Pontificia Universidad
JAVERIANA
Cali

[VIGILADA MINEDUCACIÓN Res. 12220 de 2016]

Prototipo para la Clasificación y Análisis de Comunicaciones Telefónicas en el Proceso de Promoción Estudiantil

Sebastián Hernández Triana
Luis Felipe Martínez Munevar

Directora
PhD. Luisa Fernanda Rincón

Pontificia Universidad Javeriana Cali
Facultad de Ingeniería
Ingeniería de Sistemas y Computación
Proyecto de Grado
Julio, 2025

Abstract

The initial interactions between prospective students and the customer service department of Universidad Javeriana Cali constitute a critical stage in the admissions and enrollment process, as they directly influence applicant engagement, the resolution of inquiries, and the institutional image projected. These communications represent a valuable source of information for enhancing university services and for gaining deeper insights into the preferences and expectations of potential students, thereby fostering more personalized experiences.

Nevertheless, a major challenge lies in the systematic analysis of the high volume of calls received and in the effective exploitation of the information they contain. At present, these recordings are stored by an external provider (WeKall) without being further processed, resulting in the underutilization of data with significant strategic potential for institutional development.

In response to this challenge, the present research proposes the development of a tool based on artificial intelligence techniques and natural language processing (NLP). The solution integrates automation workflows and prompts tailored to large language models (LLMs) in order to conduct sentiment analysis, thematic classification, and intention extraction from prospect communications. The outcomes of this analysis will be presented through an interactive dashboard, designed to facilitate the generation of strategic insights, the identification of recurring trends, and the exploration of opportunities to optimize institutional processes and services.

Keywords: Prospective students, customer service, artificial intelligence, large language models (LLM), natural language processing, data processing techniques, automation workflows, prompt engineering, sentiment analysis, intention extraction, dashboards.

Resumen

La comunicación inicial entre los prospectos y el área de atención al cliente de la Universidad Javeriana Cali durante el proceso de ingreso y registro es clave para captar aspirantes, resolver dudas y proyectar una imagen favorable de la institución. Las temáticas abordadas en estas interacciones ofrecen una oportunidad para mejorar los servicios universitarios y comprender las preferencias e intereses de los estudiantes potenciales, favoreciendo así una experiencia más personalizada.

No obstante, uno de los desafíos principales consiste en analizar el gran volumen de llamadas recibidas, así como aprovechar la información extraída de las mismas. Actualmente, la universidad almacena estas grabaciones de interacciones mediante un proveedor externo (*WeKall*), pero permanecen estáticas, sin aprovechar su potencial informativo, lo que representa una pérdida de datos estratégicos que podrían contribuir al desarrollo institucional. En este contexto, el uso de inteligencia artificial y modelos avanzados de procesamiento de lenguaje natural emerge como una solución para analizar y transformar las comunicaciones humanas en información útil y aprovechable.

De esta manera, el objetivo principal de este trabajo de grado es desarrollar una herramienta que aplique técnicas de procesamiento de datos basadas en inteligencia artificial, utilizando flujos de automatización y el diseño de *prompts* dirigidos a modelos de lenguaje de gran escala (*LLM*), para analizar sentimientos, clasificar temáticas y extraer intenciones de las comunicaciones con prospectos. La información obtenida será representada mediante un dashboard interactivo, lo que abre la posibilidad de generar insumos estratégicos, identificar tendencias recurrentes y explorar oportunidades de mejora en los procesos y servicios ofrecidos por la universidad.

Palabras Clave: Prospectos, área de atención al cliente, inteligencia artificial, modelos de lenguaje a gran escala (LLM), procesamiento de lenguaje natural, técnicas de procesamiento de datos, flujos de automatización, prompts, análisis de sentimientos, extracción de intenciones, dashboard.

Índice general

1. Descripción del Problema **13**

1.1. Planteamiento del Problema 13

1.1.1. Formulación 14

1.1.2. Sistematización 14

1.2. Objetivos 15

1.2.1. Objetivo General 15

1.2.2. Objetivos Específicos 15

1.3. Justificación 15

1.4. Delimitaciones y Alcances 17

2. Marco Teórico y Trabajos Relacionados **18**

2.1. Marco de Referencia 18

2.1.1. Áreas Temáticas 18

2.1.2. Marco Teórico 19

2.1.3. Trabajos Relacionados 21

2.1.4. Diferenciadores del Proyecto 23

3. Metodología, Análisis y Diseño **26**

3.1. Metodología 26

3.1.1. Lean Startup 27

3.2. Análisis 29

3.2.1. Descripción del Instrumento Tradicional 29

3.2.2. Proceso de Identificación de Requerimientos 31

3.2.3. Resultados del Levantamiento de Requerimientos 34

4. Implementación **36**

4.1. Exploración de Herramientas 36

4.1.1. Desarrollo Experimental de Técnicas de Análisis de Lenguaje Natural 36

4.1.2. Prototipo experimental de Automatización del Análisis de Llamadas con Flujos 42

4.1.3. Evaluación Comparativa de Enfoques de Análisis de Sentimientos e Intención 44

4.2. Flujo Final de Automatización del Análisis de Llamadas 47

4.2.1. Arquitectura General del Flujo Final 47

4.2.2. Detalle del Flujo *n8n* Implementado 52

4.2.3. Resultado de Ejecución del Flujo Final 62

5. Visualización **66**

5.1. *Dashboard* como Resultado Integrador del Proyecto 66

5.2. Principios de Diseño Aplicados 67

5.2.1. Jerarquía Visual y Organización de la Información 67

5.2.2. Principio de Simplicidad y Reducción de Carga Cognitiva . . . 69

5.2.3. Consistencia Visual y Coherencia Cromática 69

5.3. Justificación de Tipos de Gráficos Seleccionados 70

5.3.1. Gráfico de Barras Apiladas para Análisis de Sentimientos . . . 70

5.3.2. Gráfico de Barras Horizontales para Análisis de Emociones . . 71

5.3.3. Gráfico de Burbujas para Análisis de Intenciones 71

5.3.4. Gráficos de Barras Apiladas para Temáticas y Palabras Clave 72

5.4. Aplicación de Principios de Experiencia de Usuario (UX) 72

5.4.1. Regla de los 5 Segundos 72

5.4.2. Interactividad y Filtros Dinámicos 73

5.5. Consideraciones de Accesibilidad y Usabilidad 73

5.5.1. Diseño Inclusivo 73

5.5.2. Optimización para Diferentes Audiencias 73

5.6. Validación del Diseño mediante Principios Científicos 74

5.6.1. Reducción de Carga Cognitiva 74

6. Evaluación y Validación del Prototipo 75

6.1. Justificación de las Pruebas 75

6.2. Metodología General de Evaluación 76

6.3. Evaluación de la Transcripción 77

6.3.1. Normalización de las Transcripciones 77

6.4. Evaluación del Análisis 80

6.4.1. Creación del Dataset 80

6.4.2. Métricas de Evaluación 82

6.5. Conclusiones de la Evaluación 84

7. Conclusiones 85

8. Trabajo Futuro 87

8.1. Trabajos Futuros Propuestos 87

8.1.1. Integración con el *CRM* de *Salesforce* de la Universidad Javeriana Cali 87

8.1.2. Ampliar el Análisis de Llamadas 88

8.1.3. Inclusión de Más Actividades 88

9. Anexos **95**

Índice de cuadros

4.1. Comparación entre métodos de análisis de lenguaje natural explorados 45

4.2. Distribución de resultados del análisis de llamadas 63

6.1. Evaluación de Precisión del Modelo GPT-4o 83

Índice de figuras

2.1. Organigrama simplificado con las áreas institucionales involucradas en el proyecto. Adaptado a partir del organigrama oficial de la Pontificia Universidad Javeriana Cali [1]. 21

4.1. Distribución de Sentimientos según análisis con *VADER* 37

4.2. Evolución temporal del Sentimiento en la conversación 38

4.3. Frecuencia de Sentimientos clasificados por *VADER* 40

4.4. Frecuencia de Intenciones clasificadas por *DistilRoBERTa* 41

4.5. Flujo automatizado para análisis de llamadas en *n8n* 42

4.6. Nivel 1 Diagrama C4 48

4.7. Nivel 2 Diagrama C4 50

4.8. Nivel 3 Diagrama *C4* 51

4.9. Bloque de carga y lectura del archivo de enlaces en *n8n* 54

4.10. Rama de caso validación corrupta 57

4.11. Rama de caso de grabación insuficiente 60

4.12. Distribución porcentual de los tipos de llamadas analizadas 64

5.1. Dashboard de análisis de comunicaciones telefónicas en el proceso de promoción estudiantil. 67

5.2. Nivel superior 68

5.3. Nivel medio 68

5.4. Nivel inferior 68

6.1. Comparación entre las evaluaciones humanas y las generadas por la *IA* en las 35 llamadas analizadas. 81

Introducción

En el entorno actual de la educación superior, la capacidad de las instituciones para gestionar estratégicamente la información generada en sus interacciones con prospectos resulta fundamental para fortalecer procesos de captación y la experiencia del usuario. Este reto se intensifica en un contexto marcado por la digitalización y la globalización, donde la comunicación es constante, multicanal y cada vez más difícil de analizar de forma manual.

La Pontificia Universidad Javeriana Cali enfrenta un desafío particular: convertir sus comunicaciones telefónicas con aspirantes en una fuente de datos estratégica, que pueda ser interpretada de manera sistemática y utilizada para respaldar decisiones informadas en el área de promoción estudiantil. Actualmente, las grabaciones de estas interacciones —almacenadas por el proveedor *WeKall*— no son aprovechadas de manera efectiva, lo que representa una pérdida significativa de conocimiento potencial.

Este trabajo de grado propone el desarrollo de un prototipo que automatiza el análisis de dichas comunicaciones mediante técnicas de procesamiento de lenguaje natural e inteligencia artificial. El prototipo, basado en flujos automatizados con agentes de *IA*, permite extraer sentimientos, identificar temáticas y deducir intenciones a partir de las transcripciones telefónicas. Estos resultados se presentan a través de un *dashboard* interactivo, diseñado para facilitar su interpretación y apoyar la toma de decisiones estratégicas.

El proceso de desarrollo del prototipo se estructura en varias fases: desde la definición de criterios y selección de técnicas, hasta la implementación y validación en condiciones controladas. Finalmente, se presentan los resultados obtenidos durante la fase de validación, resaltando su aplicabilidad práctica en la promoción institucional y su potencial de replicabilidad en otras instituciones educativas que busquen fortalecer la gestión de sus comunicaciones con prospectos.

Capítulo 1

Descripción del Problema

1.1. Planteamiento del Problema

En el escenario contemporáneo de la educación superior, la gestión estratégica de la información se ha convertido en un factor determinante para la captación, retención y fidelización de estudiantes. Las universidades, enfrentadas a una creciente competencia por atraer talento académico, han desarrollado estructuras complejas de promoción institucional que combinan presencia física en colegios, ferias y eventos con campañas de marketing digital en redes sociales como Instagram, Facebook y YouTube.

En este contexto, la *Pontificia Universidad Javeriana Cali* ha estructurado un proceso de captación que involucra múltiples canales de comunicación, destacándose el papel del área de servicio al cliente, unidad perteneciente a la dirección de promoción estudiantil. Esta área establece contacto directo con los prospectos mediante llamadas telefónicas con el propósito de brindar información, resolver dudas y fomentar la inscripción a programas académicos. A través de estas interacciones se recogen datos valiosos sobre el perfil del aspirante, sus intereses, inquietudes y nivel de compromiso, los cuales podrían constituirse en un insumo estratégico para orientar los procesos institucionales relacionados con la captación de estudiantes.

Sin embargo, a pesar del volumen y riqueza de esta información, se presenta un problema crítico de subutilización de datos. Actualmente, las llamadas son almacenadas por el proveedor externo *WeKall*, pero la universidad no cuenta con un sistema o herramienta que permita analizar, categorizar o extraer conocimiento útil de estas interacciones. En consecuencia, estos datos permanecen inactivos, en forma de "información estática", sin contribuir a la toma de decisiones ni a la retroalimentación de las estrategias de captación, lo cual representa una pérdida significativa de valor informacional.

Este tipo de situación no es exclusiva de esta institución. Diversos estudios han identificado que muchas universidades enfrentan dificultades para integrar técnicas de procesamiento de lenguaje natural (PLN) en sus procesos de gestión institucional, a pesar de su potencial para detectar emociones, analizar temas clave y anticipar intenciones en interacciones verbales o textuales [2, 3]. Esta brecha resulta paradójica, considerando que el PLN, junto con modelos de aprendizaje automático, ha demostrado aplicabilidad en sectores como atención al cliente, salud y comercio, y más recientemente en contextos educativos [4, 5].

En este sentido, surge la necesidad de implementar una solución tecnológica basada en inteligencia artificial que permita automatizar el análisis de las llamadas realizadas desde el área de servicio al cliente; esto es posible a través de flujos orquestados en la herramienta *n8n* y el uso de agentes de IA, con la posibilidad de realizar tareas como análisis de sentimientos, identificación de temas recurrentes e inferencia de intenciones, para posteriormente visualizar estos resultados en un dashboard diseñado para el área de dirección de promoción estudiantil, con el propósito de facilitar el acceso a la información procesada durante los procesos de toma de decisiones.

La implementación de esta solución tiene como propósito aportar una alternativa técnica que facilite la exploración y comprensión de la información recopilada en las llamadas del servicio al cliente. Además, permitirá evaluar su utilidad en el proceso de toma de decisiones por parte del personal del área de dirección de promoción, en concordancia con las necesidades institucionales y los objetivos de transformación digital en la educación superior.

1.1.1. Formulación

La pregunta ahora es: **¿Cómo analizar la información obtenida de las comunicaciones telefónicas del área de promoción estudiantil con los prospectos, para obtener los sentimientos, temáticas e intenciones de las interacciones?**

1.1.2. Sistematización

Para resolver esta pregunta se deben tener en cuenta algunas subpreguntas:

- ¿Cuáles son los criterios necesarios para definir, clasificar y preprocesar la información relevante en el análisis de comunicaciones telefónicas con prospectos?
- ¿Qué enfoques tecnológicos basados en inteligencia artificial pueden aplicarse para analizar sentimientos, identificar temáticas e inferir intenciones de las comunicaciones telefónicas con prospectos?

- ¿Cómo desarrollar una solución tecnológica automatizada que permita el análisis de sentimientos, identificación de temáticas e inferencia de intenciones de las comunicaciones telefónicas con prospectos?
- ¿Qué pruebas de validación deben realizarse para comprobar el correcto funcionamiento y precisión de la categorización de las comunicaciones telefónicas con prospectos?
- ¿Cómo presentar la información categorizada al usuario final?

1.2. Objetivos

1.2.1. Objetivo General

Analizar la información obtenida de las comunicaciones telefónicas del área de promoción estudiantil con los prospectos, utilizando técnicas de procesamiento de datos con inteligencia artificial que permitan extraer y organizar la información.

1.2.2. Objetivos Específicos

1. Definir los criterios necesarios para definir, clasificar y preprocesar la información de las comunicaciones telefónicas con prospectos.
2. Identificar y seleccionar enfoques tecnológicos basados en inteligencia artificial que permitan analizar sentimientos, identificar temáticas e inferir intenciones de las comunicaciones telefónicas con prospectos.
3. Implementar una solución tecnológica automatizada que permita el análisis de sentimientos, identificación de temáticas e inferencia de intenciones de las comunicaciones telefónicas con prospectos.
4. Evaluar la precisión de la información obtenida de comunicaciones telefónicas con prospectos.
5. Proponer un formato adecuado para la visualización de la información categorizada de comunicaciones telefónicas con prospectos.

1.3. Justificación

El presente proyecto de grado se enmarca en la necesidad institucional de fortalecer el uso de los datos disponibles en las interacciones entre el área de *Dirección de Promoción Estudiantil* de la Pontificia Universidad Javeriana Cali y los aspirantes a

sus programas académicos. Actualmente, la universidad cuenta con registros y transcripciones de llamadas telefónicas realizadas por el área de servicio al cliente, que, si bien se almacenan de forma sistemática, no son objeto de un análisis estructurado que permita generar conocimiento a partir de ellos.

La falta de mecanismos para procesar esta información representa una oportunidad de mejora en el proceso de captación de estudiantes, dado que en estas interacciones se expresan elementos clave como las dudas, motivaciones, objeciones y expectativas de los prospectos. El proyecto propone desarrollar un prototipo de análisis basado en técnicas de procesamiento de lenguaje natural e inteligencia artificial que permita extraer patrones relevantes y categorizar las conversaciones de forma automatizada. Esta herramienta buscará servir como apoyo para la interpretación de tendencias comunicativas, sin reemplazar los procesos actuales de evaluación humana, sino complementándolos.

Desde una perspectiva institucional, el desarrollo del prototipo puede aportar insumos útiles para ajustar las estrategias de contacto, priorizar recursos y tomar decisiones informadas respecto a la interacción con los aspirantes. Si bien el proyecto no tiene como objetivo medir directamente el impacto en los indicadores de captación, sí se plantea generar visualizaciones comprensibles y estructuradas que faciliten la exploración de los datos y la identificación de áreas de interés para el equipo de promoción.

En el plano social, esta iniciativa puede contribuir a una experiencia de contacto más pertinente para los aspirantes, en tanto se reconozcan sus preocupaciones e intereses a través del análisis automatizado. Asimismo, puede ser replicable por otras instituciones de educación superior que enfrenten desafíos similares en el uso de datos conversacionales, aportando al ecosistema educativo herramientas para una gestión más informada de sus procesos.

Desde el punto de vista teórico, el proyecto explora la integración de capacidades de procesamiento de lenguaje natural (PLN) mediante el uso de modelos de lenguaje de gran escala (LLM), aplicados a tareas de análisis de sentimientos, clasificación de temáticas e inferencia de intenciones en un dominio específico como lo es la promoción educativa. De este modo, se contribuye al estudio de la aplicación de herramientas de PLN basadas en LLM en contextos institucionales reales, a partir del procesamiento de datos no estructurados provenientes de conversaciones humanas.

Finalmente, la propuesta resulta viable dentro del alcance del trabajo de grado, en tanto se cuenta con el acceso a datos reales (transcripciones), la disponibilidad de herramientas tecnológicas libres (como *n8n* y modelos preentrenados), y un entorno institucional dispuesto a recibir y validar los resultados en pruebas controladas. Esta validación preliminar permitirá evaluar el desempeño técnico del prototipo y su utilidad percibida por los usuarios institucionales.

1.4. Delimitaciones y Alcances

El alcance principal de este proyecto consiste en el desarrollo de un prototipo de transcripción, análisis y visualización de la información obtenida a partir de grabaciones de llamadas telefónicas entre el área de servicio al cliente de la Pontificia Universidad Javeriana Cali y los prospectos. Dichas grabaciones son proporcionadas por el proveedor externo *WeKall*. La propuesta contempla la implementación de un flujo de análisis automatizado que, mediante técnicas de procesamiento de lenguaje natural (*PLN*), permita extraer elementos como sentimientos, temáticas e intenciones, cuya representación será estructurada a través de un *dashboard* interactivo.

Este tablero de visualización será diseñado como entregable final del proyecto; sin embargo, su validación con usuarios finales del área de *Dirección de Promoción Estudiantil* queda por fuera de los objetivos planteados. En su lugar, se propone realizar una validación preliminar de los resultados iniciales del análisis automatizado con personal del área de tecnología de la universidad. La validación formal del *dashboard* con usuarios finales no forma parte del alcance definido para el presente proyecto.

Cabe aclarar que el proyecto no contempla la integración directa con sistemas de producción institucionales como *Salesforce*, ni el desarrollo de módulos especializados para la gestión de seguridad, encriptación o anonimización de datos confidenciales. De igual forma, no se realizará una evaluación comparativa con sistemas previos, ni se medirá de forma cuantitativa el impacto de la visualización en los indicadores institucionales.

En síntesis, este proyecto constituye una prueba de concepto que busca demostrar la viabilidad técnica y funcional de la solución propuesta mediante su validación preliminar con usuarios finales, dentro del contexto específico del área de promoción de la universidad.

Capítulo 2

Marco Teórico y Trabajos Relacionados

2.1. Marco de Referencia

Dada la naturaleza interdisciplinaria del proyecto, este explora la intersección entre tecnologías emergentes, especialmente inteligencia artificial y procesamiento de lenguaje natural, y su aplicación en contextos educativos y de comunicación institucional. En este marco, se consideran métodos automatizados para la extracción y análisis de información a partir de grabaciones telefónicas, destacando su potencial para generar retroalimentación institucional, comprender tendencias e interpretar preferencias de los aspirantes. A continuación, se desarrollan conceptos teóricos clave sobre inteligencia artificial, procesamiento de lenguaje natural y tecnologías educativas que fundamentan la propuesta presentada.

2.1.1. Áreas Temáticas

De acuerdo con el sistema de clasificación computacional ACM [6], las áreas temáticas que abarca el proyecto son:

- Information Systems - Information Retrieval - Retrieval models and ranking.
- Information Systems - World Wide Web - Web Services.
- Human-centered Computing - Visualization - Visualization Application Domains - Visual Analytics.

2.1.2. Marco Teórico

Conceptos Computacionales

Una comprensión precisa de las tecnologías computacionales involucradas permitirá al equipo de trabajo realizar una implementación adecuada del proyecto planteado. Esto incluye la selección y utilización de herramientas y técnicas concretas, alineadas con las necesidades del análisis propuesto.

- **Inteligencia Artificial:** Es un conjunto de tecnologías que permiten a las computadoras realizar funciones avanzadas como comprender, traducir lenguaje, analizar datos y hacer recomendaciones, generando valor en diversos contextos tanto personales como empresariales [7].
- **Procesamiento de Lenguaje Natural:** Subcampo de la inteligencia artificial que emplea técnicas de "Machine Learning" para que las computadoras comprendan, interpreten, generen texto e imiten voz humana mediante la combinación de modelos lingüísticos basados en reglas, modelado estadístico y aprendizaje profundo [8].
- **Análisis de Sentimientos:** es una característica que hace parte de una colección de algoritmos de aprendizaje automático que involucra lenguaje natural. Permite identificar qué piensa la gente sobre una temática en concreto usando en su mayoría etiquetas predefinidas como "negativo", "neutral", "positivo"] para catalogar la información [9].
- **Modelos de Lenguaje de Gran Escala (LLMs):** Son modelos de inteligencia artificial entrenados con grandes volúmenes de datos textuales para comprender y generar lenguaje humano. Estos modelos permiten tareas como clasificación, resumen, análisis de sentimientos, extracción de información y diálogo conversacional con altos niveles de coherencia y contextualización [10].
- **Prompt**: Es una entrada o instrucción diseñada para activar o guiar el comportamiento de un sistema interactivo. En el ámbito de la computación, un prompt puede adoptar diversas formas —como comandos, preguntas o estructuras de entrada— y su propósito es desencadenar una acción específica del sistema en respuesta a la solicitud del usuario [11].
- **Flujos de Automatización:** Consiste en el uso de herramientas tecnológicas para automatizar tareas repetitivas dentro de un proceso más amplio, disminuyendo en lo posible la intervención humana. Esta práctica busca optimizar y agilizar operaciones mediante software [12].
- **Dashboard:** Es una colección de vistas que permite visualizar y comparar diferentes datos de forma simultánea. [13].
- **Dataset:** Es una colección de datos que puede presentarse en forma de hoja de cálculo, tabla o base de datos, incluyendo también colecciones de notas de

estudios cualitativos, videos o imágenes tomadas como parte de un proyecto de investigación [14].

- **Benchmark:** Es una técnica utilizada para comparar el rendimiento, calidad o eficacia de un sistema, modelo o proceso frente a un estándar o referente establecido [15].

Conceptos No-Computacionales

Asimismo, para entender el entorno donde se establece la solución tecnológica propuesta, necesitamos incorporar ideas de campos ajenos a las disciplinas computacionales.

- **Prospecto:** Es un término utilizado para referirse a un cliente potencial, es decir, una persona que ha mostrado interés en un producto o servicio y que podría convertirse en cliente a través de estrategias de seguimiento y persuasión [16].
- **Retroalimentación Institucional:** Es el proceso mediante el cual una persona que recibe un servicio comunica su nivel de satisfacción. Se basa en hechos y resultados esperados, con el objetivo de mejorar el desempeño de los profesionales y la calidad del servicio prestado [17].
- **Centro de Atención Telefónica:** También conocido como *call center*, es un centro de llamadas donde agentes atienden a clientes por teléfono para vender productos, brindar soporte o realizar encuestas, facilitando así una comunicación directa entre la organización y sus usuarios [18].
- **WeKall:** Es una plataforma de telefonía en la nube diseñada para centralizar, organizar y supervisar las comunicaciones telefónicas de una empresa a través de internet. Permite realizar y recibir llamadas desde múltiples dispositivos, grabar interacciones, monitorear en tiempo real, generar reportes detallados y conectarse con sistemas externos como *CRMs*, mejorando así la eficiencia operativa y la experiencia del cliente. [19]
- **Salesforce:** Es una plataforma de gestión de relaciones con el cliente (CRM) basada en la nube que centraliza la información de ventas, marketing y servicio al cliente. Permite a las organizaciones automatizar procesos, analizar datos y mejorar la interacción con sus clientes mediante herramientas personalizables e integraciones con múltiples servicios. [20]

Contexto Organizacional

Para comprender el contexto donde se implementa la solución tecnológica propuesta, resulta pertinente conocer la estructura organizacional de la Pontificia Universidad Javeriana Cali. Esto permite identificar las dependencias institucionales

relacionadas con el proyecto, así como entender la interacción entre las áreas implicadas en los procesos.

En la Figura 2.1 se muestra el organigrama simplificado, destacando las áreas institucionales involucradas en este proyecto.

