

PONTIFICIA UNIVERSIDAD CATÓLICA DEL ECUADOR FACULTAD DE HÁBITAT, INFRAESTRUCTURA Y CREATIVIDAD

CARRERA: Ingeniería en sistemas de información

Trabajo de titulación como requisito previo para la obtención del título de Ingeniera en Sistemas

PERCEPCIÓN DEL USUARIO EN GOOGLE PLAY: UN ESTUDIO COMPARATIVO ENTRE UN MODELO DE LENGUAJE EXTENSO (LLM) Y SVM PARA EL ANÁLISIS DE SENTIMIENTOS DE UNA APLICACIÓN BANCARIA

Autor: Daniela Abigail Arteaga Carbajal

Tutora: Mgtr. Suyana Fabiola Arcos Villagómez

Quito, Julio 2026

DEDICATORIA

A Sandra y Gerardo, quienes me han guiado en cada paso de mi vida y han acompañado cada uno de mis sueños, incluso cada uno de ellos que alguna vez se vieron imposibles. A mis padres, por ser mi ejemplo de esfuerzo y brindarme su apoyo, ese en el que siempre puedo confiar.

A mis hermanos, Daniel y Paula, por su apoyo incondicional y sus palabras de aliento que resuenan en mí durante los momentos más desafiantes.

Finalmente, a cada error que he cometido, porque gracias a ellos me he convertido en la persona que soy hoy con todos mis defectos, errores y sabiduría.

RESUMEN

Esta investigación compara dos enfoques de análisis de sentimientos aplicados a reseñas de la aplicación Pichincha Banca Móvil publicadas en Google Play: un modelo de lenguaje extenso (Gemini vía API) y un modelo tradicional basado en Support Vector Machines (SVM con LinearSVC). El objetivo es determinar las ventajas, limitaciones y condiciones de uso de cada enfoque en el contexto de la banca digital ecuatoriana.

Con ese propósito, se recopilaron 60.923 reseñas en español mediante la librería google-play-scraper. A continuación, el texto fue preprocesado con un pipeline que incluye normalización, eliminación de duplicados y preservación de términos de negación. El etiquetado automático se realizó con PySentimiento, con una regla de respaldo numérica para reseñas de tres palabras o menos. Este corpus etiquetado sirvió como base para entrenar el modelo SVM con vectorización TF-IDF de 2.500 dimensiones y ajuste de pesos por clase. Por otra parte, Gemini fue implementado mediante prompt engineering con reglas explícitas orientadas al dominio bancario, sin necesidad de datos de entrenamiento. La evaluación comparativa se realizó sobre un Ground Truth de 300 reseñas etiquetadas manualmente, con distribución de 147 positivas, 96 negativas y 57 neutras.

ÍNDICE

DEDICATORIA i

RESUMEN iii

CAPÍTULO I – DEFINICIÓN DEL PROYECTO DE TITULACIÓN 1

1.1. TEMA 1

1.2. JUSTIFICACIÓN 1

1.3. PLANTEAMIENTO DEL PROBLEMA 4

1.4. OBJETIVOS 4

1.4.1. Objetivo General 5

1.4.2. Objetivos Específicos 5

1.5. ALCANCE 6

1.6. MARCO REFERENCIAL 8

CAPÍTULO II: MARCO TEÓRICO 10

2.1. Análisis de sentimientos y su aplicación en entornos digitales 10

2.2. Google Play como entorno de opinión pública sobre aplicaciones móviles 12

2.3. Procesamiento de lenguaje natural (NLP) aplicado al análisis de sentimientos 14

2.4. Sarcasmo, ironía, modismos y ambigüedad lingüística en el análisis de sentimientos 16

2.5. Modelos para la clasificación de sentimientos 18

2.5.1. Modelos de lenguaje extenso: Gemini 19

2.5.2. Modelos tradicionales: Support Vector Machines (SVM) 20

2.5.3. PySentimiento como herramienta de etiquetado automático en español 21

2.6. Comparación de enfoques de análisis de sentimientos 22

2.7. Toma de decisiones basada en datos y monitoreo de la percepción del usuario 24

CAPÍTULO III: DESARROLLO DE LA PROPUESTA 26

3.1. Metodología de investigación 26

3.2. Recopilación y almacenamiento de reseñas públicas de Google Play sobre Pichincha Banca Móvil 27
3.3. Preprocesamiento, depuración y etiquetado automático para el entrenamiento del modelo SVM 28
3.4. Diseño de prompts e implementación del modelo Gemini vía API 31
3.4.1. Diseño del prompt 32
3.5. Entrenamiento e implementación del modelo SVM 34
3.6. Protocolo de etiquetado manual y construcción del Ground Truth 36
3.7. Métricas y criterios de comparación entre Gemini y SVM 38
3.8. Análisis comparativo de casos complejos: sarcasmo, ironía y modismos 41
3.9. Diseño e integración del dashboard interactivo 42
CAPÍTULO IV: ANÁLISIS Y COMPARACIÓN DE RESULTADOS 45
4.1. Descripción del corpus de evaluación 45
4.2. Resultados de clasificación: Gemini API 46
4.3. Resultados de clasificación: SVM con LinearSVC 48
4.4. Comparación de métricas operativas: latencia y costo 50
4.5. Análisis de casos complejos: clase neutra y lenguaje ambiguo 52
4.6. Síntesis comparativa y recomendaciones 53
4.7. Dashboard interactivo de visualización de resultados 55
4.8. Conclusiones 58
4.9. Recomendaciones 60
REFERENCIAS BIBLIOGRÁFICAS 63
ANEXOS 66
ANEXO A 66
ANEXO B 71

ANEXO C 73

ÍNDICE DE TABLAS

Tabla 1 37

Tabla 2 39

Tabla 3 40

Tabla 4 43

Tabla 5 46

Tabla 6 46

Tabla 7 47

Tabla 8 48

Tabla 9 49

Tabla 10 50

Tabla 11 53

ÍNDICE DE FIGURAS

Figura 1 55

CAPÍTULO I – DEFINICIÓN DEL PROYECTO DE TITULACIÓN

El presente capítulo introduce la base conceptual y metodológica de la investigación enfocada en el análisis comparativo de dos enfoques de análisis de sentimientos aplicados a reseñas de la aplicación Pichincha Banca Móvil en Google Play. En este contexto, se definen el tema, la justificación, el planteamiento del problema, los objetivos, el alcance y el marco referencial que sustentan la investigación. Finalmente, se establecen las bases que demuestran la relevancia del estudio dentro del procesamiento de lenguaje natural y la evaluación de modelos tradicionales y basados en inteligencia artificial generativa.

1.1. Tema

Análisis comparativo de un enfoque tradicional y uno basado en IA generativa aplicados al análisis de sentimientos en reseñas de Pichincha Banca Móvil publicadas en Google Play, mediante el uso de Gemini vía API y un modelo tradicional basado en Support Vector Machines (SVM), para la proyección de dashboards analíticos.

1.2. Justificación

En el estudio de la conducta humana en ecosistemas digitales, el análisis de sentimientos ha tomado suma importancia como una herramienta de interpretación de interacciones en línea. En la actualidad, las valoraciones y reseñas publicadas acerca de aplicaciones móviles resultan una fuente directa de opinión de los usuarios sobre la calidad de un servicio, y su tratamiento automático se ha convertido en una necesidad estratégica para las organizaciones que buscan comprender la percepción de su audiencia a escala. Gupta, Ranjan, y Naik, en 2024 señalan que el análisis de sentimientos ha evolucionado desde métodos basados en reglas hacia técnicas avanzadas de aprendizaje automático y modelos de lenguaje grandes, consolidándose como un campo crítico dentro del procesamiento de lenguaje natural. Esta evolución refleja la profunda transformación metodológica que experimenta la disciplina y la

vigencia del debate académico sobre las ventajas comparativas de cada enfoque frente a corpus de texto generado por usuarios en contextos digitales.

Desde esta perspectiva, plataformas como Google Play constituyen un repositorio de experiencias propias de cada usuario relacionadas con el funcionamiento, la facilidad de uso, el tiempo de respuesta y los errores técnicos de las aplicaciones. La aplicación móvil del Banco Pichincha representa un caso de estudio relevante y significativo, ya que genera una gran cantidad de comentarios vinculados a sus servicios y a la experiencia general de la banca digital. Samanmali y Rupasingha (2024) demostraron empíricamente que la calificación numérica asignada por un usuario no siempre refleja el sentimiento expresado en el texto de la reseña, lo que subraya la necesidad de un análisis textual profundo capaz de capturar los matices de la experiencia del usuario más allá de la puntuación numérica. Este hallazgo cobra especial relevancia en el contexto bancario ecuatoriano, donde la confianza del usuario en la institución financiera puede verse afectada por la gestión que esta realice de su reputación digital.

Por otro lado, los modelos de lenguaje extenso se han vuelto populares por su capacidad de comprensión contextual, mientras que los modelos tradicionales son valorados por su menor costo, mayor control en el proceso de entrenamiento y facilidad de ajuste a dominios específicos. En la presente investigación se emplea Gemini como modelo de lenguaje extenso, al ofrecer una interfaz de programación de aplicaciones accesible y un desarrollo factible dentro del alcance del proyecto, además de una comprensión contextual profunda que resulta especialmente útil para clasificar reseñas que incluyen sarcasmo, ironía o modismos regionales. Araya, Mora, Villegas y Casas (2026) confirman que la aplicación de técnicas avanzadas de diseño de prompts mejora significativamente la detección de ironía y sarcasmo en tareas de clasificación de sentimientos con modelos como Gemini, lo cual justifica su selección frente a otras alternativas disponibles. Por su parte, el modelo tradicional seleccionado es Support

Vector Machines (SVM), una técnica ampliamente consolidada en la literatura por su buen desempeño en tareas de clasificación textual y análisis de sentimientos.

La presente investigación propone comparar dos enfoques tecnológicos divergentes aplicados al análisis de sentimientos, una tendencia que cobra creciente relevancia académica e institucional. Al hacerlo, es posible identificar ventajas, limitaciones y condiciones óptimas de uso de cada alternativa en el contexto de una empresa real. De manera específica, se analiza el comportamiento de ambos enfoques frente a reseñas con sarcasmo, ironía, modismos locales y expresiones coloquiales ecuatorianas, al tratarse de uno de los principales desafíos del análisis de sentimientos en lenguaje natural. Alam et al. (2025) documentan en su revisión sistemática que el lenguaje figurativo sigue representando una fuente sistemática de error de clasificación en los sistemas de análisis de sentimientos que operan en múltiples idiomas, lo que refuerza la necesidad de un estudio que aborde estas complejidades en el corpus específico de la banca digital ecuatoriana.

Asimismo, en el contexto bancario, la eficiencia con la que se realiza el análisis es un factor decisivo a la hora de seleccionar un enfoque. En 2025, Alharbi, Alotaibi, Almuqbil y Alshobaili evidenciaron que la elección del modelo impacta directamente en métricas de latencia, costo operativo y precisión de clasificación en entornos de banca digital, factores que deben ser considerados integralmente antes de adoptar una solución tecnológica a escala institucional. Por esta razón, la investigación busca no solo contrastar la calidad de clasificación de ambos modelos, sino también comparar su latencia de respuesta y su costo estimado de procesamiento, con el fin de brindar un análisis completo y de utilidad práctica para la toma de decisiones. Finalmente, la investigación genera la proyección de un dashboard analítico interactivo que facilita la identificación de tendencias, sentimientos predominantes y cambios en la opinión de los usuarios de Pichincha Banca Móvil a lo largo del tiempo, convirtiendo los

hallazgos técnicos en información accionable para los responsables de producto de la institución.

1.3. Planteamiento del problema

Actualmente, las tiendas oficiales de aplicaciones son la fuente principal de opiniones de usuarios de una aplicación móvil en concreto. De esta forma, las reseñas publicadas son una representación directa de la satisfacción de un cliente, al expresar inconformidad, problemas técnicos y recomendaciones relacionados al servicio ofrecido. Así, las reseñas de los usuarios suponen una fuente de información valiosa para una empresa como el Banco Pichincha, que puede verse afectada por una reputación negativa. Las empresas requieren métricas dinámicas que representen su impacto en sus usuarios y sean una herramienta de apoyo en la toma de decisiones.

Sin embargo, analizar estos grandes volúmenes de información es una tarea complicada, por lo que se utilizan modelos LLM o tradicionales para el análisis de sentimientos. La elección del modelo a utilizarse para un análisis de sentimientos resulta confusa, ya que la precisión, costo o rapidez de los resultados varía entre los modelos LLM y los tradicionales. Por lo tanto, resulta necesario un análisis comparativo que demuestre cuál de los dos enfoques es el adecuado para el estudio de las reseñas publicadas en Google Play acerca de una empresa.

1.4. Objetivos

Los objetivos de la presente investigación establecen el propósito general del estudio y delimitan las actividades necesarias para alcanzar los resultados propuestos. Estos objetivos orientan el desarrollo de la investigación al definir las principales etapas del proceso, que comprenden la recopilación de las reseñas, su preparación para el análisis, la implementación

de los modelos y la comparación de sus resultados. A continuación, se presenta el objetivo general y, posteriormente, los objetivos específicos.

1.4.1. Objetivo General

Realizar un análisis comparativo de herramientas utilizadas en un análisis de sentimientos aplicado a reseñas de Pichincha Banca Móvil publicadas en Google Play, mediante la implementación de un modelo LLM vía API con Gemini y modelo tradicional basado en Support Vector Machines (SVM), con el fin de determinar los límites y fortalezas de cada enfoque.

1.4.2. Objetivos Específicos

- Recopilar y almacenar un volumen significativo de reseñas de la aplicación móvil Pichincha Banca Móvil publicadas en Google Play, con el propósito de realizar un análisis de sentimientos verdaderamente representativo.
- Preprocesar las reseñas almacenadas, con el propósito de garantizar datos confiables, estructurados y normalizados que faciliten su tratamiento y análisis posterior.
- Diseñar e implementar una estrategia de prompt engineering para Gemini vía API, orientada a clasificar las reseñas según polaridad y, de ser pertinente, según sentimientos específicos.
- Entrenar e implementar un modelo tradicional de análisis de sentimientos basado en Support Vector Machines (SVM), con el fin de clasificar las reseñas de acuerdo con su sentimiento.
- Extraer una muestra aleatoria de reseñas y etiquetarla manualmente, con el fin de construir un marco referencial para la comparación de los modelos.
- Realizar una comparación de los resultados del análisis de sentimientos realizado con Gemini y SVM mediante Accuracy, Precision, Recall, F1-Score, costo estimado de

procesamiento, latencia de respuesta e interpretabilidad de los resultados, referentemente a un conjunto de reseñas etiquetadas manualmente.

- Diseñar un dashboard interactivo con los resultados del análisis de sentimientos, que permita la aplicación de filtros y métricas para visualizar los datos por sentimiento periodo y nivel de satisfacción.

1.5. Alcance

El presente trabajo de investigación está centrado en el análisis comparativo de dos enfoques que se aplican en el análisis de sentimientos, con el propósito de definir los usos, limitaciones y la viabilidad de cada enfoque. Para el análisis de sentimientos se toma como objeto de estudio a reseñas publicadas en Google Play referentes a Pichincha Banca Móvil, una aplicación móvil ofertada por la institución financiera Banco Pichincha. Con este propósito, se contempla la recolección y almacenamiento de las reseñas a través técnicas de extracción de reseñas públicas de Google Play. A partir de esta colección, se desarrolla una tarea de preprocesamiento y depuración de datos que, como resultado, permita trabajar con datos estructurados, normalizados y confiables.

