Instituto Tecnológico
y de Estudios Superiores de Occidente

Reconocimiento de validez oficial de estudios de nivel superior según acuerdo secretarial 15018, publicado en el Diario Oficial de la Federación del 29 de noviembre de 1976.

Departamento de Matemáticas y Física

Maestría en Ciencia de Datos

Aplicación de modelado de tópicos en reseñas de hospedajes de Airbnb en Berlín de 2010 a 2019

TESIS que para obtener el **GRADO** de
MAESTRO EN CIENCIA DE DATOS

Presenta: **GUSTAVO IBÁÑEZ SOSA**

Asesora **DRA. GEMA BERENICE GUDIÑO MENDOZA**

Tlaquepaque, Jalisco. 10 de mayo de 2023.

AGRADECIMIENTOS

Quiero expresar mi profundo agradecimiento a todas las personas que me brindaron su apoyo y contribuyeron de alguna manera en la realización de este trabajo.

En primer lugar, agradezco de manera especial a mi asesora, Berenice Gudiño, por su dedicación y orientación en cada etapa de la investigación, lo cual fue fundamental para el desarrollo de este proyecto.

También quiero agradecer al ITESO, por brindarme la oportunidad de realizar mis estudios y desarrollar este proyecto. El ambiente académico y el apoyo brindado por la institución fueron esenciales para el desarrollo de esta tesis.

Por otro lado, no puedo dejar de mencionar a mis padres, Jessica Sosa y Gustavo Ibáñez, quienes me dieron la oportunidad de estudiar este postgrado y siempre me mostraron su apoyo para seguir adelante y alcanzar mis metas.

A todos ellos y ellas, gracias por su apoyo y por ser una fuente de motivación en todo momento.

DEDICATORIA

A mis papás quienes siempre me han invitado a cuestionar y a aprender.

RESUMEN

El Procesamiento de Lenguaje Natural se ha convertido en una disciplina clave en la era digital. Con el aumento en la cantidad de información que se genera diariamente en diferentes formatos, es necesario contar con herramientas que permitan analizar, entender y categorizar los datos. Ante esta realidad se desarrolla este trabajo explorando una de estas herramientas, el modelado de tópicos.

Este trabajo se divide en dos principales partes, en la primera se hace una comparativa de forma metodológica, de tres modelos (LDA, GSDMM y BERTopic) para el modelado de tópicos. Y en la segunda parte, se hace un análisis de tópicos con el modelo seleccionado de la primera parte. El trabajo se desarrolló con una base de datos de reseñas de alojamientos de Airbnb Berlín. Previa a las dos principales partes se realizó un pretratamiento de la base de datos, el cual incluye pasos como selección de columnas, detección de idioma de las reseñas, traducción al inglés, tokenizado, eliminación de stopwords y stemming. Posterior a esto se realizó la comparativa empleando las siguientes métricas, valor de coherencia, análisis visual y tiempo de ejecución. De la comparativa se seleccionó el modelo LDA, debido a que es el que presenta mejores resultados en todas las métricas. Enseguida realizo una optimización de los parámetros, haciendo un barrido de los valores posibles en un rango definido. Como resultado del modelo optimizado se encontraron cinco tópicos principales en el conjunto de reseñas, los cuales se categorizan como “Positivo General”, “Negativo General”, “Ubicación Airbnb”, “Positivo Host” y “Habitaciones/Espacio del Airbnb”. Siendo “Positivo General” el tópico principal en el conjunto y el tópico “Habitaciones/Espacio del Airbnb” muestra mayor relevancia en reseñas escritas en español y francés.

Modelado de tópicos, LDA, GSDMM, BERTopic, Reseñas

TABLA DE CONTENIDO

LISTA DE FIGURAS

LISTA DE TABLAS

LISTA DE ACRÓNIMOS Y ABREVIATURAS

GSDMM	Mezcla de Muestreo Gibbs Dirichlet Multinomial
LDA	Asignación Latente de Dirichlet
LSA	Análisis Semántico Latente
NFM	Factorización No Negativa de Matrices
PLSA	Análisis Semántico Latente Probabilístico
PLDA	Asignación Latente de Dirichlet Paralela
BTM	Modelado de Tópicos Bitermino
NLP	Procesamiento de Lenguaje Natural
SVD	Descomposición en Valores Singulares
ABSA	Análisis de Sentimientos Basado en Aspectos

1. INTRODUCCIÓN

En este capítulo, se aborda brevemente los antecedentes del modelado de tópicos, la justificación y el problema que se busca abordar en el desarrollo del trabajo, que es el análisis de tópicos de conjuntos masivos de reseñas. Asimismo, se define el objetivo principal, que es la comparativa de tres modelos para el modelado de tópicos en un conjunto de reseñas, y se establecen los objetivos específicos que guiarán el desarrollo del trabajo.

1.1. Antecedentes

El modelado de tópicos es una técnica que consiste en identificar los tópicos que se encuentran en un conjunto de textos. Esta identificación es realizada analizando los patrones en la distribución y frecuencia de las palabras en los textos, y agrupándolas en clústeres, los cuales a su vez pueden ser categorizados como tópicos [1].

Esta técnica se remonta a 1998 con el desarrollo de la técnica conocida como Análisis Semántico Latente (LSA) [2] con el cual se exploraba la repetición de palabras en los conjuntos de documentos utilizando una descomposición en Valores Singulares. Buscando mejorar esta primera técnica, en 1999, se desarrolla la Factorización No Negativa de Matrices (NFM) [3]. En el mismo año de 1999 surge el modelo de Análisis Semántico Latente Probabilístico (PLSA) [4], siendo el primer modelo probabilístico para modelado de tópicos, además de ser el primer modelo que define la idea de que cada documento es una construcción de diferentes tópicos, pensando en el análisis de textos largos. Esta idea da pie al desarrollo de otros modelos enfocados en textos largos.

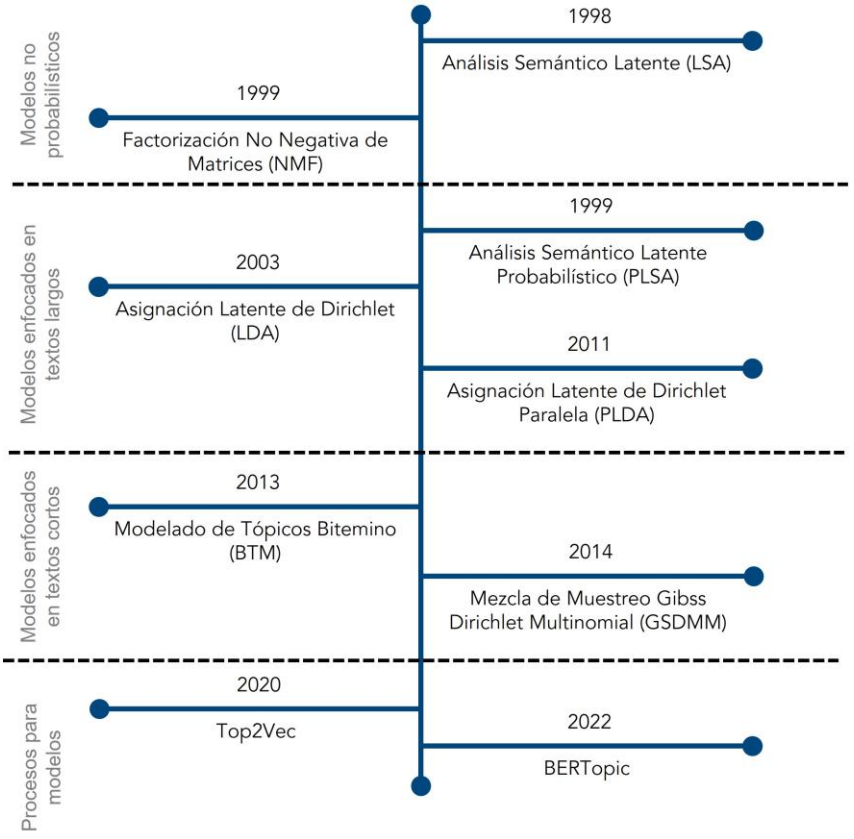
En 2003 se desarrolla el modelo de Asignación Latente de Dirichlet (LDA) [5], el cual incluye la distribución de Dirichlet en el cálculo de las probabilidades para la asignación de los tópicos a los documentos. A la fecha este es uno de los principales modelos utilizado en el análisis de tópicos. En 2011, con el objetivo de optimizar los tiempos de procesamiento de LDA se desarrolla el modelo de Asignación Latente de Dirichlet Paralela (PLDA) [6].

Aproximadamente en 2013 el modelado de tópicos entra en una nueva etapa, la cual se caracteriza por el análisis de textos cortos. Esta etapa viene acompañada del desarrollo de nuevos modelos que cambian la idea principal de varios tópicos por documento a un solo tópico por documento. En 2013 se publica el modelo de Modelado de Tópico Bitermino (BTM) [7] y en 2014 el modelo de Mezcla de Muestreo Gibbs Dirichlet Multinomial [8], este último teniendo un funcionamiento muy similar a LDA, con la diferencia de la idea base donde los documentos solo permiten ser asignados a un documento.

En años más recientes el modelado de tópicos ha entrado a una etapa diferente donde ha dejado un poco de lado el desarrollo de modelos en sí, para iniciar el desarrollo de procesos para el modelado de tópicos estos son procesos con módulos independientes, donde cada módulo permite el uso de distintas técnicas se utilicen. Esta independencia de módulos permite que distintos modelos para modelado de tópicos se construyan a partir de un mismo proceso, dando así la versatilidad de atacar textos con distintas características, solo cambiando las técnicas utilizadas en cada módulo. Ejemplos de estos procesos son Top2Vec [9], desarrollado en 2020 y BERTopic [10] desarrollado en 2022.

En la figura 1.1 se presenta la línea de tiempo de los modelos para modelado de tópicos, así como las distintas etapas a las que corresponden cada modelo.

Figura 1.1 Línea de tiempo del modelado de tópicos.



Elaboración propia.

1.2. Justificación

En los últimos años, la generación de datos ha tenido un aumento que va más allá de la capacidad humana para procesarlos. Ante esta imposibilidad y la necesidad de seguir obteniendo la información que dan estos datos, distintas áreas de investigación se han desarrollado para encontrar maneras de solucionarlo. Claramente el volumen de datos no ha sido la única dificultad con la que se han encontrado estas áreas, de hecho, la separación en áreas se ha dado por la especialización de cada una en resolver una de las dificultades. Un ejemplo, es la dificultad de tener que emplear computadoras para entender, interpretar y utilizar lenguajes humanos, para atacar este problema nace el área de Procesamiento de Lenguaje Natural (NLP), que a su vez se ha dividido en subáreas que permiten enfocarse en problemáticas más puntuales. Algunas de estas subáreas son: categorización de contenidos, modelado de tópicos, extracción contextual, análisis de sentimientos, texto a voz, voz a texto, traducción automática, entre otros.

El Procesamiento de Lenguaje Natural ha permitido a investigadores y empresas continuar explorando datos como hacían anteriormente, pero también les ha dado la oportunidad de

explorar nuevos datos que han aparecido con la digitalización de nuestro día a día. Principalmente, las empresas están interesadas en conocer la percepción que sus clientes o usuarios tienen sobre ellas, ya que esto es un criterio importante para su crecimiento en el mercado. Una forma de conocer esta percepción es por medio de las reseñas que se reciben, muchas veces de forma masiva y en distintos formatos, aquí es donde el Procesamiento de Lenguaje Natural toma relevancia ya que les extiende a estas empresas una gran gama de herramientas con las cuales trabajar.

1.3. Problema

Año con año las reseñas se han vuelto una fuente importante de información para las empresas, ya sea que ofrezcan un servicio o producto. Cada comentario positivo o negativo, da información que se puede emplear para construir una mejor experiencia para los clientes y usuarios. Conforme las empresas crecen sus operaciones también crece la cantidad de retroalimentación que reciben, llegando incluso a millones de reseñas recibidas en cierto periodo. Un ejemplo de estas empresas es Airbnb con aproximadamente 150 millones de usuarios y más de 6 millones de alojamientos disponibles en el mundo.

Si bien no se podría analizar cada reseña, existen técnicas que permiten obtener los aspectos relevantes del conjunto completo. Estas técnicas se encuentran dentro del Procesamiento de Lenguaje Natural (NLP) y una de ellas es el modelado de tópicos, la cual permite encontrar, a través de algoritmos estadísticos, los principales temas o tópicos de conjuntos de documentos.

1.4. Hipótesis

Es posible seleccionar un modelo que haga una clasificación efectiva de los tópicos principales, a través del análisis de diferentes modelos para el modelado de tópicos considerando métricas cuantitativas y cualitativas.

1.5. Objetivos

1.5.1. Objetivo General

Analizar distintos modelos de Procesamiento de Lenguaje Natural (NLP) para el modelado de tópicos. Derivado de este análisis se elegirá el de mejor desempeño, para realizar una exploración de tópicos en una base de datos de reseñas de Airbnb.

1.5.2. Objetivos Específicos

- Realizar la limpieza y el pretratamiento de la base de datos original para aplicar los modelos a explorar.

- Implementar 3 modelos y realizar una evaluación de su desempeño para definir el modelo a utilizar en la exploración de tópicos.
- Realizar un análisis de las reseñas empleando el modelo seleccionado y aplicando distintos niveles de filtrado a los datos.

1.6. Novedad científica, tecnológica o aportación

La novedad científica de este trabajo es presentar de forma metodológica la comparación de tres modelos de tópico en una base de datos de reseñas de un servicio. Para hacer la evaluación de los modelos y posterior comparación se utilizaron tres métricas, las cuales son tiempo de ejecución, valor de coherencia y un análisis visual de los tópicos generados. Cada uno de los tres modelos evaluados tiene una característica específica, un modelo está especializado en textos largos, otro en textos cortos y el último modelo es un proceso de módulos independientes que permite construir distintos modelos para el modelado de tópicos.

Como se verá en el siguiente capítulo del Estado del Arte, los trabajos anteriores en los que se ha hecho una comparativa entre diferentes modelos para el modelado de tópicos, se han enfocado principalmente en la evaluación de modelos que son una evolución uno de otro. Hay algunos trabajos que, si hacen comparativas entre modelos que no están directamente relacionados, pero suelen ser modelos con características similares y no modelos desarrollados para diferentes tipos de documentos. Adicional a la comparativa de modelos, este trabajo busca realizar un análisis más profundo con el modelo, que las métricas identificaron como el mejor.

2. ESTADO DEL ARTE O DE LA TÉCNICA

En este capítulo se presenta un resumen de los trabajos relacionados con el modelado de tópicos, se hará un repaso de los trabajos más relevantes para el desarrollo de este campo, así como los trabajos más recientes para poder entender hacia donde se ha dirigido el área.

2.1. Modelado de tópicos

El modelado de tópicos es un área de investigación del Procesamiento del Lenguaje Natural (NLP) que busca identificar los tópicos más representativos en un conjunto de documentos o textos. A lo largo de su desarrollo han existido diferentes algoritmos y técnicas que han buscado hacer la clasificación de tópicos de la mejor manera.

2.1.1. Análisis Semántico Latente (LSA)

Una de las primeras aproximaciones se conoce como Análisis Semántico Latente (LSA) [2], esta técnica explora la aparición de las palabras en los documentos y representa el texto como una matriz. Esta matriz tiene en las columnas los documentos, y en las filas las palabras que aparecen en todos los documentos, asignando en su cruce la frecuencia de las palabras en cada documento, después LSA [2] aplica una descomposición en valores singulares (SVD) [11] a la matriz, permitiendo identificar patrones de ocurrencia y similitud entre palabras y documentos.

La técnica de LSA se ha empleado en diferentes trabajos, con distintos objetivos y enfoques. Hay trabajos enfocados al análisis de conjuntos de documentos académicos como trabajos de investigación [12], resúmenes de publicaciones científicas [13], aportaciones de estudiantes a plataformas de tutoría [14], entre otros. También se ha utilizado fuera del ámbito académico para analizar disciplinas o campos de trabajo a partir de artículos relacionados [15], identificar y crear grupos de usuarios objetivo [16] y buscar aplicaciones alternativas de la técnica como la medición de la coherencia en textos [17]. También se han desarrollado trabajos donde se explora la técnica de forma que explican su funcionamiento, limitantes y aplicaciones [18] [19].