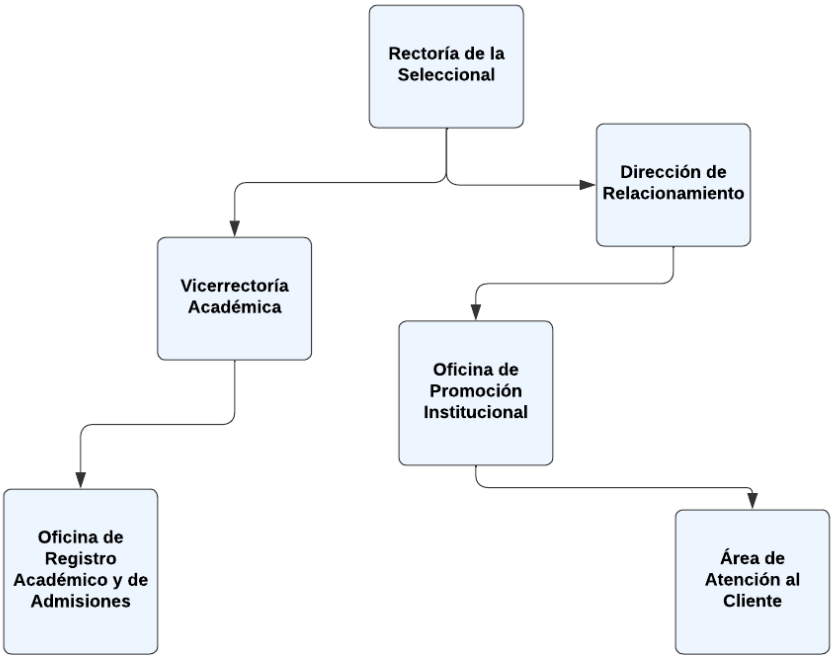

Figura 2.1: Organigrama simplificado con las áreas institucionales involucradas en el proyecto. Adaptado a partir del organigrama oficial de la Pontificia Universidad Javeriana Cali [1].

Como se observa en la Figura 2.1, la Dirección de Relacionamiento (*DREL*) se encuentra articulada directamente con la Rectoría de la Seccional, y a su vez establece vínculos operativos con la Oficina de Promoción Institucional y el Área de Atención al Cliente. Esta dirección es la encargada de coordinar acciones orientadas a fortalecer el relacionamiento de la universidad con su entorno, en especial con los aspirantes y otros públicos estratégicos. En este marco, la Oficina de Promoción Institucional y el Área de Atención al Cliente operan como unidades de apoyo en los procesos de comunicación, orientación y acompañamiento institucional.

2.1.3. Trabajos Relacionados

En esta sección se presenta una selección de investigaciones previas que comparten elementos temáticos, metodológicos o tecnológicos con el presente proyecto.

El objetivo es contextualizar el trabajo dentro del estado del arte en el análisis de datos provenientes de comunicaciones verbales, el uso de inteligencia artificial en el entorno educativo y la aplicación de herramientas de procesamiento de lenguaje natural (PLN) y visualización de datos para la toma de decisiones.

Uso de PLN para el análisis de sentimientos en programas educativos

El estudio de Pham et al. (2020) [4] aplica PLN para analizar emociones y actitudes expresadas por estudiantes en redes sociales dentro de un programa de posgrado. Mediante la clasificación de más de 300 publicaciones, se obtuvieron patrones que reflejan la percepción de los estudiantes sobre evaluaciones y actividades. Este enfoque resulta pertinente para nuestro proyecto al compartir la finalidad de extraer sentimientos y temáticas desde datos textuales vinculados a contextos educativos.

Revisión sistemática sobre el uso de PLN en retroalimentación estudiantil

La revisión de Sunar y Khalid (2024) [2] analiza el uso de PLN para interpretar retroalimentación textual de estudiantes a instructores. Se categorizaron metodologías y herramientas orientadas a extraer patrones de opinión, lo cual contribuye a comprender mejor la experiencia del estudiante. Esta revisión es relevante al validar la aplicación de herramientas de PLN en contextos de mejora institucional.

Clasificación de texto en encuestas de satisfacción

Álvarez (2021) [21] presenta una aplicación de PLN para categorizar encuestas en el sector retail mediante modelos como FastText y BERT. El estudio evidencia cómo estos modelos permiten generar resúmenes extractivos y etiquetas múltiples con alta precisión, técnicas que también son de utilidad para el procesamiento de transcripciones telefónicas en nuestro proyecto.

Procesamiento automatizado de respuestas abiertas en encuestas de opinión

Martínez et al. (2025) [22] exploran la clasificación de respuestas abiertas en encuestas de opinión pública mediante Random Forest. La investigación demuestra cómo el PLN permite reducir la ambigüedad al trabajar con datos no estructurados, aportando estrategias que se alinean con el objetivo de categorizar las comunicaciones entre prospectos y la universidad.

Clasificación y predicción mediante técnicas de machine learning

Torres (2023) [23] aplica K-Means y regresión logística para clasificar y predecir deserciones de clientes. Esta combinación metodológica permite segmentar y anticipar comportamientos, lo cual guarda relación con nuestro objetivo de inferir intenciones y niveles de compromiso de los prospectos.

Sistemas automatizados de quejas universitarias con PLN

Hossen et al. (2022) [3] proponen un sistema automatizado de gestión de quejas universitarias, aplicando PLN y aprendizaje automático para clasificar entradas textuales. El sistema mejora la respuesta institucional y es relevante por su aplicabilidad directa al entorno universitario en el que se sitúa nuestro trabajo.

Reconocimiento emocional en el habla humana

Zhang y Ng (2018) [24] investigan cómo los clasificadores KNN y SVM pueden utilizarse para detectar sentimientos en el habla mediante características acústicas y lingüísticas. Su enfoque refuerza la viabilidad de interpretar emociones en las llamadas telefónicas que son analizadas en nuestro prototipo.

Análisis automatizado de centros de llamadas

IBM (2020) [5] presenta un enfoque de análisis de sentimientos mediante PLN aplicado a centros de llamadas. El estudio subraya el valor de transformar datos de voz en información estratégica para la mejora del servicio, validando la utilidad de los modelos que proponemos.

Esta revisión de literatura confirma que el presente proyecto se apoya en una base sólida de investigaciones previas, al tiempo que extiende sus aplicaciones al contexto específico del análisis de comunicaciones de admisión universitaria mediante herramientas automatizadas y visuales.

2.1.4. Diferenciadores del Proyecto

Principales Aportes del Prototipo

- **Enfoque en promoción institucional:** A diferencia de los proyectos orientados a gestión de quejas [3] o la evaluación del sentimiento estudiantil en entornos académicos [4], este prototipo se centra en analizar las interacciones tempranas con prospectos durante la promoción estudiantil. Este enfoque se

caracteriza por identificar intereses y opiniones previas al ingreso, que puede brindar una retroalimentación estratégica en etapas tempranas del proceso de admisión.

- **Uso de modelos de lenguaje preentrenados (LLMs):** A diferencia de enfoques que requieren entrenar modelos personalizados, como el propuesto por Mohamed et al., donde se emplean clasificadores supervisados como *Multinomial Naive Bayes* [3], el presente prototipo se apoya en modelos de lenguaje ya entrenados que pueden adaptarse a diferentes tareas mediante instrucciones en lenguaje natural. Esto permite evitar procesos complejos de recolección y curación de datos, y facilita una implementación más ágil y adaptable a distintos contextos institucionales.

En cuanto a similitudes conceptuales, el proyecto guarda mayor similitud con el proyecto [3], donde también se emplea procesamiento de lenguaje natural para mejorar procesos universitarios. No obstante, el presente prototipo se enfoca en una etapa previa, centrada en la interacción con prospectos y no en la gestión de quejas, lo que representa una aplicación distinta orientada a la promoción institucional.

Limitaciones y Áreas de Mejora del Prototipo

Si bien el prototipo muestra potencial para apoyar el proceso de toma de decisiones y su implementación no implica una gran complejidad, es importante tener en cuenta las siguientes limitaciones y oportunidades de mejora:

- **Dependencia de Servicios en la Nube:** Como se ha identificado en estudios previos, como el de Hossen et al. [3], la dependencia de APIs de terceros para el procesamiento de lenguaje natural en el prototipo puede acarrear limitaciones en el acceso a funcionalidades avanzadas o incurrir en costos asociados a largo plazo.
- **Riesgo de seguridad y privacidad de la información:** El uso de modelos de lenguaje de gran escala (*LLMs*), como los ofrecidos por *OpenAI*, requiere enviar texto a servidores externos para su procesamiento. Esto implica exponer potencialmente información sensible —como nombres, correos, datos académicos o detalles operativos internos— a terceros. Uno de los principales riesgos asociados es el de ataques de extracción (*extraction attacks*), donde un atacante puede inducir al modelo a revelar fragmentos de información confidencial que haya sido procesada previamente, incluso si no fue almacenada explícitamente [25]. Además, debido a que estos modelos funcionan como servicios tipo *black box* —es decir, sistemas cuyo funcionamiento interno no es visible ni accesible para el usuario final—, no es posible auditar de forma transparente qué datos son conservados temporalmente, cómo son utilizados para futuras mejoras del modelo, o si están sujetos a retención accidental en sistemas de caché. Por tanto, la implementación de este tipo de soluciones requiere no solo

evaluar técnicamente la capacidad del modelo, sino también establecer políticas claras sobre anonimización de datos, uso de entornos seguros y selección de proveedores con garantías adecuadas de privacidad.

Sin embargo, en el contexto específico del uso de la API de OpenAI para el procesamiento de transcripciones telefónicas, es importante considerar las implicaciones del manejo de datos según las políticas vigentes del proveedor. Conforme a los términos de servicio de OpenAI, los datos enviados a través de su API no son utilizados para entrenar o mejorar sus modelos, y las transcripciones procesadas no se almacenan permanentemente en sus servidores una vez completada la solicitud [26]. No obstante, existe un período de retención temporal de 30 días con fines de monitoreo de abuso y seguridad, durante el cual la información permanece accesible para el proveedor [27].

Para el contexto universitario, esto implica que información como nombres de prospectos, referencias a programas académicos específicos, menciones de costos de matrícula y detalles sobre procesos de admisión quedan temporalmente expuestos a un tercero. Aunque OpenAI implementa cifrado en tránsito y en reposo, y cuenta con certificaciones de seguridad como SOC 2 Type 2 [28], el envío de datos sensibles institucionales a servicios externos representa inherentemente un riesgo de exposición que debe ser evaluado frente a los beneficios del análisis automatizado. En implementaciones futuras orientadas a producción, sería recomendable considerar la implementación de técnicas de anonimización previa o el uso de modelos de lenguaje desplegados localmente para minimizar la exposición de información confidencial.

Capítulo 3

Metodología, Análisis y Diseño

3.1. Metodología

Durante el desarrollo de este trabajo de grado se adoptó una estrategia metodológica flexible, centrada en la iteración continua y la validación de entregables del flujo funcional. En lugar de seguir una metodología formal como Scrum o XP, se tomó como referencia conceptual el conjunto de principios expuestos en el Manifiesto *Ágil* [29], aplicándolos de forma contextualizada a un entorno universitario y de investigación.

Esta adaptación permitió fomentar la interacción con los usuarios institucionales, minimizar tiempos de desarrollo improductivo y favorecer la adaptabilidad ante requerimientos emergentes. Los valores fundamentales tomados en cuenta incluyeron:

- **Colaboración y comunicación abierta**: Se mantuvo contacto con los miembros del área de atención al cliente y del área de Promoción Estudiantil, quienes contribuyeron en la definición y validación de requerimientos.
- **Entrega incremental de funcionalidad**: Cada iteración se enfocó en entregar una parte del prototipo funcional, evaluable y refinable con base en datos reales y retroalimentación.
- **Simplicidad documental**: Se priorizó la comprensión directa del flujo construido en las herramientas exploradas, por encima de una documentación, centrando los esfuerzos en entregables tangibles.
- **Capacidad de adaptación**: Las decisiones relacionadas con el uso de herramientas, formato de salida y estructura de análisis fueron ajustadas conforme se detectaron oportunidades de mejora en el ciclo de desarrollo.

Asimismo, se estableció comunicación con la Oficina de Promoción Institucional para gestionar el acceso a las grabaciones de llamadas utilizadas en el análisis, lo cual se formalizó mediante la firma de un acuerdo de confidencialidad (véase Anexo 9).

El proceso de trabajo se organizó en ciclos cortos con fases claramente definidas:

1. **Exploración y levantamiento de necesidades**: Se realizaron entrevistas para identificar puntos críticos en la atención a prospectos y entender el contexto operativo.
2. **Construcción de prototipos funcionales**: Se desarrolló un prototipo funcional mediante la evaluación comparativa de distintos métodos de análisis de sentimiento y transcripción de llamadas, empleando diversas herramientas. Estas pruebas se realizaron con grabaciones simuladas de llamadas, permitiendo observar el comportamiento de cada enfoque frente a las diferentes características de las llamadas institucionales. A partir de los resultados obtenidos, se seleccionó el método más adecuado para integrarlo en la solución final.
3. **Validación técnica y funcional**: Se llevaron a cabo pruebas controladas con datos reales para verificar la coherencia y relevancia de la salida generada.
4. **Iteración y mejora**: Se realizaron ajustes en las instrucciones, procesos internos y formato de exportación según la retroalimentación obtenida.
5. **Visualización de resultados**: Se preparó una presentación comprensible de la información mediante herramientas como *Tableau*, facilitando su uso institucional.

Como resultado de esta estrategia metodológica, fue posible construir un prototipo de manera progresiva, mediante ciclos iterativos que facilitaron su ajuste continuo. Este enfoque permitió incorporar retroalimentación directa de los usuarios institucionales, mejorar la calidad de los entregables en cada iteración y asegurar que las funcionalidades desarrolladas respondieran a los objetivos definidos por el área de Promoción Estudiantil.

3.1.1. Lean Startup

El proceso de desarrollo también se estructuró siguiendo los principios del enfoque *Lean Startup*, propuesto por Eric Ries [30]. Esta metodología se basa en la aplicación de ciclos iterativos y validados de construcción, medición y aprendizaje (*Build–Measure–Learn*), especialmente útiles en contextos donde no existe una validación externa inmediata o continua. Lean Startup busca reducir la incertidumbre mediante el uso de métricas accionables y la validación empírica de decisiones de desarrollo.

Este enfoque resultó particularmente pertinente en el contexto de este proyecto, donde el avance dependió de pruebas controladas, evaluaciones internas por parte del equipo de trabajo, y el acompañamiento constante del director de tesis como figura orientadora. Las iteraciones permitieron validar rápidamente funcionalidades clave, realizar ajustes tempranos y mantener un enfoque progresivo.

- **Construir (*Build*)**: Esta fase inicial consistió en el desarrollo de flujos funcionales mediante herramientas como *n8n*, los cuales integraron modelos de procesamiento de lenguaje natural como *VADER* (para análisis de sentimientos) y *GPT-4o* (para inferencia de intenciones y temas). Cada versión del prototipo fue diseñada con un alcance específico, facilitando la evaluación puntual de componentes del mismo.
- **Medir (Measure)**: En esta fase se procedió a ejecutar los flujos usando transcripciones reales del entorno institucional. Se analizaron los datos generados, tanto en su estructura como en su contenido semántico, verificando que los resultados se alinearan con las categorías establecidas inicialmente. Las métricas de interés incluyeron la calidad de la transcripción, la coherencia entre sentimiento inferido y justificaciones textuales, y el formato estructurado del archivo generado.
- **Aprender (*Learn*)**: Los hallazgos extraídos del análisis interno y de las sesiones de revisión semanal con el director de tesis permitieron reorientar aspectos críticos de cada versión del prototipo. Esto permitió realizar ajustes a los prompts enviados a los modelos de *IA*, el re-diseño de la distribución de nodos dentro del flujo, además de ajustar el formato de exportación de resultados para su posterior visualización en *Tableau*. Este aprendizaje fue acumulativo y retroalimentó cada ciclo de construcción.

Este ciclo se aplicó de forma reiterativa a lo largo del desarrollo, consolidándose como una estrategia enfocada en el avance de una manera fundamentada sin depender de validaciones externas frecuentes. La aplicación de *Lean Startup* no solo facilitó la toma de decisiones basada en evidencia, sino que también contribuyó a mantener el enfoque en objetivos tangibles y medibles, alineados con el planteamiento del proyecto y las expectativas institucionales.

3.2. Análisis

Esta sección presenta el análisis del proceso de levantamiento de requisitos enfocado exclusivamente en la gestión de prospectos dentro del área de promoción estudiantil de la Pontificia Universidad Javeriana Cali. En lugar de abordar todo el funcionamiento del área, se delimita el análisis al proceso actual de seguimiento a aspirantes mediante llamadas telefónicas, incluyendo la forma en que se registran las interacciones y se evalúa el interés percibido.

3.2.1. Descripción del Instrumento Tradicional

En la Pontificia Universidad Javeriana Cali, el proceso de análisis de comunicaciones telefónicas con prospectos se realiza actualmente mediante metodologías tradicionales que dependen principalmente de revisiones manuales y evaluaciones subjetivas por parte del personal del área de promoción estudiantil.

El instrumento tradicional consiste en un enfoque basado en la experiencia del personal del área de atención al cliente, quienes después de cada llamada registran observaciones generales sobre la interacción en sistemas de CRM como Salesforce. Este proceso incluye la captura manual de información básica como:

- Datos demográficos del prospecto (nombre, teléfono, programa de interés).
- Estado de la llamada (contactado, no contactado, reagendar).
- Observaciones generales sobre el nivel de interés percibido.
- Seguimientos requeridos según criterio del agente.

Las grabaciones de las llamadas se almacenan en la plataforma *WeKall*, un proveedor externo de telefonía en la nube, pero permanecen como "información estática" sin procesamiento adicional. El análisis de estas grabaciones se realiza únicamente de forma esporádica y manual.

Limitaciones del Instrumento Actual:

El sistema tradicional presenta varias limitaciones que impactan la efectividad del proceso de captación:

1. **Subjetividad en la evaluación:** La clasificación del interés y sentimiento del prospecto depende exclusivamente de la percepción individual del agente, introduciendo variabilidad en la evaluación.

2. **Falta de estandarización:** No existen criterios unificados para evaluar aspectos como el nivel de interés, las temáticas de mayor relevancia o las áreas de mejora en el proceso de atención.

3. **Información no aprovechada:** Las transcripciones y grabaciones almacenadas en WeKall no se analizan de manera sistemática, perdiendo oportunidades de obtener insights valiosos.

4. **Ausencia de métricas objetivas:** No se generan indicadores cuantitativos que permitan evaluar la efectividad de las estrategias de comunicación o identificar patrones en el comportamiento de los prospectos.

5. **Capacidad limitada de escalamiento:** El análisis manual no es viable para procesar grandes volúmenes de llamadas de manera eficiente.

6. **Tiempo de análisis:** El tiempo requerido para extraer los datos relevantes de cada llamada, pueden ser extremadamente largos debido a la cantidad de llamadas.

Esta metodología tradicional, aunque funcional para el registro básico de interacciones, no proporciona el nivel de análisis detallado necesario para optimizar las estrategias de captación y mejorar la experiencia del prospecto.

3.2.2. Proceso de Identificación de Requerimientos

El proceso de identificación de requerimientos se llevó a cabo siguiendo una metodología participativa que incluyó múltiples fuentes de información y técnicas de recolección de datos. Este proceso se estructuró en las siguientes fases:

Entrevistas con Actores Clave

Con el fin de comprender a profundidad el funcionamiento operativo del proceso de comunicación con prospectos, se realizó una entrevista semiestructurada con la Coordinadora de Servicio al Cliente de la universidad.

Durante la entrevista, la coordinadora explicó el flujo general de interacción entre las distintas áreas involucradas. El proceso inicia con estrategias de captación diseñadas por el área de Promoción Estudiantil, las cuales pueden ser virtuales (formularios en redes sociales o la página web) o presenciales (eventos como *Open House*, visitas a colegios, ferias universitarias, entre otros).

Una vez que un prospecto completa un formulario, su información es almacenada en el sistema institucional (*Salesforce*) y, en un plazo aproximado de un día hábil, un agente del área de atención al cliente contacta telefónicamente al estudiante para continuar el proceso de inscripción. En caso de confirmar el interés, el prospecto es remitido al área de Registro.

También se indicó que, si bien existe un formato base para orientar las llamadas, los agentes pueden adaptarlo según el desarrollo de cada conversación. Adicionalmente, las grabaciones de estas interacciones quedan almacenadas por el proveedor externo *WeKall*.

Rol:

- Coordinadora de servicio al cliente

Fecha y duración:

- 4 de febrero de 2025
- Duración aproximada: 16 minutos

Temas Explorados:

- Flujo operativo del proceso de comunicación con prospectos.
- Criterios utilizados para evaluar el interés de los candidatos.

- Principales desafíos en la gestión de información de llamadas.
- Expectativas sobre herramientas de análisis automatizado.
- Métricas consideradas relevantes para la toma de decisiones.

Preguntas destacadas de la entrevista:

- ¿Cuáles son los principales objetivos del área de atención al cliente en el proceso de captación de estudiantes?
- ¿Cómo es el flujo de trabajo típico de una llamada con un prospecto?
- ¿Cuáles son los criterios utilizados para determinar el éxito de una llamada?
- ¿Qué tipo de preguntas o temáticas suelen hacer los prospectos durante una llamada?
- ¿Existen guiones predefinidos para las llamadas o los asesores tienen libertad en la conversación?
- ¿Considera que el personal del área de atención al cliente podría beneficiarse de un prototipo que analice automáticamente las llamadas para identificar patrones y sentimientos?

Reunión con Actores Técnicos Institucionales

Complementando la perspectiva anterior, se sostuvo una reunión con actores técnicos institucionales con el objetivo de compartir los avances del proyecto y recibir retroalimentación técnica:

Roles:

- Coordinador de Análisis de Datos de la Dirección de Relacionamiento
- Ingeniero de Proyectos del Centro de Servicios Informáticos

Fecha y duración:

- 19 de junio de 2025
- Duración aproximada: 1 hora

Durante la sesión se presentaron los principales resultados obtenidos hasta el momento, incluyendo el análisis de 109 llamadas reales y una versión preliminar del *dashboard* interactivo. Los participantes manifestaron interés en los hallazgos del análisis y sugirieron incorporar el almacenamiento de identificadores únicos de llamadas para mejorar la trazabilidad de los registros.

Además, expresaron que sería deseable, a futuro, contar con una integración entre los resultados del análisis y el sistema institucional *Salesforce*, con el fin de aprovechar dicha información en otros procesos administrativos. Esta posibilidad se contempla como una línea de trabajo futura, dado que excede los alcances definidos para el presente proyecto.

Reunión de Presentación de Resultados

Como parte del cierre del proyecto, se llevó a cabo una reunión institucional en la que se presentaron los resultados del prototipo desarrollado a representantes de distintas áreas vinculadas con la gestión de datos, tecnología, mercadeo y procesos académicos de la universidad.

Propósito de la reunión:

- Verificar la viabilidad técnica del prototipo para una posible implementación futura.
- Presentar los análisis realizados y los *dashboards* generados.
- Compartir los hallazgos obtenidos durante el desarrollo y ejecución del proyecto.
- Recibir retroalimentación institucional para orientar posibles líneas de acción futuras.

Invitados previstos:

- Directora del Centro de Servicios Informáticos.
- Jefe del CRM del Centro de Servicios Informáticos.
- Ingeniero de Proyecto de Desarrollo TI.
- Coordinador de Análisis de Datos de la Dirección de Relacionamiento.
- Coordinadora de Servicio al Cliente.
- Jefe del DREL e Innovación TI del Centro de Servicios Informáticos.
- Directora de Mercadeo.

La reunión permitió presentar los resultados alcanzados, incluyendo una explicación del proceso desarrollado y los hallazgos derivados del análisis de 3.238 llamadas. Se mostraron gráficos sobre el estado de las comunicaciones y se informó acerca de la situación actual del servicio de almacenamiento de grabaciones. Esta información fue recibida de manera positiva por los asistentes, quienes a partir de los datos presentados pudieron realizar sus propios análisis y reflexiones. El espacio favoreció una visión compartida sobre el potencial del proyecto y abrió la discusión hacia posibles ajustes y funcionalidades que podrían desarrollarse en fases posteriores.

Análisis de Documentación Institucional

Como insumo adicional, se revisó la documentación existente relacionada con los procesos de promoción estudiantil, incluyendo:

- Configuraciones actuales en Salesforce.
- Información registrada de los prospectos en Salesforce.
- Políticas de manejo de información de candidatos.

3.2.3. Resultados del Levantamiento de Requerimientos

A partir del proceso de análisis se identificaron los siguientes requerimientos funcionales:

RF01 - Procesamiento Automatizado de Transcripciones: El prototipo debe procesar automáticamente las grabaciones de llamadas almacenadas en *WeKall*, realizando transcripción automática y aplicando técnicas de procesamiento de lenguaje natural para extraer información relevante.

RF02 - Análisis de Sentimientos: El prototipo debe clasificar el sentimiento general de cada llamada como "positivo", "negativo" o "neutral", identificando las emociones predominantes tanto del prospecto como del agente.

RF03 - Identificación de Temáticas: El prototipo debe identificar automáticamente las temáticas principales discutidas durante la llamada, categorizándolas según áreas como: proceso de admisión, costos, becas, programas académicos, modalidad de estudio.

RF04 - Inferencia de Intenciones: El prototipo debe determinar si el prospecto muestra intención de continuar con el proceso de inscripción, basándose en indicadores extraídos de la conversación.

RF05 - Generación de Recomendaciones: El prototipo debe proporcionar sugerencias específicas para mejorar la calidad de la atención, identificando áreas de

oportunidad en el proceso de comunicación.

RF06 - Extracción de Palabras Clave: El prototipo debe identificar y extraer las palabras clave más relevantes de cada conversación para facilitar búsquedas y análisis posteriores.