Del mismo modo, la investigación comprende la implementación de un modelo LLM, específicamente Gemini a través de una API, y de un modelo tradicional basado en Support Vector Machines (SVM); ambos enfoques aplicados al análisis de sentimientos. Las dos orientaciones son empleadas en el mismo conjunto de datos, para obtener una comparación real del desempeño de cada modelo; con este propósito se evaluan métricas de clasificación como accuracy, precision, recall y F1-score.

Los enfoques comparados en este proyecto requieren de una preparación de datos diferente, por lo que se aplican las técnicas necesarias para cada uno. Por su parte, el modelo

SVM exige un pre procesamiento de datos preciso, orientado a la limpieza, normalización y transformación del texto de cada reseña; para asegurar datos aptos para el aprendizaje automático. En cambio, el modelo Gemini no necesita una preparación de datos exhaustiva, ya que clasifica las reseñas, en lenguaje natural, mediante una estrategia de prompt engineering. Adicionalmente, la investigación aplica una muestra aleatoria de reseñas, clasificada manualmente, como marco de referencia para el contraste de la confiabilidad de Gemini y SVM. Así, la investigación busca identificar cuál de las alternativas resulta más adecuada para el análisis de sentimientos en este contexto, considerando aspectos técnicos y prácticos.

El alcance del trabajo incluye el diseño y proyección de un dashboard interactivo que permite la visualización de los resultados obtenidos del análisis de sentimientos. Así, el dashboard presenta de manera clara y comprensible indicadores relevantes, tales como la distribución de sentimientos, tendencias temporales y niveles de satisfacción identificados en las reseñas de Google Play. Como resultado, se convierte el análisis en información visual que facilita la interpretación y posible apoyo a la toma de decisiones.

Sin embargo, este proyecto presenta algunas limitaciones, ya que el estudio se centra únicamente en opiniones públicas relacionadas a la Pichincha Banca Móvil publicadas en Google Play; por lo que los resultados no podrán aplicarse directamente a otras empresas o plataformas. Es importante aclarar que el lenguaje utilizado en reseñas publicadas en plataformas digitales suele incluir ironía, sarcasmo, abreviaturas, errores ortográficos y expresiones coloquiales, lo que puede afectar la precisión de los modelos empleados.

Por otra parte, la presente investigación no contempla el entrenamiento de modelos LLM desde cero, ni el ajuste profundo de arquitecturas avanzadas. El enfoque está basado en el consumo de Gemini vía API y en la implementación de un modelo tradicional de clasificación. Del mismo modo, tampoco incluye el estudio de variables externas como perfil demográfico de usuarios, ubicación geográfica o intención de descarga. Del mismo modo, la

proyección de un dashboard interactivo, no tiene como objetivo desarrollar una plataforma en producción, sino una visualización funcional para presentar los resultados del análisis de sentimientos.

Finalmente, la investigación no predice comportamientos futuros de los usuarios ni medir la reputación digital exacta de Pichincha Banca Móvil, sino comparar dos enfoques de análisis de sentimientos en un conjunto específico de datos. El estudio se limita a un caso aplicado, con fines comparativos y académicos, para evidenciar las ventajas, limitaciones y viabilidad de ambas alternativas.

1.6. Marco referencial

El presente trabajo de investigación se sustenta en conceptos relacionados al análisis de sentimientos, el procesamiento de lenguaje natural, los modelos de aprendizaje automático y los modelos de lenguaje extensos. Estos fundamentos permiten comprender el enfoque teórico del estudio y justifican la comparación entre un modelo tradicional basado en Support Vector Machines (SVM) y un modelo LLM como Gemini, aplicados al análisis de reseñas de usuarios en Google Play.

El análisis de sentimientos es una técnica del procesamiento de lenguaje natural orientada a identificar y clasificar opiniones expresadas en texto, generalmente en categorías como positivo, negativo o neutral. Está técnica brinda la posibilidad de transformar grandes volúmenes de información no estructurada en conocimiento útil para organizaciones que requieren comprender la percepción de sus usuarios. Es por esto que, en la actualidad, las reseñas digitales constituyen una fuente valiosa para evaluar satisfacción, detectar problemas recurrentes y conocer experiencias de uso.

Por su parte, el procesamiento de lenguaje natural (NLP) proporciona las bases técnicas para que los sistemas computacionales analicen lenguaje humano. Dentro de este campo, tareas como la limpieza de texto, normalización y representación del contenido permiten preparar los

datos para su clasificación posterior. Estas etapas son especialmente relevantes cuando se trabaja con reseñas en plataformas digitales, ya que suelen contener abreviaturas, errores ortográficos, lenguaje informal, ironía o sarcasmo.

En cuanto a los enfoques tecnológicos, los modelos tradicionales como SVM han sido ampliamente utilizados en clasificación de texto por su eficiencia, menor costo y facilidad de entrenamiento sobre datos etiquetados. En contraste, los modelos de lenguaje extensos, como Gemini, ofrecen una mayor capacidad de comprensión contextual y semántica, lo que puede resultar ventajoso frente a expresiones complejas del lenguaje. Sin embargo, su uso depende del diseño adecuado de instrucciones o prompts, así como de factores prácticos como costo y tiempo de respuesta.

Finalmente, la visualización de resultados mediante dashboards analíticos permite representar de forma clara los hallazgos del análisis de sentimientos, facilitando la interpretación de tendencias, niveles de satisfacción y distribución de opiniones. De esta manera, el marco referencial integra los conceptos necesarios para sustentar una investigación orientada no solo a comparar dos enfoques de análisis, sino también a transformar los resultados en información útil para la toma de decisiones.

CAPÍTULO II: MARCO TEÓRICO

El Capítulo II presenta los fundamentos teóricos que sustentan la presente investigación, abordando los principales conceptos, técnicas y enfoques relacionados con el análisis de sentimientos en entornos digitales. Se desarrolla la base conceptual del procesamiento de lenguaje natural, los modelos tradicionales y los modelos de lenguaje extenso, así como las particularidades del tratamiento de reseñas en plataformas como Google Play. Además, se incluyen aspectos clave como el manejo del lenguaje informal, el sarcasmo y la ironía, junto con las herramientas y metodologías utilizadas para la clasificación y comparación de sentimientos.

2.1. Análisis de sentimientos y su aplicación en entornos digitales

El análisis de sentimientos es una técnica, también llamada minería de opiniones, altamente utilizada en la actualidad. Se trata de una subdisciplina del procesamiento de lenguaje natural cuyo propósito principal es identificar, extraer y cuantificar las orientaciones subjetivas presentes en fragmentos de texto. Se debe resaltar que es una tarea compleja, ya que el lenguaje humano es inherentemente ambiguo, contextual y culturalmente situado. Por lo tanto, este procesamiento encierra una complejidad técnica y lingüística considerable.

El análisis de sentimientos asigna a un texto una etiqueta de polaridad, generalmente positiva, negativa o neutral. Aunque existen versiones que permiten identificar emociones específicas, como alegría, frustración o sorpresa, y atribuirlas a aspectos concretos de un producto o servicio. Gupta et al. (2024) explican que este campo ha pasado desde métodos basados en reglas y diccionarios de opinión hacia técnicas de aprendizaje automático. En la actualidad, se utilizan modelos de lenguaje extenso que procesan el texto como una secuencia semántica completa.

Desde una perspectiva aplicada, el análisis de sentimientos ha adquirido el estatus de herramienta estratégica en la gestión organizacional, especialmente en sectores donde la percepción del usuario influye directamente sobre la competitividad institucional. De esta forma, las organizaciones utilizan esta técnica para monitorear la reputación de marca, detectar problemas latentes en productos o servicios, anticipar tendencias de insatisfacción y comprender las necesidades de sus clientes en tiempo real. Alam et al. (2025) evidencian que la integración del análisis de sentimientos con técnicas de inteligencia artificial ha permitido que las empresas procesen millones de opiniones automáticamente, una tarea que manualmente resultaba imposible.

El análisis de sentimientos tiene especial impacto en el sector financiero, donde la percepción digital del usuario puede determinar la fidelización del cliente y la reputación institucional. En el caso de la banca digital ecuatoriana, la capacidad de procesar automáticamente las reseñas publicadas, en plataformas como Google Play, representa una ventaja competitiva de primer orden para instituciones que buscan mejorar sus servicios de forma continua y basada en evidencia.

Los niveles de análisis de sentimientos tienen distintas granularidades de estudio que responden a distintos objetivos analíticos. El nivel de documento clasifica la polaridad del texto como un todo y abstraer los detalles específicos que varían dentro de un fragmento; mientras que, el nivel de oración analiza cada enunciado de forma independiente para capturar variaciones de sentimiento dentro de un único texto. Por su parte, el nivel de aspecto, identifica la polaridad expresada hacia atributos específicos de una entidad.

La presente investigación aplica el nivel de documento, ya que clasifica cada reseña completa (enunciado) en una de las tres categorías de polaridad, lo cual es coherente con el volumen del corpus analizado y con los objetivos comparativos del estudio. Esta decisión no

implica una limitación metodológica, sino una delimitación justificada por la naturaleza del corpus y los recursos disponibles.

La consolidación del análisis de sentimientos como disciplina científica ha sido acompañada por el desarrollo de recursos lingüísticos especializados, como: diccionarios de opinión, corpus etiquetados y métricas de evaluación estandarizadas. Estos recursos han facilitado la comparación rigurosa entre distintos enfoques. Sin embargo, el análisis de sentimientos no es solo un problema técnico de clasificación, sino también un desafío lingüístico y cultural.

La mayoría fueron diseñados originalmente para el idioma anglosajón, lo que ha generado una brecha importante en el tratamiento del español y, más específicamente, de las variedades regionales del español latinoamericano. Esta brecha es relevante para la presente investigación, que trabaja con reseñas escritas en español ecuatoriano informal. Es por esto que se propone la adopción de herramientas como PySentimiento, diseñada específicamente para el análisis de sentimientos en español, para el desarrollo.

2.2. Google Play como entorno de opinión pública sobre aplicaciones móviles

Google Play Store es la plataforma de distribución de aplicaciones móviles más utilizada en el mundo para dispositivos con sistema operativo Android. Cuenta con más de tres millones de aplicaciones disponibles y miles de millones de usuarios activos distribuidos en más de ciento noventa países. Su alcance global y la magnitud del contenido generado por usuarios la convierten en una fuente de datos de inmenso valor para el estudio de la percepción del usuario en el ecosistema de las aplicaciones móviles.

Las reseñas disponibles en Google Play Store constituyen un ecosistema único de opinión pública: los usuarios expresan, espontáneamente y sin mediación institucional, sus experiencias relacionadas con la funcionalidad, la usabilidad, la estabilidad técnica y el servicio

de atención al cliente de las aplicaciones que utilizan. La naturaleza pública y verificada de estas reseñas las convierte en una fuente de datos privilegiada para el análisis de sentimientos; en donde, cada entrada incluye el texto de la opinión, la calificación numérica asignada por el usuario, la fecha de publicación y la versión de la aplicación en el momento de la reseña. Estas constituyen variables que enriquecen el corpus y posibilitan análisis temporales y comparativos de gran utilidad práctica.

Samanmali y Rupasingha (2024) realizaron un estudio sobre reseñas de quince aplicaciones distintas disponibles en Google Play, en donde utilizaron algoritmos de aprendizaje automático para el análisis de sentimientos. Sus resultados evidenciaron que "la calificación con estrellas proporcionada por los usuarios no siempre coincide con la polaridad del sentimiento expresada en la reseña textual, lo que revela una discrepancia sistemática entre la retroalimentación cuantitativa y cualitativa de los usuarios." (p. 84431). Este hallazgo cuestiona la validez de emplear la calificación numérica como único indicador de satisfacción del usuario: es necesario un análisis textual independiente. Por ejemplo, un usuario puede asignar tres estrellas a una aplicación bancaria mientras su reseña expresa una queja específica sobre un error de transacción que le causó pérdidas económicas. En este caso, un sistema de monitoreo basado únicamente en las calificaciones numéricas no podría detectar correctamente el descontento causado al cliente.

Por otro lado, en el ámbito de la banca digital, las reseñas en Google Play se han consolidado como un canal directo de retroalimentación entre los usuarios y las instituciones financieras. Se ha percibido un impacto directo de las reseñas sobre la reputación institucional y la calidad del servicio. Alharbi et al. (2025) analizaron reseñas de aplicaciones de banca digital en Google Play, encontrando que los usuarios expresan sentimientos negativos relacionados principalmente con fallas en la autenticación, errores de transacción y lentitud del sistema. Estos aspectos impactan directamente en la confianza hacia la institución y, en muchos

casos, motivan la deserción hacia competidores con mejores valoraciones digitales. Por su parte, en un mercado financiero ecuatoriano cada vez más competitivo, la percepción digital del usuario puede determinar la retención de clientes y la imagen pública de la institución, en este caso del Banco Pichincha.

En la presente investigación, la extracción de reseñas de Google Play se realiza mediante herramientas como google-play-scraper, que permiten obtener datos estructurados de manera automática; resulta importante aclarar que se utilizan los datos públicamente disponibles en la plataforma. La disponibilidad de esta información de manera pública y estructurada, junto con el volumen significativo de reseñas disponibles para la aplicación Pichincha Banca Móvil, hace de Google Play una fuente de datos óptima para los propósitos de la investigación.

2.3. Procesamiento de lenguaje natural (NLP) aplicado al análisis de sentimientos

El procesamiento de lenguaje natural es la rama de la inteligencia artificial que estudia la interacción entre los sistemas computacionales y el lenguaje humano. Este campo tiene el objetivo de conceder a las máquinas la capacidad de comprender, interpretar y generar texto de manera significativa. En el análisis de sentimientos, el NLP proporciona el conjunto de técnicas que permiten transformar texto no estructurado, tal como se encuentra en las reseñas de usuarios, en representaciones numéricas procesables por modelos de clasificación.

El proceso de transformación aplicado por el NPL no es ligero, ya que el texto presenta características lingüísticas particulares: abreviaturas, errores ortográficos, neologismos, emojis y expresiones coloquiales. Estas particularidades difieren significativamente del lenguaje formal empleado en los corpus sobre los que fueron diseñados los modelos clásicos de NLP. Esta heterogeneidad lingüística es especialmente acentuada en el español latinoamericano,

donde coexisten variaciones regionales, préstamos del inglés y términos de origen indígena que añaden capas adicionales de complejidad al proceso de análisis.

El preprocesamiento del texto es la fase inicial e indispensable de cualquier pipeline de NLP orientado al análisis de sentimientos, y su calidad determina en gran medida el rendimiento del modelo de clasificación posterior. Esta fase comprende un conjunto de operaciones que buscan normalizar el texto y reducir su variabilidad sin eliminar la información semánticamente relevante para la clasificación. Bimantara y Zufria (2024) aplicaron un pipeline de preprocesamiento en un estudio de análisis de sentimientos sobre reseñas de una aplicación de banca móvil en Google Play, concluyendo que "la calidad del preprocesamiento afecta directamente el rendimiento del modelo, especialmente en corpus que contienen abreviaturas, errores ortográficos y vocabulario informal típico de las reseñas en plataformas digitales" (p. 119). Esta observación es especialmente pertinente para el presente estudio, dado que las reseñas de Pichincha Banca Móvil están escritas mayoritariamente en español informal y pueden incluir ecuatorianismos, expresiones de herencia quechua y neologismos digitales que requieren un tratamiento lingüístico específico para no distorsionar el análisis.

