Componente de Gestión del Diálogo y NLU: Rasa Open Source

Para el núcleo de procesamiento de lenguaje natural (NLU) y la gestión del estado de la conversación, se ha seleccionado el framework Rasa Open Source. Esta herramienta actuará como el orquestador principal del sistema, cumpliendo funciones críticas que requieren precisión y bajo margen de error.

Rasa será responsable de la etapa de comprensión semántica inicial. Mediante su pipeline NLU, el sistema identificará la intencionalidad del usuario (Intents) y extraerá las entidades clave (Entities) necesarias para contextualizar la solicitud, tales como fechas, códigos de asignaturas o tipos de trámites. A diferencia de un modelo puramente generativo, Rasa permite establecer "guardarraíles" o reglas de negocio estrictas. Esto garantiza que el flujo de la conversación se mantenga dentro de los parámetros institucionales, asegurando que el bot pueda saludar, despedirse y gestionar errores de comunicación de forma predecible y controlada. Rasa no generará las respuestas complejas, sino que delegará esta tarea al modelo generativo cuando se detecte una consulta que requiera información específica del corpus de conocimiento.

Arquitectura de Recuperación y Generación Aumentada (RAG)

Para mitigar el riesgo de alucinaciones (generación de información falsa) y garantizar la precisión factual, el sistema no dependerá únicamente de la memoria interna del modelo generativo. Se implementará un mecanismo de Generación Aumentada por Recuperación (RAG).

El flujo técnico diseñado opera de la siguiente manera: Una vez que Rasa identifica una entidad específica o "token" crítico en la consulta del usuario (por ejemplo, el token "biblioteca" o "reglamento_cancelacion"), se activa una acción personalizada (Custom Action) en el servidor de lógica. Este script ejecuta una búsqueda semántica en una base de conocimientos externa (documentos PDF indexados o bases de datos) relacionada con dicha entidad. La información recuperada se inyecta dinámicamente en el parámetro de Instrucción (Instruction) del modelo. De esta forma, el modelo Llama 3 recibe un prompt enriquecido que contiene no solo la pregunta del usuario (Input), sino también el contexto normativo exacto necesario para responder. Esto asegura que la generación de texto se adhiera estrictamente a la normativa vigente proporcionada en tiempo real.

Estado del Arte en Modelos de Lenguaje Open Source (SLM)

Para el componente de generación de respuestas, se evaluaron modelos de lenguaje de código abierto susceptibles de ser reentrenados (Fine-Tuning) bajo recursos computacionales limitados . El foco se centró en modelos con un tamaño inferior a los 10 billones de parámetros, conocidos técnicamente como SLM (Small Language Models), los cuales ofrecen un equilibrio óptimo entre capacidad de razonamiento y eficiencia de hardware. A continuación, se presenta la comparativa de los candidatos más relevantes:

Mistral 7B (v0.3)

Desarrollado por Mistral AI, este modelo ha demostrado un rendimiento superior a modelos mucho más grandes (como Llama 2 13B) en diversos benchmarks. Su arquitectura utiliza atención de consulta agrupada (GQA) para una inferencia más rápida y ventanas de contexto deslizantes (Sliding Window Attention) para gestionar secuencias largas con menor memoria. Aunque es altamente capaz y eficiente, su rendimiento nativo en español, aunque competente, a veces requiere un mayor esfuerzo en el ajuste fino para captar matices culturales específicos frente a competidores más recientes.

Google Gemma 7B

Gemma es la apuesta de modelos abiertos de Google, derivada de la misma investigación y arquitectura técnica que los modelos Gemini. Destaca por ser extremadamente ligero y rápido, utilizando técnicas avanzadas de normalización (RMSNorm) y embeddings rotacionales (RoPE). Sin embargo, en pruebas de razonamiento complejo y seguimiento de instrucciones largas (Instruction Following), tiende a quedarse ligeramente por detrás de la familia Llama en tareas que requieren una redacción extensa y estructurada, característica necesaria para las respuestas académicas formales.

Meta Llama 3 (8B Instruct)

Es la iteración más reciente de Meta y representa el actual estado del arte para modelos de este tamaño. Llama 3 ha sido entrenado con un dataset de 15 billones de tokens, significativamente mayor que sus predecesores, lo que le otorga una capacidad de comprensión semántica y conocimientos generales superior. La variante "Instruct" ha sido optimizada específicamente mediante Aprendizaje por Refuerzo con Retroalimentación Humana (RLHF) para seguir instrucciones detalladas y mantener la coherencia en diálogos largos, lo cual es crítico para adoptar la "persona" de un

asistente universitario. Su tokenizador (TikToken) ha sido ampliado a 128k tokens, ofreciendo una mayor eficiencia en la compresión y generación de texto en múltiples idiomas, incluido el español.