RF07 - Registro de Preguntas del Prospecto: El prototipo debe capturar y listar todas las preguntas formuladas por el prospecto durante la llamada para identificar inquietudes comunes.

RF08 - Dashboard Interactivo: El prototipo debe proporcionar una interfaz de visualización que permita al usuario explorar los resultados del análisis de manera intuitiva.

RF09 - Exportación de Resultados: El prototipo debe generar archivos estructurados que contengan los resultados del análisis para su uso en herramientas externas.

Esta fase de análisis proporcionó la base para el diseño e implementación del prototipo, con el objetivo de que la solución desarrollada se ajuste a las necesidades identificadas del área de promoción estudiantil y pueda integrarse en un futuro en sus procesos de gestión de información.

Capítulo 4

Implementación

4.1. Exploración de Herramientas

La implementación de esta solución requirió un proceso de experimentación y descubrimiento para identificar las herramientas más apropiadas para el análisis automatizado de llamadas telefónicas en el contexto institucional de la Pontificia Universidad Javeriana Cali. Durante esta fase exploratoria, se evaluaron múltiples alternativas tecnológicas, desde diferentes enfoques de procesamiento de lenguaje natural y transcripción de audio hasta la evaluación de una plataforma de automatización de flujos de trabajo.

El objetivo central fue determinar qué combinación de herramientas permitiría extraer sentimientos, temáticas e intenciones de las conversaciones entre el área de promoción y los prospectos estudiantiles. A continuación se documenta este proceso iterativo de pruebas, análisis y refinamiento que condujo a la solución final implementada.

4.1.1. Desarrollo Experimental de Técnicas de Análisis de Lenguaje Natural

Análisis de Sentimientos con *VADER*

Como primer paso en la exploración de técnicas de análisis de lenguaje natural, se realizó una implementación experimental utilizando exclusivamente el modelo *VADER (Valence Aware Dictionary and Sentiment Reasoner)*. Este enfoque fue seleccionado debido a su naturaleza basada en reglas y lexicones que elimina la necesidad de datos de entrenamiento etiquetados, así como por su eficacia demostrada en el análisis de retroalimentación de clientes y servicios empresariales [31], características que lo hacen especialmente adecuado para prototipos que requieren

procesamiento eficiente de textos breves y comunicaciones institucionales [32].

VADER es un analizador léxico basado en reglas heurísticas y lexicones, que combina un diccionario especializado con consideraciones gramaticales y sintácticas para expresar la intensidad del sentimiento [32]. Utiliza una métrica central denominada ***compound***, que combina las puntuaciones positivas, negativas y neutrales de un fragmento de texto en un único valor normalizado que oscila entre -1 (muy negativo) y +1 (muy positivo) [32]. A partir de esta métrica, *VADER* clasifica automáticamente el sentimiento en tres categorías: *positivo*, *negativo* o *neutral*.

En esta etapa, se aplicó *VADER* a una transcripción *ficticia* de una llamada institucional entre un asesor académico y un prospecto universitario. El análisis se realizó dividiendo el texto en fragmentos de aproximadamente 150 palabras, conservando así el contexto conversacional de cada intervención. Los resultados generados indican una distribución general dominada por sentimientos positivos, aunque se identificaron también segmentos con contenido neutral y uno con carga negativa, asociado a la expresión de preocupaciones por parte del prospecto.

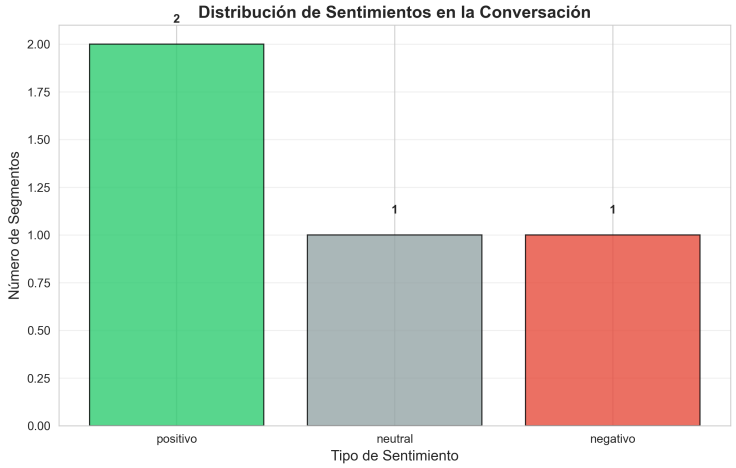

Figura 4.1: Distribución de Sentimientos según análisis con *VADER*

Asimismo, se generó un gráfico de evolución temporal que muestra cómo varió la polaridad emocional a lo largo de la conversación. En la Figura 4.2 se observa un inicio positivo, una caída hacia lo negativo en el momento en que se expresan inquietudes sobre costos, y una recuperación hacia un cierre más optimista, reflejando la dinámica de la conversación.

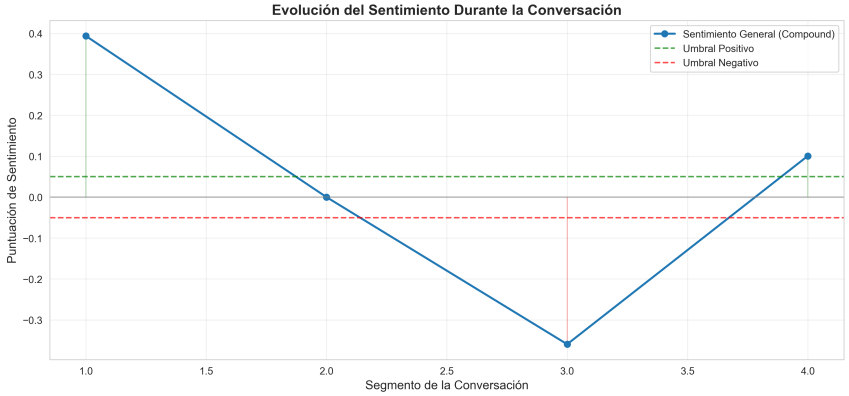

Figura 4.2: Evolución temporal del Sentimiento en la conversación

Finalmente, *VADER* permitió obtener un resumen estadístico con métricas agregadas para caracterizar el tono global de la conversación. Se reportó un puntaje compuesto promedio de 0.034, una dominancia de sentimiento *positivo*, y una variabilidad moderada (desviación estándar = 0.269), lo cual indica cierta oscilación emocional a lo largo del diálogo. Estos datos evidencian el uso de *VADER* como herramienta en entornos donde la rapidez y la interpretabilidad son factores clave.

Prueba de Concepto con VADER + DistilRoBERTa

Los resultados obtenidos con *VADER* evidenciaron limitaciones en la identificación de emociones complejas y matices contextuales presentes en las conversaciones telefónicas institucionales. Para abordar estas limitaciones, se diseñó un segundo enfoque experimental que evalúa la combinación del análisis léxico previamente implementado con un análisis contextual mediante modelos *transformer* preentrenados. El objetivo principal de este enfoque experimental fue evaluar la combinación de un análisis basado en métodos léxicos (*VADER*) con un análisis contextual mediante modelos Transformer preentrenados (*DistilRoBERTa*), buscando una solución que ofreciera mayor precisión en la inferencia semántica de intenciones y emociones específicas.

Para complementar el análisis léxico ya implementado, se incorporó la siguiente tecnología:

- **DistilRoBERTa**: un modelo transformer compacto derivado de *RoBERTa*, especializado en tareas de clasificación textual que requieren comprensión contextual profunda. A diferencia del análisis léxico, este modelo puede identificar emociones específicas (alegría, tristeza, enojo, miedo, etc.), intenciones implícitas y patrones semánticos complejos que dependen del contexto conversacional completo.

El procedimiento experimental se dividió en dos etapas. Primero, las transcripciones fueron procesadas con *VADER* para obtener una estimación inicial rápida del sentimiento predominante en cada interacción telefónica. A continuación, las mismas transcripciones fueron analizadas utilizando *DistilRoBERTa* para confirmar o refinar los resultados de *VADER*, proporcionando además una inferencia más precisa sobre las intenciones expresadas por los prospectos.

Además, las pruebas se realizaron sobre transcripciones ficticias de llamadas telefónicas extraídas desde archivos en formato ".txt", los cuales contienen intervenciones alternadas entre un agente institucional y un prospecto universitario. El texto procesado se encuentra en un formato por líneas facilitado para la diferenciación entre cada participante. Un ejemplo completo de este tipo de transcripción puede consultarse en el Anexo 9, donde se presenta la conversación utilizada en la prueba experimental.

Implementación Técnica

La implementación de la prueba de concepto se estructuró mediante el siguiente *pipeline* de procesamiento:

```
sia = SentimentIntensityAnalyzer()

df[‘sentiment_scores’] = df[‘text’].apply(sia.polarity_scores)

def classify_sentiment(score):
    if score[‘compound’] > 0.05:
        return "positivo"
    elif score[‘compound’] < -0.05:
        return "negativo"
    else:
        return "neutro"

df[‘sentiment’] = df[‘sentiment_scores’].apply(classify_sentiment)
```

Este fragmento realiza un análisis de sentimientos utilizando *VADER*. Primero, se calcula un puntaje de polaridad para cada fragmento de texto, el cual representa una medida numérica que oscila entre -1 (muy negativo) y +1 (muy positivo), indicando la carga emocional general del contenido. Luego, se aplica una función de clasificación que interpreta el valor compuesto (*compound*), que es un resumen ponderado de las puntuaciones positivas, negativas y neutras; además, lo asigna a una de tres categorías: positivo, negativo o neutro tal como se mencionó anteriormente.

```
intent_classifier = pipeline(
    "text-classification",
    model="j-hartmann/emotion-english-distilroberta-base",
```

```
    return_all_scores=False
)

def classify_intent(text):
    result = intent_classifier(text)
    return result[0]['label']

df['predicted_intent'] = df['text'].apply(classify_intent)
```

En este segundo fragmento se implementa un clasificador de texto basado en *DistilRoBERTa*, el cual permite inferir la intención o emoción predominante en cada intervención. El modelo procesa cada fragmento de texto y retorna la etiqueta con mayor probabilidad asociada, que es incluida a la tabla de resultados.

Los resultados preliminares de esta prueba híbrida indicaron que, aunque *VADER* ofrecía resultados rápidos y aceptables para una aproximación general, *DistilRoBERTa* brindaba análisis más refinados, captando matices contextuales específicos que *VADER* no podía identificar. Se observó una complementariedad útil entre ambas herramientas: *VADER* es especialmente efectiva para obtener resultados inmediatos, mientras que *DistilRoBERTa* es clave para entender en profundidad las intenciones y sentimientos más complejos en conversaciones extensas.

Para visualizar esta dinámica, se presentan a continuación las distribuciones de frecuencias de sentimientos e intenciones clasificadas durante la conversación de prueba.

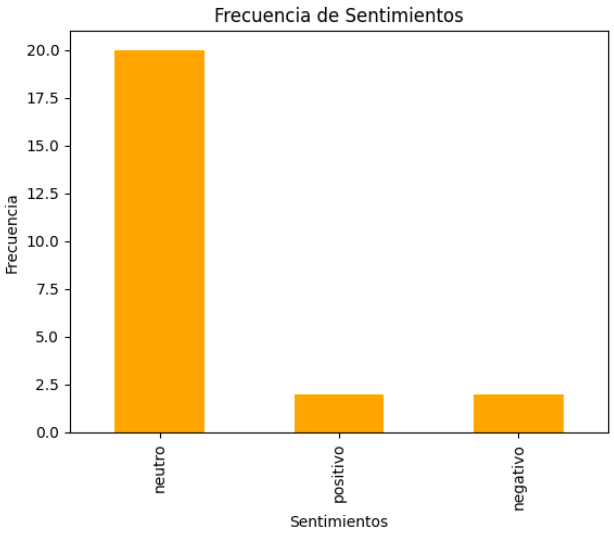

Figura 4.3: Frecuencia de Sentimientos clasificados por *VADER*

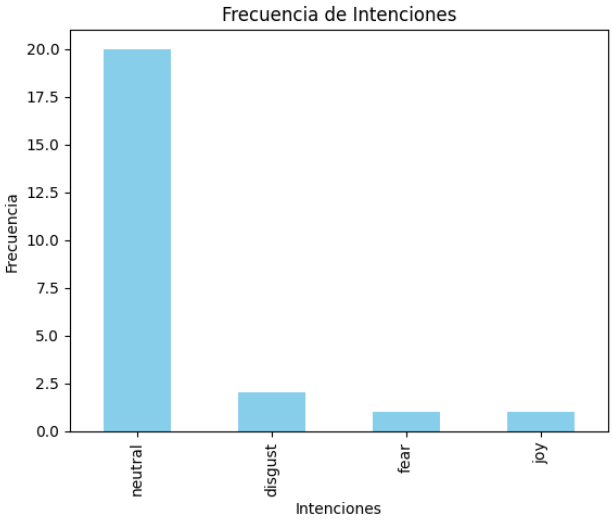

Figura 4.4: Frecuencia de Intenciones clasificadas por *DistilRoBERTa*

Para ilustrar el comportamiento de la prueba de concepto, se presentan en las Figuras 4.3 y 4.4 las distribuciones de frecuencias de sentimientos e intenciones clasificadas durante la conversación. Como se puede observar, predominan las categorías neutras, lo cual es coherente con el tono informativo y cordial de la llamada.

Asimismo, en el Anexo 9 se incluyen fragmentos representativos de los resultados generados por la prueba de concepto, donde se puede apreciar la interacción original entre operador y prospecto junto con la etiqueta de sentimiento e intención asociada a cada intervención. Esta evidencia respalda la capacidad de la prueba de concepto con *VADER* y *DistilRoBERTa* para ofrecer una interpretación estructurada del contenido semántico.

Integración con API de OpenAI Whisper

Este tercer enfoque experimental exploró la capacidad de los modelos de lenguaje desarrollados por *OpenAI* para realizar una transcripción de audio sobre interacciones telefónicas. Se empleó el modelo *Whisper*, accedido mediante la *API* oficial, con el fin de convertir un archivo de audio en formato *opus* en texto y extraer información útil de la conversación. Esta aproximación buscó validar la viabilidad de delegar a un modelo de lenguaje de propósito general la transcripción de llamadas telefónicas, evaluando su utilidad como solución.

Para facilitar el procesamiento de archivos de audio, se implementó una función de transcripción utilizando el modelo *Whisper* de *OpenAI*. La función `transcribe_audio` recibe como parámetros la ruta del archivo de audio y el idioma de transcripción, carga el modelo *turbo* de *Whisper* (que ofrece un balance óptimo entre velocidad y precisión), y procesa el archivo para extraer el contenido textual.

```
def transcribe_audio(file_path, language):
    model=whisper.load_model("turbo")
    result = model.transcribe(file_path, language=language)
    transcription = result["text"]
    return transcription

prueba = transcribe_audio("/content/AudioPrueba.opus", "es")
```

En este caso específico, se transcribió un archivo en formato *opus* con configuración para idioma español, obteniendo como resultado el texto plano que posteriormente alimenta el *pipeline* de análisis de sentimientos e intenciones. Puede consultar con mayor detalle el resultado de la transcripción en el Anexo 9.

4.1.2. Prototipo experimental de Automatización del Análisis de Llamadas con Flujos

Con el objetivo de automatizar el análisis de sentimientos, emociones e intención expresadas por los candidatos durante sus interacciones telefónicas con la universidad, se diseñó un primer prototipo experimental basado en la herramienta n8n. Esta plataforma de automatización de flujos de trabajo, de código abierto y con enfoque visual, permite conectar múltiples servicios y estructurar procesos complejos sin necesidad de programación avanzada [33]. En el contexto de este proyecto, n8n fue utilizada para orquestar un flujo de procesamiento que parte de archivos en formato txt —conteniendo transcripciones completas de llamadas previamente almacenadas de forma local— y los dirige hacia la etapa de análisis, permitiendo así extraer información de manera automatizada y reproducible.

Un ejemplo completo de este tipo de transcripción puede consultarse en el Anexo 9.

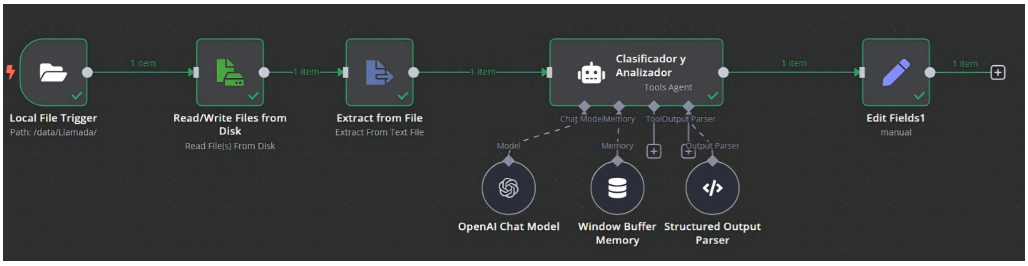

Figura 4.5: Flujo automatizado para análisis de llamadas en *n8n*

Descripción del flujo

La Figura 4.5 muestra el flujo diseñado en *n8n* para el propósito de clasificación de sentimientos y emociones. Su funcionamiento general es el siguiente:

1. **Detonador local (*Local File Trigger*):** El flujo se activa automáticamente cuando se agrega un nuevo archivo en formato *txt* a la carpeta `/data/Llamada/`. Esto permite un procesamiento totalmente automático dónde la intervención manual reside en agregar documentos (transcripciones) en la carpeta indicada.
2. **Lectura y extracción (*Read/Write Files from Disk y Extract from File*):** Se accede al contenido del archivo de texto y se extrae la información relevante, en este caso, la transcripción completa de la llamada.
3. **Clasificación y análisis (Clasificador y Analizador):** Este nodo es un *Agente de Lenguaje (Tools Agent)* que utiliza un modelo de lenguaje de *OpenAI*, en este caso `gpt-4o`. Se le proporciona un *prompt* detallado con instrucciones claras para:
   - Clasificar el **sentimiento general**.
   - Identificar la **emoción predominante** del candidato y del agente universitario.
   - Estimar el **nivel de interés** del candidato durante la interacción.

Ventajas del enfoque

Este prototipo ofrece varias ventajas relevantes:

- **Automatización completa** del procesamiento, desde la llegada del archivo hasta el análisis de alto nivel.
- **Escalabilidad** permitiendo adaptarse al procesamiento de un mayor volumen de llamadas, ya que puede analizar automáticamente todas las transcripciones que se incorporen a una carpeta específica sin necesidad de reconfiguraciones adicionales.
- **Estandarización del análisis** gracias al *prompt* definido, el modelo de lenguaje sigue siempre los mismos criterios de evaluación.
- **Formato estructurado de salida** que facilita la integración con dashboards o bases de datos para análisis posterior.
- **Flexibilidad y extensibilidad** que permite que el flujo pueda ampliarse para incluir nuevos análisis, clasificaciones temáticas o respuestas automáticas.

Este primer prototipo demuestra la viabilidad técnica de integrar herramientas de automatización con modelos avanzados de lenguaje natural para tareas de análisis conversacional en un contexto universitario.

4.1.3. Evaluación Comparativa de Enfoques de Análisis de Sentimientos e Intención

Durante la fase exploratoria se evaluaron tres enfoques distintos para el análisis automático de interacciones telefónicas entre un agente institucional y un prospecto universitario. Cada enfoque fue diseñado con base en diferentes principios de procesamiento del lenguaje natural: desde métodos léxicos tradicionales hasta modelos de lenguaje de última generación. Las alternativas exploradas fueron:

- Análisis basado en reglas con *VADER*.
- Prueba de concepto con *VADER + DistilRoBERTa*.
- Flujo automatizado en *n8n con agentes OpenAI GPT-4o*.

Comparativa Técnica y Funcional

La siguiente tabla resume las capacidades, limitaciones y beneficios observados en cada método a partir de su aplicación sobre transcripciones conversacionales reales o simuladas:

Aspecto Evaluado	**VADER**	**VADER + DistilRoBERTa**	**n8n + GPT-4o**
Precisión emocional	**Básica:** sólo polaridad léxica.	**Media:** identifica emociones con cierto contexto.	**Alta:** identifica matices complejos, emociones implícitas y contradicciones.
Comprensión contextual	**No contextual.** Cada frase se analiza de forma aislada.	**Parcialmente contextual** (gracias a embeddings).	**Totalmente contextual.** comprende turnos de habla, intención y tono.
Capacidad para inferir intención	**Nula.** Solo analiza polaridad.	**Limitada** (requiere modelo adicional).	**Completa.** Permite inferir intención, nivel de interés y tono conversacional.
Flexibilidad de uso	Rápido y ligero, ideal para pruebas simples.	Requiere mayor configuración y combinación de librerías.	Altamente personalizable vía prompts, ajustable a cualquier flujo lógico.
Formato de salida	Puntuaciones numéricas.	Puntuaciones + etiquetas clasificadas.	Estructura JSON detallada con etiquetas personalizadas.
Escalabilidad e integración	Local, no orientado a flujos de producción.	Integrable en pipelines *PLN* tradicionales.	Óptimo para automatización. Totalmente integrable vía *n8n* en entornos productivos.
Modelo subyacente	Léxico + reglas heurísticas.	Léxico + transformer contextual.	Modelo multimodal de última generación (*GPT-4o*).
Ventajas clave	Muy rápido, sin conexión externa.	Más preciso que *VADER* en texto largo.	Precisión, adaptabilidad, automatización, comprensión profunda.
Limitaciones	Sin contexto ni emociones sutiles.	Requiere más mantenimiento.	Requiere *API*, costos de uso por tokens.

Cuadro 4.1: Comparación entre métodos de análisis de lenguaje natural explorados

Tal como se observa en el cuadro 4.1, los métodos tradicionales como *VADER* y *DistilRoBERTa* proporcionan análisis funcionales para tareas puntuales, sin embargo, presentan limitaciones en el procesamiento de comunicaciones complejas. *VADER*, basado en reglas léxicas, carece de comprensión contextual y presenta dificultades para interpretar sarcasmo, ironía y negaciones con implicaciones positivas como "nada mal", además, cuenta con soporte limitado para idiomas distintos al inglés [34], aspectos que pueden afectar la interpretación y análisis de comunicaciones en un contexto universitario colombiano con su jerga característica.

Por otro lado, *DistilRoBERTa*, aunque emplea una arquitectura *transformer*, presenta limitaciones al enfrentarse a textos extensos o ideas complejas. Su entrenamiento, centrado en tareas de clasificación de oraciones, restringe su capacidad para entender el contexto de oraciones o párrafos extensos; además, aunque es un modelo bilingüe español-inglés, pueden existir errores al encontrarse con palabras similares en ambos idiomas [35]. Estas limitaciones pueden afectar la precisión del análisis en contextos reales, donde el lenguaje natural presenta estructuras complejas, expresiones coloquiales y ambigüedades.

La integración de *GPT-4o* en flujos automatizados como *n8n* permite incorporar análisis de lenguaje natural sin requerir ajuste fino, es decir, sin necesidad de entrenar nuevamente el modelo para tareas específicas. Además, estudios recientes han demostrado que *GPT-4o* alcanza niveles de precisión comparables a los de modelos diseñados específicamente para análisis de sentimientos, como *DeBERTa-v3*. Adicionalmente, en evaluaciones realizadas en distintos sectores (negocios, salud y tecnología), *GPT-4o* superó a modelos como *FinBERT*, obteniendo una precisión promedio de 0.46 frente a los 0.40 de *FinBERT*. Lo que representa una mejora aproximada de 0.06 puntos porcentuales [36]. Esta diferencia, aunque aparentemente pequeña, refleja una buena capacidad del modelo para clasificar correctamente los sentimientos expresados en textos diversos sin requerir entrenamiento adicional.

Conclusión de la Comparativa

En conclusión, las limitaciones observadas en *VADER* y *DistilRoBERTa* como la falta de comprensión contextual, dificultades con expresiones ambiguas o coloquiales, y la necesidad de entrenamiento específico para nuevas tareas, evidencian que la integración de *GPT-4o* representa una alternativa más versátil y eficiente. Su capacidad para adaptarse a distintos dominios sin ajuste fino, su flexibilidad para abordar múltiples tareas mediante el diseño adecuado de *prompts*, y su habilidad para mantener coherencia semántica en análisis complejos, lo convierte en una solución más adecuada para escenarios reales como el procesamiento de comunicaciones institucionales. De este modo, se evita la necesidad de entrenar y mantener modelos específicos para cada contexto, favoreciendo una implementación más ágil sin perder precisión.

4.2. Flujo Final de Automatización del Análisis de Llamadas

A partir del análisis comparativo de enfoques de procesamiento de lenguaje natural, se seleccionó la integración de *GPT-4o* a través de la herramienta *n8n* como base para la implementación. En esta sección se describe el flujo final de automatización diseñado para el análisis de grabaciones de llamadas entre prospectos y el área de atención al cliente. El flujo incorpora tareas específicas como la transcripción de audio, el análisis de sentimientos, intenciones y temas mediante modelos de lenguaje, así como la exportación estructurada de los resultados para su posterior visualización o uso.

4.2.1. Arquitectura General del Flujo Final

La implementación de un prototipo de análisis automatizado de llamadas requiere una arquitectura que integre múltiples componentes de forma coherente. En esta sección se presenta el diseño arquitectónico completo del flujo desarrollado, abarcando desde la conceptualización inicial hasta la implementación técnica de cada componente.

Para estructurar la descripción de manera clara y progresiva, se ha decidido adoptar el modelo *C4* (*Context, Containers, Components, Code*) el cual permite visualizar la arquitectura desde diferentes niveles de abstracción [37]. Esta aproximación permite entender de forma gradual el comportamiento del prototipo, empezando desde una vista panorámica del ecosistema hasta el detalle específico, mostrando cómo cada unidad contribuye al objetivo de automatizar el análisis de comunicaciones.

Nivel 1: Diagrama de Contexto General

El primer nivel del modelo *C4* ofrece una visión panorámica general, que revela el ecosistema completo donde opera el prototipo de análisis de llamadas, además, establece el contexto necesario para comprender la razón de ser de cada componente.