Dentro del preprocesamiento, la eliminación de palabras vacías merece una consideración especial en el contexto del análisis de sentimientos. Las palabras vacías son términos de alta frecuencia, pero escaso valor semántico, como artículos, preposiciones y conjunciones, cuya eliminación reduce la dimensionalidad del espacio de características sin comprometer la capacidad clasificadora del modelo. Sin embargo, en español existen palabras de negación, como "no", "nunca" o "jamás", que frecuentemente son clasificadas como palabras vacías en listas estándar pero que portan una carga semántica crítica para el análisis de sentimientos, ya que invierten la polaridad de la expresión a la que preceden. Por esta razón, el pipeline de preprocesamiento implementado en la presente investigación preserva explícitamente los términos de negación al realizar la eliminación de palabras vacías.

Una vez depurado el texto, la representación vectorial mediante Frecuencia de Término–Frecuencia Inversa de Documento (TF-IDF) asigna a cada término un peso proporcional a su frecuencia en el documento y a su rareza en el corpus completo, destacando los términos más discriminativos para la clasificación. Kamath et al. (2025) señalan que la representación TF-IDF con configuración de n-gramas unigrama y bigrama produce resultados consistentemente competitivos en tareas de clasificación de texto de dominio específico, incluso frente a representaciones basadas en embeddings de palabras más complejas, especialmente cuando el corpus de entrenamiento es de tamaño moderado. La elección de la lematización sobre el stemming responde a que la primera preserva la coherencia semántica de los tokens al reducirlos a formas válidas del diccionario, mientras que el stemming puede generar truncaciones que distorsionan el significado del texto original, con implicaciones negativas sobre la precisión del clasificador en idiomas morfológicamente ricos como el español.

2.4. Sarcasmo, ironía, modismos y ambigüedad lingüística en el análisis de sentimientos

Uno de los desafíos más persistentes y documentados en el campo del análisis de sentimientos es el tratamiento de expresiones lingüísticas que no comunican su significado de manera literal, entre las que se destacan el sarcasmo, la ironía, los modismos y las ambigüedades contextuales. Estas expresiones son frecuentes en el lenguaje coloquial digital y pueden inducir errores sistemáticos en los modelos de clasificación, especialmente en aquellos que operan a nivel léxico sin considerar el contexto pragmático del enunciado. Por ejemplo, una reseña que dice "excelente, como siempre la app falla cuando más la necesito" contiene una expresión superficialmente positiva que encubre una intención claramente negativa. En este caso, cualquier sistema de clasificación que no pueda acceder a ese nivel de comprensión pragmática clasificará incorrectamente el sentimiento del texto. Este tipo de error

no es aleatorio sino sistemático: afecta consistentemente a ciertas categorías de expresión y puede sesgar significativamente los resultados del análisis si no se aborda explícitamente.

Por su parte, el sarcasmo es un mecanismo retórico mediante el cual el hablante expresa una opinión contraria a la que literalmente enuncia, generalmente con una intención crítica o despectiva. Qin et al. (2025) propusieron un modelo que integra transformadores BERT con redes neuronales de grafos para detectar sarcasmo en contenido generado por usuarios, logrando mejoras significativas frente a modelos basados únicamente en representaciones superficiales del texto. Los autores destacan que "el sarcasmo en las plataformas digitales se manifiesta con frecuencia como elogios superficialmente positivos con una intención negativa subyacente, lo que lo convierte en una fuente sistemática de ruido para los clasificadores de sentimiento convencionales. " (Qin et al., 2025), lo que convierte su detección en un reto técnico de primera importancia para cualquier sistema de análisis de sentimientos orientado a reseñas de aplicaciones. La ironía, aunque estrechamente relacionada con el sarcasmo, opera a un nivel más amplio de incongruencia entre el significado literal y el intencional, y puede manifestarse incluso sin los marcadores discursivos típicamente asociados al sarcasmo, lo que la hace aún más difícil de detectar automáticamente.

En el estudio realizado por Rani et al. (2025) se desarrolló un modelo de detección de sarcasmo basado en redes neuronales recurrentes para textos en inglés, demostrando que el preprocesamiento especializado que incluye el uso de diccionarios de jerga y el tratamiento de emojis mejora sustancialmente la capacidad del modelo para identificar expresiones no literales en corpora de texto informal. Aunque su estudio se realizó en inglés, sus conclusiones son extrapolables al español ecuatoriano, donde los usuarios también recurren a emojis, signos de exclamación, mayúsculas enfatícas y términos coloquiales culturalmente específicos para expresar emociones, a veces de manera no literal. Los modismos locales ecuatorianos añaden una dimensión adicional de complejidad, ya que pueden no estar presentes en los corpus de

entrenamiento de la mayoría de los modelos de NLP disponibles, generando una brecha semántica entre el significado pretendido por el usuario y el interpretado por el sistema.

En 2026, Araya et al. demostraron que la aplicación de técnicas avanzadas de diseño de prompts, incluyendo instrucciones de cadena de pensamiento y ejemplos de pocos disparos, mejora sustancialmente la capacidad de modelos como Gemini para detectar ironía y sarcasmo en tareas de clasificación de sentimientos. Esta capacidad diferencial de los modelos de lenguaje extenso frente a los modelos tradicionales constituye uno de los ejes de comparación más relevantes de la presente investigación, ya que permite evaluar empíricamente si la capacidad de comprensión pragmática de Gemini se traduce en una ventaja medible sobre SVM al clasificar reseñas con expresión figurativa. Los modelos tradicionales basados en TF-IDF y SVM no modelan explícitamente el contexto pragmático del enunciado, lo que los hace especialmente vulnerables a errores de clasificación frente a expresiones irónicas o sarcásticas, aunque su rendimiento puede ser competitivo en corpus donde este tipo de expresiones es minoritario.

2.5. Modelos para la clasificación de sentimientos

La selección del modelo de clasificación constituye un aspecto determinante en el desempeño de un sistema de análisis de sentimientos, ya que cada enfoque presenta características particulares en términos de precisión, requerimientos computacionales, interpretabilidad y forma de procesamiento del lenguaje natural. En la presente investigación se consideran dos enfoques: un modelo de lenguaje extenso (LLM), representado por Gemini, y un modelo tradicional de aprendizaje automático, representado por Support Vector Machines (SVM). Asimismo, se implementa PySentimiento como herramienta de apoyo para el etiquetado automático del corpus, debido a su especialización en el análisis de sentimientos en idioma español.

2.5.1. Modelos de lenguaje extenso: Gemini

Los modelos de lenguaje extenso son arquitecturas de redes neuronales profundas entrenadas sobre corpus masivos de texto mediante objetivos de predicción de tokens. La escala de estas arquitecturas les confiere capacidades emergentes para comprender el contexto semántico y pragmático del lenguaje a un nivel cualitativamente superior al de los modelos tradicionales. A diferencia de los clasificadores clásicos que requieren datos etiquetados para aprender a realizar una tarea específica, los modelos de lenguaje extenso pueden ser dirigidos hacia tareas de clasificación mediante instrucciones en lenguaje natural, sin necesidad de un proceso formal de entrenamiento sobre el dominio objetivo.

Gemini es la familia de LLM desarrollada por Google DeepMind, está diseñada con capacidades multimodales desde su concepción, se encuentra disponible a través de la interfaz de programación de aplicaciones de Google AI. En su variante gemini-2.5-flash, el modelo ofrece un equilibrio óptimo entre rendimiento y eficiencia de costos para tareas de clasificación de texto, lo que lo convierte en una opción viable para la investigación. Araya et al. (2026) describen que el variante flash del modelo Gemini, configurada mediante un prompt que define el rol del modelo y el esquema de clasificación, produce resultados consistentes y confiables en tareas de clasificación de polaridad de sentimiento; especialmente, cuando se trata de opiniones expresadas de forma ambigua o figurativa. La ausencia de necesidad de preprocesamiento exhaustivo del texto es una de las principales ventajas de este enfoque, ya que el modelo procesa el lenguaje natural directamente y accede al contexto pragmático de cada reseña. De esta forma, no es necesaria ninguna transformación previa del texto, que podría causar la pérdida de información relevante para la clasificación.

El uso de modelos de lenguaje extenso a través de interfaces de programación de aplicaciones introduce consideraciones operativas que deben ser evaluadas en función del volumen de datos procesados. A diferencia de los modelos tradicionales, donde el costo de

inferencia es marginal una vez entrenado el modelo localmente, los modelos de lenguaje extenso cobran por número de tokens procesados. Esto representa una limitación económica significativa cuando el corpus es de gran tamaño. Asimismo, la latencia de respuesta está condicionada por la disponibilidad de los servidores del proveedor y por el tamaño del texto enviado. En consecuencia, el procesamiento masivo de reseñas puede ser más lento que con un modelo tradicional ejecutado localmente.

2.5.2. Modelos tradicionales: Support Vector Machines (SVM)

En la presente investigación, se realiza un análisis comparativo entre Gemini y Support Vector Machines. SVM es un algoritmo de aprendizaje automático supervisado que hace una búsqueda del hiperplano óptimo de separación entre clases en un espacio de características de alta dimensionalidad. Su fundamentación matemática, basada en la teoría de la maximización del margen de clasificación, le confiere propiedades de generalización que lo hacen especialmente robusto en contextos donde los datos de entrenamiento son de tamaño moderado y las características son numerosas y dispersas. Esta combinación de propiedades lo convierte en uno de los algoritmos más ampliamente utilizados en la literatura de clasificación de texto, un dominio caracterizado precisamente por la alta dimensionalidad de las representaciones vectoriales y la escasez relativa de datos etiquetados en dominios especializados.

Kamath et al. (2025) compararon SVM con múltiples técnicas de aprendizaje automático para análisis de sentimientos, y concluyeron: "SVM con kernel lineal logra de manera consistente un rendimiento competitivo en tareas de clasificación de texto binaria y multiclase, especialmente cuando el corpus de entrenamiento está bien preprocesado y representado mediante TF-IDF con configuraciones apropiadas de n-gramas" (p. 133). Estos antecedentes confirman la vigencia y pertinencia del modelo SVM para el tipo de tarea

abordada en la presente investigación y justifican su selección como representante del enfoque tradicional en el análisis comparativo.

La principal ventaja del modelo SVM frente a los LLM radica en su transparencia metodológica y en la posibilidad de inspeccionar los pesos asignados a cada característica del modelo entrenado. Esta capacidad de interpretación permite identificar los términos más discriminativos para cada categoría de sentimiento, lo que facilita la formulación de explicaciones comprensibles sobre el comportamiento del clasificador y su uso como herramienta de análisis lingüístico del corpus. En cambio, el proceso de clasificación de un modelo de lenguaje extenso como Gemini es esencialmente ambiguo, ya que el usuario no puede acceder a los mecanismos internos que determinan la clasificación de cada texto. Esta dimensión de interpretabilidad es un factor con suma relevancia práctica en contextos institucionales donde la justificación de las decisiones automatizadas es un requisito de gobernanza.

2.5.3. PySentimiento como herramienta de etiquetado automático en español

PySentimiento es una librería de código abierto para análisis de sentimientos y otras tareas de procesamiento de lenguaje natural social, desarrollada por Juan Manuel Pérez y colaboradores (2024). La librería está construida sobre la arquitectura HuggingFace Transformers e integra modelos pre entrenados de última generación para español e inglés. Entre los modelos se destaca RoBERTuito, un modelo basado en RoBERTa y ajustado específicamente sobre corpus de texto en español extraídos de redes sociales y plataformas digitales de uso cotidiano. Pérez et al. (2024) describen PySentimiento como

> *pysentimiento* ofrece una herramienta multilingüe y fácil de usar para la minería de opiniones en redes sociales. Está construida sobre la librería Transformers de HuggingFace y proporciona una API sencilla para que los investigadores utilicen

> modelos de última generación en análisis de sentimientos y otras tareas de PLN social, con soporte actual para español, inglés, italiano y portugués. (p. 2)

Esta definición resume con precisión las ventajas que hacen de esta herramienta un recurso importante para el etiquetado automático de corpus de texto informal en español. En el contexto hispanohablante, PySentimiento ofrece ventajas diferenciadas frente a herramientas generalistas como VADER o TextBlob, que fueron originalmente diseñadas para el inglés y cuya adaptación al español resulta superficial e incompleta. Esta capacidad de manejar variaciones lexicales regionales es especialmente relevante para el corpus de reseñas de Pichincha Banca Móvil, que está escrito en español ecuatoriano informal con posibles variaciones lexicales regionales no contempladas en los recursos lingüísticos generalistas.

La interfaz de PySentimiento produce para cada texto tres probabilidades de sentimiento correspondientes a las categorías positivo, negativo y neutral, junto con la etiqueta de máxima probabilidad como predicción principal. Esta salida estructurada permite obtener no solo la clasificación de polaridad sino también el nivel de confianza del modelo en su predicción. Por lo tanto, PySentimiento brinda información valiosa para identificar los casos frontera donde la clasificación es menos segura.

El uso de PySentimiento como herramienta de etiquetado automático en la presente investigación responde a la necesidad de construir un conjunto de referencia de gran tamaño para el entrenamiento del modelo SVM y la evaluación comparativa con Gemini, sin incurrir en los costos y tiempos asociados a un etiquetado manual exhaustivo del corpus completo. La elección de esta metodología responde a prácticas consolidadas en la literatura especializada, que reconocen el etiquetado automático mediante modelos preentrenados como una solución eficiente frente al etiquetado manual.

2.6. Comparación de enfoques de análisis de sentimientos

La comparación entre enfoques de lenguaje extenso y modelos tradicionales para el análisis de sentimientos ha sido objeto de creciente interés académico en los últimos años. Esto se debe a la necesidad de determinar las condiciones que hacen a cada alternativa más adecuada en función del tipo de corpus, los recursos disponibles y los requerimientos operativos del contexto de aplicación. Teles y Figueiredo (2025) realizaron un estudio comparativo de modelos de lenguaje extenso frente a modelos clásicos para el análisis de sentimientos en noticias de mercados financieros. En esta investigación concluyeron que los LLM superan a los modelos clásicos en la gran mayoría de los escenarios de prueba, aunque significan un costo computacional significativamente mayor y con una latencia de respuesta que puede limitar su aplicabilidad en entornos de producción de alto volumen. Este hallazgo sugiere que la superioridad de los modelos de lenguaje extenso no es absoluta sino contextualmente dependiente, y que la elección del enfoque más adecuado requiere una evaluación multidimensional que considere tanto la calidad de clasificación como las restricciones operativas del entorno de aplicación.

Kamath et al. (2025) identificaron que, en tareas de clasificación de sentimientos con conjuntos de datos etiquetados de tamaño moderado, los modelos tradicionales como SVM mantienen una competitividad notable frente a arquitecturas más complejas, con la ventaja adicional de requerir recursos computacionales significativamente menores. Esta conclusión es especialmente relevante para contextos institucionales con restricciones presupuestarias, donde el costo por consulta de una interfaz de programación de lenguaje extenso puede representar una limitación operativa significativa a escala. Alharbi et al. (2025), por su parte, encontraron que los modelos de lenguaje extenso presentan ventajas en la comprensión de textos ambiguos, pero desventajas en términos de costo y velocidad de procesamiento masivo, lo que confirma la necesidad de una evaluación empírica contextualizada antes de adoptar cualquier enfoque en un entorno productivo.

Las dimensiones de comparación más frecuentemente empleadas en la literatura incluyen las métricas de calidad de clasificación como accuracy, precision, recall y F1-score, el costo operativo estimado por unidad de texto procesada, la latencia de respuesta y la interpretabilidad de los resultados. La presente investigación adopta este marco de comparación multidimensional para evaluar Gemini y SVM sobre el corpus de reseñas de Pichincha Banca Móvil, con el fin de generar evidencia empírica que oriente la toma de decisiones sobre la adopción de cada enfoque en un contexto bancario latinoamericano específico. Además, contribuye así a llenar un vacío en la literatura que, hasta la fecha, ha abordado este tipo de comparación principalmente en el contexto del inglés y de mercados financieros de economías desarrolladas.