2.1.2. Análisis Semántico Latente Probabilístico (PLSA)

A partir de LSA [2] se construye una técnica llamada Análisis Semántico Latente Probabilístico (PLSA) [4], la cual es una variación que en lugar de identificar los tópicos utilizando SVD [11] se hace a partir de un modelo estadístico. PLSA [4] modela la probabilidad conjunta de ver una palabra y un documento juntos como una mezcla de distribuciones multinomiales condicionalmente independientes.

PLSA mostró mejores desempeños en comparación a LSA [2], pero aun así ambas técnicas tienen desventajas similares. Algunas de estas son el uso de conceptos similares como elementos diferentes de sus matrices, lo que provoca que ambas técnicas necesiten una fase de preprocesamiento bastante extensa para lograr mitigar esta característica. Otra desventaja es que se debe conocer o definir previamente el número de tópicos que se desea identificar y por último otra desventaja es que estas técnicas no son capaces de asignar una probabilidad a un documento nuevo [20].

Desde que la técnica de PLSA se desarrolló se ha utilizado en diferentes líneas de trabajo como aquellas que buscan extender la capacidad de la técnica al implementar variaciones [21], hacer comparativas de desempeño entre PLSA y otros modelos [22] [23] y líneas de trabajo con aplicaciones como la detección de comportamientos anormales en videos de tráfico [24], la segmentación de documentos a partir de un análisis de tópicos [25], análisis de patrones de navegación en internet [26], el resumen automático de documentos orales [27], entre otros.

2.1.3. Asignación Latente de Dirichlet (LDA)

Para mejorar los algoritmos planteados hasta el momento, se desarrolla la Asignación Latente de Dirichlet (LDA) [5], este algoritmo mejora los anteriores al utilizar la distribución de Dirichlet para estimar las distribuciones documento-tópico y termino-tópico con un enfoque bayesiano. Con este nuevo enfoque LDA [5] logró mejores resultados que LSA [2] y PLSA [4], además de tener otras ventajas como la posibilidad de asignar una probabilidad a documentos nuevos y la posibilidad de incorporarse a modelos más complejos como un módulo probabilístico.

LDA [5] también tiene sus desventajas y estas son principalmente, que se tiene que conocer previamente el número de tópicos en los que se quiere agrupar los documentos, un preprocesamiento importante para obtener resultados coherentes, una generación de temas demasiado generales [28] o irrelevantes [29] y que los resultados pueden ser inconsistentes entre diferentes ejecuciones [30].

LDA es una de las técnicas más ampliamente utilizada para el modelado de tópicos y sus líneas de trabajo aledañas. Algunas de las líneas de trabajo más relevantes son aquellas que buscan extender la capacidad de la técnica, implementando variaciones [31] [32] [33], la comparación con otras técnicas [34] [35] [36] [37], la clasificación de documentos [38] e imágenes [39] [40] [41], el resumen de documentos [42], la detección de tendencias en diferentes tipos de documentos [43] [44], la aplicación online del modelo [45] [46] y los trabajos donde se hace un análisis de LDA explicando su funcionamiento, limitantes y aplicaciones [47] [48].

LSA, PLSA y LDA son algunos de los principales algoritmos en el modelado de tópicos, esto ha hecho que muchas investigaciones se hayan basado en ellos para buscar mejorarlos. En estas investigaciones se han desarrollado otra gran cantidad de modelos como son Factorización no Negativa de Matrices (NMF) [3], que es una variación de LSA [2] donde obliga a que los valores de las matrices resultantes no puedan ser negativos, y Asignación Latente de Dirichlet Paralela (PLDA) [6], el cual es una variación de LDA [5] que busca reducir el tiempo de ejecución, esto lo logra dividiendo el corpus general de los documentos en subcorpus a modo de lotes y encontrando distribuciones locales con las cuales después se calcula una distribución global para el corpus completo.

Los algoritmos desarrollados hasta ahora comparten algunas características, pero la más importante de ellas es que todos estos modelos parten de la idea de que cada documento está formado por un conjunto de tópicos. Ante los cambios en la sociedad y la necesidad de analizar documentos con formatos diferentes a los disponibles cuando se inició el desarrollo del modelado de tópicos, surgen modelos que parten de una primicia distinta, esta primicia plantea que cada documento solo puede estar formado por un tópico, con este cambio lo que se buscó es tener mejores resultados en textos cortos como lo son reseñas, tweets, publicaciones de redes sociales, etc.

2.1.4. Mezcla de Muestreo Gibbs Dirichlet Multinomial (GSDMM)

Uno de estos modelos donde se plantea que cada documento solo puede estar formado por un tópico es Mezcla de Muestreo Gibbs Dirichlet Multinomial (GSDMM) [8], este modelo es una variación directa de LDA ya que utiliza la distribución de Dirichlet [49] pero además tiene la característica de utilizar iteraciones para converger a un resultado, teniendo la ventaja contra otros algoritmos de que no es necesario definir el número de tópicos en los que se quieren agrupar los documentos, sino que se plantea un número máximo de tópicos.

La técnica GSDMM al ser una variación del modelo más utilizado, hace que sus principales líneas de trabajo sean en la comparación con otros modelos con distintos tipos de documentos desde textos largos como libros [50] hasta textos cortos [51] [52] como tweets.

2.1.5. BERTopic

Ante la necesidad de obtener todavía mejores resultados inicia el desarrollo de nuevos algoritmos, la principal característica de estos algoritmos es que más que modelos son procesos, esto quiere decir que están construidos por una serie de módulos, donde cada módulo cumple con una función para poder llegar a un resultado. Al tener estos módulos permiten tener una mayor versatilidad. Los módulos que manejan este tipo de algoritmos son la integración de documentos, reducción de dimensionalidad, agrupación y representación de tópicos [53]. Al existir independencia entre módulos, permite que cada uno de ellos cuente con distintas técnicas y que un mismo algoritmo pueda construir diferentes modelos. Algunos ejemplos de este tipo de algoritmos son BERTopic [10] y Top2Vec [9], donde su principal diferencia son las técnicas que permiten en cada uno de sus módulos.

BERTopic es una técnica desarrollada recientemente y los trabajos desarrollados con este modelo siguen dos grandes líneas. La primera es la comparación con otros modelos para saber cómo es su desempeño para el modelado de tópicos en distintos tipos de documentos [50] [52] [54] [55], y la segunda línea es la prueba del modelo. Algunas de estas pruebas son el modelado de tópicos en diferentes idiomas [56] [57], la detección de tendencias [58], el análisis de tópicos de distintos campos de estudios [59] [60] [61] [62] [63] y bases de datos [64] [65].

Como ya se mencionó, la principal ventaja de estos algoritmos sobre los demás es la modularidad que tienen, adicional a esta tienen otras como que no es necesario especificar un número de tópicos en los que se desea agrupar los documentos, no es necesario un preprocesamiento tan riguroso, y que también funcionan bajo la idea de que un documento solo puede estar formado por un tópico [20]. Esta última puede ser vista como ventaja o desventaja dependiendo del tipo de documento con el que se trabaja y la aplicación del proyecto. Estos algoritmos tienen otras ventajas y desventajas asociadas a las técnicas que se seleccionan en cada uno de sus módulos.

Después de haber analizado los diferentes modelos y sus distintas aplicaciones en proyectos desarrollados, se optó por elegir tres modelos para el desarrollo del trabajo, los modelos seleccionados fueron LDA, GSDMM y BERTopic. Se encontró que estos tres modelos habían sido utilizados para analizar textos de diferentes índoles y se habían encontrado resultados satisfactorios, adicional son modelos que tienen ampliamente documentada su implementación, y además cada uno de ellos pertenece a diferentes etapas del modelado de tópicos. LDA surgió como un modelo para textos largos el cual permite asignar un conjunto de tópicos a cada documento, GSDMM cambió a un modelo para textos cortos, en el cual únicamente se asigna un tópico por documento y BERTopic siendo parte de los últimos avances donde tenemos un proceso modular que nos permite la construcción de modelos con diferentes características.

3. MARCO TEÓRICO/CONCEPTUAL

En este capítulo se presentan las bases teóricas y conceptuales sobre el modelado de tópicos. Empezando con el aprendizaje automático, sus ramas supervisado y no supervisado, para después entrar al concepto de clusterización. Posterior a estos conceptos se describe el Procesamiento del Lenguaje Natural (NLP) y dos de sus ramas principales "aspect base sentiment analysis" y "sentiment analysis". Adicional se entró a la importancia del preprocesamiento en NLP y algunos de sus pasos principales. Por último, se presenta modelado de tópicos como tema general, los modelos utilizados en el desarrollo de la tesis y las métricas utilizadas para la comparación entre modelos.

3.1. Aprendizaje automático

El aprendizaje automático es un subcampo de la inteligencia artificial que construye sistemas que aprenden automáticamente, entendiendo aprendizaje como la identificación de patrones complejos en conjuntos masivos de datos [66]. Y automático como sistemas que mejoran con el uso y el acceso a más datos [67].

Los inicios del aprendizaje automático, en su sentido moderno, se remonta a 1957 cuando Frank Rossenblatt desarrolla el primer prototipo de las redes neuronales, llamándolo perceptrón. El perceptrón fue un programa basado en las ideas de cómo trabaja el sistema nervioso humano y la interacción entre neuronas [68].

El perceptrón de Rossenblatt generó altas expectativas que rápidamente fueron desechadas al encontrar varias limitantes en distintas aplicaciones. Esta frustración llevó a que la investigación y el desarrollo en las áreas de redes neuronales y aprendizaje automático se dejara prácticamente de lado hasta los inicios de los 90 [69].

En la década de los 90 el trabajo del aprendizaje automático dio un giro para orientarse en el aprovechamiento de la generación masiva de datos. Esto llevo a los investigadores, programadores y desarrolladores a trabajar en programas que permitieran analizar estos datos y aprender de ellos. A partir de estos programas se ha logrado tener un avance enorme en la rama del aprendizaje automático yendo desde algoritmos como IBM Deep Blue (1997), que fue el primer programa en vencer a un campeón mundial de ajedrez. Pasando por Microsoft Kinect (2010) una tecnología que podía traquear 20 rasgos humanos permitiendo a las personas interactuar con computadoras por medio de movimientos y gestos. En 2011 Google presento una red neuronal que podía encontrar e identificar objetos, en 2014 Facebook presentó DeepFace, un algoritmo de reconocimiento facial en fotografías de su plataforma [70].

Actualmente los algoritmos de aprendizaje automático se encuentran en una gran variedad de industrias y en un sinfín de aplicaciones, provocando que la inversión en la investigación y desarrollo de esta rama aumente, permitiendo que la tecnología avance y abra un mayor campo de aplicación.

3.1.1. Aprendizaje supervisado y no supervisado

El aprendizaje supervisado y el aprendizaje no supervisado son dos de las principales categorías de técnicas del aprendizaje automático, cada una con sus propias aplicaciones.

El aprendizaje supervisado consiste en entrenar un modelo utilizando datos etiquetados, en los que se proporcionan las respuestas correctas junto con los datos de entrada. El modelo utiliza los datos etiquetados para hacer predicciones o clasificaciones sobre datos no etiquetados. Este tipo de aprendizaje se utiliza habitualmente en aplicaciones como el

reconocimiento de imágenes y del habla, el procesamiento del lenguaje natural y los sistemas de recomendación [71]. Por ejemplo, en el reconocimiento de imágenes, se puede entrenar un modelo para reconocer imágenes de gatos a partir de un conjunto de datos de imágenes etiquetadas de gatos y no gatos. Una vez entrenado, el modelo puede utilizarse para clasificar imágenes no etiquetadas en gato y no gato.

El aprendizaje no supervisado, en cambio, consiste en entrenar un modelo a partir de datos no etiquetados, con el objetivo de encontrar patrones o estructuras en los datos. El modelo aprende a reconocer similitudes y diferencias entre distintos datos sin que tenga la respuesta correcta. Este tipo de aprendizaje se utiliza habitualmente en aplicaciones como la clusterización, la detección de anomalías y la compresión de datos [71] [72]. Por ejemplo, en la clusterización, se puede entrenar a un modelo para que agrupe a clientes similares en función de su historial de compras, sin saber nada sobre sus características demográficas o sus preferencias.

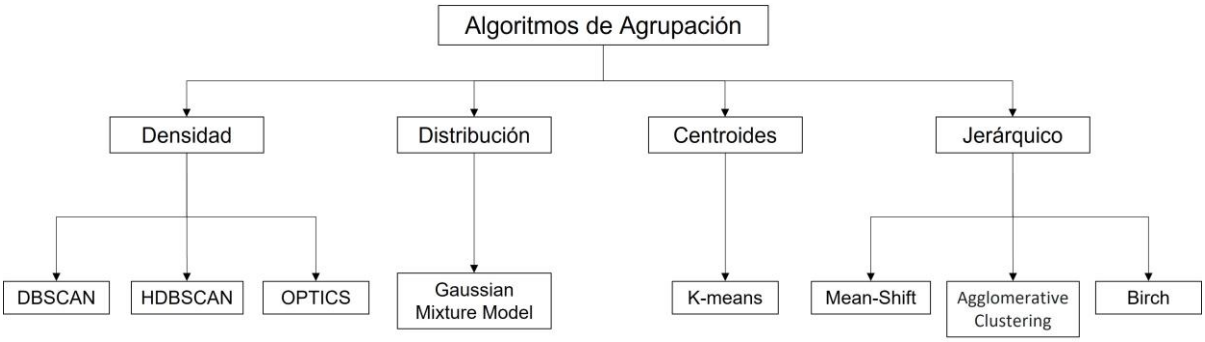
3.1.2. Clusterización

La clusterización es una técnica del aprendizaje automático no supervisado que consiste en agrupar datos similares en función de sus características. El objetivo de la clusterización es identificar patrones o estructuras en los datos sin que se les asignen etiquetas o categorías específicas [73].

En la clusterización, un modelo se entrena en un conjunto de datos sin etiquetar y debe aprender a reconocer similitudes y diferencias entre los puntos de datos. El modelo agrupa los puntos de datos en clústeres, y cada clúster representa un conjunto de puntos de datos similares [74].

En la figura 3.1 se muestran algunos ejemplos de algoritmos para clusterización, así como la métrica que utilizan para hacer la agrupación.

Figura 3.1 Ejemplos de algoritmos de clusterización.



Elaboración propia con información de: [75]

3.2. Procesamiento del Lenguaje Natural (NLP)

El Procesamiento del Lenguaje Natural (NLP) es un subcampo de la informática y la inteligencia artificial que se encarga de dar, a las computadoras, la capacidad de entender, interpretar y generar lenguajes humanos ya sean escritos o hablados [76] [77].

El NLP inicia su historia en la década de los 50, cuando los desarrolladores empezaron a buscar reducir la brecha de comunicación entre las computadoras y las personas. El primer gran avance que se tuvo fue en 1954 con el experimento de Georgetown, donde investigadores de IBM y de la universidad de Georgetown lograron desarrollar el primer sistema de traducción del ruso al inglés [78].

En los años 60, las investigaciones se centraron en la creación de sistemas capaces de entender consultas en lenguaje natural, lo que llevó al desarrollo del primer chatbot, llamado ELIZA, y al desarrollo de SHRDLU un sistema que lograba entender una serie de instrucciones escritas para realizar acciones en una interfaz con bloques de colores [78].

En las décadas de 1980 y 1990 aparecieron técnicas de NLP más avanzadas con la incorporación de los algoritmos de aprendizaje automático. Estas técnicas se utilizaron en muchas aplicaciones prácticas, como el reconocimiento del habla, el resumen automático y la clasificación de textos [79].