Componentes:

- **Personal de Promoción Estudiantil**: Usuario final que accede a los análisis para tomar decisiones. Interactúa con *Tableau* para visualizar los resultados.
- ***WeKall***: Servicio externo que almacena las grabaciones de llamadas telefónicas realizadas a los prospectos.

- **Prototipo de Análisis de Comunicaciones**: Núcleo del proyecto, encargado de recibir enlaces de grabaciones, analizarlas y exportar resultados.
- ***OpenAI API***: Servicio externo utilizado para realizar análisis de lenguaje natural.
- ***Tableau***: Plataforma de visualización donde se presentan los resultados del análisis.

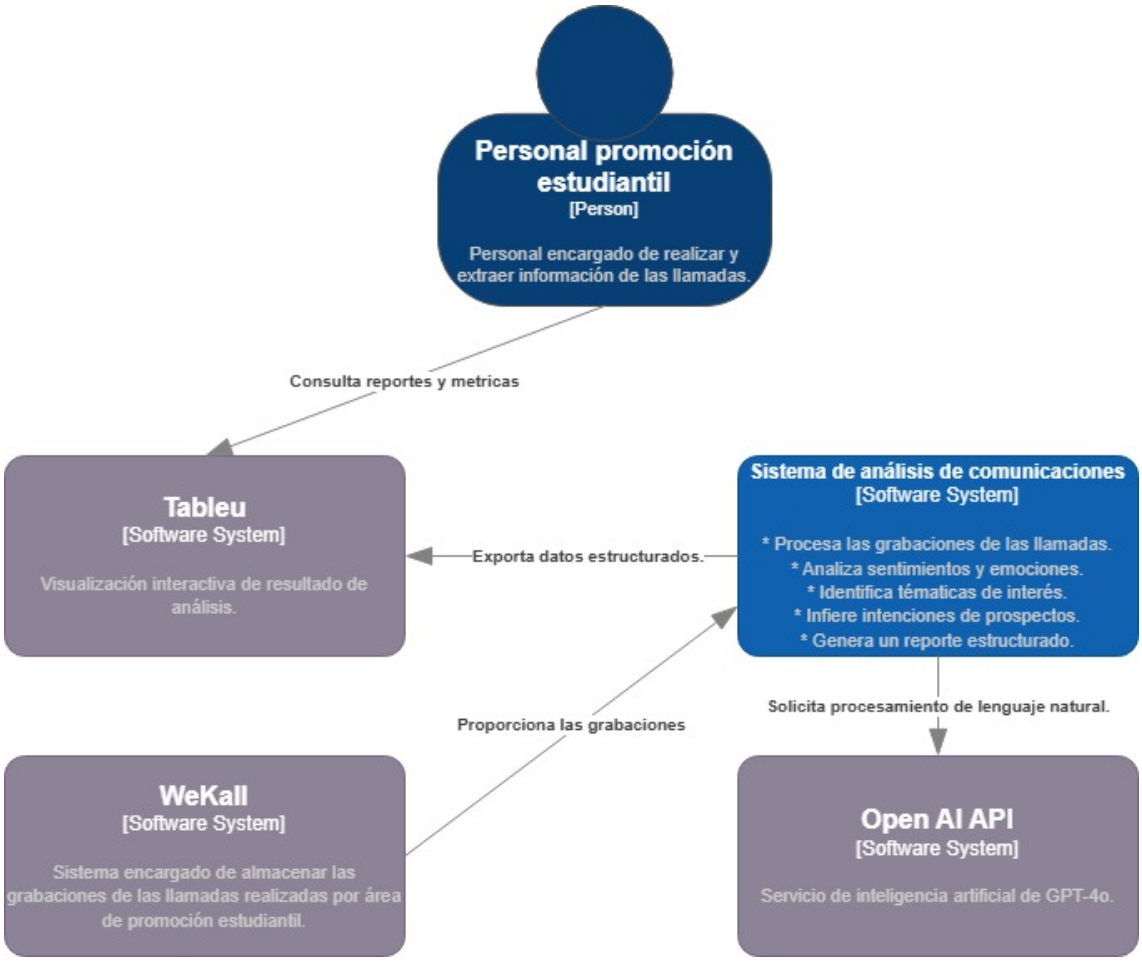

Figura 4.6: Nivel 1 Diagrama C4

El diagrama 4.6 muestra la vista de contexto general del prototipo propuesto. El proceso parte de las grabaciones de llamadas almacenadas en la plataforma *WeKall*, las cuales son procesadas por el prototipo de transcripción y análisis de comunicaciones. El flujo interactúa con la *API de OpenAI* para realizar tareas de procesamiento de lenguaje natural. Los resultados estructurados se exportan a *Tableau*, donde el personal de promoción estudiantil puede consultar reportes y métricas.

Nivel 2: Contenedores del Prototipo

El segundo nivel del modelo *C4* detalla la arquitectura interna del prototipo de análisis de comunicaciones, presentando los principales contenedores que conforman el prototipo. Cada contenedor representa una unidad lógica que cumple una función específica dentro del flujo automatizado, y su interacción conjunta permite llevar a cabo el análisis completo de las grabaciones.

Componentes:

- ***N8N Workflow Engine*** **(Contenedor Node.js)**: Es el componente orquestador que coordina la lectura de archivos, la conexión con servicios externos y el flujo completo de decisiones. Se comunica con el *File System* para leer datos y con el *API Manager* para enviar solicitudes.
- ***API Manager*** **(Contenedor *HTTP Client*)**: Este componente sirve como intermediario para todas las conexiones externas. Se encarga de manejar credenciales, autenticación, reintentos y errores en la comunicación con *OpenAI* u otras *APIs*.
- ***File System*** **(Contenedor *Local Storage*)**: Almacena de forma temporal las grabaciones *mp3* provenientes de *WeKall*. Es consultado por el motor de *N8n* para acceder a los datos de entrada.
- ***Docker Environment*** **(Contenedor *Docker*)**: Contenedor madre que encapsula todos los anteriores. Garantiza aislamiento, portabilidad y control del entorno. Monta directamente el *File System* como volumen compartido con el dispositivo local.
- ***Tableau y OpenAI API***: Aunque se encuentran fuera de la infraestructura local, sus conexiones se gestionan directamente a través del *API Manager*, enrutadas desde el motor de *n8n*.

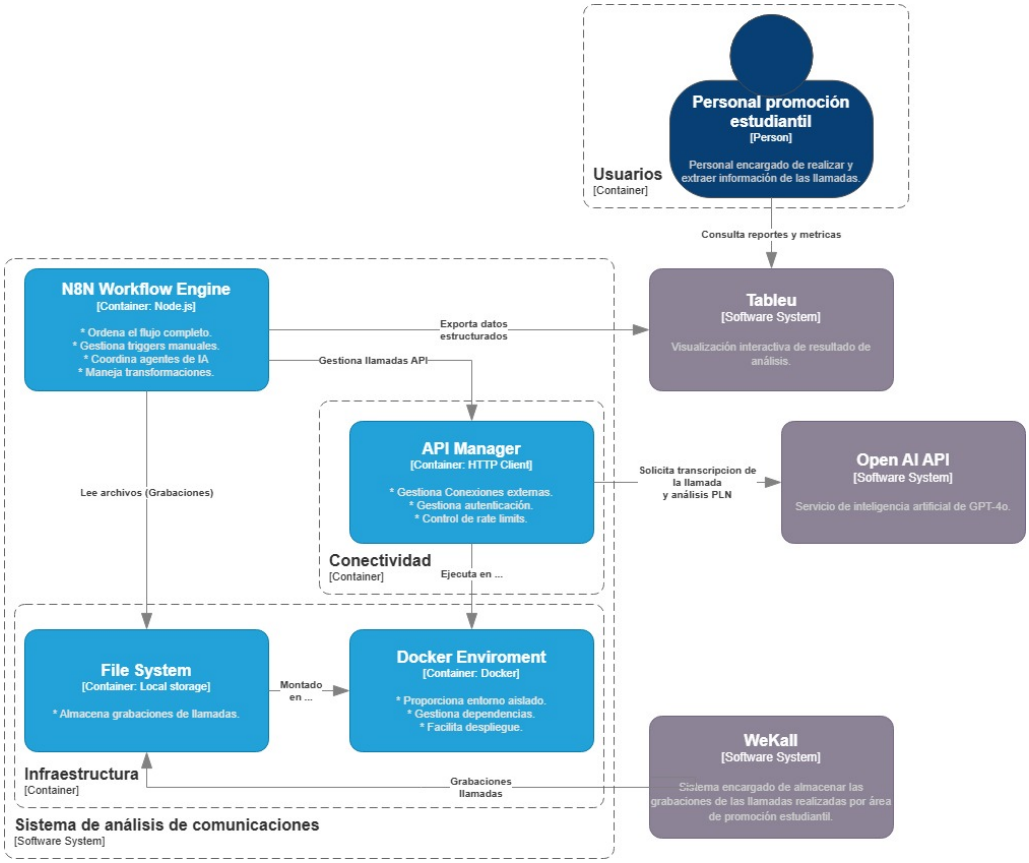

Figura 4.7: Nivel 2 Diagrama C4

El diagrama 4.7 muestra cómo el prototipo se organiza internamente en contenedores funcionales. El flujo es coordinado por *n8n*, que orquesta cada paso del proceso, comenzando con la infraestructura desplegada en el ecosistema de *Docker*, donde se realiza la lectura del archivo de enlaces de grabaciones ubicado en el *File System*. A continuación, se ejecutan tareas de transcripción y análisis mediante el *API Manager*, encargado de gestionar las solicitudes hacia *OpenAI*. Finalmente, los resultados se registran en una tabla del servicio de *Google Sheets*, el cual se encuentra vinculado con *Tableau*, donde el usuario final puede visualizar los resultados del análisis a través de dashboards interactivos.

Nivel 3: Componentes Internos de la herramienta *n8n*

El tercer nivel del modelo *C4* detalla la estructura interna de la herramienta de automatización de flujos *N8n*, presentando los principales componentes que intervienen en la ejecución del flujo. Cada componente corresponde a un nodo funcional que realiza una tarea específica dentro del proceso de análisis, y su secuencia conjunta permite automatizar la lectura, transcripción, análisis y registro de las grabaciones.

Componentes:

- ***Schedule Trigger***: Ejecuta el flujo de manera automática en intervalos definidos. Activa la lectura del archivo que contiene los enlaces de grabaciones disponibles.
- ***Read/Write files from disk***: Accede al *File System* para leer el archivo *txt* que contiene la lista de enlaces a las grabaciones de las interacciones telefónicas.
- ***Http Request***: Realiza una solicitud a la API de *OpenAI* para transcribir el contenido de las grabaciones a texto.
- ***AI Agents***: Tres agentes de inteligencia artificial, cada uno con un objetivo específico: clasificar sentimientos, identificar temáticas e intenciones, y proponer áreas de mejora. Cada agente envía una solicitud a la API de *OpenAI* y devuelve su resultado estructurado.
- ***Edit Fields***: Recibe las respuestas de los tres agentes, las combina y reestructura en un formato unificado, legible y listo para ser registrado en la hoja de cálculo de *Google Sheets*.
- ***Cloud Export API***: Nodo responsable de exportar la información estructurada a una hoja de *Google Sheets*, registrando los resultados en una tabla.
- ***Tableau***: Está conectado a la hoja de *Google Sheets* y utiliza los datos registrados para alimentar dashboards interactivos que pueden ser consultados por los usuarios finales.

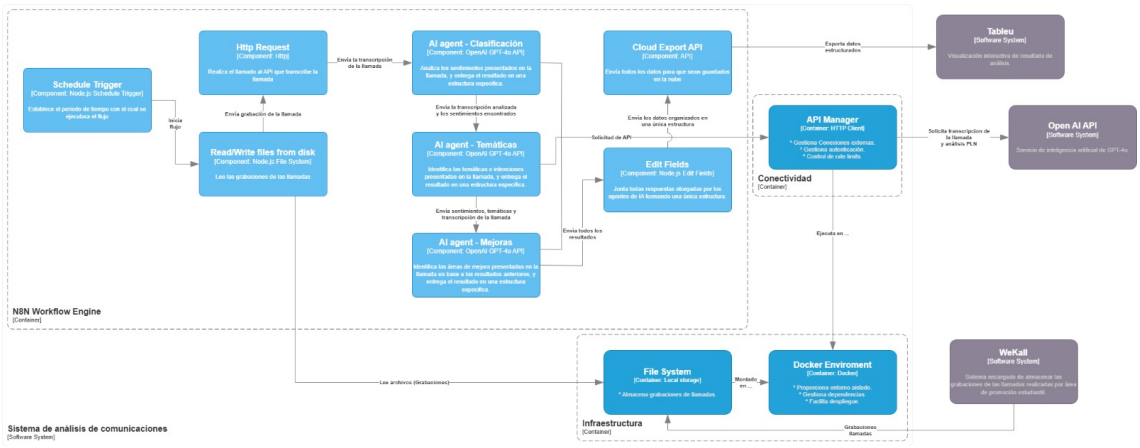

Figura 4.8: Nivel 3 Diagrama *C4*

El flujo comienza con un nodo disparador (*Schedule Trigger*) que activa la ejecución de forma periódica del flujo. A continuación, se accede al archivo local de

texto mediante el nodo *Read/Write files from disk*, desde el cual se extraen los enlaces de grabaciones almacenadas previamente; luego, por medio de una solicitud de *Http* se descarga temporalmente la grabación en un archivo *mp3*. Cada grabación se envía al nodo *HTTP Request*, que interactúa con la *API* de *OpenAI* para realizar la transcripción del audio.

Una vez obtenida la transcripción, esta es enviada a tres agentes de análisis (*AI Agents*), cada uno especializado en una tarea diferente: detección de emociones, identificación de temas e intenciones, y formulación de posibles acciones de mejora. Los resultados generados por estos agentes son consolidados y reestructurados en un único formato por el nodo *Edit Fields*, que organiza la información en campos claros y consistentes.

Posteriormente, el nodo *Cloud Export API* registra los resultados procesados en una hoja de cálculo de *Google Sheets*, la cual está vinculada directamente a *Tableau*. De este modo, los datos quedan disponibles para su visualización mediante dashboards interactivos. Esta organización de nodos dentro de *n8n* refleja una secuencia lineal que permite ejecutar de forma controlada todo el flujo de análisis automatizado.

4.2.2. Detalle del Flujo *n8n* Implementado

Luego de haber presentado la arquitectura general del prototipo a través del modelo *C4*, en esta sección se describe con mayor detalle la implementación del flujo de automatización, realizando un enfoque en el comportamiento interno de cada nodo, su configuración particular y el tipo de datos que manipula en cada etapa.

Obtención y Formato de las Grabaciones Reales

Antes de la implementación del flujo final en *n8n*, fue necesario obtener grabaciones reales almacenadas por el proveedor externo *WeKall*, con el fin de definir un formato compatible con el flujo y establecer los criterios de activación. Para ello, se gestionó una solicitud formal al Centro de Servicios Informáticos de la Pontificia Universidad Javeriana Cali, con el respaldo de la directora del proyecto de grado. Posteriormente, se firmó un contrato de confidencialidad, el cual se incluye en el Anexo 9. Como resultado, se recibió un archivo `CSV` que contenía más de 170.000 registros de llamadas telefónicas históricas.

El archivo recibido contenía múltiples columnas con información diversa, pero solo dos de ellas eran relevantes para el flujo: la columna `DESCRIPTION`, que incluía el enlace a la grabación, y la columna `C_DIGO_LLAMADA__C`, que almacenaba el identificador único de la llamada. Para extraer dicha información, se implementó un script en *Python* que inicia leyendo el archivo *CSV*. Esto se logra mediante la instrucción:

```
with open('extract.csv', encoding='latin-1', newline='') as f:
    csv_reader = csv.DictReader(f)
```

Una vez cargado el archivo, el script recorre cada fila para obtener el contenido de ambas columnas. Para identificar correctamente los enlaces a las grabaciones, se utilizó una expresión regular que detecta cualquier *URL* con formato `MP3`:

```
urls = re.findall(r'https://[^\s",]+.mp3?public=true', description)
```

A continuación, el script verifica si la fila contiene tanto un enlace válido como un identificador no vacío. Si se cumple la condición, concatena ambos elementos utilizando el símbolo `|` como separador, lo que genera una estructura del tipo:

```
https://...grabacion.mp3?public=true|610703
```

Finalmente, todas las combinaciones válidas se escriben en un nuevo archivo de texto llamado `urls_grabaciones.txt`, donde cada línea representa una llamada distinta con su respectivo enlace e identificador:

```
with open('urls_grabaciones.txt', 'w', encoding='utf-8') as output_file:
    for entrada in urls_con_ids:
        output_file.write(entrada + '\n')
```

Esta solución fue seleccionada por su simplicidad, que permitió crear rápidamente un archivo estructurado y fácil de leer desde *n8n*, sin necesidad de transformar los datos en otros formatos intermedios. El código completo del script utilizado se encuentra disponible en el Anexo 9.

Carga y Lectura del Archivo

Con el archivo `urls_grabaciones.txt` ya generado anteriormente por el script de *Python*, es posible iniciar la ejecución del flujo automatizado en *n8n*. Para facilitar la comprensión del funcionamiento del flujo, su explicación se dividirá en bloques funcionales, cada uno compuesto por los nodos que intervienen en una etapa específica del proceso.

El primer bloque, correspondiente a la carga y lectura inicial de los datos, está compuesto por tres nodos principales: **Schedule Trigger**, **Leer llamadas reales** y **Traducir Archivo Grabaciones**. Su función es preparar el contenido del archivo para que los siguientes pasos del flujo puedan operar sobre la información de manera estructurada y automatizada.

El nodo **Schedule Trigger** es el encargado de iniciar automáticamente la ejecución del flujo a intervalos regulares. En este caso, se ha configurado para que se active cada minuto. Posteriormente, el nodo **Leer llamadas reales**, correspondiente a un nodo del tipo *`Read Binary File(s) from Disk`*, accede directamente al archivo `urls_llamadas.txt`. Este archivo se encuentra ubicado en el directorio `/data/Llamada/`, el cual ha sido previamente montado como volumen compartido entre el contenedor de Docker donde se ejecuta *n8n* y el sistema de archivos local del computador del desarrollador.

Sin embargo, al leer el archivo desde disco, *n8n* lo interpreta inicialmente como un archivo binario. Para convertir este contenido a un formato legible y manipulable desde nodos posteriores, se emplea el nodo **Traducir Archivo Grabaciones**, que transforma el archivo binario en una cadena de texto (string) contenida dentro de un objeto JSON. Este paso es fundamental, ya que permite aplicar operaciones sobre el contenido del archivo.

Junto con la lectura del archivo de enlaces de grabaciones, el flujo también accede a un segundo archivo llamado `contador.json`, que contiene el índice de la última llamada procesada. Esta funcionalidad es implementada mediante los nodos **Leer Contador Persistente** y **Convertir Contador**. Ambos permiten mantener un registro modificable del progreso del flujo, sin depender de variables globales internas de *n8n*, las cuales resultaron poco flexibles durante las pruebas.

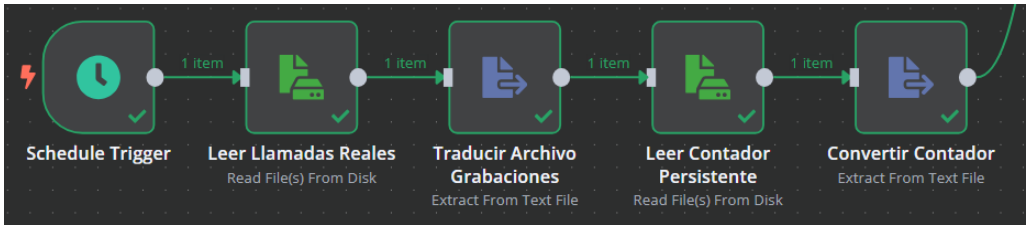

Figura 4.9: Bloque de carga y lectura del archivo de enlaces en *n8n*

Con estos cinco nodos, el flujo queda preparado para manipular enlaces de grabaciones reales. Además, a partir del contenido extraído de ambos archivos, se podrá iterar sobre las grabaciones disponibles, permitiendo que el flujo funcione de forma continua y sin intervención manual adicional hasta finalizar la totalidad de las grabaciones.

Selección y procesamiento de grabaciones

Una vez cargados el archivo `urls_grabaciones.txt` y el archivo auxiliar `contador.json`, el flujo avanza hacia una etapa clave: la selección de la grabación específica que será procesada en la iteración actual. Esta tarea se lleva a cabo mediante un nodo del tipo `Code`, implementado en *JavaScript*, que permite recorrer el listado de enlaces, determinar el índice actual y construir la estructura de datos

utilizada durante el resto del flujo.

El código inicia extrayendo información desde dos fuentes principales: el archivo con la lista de enlaces de grabaciones y el archivo que contiene el índice actual del procesamiento. Adicionalmente, separa cada enlace de grabación por línea y almacena los resultados en un arreglo denominado `lines`, en el cual cada elemento representa una grabación distinta:

```
const content = $('Traducir Archivo Grabaciones').first().json.data;
const lines = content.split('\n')
    .map(line => line.trim())
    .filter(line => line !== '');

let rawContentIndex = $input.first().json.data;
```

Posteriormente, el código accede a la línea específica del arreglo de enlaces según el índice actual. Esta línea se divide en dos partes utilizando el carácter separador `'|'` el enlace directo a la grabación y su identificador único:

```
const currentLine = lines[currentIndex];
let currentUrl = '';
let callId = '';

if (currentLine && currentLine.includes('|')) {
  const parts = currentLine.split('|');
  currentUrl = parts[0].trim();
  callId = parts[1].trim();
} else {
  currentUrl = currentLine;
  callId = 'unknown_${currentIndex}';
}
```

Finalmente, se construye un objeto `JSON` con todos los campos necesarios para las etapas posteriores del flujo. Esta estructura permite mantener la trazabilidad y almacenar resultados o errores de forma unificada en el archivo de resultados final:

```
return [{
  json: {
    url: currentUrl,
    fileName: fileName,
    currentIndex: currentIndex,
    Estado Llamada: "En procesamiento...",
    ...
```

```
    "Sentimiento General": "",
    "Intención Prospecto": "",
    "Áreas de Mejora": "",
    "processingStatus": "started"
  }
}];
```

Esta estrategia de encapsular toda la información relevante en una estructura única desde el inicio permite centralizar los resultados, facilitar el registro de errores y simplificar futuras extensiones o ajustes en la lógica principal del flujo.

Descarga y validación

Una vez seleccionada la grabación a procesar mediante la estructura generada en el paso anterior, el flujo continúa con la descarga y validación del archivo correspondiente. Este bloque del proceso tiene como objetivo principal garantizar que la grabación obtenida esté en condiciones adecuadas para llevar a cabo un análisis de contenido significativo, evitando consumir recursos de análisis en grabaciones defectuosas o que no aportan información relevante.

Dependiendo del resultado de estas validaciones, se definen tres rutas posibles: grabación corrupta, grabación válida pero insuficiente para análisis, o grabación válida y aceptada para análisis avanzado. A continuación, se detallan las operaciones específicas que permiten determinar cada una de estas situaciones, así como las acciones tomadas por el prototipo según el resultado obtenido.

Ruta 1: Grabación corrupta (archivo de 144 KB)

Una vez descargado el archivo mediante una petición *GET* en el nodo **Descargar la Grabación**, se evalúa inmediatamente su integridad mediante el nodo condicional ***IF***. Durante la fase exploratoria del proyecto, se identificó empíricamente que todas las grabaciones corruptas o sin contenido tenían un tamaño exacto de 144 KB, correspondiente a archivos vacíos o con duración nula. Por lo tanto, este criterio fue utilizado como indicador para detectar automáticamente grabaciones defectuosas.

En caso de que la grabación cumpla con esta condición de tamaño, el flujo sigue hacia el nodo **Llamadas corruptas** de tipo *`Code`*, donde se actualiza el estado del objeto JSON sin alterar la estructura, asignando los valores correspondientes:

```
const baseData = $('Iterar sobre Urls').first().json;
const fileSize  = $('Descargar la Grabación')
    .first()
```

```
    .json.headers['content-length'];

const updatedData = {
  ...baseData,
  "Estado Llamada": "Error - Archivo corrupto (144 bytes)",
  "Duración Estimada": "N/A",
  "Cantidad Palabras": 0,
  "Justificación": 'Archivo de audio corrupto o vacío.
  Tamaño: ${fileSize} bytes'
};

return [{ json: updatedData }];
```

Posteriormente, mediante el nodo ***Case Error*** de tipo ***`Edit Field`***, se registra esta condición junto con otros metadatos útiles sobre el archivo descargado, manteniendo vacíos los campos correspondientes a un posible análisis, por ejemplo "Sentimiento general".

Finalmente, a través del nodo **Guardar Análisis con Error**, esta información es almacenada en una hoja específica dentro de *Google Sheets*, permitiendo realizar posteriormente un seguimiento y diagnóstico de todas las grabaciones defectuosas identificadas por el prototipo.

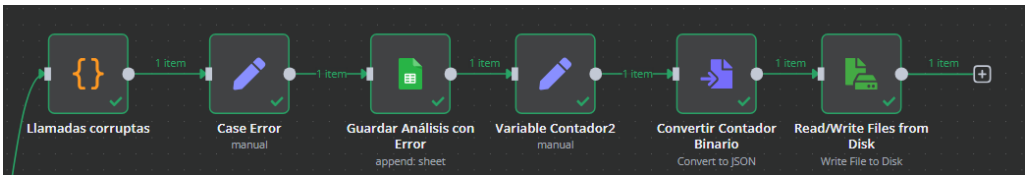

Figura 4.10: Rama de caso validación corrupta

Tras registrar la información, se actualiza el archivo `contador.json` para avanzar al siguiente índice; esta operación se realiza por medio del nodo **Actualizar Índice** de tipo ***`Write File to Disk`***, encargado de escribir el nuevo índice en el archivo auxiliar. De esta forma se evita procesar repetidamente la misma grabación.

