsibilities:** Describe the specific tasks and responsibilities you handled in the project.
* **Challenges and Solutions:** Discuss any challenges you faced and how you addressed them.
* **Collaboration:** Explain how you collaborated with your team members and your role in team dynamics.

#### **Skills Development**

In the table below, let us know which skills you believe you have applied in the project you submitted and why.

| **TARGET SKILLS** | **EVALUATION CRITERIA** | **Have you applied this skill in your project ?** | **If yes, please justify :** |
| --- | --- | --- | --- |
| **Core Skills Certification** |  |  |  |
| **S.1 Analyze needs for LLM customization.**  Analyze the specific requirements and use cases for integrating a large language model (LLM). Draft comprehensive functional specifications that clearly define the development objectives, ensuring alignment with identified needs and technical feasibility. | - The user flow modeling adheres to a formalism, including functional diagrams, wireframes, etc.  - Each functional specification covers the context, usage scenarios, and validation criteria. | Yes/No |  |
| **S.2 Identify pre-existing services for LLM customization**.  Based on the identified organizational and/or business requirements as well as desired functionalities, conduct a comprehensive evaluation of existing AI services and large language models. Analyze their capabilities, performance characteristics, and suitability for customization. Formalize recommendations for one or more AI services and LLMs that best align with the identified needs. | -The expression of needs is reformulated to clearly outline the objectives and constraints of the artificial intelligence integration project.  -The benchmark lists the examined services, along with those that are not studied.  -The reasons for dismissing a service are explained in detail.  -The conclusions precisely define the services that meet the requirements, highlighting their advantages and disadvantages. | Yes/No |  |
| **S.3 Prompt engineering.**  Design and iteratively refine effective prompts to optimize the outputs of a large language model (LLM). Leverage prompt engineering techniques and best practices to ensure the LLM generates accurate, relevant, and task-specific responses aligned with the desired use case and requirements. | - The prompts are clear, concise, and specifically formulated to address the desired output.  - The prompts demonstrate an understanding of the language model's capabilities and limitations.  - Variations in prompt structures are explored to identify the most effective input for the given task.  - The refined prompts lead to outputs that align closely with the intended objectives.  - The student can articulate the rationale behind prompt choices and adjustments made during the prompt engineering process.  - The student can identify instances where prompt engineering positively influences the quality of the model's responses. | Yes/No |  |
| **S.4 Integrate LLMs via APIs**  Use APIs to integrate large language models (LLMs) into existing applications. Develop and implement API-based solutions to enable real-time interaction and data exchange between the LLM and the target application, ensuring efficient and secure communication while adhering to best practices for API design and development. | - The student successfully accesses and understands the documentation of relevant LLM APIs.  - The student can articulate the purpose and goals of integrating the LLM into the application.  - API calls are appropriately implemented, demonstrating an understanding of request formats and authentication methods.  - The student can provide examples of how the LLM integration contributes to specific features or improvements within the application. | Yes/No |  |
| **S.5 Select the appropriate LLMs tools based on project needs.**  After defining the project needs and technical requirements, select the most relevant LLM ecosystem and its appropriate tools, based on their specific capabilities (e.g. summarization, content creation, Q&A). | - The choice of tools aligns with the specific requirements and objectives of the task.  - Adaptability is showcased by selecting and integrating various tools within a LLM ecosystem (such as Meta or Mistral AI) based on evolving project needs.  - The tools selected contribute effectively to the overall efficiency and success of the project.  - The student can justify their tool selection by highlighting key features and functionalities within Meta's ecosystem. | Yes/No |  |
| **S6. Analyze domain-specific data for LLM customization.**  Analyze the characteristics, formats, and nuances of data within specific domains or industries. Leverage this knowledge to inform effective data preprocessing, feature extraction, and customized training strategies for large language models (LLMs). | - The student demonstrates a thorough understanding of domain-specific data, identifying unique characteristics and nuances relevant to Language Models.  - The student justifies the selection of features by connecting them to linguistic elements and context within the LLM domain.  - The student effectively communicates the pivotal role of domain-specific data understanding in optimizing Language Model training outcomes. | Yes/No |  |
| **S.7 Preprocess data for LLM Customization.**  Clean, transform, and preprocess data to ensure its quality, consistency, and suitability for customizing large language models (LLMs). Identify and address potential data issues, such as noise, inconsistencies, or biases, that might adversely affect the performance and outputs of the customized LLM. Apply appropriate data preprocessing techniques to enhance the quality and relevance of the data for the target domain or use case, enabling effective LLM customization through methods like fine-tuning, retrieval augmentation, or other customization approaches. | - The student proficiently cleans and preprocesses data, ensuring its quality and relevance for training.  - Data cleaning techniques are applied effectively to address issues such as missing values, outliers, and inconsistencies within the dataset.  - The student demonstrates an understanding of the impact of data quality on LLM model performance.  - Preprocessing steps are tailored to the specific requirements of LLMs, considering linguistic nuances and context.  - The student can justify the choice of data cleaning methods and preprocessing steps in the context of LLM training.  - The cleaned and preprocessed data contributes significantly to improved LLM model accuracy and generalization. | Yes/No |  |