2.7. Toma de decisiones basada en datos y monitoreo de la percepción del usuario

La toma de decisiones basada en datos es el paradigma de gestión organizacional que fundamenta las decisiones estratégicas y operativas en el análisis sistemático de datos empíricos, en contraposición a la intuición o la experiencia no cuantificada. En el sector financiero, este paradigma ha cobrado relevancia crítica ante la necesidad de responder con agilidad a las dinámicas del mercado y a las expectativas cambiantes de los usuarios de servicios digitales. La transformación digital de la banca ha generado flujos masivos de datos sobre el comportamiento y la satisfacción de los clientes, entre los que las reseñas en plataformas digitales ocupan un lugar destacado por su riqueza informativa y su actualización continua. El análisis sistemático de estos datos mediante técnicas de procesamiento de lenguaje natural permite convertir millones de opiniones textuales en indicadores cuantitativos de percepción que pueden ser incorporados directamente en los procesos de toma de decisiones institucional.

Alharbi et al. (2025) señalan que el análisis de sentimientos se ha consolidado como una herramienta clave de apoyo a la toma de decisiones basada en datos en el sector bancario, puesto que proporciona señales tempranas acerca de tendencias de insatisfacción que pueden anticipar problemas de retención de clientes. Esto es especialmente relevante en el contexto de la banca digital ecuatoriana, donde la competencia entre instituciones financieras es creciente y la experiencia del usuario en los canales digitales es un factor determinante de diferenciación. La capacidad de detectar automáticamente un aumento en las reseñas negativas relacionadas con un error específico de la aplicación, por ejemplo, puede permitir al equipo técnico del banco priorizar la corrección del problema.

La visualización de los resultados del análisis de sentimientos mediante dashboards analíticos interactivos es el mecanismo a través del cual los hallazgos del modelo de NLP se traducen en información comprensible y accionable para los tomadores de decisiones que no necesariamente poseen formación técnica en ciencia de datos. Datta et al. (2025) realizaron una revisión sistemática de la literatura sobre visualización de datos en contextos de toma de decisiones asistida por inteligencia artificial, identificando que los dashboards efectivos deben integrar filtros dinámicos, representaciones temporales de tendencias y métricas de resumen que permitan al usuario explorar los datos desde múltiples perspectivas. El diseño del dashboard propuesto en la presente investigación, busca no solo presentar los resultados del análisis sino también facilitar su interpretación y uso por parte de los equipos de producto y gestión del Banco Pichincha en su proceso continuo de mejora de la experiencia del usuario.

CAPÍTULO III: DESARROLLO DE LA PROPUESTA

El Capítulo III presenta el desarrollo metodológico y técnico de la propuesta de investigación, orientada a la implementación y comparación de dos enfoques para el análisis de sentimientos en reseñas de Pichincha Banca Móvil: un modelo tradicional basado en Support Vector Machines (SVM) y un modelo de lenguaje extenso mediante Gemini vía API. Se describe de manera detallada el proceso de recolección, preprocesamiento y etiquetado de datos, así como el diseño de prompts, el entrenamiento de los modelos y la construcción del conjunto de validación. Asimismo, se exponen los criterios de evaluación y las métricas de comparación, considerando tanto el desempeño predictivo como aspectos operativos como costo, latencia e interpretabilidad.

3.1. Metodología de investigación

La presente investigación adopta un enfoque metodológico mixto que articula elementos de la investigación aplicada, descriptiva y comparativa. El estudio busca implementar herramientas tecnológicas concretas para la resolución de un problema real relacionado con la clasificación automatizada de sentimientos en reseñas de una aplicación bancaria, así se genera un artefacto de valor práctico para la institución estudiada. Por otro lado, el análisis busca caracterizar el corpus de reseñas de Pichincha Banca Móvil en términos de su distribución de sentimientos, sus tendencias temporales y sus patrones lingüísticos predominantes, generando así un retrato representativo de la percepción del usuario en la plataforma Google Play. Finalmente, el estudio contrasta sistemáticamente el desempeño de dos enfoques tecnológicos diferenciados sobre el mismo conjunto de datos, lo que garantiza la validez interna de la comparación al utilizar el mismo conjunto de datos.

El diseño de la investigación sigue una lógica secuencial que parte de la recopilación y almacenamiento de datos, continúa con el preprocesamiento y la clasificación automática

mediante PySentimiento, y avanza hacia la implementación del modelo Gemini mediante prompt engineering y el entrenamiento del modelo SVM. La secuencia culmina con la evaluación comparativa de ambos modelos a partir de una muestra estratificada del corpus etiquetado y con el diseño del dashboard interactivo de resultados.

El desarrollo del presente proyecto se apoya de una metodología secuencial que permite mantener la trazabilidad del proceso, desde la extracción de los datos hasta la presentación de los resultados obtenidos. Para empezar, se recopilan las reseñas públicas de Google Play, y se almacenan en un archivo estructurado para su posterior clasificación. A continuación, se preprocesan los datos, de esta forma, se obtienen reseñas con el texto limpio y normalizadas, y se eliminan duplicados. A partir las reseñas depuradas, el análisis se divide en dos enfoques: la clasificación mediante Gemini vía API, apoyada en una estrategia de prompt engineering; y, por otro lado, el entrenamiento y aplicación del modelo SVM, basado en la representación vectorial del texto. Finalmente, los resultados de ambos enfoques se comparan frente a una muestra etiquetada manualmente.

3.2. Recopilación y almacenamiento de reseñas públicas de Google Play sobre Pichincha Banca Móvil

La recopilación de datos constituye la fase inicial del desarrollo de la propuesta y determina en gran medida la representatividad y la validez del análisis posterior. Para obtener las reseñas públicas de la aplicación Pichincha Banca Móvil disponibles en Google Play, se emplea la librería google-play-scraper para Python, que permite extraer de manera automatizada los datos asociados a cada reseña disponible en la plataforma. La información recuperada para cada registro incluye el texto de la opinión expresada por el usuario, la calificación numérica asignada en una escala de uno a cinco estrellas, y la fecha de publicación de la reseña. Además, con el propósito de mantener el enfoque de la investigación, se

delimitaron los campos recolectados, por lo que no se recopiló información como el nombre del usuario ni la versión de la aplicación utilizada. Las variables obtenidas enriquecen el corpus y posibilitan un análisis temporal y de correlación entre la calificación numérica y el sentimiento textual.

El proceso de recopilación se configura para extraer reseñas en idioma español, dado que la aplicación opera primordialmente en el mercado ecuatoriano. Las reseñas extraídas se almacenan en formato CSV bajo el nombre pichincha_google_play_review.csv, conservando todas las variables mencionadas para su uso en las fases de análisis. Bimantara y Zufria (2024) emplearon un procedimiento análogo de extracción de reseñas de banca móvil en Google Play, obteniendo un corpus de más de diez mil registros textuales sobre los cuales aplicaron su pipeline de análisis de sentimientos con resultados sólidos. La extracción de datos públicos en el contexto académico se realiza respetando los términos de uso de la plataforma y las directrices éticas correspondientes, sin procesar datos personales sensibles de los usuarios y garantizando que la información extraída es de acceso público.

3.3. Preprocesamiento, depuración y etiquetado automático para el entrenamiento del modelo SVM

Una vez constituido el corpus de reseñas, el proceso de preprocesamiento y clasificación se desarrolla como una secuencia metodológica integrada que transforma el texto crudo en datos etiquetados y normalizados, aptos para el entrenamiento del modelo SVM y para la evaluación comparativa con Gemini. Esta fase se implementa en Python e integra técnicas estándar de procesamiento de lenguaje natural con el uso de PySentimiento como herramienta central de clasificación automática de sentimientos en español, siguiendo las recomendaciones metodológicas documentadas en la literatura especializada.

El preprocesamiento del texto se implementa a través de una función de limpieza que actúa sobre el campo de contenido de cada reseña. Esta función elimina los prefijos de retuit, los enlaces web en cualquiera de sus formatos, los caracteres de almohadilla y las menciones de usuario. Se eliminan también los caracteres no alfanuméricos, las cadenas compuestas exclusivamente por dígitos y cualquier espacio en blanco redundante, procediendo finalmente a convertir todo el texto a minúsculas. Tras la limpieza, se descartan los registros con contenido nulo y se eliminan los duplicados sobre la columna de contenido limpio para garantizar que cada reseña del corpus sea única e informativa. El resultado de esta primera fase es un corpus depurado y normalizado, almacenado en el campo clean_content del conjunto de datos.

La clasificación de sentimientos se realiza mediante PySentimiento, librería desarrollada por Pérez et al. (2024). Esta herramienta inicializa el analizador al configurar la tarea de sentimiento en español y produce para cada reseña una etiqueta de polaridad (POS para positivo, NEG para negativo, y NEU para neutro) acompañada de las probabilidades de pertenencia a cada categoría. El proceso de clasificación se encapsula en una función que invoca el analizador y registra tanto la etiqueta de mayor probabilidad como el valor de confianza asociado, datos que son almacenados en el conjunto de datos para su uso posterior.

Un aspecto metodológico de especial relevancia dentro de este proceso es el tratamiento de las reseñas de muy corta extensión. Cuando una reseña contiene tres palabras o menos, el modelo de lenguaje enfrenta limitaciones contextuales significativas que pueden comprometer la fiabilidad de su predicción. Por ejemplo, un texto breve como "no funciona" o "muy bien" carece del contexto suficiente para que el clasificador opere con plena seguridad semántica, ya que la ausencia de información adicional impide distinguir matices importantes sobre la intención del usuario. En estos casos, la calificación numérica otorgada por el propio usuario resulta una indicación complementaria más confiable de su intención expresiva, dado que proviene directamente del usuario.

Por lo anterior, cuando la reseña tiene tres palabras o menos y la puntuación asignada es de cuatro o cinco estrellas, la etiqueta de PySentimiento se ajusta a POS con un nivel de confianza mínimo garantizado; del mismo modo, si la puntuación es de una o dos estrellas, la etiqueta se establece como NEG. Para las reseñas de mayor extensión, la predicción de PySentimiento se mantiene sin modificación alguna, dado que el modelo dispone de contexto semántico suficiente. Esta estrategia híbrida, que combina la fortaleza del modelo de lenguaje para textos amplios con la calificación del usuario para reseñas cortas, está alineada con el principio general de diseño de pipelines de etiquetado que aprovechan múltiples fuentes de información, para mejorar la calidad de las etiquetas generadas automáticamente.

El resultado de este proceso integrado es un conjunto de datos etiquetado que contiene las columnas de sentimiento y puntuación de confianza, exportado en el archivo classified.csv. Este archivo constituye la fuente principal de referencia para el entrenamiento del modelo SVM y para la muestra estratificada empleada en la evaluación comparativa con Gemini, garantizando que ambos modelos son evaluados sobre las mismas instancias con el mismo conjunto de etiquetas de referencia y, por tanto, bajo condiciones de comparación equivalentes.

En esta investigación, PySentimiento se utiliza únicamente como herramienta de etiquetado automático para construir el conjunto de entrenamiento del modelo SVM. Al ser una herramienta de código abierto orientada a tareas de minería de opiniones y procesamiento de lenguaje natural social, con soporte para español y otros idiomas, es adecuada para trabajar con texto generado por usuarios en plataformas digitales (Pérez et al., 2024). Sin embargo, estas etiquetas no se consideran el estándar de oro (ground truth) de la investigación, ya que la evaluación formal de Gemini y SVM se realiza con una muestra de validación etiquetada manualmente.

3.4. Diseño de prompts e implementación del modelo Gemini vía API

Desde un punto de vista técnico, metodológico y económico, Gemini representó la alternativa más adecuada para satisfacer las necesidades del presente estudio. En primer lugar, el estudio procesa un volumen considerable de reseñas, por lo que la disponibilidad de un nivel gratuito resultó un factor relevante. Además, Gemini ofrece una API oficial estable, ampliamente documentada y de fácil integración con entornos de desarrollo en Python (lenguaje utilizado durante el desarrollo). De esta forma, es posible reproducir la metodología, ya que permite automatizar el proceso de clasificación y replicar los experimentos bajo las mismas condiciones.

Asimismo, la investigación se centra en una tarea de clasificación supervisada de sentimientos con tres categorías (positivo, negativo y neutro). Debido a que el objetivo principal consiste en obtener etiquetas consistentes y no en generar texto complejo o realizar razonamientos especializados, el empleo de modelos de mayor costo, como GPT-4o o Claude, no representaba una ventaja metodológica significativa. En consecuencia, la relación entre desempeño, accesibilidad y costo convierte a Gemini en una alternativa adecuada para el alcance del estudio.

Por otro lado, el uso de una API permite ampliar el volumen de información procesada sin modificar la metodología propuesta. Así, se facilita que el procedimiento pueda reproducirse posteriormente sobre conjuntos de datos de mayor tamaño o aplicarse a otros dominios con cambios mínimos en la implementación. En particular, Araya et al. (2026) reportan que Gemini, acompañado de un diseño de prompts estructurados, alcanza resultados sólidos en tareas de clasificación de sentimientos en español.

La implementación del modelo Gemini para la clasificación de sentimientos en las reseñas de Pichincha Banca Móvil se realiza a través de la interfaz de programación de aplicaciones de Google AI con la versión gemini-2.5-flash del modelo. El acceso se gestiona

mediante el SDK oficial de Python google-genai, que permite enviar solicitudes estructuradas al modelo y recibir respuestas en formato texto. La temperatura del modelo se configura en un valor bajo para maximizar la consistencia de las respuestas, lo que resulta especialmente importante en tareas de clasificación donde se requiere que el modelo asigne la misma etiqueta ante textos de contenido semántico equivalente.

La implementación se ejecuta de forma incremental mediante un mecanismo de guardado por reseña en el archivo gemini_results.csv, que permite retomar el proceso en caso de interrupción sin pérdida de resultados ya obtenidos (Anexo A). Esta decisión de diseño responde a las restricciones del nivel gratuito de la API de Gemini, que establece un límite de 20 solicitudes diarias para el modelo gemini-2.5-flash, lo que determinó la ejecución distribuida del proceso de clasificación a lo largo de varios días. Para cada reseña se registran la etiqueta asignada, la latencia de respuesta en milisegundos y el conteo de tokens de entrada y salida reportados por la API mediante response.usage_metadata. Estos datos son utilizados posteriormente en el cálculo del costo estimado de procesamiento y en la comparación de eficiencia operativa entre Gemini y SVM.

3.4.1. Diseño del prompt

El diseño del prompt constituye el núcleo del uso de Gemini en esta investigación. Por lo tanto, se adopta un prompt estructurado y único que define explícitamente el rol del modelo como clasificador especializado en reseñas de aplicaciones bancarias en español. El prompt establece el esquema de clasificación de polaridad con tres categorías: POS para opiniones positivas, satisfacción o elogios; NEG para quejas, problemas, fallas o malas experiencias; y NEU para comentarios ambiguos, informativos o sin emoción clara.

Con el objetivo de orientar la clasificación de casos potencialmente ambiguos, el prompt incorpora reglas explícitas: si la reseña menciona un problema técnico, error, bloqueo,

cobro indebido, lentitud o funcionamiento deficiente de la aplicación, debe clasificarse como NEG; si expresa satisfacción con la aplicación o el servicio, debe clasificarse como POS; y si la información disponible es insuficiente para determinar una polaridad clara, debe clasificarse como NEU. El modelo recibe la instrucción de responder únicamente con la etiqueta de clasificación, sin incluir explicaciones adicionales, lo que simplifica el procesamiento automatizado de la respuesta y reduce el riesgo de errores de interpretación.