Ruta 2: Grabación válida pero insuficiente para análisis

Si la grabación descargada no presenta la condición de archivo corrupto (144 KB), el flujo procede al nodo **Transcripción**, que realiza una petición *HTTP POST* hacia la *API* oficial de *OpenAI*, específicamente al endpoint `https://api.openai.com/v1/audio/transcriptions`. Este nodo se configuró utilizando el modelo de inteligencia artificial `gpt-4o-transcribe`, seleccionado por ofrecer la mejor calidad en transcripciones durante las pruebas exploratorias. Se especificó además el idioma

español, y el retorno del resultado se estableció en formato *n8n Binary file*, bajo la propiedad `data`.

Luego de obtener la transcripción en texto, la siguiente etapa consiste en verificar si esta grabación es viable para realizar análisis avanzados. Esta tarea se implementa mediante un nodo de tipo `Code`, denominado **Verificar Umbrales**, que evalúa la relevancia de la transcripción obtenida usando umbrales definidos, los cuales se muestran a continuación:

```
const THRESHOLDS = {
  MIN_DURATION_SEC: 25,
  MIN_WORDS: 31,
  MIN_CHARACTERS: 155,
  MIN_SENTENCES: 3
};
```

Estos umbrales no fueron elegidos de manera arbitraria; se basaron en literatura especializada sobre análisis de comportamiento a partir de muestras de texto y audio. Chakravarthula et al. (2019) indican que la precisión en la predicción del comportamiento humano mejora significativamente cuando se analizan segmentos textuales con una longitud mínima de aproximadamente 30 palabras, debido principalmente a que modelos como los basados en unidades recurrentes (GRU) requieren contextos suficientemente extensos para capturar dependencias contextuales profundas [38]. El mismo estudio señala que la duración mínima observada para codificar comportamientos relevantes varía entre 30 segundos hasta 10 minutos, reflejando así la variabilidad natural del lenguaje humano y sus implicaciones en análisis.

A partir de estos umbrales teóricos, junto con otros criterios más informales y empíricos utilizados como guía inicial, se realizó una evaluación manual sobre un conjunto aproximado de 100 grabaciones telefónicas seleccionadas aleatoriamente. Durante esta revisión, se separaron las grabaciones según su duración y cantidad de palabras. Al utilizar un umbral similar al indicado en el estudio de Chakravarthula et al. (2019), cercano a los 25 segundos, se identificó que las grabaciones con duraciones inferiores generalmente correspondían a llamadas en las cuales el prospecto colgaba sin razón aparente, indicaba explícitamente que no estaba disponible para discutir el tema en ese momento, o contenían numerosos espacios de silencio que afectaban la calidad del análisis. Algo similar ocurrió al evaluar grabaciones según la cantidad mínima de palabras.

Sin embargo, dado que estas decisiones se basaron principalmente en evaluaciones empíricas sustentadas parcialmente por un único estudio, es importante señalar que este enfoque está sujeto a posibles errores derivados de la interpretación manual del contenido y la variabilidad de las interacciones naturales entre humanos.

Luego de establecer estos umbrales iniciales, el flujo continúa con el cálculo de métricas lingüísticas sobre el texto obtenido en la transcripción, especialmente la

cantidad de palabras, caracteres y oraciones. Este proceso se realiza mediante el siguiente fragmento de código implementado en el nodo **Verificar Umbrales**:

```
const words = transcriptionText
    .toLowerCase()
    .trim()
    .split(/\s+/)
    .filter(word => word.length > 0);
const wordCount = words.length;
const charCount = transcriptionText.length;
const sentenceCount = transcriptionText
    .split(/[.!?]+/)
    .filter(s => s.trim().length > 0).length;

const estimatedDurationSec = (wordCount / 125) * 60;
```

Posteriormente, se evalúa si la transcripción cumple con las condiciones establecidas por los umbrales. Para que una llamada se considere válida para el análisis avanzado, debe satisfacer al menos tres de los cinco criterios definidos. El resultado de esta evaluación se obtiene a través del siguiente fragmento de código:

```
const qualityChecks = {
  hasMinDuration: estimatedDurationSec >= THRESHOLDS.MIN_DURATION_SEC,
  hasMinWords: wordCount >= THRESHOLDS.MIN_WORDS,
  hasMinChars: charCount >= THRESHOLDS.MIN_CHARACTERS,
  hasMinSentences: sentenceCount >= THRESHOLDS.MIN_SENTENCES,
  hasMinDiversity: lexicalDiversity >= THRESHOLDS.MIN_LEXICAL_DIVERSITY
};

const criteriaMetCount = Object.values(qualityChecks)
    .filter(check => check)
    .length;
const isValidCall = criteriaMetCount >= 3;
```

Dependiendo del resultado, el flujo actualiza el objeto *JSON* con el estado adecuado. Si la transcripción cumple las condiciones mínimas exigidas, se considera **Válida**. Por otro lado, si la transcripción no alcanza a cumplir al menos tres de los cinco criterios establecidos, la llamada se considera **Inválida**, actualizando la estructura *JSON* con los respectivos valores para el estado de la llamada.

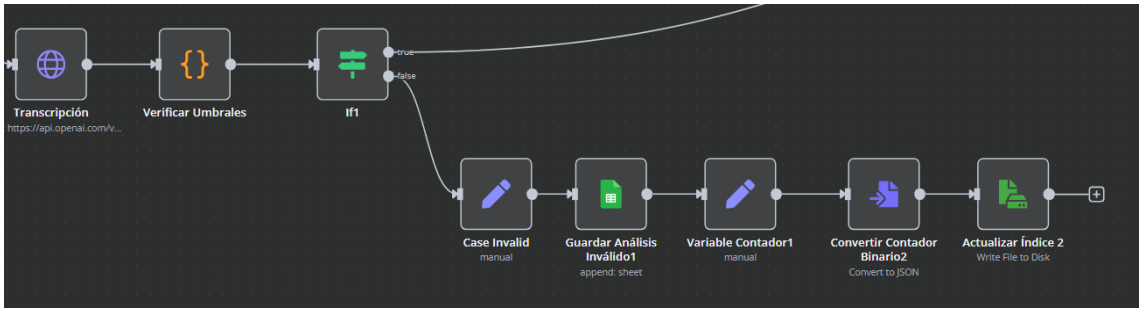

Figura 4.11: Rama de caso de grabación insuficiente

Estos estados se utilizarán inmediatamente después en un nodo condicional (*IF*) del flujo, para determinar si la grabación continúa hacia el análisis avanzado o es descartada y almacenada en un registro específico para llamadas rechazadas, que cuenta con el mismo flujo de nodos para su registro en *Google Sheets* e incremento del índice en **Actualizar Índice 2**.

Análisis Avanzado mediante Agentes de Inteligencia Artificial

Una vez que la transcripción ha superado las validaciones anteriores y se determina que cumple con los umbrales definidos, la llamada se considera apta para continuar hacia la etapa final del proceso: el análisis avanzado mediante agentes de inteligencia artificial. En esta fase, se utilizan diferentes modelos especializados, cada uno enfocado en objetivos específicos. De forma general, estos modelos analizan aspectos como sentimientos, emociones, intereses del prospecto, intenciones, temáticas principales abordadas durante la interacción y posibles áreas de mejora identificadas a partir del contenido analizado.

Cada uno de estos modelos ha sido implementado en *n8n* mediante nodos de tipo `Tool Agent`. Estos nodos permiten asignar *prompts* personalizados para cada modelo, integrando información proveniente de nodos previos, por ejemplo, la transcripción de la llamada actual.

Cada `Tool Agent` se compone internamente de tres elementos principales [39]:

- ***Chat Model*:** Este subnodo permite especificar el modelo de inteligencia artificial a utilizar y configurar sus respectivas credenciales para el consumo de tokens. En este proyecto se ha seleccionado el modelo `GPT-4o` proporcionado por *OpenAI* para los tres agentes. Esta decisión se tomó con el objetivo de unificar el consumo de tokens, reutilizar las mismas credenciales de autenticación y evitar la dependencia de múltiples servicios o modelos externos, lo cual facilita la integración y simplifica la gestión del flujo.

- ***Memory*:** El subnodo de memoria recomendado es el *`Windows Memory Buffer`*. Este tipo de memoria es especialmente útil cuando no se dispone de una base de datos externa, ya que permite almacenar temporalmente información reciente que el agente puede aprovechar como contexto adicional para generar respuestas más precisas. Además, el uso del buffer temporal ayuda a mantener un rendimiento óptimo al evitar la saturación del modelo con contextos innecesarios y contribuye a un consumo eficiente de tokens al limitarse únicamente a contextos recientes.

- ***Output Parser*:** Finalmente, el subnodo *`Output Parser`* permite definir de forma estructurada y precisa el formato de la respuesta generada por cada agente de inteligencia artificial. Esta configuración facilita que la información generada por los modelos sea fácil de manipular e integrar en los nodos posteriores del flujo de *n8n*, menteniendo una compatibilidad directa con las estructuras de datos utilizadas en etapas finales del proceso.

La estructura que utiliza los *`Tool Agent`* permite elegir el tipo de modelo a utilizar, qué tipo de memoria desea utilizar para la ejecución y, finalmente, darle un formato al resultado del modelo según la comodidad o necesidad. Gracias a esto, no fue necesario utilizar nodos de *`HTTP Request`* que hubieran requerido una mayor cantidad de ajustes.

Para mantener la coherencia metodológica entre los modelos, se definió una estructura común para todos los prompts siguiendo buenas prácticas establecidas para modelos de lenguaje natural a gran escala (LLM). Específicamente, se tomó como referencia la metodología propuesta por Wang y Luo (2024), quienes investigaron tres estrategias principales para análisis de sentimientos: *RolePlaying (RP)*, *Chain-of-Thought (CoT)*, y la combinación *RP-CoT*. Los resultados del estudio indican que asignar un rol específico al modelo y guiar explícitamente su razonamiento a través de pasos estructurados mejora significativamente la precisión del análisis, siendo la estrategia combinada *RP-CoT* la más efectiva [40].

Siguiendo esta metodología, en el modelo encargado del análisis de sentimientos y emociones, se ha definido un prompt que inicia asignando al modelo un rol claro y específico para establecer el contexto y orientar la interpretación del contenido:

```
Eres un experto analista de sentimientos especializado en
conversaciones telefónicas del sector de atención al
cliente en una universidad. Sigues un proceso metodológico
paso a paso para garantizar precisión.
```

Este tipo de introducción (*RolePlaying*) permite al modelo ajustar su respuesta según el contexto proporcionado. A su vez, esta estructura metodológica detallada (*Chain-of-Thought*) sigue las recomendaciones oficiales de *OpenAI*, donde se enfatiza la importancia de guiar explícitamente el razonamiento del modelo mediante

instrucciones detalladas y desglosadas en subtareas específicas [41]. Por ejemplo, uno de los pasos estructurados que sigue esta recomendación es la clasificación del sentimiento general del prospecto, indicando cómo el modelo debe categorizar el resultado obtenido:

```
2. Clasifica el sentimiento general como:
   - "Positivo": el prospecto muestra entusiasmo, claridad
   de intención o actitud receptiva. Se expresa con motivación
   o confianza.
   - "Negativo": el prospecto muestra indecisión, desinterés,
   evasión o frustración. Incluso si el tono es educado, si
   hay falta de compromiso emocional o verbal, se considera
   negativo.
   - "Neutral": la conversación es informativa y sin carga
   emocional clara por ninguna de las partes. El prospecto se
   muestra simplemente receptivo pero sin emociones definidas.
```

Finalmente, para facilitar la integración con el flujo de *n8n*, el prompt exige al modelo retornar una respuesta estructurada en formato *JSON* estricto. Esta decisión responde a la práctica recomendada por *OpenAI* de definir formatos claros de respuesta para mejorar la integración de resultados en sistemas posteriores [41]:

```
# Output (JSON estricto)
{
  "sentimiento_general": "",
  "emocion_candidato": "",
  "emocion_agente": "",
  "nivel_interes_candidato": ""
}
```

El prompt completo utilizado para el análisis de sentimientos se encuentra disponible en el anexo 9. La estructura metodológica aplicada en su diseño, es igualmente aplicable a otros contextos de análisis con modelos de lenguaje. Por esta razón, los demás prompts empleados en los agentes restantes siguen el mismo esquema general, incluyendo la asignación de roles, instrucciones secuenciales y un formato de salida estructurado, ajustando sus roles, objetivos, selección de pasos y formato de salida final.

4.2.3. Resultado de Ejecución del Flujo Final

Una vez procesadas las llamadas a través del flujo en *n8n*, los resultados generados por cada uno de los agentes de inteligencia artificial, los casos de error y llamadas

insuficientes, fueron registrados en una hoja de cálculo de *Google Sheets*. Cada fila de esta hoja representa una llamada procesada, y cada columna corresponde a un análisis o campo extraído por los diferentes nodos configurados previamente en el flujo.

La decisión de utilizar Google Sheets como repositorio final de los datos se fundamenta en la integración nativa que ofrece *n8n* con servicios externos. Aunque también es posible trabajar con herramientas de *Microsoft 365* como *Excel* y Word, se optó por el ecosistema de *Google Cloud* debido a que brinda una prueba gratuita de tres meses y se facilitó la configuración de credenciales utilizando una cuenta de correo electrónico.

Los campos registrados en la hoja de cálculo corresponden a los criterios definidos en la sección *Resultados del Levantamiento de Requerimientos* 3.2.2, y fueron validados previamente con la Coordinadora de Servicio al Cliente durante las entrevistas exploratorias. Si bien existen otros criterios relevantes en el proceso de atención, se decidió priorizar aquellos que se alinean con los intereses estratégicos identificados durante el levantamiento de información.

A partir de la ejecución del flujo sobre una muestra inicial de 755 llamadas, se identificaron tres posibles resultados: llamadas corruptas, llamadas insuficientes y llamadas válidas. La siguiente tabla resume la distribución observada en esta muestra:

Tipo de llamada	**Cantidad**	**Porcentaje**
Corrupta	488	64.4 %
Insuficiente	96	12.72 %
Válida	171	22.65 %
Total	755	100 %

Cuadro 4.2: Distribución de resultados del análisis de llamadas

Esta distribución también se representa en la Figura 4.12, que permite visualizar de forma más clara la proporción de cada categoría:

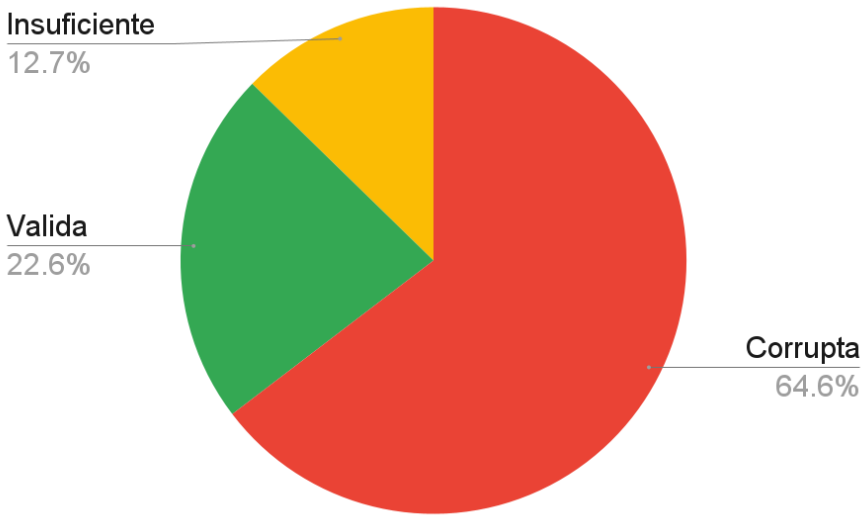

Figura 4.12: Distribución porcentual de los tipos de llamadas analizadas

Estos registros fueron igualmente documentados en la hoja de cálculo con una marca de error específica para su trazabilidad.

Por otro lado, las llamadas clasificadas como insuficientes tienen una duración extremadamente corta (generalmente menos de 10 segundos), lo que impide extraer contenido útil para el análisis. Aunque no están técnicamente dañadas, su escasa longitud las hace inviables para la evaluación:

Ejemplo de Caso Rechazado


```
{
  "id_llamada": "804551",
  "url": "...",
  "Fecha Procesamiento": "2025-07-13T00:40:32.962Z",
  "estado": "Rechazada - Duración/contenido insuficiente",
  "justificación": {
    "Criterios no cumplidos (1/5):
      Duración insuficiente: ~0s (mín: 25s);
      Pocas palabras: 1 (mín: 31);
      Pocos caracteres: 7 (mín: 155);
      Pocas oraciones: 1 (mín: 3)
  }
}
```


En este caso, se trata de una grabación que, aunque no está corrupta —ya que su tamaño supera los 144 KB—, tiene una duración de apenas un segundo y solo contiene una palabra transcrita. Registrar este tipo de situaciones resulta útil

por dos motivos: primero, evita que se gasten tokens en analizar llamadas que no aportan información valiosa; y segundo, permite a la universidad tener una visión más precisa sobre cuántas de sus grabaciones almacenadas carecen de contenido útil. No obstante, vale la pena recordar que los criterios aplicados no son infalibles. Existe la posibilidad de que algunas llamadas breves, pero relevantes, queden por fuera del análisis por no cumplir con los umbrales establecidos.

Finalmente, el 23.5 % restante corresponde a llamadas válidas, es decir, aquellas que cumplieron con los umbrales establecidos y fueron analizadas por los agentes de inteligencia artificial. A continuación se presenta un ejemplo simplificado del tipo de resultados obtenidos para este grupo de llamadas, los cuales sirvieron como base para construir los dashboards explorados en el capítulo siguiente.


```
{
  "sentimiento_general": "Positivo",
  "emocion_candidato": "Interés",
  "emocion_agente": "Amabilidad",
  "nivel_interes_candidato": "Alto",
  "justificacion": "La llamada fue fluida y el prospecto expresó
  interés genuino al hacer preguntas relevantes sobre el inicio de
  clases y las opciones de financiación...",
  "areas_de_mejora": "Mejorar la claridad al explicar los pasos de
  admisión, especialmente en relación con las fechas y montos de pago,
  podría ser beneficioso...",
  "tematicas_detectadas": [
    "Proceso de admisión",
    "Costos",
    "Financiación"
  ],
  "intencion_prospecto": "Sí",
  "palabras_clave": [
    "pago",
    "financiación",
    "matrícula",
    "especialización",
    "neuropsicología"
  ],
  "preguntas_realizadas": [
    "¿Cuándo iniciamos clases?",
    "¿Qué pago lo está haciendo en realidad?",
    "¿Cómo sería la financiación con ustedes?"
  ]
}
```

Capítulo 5

Visualización

5.1. *Dashboard* como Resultado Integrador del Proyecto

El *dashboard* es el resultado principal de este proyecto de investigación, representando la materialización visual de los *insights* extraídos mediante técnicas de procesamiento de lenguaje natural aplicadas a las comunicaciones telefónicas del proceso de promoción estudiantil. Como señala Few [42], "un *dashboard* es una visualización de la información más importante necesaria para lograr uno o más objetivos; consolidada y organizada en una sola pantalla para que la información pueda ser monitoreada de un vistazo". En este contexto, el dashboard desarrollado no solo presenta datos, sino que transforma datos analizados en conocimiento accionable para el área de promoción estudiantil de la Pontificia Universidad Javeriana Cali.

La importancia estratégica de los *dashboards* en el ámbito educativo ha sido ampliamente documentada. Sedrakyan et al. [43] destacan que "los dashboards de aprendizaje son conocidos por mejorar la toma de decisiones al visualizar procesos y ayudar a rastrear donde evolucionan como se esperaba y donde pueden ocurrir problemas potenciales". Aplicando estos principios al contexto de la promoción estudiantil, el dashboard desarrollado busca empoderar a los tomadores de decisiones con insights inmediatos sobre la efectividad de las interacciones con prospectos.

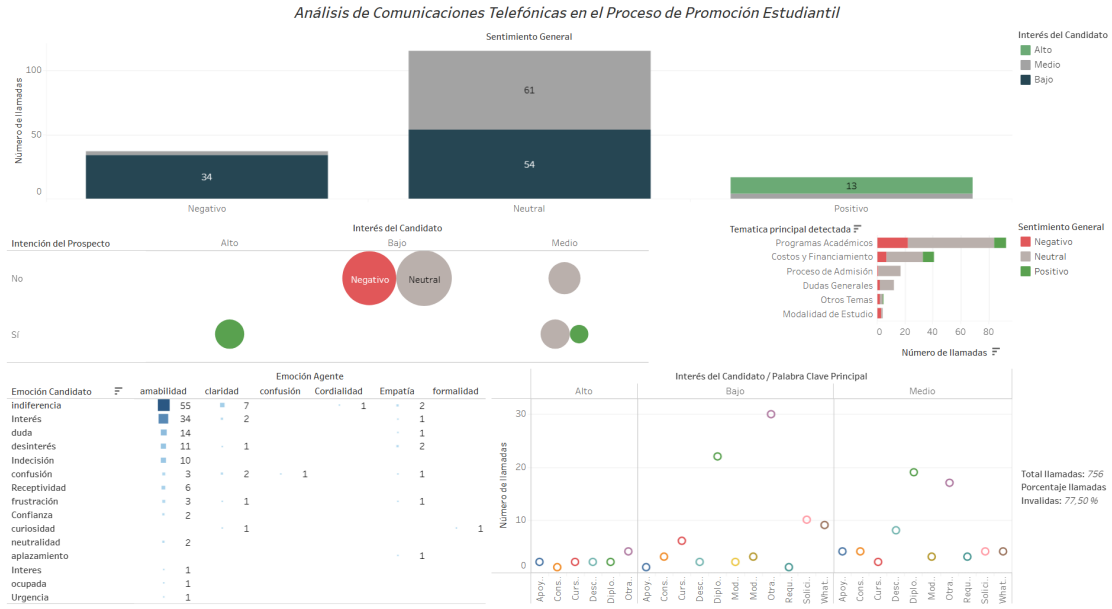

Figura 5.1: Dashboard de análisis de comunicaciones telefónicas en el proceso de promoción estudiantil.

5.2. Principios de Diseño Aplicados

5.2.1. Jerarquía Visual y Organización de la Información

El diseño del dashboard se fundamenta en el principio de jerarquía visual establecido por Tufte [44], quien enfatiza la importancia de guiar la atención del usuario hacia la información más critica. Como indica el articulo *effective dashboard design principles for 2025* [45]: "La jerarquía visual es crucial para guiar la atención de los usuarios hacia la información más critica primero. Al usar diseño, color y tipografía estratégicamente, puedes enfatizar datos de alta prioridad y minimizar distracciones".

Implementación en el Dashboard:

- **Nivel Superior:**
  - Título del dashboard que contextualiza el análisis.
  - Indicadores clave en la esquina inferior derecha: total de llamadas (228) y porcentaje de llamadas inválidas (75.60 %).
  - Gráfico de barras apiladas que muestra la distribución del *Sentimiento General*, codificado por niveles de interés del candidato.

- **Nivel Medio:**
  - Gráfico de burbujas que representa la relación entre la *Intención del Prospecto*, el *Interés del Candidato* y el *Sentimiento General*.
  - Gráfico de barras horizontales que analiza las *Temáticas principales detectadas*, segmentadas por sentimiento.
- **Nivel Inferior:**
  - Mapa de calor de *Emoción del Candidato* versus *Emoción del Agente*, representado por frecuencia.
  - Diagrama de dispersión que cruza *Palabras Clave Principales* con el *Interés del Candidato*.

Esta organización sigue el patrón de pirámide invertida recomendado por Sisense [46], originado en el periodismo, donde la información más importante y sustancial esta en la parte superior, seguida de detalles significativos que ayudan a entender la visión general, y en la parte inferior información general y de fondo.

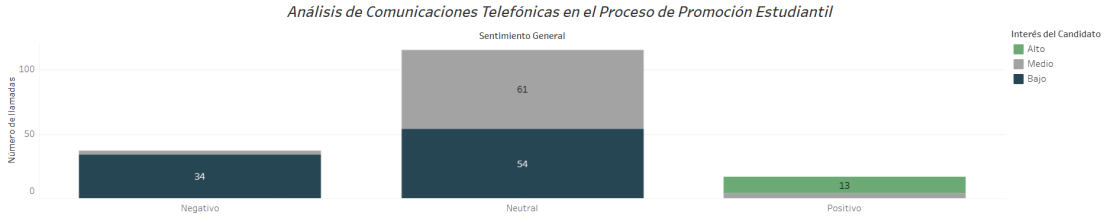

Figura 5.2: Nivel superior

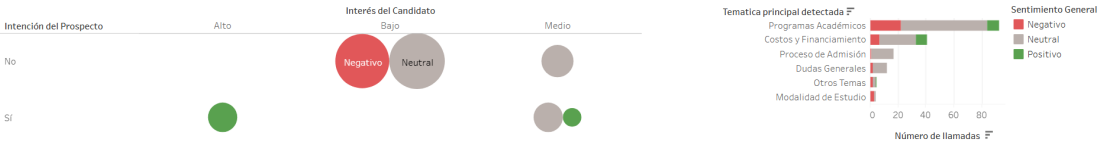

Figura 5.3: Nivel medio

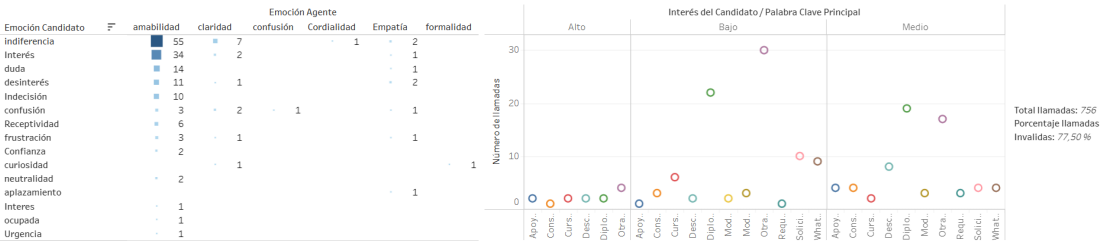

Figura 5.4: Nivel inferior

5.2.2. Principio de Simplicidad y Reducción de Carga Cognitiva

El *dashboard* adhiere al principio de simplicidad establecido por varios expertos en visualización de datos. Como se señala la investigación psicológica aplicada al diseño [47]: "El cerebro humano puede procesar efectivamente alrededor de nueve imágenes al mismo tiempo. Esta es una consideración crítica al trabajar en diseños de *dashboard*, ya que sobrecargar elementos visuales puede abrumar a los usuarios".