Actualmente, el Procesamiento del Lenguaje Natural es un campo en rápido crecimiento con una amplia gama de aplicaciones, incluyendo chatbots [80], asistentes de voz [81], análisis de sentimiento [82], traducción automática [83] y análisis de texto [84]. El desarrollo de algoritmos de aprendizaje profundo ha permitido la creación de sistemas de NLP más sofisticados que pueden comprender y generar respuestas cada vez más similares a las humanas.

3.2.1. Análisis de Sentimientos y Análisis de Sentimientos Basado en Aspectos (ABSA)

El Análisis de Sentimientos es una rama del Procesamiento del Lenguaje Natural que tiene como objetivo identificar y clasificar el tono emocional de un texto [82]. Esto permite comprender como se sienten las personas respecto a un tema, producto, servicio o marca [85].

Dada la complejidad del Análisis de Sentimientos esta rama se separa en tres principales procesos, que muchas veces terminan combinándose para obtener resultados. El primer proceso es la detección de sentimientos u opiniones el cual se basa principalmente en la detección de los adjetivos de los textos. El segundo proceso es la clasificación de la polaridad, que tiene como objetivo posicionar el texto, ya sea de forma binaria o en un continuo, de dos opuestos, pudiendo ser bueno-malo, positivo-negativo, etc. Por último, el tercer proceso es

la identificación del objetivo del texto, el cual puede tener distintos niveles de complejidad dependiendo el ámbito de análisis [86].

El Análisis de Sentimientos Basado en Aspectos (ABSA) es una técnica que realiza un análisis más profundo. Esta técnica determina el sentimiento de un texto referido a un aspecto en específico [87]. Los aspectos son los atributos o componentes de un producto o servicio [88]. Por ejemplo, en una reseña a un juguete se puede tener una opinión general positiva, pero mencionar que la pintura era de mala calidad. En este caso se podría identificar un sentimiento negativo hacia un aspecto (pintura) dentro de un sentimiento general positivo.

3.2.2. Preprocesamiento en NLP

El preprocesamiento de los datos en NLP es uno de los aspectos más importantes al momento de querer implementar cualquier tipo de modelo. Un buen preprocesamiento asegura que el texto a analizar esté en un formato y con características que permita al modelo implementado entenderlo y analizarlo.

Al no ser un proceso estándar para todas las implementaciones de NLP, existen un sinfín de métodos que se pueden realizar individualmente o en conjunto para llegar al resultado deseado. Algunos de los principales métodos son.

Tokenizado, es un método que consiste en partir un texto en trozos más pequeños, lo más común es partir en palabras individuales, pero también es posible hacerlo en conjuntos de un número específico de palabras o en oraciones más pequeñas que el texto original [89]. Un ejemplo, sería la oración “¡Hola mundo!” Al tokenizarla en palabras individuales quedaría como “¡”, “Hola”, “mundo”, “!”.

Stopwords o palabras vacías, es un método que consiste en retirar del texto aquellas palabras que no aportan mucha información al texto [90]. Algunos ejemplos son los artículos, preposiciones, pronombres, conjugaciones, etc. Un ejemplo, sería la oración “El perro tiene un juguete azul” al retirar las stopwords quedaría como “perro tiene juguete azul”.

Stemming, es un método que consiste en remover una parte de las palabras para llevarlas a su raíz, esto con el objetivo de disminuir el número de palabras distintas que podrían tener un significado similar. Al realizar este procedimiento no aseguramos que la raíz sea una palabra que exista y, por lo tanto, podemos llegar a tener palabras que llegan a una misma raíz sin necesariamente tener significados similares [91]. Un ejemplo serían las palabras “ferrocarril”, “ferroviario”, “ferroso” se reducirían a “ferro”.

3.3. Modelado de tópicos

El modelado de tópicos es una técnica de aprendizaje no supervisado del procesamiento de lenguaje natural [92]. Es una técnica que consiste en identificar los tópicos que se encuentran

en un conjunto de textos. Esta identificación es realizada analizando los patrones en la distribución y frecuencia de las palabras en los textos, y agrupándolas en clústeres, los cuales a su vez pueden ser categorizados como tópicos [1].

El modelado de tópicos tiene distintas aplicaciones, como la clasificación, categorización, y síntesis de textos [93]. Puede utilizarse para analizar publicaciones en redes sociales, opiniones de clientes, artículos científicos, noticias y otros tipos textos. Algunos de los algoritmos que se utilizan son la Asignación Latente de Dirichlet (LDA), el Mezcla de Muestreo Gibbs Dirichlet Multinomial (GSDMM) y el BERTopic.

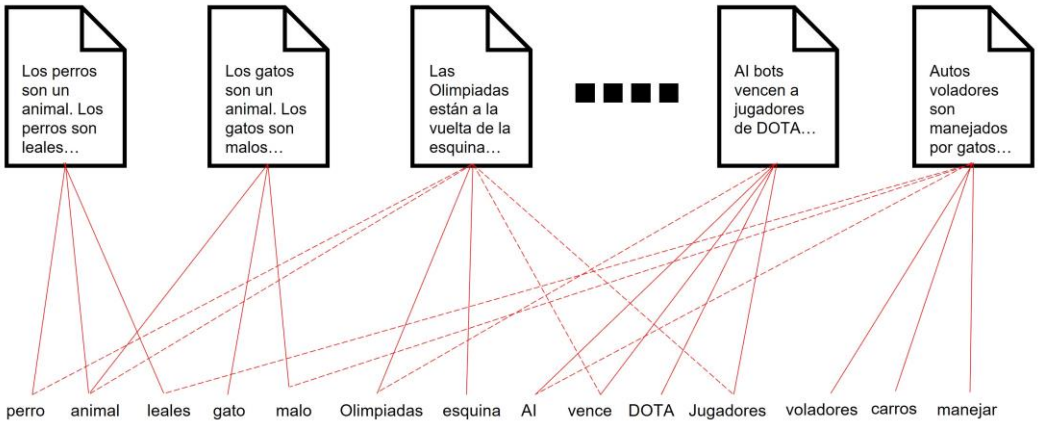
3.3.1. Asignación Latente de Dirichlet (LDA)

El modelo de LDA (Asignación Latente de Dirichlet) es un modelo de tópicos generativo desarrollado en 2003 por David Blei, Andrew Ng y Michael Jordan [5]. Este modelo funciona sobre la idea de que cada documento se construye de una mezcla de tópicos, y que estos tópicos se forman de una lista de palabras con cierta probabilidad de pertenecer a cada tópico en específico, estas probabilidades son generadas a partir de una distribución de Dirichlet [49], y con estas probabilidades podemos encontrar para cada documento de qué tópicos está formado y en qué porcentaje [94].

LDA es un modelo no supervisado por lo que una parte importante para conseguir resultados funcionales es la definición de los parámetros con los que iniciará. El parámetro más importante es el número de tópicos en los cuales se clasificarán los documentos, y los parámetros a definir son el Alpha y Beta de la distribución de Dirichlet. Alpha controla la distribución de tópicos por documento y Beta la distribución de palabras por tópico. Esto nos dice que una Alpha alta refiere a que cada documento tiene una mezcla de más número de tópicos y una Alpha pequeña que cada documento se compone de menor número de tópicos, y una Beta alta refiere a que cada tópico tendrá una mezcla de más palabras lo que nos daría tópicos más similares [95].

La lógica con la que trabaja LDA se puede entender de la siguiente manera. En la figura 3.2 tenemos un conjunto de textos los cuales están compuestos de un conjunto de palabras. Podemos tomar todo el listado de palabras y relacionarlas a los textos en los que aparecen.

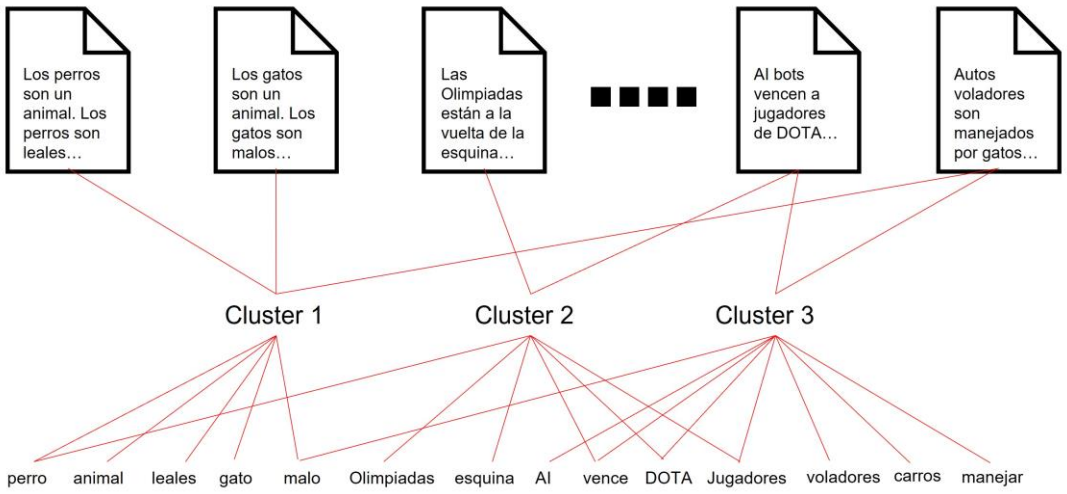
Figura 3.2 Asignación de las palabras a su documento.



Tomada y traducida de: [95]

Una vez que se tiene esta relación palabras-textos, a partir de los patrones de distribución y frecuencia es posible crear clústeres de las palabras y así, si una palabra que aparecía en el texto 1 está ligada al clúster 1 entonces el texto 1 estará ligado al clúster 1 pero si además en el texto 1 aparecen palabras del clúster 2, el texto 1 estará ligado tanto al clúster 1 como al 2. Dependiendo del número de palabras y la frecuencia de estas cada clúster tendrá una mayor probabilidad dentro del texto 1, reafirmando el planteamiento principal de LDA que dice que todos los textos son una mezcla de distintos tópicos. Este proceso se observa en la figura 3.3.

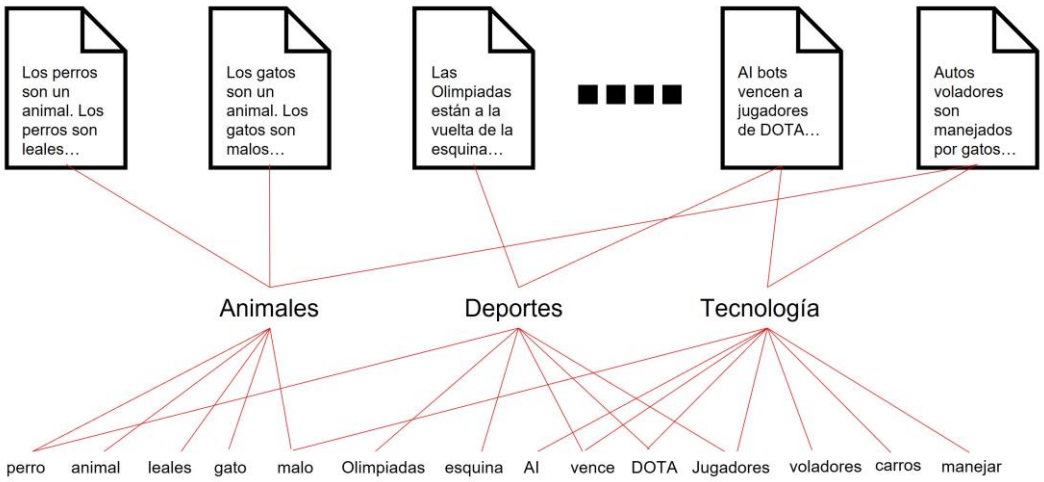
Figura 3.3 Generación de clústeres.



Tomada y traducida de: [95]

Por último, solo quedaría asignar un tópico a cada clúster, para lograr esto es necesario que se haga una inspección visual a los conjuntos de palabras que componen cada clúster. Con esto ya es posible etiquetar cada clúster y, por lo tanto, realizar un análisis, de forma más sencilla, de los textos. Este proceso se muestra en la figura 3.4.

Figura 3.4 Definición de tópicos.



Tomada y traducida de: [95]

3.3.2. Mezcla de Muestreo Gibbs Dirichlet Multinomial (GSDMM)

El modelo GSDMM (Mezcla de Muestreo Gibbs Dirichlet Multinomial) fue desarrollado por Jianhua Yin y Jianyong Wang en 2014 [96]. Es un modelo que se basa en la teoría de LDA, pero se plantearon ciertos cambios para convertirlo en un modelo orientado a textos cortos. Las principales diferencias son que, a diferencia de LDA, GSDMM funciona bajo la lógica que cada documento únicamente se puede clasificar en un tópico y que GSDMM no es necesario plantear un número de tópicos objetivo, sino que se da el parámetro de tópicos máximos [94]. Adicional a los tópicos máximos este modelo tiene por parámetros alpha y beta de la distribución de Dirichlet al igual que LDA.

El proceso con el que trabaja GSDMM se puede entender con la analogía de Movie Group Process. Esta analogía nos plantea un grupo de estudiantes a los cuales su maestro les pide escribir en una hoja películas que hayan visto, después los estudiantes se sientan aleatoriamente en diferentes mesas formando unos grupos iniciales. Una vez armados estos primeros grupos se pasa a que, uno a uno, cada estudiante elige una nueva mesa considerando los dos siguientes criterios:

- Elegir una mesa que tenga más estudiantes que en la que está actualmente.
- Elegir una mesa donde los demás estudiantes tengan películas similares a las que escribió.

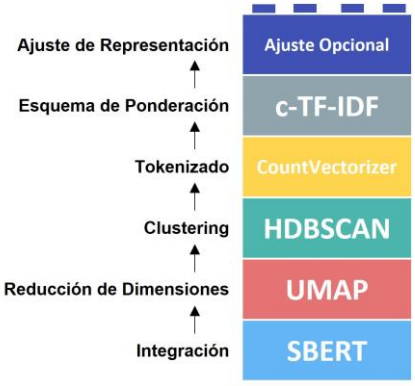
Con estas dos reglas cada estudiante se va reubicando y así algunas mesas van creciendo y otras reduciéndose, inclusive llegando a desaparecer. Este procedimiento de cambiar de mesa se puede repetir las veces necesarias hasta llegar a un número de mesas óptimo y que los estudiantes que se encuentren en ellas tengan similitud en sus listas de películas [97].

3.3.3. BERTopic

El algoritmo de BERTopic es un modelo de tópicos desarrollado en 2022 por Maarten Grootendorst [10]. Este algoritmo está construido como una secuencia de pasos, con cinco pasos principales más uno opcional. Estos pasos son Integración (Embeddings), Reducción de Dimensiones (Dimensionality Reduction), Clusterización Tokenizado, Esquema de ponderación (Weighting Scheme) y el opcional, ajuste de representación (Representation Tunning) [53].

Por la forma en que está construido BERTopic permite que cada uno de sus pasos funcionen como módulos independientes, esto provoca que cada paso tenga una variedad de opciones de submodelos para realizarlos y permitiendo construir distintos modelos de tópicos a partir de la estructura de BERTopic [53]. En este trabajo revisaremos las opciones predeterminadas de los cinco pasos principales, que tiene BERTopic, estas opciones se muestran en la figura 3.5.

Figura 3.5 Pasos de BERTopic



Tomada y traducida de: [53]

La integración de documentos, el primer paso, consiste convertir los textos en representaciones numéricas, para esto se utiliza el método de transformadores de oraciones [98] y más específicamente “all-MiniLM-L6-v2” [98] el cual es un modelo de idioma inglés entrenado específicamente para tareas de similitud semántica.

El segundo paso es la reducción de dimensiones, que como su nombre lo indica consiste en reducir las dimensiones de las representaciones numéricas creadas en el paso anterior. Para este paso el modelo predeterminado es UMAP [99]. Este es un modelo que permite conservar parte de la estructura local y global del conjunto de datos al reducir su dimensionalidad, lo cual es importante ya que estas estructuras contienen información necesaria para crear clústeres de documentos semánticamente similares [53].