Araya et al. (2026) demostraron que los prompts con reglas explícitas y criterios bien delimitados para el dominio específico mejoran significativamente la consistencia de los modelos de lenguaje extenso en tareas de clasificación de sentimientos, especialmente en presencia de expresiones ambiguas o figurativas. Es por esto que el diseño del prompt de la presente investigación prioriza la claridad de las instrucciones y la especificidad del dominio bancario en español. Así, el proceso de clasificación se implementa mediante una función Python que envía cada reseña a la interfaz de programación con el prompt diseñado, extrae la etiqueta de la respuesta.

Para la evaluación comparativa, el modelo Gemini se aplica sobre un conjunto de 300 reseñas correspondiente al Ground Truth manual, seleccionado mediante muestreo estratificado a partir del corpus etiquetado por PySentimiento. Este conjunto garantiza una representación equilibrada y estadísticamente suficiente de las tres polaridades, para evitar el sesgo que una distribución natural desbalanceada introduciría en las métricas comparativas. De esta forma, tanto Gemini como SVM son evaluados bajo condiciones equivalentes y las métricas de precision, recall y F1-score reflejan el comportamiento real de cada modelo ante los tres tipos de sentimiento sin alteración por desbalance.

Por lo anterior, se construye el siguiente prompt:

“Eres un clasificador de análisis de sentimientos especializado en reseñas de aplicaciones bancarias en español.

Tu tarea es clasificar la reseña en una sola categoría:

POS = opinión positiva, satisfacción, elogio o buena experiencia.

NEG = queja, problema, falla, molestia, mala experiencia o reclamo.

NEU = comentario ambiguo, informativo, incompleto o sin emoción clara.

Reglas:

- Responde únicamente con POS, NEG o NEU.

- No expliques tu respuesta.

- Si menciona un problema técnico, error, bloqueo, cobro, lentitud o que no funciona, clasifica como NEG.

- Si solo dice "excelente", "buena", "muy bien", "me gusta", clasifica como POS.

- Si no hay suficiente información, clasifica como NEU.”

En primer lugar, se asigna al modelo el rol de clasificador experto en análisis de sentimientos de reseñas de aplicaciones bancarias en español. A continuación, se indica la tarea específica y se incluyen reglas de decisión para orientar la clasificación de casos frecuentes en el dominio bancario. Finalmente, se instruye al modelo para que entregue exclusivamente la etiqueta de clasificación como cadena de texto plano (POS, NEG o NEU), sin explicaciones adicionales, para simplificar el procesamiento automatizado de la respuesta y reducir el riesgo de errores de extracción durante la ejecución del pipeline.

3.5. Entrenamiento e implementación del modelo SVM

El entrenamiento del modelo SVM se implementa en Python mediante la clase LinearSVC de la librería scikit-learn, seleccionada en lugar de SVC. Mientras SVC resuelve un problema de programación cuadrática con complejidad $O(n^2)$ respecto al número de muestras, LinearSVC emplea el algoritmo LIBLINEAR, diseñado específicamente para clasificación lineal en espacios de alta dimensionalidad, con complejidad aproximadamente

lineal respecto al tamaño del corpus (Fan et al., 2008). Esta diferencia es determinante en el contexto de la presente investigación, donde el corpus de entrenamiento comprende miles de reseñas representadas como vectores TF-IDF de 2500 dimensiones.

El corpus etiquetado por PySentimiento presenta una distribución de clases desbalanceada, característica inherente a los corpus de reseñas de aplicaciones bancarias donde los usuarios tienden a expresar predominantemente opiniones polarizadas. Frente a este escenario, se evaluaron tres estrategias de manejo del desbalance: entrenamiento sobre datos originales sin corrección (baseline), submuestreo aleatorio estratificado mediante la función resample de scikit-learn, y sobremuestreo sintético mediante SMOTE (Synthetic Minority Over-sampling Technique). Sin embargo, dado que SMOTE fue diseñado originalmente para datos tabulares continuos y opera mediante interpolación de vecinos cercanos en el espacio de características (Chawla et al., 2002), su aplicación sobre matrices TF-IDF dispersas de alta dimensionalidad genera instancias sintéticas que no corresponden a texto lingüísticamente válido. Blagus y Lusa (2013) demostraron que "SMOTE puede degradar el rendimiento del clasificador en datos de alta dimensionalidad, donde la interpolación entre vectores sparse produce puntos en regiones del espacio de características que no representan observaciones reales" (p. 5). Por esta razón, la estrategia adoptada para el modelo final es el ajuste de pesos de clase mediante el parámetro class_weight='balanced' de LinearSVC, que pondera inversamente la penalización de cada clase según su frecuencia en el corpus de entrenamiento sin alterar la distribución original ni generar instancias sintéticas. Así se sigue la recomendación de King y Zeng (2001) para contextos donde el desbalance refleja la distribución poblacional real del fenómeno estudiado.

La representación vectorial se construye con TfidfVectorizer configurado con max_features=2500, manteniendo el formato sparse de la matriz resultante para reducir el consumo de memoria durante el entrenamiento, dado que LinearSVC acepta directamente

representaciones sparse sin necesidad de conversión a formato denso. El corpus se divide en un 80% para entrenamiento y un 20% para prueba mediante división estratificada (stratify=y) con random_state=42, garantizando que la proporción de clases se preserve en ambos subconjuntos (Anexo B). El modelo final, entrenado sobre los datos originales con class_weight='balanced', es el utilizado para la función de predicción, dado que conserva la distribución real del corpus y produce la representación TF-IDF coherente con el conjunto de entrenamiento completo.

3.6. Protocolo de etiquetado manual y construcción del Ground Truth

Para realizar una evaluación objetiva del desempeño de Gemini y SVM, se construye un conjunto de validación etiquetado manualmente, denominado Ground Truth o estándar de oro. Este conjunto está conformado por una muestra representativa de reseñas extraídas del corpus depurado, seleccionada de forma aleatoria y procurando mantener presencia de las tres categorías de sentimiento: positivo, negativo y neutral. La finalidad de este procedimiento es contar con una referencia humana independiente frente a la cual se puedan comparar las predicciones generadas por ambos modelos.

El conjunto de validación (Ground Truth) estuvo conformado por 300 reseñas etiquetadas manualmente. La selección de este tamaño de muestra respondió a criterios metodológicos y de viabilidad. En primer lugar, el etiquetado manual constituye un proceso intensivo que requiere una evaluación individual de cada instancia, por lo que el tamaño del corpus debe ser suficiente para representar las categorías de análisis y, al mismo tiempo, permitir mantener la consistencia durante el proceso de anotación. En este sentido, Artstein y Poesio (2008) señalan que la calidad del etiquetado depende de la consistencia del anotador y que el incremento excesivo del volumen de datos puede afectar la confiabilidad de las anotaciones debido a la fatiga y la variabilidad humana.

Asimismo, la literatura sobre evaluación de modelos de aprendizaje automático indica que los conjuntos de validación utilizados para comparar clasificadores no requieren necesariamente grandes volúmenes de datos cuando la muestra es representativa del problema de estudio y conserva la distribución de las clases (Kohavi, 1995). En consecuencia, se seleccionaron 300 reseñas distribuidas

entre las tres categorías de sentimiento (positivo, negativo y neutro), garantizando la representación de cada clase y permitiendo una evaluación consistente del desempeño de Gemini y SVM frente al conjunto de referencia etiquetado manualmente.

El etiquetado manual es realizado tomando como unidad de análisis cada reseña completa. Para cada registro, el evaluador asigna una única etiqueta de polaridad: POS, NEG o NEU, como se especifica en la Tabla 1. La etiqueta POS corresponde a reseñas que expresen satisfacción, aprobación, recomendación o una experiencia favorable con la aplicación. La etiqueta NEG se asigna a reseñas que manifiesten quejas, fallas técnicas, errores de autenticación, lentitud, bloqueo, cobros indebidos, frustración o una experiencia negativa. La etiqueta NEU se utiliza en reseñas informativas, ambiguas, incompletas o que no expresen una emoción clara.

Tabla 1

Criterios de etiquetado manual para la construcción del Ground Truth

Etiqueta	Criterio de etiquetado manual	Ejemplo de interpretación
POS	Expresa satisfacción, aprobación o buena experiencia.	"La aplicación funciona muy bien y es rápida."
NEG	Expresa queja, error o frustración.	"No puedo ingresar, siempre se queda cargando."
NEU	Comentario ambiguo, informativo o sin emoción clara.	"Actualicé la aplicación ayer."

Nota. Esta tabla presenta los criterios utilizados para asignar manualmente las etiquetas de sentimiento positivo, negativo y neutral a las reseñas seleccionadas como muestra de validación.

En los casos donde exista contradicción entre la calificación numérica y el texto de la reseña, prevalece el contenido textual, debido a que el objetivo de la investigación es evaluar el sentimiento expresado en lenguaje natural. Asimismo, las reseñas con sarcasmo, ironía, modismos o expresiones coloquiales son etiquetadas considerando la intención comunicativa del usuario y no únicamente el significado literal de las palabras. Este criterio resulta necesario para evitar errores de interpretación en

opiniones donde el usuario aparenta usar expresiones positivas para comunicar una experiencia negativa.

3.7. Métricas y criterios de comparación entre Gemini y SVM

La comparación entre el modelo Gemini y el modelo SVM se estructura en torno a un conjunto de métricas que abarcan tanto la calidad de clasificación como las características operativas de cada enfoque. Para garantizar la comparabilidad de las métricas, tanto Gemini como SVM son evaluados sobre el mismo conjunto de ground truth etiquetado manualmente, descrito en la sección 3.6. En el caso del SVM, las predicciones sobre la muestra de evaluación se obtienen del modelo entrenado con datos originales y class_weight='balanced', cuya implementación completa se documenta en el archivo svm.ipynb (Anexo B). Esta aproximación multidimensional permite una evaluación integral que va más allá del rendimiento predictivo e incorpora consideraciones prácticas relevantes para la toma de decisiones institucional. Alharbi et al. (2025) argumentan que una evaluación comprensiva de los modelos de análisis de sentimientos debe considerar no solo las métricas de calidad de clasificación sino también factores operativos como el costo de procesamiento y la latencia de respuesta, que impactan directamente en la viabilidad de cada enfoque a escala de producción.

En cuanto a la calidad de clasificación, se emplean las métricas estándar de accuracy, precision, recall y F1-score, calculadas en sus versiones macro y weighted para el caso multiclase. La accuracy mide la proporción de instancias clasificadas correctamente sobre el total del conjunto de evaluación. La precision mide la proporción de verdaderos positivos sobre el total de instancias clasificadas como positivas para cada categoría. El recall mide la proporción de verdaderos positivos sobre el total de instancias que pertenecen realmente a esa categoría. El F1-score, calculado como la media armónica de precision y recall, penaliza los desequilibrios entre ambas métricas y resulta especialmente adecuado para contextos donde las

clases no están perfectamente balanceadas, lo que ocurre frecuentemente en corpus de reseñas donde las opiniones negativas suelen ser minoría.

El análisis comparativo entre Gemini y SVM se realiza en referencia con el Ground Truth construido mediante etiquetado manual. Para cada reseña incluida en la muestra de validación, se compara la etiqueta asignada por el evaluador humano con la etiqueta predicha por Gemini y con la etiqueta predicha por SVM. A partir de esta comparación se calculan las métricas de Accuracy, Precision, Recall y F1-Score, considerando las tres clases de sentimiento: positivo, negativo y neutral. Por su parte, la latencia y el costo resultan factores esenciales para el análisis, ya que, en contextos de análisis de reseñas bancarias, la selección del modelo no depende únicamente de la precisión, sino también de su viabilidad técnica y económica para procesar grandes volúmenes de datos (Alharbi et al., 2025).

En cuanto a las métricas operativas, se evalúan el costo estimado de procesamiento, calculado para el modelo Gemini en función del número de tokens procesados por reseña y la tarifa vigente de la interfaz de programación de aplicaciones de Google AI, y para el modelo SVM en términos de tiempo de cómputo de inferencia ejecutado localmente; la latencia de respuesta, medida como el tiempo promedio de procesamiento por reseña en milisegundos; y la interpretabilidad, evaluada cualitativamente a partir de la capacidad de cada modelo de proporcionar justificaciones comprensibles para sus clasificaciones. La combinación de estas métricas permite formular recomendaciones equilibradas sobre la adopción de cada enfoque en función de los requerimientos específicos del contexto de aplicación.

Estos criterios se resumen en la Tabla 2, donde se presentan las métricas utilizadas para la evaluación comparativa de los modelos.

Tabla 2

Métricas utilizadas para la evaluación comparativa de los modelos

Métrica	Forma de cálculo	Propósito
Accuracy	Predicciones correctas / total de reseñas evaluadas	Medir el acierto general del modelo.
Precision	Verdaderos positivos / total predicho como una clase	Medir qué tan confiables son las predicciones de cada clase.
Recall	Verdaderos positivos / total real de una clase	Medir qué tanto recupera el modelo de cada sentimiento real.
F1-Score	Media armónica entre Precision y Recall	Evaluar equilibrio entre precisión y cobertura.
Latencia promedio	Tiempo total de clasificación / número de reseñas	Comparar rapidez de procesamiento.
Costo por 1.000 reseñas	Costo estimado del procesamiento proyectado a 1.000 reseñas	Comparar viabilidad económica.

Nota. La tabla describe las métricas empleadas para evaluar el desempeño de Gemini y SVM frente al conjunto de validación etiquetado manualmente.

Por su parte, la estimación específica de costos y tiempos de procesamiento se construye con los componentes considerados para cada modelo, diferenciando entre el enfoque basado en API (Gemini) y el modelo tradicional de aprendizaje automático (SVM). Así, es posible comprender, con mayor precisión, cómo se comportan ambos sistemas en términos de eficiencia computacional y viabilidad económica en escenarios de uso real. La Tabla 3 presenta esta descomposición de variables, incluyendo los elementos considerados para el cálculo de costo y latencia en cada modelo.

Tabla 3

Estimación de costo y latencia por modelo de análisis de sentimientos

Métrica	Modelo	Elementos considerados	Cálculo estimado
Costo por 1.000 reseñas	Gemini	Tokens de entrada + tokens de salida + tarifa API	Costo promedio por reseña × 1.000
Costo por 1.000 reseñas	SVM	Ejecución local después del entrenamiento	Costo computacional marginal
Latencia	Gemini	Tiempo de solicitud y respuesta API	Tiempo total / número de reseñas
Latencia	SVM	Vectorización TF-IDF + predicción local	Tiempo total / número de reseñas

Nota. La tabla presenta los elementos considerados para estimar el costo y la latencia de procesamiento en Gemini y SVM.

3.8. Análisis comparativo de casos complejos: sarcasmo, ironía y modismos

El análisis comparativo de casos complejos representa una dimensión cualitativa de la evaluación que complementa las métricas cuantitativas descritas en la sección anterior. Este estudio se centra, específicamente, en la capacidad de cada modelo para clasificar correctamente reseñas que contienen sarcasmo, ironía, modismos y expresiones coloquiales ecuatorianas. Esta dimensión es especialmente relevante porque los errores cometidos por los modelos en este tipo de textos no son aleatorios sino sistemáticos, y su efecto acumulado puede sesgar significativamente los resultados del análisis de sentimientos. Por lo tanto, es necesario tomar en cuenta estos casos al interpretar las métricas globales de evaluación.