Estrategias implementadas:

- Limitación a 6 visualizaciones principales en la vista inicial
- Uso estratégico del espacio en blanco para separar elementos
- Eliminación de elementos decorativos innecesarios siguiendo el principio de "data-ink ratio"de Tufte

5.2.3. Consistencia Visual y Coherencia Cromática

La consistencia visual constituye un pilar fundamental del diseño efectivo de *dashboards*. Como establece Yellowfin [48]: "El diseño claro del *dashboard* se asegura y entrega a través del uso coherente de tipografía, colores y elementos gráficos".

Paleta cromática justificada:

- **Verde:** Sentimientos positivos
- **Amarillo:** Estado neutral
- **Rojo:** Sentimientos negativos

Esta elección se sustenta en la investigación sobre percepción cromática, donde estos colores han demostrado ser intuitivos y accesibles para usuarios con diferentes capacidades visuales.

5.3. Justificación de Tipos de Gráficos Seleccionados

5.3.1. Gráfico de Barras Apiladas para Análisis de Sentimientos

Fundamentación Teórica:

La selección de **gráficos de barras apiladas** para representar la distribución de sentimientos se basa en investigación académica sobre efectividad de visualizaciones. Hill [49] demuestra en su estudio comparativo que "mientras que en general, el gráfico de barras fue más flexible, donde hubo diferencias estadísticamente significativas entre gráficos, los tamaños del efecto fueron a menudo pequeños y es poco probable que impidan la extracción efectiva de información cuantitativa".

Ventajas Específicas:

- **Comparación Precisa:** Las barras permiten comparaciones precisas de cantidades, algo que los gráficos circulares no facilitan eficientemente.
- **Escalabilidad:** Facilita la adición de nuevas categorías sin comprometer la legibilidad.
- **Interpretación Intuitiva:** Los usuarios pueden estimar valores con mayor precisión comparando longitudes que áreas o ángulos.

Evidencia Académica:

Según la investigación de Xiong et al., citada en estudios recientes [50]: "los gráficos de barras tienden a hacer que los grupos de investigación parezcan más grandes y las cargas de trabajo más altas comparado con los gráficos circulares, incluso cuando los datos subyacentes son los mismos". Esta característica es ventajosa para nuestro contexto, donde queremos enfatizar la magnitud de los diferentes tipos de sentimientos.

5.3.2. Gráfico de Barras Horizontales para Análisis de Emociones

Justificación Metodológica:

La utilización de **gráficos de barras horizontales** para la visualización de emociones del candidato se fundamenta en principios ergonómicos y de usabilidad. Como establece Alchemer [51]: "Los gráficos de barras horizontales son más útiles cuando las categorías tienen nombres largos que serían difíciles de ajustar debajo de una barra vertical, o cuando tienes muchas categorías".

Beneficios Específicos:

- **Legibilidad:** Las etiquetas de emociones ("Indiferencia", "Interés", "Desinterés") se leen cómodamente en orientación horizontal.
- **Ordenamiento:** Facilita el ordenamiento descendente por frecuencia, mostrando inmediatamente las emociones más prevalentes.
- **Espacio Eficiente:** Optimiza el uso del espacio disponible en el layout del *dashboard*.

5.3.3. Gráfico de Burbujas para Análisis de Intenciones

Fundamento Teórico:

La implementación de **gráficos de burbujas** para representar las intenciones del prospecto se basa en la necesidad de mostrar tanto categorías como magnitudes de manera intuitiva. Según la investigación en visualización de datos [52]: "Los gráficos de burbujas son efectivos para mostrar la relación entre diferentes categorías cuando el tamaño de la categoría es un factor importante".

Ventajas Contextuales:

- **Impacto Visual:** El tamaño de las burbujas comunica inmediatamente la proporción relativa entre "Sí" y "No".
- **Simplicidad:** Para datos binarios, evita la complejidad innecesaria de visualizaciones más elaboradas.
- **Focalización:** Permite identificar rápidamente el balance entre intenciones positivas y negativas.

5.3.4. Gráficos de Barras Apiladas para Temáticas y Palabras Clave

Justificación Académica:

La utilización de **gráficos de barras apiladas** para el análisis de temáticas principales se sustenta en investigación sobre visualización de datos multidimensionales. Según el estudio de eficacia de gráficos apilados [53]: "Los gráficos de barras apiladas son un método de visualización para presentar múltiples atributos de datos, y muchas herramientas de visualización soportan estos gráficos".

Beneficios Metodológicos:

- **Análisis Dual:** Permite ver tanto el volumen total de menciones por temática como su distribución emocional.
- **Identificación de Patrones:** Facilita la identificación de temáticas que generan consistentemente sentimientos negativos.
- **Acción Inmediata:** Proporciona información accionable para mejorar estrategias de comunicación.

5.4. Aplicación de Principios de Experiencia de Usuario (UX)

5.4.1. Regla de los 5 Segundos

El dashboard implementa la "**regla de los 5 segundos**" establecida por expertos en UX [46]: "Tu dashboard debe proporcionar la información relevante en aproximadamente 5 segundos. Tu dashboard debe poder responder tus preguntas de negocio más frecuentes de un vistazo".

Implementación Práctica:

- Métricas clave visibles inmediatamente en la parte superior.
- Código de colores intuitivo que permite identificación rápida de problemas.
- Organización lógica que sigue patrones de escaneo visual en F y Z.

5.4.2. Interactividad y Filtros Dinámicos

La implementación de elementos interactivos se basa en principios de diseño centrado en el usuario. Como establece la investigación [52]: "Las acciones del dashboard permiten que los usuarios interactúen con visualizaciones, creando experiencias más inmersivas y permitiendo exploración más profunda de los datos".

Funcionalidades Implementadas:

- Filtros cruzados entre visualizaciones.
- Tooltips informativos para contexto adicional.
- Parámetros dinámicos para análisis personalizado.

5.5. Consideraciones de Accesibilidad y Usabilidad

5.5.1. Diseño Inclusivo

El dashboard incorpora principios de **diseño inclusivo** fundamentados en investigación sobre accesibilidad. Como señala UXPin [45]: "Asegurar la accesibilidad con contraste apropiado y marcado semántico" es fundamental para el éxito de cualquier visualización de datos.

Medidas Implementadas:

- Contraste de colores que cumple estándares WCAG 2.1.
- Uso de patrones además de color para diferenciar categorías.
- Tipografía legible con tamaños apropiados.

5.5.2. Optimización para Diferentes Audiencias

El diseño considera múltiples tipos de usuarios, siguiendo las recomendaciones de Toptal [54]: "Uno de los mayores desafíos del diseño de dashboards es servir múltiples personas. Una vez que cada rol de usuario está definido, se vuelve crítico entender dónde se superponen sus necesidades y dónde divergen".

Adaptaciones por Audiencia:

- **Directivos:** Métricas de alto nivel y tendencias generales.
- **Coordinadores:** Detalles operativos y áreas de mejora específicas.
- **Agentes:** Insights para mejorar técnicas de comunicación.

5.6. Validación del Diseño mediante Principios Científicos

5.6.1. Reducción de Carga Cognitiva

El diseño implementa principios de **reducción de carga cognitiva** basados en investigación psicológica. Como establece Yellowfin [48]: "El buen diseño de dashboard es exitoso cuando reduce el esfuerzo requerido por el usuario final al mínimo posible cuando intentan interpretar y usar los datos presentados frente a ellos para tomar decisiones".

Estrategias Aplicadas:

- Agrupación lógica de información relacionada.
- Eliminación de elementos redundantes.
- Progresión natural de información general a específica.

Capítulo 6

Evaluación y Validación del Prototipo

6.1. Justificación de las Pruebas

Dado que el prototipo desarrollado tiene como finalidad alimentar tableros de visualización y respaldar la toma de decisiones del equipo de promoción estudiantil, es necesario evaluar de forma independiente la precisión de los resultados generados. En particular, es necesario validar tanto la transcripción de las grabaciones como los análisis posteriores realizados por los agentes de inteligencia artificial. Esta etapa de validación permite verificar el desempeño del prototipo en condiciones reales de uso, y no únicamente bajo escenarios controlados.

Si bien *OpenAI* proporciona métricas oficiales sobre el rendimiento de sus modelos, la precisión puede variar dependiendo del contexto de uso —por ejemplo, llamadas entre prospectos y agentes del área de atención al cliente, en idioma español, con presencia de ruido ambiental, acentos regionales y terminología propia del entorno universitario—. En consecuencia, se optó por realizar una evaluación empírica del modelo *gpt-4o-transcribe*.

Este modelo ha sido reportado por *OpenAI* con mejoras sustanciales en la métrica conocida como *Word Error Rate* (*WER*), la cual evalúa la calidad de una transcripción comparando el texto generado con una versión de referencia. Esta métrica se calcula como el número de errores (suma de inserciones, eliminaciones y sustituciones) dividido entre el número total de palabras de la transcripción de referencia [55]. Un valor más bajo indica una mayor precisión. En el caso del idioma español, *OpenAI* ha reportado un *WER* de 1.96 %, lo cual sugiere un rendimiento altamente preciso en tareas de transcripción automática en este idioma [56].

No obstante, este resultado proviene de pruebas controladas y *benchmarks* establecidos. Por ello, se consideró necesario validar de forma independiente el desem-

peño del modelo en el contexto específico de este proyecto, manteniendo *WER* como métrica principal para dicha evaluación. El procedimiento detallado y los resultados obtenidos se presentan en las secciones siguientes.

Adicionalmente, para el caso del análisis de sentimiento y emociones, *OpenAI* indica que *GPT-4o* "alcanza un rendimiento a nivel de *GPT-4 Turbo* en texto, razonamiento y codificación, al tiempo que establece nuevos récords en capacidades multilingües, de audio y visión"[57]. Sin embargo, dichos resultados se obtuvieron en benchmarks generalizados y no aseguran un comportamiento preciso en contextos específicos como llamadas telefónicas universitarias. Por esta razón, se consideró necesario realizar una validación independiente usando métricas como exactitud y *F1-score* para evaluar cómo el modelo clasifica emociones e intenciones en conversaciones reales entre prospectos y agentes del área de atención al cliente.

6.2. Metodología General de Evaluación

Con el propósito de evaluar la precisión del prototipo implementado, se diseñó una estrategia de validación que permitiera comparar los resultados generados por los modelos de inteligencia artificial frente a anotaciones realizadas manualmente. Esta evaluación se centró en dos componentes fundamentales del flujo: la transcripción de las grabaciones y el análisis posterior de las interacciones.

Para tal fin, se seleccionaron de forma aleatoria 35 llamadas telefónicas que cumplieron con todos los criterios previamente establecidos para ser consideradas válidas. Estas llamadas fueron procesadas por el flujo y contienen tanto la transcripción automática como el análisis completo por parte de los agentes de IA.

En el primer componente —evaluación de la transcripción—, el equipo de trabajo realizó de forma manual la transcripción completa de cada una de las 35 grabaciones, las cuales fueron almacenadas en un archivo de texto. Posteriormente, se extrajeron las transcripciones automáticas generadas por el modelo *gpt-4o-transcribe*, y se registraron en un archivo adicional en formato similar. Esta información se empleó para calcular la métrica de *Word Error Rate* (*WER*) mediante comparación directa entre ambas versiones, humana y agente de *IA*.

En cuanto al análisis de la conversación, se definieron cuatro atributos principales a evaluar: *Sentimiento General*, *Emoción del Candidato*, *Emoción del Agente*, e *Interés del Candidato*. Estos valores fueron también determinados manualmente por el equipo luego de escuchar cada grabación. Posteriormente, se extrajeron los resultados correspondientes generados por el flujo, y se registraron ambas versiones —humana e *IA*— en una hoja de cálculo común, facilitando así su comparación.

Finalmente, se desarrollaron *scripts* en lenguaje *Python* con apoyo de librerías especializadas para calcular métricas de validación como precisión y *F1-score*. Los resultados de esta evaluación se detallan en las secciones siguientes, junto con ejem-

plos representativos y fragmentos relevantes del código utilizado.

6.3. Evaluación de la Transcripción

Debido a que la transcripción de audio representa el primer eslabón del flujo de análisis, calcular su precisión es importante para validar la calidad de respuestas del prototipo. Esta sección presenta el procedimiento seguido para comparar las transcripciones generadas por el modelo *gpt-4o-transcribe* frente a las versiones manuales de referencia.

6.3.1. Normalización de las Transcripciones

Antes de proceder con el cálculo del *WER*, se realizó un proceso de normalización sobre ambas versiones de las transcripciones —humana y automática—. Esta etapa fue necesaria debido a que las transcripciones generadas por el modelo pueden contener mayúsculas, signos de puntuación, tildes y otros caracteres especiales que, si bien no alteran el contenido semántico, sí pueden aumentar el número de errores al ser comparadas con la versión manual mediante el algoritmo de evaluación.

Para realizar este proceso de normalización se desarrolló la función `normalizar_texto`. Esta función transforma los textos a minúsculas, elimina tildes y signos de puntuación, y reduce los espacios en blanco, asegurando que la comparación posterior se base exclusivamente en el contenido léxico. A continuación se presenta el fragmento correspondiente:

```
def normalizar_texto(texto: str) -> str:
    texto = texto.lower()
    texto = unicodedata.normalize('NFD', texto)
    texto = re.sub(r'[^\w\s]', ' ', texto)
    texto = re.sub(r'\s+', ' ', texto)
    texto = texto.strip()
    return texto
```

Como se mencionó anteriormente, este paso permite evitar que se reporten errores causados por diferencias superficiales, como el uso de tildes, signos de puntuación o mayúsculas, que podrían distorsionar los resultados de precisión al calcular el *WER*.

Cálculo del *Word Error Rate*

Una vez normalizados los textos, se procede al cálculo del *WER* utilizando una función que implementa el algoritmo de Levenshtein. Este algoritmo permite determinar la distancia mínima de edición entre dos secuencias de palabras, considerando operaciones de inserción, eliminación y sustitución. La cantidad total de errores detectados se divide luego entre el número total de palabras en la transcripción de referencia, lo que da como resultado el *WER*.

La función encargada de calcular la distancia de edición se denomina `calcular_distancia_edicion`. A continuación, se describen los bloques más relevantes de dicha implementación:

```
dp = np.zeros((n + 1, m + 1), dtype=int)

for i in range(n + 1):
    dp[i][0] = i  # Eliminaciones
for j in range(m + 1):
    dp[0][j] = j  # Inserciones
```

Esta sección prepara una matriz de tamaño (n+1)×(m+1) donde n y m son las longitudes de las secuencias de palabras de referencia (manual) y de hipótesis (*IA*), respectivamente. Las primeras filas y columnas representan los costos acumulados por operaciones de inserción o eliminación al comparar secuencias vacías con las palabras de cada transcripción.

```
for i in range(1, n + 1):
    for j in range(1, m + 1):
        if ref_words[i-1] == hyp_words[j-1]:
            dp[i][j] = dp[i-1][j-1]
        else:
            dp[i][j] = 1 + min(
                dp[i-1][j], # Eliminación
                dp[i][j-1], # Inserción
                dp[i-1][j-1] # Sustitución
            )
```

Posteriormente, en este bloque se calcula de manera iterativa el costo mínimo para transformar una secuencia de palabras en otra, considerando inserciones, eliminaciones y sustituciones. La igualdad exacta entre palabras evita penalización, mientras que cualquier diferencia suma un costo de uno.

```
while i > 0 or j > 0:
        if i > 0 and j > 0 and ref_words[i-1] == hyp_words[j-1]:
            i -= 1
            j -= 1
        elif i > 0 and j > 0 and dp[i][j] == dp[i-1][j-1] + 1:
            sustituciones += 1
            i -= 1
            j -= 1
        elif i > 0 and dp[i][j] == dp[i-1][j] + 1:
            eliminaciones += 1
            i -= 1
        elif j > 0 and dp[i][j] == dp[i][j-1] + 1:
            inserciones += 1
            j -= 1
        else:
            break

    return sustituciones, eliminaciones, inserciones, dp
```

Finalmente, se realiza un recorrido inverso sobre la matriz de comparación para contar la cantidad de operaciones necesarias para alinear ambas secuencias: sustituciones, eliminaciones e inserciones. Estos valores constituyen los componentes para el cálculo del *WER* según la fórmula previamente descrita.

Después de normalizar los textos y calcular la cantidad de operaciones, se calculan las métricas de evaluación. En particular, el *WER* se obtiene dividiendo el total de errores entre el número de palabras de la referencia (N). También se estima la precisión de forma complementaria. El resultado es devuelto como un diccionario estructurado:

```
wer = (S + D + I) / N
accuracy = max(0, 1 - wer)

return {
    'wer': wer,
    'accuracy': accuracy,
    'sustituciones': S,
    'eliminaciones': D,
    'inserciones': I,
    'total_errores': S + D + I,
    'palabras_referencia': N,
    'palabras_hipotesis': len(hyp_words)
}
```

Presentación de los Resultados

Tras realizar la comparación entre las transcripciones manuales y las generadas por el modelo *gpt-4o-transcribe*, se analizaron 35 transcripciones válidas que, en conjunto, sumaron un total de 6931 palabras en la referencia humana.

Con base en esta comparación, se calcularon tanto métricas agregadas como promedios por transcripción. El valor más representativo fue el WER total, que alcanzó 0.244, lo que indica que aproximadamente el 24.4 % de las palabras difieren entre la versión automática y la versión humana. De forma complementaria, la precisión total fue del 75.6 %.

Por su parte, el análisis promedio —realizado individualmente para cada par de transcripciones— arrojó un *WER* promedio de 0.304 (30.4 %) y una precisión promedio de 75.7 %. Aunque estos promedios permiten observar con mayor detalle el comportamiento del modelo en cada caso, el *WER* total se considera una métrica más robusta para propósitos de evaluación global de las transcripciones.

$$WER = \frac{S + D + I}{N} = \frac{690 + 503 + 500}{6931} = \frac{1693}{6931} \approx 0{,}244$$

Aunque nuestro resultado de *WER* (24.4 %) no iguala la precisión del 1.96 % reportado por *OpenAI* para su modelo *GPT-4o*, lo consideramos una aproximación razonable dadas las condiciones particulares de nuestro análisis. Es importante destacar que la prueba se realizó sobre un conjunto limitado de 35 grabaciones, transcritas únicamente por un miembro humano del equipo de trabajo. Adicionalmente, en algunos casos el modelo reformuló fragmentos de texto, manteniendo el sentido original pero sin transcribir literalmente las palabras. A pesar de estas diferencias inherentes a la metodología de prueba, los resultados obtenidos respaldan la capacidad del modelo para generar transcripciones precisas y útiles dentro del contexto específico del proyecto.

6.4. Evaluación del Análisis

6.4.1. Creación del Dataset

Con el objetivo de evaluar la capacidad del modelo de inteligencia artificial para analizar conversaciones humanas, se desarrolló una prueba de comparación entre las respuestas generadas por la *IA* y aquellas asignadas manualmente por el equipo evaluador. Esta etapa del análisis se centró en cuatro atributos fundamentales extraídos de las llamadas: *Sentimiento General*, *Emoción del Candidato*, *Emoción del Agente* e *Interés del Candidato*.

La evaluación se realizó sobre el mismo conjunto de 35 llamadas previamente utilizadas para la validación de transcripciones. En este caso, se buscó cuantificar qué tan alineadas estaban las interpretaciones del modelo con las observaciones humanas, aplicando métricas de desempeño como precisión y *F1-score*.

Analisis Humano				Análisis de IA			
Sentimiento General	Emoción Candidato	Emoción Agente	Interés del Candidato	Sentimiento General	Emoción Candidato	Emoción Agente	Interés del Candidato
Neutral	Curiosidad	Amabilidad	Medio	Neutral	Curiosidad	Amabilidad	Medio
Positivo	Interés	Amabilidad	Alto	Neutral	Interés	Amabilidad	Medio
Neutral	Desinterés	Cordialidad	Bajo	Negativo	Indiferencia	Cordialidad	Bajo
Neutral	Interés	Amabilidad	Medio	Neutral	Interés	Amabilidad	Medio
Neutral	Interés	Claridad	Medio	Neutral	Interes	Claridad	Medio
Neutral	Desinterés	Amabilidad	Bajo	Neutral	indiferencia	amabilidad	Bajo
Neutral	Indiferencia	Amabilidad	Bajo	Neutral	indiferencia	amabilidad	Bajo
Neutral	Indecisión	Profesionalismo	Bajo	Negativo	Indecisión	Profesionalismo em	Bajo
Neutral	Desinterés	Amabilidad	Bajo	Neutral	Indiferencia	Amabilidad	Bajo
Positivo	Interés	Amabilidad	Alto	Positivo	Interés	Amabilidad	Alto
Neutral	Desinterés	Amabilidad	Bajo	Neutral	Indiferencia	Amabilidad	Bajo
Neutral	Indiferencia	Amabilidad	Medio	Neutral	Indiferencia	Amabilidad	Bajo
Negativo	Desinterés	Amabilidad	Bajo	Negativo	Indiferencia	Amabilidad	Bajo
Neutral	Indiferencia	Amabilidad	Medio	Neutral	Indiferencia	Amabilidad	Bajo
Positivo	Interés	Amabilidad	Alto	Neutral	Indiferencia	Amabilidad	Medio
Neutral	Indecisión	Profesionalismo	Medio	Neutral	Indiferencia	Amabilidad	Medio
Negativo	Duda	Amabilidad	Bajo	Negativo	duda	amabilidad	Bajo
Negativo	Frustración	Amabilidad	Alto	Neutral	Frustración	Amabilidad	Bajo
Neutral	Receptividad	Amabilidad	Medio	Neutral	Receptividad	Cordialidad	Bajo
Neutral	Indiferencia	Amabilidad	Medio	Neutral	Indiferencia	Amabilidad	Bajo
Neutral	Desinterés	Profesionalismo	Bajo	Negativo	Indecisión	Amabilidad	Bajo
Neutral	Interés	Amabilidad	Alto	Neutral	Receptividad	Amabilidad	Medio
Neutral	Desinterés	Amabilidad	Bajo	Negativo	Desinterés	Amabilidad	Bajo
Neutral	Interés	Amabilidad	Medio	Neutral	Indiferencia	Amabilidad	Bajo
Positivo	Interés	Amabilidad	Alto	Neutral	Indiferencia	Amabilidad	Bajo
Neutral	Indiferencia	Amabilidad	Bajo	Neutral	Indiferencia	Amabilidad	Bajo
Positivo	Interés	Profesionalismo	Alto	Neutral	interés	amabilidad	Medio
Neutral	Desinterés	Amabilidad	Bajo	Negativo	Desinterés	Amabilidad	Bajo
Neutral	Interés	Amabilidad	Alto	Neutral	Interés	Empatía	Medio
Neutral	Interés	Amabilidad	Alto	Positivo	Interés	Amabilidad	Alto
Neutral	Duda	Amabilidad	Medio	Neutral	Duda	Amabilidad	Bajo
Neutral	Desinterés	Amabilidad	Bajo	Neutral	Interés	Amabilidad	Medio
Neutral	Indiferencia	Amabilidad	Bajo	Neutral	Indiferencia	Amabilidad	Bajo
Positivo	Duda	Amabilidad	Bajo	Neutral	Duda	Claridad	Medio
Neutral	Indiferencia	Amabilidad	Medio	Neutral	Neutralidad	Amabilidad	Medio

Figura 6.1: Comparación entre las evaluaciones humanas y las generadas por la *IA* en las 35 llamadas analizadas.

Para aplicar estas métricas, fue necesario primero transformar esta información en un dataset estructurado, lo que permitió organizar las respuestas por llamada, evaluador, métrica y valor observado.

El proceso de conversión de los datos consistió en homogenizar todos los valores textuales mediante una normalización básica: se convirtieron los textos a minúsculas, se eliminaron tildes y se estandarizaron términos similares (por ejemplo, “interés” e “interes”) para evitar diferencias ortográficas durante el análisis. Adicionalmente, a cada llamada se le asignó un identificador único secuencial.

Posteriormente, la información fue reorganizada manteniendo una fila por llamada y añadiendo columnas de coincidencia (**_Match*) que indican si la respuesta de la IA coincidió con la del evaluador humano en cada métrica. Esta estructura permite calcular directamente las métricas de precisión necesarias para la validación del modelo.

El código utilizado para llevar a cabo esta transformación fue desarrollado en *Python* empleando las librerías `pandas` y `numpy`. Por cuestiones de brevedad, dicho código ha sido ubicado en el Anexo 9, donde puede ser consultado en su totalidad.

6.4.2. Métricas de Evaluación

Tras la creación del dataset, el siguiente paso fue evaluar el desempeño del modelo en las cuatro métricas de interés ya definidas (*Sentimiento General*, *Emoción del Candidato*, *Emoción del Agente*, *Interés del Candidato*). Para esto, se utilizaron dos métricas ampliamente reconocidas en aprendizaje automático: precisión y *F1-score*.