| **S.8 Store data in vector databases.**  Transform data into high-dimensional vector embeddings and store them in vector databases for efficient retrieval and similarity-based querying. | - The student effectively applies data augmentation techniques to increase the diversity of the training dataset.  - The student understands the importance of enhancing the model's ability to generalize across different linguistic scenarios through data augmentation.  - The student can articulate the rationale behind the selection of particular augmentation techniques for LLMs.  - The augmented dataset contributes significantly to the robustness and generalization capabilities of the LLM. | Yes/No |  |
| **S9. Implement Retrieval-Augmented Generation for LLMs.**  Integrate retrieval-augmented generation techniques to enhance the performance and outputs of large language models (LLMs). Leverage external knowledge sources or corpora to retrieve relevant information and incorporate it into the LLM's generation process, resulting in more accurate, contextualized, and informative responses, tailored to the specific domain or use case. | - The student demonstrates a clear understanding of how retrieval-based and generation-based models work individually and how they are combined in RAG.  - The student is able to effectively leverage a vector database with an appropriate retrieval method  - The student is able to integrate a generative model with the retrieval system to ensure that the retrieved information is used effectively in generating responses. | Yes/No |  |
| **S.12 - Monitor LLM performance.**  Implement monitoring solutions to continuously track the performance and behavior of deployed large language models (LLMs) in production environments. Establish metrics and processes to assess accuracy, reliability, and adherence to expected outputs, identifying any issues or degradation over time. | - The monitoring indicators are defined and applied to the critical parts that were identified earlier.  - The monitoring system is in place and working properly.  - The logs are organized and stored in accordance with best practices.  - An automatic alert system (email, notification, etc.) is set up and working properly. | Yes/No |  |
| **S.13 Evaluate LLM performance.**  Select and apply appropriate metrics for evaluating model performance, providing insights for iterative model improvements. | - The student proficiently selects and applies appropriate metrics for evaluating model performance, offering valuable insights for iterative improvements.  - Metric selection is based on a deep understanding of the specific goals and characteristics of the model, ensuring relevance to the intended application.  - The student effectively interprets and communicates the results of model evaluations, highlighting strengths, weaknesses, and areas for improvement.  - The student can justify the choice of specific evaluation metrics, linking them to the goals and nuances of the model and the application it serves. | Yes/No |  |
| **S.14 Develop an interactive LLM application.**  Design and build an application enabling human-LLM interaction, implementing specified user interfaces, features, and user flows. Iteratively enhance the application based on user feedback to improve the experience. | - The student skillfully creates an application enabling seamless interaction with a LLM  - The interfaces are developed and match the mockups.  - The developed features align with the user flow.  - The student can articulate the key features and functionalities of the application, highlighting its potential use cases and benefits.  - User feedback and iterative improvements to the application reflect a commitment to enhancing the user experience and addressing any potential issues. | Yes/No |  |
| **Mastery Skills Certification** |  |  |  |
| **S.10 Fine-tune LLMs for specific use-cases.**  Apply fine-tuning techniques to pre-trained large language models (LLMs) to adapt and optimize their performance for specific tasks or targeted use cases. Leverage relevant domain data and objective functions to specialize the LLM's knowledge and generation capabilities, enhancing its accuracy, relevance, and effectiveness in addressing the requirements of the intended application or domain. | - The student effectively applies fine-tuning techniques to pre-trained language models, adapting them to specific tasks and optimizing performance for targeted use cases.  - The student is able to implement RLHF to enhance the model’s ability to learn through experience.  - The fine-tuned language models exhibit improved performance metrics, indicating successful adaptation to the intricacies of the targeted use cases.  - The student can articulate the rationale behind the choice of fine-tuning strategies, linking them to the specific characteristics and demands of the designated tasks. | Yes/No |  |
| **S.11 Optimize LLMs efficiency through quantization.**  Apply quantization techniques like weight quantization, precision reduction, and model compression to reduce the computational requirements and model size of large language models (LLMs), enabling more efficient deployment and inference while minimizing accuracy or generation capability degradation. | - The student demonstrates a thorough understanding of quantization and its purpose (e.g., reducing model size, improving inference speed).  - The student is able to explain different types of quantization (e.g., post-training quantization, quantization-aware training) and their trade-offs.  - The student is able to choose an appropriate quantization technique based on the specific use case and model architecture.  -The student correctly implements quantization techniques using relevant frameworks and tools | Yes/No |  |
| **S.15 Deploy LLM applications.**  Deploy large language model (LLM) applications to various platforms and real-world environments, ensuring scalability to handle increasing workloads and optimization for efficient performance and resource utilization. | - The application is deployed  - Deployment processes are executed efficiently, ensuring the application is accessible.  - The student demonstrates proficiency in optimizing the application for performance.  - Scalability considerations are incorporated into the deployment strategy.  - The student can articulate the deployment architecture, highlighting scalability and optimization goals. | Yes/No |  |
| **S.16 Practice responsible and ethical AI development.**  Integrate risk analysis, governance principles, and ethical considerations throughout the development lifecycle to ensure the responsible and trustworthy deployment of large language models (LLMs). Implement safeguards to mitigate potential risks, biases, and harmful outputs, upholding ethical AI standards and practices. | - The student ensures responsible and ethical use of deployed models, integrating comprehensive risk analysis and governance principles into the development process.  - A thorough understanding of responsible AI practices is demonstrated  - Risk analysis is conducted systematically, identifying and addressing potential biases, privacy concerns  - The student can articulate the governance principles applied to the development process | Yes/No |  |