Para este análisis, se extrae del corpus de evaluación una submuestra de reseñas identificadas como instancias que contienen expresiones no literales, mediante la detección de patrones de incongruencia entre el contenido léxico y la calificación numérica del usuario, la presencia de marcadores discursivos irónicos y el uso de términos coloquiales con carga

semántica específica en el español ecuatoriano. Qin et al. (2025) destacan que el sarcasmo en plataformas digitales se manifiesta frecuentemente como una alabanza superficial con una intención subyacente negativa, lo que requiere que el sistema de clasificación acceda a información contextual que va más allá de la semántica léxica de las palabras individuales.

Los casos de ironía, sarcasmos y modismos son especialmente relevantes para comparar la capacidad de Gemini, que puede acceder al contexto pragmático completo mediante su prompt especializado, frente al modelo SVM, que opera sobre representaciones vectoriales estáticas del vocabulario sin modelado explícito del contexto enunciativo. Los resultados de este análisis se presentan como un estudio de caso que ilustra con ejemplos concretos del corpus las fortalezas y limitaciones de cada enfoque. Como resultado, se formulan recomendaciones prácticas sobre las condiciones de uso más adecuadas para cada modelo.

3.9. Diseño e integración del dashboard interactivo

El dashboard interactivo constituye el producto de visualización, que sintetiza y comunica los resultados del análisis de sentimientos de manera accesible para usuarios sin formación técnica especializada, como gerentes o analistas de producto del Banco Pichincha. Su diseño sigue los principios de visualización de datos en contextos de toma de decisiones asistida por inteligencia artificial identificados por Datta et al. (2025), que incluyen la claridad visual, la capacidad de filtrado interactivo, las representaciones temporales de tendencias y las métricas de resumen prominentes. Los autores destacan que el diseño de herramientas de visualización de datos para responsables de la toma de decisiones debe priorizar la claridad, la interactividad y la divulgación progresiva de la complejidad analítica. El diseño no debe abrumar a los usuarios con resultados estadísticos crudos.

El dashboard se desarrolla en Microsoft Power BI Desktop, herramienta de visualización de datos que permite construir informes interactivos con capacidad de filtrado

dinámico sin requerir programación. La arquitectura del informe sigue un modelo de filtros cruzados donde el usuario selecciona parámetros a través de segmentadores de datos: periodo temporal (Año, Mes), categoría de sentimiento y rango de calificación numérica (score 1–5). De esta forma, todos los componentes visuales se actualizan automáticamente en respuesta a esas selecciones.

Entre los componentes principales se definen los siguientes indicadores clave: tarjetas de resumen con el promedio de calificación, el total de reseñas analizadas y la distribución porcentual de sentimientos por clase; la distribución de sentimientos segmentada por calificación numérica (score 1–5); la evolución temporal de los sentimientos por año; y los tópicos más frecuentes del corpus mediante una nube de palabras sobre el texto preprocesado. Por lo tanto, la Tabla 4 presenta los KPIs definidos para implementar en el dashboard:

Tabla 4

Indicadores clave para el dashboard de análisis de sentimientos

KPI	**Descripción**	**Utilidad**
Total de reseñas analizadas	Número total de reseñas procesadas.	Mide el tamaño del corpus trabajado.
Distribución de sentimientos	Porcentaje de reseñas POS, NEG y NEU.	Identifica la percepción general de los usuarios.
Evolución temporal del sentimiento	Variación de sentimientos por fecha, semana o mes.	Permite detectar periodos de mejora o deterioro.
Promedio de calificación por sentimiento	Relación entre estrellas y polaridad textual.	Ayuda a verificar coherencia entre puntuación y opinión.

KPI	Descripción	Utilidad
Tópicos frecuentes	Palabras o temas más mencionados en las reseñas.	Identifica asuntos recurrentes para los usuarios.

Nota. La tabla presenta los indicadores clave de desempeño que serán incorporados en el dashboard interactivo para visualizar los resultados de la clasificación de sentimientos en las reseñas de Google Play.

De esta manera, el dashboard no solo cumple una función comunicativa de los resultados de la investigación, sino que también actúa como prototipo de herramienta de monitoreo continuo de la percepción del usuario en la banca digital, cuya implementación permanente podría generar valor sostenido para la institución.

CAPÍTULO IV: ANÁLISIS Y COMPARACIÓN DE RESULTADOS

El Capítulo IV presenta los resultados obtenidos a partir de la aplicación de los modelos Gemini API y Support Vector Machines (SVM) sobre el conjunto de validación construido para la investigación. Se expone la distribución del corpus de evaluación, las métricas de desempeño de ambos enfoques y el análisis comparativo de sus resultados en términos de precisión, recall, F1-score, latencia y costo computacional. Finalmente, se sintetizan los hallazgos en una comparación global y se presentan las conclusiones y recomendaciones derivadas del estudio, junto con la integración de los resultados en un dashboard interactivo que permite su visualización y análisis aplicado.

4.1. Descripción del corpus de evaluación

La investigación emplea dos conjuntos de datos con propósitos diferentes. Por un lado, se utilizó una muestra de reseñas para el procesamiento automático mediante Gemini durante la fase de implementación. Por otro lado, se construyó un conjunto de validación (Ground Truth) conformado por 300 reseñas etiquetadas manualmente siguiendo el protocolo descrito en la Sección 3.6. Estas 300 reseñas fueron posteriormente clasificadas por Gemini para comparar sus predicciones con las etiquetas de referencia humana. En consecuencia, todas las métricas de desempeño presentadas en este capítulo, incluidas las Tablas 6, 7 y 9, se calcularon exclusivamente sobre este conjunto de validación. La Tabla 5 resume la distribución de clases de los conjuntos utilizados en el estudio.

La Tabla 5 presenta la distribución de clases de los conjuntos considerados en la investigación. La columna correspondiente a PySentimiento (entrenamiento) muestra la distribución completa del corpus etiquetado automáticamente, conformado por 60923 reseñas, este sirvió como base para el entrenamiento del modelo SVM. En contraste, las columnas Ground Truth manual y Gemini API (procesadas) corresponden exclusivamente al conjunto de

evaluación de 300 reseñas. En este último caso, la distribución del Ground Truth refleja las etiquetas asignadas manualmente, mientras que la distribución de Gemini representa las categorías predichas por el modelo sobre esas mismas 300 reseñas.

Tabla 5

Distribución de clases por conjunto de datos

Clase	**PySentimiento (entrenamiento)**	**Ground Truth manual**	**Gemini API (procesadas)**
Positivo (POS)	34344	147	135
Negativo (NEG)	19423	96	146
Neutro (NEU)	7156	57	19
Total	60923	300	300

Nota. La columna PySentimiento (entrenamiento) presenta la distribución de clases del corpus completo (60 923 reseñas), mientras que las columnas Ground Truth manual y Gemini API (procesadas) corresponden únicamente al conjunto de evaluación de 300 reseñas.

4.2. Resultados de clasificación: Gemini API

La Tabla 6 presenta las métricas de clasificación obtenidas por Gemini API sobre el conjunto de evaluación (Ground Truth), conformado por 300 reseñas etiquetadas manualmente. Estas métricas se calcularon comparando las predicciones del modelo con las etiquetas de referencia humana.

Tabla 6

Métricas de clasificación de Gemini API por clase de sentimiento

Clase	**Precision**	**Recall**	**F1-Score**	**Support**
POS	0.97	0.89	0.93	147
NEG	0.65	0.99	0.79	96

NEU	0.89	0.30	0.45	57
Accuracy global	0.81	—	—	300

Nota. Resultados obtenidos sobre el Ground Truth de 300 reseñas etiquetadas manualmente (Anexo C).

Los resultados obtenidos contra el Ground Truth manual revelan un comportamiento diferenciado por clase. Gemini logra una precision muy alta en POS (0.97) y NEU (0.89), lo que indica que cuando predice estas clases lo hace con gran certeza. Sin embargo, su recall en NEU es bajo (0.30), lo que significa que clasifica incorrectamente el 70% de las reseñas neutras, asignándolas mayoritariamente como NEG (37 casos). Para la clase negativa, Gemini alcanza un recall casi perfecto (0.99), clasificando correctamente 95 de las 96 reseñas negativas del Ground Truth. Este patrón sugiere que el modelo tiene un sesgo hacia la detección de negatividad, coherente con el diseño del prompt que incorpora múltiples reglas orientadas a identificar quejas y fallas técnicas.

La causa principal de este comportamiento se evidencia en la matriz de confusión presentada en la Tabla 6: el modelo tiende a clasificar las reseñas neutras como positivas (3 casos) o negativas (37 casos), asignando solo 17 de las 57 reseñas neutras correctamente. Este fenómeno es atribuible a la naturaleza del prompt diseñado, cuyas reglas explícitas priorizan la detección de polaridad sobre la identificación de ambigüedad. Reseñas como "muy segura la app" o "sin problemas hasta ahora" fueron clasificadas como POS cuando PySentimiento las etiquetó como NEU por carecer de expresión emocional explícita.

Tabla 7

Matriz de confusión de Gemini API

Real \ Predicho	**POS**	**NEG**	**NEU**
POS	131	14	2
NEG	1	95	0

NEU	3	37	17

Nota. Filas: etiqueta real del Ground Truth manual. Columnas: etiqueta predicha por Gemini API.

La matriz de confusión, demostrada en la Tabla 7, confirma que el principal error de Gemini es la confusión entre NEU y NEG: 37 de las 57 reseñas neutras fueron clasificadas como negativas, lo que eleva artificialmente la recall NEG pero penaliza la detección de ambigüedad. Reseñas como "mejor funcionamiento muy hábiles para cualquier transferencia" o "una buena aplicación lamentable cuando se colapsa en fin de mes" fueron clasificadas como NEG cuando en el Ground Truth se asignaron NEU por su carácter mixto o meramente descriptivo.

4.3. Resultados de clasificación: SVM con LinearSVC

El modelo SVM fue entrenado con LinearSVC sobre el corpus completo etiquetado por PySentimiento, empleando vectorización TF-IDF con max_features=2500 y ngram_range=(1,2), y el parámetro class_weight='balanced' para manejar el desbalance de clases sin alterar la distribución original del corpus. La división entrenamiento y prueba fue estratificada en proporciones 80/20 con random_state=42 (Anexo B). La Tabla 8 presenta las métricas obtenidas sobre el conjunto de prueba.

Tabla 8

Métricas de clasificación de SVM (LinearSVC) por clase de sentimiento

Clase	**Precision**	**Recall**	**F1-Score**	**Support**
POS	0.96	0.74	0.84	147
NEG	0.74	0.86	0.80	96
NEU	0.43	0.56	0.49	57
Accuracy global	0.75	—	—	300

Nota. Resultados obtenidos sobre el Ground Truth de 300 reseñas etiquetadas manualmente. Modelo entrenado con class_weight='balanced' (Anexo C).

Como se puede ver en la tabla 8, el modelo SVM alcanza una accuracy de 0.75 sobre el Ground Truth manual, inferior a Gemini (0.81). Su comportamiento más equilibrado entre clases se evidencia en la recall de NEU (0.56), superior a la de Gemini (0.30), lo que indica que SVM identifica con mayor eficacia las reseñas ambiguas gracias a los patrones léxicos aprendidos del corpus de entrenamiento, como los conectores "pero", "sería" y "gustaría" identificados en los coeficientes del modelo. Sin embargo, SVM presenta una precision baja en NEU (0.43), lo que significa que casi la mitad de sus predicciones neutras son incorrectas.

El análisis de los términos más discriminativos por clase, extraídos de los coeficientes del modelo, revela patrones léxicos coherentes con el dominio bancario. Para la clase negativa, los términos con mayor peso son "peor" (3.7219), "pésimo" (3.5877), "no" (3.1460) y "basura" (2.4644), lo que refleja el vocabulario de queja y frustración típico de reseñas insatisfechas. Para la clase positiva destacan "exelente" (3.8672), "encanta" (3.7411), "excelente" (3.2310) y "buena" (3.2682), incluyendo variantes ortográficas coloquiales frecuentes en el corpus. Para la clase neutra, los términos más discriminativos incluyen "pero" (2.5847), "sería" (1.8861) y "gustaría" (1.6473), conectores discursivos y condicionales que caracterizan reseñas con valoraciones mixtas o sugerencias de mejora.

Tabla 9

Matriz de confusión de SVM: valores de verdaderos positivos por clase

Real \ Predicho	**POS**	**NEG**	**NEU**
POS	109	8	30
NEG	1	83	12
NEU	4	21	32

Nota. Los valores en diagonal corresponden a verdaderos positivos por clase según el support y accuracy reportados (Anexo B).

La Tabla 9 presenta la matriz de confusión del modelo SVM evaluado exclusivamente sobre el Ground Truth de 300 reseñas etiquetadas manualmente, el conjunto de referencia utilizado para la comparación directa con Gemini API. Esta evaluación es independiente del conjunto de prueba del corpus de PySentimiento, sobre el cual el modelo fue entrenado y evaluado internamente (svm.ipynb). El principal error del modelo se concentra en la confusión entre POS y NEU (30 reseñas positivas clasificadas como neutras), lo que sugiere que el vocabulario moderado sin marcadores de entusiasmo explícito tiende a ser interpretado por el clasificador como ambiguo.

4.4. Comparación de métricas operativas: latencia y costo

La selección de un modelo de análisis de sentimientos para aplicaciones bancarias requiere considerar su viabilidad operativa en términos de latencia de respuesta y costo de procesamiento, además de la calidad clasificatoria. Alharbi et al. (2025) argumentan que, en contextos institucionales, los factores de eficiencia computacional y económica son determinantes para la adopción de modelos a escala de producción. La Tabla 10 consolida las métricas operativas medidas durante la ejecución de ambos modelos.

Tabla 10

Comparación de métricas operativas entre Gemini API y SVM

Métrica operativa	**Gemini API**	**SVM (LinearSVC)**	**Observación**
Latencia promedio	3.555,57 ms	< 1 ms (local)	SVM sin dependencia de red
Latencia mínima	2,42 ms	< 1 ms	

Latencia máxima	29.982,58 ms	< 1 ms	Alta variabilidad en Gemini por API
Costo por reseña	$0,00003056 USD	$0 (local)	SVM sin costo marginal de inferencia
Costo 300 reseñas	~$0,009168 USD	$0	
Costo por 60923 reseñas	~$1,86 USD	$0	Gemini viable a escala pequeña
Requiere internet	Sí	No	SVM apto para entornos sin conectividad

Nota. La latencia de Gemini fue registrada en milisegundos por llamada mediante response.usage_metadata (Anexo A). El costo se calculó con la tarifa vigente de gemini-2.5-flash: $0,15 USD por millón de tokens de entrada y $0,60 USD por millón de tokens de salida (Google, 2025). La latencia del SVM corresponde a inferencia local sobre vectores TF-IDF sparse con LinearSVC.

Los resultados muestran una diferencia significativa en latencia promedio: Gemini API registra un promedio de 3.555,57 ms por reseña, con alta variabilidad (mínimo 2,42 ms, máximo 29.982,58 ms), atribuible a la congestión de la red y la disponibilidad del servidor de Google en el momento de cada solicitud. Esta variabilidad constituye una limitación operativa relevante para sistemas en tiempo real. El modelo SVM ejecuta la inferencia localmente, eliminando la dependencia de red y reduciendo la latencia a tiempos de cómputo puro inferiores a 1 ms por instancia.

