Departamento de Ingeniería de Sistemas

SOLUCIONES PARA PROCESOS 401K MEDIANTE AGENTES DE LENGUAJE

UNIVERSIDAD DE ANTIOQUIA
1803
Facultad de Ingeniería

PRACTICANTE: Felipe Rodriguez Angel

ASESORES: Raúl Ramos Pollán

PROGRAMA: Ingeniería de Sistemas

Semestre de la práctica: 2024-2

Introducción

El 401K es un plan de retiro ofrecido en los Estados Unidos que permite a los empleados ahorrar para su jubilación con contribuciones pre-impositivas. Bajo este esquema, los empleados pueden deducir una porción de sus ingresos antes de impuestos y depositarla en una cuenta de inversión, que crecerá libre de impuestos hasta el retiro. Este sistema está regulado bajo ERISA, una ley que busca proteger a los trabajadores garantizando que los planes de retiro sean gestionados en su mejor interés

La administración de datos en la industria del 401K enfrenta desafíos relacionados con la gobernanza de datos, seguridad y cumplimiento normativo. Dado el contexto regulado y la importancia de proteger la Información de Identificación Personal (PII), este proyecto plantea una arquitectura sistémica que utiliza pipelines ETL y agentes de lenguaje con contexto aumentado. La solución, basada en Ruby on Rails, Python y herramientas de AWS, facilita la automatización y la seguridad en la administración de datos, alineándose con los estándares de la industria financiera y las expectativas de los stakeholders

IRS
Department of the Treasury
Internal Revenue Service

Metodología

1. Enfoque General del Estudio

- Metodología mixta (cualitativa y cuantitativa).
- Uso de enfoque ágil con iteraciones continuas.

2. Fases del Proyecto

- **Planificación (6 semanas):** Reuniones con stakeholders, análisis de flujos de trabajo, diseño preliminar.
- **Desarrollo (12 semanas):** Capacitación en herramientas clave, desarrollo en sprints, Pair Programming.
- **Integración (6 semanas):** Configuración de workflows RAG, auditorías externas (SOC).
- **Pruebas (6 semanas):** UAT, pruebas de estrés y seguridad (OWASP).
- **Despliegue (2 semanas):** Implementación gradual, guías de usuario, plan de contingencia.

3. Métodos y Técnicas Específicas

- Prácticas ágiles: reuniones diarias, revisiones de código, pruebas unitarias e integración.

4. Recursos y Herramientas

- Ruby on Rails, AWS RDS, ElasticSearch, AWS CloudFormation, AWS Bedrock.

5. Control de Calidad

- Pruebas en cada fase, auditorías de seguridad (OWASP, SOC).

6. Justificación Metodológica

- Selección de tecnologías por seguridad, escalabilidad y cumplimiento normativo.

7. Resultados Esperados

- Módulos seguros y escalables; sistema funcional y validado para producción.

Amazon Bedrock Studio

Resultados

Base de Conocimiento. Estructurada para cubrir diferentes escenarios operativos y regulatorios, proporcionando información detallada sobre excepciones, contribuciones y préstamos dentro de los planes 401K. La tabla a continuación muestra un resumen de los documentos generados y su distribución temática:

Categoría	# Docs	%
Excepciones y Normativas	78	32.8%
Contribuciones	65	27.3%
Préstamos	52	21.8%
Procedimientos Internos	43	18.1%
Otros	0	0%

GuardRails. Limitantes y reglas que moldean las respuestas entregadas por los agentes. Estos son utilizados para evitar que el agente responda con discursos de odio, información fuera de contexto y buscan limitar las alucinaciones en las respuestas.

Objectivos

- Optimizar la gobernanza de datos y procesos internos en la industria del 401K mediante la implementación de soluciones ETL y agentes RAG que mejoren la eficiencia, seguridad y cumplimiento normativo en la gestión de datos sensibles.
- Desarrollar agentes RAG que automaticen procesos internos y mejoren la precisión en respuestas contextuales.
- Aplicar estándares de seguridad de datos según OWASP para la protección de la PII.
- Asegurar el cumplimiento con normativas y auditorías SOC mediante procesos de revisión y pruebas regulares.

Satisfacción de Usuarios Internos. En colaboración con los equipos de operaciones y auditoría, se realizaron pruebas internas para evaluar la satisfacción de los usuarios con el sistema en fase de integración. Se pidió a los usuarios que evaluaran la precisión de las respuestas de los agentes de lenguaje. Se recogieron 387 usos de la herramienta en una población de 23 usuarios, luego de cada uso el usuario puntuaba la herramienta en una escala del 1 al 5

Criterio de Satisfacción	Puntaje Promedio (1-5)
Precisión de respuestas de agentes RAG	4.2
Facilidad de acceso a información	4.5
Claridad en la documentación	4.3
Satisfacción general	4.4

Conclusiones

- En su fase preliminar, el agente presenta un nivel de satisfacción en el cuartil superior, según sus principales usuarios.
- Una base de conocimiento robusta es crucial para la creación del RAG, lo que posibilita un desempeño adecuado del modelo.
- Se encontró que, en esta implementación, los agentes no alcanzan una precisión total. Su rol es de asistencia a un operario humano que tome la decisión final.
- **AWS Bedrock - GuardRails** permitió limitar completamente los discursos de odio en pruebas internas. Sin embargo, estos resultados no son generalizables a otros contextos más vulnerables a *adversive prompting*

DATOS DE CONTACTO DEL AUTOR:

+57 321 754 4505 felipe.rodrigueza@udea.edu.co felipeangel50@gmail.com github.com/felipe-RA linkedin.com/in/felipe-ra-tech/