Familia GPT (Modelos Propietarios en Azure OpenAI)

Se incluye en el análisis la serie de modelos "Mini" de OpenAI, accesibles mediante la infraestructura de Azure (GPT-4o Mini y la evolución hacia GPT-5 Mini). Estos modelos representan un cambio de paradigma hacia la eficiencia mediante técnicas de destilación de conocimiento (Knowledge Distillation) y arquitecturas de Mezcla de Expertos (MoE). A diferencia de los modelos monolíticos tradicionales, estos modelos activan solo una fracción de sus parámetros por inferencia, lo que reduce drásticamente la latencia y el coste. Ofrecen una ventana de contexto masiva (128k tokens) y capacidades multimodales nativas. Al ser modelos de frontera (SOTA), su capacidad de razonamiento lógico y "sentido común" suele superar a los modelos pequeños de código abierto sin necesidad de ajustes finos extensivos. Sin embargo, operan bajo un esquema de "Caja Negra" (Black Box), donde los pesos del modelo y el proceso de entrenamiento no son accesibles ni modificables por el usuario.

Matriz Comparativa de Modelos Candidatos

A continuación se presenta la evaluación técnica comparativa considerando las restricciones del proyecto (Hardware limitado, necesidad de español fluido y capacidades de razonamiento):

Característica	Mistral 7B (v0.3)	Google Gemma 7B	Meta Llama 3 (8B)	GPT-4o/5 Mini (Azure)
Tipo de Licencia	Apache 2.0 (Abierta)	Gemma Terms (Abierta)	Llama Community (Abierta)	Propietaria (API Cerrada)
Arquitectura	Transformer (GQA)	Transformer (RoPE)	Transformer (Optimizado)	MoE / Destilado
Despliegue	Local (On-Premise)	Local (On-Premise)	Local (On-Premise)	Nube (SaaS - Azure)
Requerimiento VRAM	~6 GB (Cuantizado 4-bit)	~6 GB (Cuantizado 4-bit)	~6-7 GB (Cuantizado 4-bit)	N/A (Gestionado por MS)

Rendimiento Español	Alto	Medio-Alto	Muy Alto	Excelente (Nativo)
Ventana Contexto	32k tokens	8k tokens	8k tokens (extensible)	128k tokens
Facilidad de Uso	Media (Requiere MLOps)	Media (Requiere MLOps)	Media (Requiere MLOps)	Muy Alta (SDK Python)

Análisis de Compromiso: Código Abierto (Llama 3) vs. Propietario (GPT/Azure)

La decisión final implica un compromiso técnico (trade-off) entre soberanía tecnológica y facilidad operativa.

La opción **GPT/Azure (Familia Mini)** ofrece la mayor viabilidad técnica inmediata. Al utilizar la infraestructura de nube de Microsoft Azure (disponible mediante créditos académicos), se elimina la complejidad de gestión de hardware, drivers GPU y cuantización. Además, los modelos GPT presentan métricas de razonamiento superiores "out-of-the-box". Sin embargo, esta opción genera una dependencia crítica de terceros: los datos deben salir de la infraestructura local para ser procesados, y el modelo resultante del Fine-Tuning en Azure no es exportable; si el servicio se interrumpe o los créditos expiran, la universidad pierde el activo intelectual del modelo entrenado.

Por otro lado, la opción **Meta Llama 3 (Open Source)** otorga control total. Permite la ejecución del modelo en servidores locales (On-Premise) sin conexión a internet, garantizando la máxima privacidad de los datos sensibles extraídos de los correos. Aunque requiere una implementación más compleja de ingeniería de IA (uso de cuantización QLoRA y gestión de memoria), el resultado es un modelo "propio" (pesos adaptadores) que puede ser archivado, versionado y migrado libremente. Desde una perspectiva académica y de ingeniería, esta ruta demuestra competencias avanzadas en MLOps y gestión de LLMs.

Selección del Modelo y Justificación Técnica

Tras el análisis comparativo, se selecciona **Meta Llama 3 (8B Instruct)** como el motor generativo para este proyecto de grado.

Esta elección se fundamenta en tres pilares alineados con los objetivos de investigación. Primero, su arquitectura de 8 billones de parámetros es compatible con las limitaciones de hardware del proyecto; utilizando técnicas de cuantización, es posible ejecutarlo y reentrenarlo en una instancia estándar de Google Colab (GPU T4) o

servidores universitarios modestos. Segundo, prioriza la soberanía tecnológica y el rigor académico: el uso de un modelo Open Source permite validar técnicas de entrenamiento modernas (SFT/QLoRA) y asegura que el activo generado (el modelo entrenado) pertenezca perpetuamente a la institución sin dependencias de licencias de API externas. Tercero, su capacidad nativa para el "Instruction Tuning" permite definir una instrucción de sistema robusta (System Prompt) que mantenga el tono formal y académico requerido por la universidad, minimizando las alucinaciones.

Estrategia de Entrenamiento y Metodología

Dado que el volumen de datos es moderado y los recursos computacionales son finitos, no se realizará un entrenamiento completo del modelo. En su lugar, se aplicará la técnica **QLoRA (Quantized Low-Rank Adaptation)**.

Esta metodología permite congelar los pesos originales del modelo Llama 3 y entrenar únicamente un conjunto reducido de adaptadores de bajo rango. Esto reduce drásticamente el consumo de memoria VRAM sin sacrificar la calidad del aprendizaje. El entrenamiento será de tipo supervisado (Supervised Fine-Tuning - SFT), utilizando el dataset extraído de los correos electrónicos, el cual será estructurado en formato JSONL bajo el esquema Instrucción-Entrada-Salida. Esta estrategia garantiza que el modelo aprenda no solo la información fáctica contenida en los correos, sino también el estilo de redacción y la estructura de respuesta adecuada para el entorno académico.