En cuanto al costo, Gemini API presenta un costo unitario de $0,00003056 USD por reseña, equivalente a aproximadamente $0,009168 USD para la muestra de 300 reseñas y $1,86 USD para el corpus completo. Aunque este costo es económicamente marginal a la escala del presente estudio, se incrementa linealmente con el volumen de datos, lo que puede resultar relevante para instituciones financieras con grandes volúmenes de reseñas. El modelo SVM, al

ejecutarse localmente una vez entrenado, no incurre en costos de inferencia adicionales más allá de los recursos computacionales del entorno de ejecución, lo que lo convierte en la opción más económica a cualquier escala de procesamiento.

4.5. Análisis de casos complejos: clase neutra y lenguaje ambiguo

El análisis de los errores sistemáticos de ambos modelos revela que la clase neutra representa el principal desafío compartido, aunque con magnitudes distintas. Gemini clasifica incorrectamente 40 de las 57 reseñas neutras (70% de error), mientras que SVM alcanza un recall de 0.56 para esta clase, lo que equivale a un 46% de error. Esta diferencia es atribuible a mecanismos de aprendizaje distintos: Gemini aplica reglas léxicas basadas en el prompt diseñado, que priorizan la detección de polaridad explícita; mientras que SVM aprende patrones estadísticos a partir del corpus de entrenamiento.

Entre los casos más representativos de error de Gemini se encuentran reseñas como "muy segura la app", "mejor funcionamiento muy hábiles para cualquier transferencia" o "sin problemas hasta ahora", clasificadas como POS cuando PySentimiento las etiquetó como NEU por ser meramente descriptivas. De manera inversa, reseñas como "lenta lenta" o "falta que se caiga menos" fueron clasificadas como NEG cuando expresan una sugerencia de mejora sin frustración directa. Qin et al. (2025) señalan que la ambigüedad semántica en reseñas de aplicaciones digitales se manifiesta frecuentemente como expresiones que contienen valoraciones mixtas o intenciones pragmáticas que no coinciden con el sentido literal de las palabras individuales. Esto representa un desafío estructural para sistemas de clasificación basados en reglas léxicas o prompts de instrucción.

Una limitación adicional identificada en la comparación corresponde al tratamiento de reseñas de longitud muy reducida. PySentimiento requiere contexto textual mínimo para inferir polaridad, razón por la cual se implementó una regla de respaldo basada en la calificación

numérica para reseñas de tres palabras o menos. Gemini, en cambio, clasifica correctamente reseñas como "lags" (NEG) o "práctica" (POS) mediante conocimiento contextual del dominio, sin necesidad de información adicional. Brown et al. (2020) documentan esta capacidad de los modelos de lenguaje extenso de inferir sentimiento a partir de una sola palabra mediante conocimiento del mundo preentrenado, lo que representa una ventaja estructural de Gemini frente a SVM en corpus con alta proporción de reseñas cortas.

4.6. Síntesis comparativa y recomendaciones

La Tabla 11 consolida las dimensiones de comparación entre Gemini API y SVM para facilitar la interpretación global de los resultados.

Tabla 11

Síntesis comparativa entre Gemini API y SVM por dimensión de evaluación

Dimensión	**Gemini API**	**SVM (LinearSVC)**	**Ventaja**
Accuracy global (vs GT)	0.81	0.75	Gemini
F1-Score POS	0.93	0.84	Gemini
F1-Score NEG	0.79	0.80	SVM
F1-Score NEU	0.45	0.49	SVM
Recall POS	0.89	0.74	Gemini
Recall NEG	0.99	0.86	Gemini
Recall NEU	0.30	0.56	SVM
Latencia promedio	3.555 ms	< 1 ms	SVM
Costo de inferencia	$0,00003 / reseña	$0	SVM

Dimensión	**Gemini API**	**SVM (LinearSVC)**	**Ventaja**
Textos muy cortos	Alto rendimiento	Limitado (TF-IDF)	Gemini
Interpretabilidad	Baja (caja negra)	Alta (pesos TF-IDF)	SVM
Sin internet	No funciona	Funciona	SVM
Sin corpus etiquetado	Funciona (zero-shot)	No funciona	Gemini

Nota. Los valores corresponden a la evaluación final sobre el Ground Truth de 300 reseñas.

Los resultados obtenidos contra el Ground Truth manual revelan que Gemini API supera a SVM en accuracy global (0.81 vs 0.75) y en la detección de reseñas positivas (F1=0.93 vs 0.84) y negativas (recall=0.99 vs 0.86). Este resultado es coherente con los hallazgos de Qin et al. (2025), quienes documentan que los modelos de lenguaje extenso superan a los clasificadores tradicionales en tareas de análisis de sentimientos cuando el criterio de evaluación es el juicio humano, dado que estos modelos incorporan conocimiento pragmático y contextual que se aproxima más al razonamiento del evaluador humano. Sin embargo, SVM demuestra mayor capacidad para identificar reseñas neutras (recall NEU = 0.56 vs 0.30), gracias a los patrones léxicos de ambigüedad aprendidos del corpus de entrenamiento.

La elección entre Gemini y SVM depende por tanto del contexto específico de aplicación y de los recursos disponibles. Gemini es superior cuando se requiere máxima sensibilidad en la detección de reseñas negativas, una característica crítica en monitoreo de crisis de reputación bancaria, o cuando no se dispone de corpus etiquetado en el dominio, operando en modalidad zero-shot sin necesidad de datos de entrenamiento previos (Brown et al., 2020). SVM es preferible cuando se prioriza la eficiencia operativa sin costo de inferencia, la interpretabilidad de las predicciones mediante pesos TF-IDF auditables, la operación en entornos sin conectividad, o la identificación de reseñas con lenguaje ambiguo y mixto. Alharbi

et al. (2025) argumentan que, en contextos institucionales bancarios, la interpretabilidad y el costo operativo son frecuentemente tan determinantes como la precisión clasificatoria para la adopción sostenida de modelos de inteligencia artificial.

En consecuencia, para el contexto específico de análisis de sentimientos de reseñas bancarias en español ecuatoriano con un corpus etiquetado disponible, se recomienda el modelo SVM con LinearSVC como herramienta principal de clasificación debido a su superior eficiencia operativa, nulo costo de inferencia e implementación, y mejor capacidad de identificar la clase neutra; a pesar de que Gemini API alcanzó una exactitud global moderadamente mayor. Gemini API representa una alternativa viable para escenarios de análisis exploratorio, clasificación de textos muy cortos, o contextos institucionales donde la disponibilidad de datos etiquetados es limitada y el costo de la API es asumible dentro del presupuesto operativo.

4.7. Dashboard interactivo de visualización de resultados

El dashboard implementado en Microsoft Power BI Desktop integra los resultados del análisis de sentimientos sobre el corpus completo de 60.923 reseñas de Pichincha Banca Móvil en una interfaz interactiva accesible para usuarios no técnicos. La Figura 1 presenta la vista general del dashboard, que consolida los principales indicadores definidos en la sección 3.9.

Figura 1

Dashboard interactivo de análisis de sentimientos de reseñas de Pichincha Banca Móvil en Google Play



Nota. El dashboard fue desarrollado en Microsoft Power BI Desktop sobre el corpus de 60.923 reseñas procesadas. Elaboración propia (2026).

Como se observa en la Figura 1, el panel superior incluye tres segmentadores de datos que permiten filtrar el corpus por periodo (Año, Mes), rango de calificación numérica (score 1–5) y categoría de sentimiento (POS, NEG, NEU). De esta forma, es posible actualizar dinámicamente todos los componentes de la vista. Además, el dashboard incorpora una opción para restablecer los filtros aplicados, así el dashboard brinda la opción de regresar a la configuración inicial de las visualizaciones. Las tarjetas de resumen en el panel izquierdo presentan los indicadores principales: total de reseñas analizadas (61K), índice de satisfacción (24.49%), promedio de calificación general (3.78), y distribución porcentual de sentimientos (56.37% positivas, 31.88% negativas, 11.75% neutras). Estos valores revelan que la percepción general del usuario hacia Pichincha Banca Móvil es predominantemente positiva, aunque la proporción de reseñas negativas (31.88%) indica un nivel de insatisfacción relevante que justifica el monitoreo continuo de la herramienta.

El gráfico de barras apiladas titulado "Distribución de sentimientos por calificación" evidencia la coherencia entre la calificación numérica del usuario y la polaridad textual identificada por el modelo: las reseñas con calificación 1 y 2 concentran la mayor proporción de sentimientos negativos, mientras que las calificaciones 4 y 5 corresponden predominantemente a reseñas positivas. La calificación 3 presenta una distribución más equilibrada entre las tres clases, lo que confirma su naturaleza ambigua y su correspondencia con la clase neutra identificada en el análisis.

El gráfico de líneas "Evolución temporal de los sentimientos" muestra la distribución de reseñas POS, NEG y NEU a lo largo del tiempo, permitiendo identificar periodos de variación atípica en la percepción del usuario. Se observa un incremento significativo en el volumen de reseñas hacia el año 2025, con un pico pronunciado que podría estar asociado a una actualización mayor de la aplicación o a un evento de servicio relevante. La nube de palabras generada con el campo clean_content del corpus identifica los términos más frecuentes, entre los que destacan "aplicación", "banco", "banca", "transferencia" y "error", lo que confirma que las reseñas giran en torno a la funcionalidad transaccional de la aplicación y a incidencias técnicas.

CAPÍTULO V: CONCLUSIONES Y RECOMENDACIONES

El Capítulo V presenta las conclusiones derivadas del desarrollo de la investigación, las cuales se fundamentan en el análisis e interpretación de los resultados obtenidos. Las conclusiones se elaboran en función del cumplimiento del objetivo general y de los objetivos específicos planteados. Asimismo, se exponen las recomendaciones orientadas tanto a la mejora de la metodología propuesta como a su posible implementación en entornos reales y al desarrollo de futuras investigaciones.

5.1. Conclusiones

El corpus de 60.923 reseñas extraídas de Google Play mediante google-play-scraper, constituyó una base amplia y diversa. Este conjunto de datos incluyó reseñas desde una sola palabra hasta textos de varias líneas, con calificaciones que van de una a cinco estrellas y publicadas en un rango de varios años. La variedad recolectada fue relevante para la investigación porque evidenció, desde el inicio, que el corpus no era homogéneo: había reseñas demasiado cortas para que cualquier modelo de lenguaje las procesara con fiabilidad, y reseñas con expresiones coloquiales ecuatorianas que ninguna herramienta generalista maneja bien. El pipeline de preprocesamiento implementado, que preservó los términos de negación y aplicó una regla de respaldo numérico para textos de tres palabras o menos, respondió directamente a estas características del corpus.

El hallazgo más relevante de la investigación no es cuál modelo fue mejor, sino en qué falló cada uno y por qué. Por su parte, Gemini clasificó correctamente el 81% de las reseñas del Ground Truth, con un recall casi perfecto en la clase negativa (0.99), pero fue incapaz de identificar reseñas neutras: clasificó incorrectamente 40 de las 57 reseñas que el evaluador humano etiquetó como NEU, asignándolas mayoritariamente como NEG.

Por lo anterior, se puede identificar un error estructural, en el que el prompt diseñado priorizaba la detección de polaridad y las reglas explícitas no contemplaban la ambigüedad como categoría válida por defecto. Reseñas como "sin problemas hasta ahora" o "mejor funcionamiento, muy hábiles" fueron clasificadas como positivas cuando en realidad son descriptivas y no expresan una emoción clara.

Por otro lado, SVM, alcanzó un accuracy de 0.75 contra el Ground Truth, pero su recall en la clase neutra fue de 0.56, notablemente superior al de Gemini. Esta ventaja de debe a que el modelo había aprendido del corpus que conectores como "pero", "sería" y "gustaría" son señales de ambigüedad. Sin embargo, ninguno de los dos modelos resolvió bien la clase neutra, y este resultado es en sí mismo un aporte, puesto que evidencia que la ambigüedad en reseñas de banca digital ecuatoriana representa un desafío sistemático que no se resuelve ni con reglas de prompt ni con TF-IDF estándar.

Por el lado operativo, la diferencia entre los modelos es drástica. SVM clasifica una reseña en menos de 1 ms sin conexión a internet y sin costo por consulta; en cambio, Gemini tardó en promedio 3.555 ms por reseña, con picos de casi 30 segundos atribuidos a la congestión del servidor, y tiene un costo que escala linealmente con el volumen. Para las 300 reseñas del corpus de evaluación, el costo total de Gemini fue de aproximadamente $0.01, lo que parece marginal, pero proyectado a las 60.923 reseñas del corpus completo representaría alrededor de $1.86, y a escala de producción continua, el costo acumulado puede volverse relevante para una institución que procese decenas de miles de reseñas mensuales. SVM, una vez entrenado, no tiene ese problema.

Dicho esto, Gemini tiene una ventaja real que SVM no puede replicar: funciona sin datos de entrenamiento. Si el Banco Pichincha lanzara mañana una nueva aplicación sin ninguna reseña etiquetada disponible, Gemini clasificaría desde el primer día con un rendimiento aceptable. SVM, en cambio, requiere un corpus etiquetado en el dominio para

funcionar; sin él, no hay modelo. Esta diferencia es prácticamente importante en contextos donde los datos son escasos o el dominio cambia.

El dashboard desarrollado en Power BI mostró que la distribución del corpus completo es 56.37 positiva, 31.88% negativa y 11.75% neutra. La proporción de reseñas negativas es significativa: casi un tercio de los usuarios que dejaron una reseña expresó una experiencia desfavorable. Los términos más frecuentes en ese segmento, "error", "no funciona", "lenta" y "transferencia", apuntan a que los problemas técnicos en funcionalidades transaccionales son la principal fuente de insatisfacción, no la experiencia general de la aplicación. Esta información concreta es la que el análisis de sentimientos convierte en accionable para el equipo de producto.

Finalmente, es importante reconocer que los resultados de esta investigación no conducen a una respuesta única sobre cuál modelo debe utilizarse, si Gemini o SVM. Por lo que, es imprescindible evaluar cada situación, en la que: si la prioridad es no perder ninguna reseña negativa, Gemini es más adecuado. Por el contrario, si la prioridad es operar de forma autónoma, económica e interpretable sobre un corpus del dominio, SVM es la mejor opción. Además, en caso de que la institución se enfrentara un corpus nuevo sin etiquetas, Gemini ofrece un beneficio significante, ya que permite comenzar sin realizar un entrenamiento previo. En este sentido, la decisión no es únicamente técnica, sino también estratégica, pues depende del equilibrio entre precisión, costo, interpretabilidad y disponibilidad de datos.

5.2. Recomendaciones

La investigación evidencia que el problema más urgente a resolver en ambos modelos es la clase neutra. Para SVM, la alternativa más directa es enriquecer el corpus de entrenamiento con reseñas de esa categoría recopiladas de forma dirigida. Del mismo modo, se podría explorar el aumento de características lingüísticas explícitas en la vectorización:

marcadores de condicionalidad como "sería", "podría", "si" y conectores adversativos como "pero" o "aunque" tienen un peso discriminativo demostrado en el modelo entrenado y podrían potenciarse con n-gramas de tres palabras. Para Gemini, la solución es incorporar al diseño del prompt ejemplos de reseñas neutras tomados directamente del corpus ecuatoriano; por ejemplo, textos como "sin problemas hasta ahora" o "la app funciona bien en general" para evitar que el modelo los clasifique como positivos.

Se recomienda que el Banco Pichincha considere la implementación del modelo SVM dentro de un pipeline automatizado que procese de forma periódica las nuevas reseñas publicadas en Google Play e integre los resultados con el dashboard de Power BI desarrollado en este estudio. Este pipeline podría emitir alertas cuando la proporción de reseñas negativas supere un umbral predefinido, o cuando aparezcan términos de error nuevos no registrados previamente en el corpus. D esta forma, sería posible que el equipo técnico reaccione de manera oportuna antes de que el problema afecte la calificación pública de la aplicación. Del mismo modo, el dashboard funcionaría como una herramienta de apoyo para la toma de decisiones, facilitando el monitoreo continuo del comportamiento de las reseñas.