El tercer paso, la clusterización, se realiza con HDBSCAN [100], la cual es una técnica de agrupamiento basada en la densidad. Esto quiere decir que permite encontrar clústeres con

diferentes formas, además de detectar valores atípicos de buena manera, logrando no forzar documentos en clústeres a los que no pertenecen [53].
El tokenizado, cuarto paso, consiste en crear una bolsa de palabras (Bag-of-words) por clúster con esto se obtiene que palabras conforman cada clúster y su número de apariciones. Es importante resaltar que la bolsa de palabras es a nivel de clúster y no de documento porque esto permite que no se hagan suposiciones sobre la estructura de los clústeres [53]. Para crear estas bolsas de palabras se utiliza el método de CountVectorizer [101].

El último paso principal es el esquema de ponderación. En este paso se quiere saber, a partir de las bolsas de palabras, que hace diferente un clúster de otro. Para esto se utiliza una variación de TF-IDF nombrada como c-TF-IDF o TF-IDF basado en clases [102]. TF-IDF es una medida numérica que nos permite comparar la importancia de las palabras entre documentos. Con c-TF-IDF lo que se hace es tomar todos los documentos, como un documento único, en un solo clúster y después aplicar TF-IDF, al realizar este proceso lo que se obtiene es un puntaje de importancia para las palabras dentro de un clúster. Cuantas más palabras importantes hay dentro de un clúster, más representativo es de ese tópico. En otras palabras, si extraemos las palabras más importantes por clúster, obtenemos la descripción de tópicos [53].

3.4. Métricas

En este trabajo se utilizan distintos modelos para el modelado de tópicos y, por lo tanto, es importante poder evaluarlos y compararlos, para esto se plantearon tres métricas, dos cuantitativas y una cualitativa. La métrica cualitativa es un análisis visual de los resultados y las métricas cuantitativas son el tiempo de ejecución de cada modelo y una puntuación de coherencia, la cual es una métrica utilizada ampliamente para calificar el desempeño de los modelos de tópicos.

3.4.1. Análisis visual

El modelado de tópicos es un algoritmo matemático y si bien los tópicos son matemáticamente óptimos, para el ojo humano pueden no ser tan buenos. Un ejemplo sería tener dos tópicos, el primero formado por las palabras (llantas, volante, pedales, ventana) y el segundo con las palabras (pelota, enfermera, ladrillo, supermercado). Para el modelo ambos tópicos pueden ser igualmente correctos, pero desde la perspectiva humana el primer tópico tiene mayor sentido que el segundo [103].

Para evitar que se tomaran como correctos todos los resultados se planteó, para este trabajo, un sistema de calificación de los resultados obtenidos por los modelos, en el cual era necesario hacer un análisis visual, en una nube de palabras, de las principales palabras obtenidas para cada uno de los tópicos generados por un modelo. El sistema de calificación se describe en la tabla 3.1.

Tabla 3.1 Sistema de calificación para el análisis visual.

Calificación	Descripción
Mejor	La mayoría de las principales palabras son diferentes entre clústeres y además es posible definir un tópico a cada clúster.
Medio	La mayoría de las principales palabras son diferentes entre clústeres, pero no es posible definir un tópico a cada clúster.
Peor	La mayoría de las principales se repiten entre clústeres y no es posible definir un tópico a cada clúster.

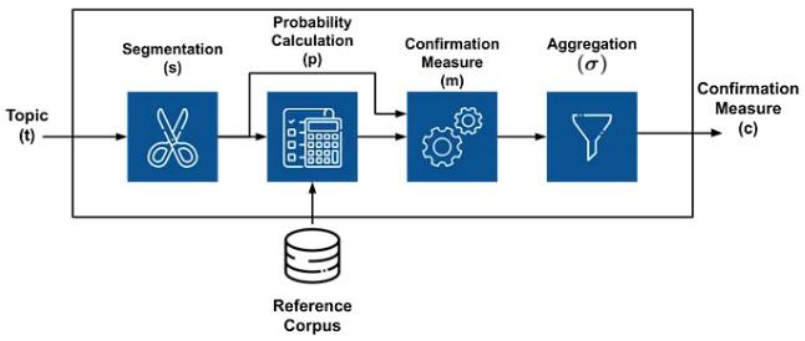
3.4.2. Tiempo de ejecución

El tiempo de ejecución es el tiempo que le toma en ejecutarse por completo a cada modelo. Para esta métrica se utilizó la librería de time en Python que nos permite registrar la hora en la que se inicia a correr el modelo y la hora en la que termina, con estos dos datos es posible hacer una diferencia y tener el tiempo de ejecución. Para esta métrica es importante considerar las características técnicas del equipo en el que se ejecutan los modelos y tener la precaución de correr todos los modelos en el mismo equipo para que la métrica sea comparable.

3.4.3. Coherencia

La coherencia, en el modelado de tópicos, es una métrica que evalúa que tan bien un tópico está "respaldado" por un conjunto de texto (llamado corpus de referencia). Utiliza estadísticas y probabilidades extraídas del corpus de referencia, especialmente enfocadas en el contexto de la palabra, para dar una puntuación de coherencia a un tópico. Para lograr este cálculo de coherencia se plantea una estructura general que seguir y la cual se compone de cuatro principales módulos, esta estructura se muestra en la figura 3.6 [103].

Figura 3.6 Estructura general del cálculo de coherencia.



Tomada de: [103]

El primer módulo es la segmentación, que consiste en crear pares de subconjuntos de palabras para evaluar la coherencia del tema. De forma sencilla, en este módulo lo que se hace es elegir la forma en que se van a mezclar las palabras, existen diferentes formas de hacer la segmentación, una de ellas es la segmentación S-one-one que nos dice que tenemos que hacer pares con de diferentes palabras. Por ejemplo, si nuestro conjunto de palabras es [perro, gato, juguete], nuestros pares quedarán como [(perro, gato), (perro, juguete), (gato, perro), (gato, juguete), (juguete, perro), (juguete, gato)] [103].

El segundo módulo es el cálculo de la probabilidad, en este módulo se define que técnica se utilizará para hacer el cálculo, al igual que la segmentación existe una gran variedad de técnicas, una de ellas es la Probabilidad de documento booleano (Pbd) en la cual se calcula la probabilidad de que aparezca una palabra P(w) como el número de documentos en los que aparece la palabra, dividido entre el total de documentos. Esta misma probabilidad se puede utilizar para conjuntos de palabras P(w1, w2) donde se toma el número de documentos en los que aparecen ambas palabras, dividido entre el total de documentos [103].

El tercer módulo, la medida de confirmación, lo que hace es calcular esta medida de confirmación sobre los pares obtenidos en la segmentación y con las probabilidades obtenidas en el segundo módulo. Es decir, este módulo intenta cuantificar la relación entre las dos palabras de cada subconjunto. Existen distintas medidas de confirmación, y en las ecuaciones 1 y 2 se muestran dos ejemplos de estas [103].

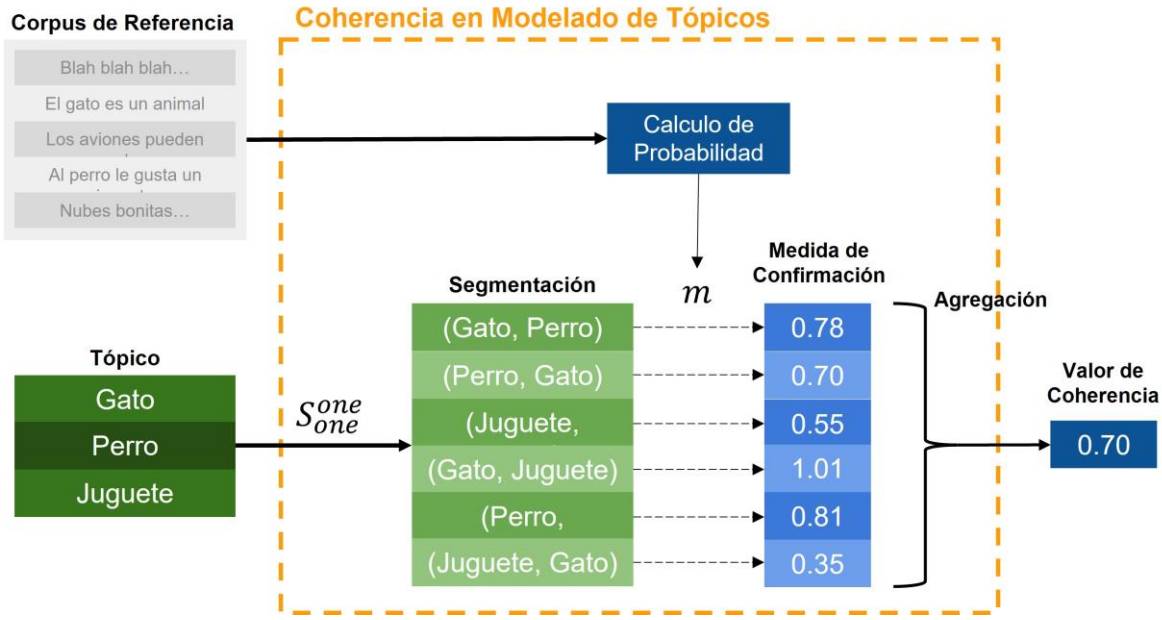
$$m_r(S_i) = \frac{P(w_1,w_2)}{P(w_1)P(w_2)} \quad (1)$$

$$m_c(S_i) = \frac{P(w_1,w_2)}{P(w_2)} \quad (2)$$

El último módulo es la agregación, donde lo único que se hace es tomar todos los valores obtenidos para cada conjunto de palabras, y se agregan en un solo valor el cual es el valor de coherencia del tópico. Y tomando el valor de coherencia de cada tópico y agregando a un único valor, obtenemos el valor de coherencia del modelo. La agregación se puede hacer por medio de la media aritmética, la mediana, la media, etc. [103]

A modo de resumen la figura 3.7 presenta los cuatro módulos que componen el cálculo de coherencia. Donde tenemos la entrada de un corpus con el conjunto de textos y un conjunto de palabras que forman un tópico. Después hacemos una segmentación de estas palabras uniéndolas en pares de palabras y se calculan sus probabilidades de aparición en el corpus. Una vez teniendo estas probabilidades se calculan las medidas de confirmación para cada subconjunto creado en la segmentación y por último se hace una agregación de estas medidas de confirmación y se tiene el valor de coherencia del modelo general.

Figura 3.7 Proceso de cálculo de la coherencia



Tomada y traducida de: [103]

El valor de coherencia es el que nos indica si nuestro resultado es bueno o malo, al existir distintas formas de calcular la medida de confirmación, el rango de valores que puede tomar la coherencia cambia. Para la medida de confirmación utilizada en el desarrollo de la tesis se tiene un rango entre 0 y 1. Para poder definir que un valor de coherencia es bueno no existe una regla como tal, sino que lo que se busca es maximizar este valor en nuestro conjunto de documentos sin llegar a los extremos, tener valor cercanos a 0 se podría interpretar como resultados malos pero el tener valores cercanos al 1 también se interpreta como resultados malos. Lo óptimo es tener valores en la zona media de nuestro rango y maximizarlo. También considerando que los valores de coherencia siempre deben de ir acompañados de otras métricas para analizar los resultados.

4. DESARROLLO METODOLÓGICO

En este capítulo se presenta la metodología que se siguió en el desarrollo de la tesis. Primero se hizo la selección del conjunto de datos a analizar, explicado a detalle en la sección "Base de datos original". Después de la base de datos original se eliminó información que no es relevante para el proyecto de tesis y se preproceso el texto de las reseñas, esto se describe en la sección "Preprocesamiento". En la sección de "Base de datos procesada" se describe la base de datos que resulta del preprocesamiento. En la sección "Implementación de modelos" se detalla la metodología seguida para la implementación de los tres modelos evaluados, a pesar de ser tres modelos con el mismo propósito: crear una clusterización basada en aspectos, cada uno tiene requerimientos específicos, los cuales se detallan en esta sección. En la Figura 4.1. se muestra un diagrama de la metodología general utilizada. Por último, en la sección "Características técnicas y librerías" se dan detalles del equipo de cómputo utilizado y las librerías para llevar a cabo la implementación de los procedimientos descritos en este capítulo.

Figura 4.1. Diagrama de metodología general.

4.1. Base de datos original

La base de datos original se obtuvo de la plataforma “Inside Airbnb” [104], esta plataforma es un proyecto que proporciona datos y promociona sobre el impacto de Airbnb en las comunidades residenciales. Buscan que los datos e información proporcionada capaciten a las comunidades para comprender, decidir y controlar el papel del alquiler de viviendas residenciales a turistas.

La base de datos, seleccionada para este trabajo, consiste en una recopilación de las reseñas recibidas por las propiedades dadas de alta en Airbnb, en Berlín, entre los años 2010 y 2019, así como 47 otras variables como características del hospedaje, dueño del inmueble y usuarios. Estas variables sirven para contextualizar cada una de las reseñas. La base cuenta con 456,977 registros.

4.2. Preprocesamiento

El preprocesamiento de los datos se llevó a cabo en 11 etapas consecutivas, estas etapas se muestran en la Figura 4.2.

Figura 4.2. Diagrama del preprocesamiento.

La primera etapa del preprocesamiento fue la selección de columnas, aquí se hizo una selección de aquellas variables que eran potencialmente utilizables durante el desarrollo del proyecto, se descartaron aquellas variables que eran redundantes o que no se consideraban

utilizar en posteriores pasos del trabajo. En la segunda etapa se eliminaron todos los registros que tuvieran cuando menos un valor nulo en las variables seleccionadas en la primera etapa.

La tercera etapa consistió en la detección del idioma en la que estaba escrita la reseña correspondiente a cada registro. Este proceso, por cuestión de velocidad de procesamiento, se realizó en R a diferencia del resto de las etapas que se realizaron en Python 3.9.12. Se detectaron 35 idiomas distintos, siendo alemán, inglés, español y francés los idiomas con mayor número de reseñas. En esta etapa también se creó una columna nueva en la que se especificaba el idioma original de la reseña.
Para la cuarta etapa se homologaron todas las reseñas en un mismo idioma, el idioma seleccionado para trabajar fue el inglés. El proceso de traducción se realizó empleando la librería Googletrans [105], la cual emplea la API Google Translate en su versión gratuita por lo que se tuvo que realizar en lotes pequeños de reseñas.

Una vez homologado el idioma en las reseñas en la quinta etapa, se removieron los signos de puntuación dejando únicamente caracteres alfanuméricos. En la sexta etapa se pasaron todas las reseñas a minúsculas buscando una mejor homologación. Posterior a esto en la séptima etapa se hizo un tokenizado de todas las reseñas para facilitar la identificación de palabras individuales en siguientes etapas del preprocesamiento.

En la octava etapa, de las reseñas tokenizadas se eliminaron las stopwords o palabras vacías, que son aquellas palabras sin un significado por sí solas como son los artículos, pronombres, preposiciones, etc. Para hacer la eliminación se tomó una lista ya preconstruida en la librería utilizada. Una vez eliminadas estas palabras se pasó a la novena etapa en la que se realizó un proceso de stemming el cual consiste en llevar las palabras a su palabra raíz, por ejemplo, las palabras auto, automóvil, automovilista todas quedarían en la palabra auto. Esto nos permite reducir el vocabulario y simplificar el conjunto de textos.

En la décima etapa se eliminaron todos aquellos registros, que después de las nueve anteriores etapas, tuvieran menos de dos palabras o tokens en sus reseñas, ya que se consideró que al tener tan poco texto no eran registros que pudieran contribuir en el desarrollo de los modelos.

Por último, en la onceava etapa se pasaron los tokens a oraciones, esto con el objetivo de facilitar la implementación de uno de los modelos utilizados en el desarrollo del proyecto.

4.3. Base de datos procesada

Una vez realizado el preprocesamiento, la base de datos quedó compuesta por 381,095 registros (83% de la original) y 22 variables, incluyendo las reseñas. En estas 22 variables contamos con la información que se consideró podría ser relevante para el desarrollo del proyecto, estas variables, además de las reseñas, cuentan con información del alojamiento, dueño del alojamiento, las reseñas y quien renta.