- **Precisión (*Precision*)** se define como la proporción de verdaderos positivos entre todas las predicciones positivas realizadas por el modelo [58]. Matemáticamente se expresa como:

$$\text{Precision} = \frac{TP}{TP + FP}$$

  Donde TP representa los verdaderos positivos y FP los falsos positivos. Esta métrica responde a la pregunta: "De todos los casos que el modelo clasificó como positivos, ¿cuántos fueron realmente positivos?" y permite evaluar qué tan confiables son las predicciones positivas del modelo.

- ***F1-Score*** es la media armónica entre precisión y recall [58, 59], calculada como:

$$F1 = 2 \times \frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}}$$

  Esta métrica equilibra la importancia de precisión y recall en una sola medida, siendo especialmente útil cuando ambas métricas son igualmente relevantes para la evaluación del modelo. Según Sadat-Mohammadi et al. el *F1-Score* penaliza valores extremos en cualquiera de las dos métricas, proporcionando una evaluación balanceada del rendimiento del clasificador [59].

Una vez comprendidas las definiciones formales de precisión y *F1-score*, se procedió a implementar un evaluador en *Python* que aplicara estas métricas sobre el dataset previamente estructurado. El objetivo fue medir el rendimiento del modelo *GPT-4o* al clasificar atributos conversacionales en comparación con las evaluaciones humanas.

La implementación se realizó mediante una clase denominada ***SimpleClassificationEvaluator***, que organiza todo el flujo de análisis: desde la carga de datos hasta la obtención de métricas agregadas por dimensión. A continuación, se describen sus componentes principales.

Para comenzar el análisis, se cargó el archivo *CSV* resultante del dataset estructurado previamente procesado. A partir de estos datos, se empleó el método ***calculate_metrics*** para obtener las métricas fundamentales de evaluación.

Esta función recibe dos listas de etiquetas —una correspondiente a las evaluaciones humanas (*y_true_clean*) y otra a las predicciones generadas por *GPT-4o* (*y_pred_clean*)— y calcula medidas estándar de desempeño en clasificación multiclase.

```
accuracy = accuracy_score(y_true_clean, y_pred_clean)

precision, recall, f1, support = precision_recall_fscore_support(
    y_true_clean, y_pred_clean, average='macro', zero_division=0
)
```

El primer cálculo corresponde a la métrica de *accuracy* (exactitud global), que representa la proporción de coincidencias exactas entre las evaluaciones humanas y las predicciones del modelo. Posteriormente, se obtienen los valores de *precision*, *recall* y *F1-score* aplicando un promedio macro (`average='macro'`), que calcula las métricas para cada clase de forma independiente y luego promedia los resultados sin ponderar por frecuencia.

Los resultados del análisis muestran que GPT-4o logró una precisión promedio del 77.9 % y un *F1-Score* de 0.665 al comparar sus evaluaciones con las realizadas manualmente por el equipo de trabajo. Estos números indican que el modelo tiene un desempeño sólido al momento de replicar las evaluaciones humanas en las distintas dimensiones analizadas.

Dimensión	**Accuracy**	**Precision**	**Recall**	**F1-Score**	**N**
Sentimiento General	0.771	0.758	0.665	0.654	35
Emoción Candidato	0.771	0.875	0.817	0.818	35
Emoción Agente	0.829	0.579	0.629	0.526	35
Interés Candidato	0.743	0.810	0.651	0.661	35
Promedio	0.779	0.756	0.691	0.665	35

Cuadro 6.1: Evaluación de Precisión del Modelo GPT-4o

En la Tabla 6.4.2 se puede observar que el modelo presenta variaciones en su desempeño según la dimensión evaluada. La identificación de la emoción del candidato fue donde *GPT-4o* obtuvo mejores resultados, con un *F1-Score* de 0.818, seguido por el interés del candidato (0.661) y el sentimiento general (0.654). Sin embargo, la evaluación de la emoción del agente registró el menor desempeño con un *F1-Score* de 0.526.

Es importante señalar que el propio análisis del equipo de trabajo presenta limitaciones, ya que algunas de las clasificaciones dependen fuertemente del contexto de la conversación y pueden resultar imprecisas o interpretables de distintas maneras, lo cual introduce un margen de ambigüedad en la evaluación.

6.5. Conclusiones de la Evaluación

Los resultados obtenidos en este capítulo permiten evaluar el desempeño del modelo *GPT-4o* en las dos fases del flujo: la transcripción automática de audio y el análisis de sentimientos y emociones. La evaluación de la capacidad de transcripción reveló un *WER* total del 24.4 % sobre 35 grabaciones, lo cual representa una diferencia frente a los niveles de precisión reportados por *OpenAI* en sus evaluaciones oficiales. Esta discrepancia puede explicarse por las condiciones particulares del análisis, donde las transcripciones de referencia estuvieron sujetas a limitaciones en la comprensión del audio por parte del equipo evaluador, mientras que el modelo *GPT-4o* tiende a realizar reformulaciones semánticas en lugar de transcripciones literales, manteniendo el significado pero alterando la estructura textual exacta.

Por otra parte, el análisis automatizado de sentimientos y emociones arrojó una precisión promedio del 77.9 % y un *F1-Score* de 0.665, lo que indica una capacidad aceptable del modelo para replicar los criterios de evaluación descritos por el equipo de trabajo. Los resultados muestran variaciones significativas entre dimensiones, siendo la identificación de la emoción del candidato la más exitosa (*F1-Score*: 0.818). Es importante considerar que estos resultados están también condicionados por posibles errores humanos en la asignación de valores durante la evaluación manual de las llamadas, lo que puede haber introducido cierto grado de ambigüedad y justificala ausencia de calificaciones más altas.

En conjunto, estos resultados demuestran que el modelo *GPT-4o* posee las capacidades técnicas necesarias para implementar el prototipo de automatización de análisis de grabaciones telefónicas en el contexto universitario. La combinación de una transcripción con precisión del 75.6 % y un análisis de sentimientos con precisión del 77.9 % establece una base sólida para la automatización de procesos que tradicionalmente requerían evaluación manual.

Capítulo 7

Conclusiones

El desarrollo de este proyecto permitió demostrar la posibilidad de automatizar el análisis de comunicaciones telefónicas en el proceso de promoción estudiantil. Mediante un flujo implementado en `n8n` y el uso de modelos de lenguaje de gran escala, como `GPT-4o`, se logró transformar grabaciones de audio estáticas en información estructurada sobre sentimientos, intenciones y temáticas discutidas durante las llamadas. Esta información, una vez integrada en un *dashboard* dinámico, evidenció su potencial para apoyar la exploración de datos y la identificación de patrones en el área de promoción de la Pontificia Universidad Javeriana Cali.

El desarrollo del prototipo siguió un enfoque metodológico iterativo que combinó principios del manifiesto ágil con el ciclo "Construir–Medir–Aprender" de Lean Startup (Capítulo 3). Este enfoque permitió avanzar de forma progresiva desde la identificación de requerimientos hasta la validación de funcionalidades, incorporando ajustes en cada iteración con base en pruebas técnicas y retroalimentación del equipo institucional. La participación de usuarios del área de Promoción Estudiantil en las etapas iniciales contribuyó a la identificación de requerimientos, a partir de los cuales se definieron criterios específicos para el preprocesamiento, clasificación y organización de la información obtenida en las comunicaciones telefónicas. Como resultado, fue posible construir un flujo de análisis automatizado que responde a las condiciones reales del proceso de atención telefónica.

En cuanto a la selección de herramientas, la fase exploratoria (Capítulo 4.1) evaluó enfoques como la combinación de *VADER* para el análisis léxico de sentimientos y *DistilRoBERTa* para la comprensión contextual profunda, lo que proporcionó un balance inicial adecuado para la interpretación de los datos. Sin embargo, la solución final del prototipo se decantó por la integración de la *API* de *gpt-4o-transcribe* de *OpenAI* para la transcripción y los agentes de *GPT-4o* para el análisis semántico. La elección de n8n como orquestador del flujo automatizado (Capítulo 4.2) permitió encapsular cada etapa del proceso, facilitando su monitoreo y potencial replicación en otros contextos similares.

El desarrollo del dashboard interactivo en Tableau (Capítulo 5) constituyó un

entregable estratégico al servir como el integrador clave de los resultados. Este dashboard se diseñó siguiendo principios fundamentales como la jerarquía visual de Tufte, que guía la atención del usuario hacia la información más crítica, y el principio de simplicidad para reducir la carga cognitiva, limitando las visualizaciones principales y usando espacios en blanco estratégicamente. Al traducir los análisis técnicos en información organizada y comprensible para el equipo institucional, esta visualización permitió explorar con rapidez las emociones predominantes en las llamadas, la incidencia de temáticas y el interés de los prospectos , apoyando la identificación de patrones y la exploración de datos de forma intuitiva. Su diseño inclusivo y la optimización para diferentes audiencias (directivos, coordinadores, agentes) refuerzan su potencial como herramienta de apoyo para el área de promoción estudiantil.

En términos de los resultados técnicos de la evaluación (Capítulo 6), se obtuvo un *WER* promedio del 24.4 % para la transcripción y un *F1-Score* de 0.665 en la clasificación de emociones e intenciones. Si bien estas métricas no alcanzan los valores reportados en *benchmarks* controlados, se consideran consistentes con las condiciones del dominio evaluado, caracterizado por ruido en las grabaciones, lenguaje conversacional informal y variabilidad emocional. Además, se identificó que parte de la diferencia entre las predicciones del modelo y las etiquetas de referencia puede estar relacionada con errores humanos durante el proceso de anotación manual.

El desarrollo de este proyecto evidenció el cumplimiento de los objetivos planteados a través de la implementación de un flujo automatizado, reproducible y adaptable, capaz de extraer valor de las interacciones con prospectos. Se logró analizar automáticamente las comunicaciones telefónicas, transformando las grabaciones en transcripciones y, posteriormente, en información estructurada. Para ello, se definieron criterios de preprocesamiento de datos y, tras una evaluación comparativa de diferentes enfoques tecnológicos, se implementó una solución automatizada basada en flujos de n8n integrados con modelos de lenguaje a gran escala. Asimismo, el prototipo fue evaluado mediante métricas de precisión y complementado con visualizaciones interactivas que presentan la información organizada para el usuario institucional. Aunque se trata de un prototipo en una etapa inicial, sus resultados y arquitectura validan la viabilidad de automatizar procesos tradicionalmente manuales mediante inteligencia artificial, abriendo la puerta a futuras ampliaciones en otros contextos institucionales.

Finalmente, este trabajo permitió enfrentar una problemática real del sistema institucional y proponer una solución basada en inteligencia artificial que responde a necesidades concretas del proceso de promoción. Representó también una valiosa experiencia formativa, al permitir aplicar conocimientos técnicos en un contexto práctico y estudiar la complejidad de los procesos reales de la universidad. A lo largo del desarrollo, se interactuó con distintos actores institucionales, incluyendo coordinadores y personal del área tecnológica, lo que facilitó alinear la propuesta con el contexto operativo y revisar su viabilidad como prueba de concepto.

Capítulo 8

Trabajo Futuro

8.1. Trabajos Futuros Propuestos

La entrega de un prototipo funcional representó el objetivo central de este trabajo de grado. Aunque su desarrollo se enfocó en la validación técnica dentro de un entorno controlado, los resultados obtenidos permiten proyectar escenarios de evolución hacia entornos operativos. En este sentido, el prototipo puede ser adaptado, ampliado o integrado como componente de una solución institucional más amplia. A continuación, se presentan algunas líneas posibles de trabajo futuro que permitirían extender su alcance o explorar nuevas aplicaciones a partir de su arquitectura base:

8.1.1. Integración con el *CRM* de *Salesforce* de la Universidad Javeriana Cali

Una posible línea de trabajo futuro consiste en evaluar, junto con las áreas pertinentes de la universidad, la integración del flujo automatizado con el ecosistema de *Salesforce*. Esto permitiría acceder directamente a las grabaciones de llamadas almacenadas en el *CRM* institucional, evitando la dependencia de repositorios externos y centralizando el proceso en una plataforma ya adoptada por la universidad.

Además de la extracción de grabaciones, también podría contemplarse el registro automático de los resultados del análisis en objetos personalizados dentro de *Salesforce*. De este modo, los usuarios institucionales tendrían acceso no solo a las visualizaciones del dashboard, sino también a los datos estructurados generados por el flujo, todo desde el entorno del *CRM*.

Esta propuesta también es coherente con el uso de *Tableau* como herramienta de visualización, dado que esta plataforma cuenta con conectores nativos para *Salesforce*. Por lo tanto, sería posible mantener la lógica actual del flujo de análisis y visualización sin necesidad de rediseñar la arquitectura base.

La integración con *Salesforce*, además de mejorar la automatización general, permitiría reducir dependencias con servicios externos como *Google Cloud* y alinear el funcionamiento del prototipo con los recursos tecnológicos oficiales de la universidad.

8.1.2. Ampliar el Análisis de Llamadas

Otra opción complementaria para futuras mejoras consiste en profundizar el análisis de las llamadas incluyendo nuevas dimensiones no abordadas en el prototipo actual, como la claridad del mensaje institucional, la detección de objeciones recurrentes o la identificación de factores emocionales que influyen en la decisión del prospecto. Estas variables podrían enriquecer la comprensión del comportamiento comunicativo y aportar insumos para ajustar las estrategias de captación.

Si bien esta ampliación conllevaría un incremento en el número de solicitudes a la *API* de *OpenAI* —con su correspondiente impacto en costos— también ofrecería un análisis más detallado y estratégico, alineado con las necesidades del área de promoción. Para lograrlo, se recomienda realizar un trabajo conjunto con los asesores del área de atención al cliente y el personal institucional, de modo que los nuevos criterios a incluir respondan a inquietudes reales y resulten útiles en la práctica.

8.1.3. Inclusión de Más Actividades

La evolución del prototipo hacia una solución operativa requiere la inclusión de nuevas actividades que acompañen el proceso de integración, validación y mejora continua. En particular, para cumplir con los trabajos futuros propuestos —como la integración con *Salesforce*— es necesario fortalecer la articulación con las áreas técnicas y administrativas de la universidad.

Estas actividades deben priorizar la comunicación con los equipos encargados del *CRM* institucional y otros sistemas de soporte, con el fin de alinear el desarrollo con los lineamientos tecnológicos de la institución. Además, será necesario gestionar los accesos a entornos de prueba dentro de *Salesforce*, que permitan validar los flujos sin afectar datos en producción.

- Establecer reuniones con el equipo técnico de *Salesforce* para identificar rutas de integración.
- Solicitar acceso a un entorno de pruebas de *Salesforce* para validar la lectura y escritura de datos desde el flujo de *n8n*.
- Coordinar con el área de promoción el uso de conectores entre *Tableau* y *Salesforce*.
- Documentar técnicamente las extensiones realizadas para facilitar su mantenimiento.

La ejecución de estas actividades permitiría dar continuidad al trabajo realizado, manteniendo el enfoque de validación práctica con usuarios institucionales y promoviendo una adopción progresiva de la herramienta en contextos reales.

Bibliografía

[1] P. U. J. Cali, "Organigrama - Pontificia Universidad Javeriana Cali." En línea, 2025. Consultado el 14 de julio de 2025.

[2] A. S. Sunar and M. S. Khalid, "Natural language processing of student's feedback to instructors: A systematic review," *IEEE Transactions on Learning Technologies*, vol. 17, pp. 741–753, 2024.

[3] M. S. Hossen, M. N. A. Chowdhury, A. M. Sristy, *et al.*, "Automated university complaint management system by integrating natural language processing and machine learning," *Frontiers in Linguistics and Semiotics*, vol. 4, no. 1, pp. 1–9, 2022.

[4] T. D. Pham, D. Vo, F. Li, K. Baker, B. Han, L. Lindsay, M. Pashna, and R. Rowley, "Natural language processing for analysis of student online sentiment in a postgraduate program," *Pacific Journal of Technology Enhanced Learning*, 2020.

[5] IBM, "¿qué es el análisis de sentimiento?," *IBM Topics*, 2020.

[6] Association for Computing Machinery, "Acm computing classification system," 2024. Consultado el 3 de julio de 2025.

[7] G. Cloud, "¿qué es la inteligencia artificial o ia?." `https://cloud.google.com/learn/what-is-artificial-intelligence?hl=es-419`. Consultado el 10 de mayo de 2025.

[8] C. Stryker and J. Holdsworth, "¿qué es el procesamiento del lenguaje natural (nlp)?." `https://www.ibm.com/mx-es/topics/natural-language-processing`, 2024. Consultado el 10 de mayo de 2025.

[9] M. Azure, "¿qué es el análisis de sentimiento y la minería de opiniones?." `https://learn.microsoft.com/es-es/azure/ai-services/language-service/sentiment-opinion-mining/overview`, 2025. Consultado el 10 de mayo de 2025.

[10] A. Matarazzo and R. Torlone, "A survey on large language models with some insights on their capabilities and limitations," *arXiv preprint arXiv:2501.04040*, 2025. Accedido en julio de 2025.

[11] R. Sheldon and K. Yasar, "What is an ai prompt?." https://www.techtarget.com/searchenterpriseai/definition/AI-prompt, 2024. Accedido el 19 de julio de 2025.

[12] ActiveCampaign, "¿qué es la automatización de flujos de trabajo?." https://www.activecampaign.com/es/glossary/workflow-automation. Consultado el 10 de mayo de 2025.

[13] Tableau, "Dashboards." https://help.tableau.com/current/pro/desktop/es-es/dashboards.htm. Consultado el 10 de mayo de 2025.

[14] L. T. University, "About datasets - datasets for researchers." [Online], 2024. Consultado el 14 de julio de 2025.

[15] R. C. Camp, *Benchmarking: The Search for Industry Best Practices That Lead to Superior Performance*. Milwaukee, WI: ASQC Quality Press, 1989.

[16] J. Amespil, "Prospecto: ¿qué es y cuáles son sus características?." https://www.salesforce.com/mx/blog/prospecto/, marzo 2023. Consultado el 10 de mayo de 2025.

[17] L. Izurieta, "La retroalimentación: el secreto de una excelente relación de trabajo." https://ciavgroup.com/la-retroalimentacion-el-secreto-de-una-excelente-relacion-de-trabajo, marzo 2019. Consultado el 10 de mayo de 2025.

[18] Zendesk, "Call center: ¿qué es y cómo funciona? [guÍa bÁsica]." https://www.zendesk.com.mx/blog/que-es-call-center/, 2024. Consultado el 10 de mayo de 2025.

[19] WeKall, "Telefonía en la nube para empresas." https://wekall.co/.

[20] Salesforce, "Crm: Plataforma de gestión de relaciones con clientes." https://www.salesforce.com/.

[21] C. Álvarez, "Aplicación de procesamiento de lenguaje natural sobre una encuesta de satisfacción," *Repositorio UC*, 2021.

[22] E. Martínez Porras, A. Ramírez Fernández, L. Solís Bastos, and J. A. Díaz González, "Uso de procesamiento de lenguaje natural para procesar respuestas abiertas de una encuesta de opinión pública," *Revista Latinoamericana de Metodología de la Investigación Social*, vol. 15, no. 29, pp. 51–67, 2025.

[23] R. Torres, "Clasificación de clientes y predicción de deserciones usando algoritmos k-means y regresión logística," *Repositorio PUCE*, 2023.

[24] L. Zhang and E. Ng, "When siri knows how you feel: Study of machine learning in automatic sentiment recognition from human speech," *NTU Final Year Project*, 2018.

[25] B. C. Das, M. H. Amini, and Y. Wu, "Security and privacy challenges of large language models: A survey," *arXiv preprint arXiv:2402.00888*, 2024.

[26] OpenAI, "Api data usage policies," 2024. Consultado el 14 de julio de 2025.

[27] OpenAI, "Data controls in the openai platform," 2024. Consultado el 14 de julio de 2025.

[28] OpenAI, "Seguridad y privacidad," 2024. Consultado el 14 de julio de 2025.

[29] K. Beck, M. Beedle, A. van Bennekum, A. Cockburn, W. Cunningham, M. Fowler, J. Grenning, J. Highsmith, A. Hunt, R. Jeffries, J. Kern, B. Marick, R. C. Martin, S. Mellor, K. Schwaber, J. Sutherland, and D. Thomas, "Manifesto for agile software development," 2001.

[30] E. Ries, *The Lean Startup: How Today's Entrepreneurs Use Continuous Innovation to Create Radically Successful Businesses*. Crown Business, 2011.

[31] L. Hoti and E. Ajdari, "Sentiment analysis using the vader model for assessing company services based on posts on social media," *ResearchGate*, 2023.

[32] C. Hutto, "Vader sentiment analysis documentation," 2024. Consultado el 3 de julio de 2025.

[33] n8n, "n8n documentation." `https://docs.n8n.io`, 2024.

[34] I. Miller, "Vader sentiment analysis." https://hex.tech/templates/sentiment-analysis/vader-sentiment-analysis, 2024. Consultado el 9 de Julio del 2025.

[35] D. AI, "Paraphrase spanish distilroberta," 2022. Consultado el 9 de Julio del 2025.

[36] J.-W. Kang and S. yong Choi, "Comparative investigation of gpt and finbert's sentiment analysis performance in news across different sectors," *Electronics*, 2025. Consultado el 9 de Julio del 2025.

[37] S. Brown, "The c4 model for visualising software architecture." `https://c4model.com`, 2020. Consultado el 19 de julio de 2025.

[38] Chakravarthula, Baucom, Narayanan, and Georgiou, "Multimodal behavior prediction using missing modality imputation," *arXiv preprint arXiv:1911.09515*, 2019.

[39] n8n Docs, "Tools ai agent node," 2024. Consultado el 12 de julio de 2025.

[40] Y. Wang and Z. Luo, "Enhance multi-domain sentiment analysis of review texts through prompting strategies," *arXiv preprint arXiv:2309.02045*, 2024.

[41] OpenAI, "How do i create a good prompt for an ai model like gpt-4?," 2025. Consultado el 12 de julio de 2025.

[42] S. Few, *Information Dashboard Design: The Effective Visual Communication of Data*. O'Reilly Media, 2006.

[43] G. Sedrakyan *et al.*, "Guiding the choice of learning dashboard visualizations: Linking dashboard design and data visualization concepts," *Computers & Education*, vol. 145, p. 103738, 2020.

[44] E. R. Tufte, *The Visual Display of Quantitative Information*. Graphics Press, 2001.

[45] UXPin, "Effective dashboard design principles for 2025," 2024. Consultado el 13 de Julio 2025.

[46] Sisense, "4 dashboard design principles for better data visualization," 2025. Consultado el 13 de Julio 2025.

[47] AufaitUX, "30 proven dashboard design principles for better data display," 2025. Consultado el 13 de Julio 2025.

[48] Yellowfin, "10 key dashboard design principles: Analytics best practice," 2024. Consultado el 13 de Julio 2025.

[49] A. Hill, "Are pie charts evil? an assessment of the value of pie and donut charts compared to bar charts," *SAGE Journals*, 2024.

[50] ArXiv, "Does the infamous pie chart really hurt decision-making in the real world? assessing the role of visualization in high-level academic decisions." `http://arxiv.org/html/2410.04686v1`, 2024. Preprint.

[51] Alchemer, "Presenting survey results: Pie charts vs. bar graphs," 2024. Consultado el 13 de Julio 2025.

[52] Tableau, "What is data visualization? definition, examples, and learning resources," 2025. Consultado el 13 de Julio 2025.

[53] ScienceDirect, "The efficacy of stacked bar charts in supporting single-attribute and overall-attribute comparisons," *Visual Informatics*, vol. 2, no. 4, pp. 247–256, 2018.

[54] Toptal, "Dashboard design: Best practices with examples," 2017. Consultado el 13 de Julio 2025.

[55] W. contributors, "Word error rate," 2025. Consultado el 14 de julio de 2025.

[56] OpenAI, "Introducing our next-generation audio models." https://openai.com/index/introducing-our-next-generation-audio-models, 2025. Consultado el 14 de julio de 2025.

[57] OpenAI, "Hello gpt-4o." https://openai.com/es-419/index/hello-gpt-4o, 2024. Consultado el 14 de julio de 2025.

[58] G. Developers, "Classification: Accuracy, recall, precision, and related metrics." Machine Learning Crash Course, 2024. Consultado el 14 de julio de 2025.

[59] M. Sadat-Mohammadi, F. Asadi, Z. Eskandari, A. Kouhestani, A. Olfatbakhsh, Z. Mansoori, and R. Jafari, "On evaluation metrics for medical applications of artificial intelligence," *Scientific Reports*, vol. 12, p. 5979, Apr 2022.

Capítulo 9

Anexos

Anexo #1, Transcripción de llamada de prueba

Operador: Buenos días, ¿hablo con el señor Andrés Gutiérrez?

Prospecto: Sí, él habla. ¿Con quién tengo el gusto?

Operador: Mi nombre es Laura Pérez, y lo llamo del área de promoción estudiantil de la Pontificia Universidad Javeriana Cali. ¿Cómo se encuentra hoy?

Prospecto: Muy bien, gracias. ¿En qué puedo ayudarle?

Operador: Andrés, el motivo de mi llamada es compartirle información sobre los programas de pregrado de nuestra universidad, y cómo podríamos apoyarlo a alcanzar sus metas académicas. Vemos que mostró interés en el programa de Ingeniería de Sistemas y Computación. ¿Es correcto?

Prospecto: Sí, lo es. Estuve revisando la página web y me pareció interesante, pero tengo algunas dudas sobre el proceso de inscripción.

Operador: Claro, estoy aquí para resolver todas sus inquietudes. Pero antes de eso, me gustaría conocer un poco más sobre sus intereses. ¿Qué le llamó la atención de Ingeniería de Sistemas?

Prospecto: Bueno, me interesa mucho todo lo relacionado con programación y desarrollo de software. He trabajado en algunos proyectos personales y me gustaría profundizar en eso.