Para investigaciones futuras se recomienda ampliar el Ground Truth e incorporar un segundo evaluador para calcular el coeficiente kappa de Cohen. En esta investigación, las 300 reseñas fueron etiquetadas por una sola persona, lo que introduce el riesgo de criterios inconsistentes en los casos frontera, precisamente donde los modelos también fallan. Conocer el nivel de acuerdo entre varios evaluadores humanos permitiría establecer una cota superior más realista sobre el desempeño alcanzable por cualquier modelo automatizado en esta tarea.

Finalmente, se recomienda explorar en trabajos futuros el uso de modelos con fine-tuning sobre corpus bancarios en español latinoamericano, como RoBERTuito ajustado con reseñas de aplicaciones financieras, o la aplicación de Gemini bajo un esquema de few-shot learning sistemático en lugar de un enfoque zero-shot. Los resultados de esta investigación

sugieren que la brecha entre ambos enfoques se concentra en la clase neutra y en textos muy cortos, dos casos donde la representación léxica estática de TF-IDF y las reglas de prompt sin ejemplos llegan a su límite. En este sentido, un modelo ajustado específicamente a este tipo de datos podría reducir dicha brecha.

REFERENCIAS BIBLIOGRÁFICAS

Alam, M. S., Mrida, M. S. H., y Rahman, M. A. (2025). *Sentiment analysis in social media: How data science impacts public opinion knowledge integrating Natural Language Processing (NLP) with Artificial Intelligence (AI).* American Journal of Scholarly Research and Innovation, 4(01), 63-100.

Alharbi, A., Alotaibi, S., Almuqbil, M., y Alshobaili, J. (2025). *Sentiment analysis of digital banking reviews using machine learning and large language models.* Electronics, 14(11), 2125. https://doi.org/10.3390/electronics14112125

Araya, J., Mora, F., Villegas, R., y Casas, C. (2026). *Enhancing sentiment classification and irony detection in large language models through advanced prompt engineering techniques*. arXiv:2601.08302. https://arxiv.org/abs/2601.08302

Artstein, R., & Poesio, M. (2008). *Inter-coder agreement for computational linguistics*. Computational Linguistics, 34(4), 555–596. https://doi.org/10.1162/coli.07-034-R2

Bimantara, R., y Zufria, I. (2024). *Text mining sentiment analysis on mobile banking application reviews using TF-IDF method with natural language processing approach.* JINAV: Journal of Information and Visualization, 5(1), 115-123. https://doi.org/10.35870/jinav.v5i1.2772

Blagus, R., y Lusa, L. (2013). *SMOTE for high-dimensional class-imbalanced data*. *BMC Bioinformatics*, *14*(1), 106. https://doi.org/10.1186/1471-2105-14-106

Chawla, N. V., Bowyer, K. W., Hall, L. O., y Kegelmeyer, W. P. (2002). *SMOTE: Synthetic minority over-sampling technique*. *Journal of Artificial Intelligence Research*, *16*, 321–357. https://doi.org/10.1613/jair.953

Datta, A., Sharma, P., Rajan, V., y Gupta, N. (2025). *Data visualization in AI-assisted decision-making: A systematic review*. Frontiers in Communication, 10. https://doi.org/10.3389/fcomm.2025.1605655

Fan, R.-E., Chang, K.-W., Hsieh, C.-J., Wang, X.-R., y Lin, C.-J. (2008). LIBLINEAR: A library for large linear classification. *Journal of Machine Learning Research*, *9*, 1871–1874. https://www.jmlr.org/papers/v9/fan08a.html

Gupta, S., Ranjan, R., y Naik, A. (2024). *Comprehensive study on sentiment analysis: From rule-based to modern LLM based system*. arXiv:2409.09989. https://arxiv.org/abs/2409.09989

Kamath, S., Narendra, V. G., y Shivaprasad, G. (2025). *Comparative analysis of sentiment analysis techniques employing machine learning*. En Trends in Sustainable Computing and Machine Intelligence (ICTSM 2024) (pp. 129-138). Springer. https://doi.org/10.1007/978-981-96-1452-3_10

King, G., y Zeng, L. (2001). *Logistic regression in rare events data*. *Political Analysis*, *9*(2), 137–163. https://doi.org/10.1093/pan/9.2.137

Kohavi, R. (1995). *A study of cross-validation and bootstrap for accuracy estimation and model selection*. In Proceedings of the 14th International Joint Conference on Artificial Intelligence (Vol. 2, pp. 1137–1143).

Pérez, J. M., Rajngewerc, M., Giudici, J. C., Furman, D. A., Luque, F., Alonso Alemany, L., y Martínez, M. V. (2024). *pysentimiento: A Python toolkit for opinion mining and social NLP tasks*. arXiv:2106.09462. https://arxiv.org/abs/2106.09462

Qin, Z., Luo, Q., Zang, Z., y Fu, H. (2025). *Detecting sarcasm in user-generated content integrating transformers and gated graph neural networks. PeerJ Computer Science*. https://doi.org/10.7717/peerj-cs.2817

Rani, M., Kumar, S., Sharma, A., y Goyal, P. (2025). *Sarcasm detection in sentiment analysis using recurrent neural networks*. International Journal of Distributed Sensor Networks. https://doi.org/10.1155/dsn/8479411

Samanmali, P. H. C., y Rupasingha, R. A. H. M. (2024). *Sentiment analysis on Google Play Store app users' reviews based on deep learning approach*. Multimedia Tools and Applications, 83, 84425-84453. https://doi.org/10.1007/s11042-024-19185-w

Teles, L. E. P., y Figueiredo, C. M. S. (2025). *Comparing LLMs for sentiment analysis in financial market news.* arXiv:2510.15929. https://arxiv.org/abs/2510.15929

ANEXOS

ANEXO A

Jupyter Notebook para el procesamiento automático de reseñas mediante Gemini

Este anexo presenta el código utilizado para el procesamiento automático de las 300 reseñas que conforman el conjunto de validación (Ground Truth). El procedimiento comprende la carga del conjunto de datos, la configuración de la API de Gemini, la clasificación automática de las reseñas, el registro de la latencia y del consumo de tokens, así como el almacenamiento incremental de los resultados.

A.1 Importación de librerías

```
import pandas as pd
import time
import os
from google import genai
```

A.2 Carga y preparación del conjunto de datos

```
df = pd.read_excel("ground_truth_etiquetar.xlsx")
df = df.rename(columns={"etiqueta_manual": "sentiment"})
df = df[df["clean_content"].notnull()]
print(df["sentiment"].value_counts())
```

A.3 Configuración del cliente Gemini

```
client = genai.Client(api_key=api_key)
archive_result = "gemini_results.csv"
if os.path.exists(archive_result):
    df_result = pd.read_csv(archive_result)
    df_result = df_resultdf_result['sentiment_gemini'] != 'ERROR']
```

```
    already_processed = set(df_result['index_original'].tolist())
    print(f"Reseñas ya procesadas correctamente: {len(already_processed)} reseñas")
else:
    df_result = pd.DataFrame(columns=[
        'index_original', 'clean_content', 'sentiment_real',
        'sentiment_gemini', ' latency', 'tokens_input', 'tokens_output'
    ])
    already_processed = set()
def classify_gemini(text):
    prompt = f"""Eres un clasificador de análisis de sentimientos especializado en reseñas de aplicaciones bancarias en español.

Tu tarea es clasificar la reseña en una sola categoría:

POS = opinión positiva, satisfacción, elogio o buena experiencia.
NEG = queja, problema, falla, molestia, mala experiencia o reclamo.
NEU = comentario ambiguo, informativo, incompleto o sin emoción clara.

Reglas:
- Responde únicamente con POS, NEG o NEU.
- No expliques tu respuesta.
- Si menciona un problema técnico, error, bloqueo, cobro, lentitud o que no funciona, clasifica como NEG.
- Si solo dice "excelente", "buena", "muy bien", "me gusta", clasifica como POS.
- Si no hay suficiente información, clasifica como NEU.
```

```
Reseña:
\"\"\"{text}\"\"\"""
    intentos = 0
    max_intentos = 5
    while intentos < max_intentos:
        inicio = time.time()
        try:
            response = client.models.generate_content(
                model="gemini-2.5-flash",
                contents=prompt
            )
            latency = round((time.time() - inicio) * 1000, 2)
            label = response.text.strip().upper()
            if label not in ["POS", "NEG", "NEU"]:
                label = "NEU"
            tokens_in = response.usage_metadata.prompt_token_count
            tokens_out = response.usage_metadata.candidates_token_count
            return label, latency, tokens_in, tokens_out

        except Exception as e:
            intentos += 1
            if '429' in str(e) and 'PerDay' in str(e):
                print("Límite diario alcanzado. Guarda el progreso y vuelve mañana.")
                break
```

```
            espera = 15 * intentos
            print(f"  Intento {intentos}/{max_intentos} — esperando {espera}s... ({e})")
            time.sleep(espera)

    return "LIMITE", 0, 0, 0

for idx, row in df.iterrows():
    if idx in already_processed:
        continue
    label, latency, tokens_in, tokens_out = classify_gemini(row['clean_content'])
    if label == "LIMITE":
        print(f"Progreso guardado: {len(df_result)} reseñas.")
        break
    new_row = pd.DataFrame([{
        'index_original': idx,
        'clean_content': row['clean_content'],
        'sentiment_real': row['sentiment'],
        'sentiment_gemini': label,
        'latency': latency,
        'tokens_input': tokens_in,
        'tokens_output': tokens_out
    }])
    df_result = pd.concat([df_result, new_row], ignore_index=True)
    df_result.to_csv(archive_result, index=False)
    processed = len(df_result)
```

```
    if processed % 50 == 0:
        print(f"Progreso: { processed }/300")

    time.sleep(7)
print(f"{len(df_result)} reseñas procesadas")
```

ANEXO B

Jupyter Notebook para la implementación del clasificador SVM

Este anexo presenta el código utilizado para la implementación del modelo SVM empleado en la investigación. Se incluyen únicamente las etapas necesarias para la preparación de los datos, entrenamiento del modelo y evaluación de su desempeño.

B.1 Importación de librerías

```
import pandas as pd
import numpy as np
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.model_selection import train_test_split
from sklearn.svm import LinearSVC
from sklearn.metrics import (
    accuracy_score,
    precision_score,
    recall_score,
    f1_score,
    classification_report,
    confusion_matrix
)
```

B.2 Carga y preparación del conjunto de datos

```
path = 'classified.csv'
df = pd.read_csv(path, sep='','', encoding='utf-8', engine='python' )
```

B.3 Vectorización mediante TF-IDF

```
tfidf = TfidfVectorizer(max_features=2500)
```

```
X = tfidf.fit_transform(df["clean_content"])

y = df["sentiment"]
```

B.4 División en entrenamiento y prueba

```
X_train, X_test, y_train, y_test = train_test_split(X, y,  test_size=0.20, random_state=42,

stratify=y)
```

B.5 Entrenamiento del modelo SVM

```
svm = LinearSVC()

svm.fit(X_train, y_train)
```

B.6 Predicción y evaluación

```
y_pred = svm.predict(X_test)

accuracy = accuracy_score(y_test, y_pred)

precision = precision_score(y_test, y_pred, average="weighted")

recall = recall_score(y_test, y_pred, average="weighted")

f1 = f1_score(y_test, y_pred, average="weighted")
```

ANEXO C

Jupyter Notebook para la evaluación comparativa entre Gemini y SVM

Este anexo presenta el código utilizado para comparar el desempeño de Gemini y SVM frente al conjunto de validación (Ground Truth). Se incluyen las etapas de carga de datos, integración de resultados, cálculo de métricas de evaluación y generación de matrices de confusión.

C.1 Importación de librerías

```
import pandas as pd
from sklearn.metrics import classification_report, accuracy_score, confusion_matrix
```

C.2 Carga del conjunto de validación

```
gt = pd.read_excel('ground_truth_etiquetar.xlsx')
gt = gt.rename(columns={'etiqueta_manual': 'label_gt'})
gemini = pd.read_csv('gemini_results.csv')
gemini = gemini.rename(columns={'sentiment_gemini': 'label_gemini'})
print(f"Ground Truth: {len(gt)} reseñas")
print(f"Gemini: {len(gemini)} reseñas")
```

Resultado obtenido

Ground Truth: 300 reseñas

Gemini: 300 reseñas

C.3 Integración de resultados GT y Gemini

```
df_gemini = gt.merge(
    gemini[['clean_content', 'label_gemini', 'latencia', 'tokens_input', 'tokens_output']],
    on='clean_content',
    how='inner'
)
```

```
print(f"Reseñas con match Gemini: {len(df_gemini)}")
print(df_gemini['label_gt'].value_counts())
```

Resultado obtenido

```
Reseñas con match Gemini: 300
label_gt
POS    147
NEG     96
NEU     57
Name: count, dtype: int64
```

C.4 Integración de resultados GT y SVM

```
svm_pred = pd.read_csv('svm_predictions.csv')
df_svm = gt.merge(
    svm_pred[['clean_content', 'label_svm']],
    on='clean_content',
    how='inner'
)
print(f"Reseñas con match SVM: {len(df_svm)}")
```

Resultado obtenido

```
Reseñas con match SVM: 300
```

C.5 Evaluación de modelos

```
print("=== GEMINI vs GROUND TRUTH ===")
print(f"Accuracy: {accuracy_score(df_gemini['label_gt'], df_gemini['label_gemini']):.4f}")
print(classification_report(df_gemini['label_gt'], df_gemini['label_gemini'],
labels=['POS','NEG','NEU']))
```

```
cm_g = confusion_matrix(df_gemini['label_gt'], df_gemini['label_gemini'],
labels=['POS','NEG','NEU'])
print(pd.DataFrame(cm_g, index=['POS','NEG','NEU'], columns=['POS','NEG','NEU']))
print("\n=== SVM vs GROUND TRUTH ===")
print(f"Accuracy: {accuracy_score(df_svm['label_gt'], df_svm['label_svm']):.4f}")
print(classification_report(df_svm['label_gt'], df_svm['label_svm'],
labels=['POS','NEG','NEU']))
cm_s = confusion_matrix(df_svm['label_gt'], df_svm['label_svm'],
labels=['POS','NEG','NEU'])
print(pd.DataFrame(cm_s, index=['POS','NEG','NEU'], columns=['POS','NEG','NEU']))
```

Resultado obtenido

```
=== GEMINI vs GROUND TRUTH ===
Accuracy: 0.8100
              precision    recall  f1-score   support

         POS       0.97      0.89      0.93       147
         NEG       0.65      0.99      0.79        96
         NEU       0.89      0.30      0.45        57

    accuracy                           0.81       300
   macro avg       0.84      0.73      0.72       300
weighted avg       0.85      0.81      0.79       300

     POS  NEG  NEU
```

POS 131 14 2

NEG 1 95 0

NEU 3 37 17

=== SVM vs GROUND TRUTH ===

Accuracy: 0.7467

	precision	recall	f1-score	support
POS	0.96	0.74	0.84	147
NEG	0.74	0.86	0.80	96
NEU	0.43	0.56	0.49	57
accuracy			0.75	300
macro avg	0.71	0.72	0.71	300
weighted avg	0.79	0.75	0.76	300

POS NEG NEU

POS 109 8 30

NEG 1 83 12

NEU 4 21 32