De las 22 variables seleccionadas en el preprocesamiento, durante la obtención de resultados se utilizaron las siete variables que se consideró pudieran mostrar un comportamiento relevante de la relación de sus características y la distribución de los tópicos identificados. Estas siete variables se detallan en la Tabla 4.1.

Tabla 4.1. Variables utilizadas y su descripción

Variable	Descripción	Tipo de dato
Comments	Reseñas originales	String
Is_superhost	Si el anfitrión tiene la categoría de "superhost". Para lograr esta categoría el anfitrión debe cumplir una calificación general de 4.8 o más, más de 10 estancias diferentes o 100 noches en 3 estancias, un índice de cancelación menor al 1% y un índice de respuesta de 90% o más. Todo esto en el último año cada tres meses.	Binario
Overall_rating	Calificación general de la estancia (0 a 100)	Integer
Property_type	El tipo de propiedad que es el inmueble (departamento, casa, dúplex, cuarto, etc.).	String
idioma	Idioma de la reseña original (en, de, fr, es, etc.).	String
stemmed	Las reseñas tokenizadas después del preprocesamiento	String
Clean_sentence	Las reseñas después del preprocesamiento reconvertidas en oración.	String

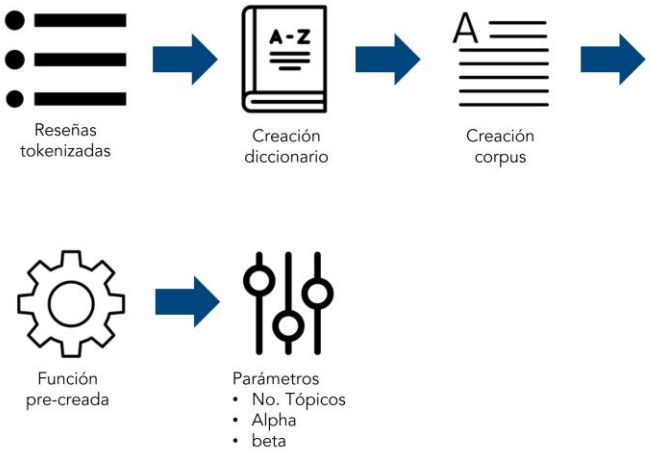
4.4. Implementación de modelos

En esta sección se presenta la metodología seguida para la implementación de los tres modelos que se utilizaron durante la elaboración de la tesis.

4.4.1. Asignación Latente de Dirichlet (LDA)

El primer modelo implementado fue el LDA, en la Figura 4.3. se presenta la metodología general que se siguió.

Figura 4.3. Metodología de implementación de LDA.

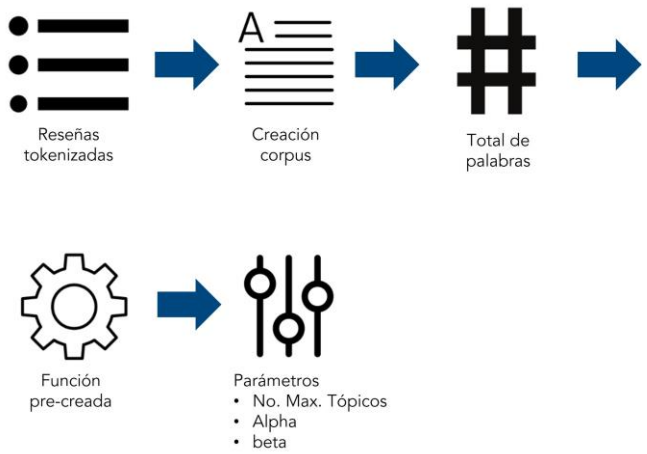


El primer paso de la implementación consistió en tomar las reseñas tokenizadas ya preprocesadas, a partir de estas reseñas se creó un diccionario con el total de palabras de todo el conjunto. Una vez creado el diccionario se creó un corpus donde se incluyen todas las distintas palabras del diccionario y el número de repeticiones que tiene en todo el conjunto de reseñas. Ya creado el corpus se utilizó como insumo para alimentar la función de LdaMulticore de la librería gensim. Esta función se utilizó para aplicar el modelo tomando en cuenta los parámetros del número de tópicos, alpha y beta.

4.4.2. Mezcla de Muestreo Gibbs Dirichlet Multinomial (GSDMM)

GSDMM fue el segundo modelo implementado y la metodología que se siguió se presenta en la Figura 4.4.

Figura 4.4. Metodología de implementación de GSDMM.



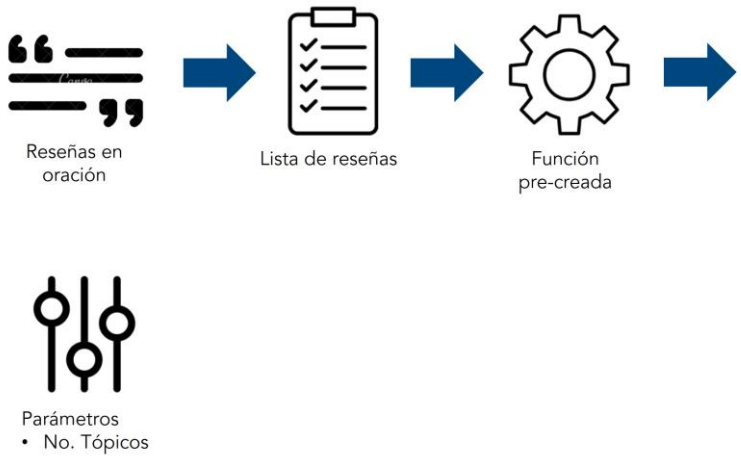
Para la implementación de este modelo se tomaron las reseñas tokenizadas ya preprocesadas, posterior a eso se creó un corpus con todas las palabras distintas y su número

de repeticiones del total de reseñas. Teniendo el corpus se obtuvo el total de palabras distintas que tiene el conjunto de reseñas. Posterior a esto se utilizó el corpus y el total de palabras para alimentar la función de MovieGroupProcess de la librería de gsdmm en Python. Esta función se utilizó para aplicar el modelo tomando en cuenta los parámetros del número máximo de tópicos, alpha y beta.

4.4.3. BERTopic

Por último, el modelo que se implementó fue el de BERTopic y en la Figura 4.5. podemos observar la metodología seguida.

Figura 4.5. Metodología de implementación de BERTopic.



Para la implementación de BERTopic se tomaron las reseñas preprocesadas regresadas a oraciones. Con estas reseñas se creó una lista que fue la información que se alimentó a la función BERTopic. Para este modelo se consideró el hiperparámetro del número de tópicos, dejando el resto con los valores predeterminados.

4.5. Características técnicas y librerías

En esta sección se presentan las características técnicas del equipo de cómputo empleado (Tabla 2) para el desarrollo de la tesis, así como un listado de las librerías utilizadas. El listado se encuentra en la Tabla 4.3., donde además se presenta la función utilizada y el uso que tuvo en el desarrollo metodológico.

4.5.1. Características técnicas

Tabla 4.1. Características técnicas del equipo empleado

Fabricante del sistema	Acer
Modelo del sistema	Spin SP314-53N
Sistema operativo	Microsoft Windows 10 Home
Tipo de sistema	X64 based PC
Procesador	Intel® Core™ i7-8565U CPU @ 1.80GHz, 1992 Mhz, 4 procesadores principales, 8 procesadores lógicos
Memoria física instalada (RAM)	16 GB

4.5.2. Librerías

Tabla 4.2. Librerías utilizadas en el desarrollo metodológico.

librería	**Función**	**Uso**
Pandas	.dropna()	Eliminar registros nulos (Etapa 2 del preprocesamiento)
time	.time()	Medir el tiempo de ejecución
cld2 (Compact Language Detector 2) * [106]	detect_language()	Detectar el idioma (Etapa 3 del preprocesamiento)
googletrans	Translator()	Traducción de las reseñas (Etapa 4 del preprocesamiento)
string	.punctuation	Remover signos de puntuación (Etapa 5 del preprocesamiento)
	.lower()**	Pasar las reseñas a minúsculas (Etapa 6 del preprocesamiento)
NLTK [107]	word_tokenize	Tokenizado de las reseñas (Etapa 7 del preprocesamiento)
NLTK	stopwords	Lista de stopwords para remover de las reseñas (Etapa 8 del preprocesamiento)
NLTK	Snowballstemmer()	Proceso de stemming (Etapa 9 del preprocesamiento)
NLTK	TreebankWordDetokenizer()	Pasar de tokens a oración las reseñas (Etapa 11 del preprocesamiento)
gensim [108]	models.LdaMulticore()	Implementación del modelo LDA
gensim	CoherenceModel()	Cálculo del score de coherencia
gsdmm [109]	MovieGroupProcess()	Implementación del modelo GSDMM
BERtopic [10]	BERTopic.fit.transform()	Implementación del modelo BERTopic

* Es un paquete de R.

** Es un método incorporado en Python.

5. RESULTADOS Y DISCUSIÓN

En este capítulo se presentan los resultados obtenidos de la comparativa entre modelos, los valores obtenidos en sus métricas y la selección del mejor modelo para el conjunto de datos. De igual forma se presentan los resultados de la optimización del modelo seleccionado, los tópicos encontrados y el conjunto de principales palabras que componen a cada uno de estos tópicos. También se presenta un análisis y discusión de los resultados obtenidos.

5.1. Resultados

5.1.1. Selección de Modelo

Para la selección del modelo que se utilizó en el desarrollo del trabajo, se implementaron los modelos de GSDMM, LDA y BERTopic. El primer modelo que se implementó fue el de GSDMM, el cual permite definir un número máximo de clústeres y no un número exacto, con esta ventaja, y con los valores predeterminados de los parámetros, se encontró que el modelo tendía a agrupar el conjunto de reseñas en 16 clústeres. Con este dato se hizo la implementación de los modelos LDA y BERTopic dejando los valores predeterminados de cada modelo y con un número de clústeres de 16, esto para poder comparar los tres modelos en las tres distintas métricas.

Para determinar la métrica del análisis visual se crearon nubes de palabras, para cada modelo, de los cuatro clústeres con mayor número de reseñas asignadas. En las figuras 5.1., 5.2., 5.3. y 5.4. podemos observar los clústeres obtenidos para el modelo LDA.

Figura 5.1. Clúster 0 de LDA.

Figura 5.2. Clúster 2 de LDA.

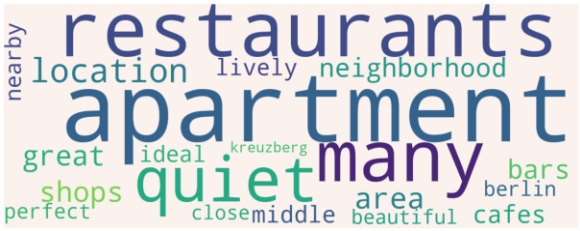
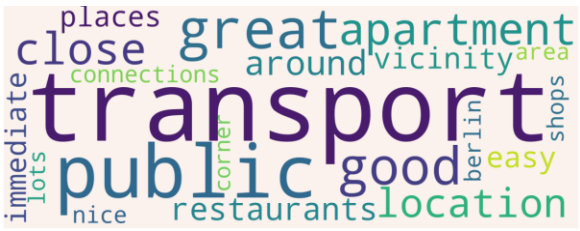


Figura 5.3. Clúster 3 de LDA.

Figura 5.4. Clúster 4 de LDA.



En las figuras 5.5., 5.6., 5.7. y 5.8. tenemos las nubes de palabras del modelo de GSDMM y en las figuras 5.9., 5.10., 5.11. y 5.12. las del modelo de BERTopic.

Figura 5.5. Clúster 15 de GSDMM.

Figura 5.6. Clúster 14 de GSDMM.

Figura 5.7. Clúster 13 de GSDMM.

Figura 5.8. Clúster 7 de GSDMM.

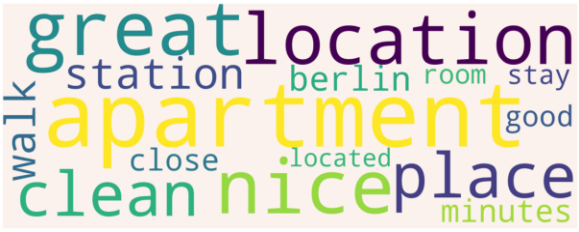


Figura 5.9. Clúster 0 de BERTopic.

Figura 5.10. Clúster 1 de BERTopic.

Figura 5.11. Clúster 2 de BERTopic.

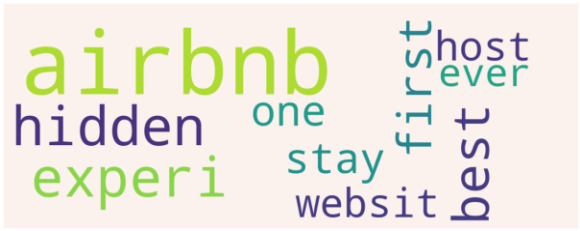


Figura 5.12. Clúster 3 de BERTopic.

En las figuras 5.1., 5.2., 5.3. y 5.4. se observó que las palabras que componen cada uno de los clústeres se separaron de forma que es posible identificarlas de manera única en cada clúster, y además era posible asignarle un tópico a cada uno de los clústeres. Por estas características en la métrica de análisis visual se le asignó a LDA la calificación de “Mejor”. En las figuras 5.5., 5.6., 5.7. y 5.8., donde tenemos las nubes de palabras del modelo GSDMM, se observó que la mayoría de las palabras se repiten entre los clústeres por lo cual fue imposible asignarle un tópico a cada clúster, ante esto en análisis visual se le asignó “Peor”. Al modelo de BERTopic en análisis visual se le asignó “Medio”, ya que las nubes de palabras mostraron una separación clara de las palabras, pero fue difícil encontrar una relación entre las palabras que componían cada clúster por lo que no se les pudo asignar un tópico, estas nubes de palabras las encontramos en las figuras 5.9., 5.10., 5.11. y 5.12.

Los resultados obtenidos en las tres métricas tomadas en cuenta para la selección del modelo se muestran en la tabla 5.1.

Tabla 5.1. Métricas de los tres modelos

Modelo	Coherencia	Tiempo ejecución	Análisis visual
BERTopic	0.4201	1hr 54min	Medio
GSDMM	0.3397	2hrs 13min	Peor
LDA	0.4343	13min	Mejor

Con los resultados de las métricas para cada modelo que se muestran en la Tabla 5.1. Se concluyó que el modelo a utilizar en el resto del desarrollo del trabajo era LDA. Este modelo se seleccionó, ya que es el que tiene mejores resultados en las tres métricas consideradas, con el valor más alto en coherencia, el menor tiempo de ejecución y el mejor análisis visual. Siendo el tiempo de ejecución la métrica con mayor peso por la diferencia entre LDA y los otros dos modelos.

5.1.2. Optimización de parámetros

Con el modelo LDA seleccionado se pasó a la etapa de optimizar los parámetros del modelo (número de tópicos, Alpha y Beta). Para esto se realizó un barrido de todas las combinaciones posibles con los rangos de valores, mostrados en la tabla 5.2., para cada parámetro.

Tabla 3.2. Rangos de valores para barrido de parámetros.

Parámetro	Rango	Aumento	Adicionales
Número de tópicos	2 a 20	1	NA
Alpha	0.1 a 1	0.3	Symmetric y Asymmetric
Beta	0.1 a 1	0.3	Symmetric

El barrido se realizó con el 10% de la base de datos total (38,109 registros) y se calculó la coherencia de cada una de las combinaciones para poder compararlas. Con el barrido realizado se tomaron las 20 combinaciones con mayor valor de coherencia y se implementaron con la base de datos completa para hacer un análisis visual de los tópicos encontrados por cada combinación. La combinación de parámetros que tuvo el mejor desempeño con un valor de coherencia de 0.4164, se muestra en la tabla 5.3.

Tabla 5.3. Valores de parámetros.