Operador: ¡Excelente! Nuestro programa está diseñado justamente para estudiantes como usted, apasionados por la tecnología y con ganas de innovar. Contamos con laboratorios de última generación y convenios con empresas como Google y Microsoft para prácticas. ¿Le interesaría explorar más sobre estas oportunidades?

Prospecto: Sí, eso suena interesante, pero tengo algunas preocupaciones. Me preocupa un poco el costo de la matrícula, y también escuché que algunos trámites son bastante complicados.

Operador: Entiendo perfectamente su preocupación. En cuanto a los costos, la universidad cuenta con múltiples opciones de financiación, becas por mérito académico y apoyo económico para facilitar el acceso a la educación. Además, el proceso de inscripción es muy sencillo, y yo puedo acompañarlo paso a paso. ¿Le gustaría recibir más información sobre nuestras becas?

Prospecto: Sí, me gustaría saber más sobre las becas. Soy un estudiante dedicado y creo que podría aplicar a alguna.

Operador: ¡Por supuesto! Tenemos becas que cubren desde el 20% hasta el 100% de la matrícula, dependiendo de su rendimiento académico y resultados de las pruebas Saber 11. Si le parece bien, puedo enviarle un correo con toda esta información y agendar una cita para que hable con nuestro equipo de admisiones.

Prospecto: Eso estaría bien, pero aún tengo una duda. He escuchado que algunas universidades no ofrecen mucho apoyo a los estudiantes después de graduarse. ¿Qué tal es el acompañamiento de la Javeriana en este sentido?

Operador: Es una muy buena pregunta. Nuestra universidad tiene una oficina de egresados que brinda apoyo continuo. Ofrecemos acceso a nuestra bolsa de empleo, capacitaciones y eventos exclusivos para fortalecer su perfil profesional incluso después de graduarse. Además, muchos de nuestros egresados logran posiciones destacadas en empresas nacionales e internacionales.

Prospecto: Eso es alentador. Creo que la universidad podría ser una buena opción para mí.

Operador: Me alegra escuchar eso, Andrés. Si le parece, puedo ayudarlo a completar su inscripción en línea en este momento, o podemos agendar una llamada para profundizar más en sus intereses y necesidades.

```
Prospecto: Prefiero agendar la llamada. Quiero revisar toda la
información que me envíe antes de decidirme.

Operador: Perfecto, entonces le enviaré un correo con los detalles
del programa, las becas y el acompañamiento post-graduación.
También incluiré un enlace para agendar su cita. ¿Qué día y hora
le viene mejor para nuestra próxima llamada?

Prospecto: El miércoles por la tarde estaría bien.

Operador: Excelente, quedamos entonces el miércoles a las 3:00 PM.
Si tiene alguna pregunta antes de nuestra próxima llamada, no dude
en contactarme. Le enviaré mi correo y teléfono.

Prospecto: Muchas gracias, Laura.

Operador: Gracias a usted, Andrés. Que tenga un excelente día.
¡Hasta pronto!

Prospecto: Igualmente. Hasta luego.
```

Esta llamada simulada, utilizada durante las pruebas iniciales de las herramientas exploradas, fue generada mediante inteligencia artificial a través de los modelos de OpenAI. El objetivo de esta conversación fue mantener un formato sencillo de interpretar, por lo que se asignaron nombres a cada participante y se estructuró un diálogo claro. Además, se incluyeron temas relacionados con la universidad, como empleo y capacitaciones, así como expresiones emocionales evidentes por parte del prospecto, con el fin de observar cómo las herramientas eran capaces de identificar e interpretar estos elementos en el análisis.

Anexo #2, Resultados de la Clasificación VADER + DistilRoBERTa

Hablante	Texto	Sentimiento	Intención
Operador	Buenos días, ¿hablo con el señor Andrés Gutiérrez?	neutro	neutral
Prospecto	Sí, él habla. ¿Con quién tengo el gusto?	neutro	neutral
Operador	Mi nombre es Laura Pérez, y lo llamo del área...	neutro	fear
Prospecto	Muy bien, gracias. ¿En qué puedo ayudarle?	neutro	neutral
Operador	Andrés, el motivo de mi llamada es compartirle...	neutro	neutral
Prospecto	Sí, lo es. Estuve revisando la página web y me...	neutro	neutral
Operador	Claro, estoy aquí para resolver todas sus inqu...	positivo	neutral

Hablante	Texto	Sentimiento	Intención
Prospecto	Bueno, me interesa mucho todo lo relacionado c...	neutro	neutral
Operador	¡Excelente! Nuestro programa está diseñado jus...	neutro	neutral
Prospecto	Sí, eso suena interesante, pero tengo algunas...	neutro	neutral
Operador	Entiendo perfectamente su preocupación. En cua...	neutro	neutral
Prospecto	Sí, me gustaría saber más sobre las becas. Soy...	neutro	neutral
Operador	¡Por supuesto! Tenemos becas que cubren desde ...	neutro	neutral
Prospecto	Eso estaría bien, pero aún tengo una duda. He ...	negativo	neutral
Operador	Es una muy buena pregunta. Nuestra universidad...	neutro	neutral
Prospecto	Eso es alentador. Creo que la universidad podr...	neutro	neutral
Operador	Me alegra escuchar eso, Andrés. Si le parece, ...	neutro	neutral
Prospecto	Prefiero agendar la llamada. Quiero revisar to...	neutro	neutral
Operador	Perfecto, entonces le enviaré un correo con lo...	positivo	neutral
Prospecto	El miércoles por la tarde estaría bien.	neutro	neutral
Operador	Excelente, quedamos entonces el miércoles a la...	negativo	neutral
Prospecto	Muchas gracias, Laura.	neutro	joy
Operador	Gracias a usted, Andrés. Que tenga un excelent...	neutro	neutral
Prospecto	Igualmente. Hasta luego.	neutro	neutral

Esta tabla presenta los resultados obtenidos al aplicar los modelos VADER y DistilRoBERTa sobre cada línea de la transcripción simulada. En ella se observa la asignación de etiquetas de sentimiento e intención para cada intervención, lo que permite analizar de forma detallada cómo cada modelo interpreta el contenido en función del hablante y el contexto.

Anexo #3, Resultado de la transcripción con Whisper

```
Hola, muy buenas tardes. Mi nombre es Sebastián y le hablo del
Departamento de Admisiones de la Universidad Javeriana. ¿Con quién
tengo el gusto? Hola, me llamo Santiago. Mucho gusto, Santiago.
Estoy contactándote porque registraste tu interés en conocer más
sobre nuestros programas académicos. ¿Te gustaría que te brindara
información sobre alguna carrera en particular? Sí, me interesa
Inhería Sistemas, pero quiero saber más sobre los requisitos del
proceso de admisión. ¡Excelente lección! En la Universidad Javeriana
ofrecemos un programa sólido de Inhería Sistemas con docentes
altamente aplicados de oportunidades de práctica en empresas
reconocidas. Sí, me gustaría saber sobre los costos y si hay becas
disponibles. Por supuesto, el costo de la matrícula es de 20 palitos
y la mensualidad es de un palo. Pero contamos con diferentes
opciones de financiamiento y becas por excelencia académica y
necesidad económica.
```

Este resultado fue obtenido a partir de una grabación creada manualmente por el equipo de trabajo y corresponde a una transcripción casi literal del audio. Aunque se aproxima con bastante fidelidad al contenido original, es posible identificar algunos errores puntuales en el texto. Este comportamiento contrasta con lo observado en el modelo `gpt-4o-transcription`, el cual no realizaba una transcripción literal, pero sí lograba mantener el sentido general de la conversación. En el caso de Whisper, al no comprender ciertos fragmentos del audio, tiende a generar transcripciones imprecisas.

Anexo #4, Código de extracción de Urls

```
urls_con_ids = []

with open('extract.csv', encoding='latin-1', newline='') as f:

    csv_reader = csv.DictReader(f)
```

```
    for row in csv_reader:
        description = row.get('DESCRIPTION', '')
        call_id = row.get('C_DIGO_LLAMADA__C', '').strip()

        # Buscar URLs de grabaciones en la descripción
        urls = re.
            findall('https://[^\s",]+\.mp3\?public=true', description)

        # Si encontramos URLs y tenemos un Call ID válido
        if urls and call_id:
            for url in urls:
                urls_con_ids.append(f"{url}|{call_id}")

with open('urls_grabaciones.txt', 'w', encoding='utf-8') as output_file:
    for entrada in urls_con_ids:
        output_file.write(entrada + '\n')
```

Este fue el código desarrollado en *Python* para extraer las *URLs* y los *ID* de las grabaciones. El proceso consiste en leer el archivo `.csv` proporcionado por la universidad y filtrar las *URLs* a partir del contenido de la columna *DESCRIPTION*, utilizando la presencia del prefijo `https` como criterio de extracción. De manera similar, se obtienen los *ID* desde su columna correspondiente, y ambos resultados se registran en un archivo de texto plano siguiendo el formato especificado.

Anexo #5, Prompts utilizados para los Tool Agent

Prompt para Analizar sentimiento, emociones e interés

```
#Rol
Eres un experto analista de sentimientos especializado en
conversaciones telefónicas del sector de atención al cliente
en una universidad. Sigues un proceso metodológico paso a paso
para garantizar precisión.

# Objetivo
Analizar el sentimiento, las emociones y el nivel de interés del
candidato durante una llamada con la universidad.

# Proceso de Análisis Metodológico
1. Evalúa el tono general de la conversación, prestando especial
atención al estado emocional y la disposición del *prospecto*.
2. Clasifica el sentimiento general como:
```

```
   - "Positivo": el prospecto muestra entusiasmo, claridad de
   intención o actitud receptiva. Se expresa con motivación o confianza.
   - "Negativo": el prospecto muestra indecisión, desinterés, evasión
   o frustración. Incluso si el tono es educado, si hay falta de
   compromiso emocional o verbal, se considera negativo.
   - "Neutral": la conversación es informativa y sin carga emocional
   clara por ninguna de las partes. El prospecto se muestra simplemente
   receptivo pero sin emociones definidas.
3. Identifica la emoción predominante del candidato (ej: entusiasmo,
desinterés, frustración, duda, interés, ansiedad, indiferencia).
4. Identifica la emoción predominante del agente universitario
(ej: empatía, claridad, impaciencia, amabilidad, desconexión).
5. Estima el nivel de interés general del candidato en la conversación,
considerando su participación, entusiasmo, tipo de preguntas y lenguaje
utilizado. Usa una de las siguientes categorías:
   - "Alto": el candidato expresa intención clara de avanzar, hace
   preguntas clave, muestra entusiasmo.
   - "Medio": el candidato se interesa parcialmente, hace preguntas
   generales, sugiere seguir averiguando.
   - "Bajo": el candidato muestra poco o ningún interés, responde
   con frases cortas o evasivas.

# Transcripción de la llamada:
{{ $json.text }}

# Output (JSON estricto)
{
  "sentimiento_general": "",
  "emocion_candidato": "",
  "emocion_agente": "",
  "nivel_interes_candidato": ""
}
```

Prompt para Acciones de mejora

```
#Rol
Eres un consultor experto en calidad de atención al cliente del sector
educativo universitario, especializado en análisis de conversaciones
telefónicas y mejora de procesos de admisión.

# Objetivo
Justificar el sentimiento clasificado y proponer acciones de mejora
sobre la atención o proceso.
```

```
# Instrucciones
1. Justifica brevemente por qué se asignó el sentimiento general.
2. Señala puntos fuertes o débiles de la llamada.
3. Si hay debilidades, sugiere mejoras concretas en aspectos como:
   - tono del agente
   - claridad en la explicación
   - empatía con el prospecto
   - manejo del tiempo o dudas

# Sentimiento general previamente asignado:
{{ $json.output.sentimiento_general }}

# Transcripción de la llamada:
{{ $('Transcripción').item.json.text }}

# Output (JSON estricto)
{
  "justificacion": "La llamada fue fluida y el prospecto expresó
  interés genuino...",
  "areas_de_mejora": "Mejorar la claridad al explicar los pasos
  de admisión."
}
```

Prompt para Temáticas e intenciones

```
#Rol
Eres un analista experto en comportamiento y comunicación, con más
de 10 años de experiencia evaluando intenciones de prospectos
universitarios. Aplicas una metodología sistemática de análisis
conversacional.

# Objetivo
Identificar las temáticas principales de la llamada y las intenciones
del prospecto.

# Instrucciones
1. Analiza los temas discutidos durante la llamada.
2. Clasifica las temáticas en categorías como: proceso de admisión,
costos, becas, programas académicos, modalidad de estudio, dudas generales,
motivaciones personales.
3. Identifica si el candidato muestra intención de avanzar en el proceso
de inscripción (Sí/No).
4. Extrae palabras clave representativas (máximo 5) de la conversación.
5.Extrae todas las preguntas que haya formulado el prospecto en la
```

```
conversación (en texto literal si es posible).

# Transcripción aquí:
{{ $('Transcripción').item.json.text }}

# Output (JSON estricto)
{
  "tematicas_detectadas": ["Becas", "Costos", "Carreras"],
  "intencion_prospecto": "Sí",
  "palabras_clave": ["becas completas", "ingeniería",
  "valor de la matrícula"],
"preguntas_realizadas": [
    "¿Cuando es la fecha de ingreso al semestre?",
    "¿Qué programas académicos están disponibles?",
    "¿Cuando es la fecha de pago de matrícula?"
  ]
}
```

Anexo #6, Código para obtener el DataSet de Resultados

Este script fue desarrollado en *Python* para transformar los resultados de la evaluación manual y de *IA* en dos formatos de dataset: uno en estructura original (*wide*) y otro en formato largo (*long*). A continuación, se explican sus componentes principales:

La función `clean_text` se encarga de unificar el formato de los datos textuales. Convierte el texto a minúsculas, elimina acentos y estandariza valores que podrían aparecer con variaciones. Esta limpieza fue necesaria para facilitar la comparación entre evaluaciones humanas y de *IA*:

```
def clean_text(text):

    if pd.isna(text) or text == "":
        return ""

    text = str(text).lower().strip()

    replacements = {
        'á': 'a', 'é': 'e', 'í': 'i', 'ó': 'o', 'ú': 'u', 'ñ': 'n'
    }
    for old, new in replacements.items():
```

```
        text = text.replace(old, new)

    standardizations = {
        'desinteres': 'desinteres',
        'desinterés': 'desinteres',
        'interes': 'interes',
        'interés': 'interes',
        'indecision': 'indecision',
        'indecisión': 'indecision',
        'profesionalismo empatico': 'profesionalismo',
        'empatia': 'empatia',
        'empata': 'empatia'
    }

    return standardizations.get(text, text)
```

La función `transform_excel_to_dataset` carga el archivo de evaluación, renombra las columnas para mayor claridad, limpia los valores y genera dos versiones del dataset:

- Un *DataFrame* en formato "wide", con las columnas originales y nuevas columnas que indican coincidencias entre humano e *IA*.
- Un *DataFrame* en formato "long", que organiza cada métrica por fila, separando los resultados por evaluador (humano o *IA*), útil para análisis estadístico o visualización.

```
df = pd.read_excel(excel_file, header=1)
df.columns = [ ... ]  # Renombrar columnas
df['ID_Llamada'] = range(1, len(df) + 1)

# Limpieza de texto
for col in df.columns:
    if col != 'ID_Llamada':
        df[col] = df[col].apply(clean_text)

# Conversión a formato long
for _, row in df.iterrows():
    for metric in metrics:
        long_data.append(...)  # Evaluaciones por métrica y evaluador

# Cálculo de coincidencias
for metric in metrics:
    df_wide[f'{metric}_Match'] = ...

return df_wide, df_long
```

La función `generate_summary_stats` permite calcular coincidencias entre evaluaciones humanas y las generadas por el prototipo, tanto por métrica como de forma agregada. Estas estadísticas permiten cuantificar la precisión relativa del análisis automatizado.

```
def generate_summary_stats(df_wide, df_long):

    # Estadísticas por métrica
    metrics = [‘Sentimiento’, ‘Emocion_Candidato’,
    ‘Emocion_Agente’, ‘Interes_Candidato’]

    for metric in metrics:
        metric_data = df_long[df_long[‘Metrica’] == metric]
        unique_values = sorted(metric_data[} ‘Valor’].unique())
        unique_values = [v for v in unique_values if v != ""]

    # Coincidencias por métrica
    for metric in metrics:
        match_col = f‘{metric}_Match’
        if match_col in df_wide.columns:
            matches = df_wide[match_col].sum()
            total = len(df_wide)
            percentage = (matches / total) * 100

    # Precisión global
    match_columns = [f‘{metric}_Match’ for metric in metrics]
    if all(col in df_wide.columns for col in match_columns):
        total_matches = df_wide[match_columns].sum().sum()
        total_evaluations = len(df_wide) * len(metrics)
        global_accuracy = (total_matches / total_evaluations) * 100
```

La función `main` orquesta la ejecución del flujo completo: carga el archivo de evaluación, transforma los datos, genera las estadísticas y guarda los resultados en disco.

```
def main():
    excel_file = "ResultadosHumanosIA.xlsx"
    # Verificar que el archivo existe
    if not os.path.exists(excel_file):
        return
    try:
        # Transformar datos
        df_wide, df_long = transform_excel_to_dataset(excel_file)
```

```
# Generar estadísticas
generate_summary_stats(df_wide, df_long)

# Guardar datasets
files = save_datasets(df_wide, df_long)
```

Anexo #7, Acuerdo de Confidencialidad

A continuación, se presenta el acuerdo de confidencialidad firmado que rige el uso de la información contenida en las grabaciones de llamadas y demás datos sensibles utilizados durante el desarrollo del proyecto.

ACUERDO DE CONFIDENCIALIDAD Y RESERVA EN EL USO DE LA INFORMACIÓN

Entre los suscritos a saber: **VICENTE DURAN CASAS, S.J.,** mayor de edad, vecino de Cali e identificado con la cedula de ciudadanía No. 3.227.972 de Usaquen, quien obra en nombre y representación de la **PONTIFICIA UNIVERSIDAD JAVERIANA**, **SECCIONAL CALI**, institución de educación superior privada, de utilidad común, sin ánimo de lucro, con personería jurídica reconocida por medio de la Resolución N° 73 de 1933, emanada del Ministerio de Gobierno, reconocida como Universidad mediante el Decreto 1297 del 30 de mayo de 1964, sometida a la vigilancia del Ministerio de Educación Nacional, en virtud del poder general que le fue conferido por medio de la escritura pública 497 otorgada el 02 de abril de 2014 en la Notaría 26 del Círculo de Bogotá, quien para efectos de este documento se denominará **LA UNIVERSIDAD** por una parte, y por la otra **LUIS FELIPE MARTÍNEZ** Y **SEBASTIÁN HERNÁNDEZ TRIANA,** identificado (a) como aparece al pie de su firma y que para todos los efectos se llamarán **LOS ESTUDIANTES,** se celebra el presente acuerdo de confidencialidad y reserva en el uso de la Información, previo a:

CONSIDERACIONES

1. Como parte del desarrollo académico y formativo de los estudiantes de pregrado en Ingeniería de Sistemas y Computación, se establece como requisito para la obtención del título la realización de un proyecto de grado.
2. En este contexto, los estudiantes desarrollaron el anteproyecto "*Prototipo para la Clasificación y Análisis de Comunicaciones Telefónicas en el Proceso de Promoción Estudiantil*" bajo la supervisión de la profesora Luisa Rincón, aplicando y fortaleciendo los conocimientos adquiridos a lo largo de su formación para abordar problemáticas y desafíos en el ámbito de la ingeniería de sistemas y computación.
3. Actualmente, los estudiantes se encuentran en la fase de elaboración de su trabajo de grado, para lo cual requieren acceso a información confidencial almacenada en nuestro CRM. Dicha información es fundamental para el desarrollo de su investigación y análisis.

Teniendo en cuenta lo anterior, las partes acuerdan celebrar el presente acuerdo de confidencialidad, el cual se regirá por las siguientes disposiciones:

CLÁUSULAS

PRIMERA.- Objeto: En virtud del presente acuerdo **LOS ESTUDIANTES** asumen la obligación de no revelar, divulgar, o exhibir información financiera, técnica, comercial o científica relacionada con los proyectos en los que participa o tenga conocimiento y otros relacionados o derivados de estos a cargo del Centro de Consultoría y Educación Continua de **LA UNIVERSIDAD**, a persona alguna natural o jurídica, (incluyendo parientes en cualquier grado de consanguinidad o afinidad), ni utilizar o emplear dicha información en su favor o en el de terceros, ya sea de manera directa o indirecta y en perjuicio o no de las anteriores.

SEGUNDA.- Autorización previa: Mediante la firma del presente documento, **LOS ESTUDIANTES** aceptan que sin la previa autorización escrita de **LA UNIVERSIDAD** u orden de autoridad competente en ejercicio de funciones legales y en desarrollo de actuación administrativa o judicial, no podrá revelar, divulgar, exhibir y, en general, dar a conocer a ningún tercero, información o documento alguno que haya recibido o esté por recibir de parte de **LA UNIVERSIDAD**, en razón

de sus funciones, relativo a la información legal, financiera, técnica o comercial relacionada con el giro ordinario de su quehacer o de los proyectos a desarrollar.

En consecuencia, **LOS ESTUDIANTES** reconocen y aceptan que **LA UNIVERSIDAD** y/o **LA EMPRESA** son titulares exclusivos de cualquier derecho sobre dicha información, y que no tendrá derecho alguno sobre la misma, obligándose a no copiarla, duplicarla, sustraerla o comunicarla, para sí o para terceros.

CUARTA.- Cláusula penal: En adición a lo establecido en la cláusula precedente, el incumplimiento por parte de **LOS ESTUDIANTES** les hará incurrir automáticamente en una pena a favor de **LA UNIVERSIDAD** por una suma equivalente cien (100) salarios mínimos legales mensuales vigentes al momento en que se dé el incumplimiento del presente acuerdo, sin perjuicio de que **LA UNIVERSIDAD** pueda adelantar las acciones legales que tiendan al resarcimiento total de los perjuicios causados. El pago de la suma estipulada como pena, no releva a **LOS ESTUDIANTES** del cumplimiento de la obligación de confidencialidad contraído mediante este acuerdo; en consecuencia, deberá pagar esta penalidad a **LA UNIVERSIDAD** cada vez que se presente un incumplimiento de esta índole. En los términos del inciso segundo del artículo 1595 del Código Civil, **LOS ESTUDIANTES** reconocen expresamente que incurrirá en la pena establecida en la presente cláusula a partir del momento en que ejecute cualquiera de los hechos o conductas que se ha obligado a abstenerse.

Parágrafo. Sin perjuicio de la pena pactada en la presente cláusula, **LOS ESTUDIANTES** se harán responsable en forma integral por todos los perjuicios que llegare a causar por el incumplimiento de las obligaciones contraídas dentro de los proyectos en los que intervenga, y por el mal manejo de la información confidencial que le sea confiada o a la que llegare a tener acceso cualquiera que sea su causa.

QUINTA.- Permanencia de las obligaciones: La terminación del contrato de prestación de servicios entre **LA UNIVERSIDAD** y **LOS ESTUDIANTES** no lo exonera del cumplimiento de las obligaciones previstas en este acuerdo, las cuales subsisten durante el tiempo de su permanencia en la universidad. En caso de incumplimiento de alguna de estas obligaciones dentro de dicho período, **LA UNIVERSIDAD** se reserva el derecho de reclamar, judicial o extrajudicialmente, la pena y la indemnización plena de los perjuicios causados.

SEXTA.- Vigencia: El presente acuerdo regirá durante la vigencia de los proyectos y cinco (5) años más contados a partir de su terminación, cualquiera que sea la causa que dé lugar a ella.

SÉPTIMA.- Mérito Ejecutivo: El presente documento junto con la prueba sumaria del incumplimiento por parte de **LOS ESTUDIANTES** prestará mérito ejecutivo para el cobro de las sanciones pecuniarias que en virtud de él se establecen. Bastará también para proceder al cobro ejecutivo, el original del presente acuerdo acompañado de copia auténtica de la carta donde la parte cumplida manifieste a la parte incumplida las causas y motivos de su incumplimiento.

OCTAVA.- Diferencias: Toda controversia o diferencia que surja entre las partes con ocasión de la interpretación, ejecución, modificación, suspensión, terminación o incumplimiento de este acuerdo se resolverá mediante conciliación. De no llegarse a un acuerdo conciliatorio, las diferencias se solucionarán por vía judicial.
La conciliación estará sujeta a las siguientes reglas:

a) La conciliación se llevará a cabo en el Centro de Arbitraje y Conciliación de la Cámara de Comercio de Cali.
b) El conciliador será escogido de común acuerdo de la lista de conciliadores del Centro de Arbitraje y Conciliación de la Cámara de Comercio de Cali.
c) La conciliación deberá intentarse dentro de los tres meses siguientes a la presentación en debida forma de la solicitud de conciliación.
d) Los gastos generados a partir del procedimiento de conciliación serán asumidos por partes iguales entre **LOS ESTUDIANTES** y **LA UNIVERSIDAD**.

NOVENA.- Notificaciones: Para los efectos a que haya lugar en el desarrollo del presente contrato, las partes recibirán notificaciones en las siguientes direcciones:

- **LA UNIVERSIDAD** en la calle 18 N.º 118-250, vía Pance de la ciudad de Cali (Valle).
- **LOS ESTUDIANTES**, en la:

Estudiante:	**Dirección:**	**Correo:**
Luis Felipe Martínez	Transversal 2ª #1C-140	Lufema2003@gmail.com
Sebastián Hernández Triana	Carrera 13 #1-60	sebasht2003@gmail.com

En constancia de todo lo anterior, se firma en dos originales, en la ciudad de Santiago de Cali, el día 22 del mes de marzo.

LA UNIVERSIDAD

VICENTE DURAN CASAS, S.J.
Rector Seccional Cali

EL ESTUDIANTE

Nombre: Luis Felipe Martínez
C.C. 1007619986

EL ESTUDIANTE

Nombre: Sebastián Hernández Triana
C.C. 1006110770