Study Guide: Building AI Solutions with ML APIs & Foundational Models

Subject: Google Cloud Professional Machine Learning Engineer Exam Topics:

- ML APIs: Vision, Natural Language, Speech, Translation
- Model Garden & Pre-trained Foundational Models
- Industry-Specific APIs (Document AI, Retail API)
- RAG Applications with Vertex AI Agent Builder
- Model Deployment & Endpoint Management

Summary

This study guide focuses on leveraging pre-trained ML APIs and foundational models available in Google Cloud's Model Garden to rapidly build AI-powered applications without extensive ML expertise. It covers the practical application of specialized APIs (Vision, Natural Language, Speech, Translation), industry-specific solutions (Document AI, Retail), deployment of foundational models, and building Retrieval Augmented Generation (RAG) applications using Vertex AI Agent Builder for context-aware AI systems.

Key Concepts

Model Garden: A curated repository of pre-trained models and APIs in Vertex AI, including Google's proprietary models (PaLM 2, Gemini) and open-source models (Llama 2, Stable Diffusion). Enables rapid deployment without training from scratch.

ML APIs - Quick Integration: Pre-built APIs like `Vision API` (image analysis, OCR, object detection), `Natural Language API` (sentiment analysis, entity recognition), `Speech-to-Text`, and `Translation API` that require minimal setup and no ML expertise.

Document AI: Industry-specific API for intelligent document processing, featuring specialized processors for forms, invoices, receipts, contracts, and custom documents. Uses OCR combined with ML to extract structured data from unstructured documents.

Foundational Models: Large pre-trained models (LLMs, vision transformers) that can be fine-tuned or used via prompt engineering for various tasks. Examples include PaLM 2 for text generation, Gemini for multimodal tasks, and Imagen for image generation.

RAG (Retrieval Augmented Generation): An architecture pattern that combines information retrieval with generative AI. Vertex AI Agent Builder enables RAG by connecting LLMs to external knowledge bases (documents, websites) to provide accurate, contextually relevant responses with reduced hallucinations.

Vertex AI Endpoints: Managed infrastructure for deploying and serving ML models with automatic scaling, traffic splitting for A/B testing, and monitoring. Supports both Google's models and custom models.

Vocabulary List

Vision API: Google Cloud's image analysis service providing features like label detection, face detection, OCR, explicit content detection, and landmark recognition.

Natural Language API: Provides sentiment analysis, entity recognition, content classification, and syntax analysis for text using pre-trained models.

Vertex AI Agent Builder: A platform for building conversational AI agents with RAG capabilities, integrating search, recommendations, and generative responses from foundational models.

Document AI Processors: Pre-trained or custom processors for specific document types (invoices, forms, IDs, receipts) that extract structured data with high accuracy.

Prompt Engineering: The practice of crafting effective input prompts to guide foundational models toward desired outputs without fine-tuning.

Model Tuning vs. Prompt Design: Model tuning involves fine-tuning with custom data; prompt design achieves results through clever input construction. Choose tuning for domain-specific accuracy, prompting for rapid iteration.

Vertex AI Search: Enterprise search solution that can index internal documents and websites, powering RAG applications with relevant context retrieval.

Translation API: Provides neural machine translation for 100+ languages with support for custom terminology and domain-specific glossaries.

Speech-to-Text API: Converts audio to text with support for multiple languages, speaker diarization, and automatic punctuation.

AutoML Vision/NLP: Allows custom model training for vision and NLP tasks with minimal ML expertise, bridging the gap between pre-trained APIs and fully custom models.

Key Questions

1. **When should you choose a pre-trained Vision API over training a custom AutoML Vision model?**

2. **What are the key components of a RAG application in Vertex AI Agent Builder, and how do they work together?**

3. **How does Document AI differ from Vision API's OCR capabilities, and when would you use each?**

4. **What strategies can you employ to reduce hallucinations when using foundational models for enterprise applications?**

5. **Explain the difference between deploying a model from Model Garden to Vertex AI Endpoints versus using it through an API call.**