Número de tópicos	5
Alpha	Asymmetric
Beta	0.61

5.1.3. Análisis de datos

5.1.3.1. Análisis general

Con los parámetros optimizados se corrió un modelo para la base de datos completa y se obtuvieron los 5 clústeres de palabras que se muestran en las figuras 5.13., 5.14., 5.15., 5.16. y 5.17.

Figura 5.13. Clúster 1 de modelo general LDA.

Figura 5.14. Clúster 2 de modelo general LDA.

Figura 5.15. Clúster 3 de modelo general LDA.

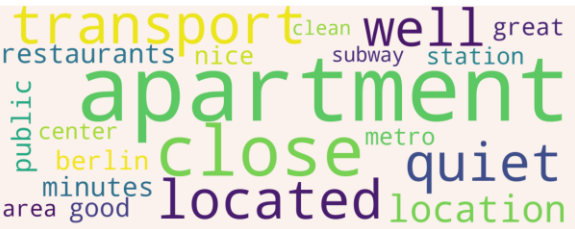


Figura 5.16. Clúster 4 de modelo general LDA.

Figura 5.17. Clúster 5 de modelo general LDA.

Se analizaron cada una de las nubes de palabras para ver la relación entre sus palabras y poder así asignarle un tópico a cada uno de los clústeres generados. Los tópicos asignados se muestran en la tabla 5.4.

Tabla 5.4. Tópicos asignados y porcentaje de reseñas asignadas

Clúster	Tópico	Porcentaje de reseñas
Clúster 1	Positivo General	49%
Clúster 2	Negativo General	8%
Clúster 3	Ubicación Airbnb	22%
Clúster 4	Positivo Host	11%
Clúster 5	Habitaciones/espacios del Airbnb	10%

Posterior a haber definido un tópico para cada uno de los clústeres, se determinó para cada reseña a qué tópico estaba asignada. Teniendo en cuenta que el modelo LDA plantea que cada documento es un conjunto de tópicos, se tomó el clúster que tuviera mayor representatividad en cada una de las reseñas. En la tabla 5.4. Se muestra, para cada uno de los tópicos, qué porcentaje de las reseñas totales están asignadas a ellos.

Para realizar un análisis más amplio de las reseñas se realizaron distintos filtros para ver si existen comportamientos distintos por las características del alojamiento, dueño y quien renta, graficando el número de reseñas por tópico, en cada uno de los filtros. El primer filtro aplicado fue por calificación (alta, media y baja) definiendo alta 8 a 10, media 6 a 8 y baja, menor de 6. En las figuras 5.18., 5.19. y 5.20. se observan los resultados obtenidos para este primer filtro, en los cuales vemos que para los tres rangos de calificaciones se tienen los tópicos de Positivo General y Ubicación Airbnb en las primeras dos posiciones. Para la Calificación baja no hizo sentido que se tuviera en primer lugar "Positivo General" por lo que se realizó un análisis más profundo en las reseñas que estaban en ese rango de calificaciones, encontrando que no había relación entre la reseña escrita y la calificación asignada, por lo que se consideraron como error.

Figura 5.18. Calificación Alta

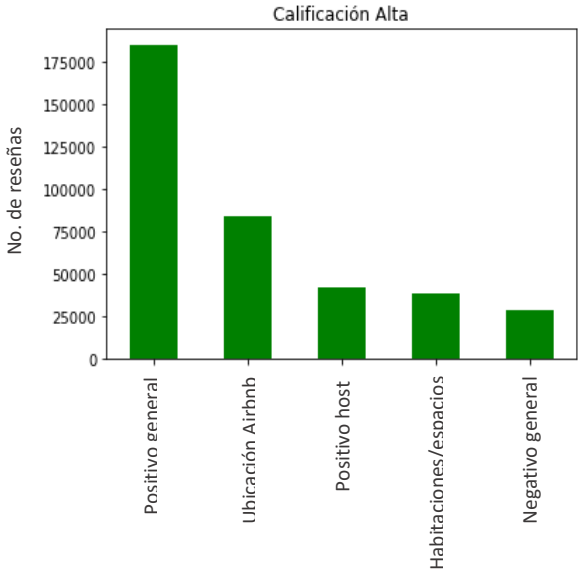


Figura 5.19. Calificación Media

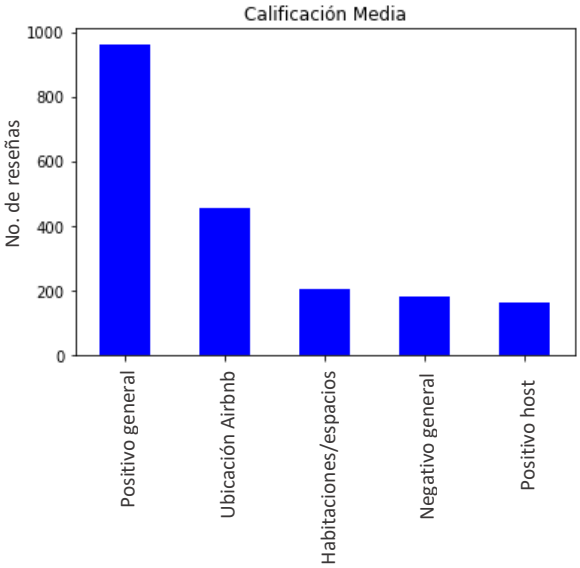
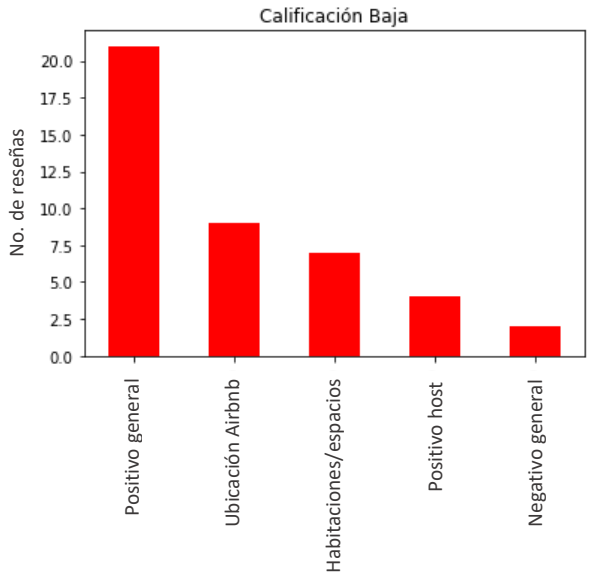


Figura 5.20. Calificación Baja



El segundo filtro realizado fue si el dueño del Airbnb tiene la categoría de Superhost o no. Los resultados del filtro se muestran en las figuras 5.21. y 5.22., encontrando que esta categoría no tiene un impacto claro en la distribución de las reseñas en los tópicos.

Figura 5.21. Superhost

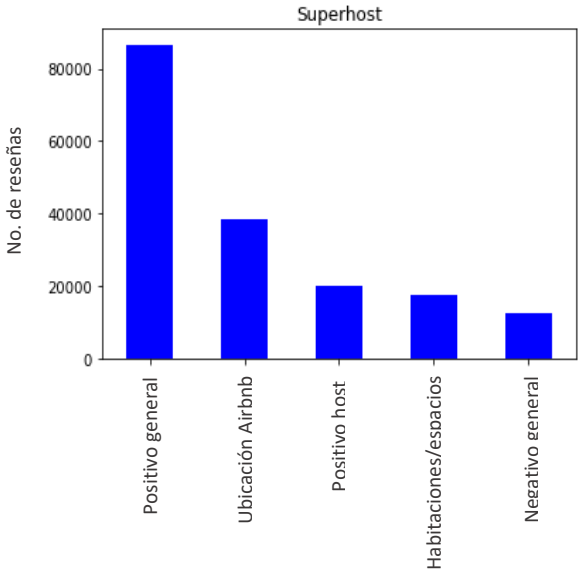
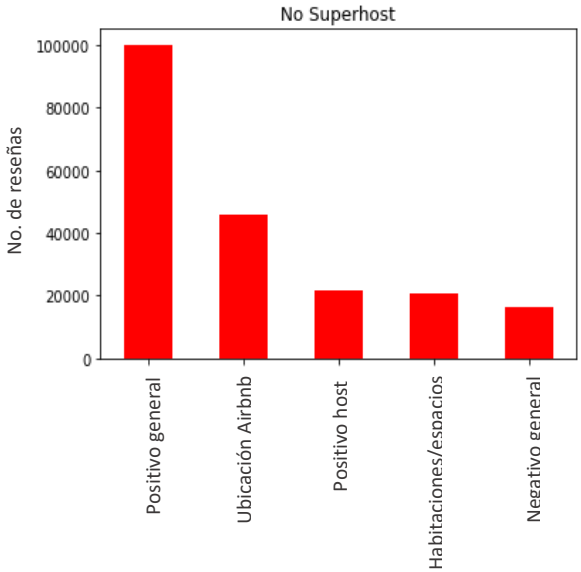


Figura 5.22. No Superhost



El tercer filtro realizado fue según el tipo de alojamiento, si es un departamento o no, encontrando que el tipo de alojamiento no tiene un impacto claro en la distribución de los

topicos en las reseñas recibidas para cada categoria. Estos resultados se muestran en las figuras 5.23. y 5.24.

Figura 5.23. Departamentos

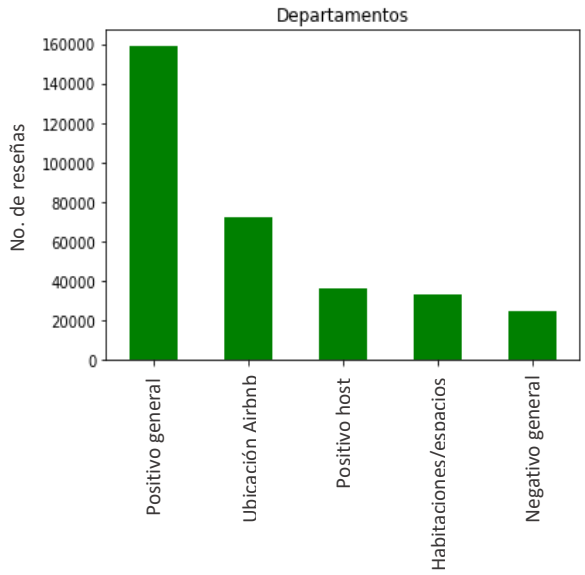
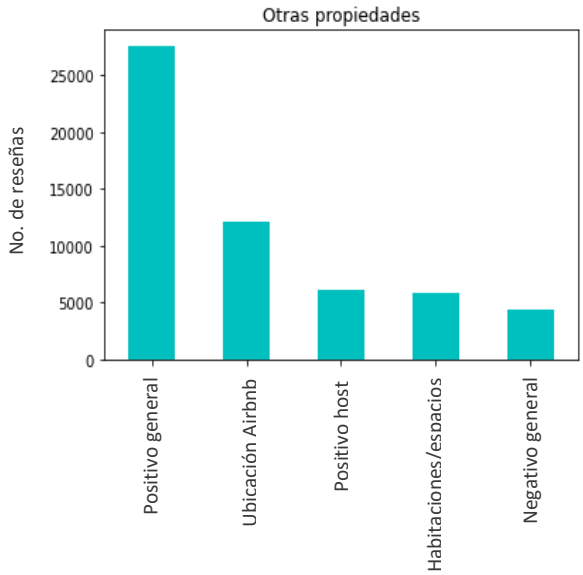


Figura 5.24. Otras propiedades.



El último filtro que se realizó fue por idioma original de la reseña, considerando unicamente los cuatro idiomas con mayor número de reseñas (inglés, español, alemán y frances). Para este filtro se encontró que para los cuatro idiomas el mayor número de reseñas se encuentran en el topico “Positivo General” y “Ubicación Airbnb” pero en tercer lugar de tópicos tenemos que para inglés y español se vuelven más relevantes los comentarios positivos para el Host y para alemán y francés las habitaciones/espacios del Airbnb. También se observó que para el idioma inglés el tópico menos relevante son los comentarios negativos. Estos resultados se observan en las figuras 5.25., 5.26., 5.27. y 5.28.

Figura 5.25. Inglés

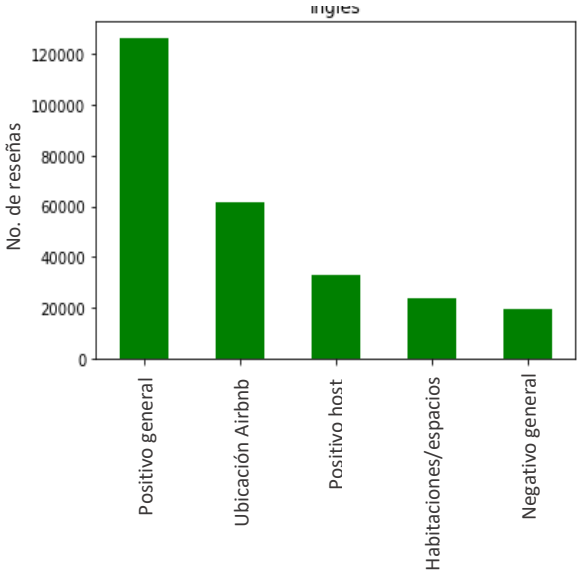


Figura 5.26. Español

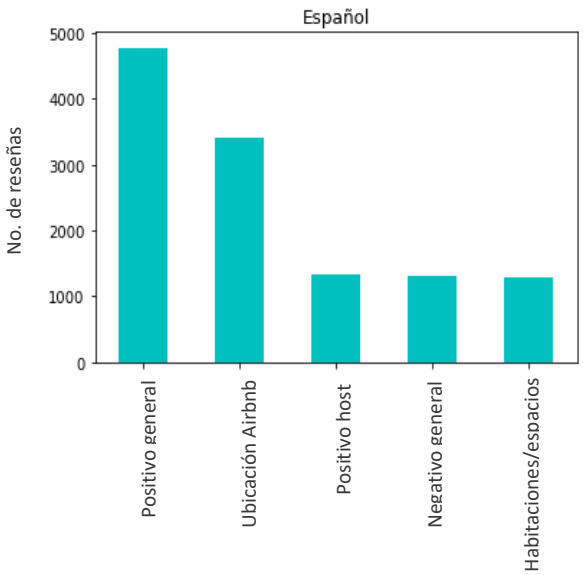


Figura 5.27. Alemán

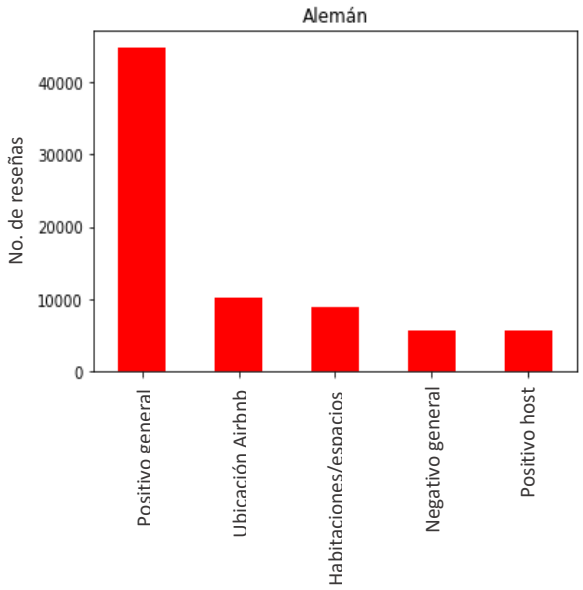
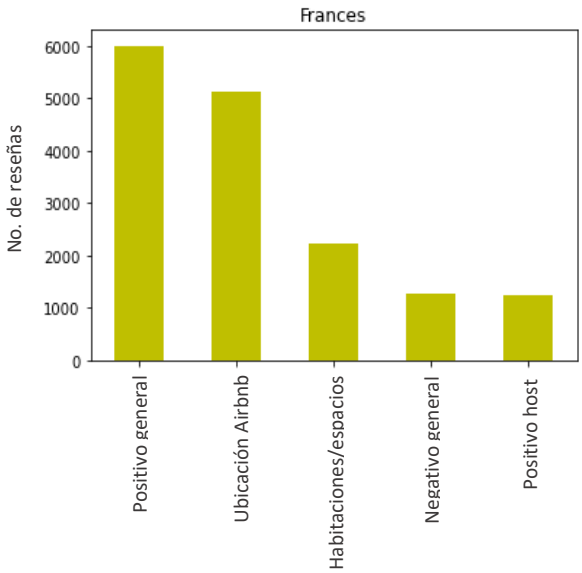


Figura 5.28. Francés



5.1.3.2. Filtrado de datos

Para continuar con el trabajo se planteó el implementar un modelo específico para cada uno de los filtros hechos anteriormente. Esto con el propósito de encontrar tópicos que describieran de mejor manera las distribuciones de reseñas en el modelo general.

Se corrieron modelos para cada uno de los filtros, haciendo un ajuste de parámetros manual y se hizo un análisis visual de sus resultados. Se encontró que ningún modelo específico describía de manera diferente el comportamiento observado en el modelo general, por lo que no se continuó con esta línea de trabajo.

5.2. Discusión

En la comparativa de los tres modelos se seleccionó como el mejor al modelo implementado de LDA. Esto después de encontrar que era el modelo que mejor desempeño en las tres métricas analizadas, teniendo el menor tiempo de ejecución, el valor de coherencia más alto y el mejor análisis visual.

La implementación de los tres modelos se hizo con los valores predeterminados en todos los parámetros de las funciones a excepción del número de tópicos máximos o esperados, donde se determinaron 16 clústeres, con el fin de que la comparación fuera sobre una misma base. La selección de los parámetros predeterminados se hizo por los tiempos de ejecución que

tenían los modelos de GSDMM y BERTopic. Con los tiempos bastante altos fue inviable realizar una optimización de parámetros haciendo un barrido en rangos de estos.

Una vez seleccionado el modelo LDA para realizar el análisis de tópicos, se realizó la optimización de los parámetros buscando tener el mejor modelo posible para el análisis del conjunto de reseñas. Este proceso fue uno de los más tardados de todo el desarrollo del trabajo, con aproximadamente 21 horas de ejecución, utilizando el 10% del total de la base de datos. Se utilizó una fracción de la base de datos, ya que realizar la optimización de los parámetros con la base completa hubiera sido inviable. Esto también sirvió para confirmar que realizar una optimización de los tres modelos para la comparativa de los modelos era completamente inviable con el tiempo de elaboración con el que se contó.

Con el modelo optimizado fue posible detectar cinco tópicos en el conjunto de reseñas siendo "Positivo General", "Negativo General", "Ubicación Airbnb", "Positivo Host" y "Habitaciones/Espacios Airbnb". Una vez identificados y categorizados los cinco tópicos se determinó el tópico principal asignado a cada una de las reseñas. Encontrando que "Positivo General" es el tópico principal con un 49% del conjunto total de reseñas, seguido de "Ubicación Airbnb" con un 22% y en último lugar "Negativo General" con un 8%. Analizando estos resultados hace sentido que Positivo se encuentre en primer lugar y Negativo en último, ya que al ser un conjunto de reseñas recibidas a lo largo de varios años se esperaría que aquellos alojamientos que tuvieran malas reseñas ya no se rentaran perdiendo así su información o que mejoran su servicio y empezaran a recibir reseñas positivas diluyendo un tanto las reseñas negativas.

Aprovechando algunas características de las reseñas como la calificación general que tenían relacionada con el texto, el idioma en el que se había escrito y el tipo de propiedad que era el alojamiento, se realizó un análisis más profundo. Este análisis consistió en obtener la distribución de los tópicos según las diferentes características. En el tema de idioma original se encontró que, en los cuatro idiomas con mayor número de reseñas, inglés, alemán, francés y español, el tópico principal seguía siendo "Positivo General" pero también se observó que las reseñas en idioma francés y español tienen una mayor tendencia a hablar de la ubicación del alojamiento en sus reseñas.

En el tema de la calificación relacionada con la reseña se encontró un fenómeno contradictorio, ya que en las reseñas con una calificación baja (entre 0 y 6) el tópico principal continuaba siendo "Positivo General". Ante este fenómeno se hizo un análisis particular de estas reseñas y se encontró que muchas de las reseñas con una calificación baja en realidad estaban dejando comentarios positivos. Esto se podría entender como un error de quienes calificaron y que el error se pudo tener al pensar que el sistema para calificar iba del 0 al 5 y no del 0 al 10.

Por último, se planteó continuar con el análisis profundo al implementar modelos específicos para cada una de las características, con el objetivo de encontrar tópicos particulares de cada característica. Este análisis no se completó, ya que en los modelos que se corrieron no se

encontró una diferencia entre lo detectado por el modelo general y estos modelos específicos. Un punto para considerar de estos modelos es que se hizo modificaciones a los parámetros de forma manual y no con una optimización, lo cual pudiese ser un factor determinante para encontrar los tópicos buscados. Otro punto para considerar es que como en el caso del modelo general los modelos específicos fueron modelos LDA, dejando de lado posibles modelos específicos con GSDMM o BERTopic.

6. CONCLUSIONES

En este capítulo se presentan las conclusiones y trabajo futuro del análisis de modelos para el modelado de tópicos, y del análisis de reseñas con el modelo que mostró el mejor desempeño.

6.1. Conclusiones

En este trabajo se implementaron tres distintos modelos de modelado de tópicos (LDA, GSDMM y BERTopic) y se realizó una comparativa entre ellos. Las métricas consideradas para evaluar los modelos fueron coherencia, tiempo de ejecución y un análisis visual de los tópicos generados. El modelo LDA presentó el mejor desempeño en todas las métricas, diferenciándose principalmente en el análisis visual, observando una clara separación de palabras entre tópicos y permitiendo definir un tópico claramente a cada conjunto de palabras, y en el tiempo de ejecución, con un tiempo de 13 minutos, casi 9 veces menor que BERTopic que fue el siguiente modelo más rápido.

Optimizando los parámetros de LDA se logró implementar un modelo que clasificó las 381,095 reseñas en cinco tópicos principales. Estos tópicos fueron:

- Positivo general
- Negativo general
- Ubicación del Airbnb
- Positivo host
- Habitaciones/Espacios

Al analizar los resultados, se concluyó que el tópico de 'Positivo general' fue el más relevante en el total de reseñas, ya que en el 49% de las reseñas totales fue el tópico principal.

Para continuar con el análisis de los resultados obtenidos por el modelo de LDA se separaron los resultados según algunos filtros, basados en características de las reseñas. Uno de los filtros del cual se obtuvieron resultados más relevantes, fue el filtro por el idioma en el que se escribió la reseña. Con este filtro se encontró que, si bien los cuatro idiomas tienen como principal las reseñas positivas generales, las personas que escriben en francés y español tienden a resaltar más en sus reseñas los temas relacionados con la ubicación de los alojamientos, a diferencia de las personas que escriben en inglés y alemán.

Se buscó realizar un análisis más profundo de las reseñas, para lograrlo se planteó un modelo LDA específico basado en los datos obtenidos por cada uno de los filtros seleccionados, estos modelos específicos tenían el objetivo de encontrar tópicos particulares de cada filtro. Al realizar este proceso se encontró que los modelos específicos encontraban resultados prácticamente iguales al modelo general por lo que no se continuó explorando estos modelos.

6.2. Trabajo Futuro

Como trabajo futuro se pudiera profundizar en el pretratamiento de la base de datos, con el fin de que las palabras que queden en el conjunto no causen ruido en la clasificación, por

ejemplo, números, nombres propios y otras que se pudieran identificar que no aporten al contexto.

Otro aspecto importante que explorar es aumentar la capacidad computacional disponible para reducir los tiempos de implementación de los modelos BERTopic y GSDMM, lo que permitirá hacer un barrido exhaustivo que optimizaría sus parámetros, con esto se podría lograr un mejor resultado en la detección de tópicos mediante estos modelos.

En este trabajo se vio que al construir diferentes modelos de LDA basados en filtros, no se pudo obtener más información. Por esto, pudiera explorarse implementar los modelos BERTopic y GSDMM en la base de datos con los distintos filtros, para analizar si estos modelos pueden detectar más información.

Este trabajo estuvo centrado en el análisis de reseñas de Airbnb Berlín, sería interesante extender el trabajo hacia el estudio de reseñas de distintas ciudades, permitiría identificar patrones y tendencias en distintas ubicaciones geográficas. Estos resultados pudieran ser relevantes para proveedores de alojamiento, para entender las demandas de los usuarios y la propia empresa Airbnb para encontrar nuevas formas de negocio o como mejorar su aplicación.

En un contexto más general la implementación de una metodología para la comparativa de tres modelos de modelado de tópicos diferentes permite que se utilice para diferentes bases de datos y no únicamente en reseñas de hospedajes vacacionales. Esta metodología se podría aplicar en conjuntos de textos cortos como comentarios en redes sociales, tweets, comentarios de productos o servicios, entre otros. Y en textos largos cwomo libros, noticias, canciones, etc.

El modelado de tópicos ha tenido avances grandísimos en pocos años, logrando disminuir las desventajas de algoritmo a algoritmo y moldeándose a las necesidades actuales del análisis de textos. Los algoritmos modulares son el último escalón que se ha alcanzado, pero todavía tienen mucho campo de mejora en aspectos como la velocidad de ejecución y la interpretabilidad. En trabajos futuros de algoritmos que se desarrollen en el área de modelado de tópicos probablemente atacarán estos aspectos y además se irán ajustando a nuevos cambios que se tengan en las características de los textos que consumimos.

7. BIBLIOGRAFÍA

[1] F. Pascual, «Topic Modeling: An Introduction,» MonkeyLearn, 26 09 2019. [En línea]. Available: https://monkeylearn.com/blog/introduction-to-topic-modeling/.

[2] T. F. P. a. L. D. Landauer, «An introduction to latent semantic analysis,» *Discourse Processes,* vol. 25, nº 2-3, pp. 259-284, 1998.

[3] D. D. Lee y H. S. Seung, «Learning the parts of objects by non-negative matrix factorization,» *Nature,* vol. 401, pp. 788-791, 1999.

[4] T. Hofmann, «Probabilistic latent semantic analysis,» de *UAI'99: Proceedings of the Fifteenth conference on Uncertainty in artificial intelligence*, San Francisco, CA, Morgan Kaufmann Publishers Inc., 1999, pp. 289-296.

[5] D. Blei, A. Ng y M. Jordan, «Latent Dirichlet Allocation,» *Journal of Machine Learning Research,* vol. 3, pp. 993-1022, 2003.

[6] Y. Z. E. Y. C. M. S. Zhiyuan Liu, «PLDA+: Parallel Latent Dirichlet Allocation with Data Placement and Pipeline Processing,» *ACM Transactions on Intelligent Systems and Technology,* vol. 2, nº 3, pp. 1-18, 2011.

[7] X. G. J. L. Y. &. C. X. Yan, «A biterm topic model for short texts,» *Proceedings of the 22nd international conference on World Wide Web,* pp. 1445-1456, 2013.

[8] J. a. W. J. A. Yin, «A dirichlet multinomial mixture model-based approach for short text clustering,» de *KDD '14: Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, Nueva York, NY, Association for Computing Machinery, 2014, p. 233–242.

[9] D. Angelov, «Top2vec: Distributed representations of topics,» *arXiv preprint arXiv:2008.09470.,* 2020.

[10] M. Grootendorst, «BERTopic: Neural topic modeling with a class-based TF-IDF procedure,» *arXiv preprint arXiv:2203.05794,* 2022.

[11] M. R. A. R. L. Wall, «Singular Value Decomposition and Principal Component Analysis,» de *A Practical Approach to Microarray Data Analysis*, Boston, MA, Springer, 2003, p. 91–109.

[12] P. K. &. P. Bansal, «Latent Semantic Analysis: An Approach to Understand Semantic of Text,» de *2017 International Conference on Current Trends in Computer, Electrical, Electronics and Communication (CTCEEC)*, Mysore, India, 2017.

[13] N. Z. X. &. P. V. Evangelopoulos, «Latent Semantic Analysis: five methodological recommendations,» *European Journal of Information Systems,* vol. 21, pp. 70-86, 2012.

[14] A. C. W.-H. P. W.-H. K. H. D. T. R. G. T. R. G. &. P. N. Graesser, «Using latent semantic analysis to evaluate the contributions of students in AutoTutor.,» *Interactive learning environments,* vol. 8, nº 2, pp. 129-147, 2000.

[15] U. M. A. N. E. E. Shailesh S. Kulkarni, «The Use of Latent Semantic Analysis in Operations Management Research,» *Decision Sciences,* vol. 45, nº 5, pp. 971-994, 2014.

[16] T. S. &. K. A. K. Tomasz Miaskiewicz, «A latent semantic analysis methodology for the identification and creation of personas,» de *In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '08)*, New York, USA, 2008.

[17] W. K. &. T. K. L. Peter W. Foltz, «The measurement of textual coherence with latent semantic analysis,» *Discourse Processes,* vol. 25, nº 2-3, pp. 285-307, 1998.

[18] M. S. B. W. W. L. K. L. &. K. Bob Rehder, «Using latent semantic analysis to assess knowledge: Some technical considerations,» *Discourse Processes,* vol. 25, nº 2-3, pp. 337-354, 1998.

[19] P. Wiemer-Hastings, «Latent Semantic Analysis,» de *DePaul University, Schoocl of Computer Science, Telecommunications, and Information Systems*, Chicago, USA, 2004.

[20] N. C. Albanese, «Topic Modeling with LSA, pLSA, LDA, NMF, BERTopic, Top2Vec: a Comparison,» Towards Data Science, 19 09 2022. [En línea]. Available: https://towardsdatascience.com/topic-modeling-with-lsa-plsa-lda-nmf-bertopic-top2vec-a-comparison-5e6ce4b1e4a5#78f7.

[21] T. L. &. C. D. C. Shen, «Integrating Clustering and Multi-Document Summarization by Bi-Mixture Probabilistic Latent Semantic Analysis (PLSA) with Sentence Bases,» *AAAI,* vol. 25, nº 1, pp. 914-920, 2011.

[22] J. Z. &. S. Gong, «Action categorization by structural probabilistic latent semantic analysis,» *Computer Vision and Image Understanding,* vol. 114, nº 8, pp. 857-864, 2010.

[23] N. M. J. T. &. E. S. Tuomo Kakkonen, «Automatic Essay Grading with Probabilistic Latent Semantic Analysis,» *Proceedings of the 2nd Workshop on Building Educational Applications Using NLP,* pp. 29-36, 2005.

[24] J. a. G. S. a. X. T. Li, «Global Behaviour Inference using Probabilistic Latent Semantic Analysis,» de *Proceedings of the British Machine Vision Conference*, BMVA Press, 2008, pp. 20.1-20.10.

[25] F. C. &. I. T. Thorsten Brants, «Topic-based document segmentation with probabilistic latent semantic analysis,» de *In Proceedings of the eleventh international conference on Information and knowledge management (CIKM '02)*, Nueva York, USA, 2002.

[26] Y. Z. &. B. M. Xin Jin, «Web usage mining based on probabilistic latent semantic analysis,» de *In Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining (KDD '04)*, Nueva York, USA, 2004.

[27] S.-Y. K. &. L.-s. Lee, «Improved Spoken Document Summarization Using Probabilistic Latent Semantic Analysis (PLSA),» de *2006 IEEE International Conference on Acoustics Speech and Signal Processing Proceedings*, Toulouse, Francia, 2006.

[28] W. Y. N. T. V. J. B. J. H. Z. Z. R. Rizvi RF, «Analyzing social media data to understand consumers' information needs on dietary supplements,» *Stud. Health Technol. Inform.,* vol. 264, pp. 323-327, 2019.

[29] R. A. R. A. S. Alnusyan, «A Semi-Supervised Approach for User Reviews Topic Modeling and Classification,» *International Conference on Computing and Information Technology,* pp. 1-5, 2020.

[30] R. Y. J. Egger, «Identifying hidden semantic structures in Instagram data: A topic modelling comparison,» *Tourism Review,* Vols. %1 de %2ahead-of-print, nº ahead-of-print , 2021.

[31] A. F. D. P. H. D. J. K. S. M. &. L. Y. K. Anandkumar, «A spectral algorithm for latent dirichlet allocation,» *Advances in neural information processing systems,* vol. 25, 2012.

[32] J. B. W. N. S. C. T. &. S. A. Petterson, «Word features for latent dirichlet allocation,» *Advances in Neural Information Processing Systems,* vol. 23, 2010.

[33] D. &. Z. X. Andrzejewski, «Latent dirichlet allocation with topic-in-set knowledge,» *In Proceedings of the NAACL HLT 2009 Workshop on Semi-Supervised Learning for Natural Language Processing,* pp. 43-48, 2009.

[34] N. M. &. E. S. Tuomo Kakkonen, «Applying Latent Dirichlet Allocation to Automatic Essay Grading. In: Salakoski, T., Ginter, F., Pyysalo, S., Pahikkala, T.,» *Advances in Natural Language Processing. FinTAL 2006,* vol. 4139, pp. 110-120, 2006.

[35] J. M. &. A. d. Waal, «A comparison of the performance of latent Dirichlet allocation and the Dirichlet multinomial mixture model on short text,» de *2016 Pattern Recognition Association of South Africa and Robotics and Mechatronics International Conference (PRASA-RobMech)*, Stellenbosch, South Africa, 2016.

[36] S. H. M. &. S. Al-augby, «LSA & LDA Topic Modeling Classification:,» *Indonesian Journal of Electrical Engineering and Computer Science,* vol. 19, nº 1, 2020.

[37] S. T. P. D. P. T. a. S. M. Y. Kalepalli, «Effective Comparison of LDA with LSA for Topic Modelling,» de *2020 4th International Conference on Intelligent Computing and Control Systems (ICICCS)*, Madurai, India, 2020.

[38] M. R. &. D. P. K. Tian, «Using Latent Dirichlet Allocation for automatic categorization of software,» de *2009 6th IEEE International Working Conference on Mining Software Repositories*, Vancouver, Canada, 2009.

[39] N. R. &. N. Vasconcelos, «Latent Dirichlet Allocation Models for Image Classification,» *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 35, nº 11, pp. 2665-2679, 2013.

[40] H. M. &. M. D. M. Lienou, «Semantic Annotation of Satellite Images Using Latent Dirichlet Allocation,» *IEEE Geoscience and Remote Sensing Letters,* vol. 7, nº 1, pp. 28-32, 2010.

[41] I. G. &. M. D. C. Văduva, «Latent Dirichlet Allocation for Spatial Analysis of Satellite Images,» *IEEE Transactions on Geoscience and Remote Sensing,* vol. 51, nº 5, pp. 2770-2786, 2013.

[42] R. A. &. B. Ravindran, «Latent dirichlet allocation based multi-document summarization,» de *In Proceedings of the second workshop on Analytics for noisy unstructured text data (AND '08)*, Nueva York, USA, 2008.

[43] F. O. O. &. C. N. Gurcan, «Investigation of Emerging Trends in the E-Learning Field Using Latent Dirichlet Allocation,» *International Review of Research in Open and Distributed Learning,* vol. 22, nº 2, pp. 1-18, 2021.

[44] M. L. L. S.-L. Travis Dyer, «The evolution of 10-K textual disclosure: Evidence from Latent Dirichlet Allocation,» *Journal of Accounting and Economics,* vol. 64, nº 2-3, pp. 221-245, 2017.

[45] M. B. F. &. B. D. Hoffman, «Online learning for latent dirichlet allocation,» *Advances in neural information processing systems,* vol. 23, 2010.

[46] K. S. L. &. Canini y T. Griffiths, «Online Inference of Topics with Latent Dirichlet Allocation,» *Artificial Intelligence and Statistics ,* pp. 65-72, 2009.

[47] U. C. &. A. Shah, «Topic Modeling Using Latent Dirichlet allocation: A Survey,» *ACM Comput. Surv.,* vol. 54, nº 7, 2021.

[48] H. W. Y. Y. C. Jelodar, «Latent Dirichlet allocation (LDA) and topic modeling: models, applications, a survey,» *Multimedia Tools and Applications,* vol. 78, pp. 15169-15211, 2019.

[49] J. Lin, «On the dirichlet distribution,» *Department of Mathematics and Statistics, Queens University,* pp. 10-11, 2016.

[50] A. Glazkova, «Using topic modeling to improve the quality of age-based text classification,» *CEUR Workshop Proceedings,* pp. 92-97, 2021.

[51] C. G. C. T. A. Weisser, «Pseudo-document simulation for comparing LDA, GSDMM and GPM topic models on short and sparse text using Twitter data,» *Computational Statistics,* 2022.

[52] K. N. A. A. A. N. H. G. a. N. K. A. Udupa, «An Exploratory Analysis of GSDMM and BERTopic on Short Text Topic Modelling,» de *2022 Fourth International Conference on Cognitive Computing and Information Processing (CCIP)*, Bengaluru, India, 2022.

[53] M. Grootendorst, «The Algorithm,» BERTopic, [En línea]. Available: https://maartengr.github.io/BERTopic/algorithm/algorithm.html.

[54] Y. J. Egger R, «A Topic Modeling Comparison Between LDA, NMF, Top2Vec, and BERTopic to Demystify Twitter Posts.,» *Front Sociol,* vol. 7, 2022.

[55] M. A. M. &. H. M. R. de Groot, «Experiments on Generalizability of BERTopic on Multi-Domain Short Text,» *arXiv preprint arXiv:2212.08459.,* 2022.

[56] H. A.-K. Abeer Abuzayed, «BERT for Arabic Topic Modeling: An Experimental Study on BERTopic Technique,» *Procedia Computer Science,* vol. 189, pp. 191-194, 2021.

[57] D. S. Leonnardo Benjamin Hutama, «Indonesian Hoax News Classification with Multilingual Transformer Model and BERTopic,» *Informatica,* vol. 46, nº 8, 2022.

[58] N. Y. S. Y. S. Eunji Jeon, «Exploring new digital therapeutics technologies for psychiatric disorders using BERTopic and PatentSBERTa,» *Tchnological Forecasting and Social Change,* vol. 186, 2023.

[59] M. A. S.-F. M. J. M. F. C. &. S. M. H. G. dos Santos, «Modelling the structure of the sports management research field using the BERTopic approach.,» *Retos: nuevas tendencias en educación física, deporte y recreación,* vol. 47, pp. 648-663, 2023.

[60] M. S. S. K. S. N. S. N. S. &. M. B. Adewunmi, «Cancer Health Disparities drivers with BERTopic modelling and PyCaret Evaluation,» *Cancer Health Disparities,* vol. 6, 2022.

[61] J. U. W.-R. Y. A. N. G. &. H.-C. Y. A. N. G. Yoon-Hwang, «Online Shopping Research Trend Analysis Using BERTopic and LDA,» *융합경영연구,* vol. 11, nº 1, pp. 21-30, 2023.

[62] Y. A. N. G. &. W.-D. L. E. E. Hoe-Chang, «Topic Modeling Analysis of Beauty Industry using BERTopic and LDA,» *융합경영연구,* vol. 10, nº 6, pp. 1-7, 2022.

[63] Y. A. N. G. &. H.-C. Y. A. N. G. Woo-Ryeong, «Exploring Depression Research Trends Using BERTopic and LDA.,» *식품보건융합연구,* vol. 9, nº 1, pp. 19-28, 2023.

[64] C. S. Ponay, «Topic Modeling on Customer Feedback from an Online Ticketing System using Latent Dirichlet Allocation and BERTopic,» de *2022 2nd International Conference in Information and Computing Research (iCORE)*, Cebu, Filipinas, 2022.

[65] Y. Jin, Travel Guide Using Text Mining and BERTopic, Los Angeles, USA: University of California, 2022.

[66] A. González, «¿Qué es Machine Learning?,» cleverdata, 2021. [En línea]. Available: https://cleverdata.io/que-es-machine-learning-big-data/. [Último acceso: 17 02 2023].

[67] SAP, «¿Qué es machine learning?,» [En línea]. Available: https://www.sap.com/latinamerica/insights/what-is-machine-learning.html. [Último acceso: 17 02 2023].

[68] A. L. Fradkov, «Early History of Machine Learning,» *IFAC,* vol. 53, nº 2, pp. 1385-1390, 2020.

[69] K. D. Foote, «A Brief History of Machine Learning,» Dataversity, 3 12 2021. [En línea]. Available: https://www.dataversity.net/a-brief-history-of-machine-learning.

[70] G. Firican, «The history of Machine Learning,» lights on data, 2022. [En línea]. Available: https://www.lightsondata.com/the-history-of-machine-learning/.

[71] J. Delua, «Supervised vs. Unsupervised Learning: What's the Difference?,» IBM, 12 03 2021. [En línea]. Available: https://www.ibm.com/cloud/blog/supervised-vs-unsupervised-learning.

[72] V. Roman, «Aprendizaje No Supervisado en Machine Learning: Agrupación,» medium, 12 06 2019. [En línea]. Available: https://medium.com/datos-y-ciencia/aprendizaje-no-supervisado-en-machine-learning-agrupaci%C3%B3n-bb8f25813edc.

[73] S. Joshi, «What is Clustering in Machine Learning: Types and Methods,» ANALYTIXLABS, 5 08 2020. [En línea]. Available: https://www.analytixlabs.co.in/blog/types-of-clustering-algorithms/.

[74] S. Mishra, «Unsupervised Learning and Data Clustering,» Medium, 19 05 2017. [En línea]. Available: https://towardsdatascience.com/unsupervised-learning-and-data-clustering-eeecb78b422a.

[75] J. C. Manjarrés, «8 algoritmos de agrupación en clústeres en el aprendizaje automático que todos los científicos de datos deben conocer,» freeCodeCamp, 24 04 2021. [En línea]. Available: https://www.freecodecamp.org/espanol/news/8-algoritmos-de-agrupacion-en-clusteres-en-el-aprendizaje-automatico-que-todos-los-cientificos-de-datos-deben-conocer/.

[76] IBM, «What is natural language processing (NLP)?,» [En línea]. Available: https://www.ibm.com/topics/natural-language-processing.

[77] B. Lutkevich, «natural language processing (NLP),» TechTarget, [En línea]. Available: https://www.techtarget.com/searchenterpriseai/definition/natural-language-processing-NLP.

[78] Deep Talk, «Historia y actualidad del Procesamiento de Lenguaje Natural,» 29 11 2021. [En línea]. Available: https://blog.deep-talk.ai/historia-y-actualidad-del-procesamiento-de-lenguaje-natural-8de41a357ca9.

[79] IBERDROLA, «¿Qué es el procesamiento de lenguaje natural y qué aplicaciones tiene?,» [En línea]. Available: https://www.iberdrola.com/innovacion/procesamiento-lenguaje-natural-pln#:~:text=El%20procesamiento%20de%20lenguaje%20natural%20tiene%20sus%20ra%C3%ADces%20en%20la,como%20el%20Test%20de%20Turing..

[80] K. Krayewski, «How NLP Chatbots Work,» Ultimate, 13 04 2022. [En línea]. Available: https://www.ultimate.ai/blog/ai-automation/how-nlp-text-based-chatbots-work.

[81] P. Galiana, «¿Cómo funcionan los asistentes de voz y por qué son el futuro?,» IEBS, 15 12 2022. [En línea]. Available: https://www.iebschool.com/blog/futuro-asistentes-voz-business-tech/.

[82] Amazon Web Service, «¿Qué es el análisis de opiniones?,» AWS, [En línea]. Available: https://aws.amazon.com/es/what-is/sentiment-analysis/.

[83] Systran, «¿Qué es la traducción automática? Traducción automática basada en reglas vs. traducción automática estadística,» [En línea]. Available: https://www.systransoft.com/es/systran/tecnologia/que-es-la-traduccion-automatica/.

[84] Amazon Web Service, «¿Qué es el análisis de textos?,» AWS, [En línea]. Available: https://aws.amazon.com/es/what-is/text-analysis/.

[85] S. Gupta, «Sentiment Analysis: Concept, Analysis and Applications,» Towards Data Science, 7 06 2018. [En línea]. Available: https://towardsdatascience.com/sentiment-analysis-concept-analysis-and-applications-6c94d6f58c17.

[86] Y. Mejova, «Sentiment Analysis: An Overview,» University of Iowa, 16 11 2009. [En línea]. Available: https://d1wqtxts1xzle7.cloudfront.net/3243118/CompsYelenaMejova-libre.pdf?1390830364=&response-content-disposition=inline%3B+filename%3DSentiment_Analysis_An_Overview.pdf&Expires=1676741479&Signature=P8mHS3wq~rHvckfNd1gHgBEUJCoUNEVofqbNyQhyk70xfbw5WUkOVft.

[87] F. Chiusano, «Quick intro to Aspect-Based Sentiment Analysis,» NLPlanet, 19 08 2022. [En línea]. Available: https://medium.com/nlplanet/quick-intro-to-aspect-based-sentiment-analysis-c8888a09eda7.

[88] F. Pascual, «Guide to Aspect-Based Sentiment Analysis,» MonkeyLearn, 08 03 2019. [En línea]. Available: https://monkeylearn.com/blog/aspect-based-sentiment-analysis/.

[89] S. Chakravarthy, «Tokenization for Natural Language Processing,» Towards Data Science, 19 06 2020. [En línea]. Available: https://towardsdatascience.com/tokenization-for-natural-language-processing-a179a891bad4.

[90] C. Khanna, «Text pre-processing: Stop words removal using different libraries,» Towards Data Science, 10 02 2021. [En línea]. Available: https://towardsdatascience.com/text-pre-processing-stop-words-removal-using-different-libraries-f20bac19929a#:~:text=What%20are%20stop%20words%3F,much%20information%20to%20the%20text..

[91] S. Srinidhi, «Stemming of words in Natural Language Processing, what is it?,» Towards Data Science, 19 02 2020. [En línea]. Available: https://towardsdatascience.com/stemming-of-words-in-natural-language-processing-what-is-it-41a33e8996e2.

[92] A. Dhuriya, «What is Topic Modeling?,» Analytics Vidhya, 01 02 2021. [En línea]. Available: https://medium.com/analytics-vidhya/what-is-topic-modeling-161a76143cae.

[93] cogitotech, «Topic Modeling: Algorithms, Techniques, and Application,» data science central, 27 09 2021. [En línea]. Available: https://www.datasciencecentral.com/topic-modeling-algorithms-techniques-and-application/.

[94] R. Pelgrim, «Short-Text Topic Modelling: LDA vs GSDMM,» Towards Data Science, 17 06 2021. [En línea]. Available: https://towardsdatascience.com/short-text-topic-modelling-lda-vs-gsdmm-20f1db742e14.

[95] T. Ganegedara, «Intuitive Guide to Latent Dirichlet Allocation,» Towards Data Science, 23 08 2018. [En línea]. Available: https://towardsdatascience.com/light-on-math-machine-learning-intuitive-guide-to-latent-dirichlet-allocation-437c81220158.

[96] R. Walker, «GSDMM: Short text clustering,» github, 17 02 2017. [En línea]. Available: https://github.com/rwalk/gsdmm.

[97] J. Yin y J. Wang, «A dirichlet multinomial mixture model-based approach for short text clustering.,» de *The 20th ACM SIGKDD international conference on Knowledge discovery and data mining.*, Nueva York, 2014.

[98] SBERT.net, «SentenceTransformers Documentation,» Sentence-Transformers, [En línea]. Available: https://www.sbert.net/.

[99] L. H. J. McInnes, «UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction,» *ArXiv e-prints 1802.03426,* 2018.

[100] J. H. S. A. Leland McInnes, «How HDBSCAN Works,» hdbscan, [En línea]. Available: https://hdbscan.readthedocs.io/en/latest/how_hdbscan_works.html.

[101] Scikit Learn, «sklearn.feature_extraction.text.CountVectorizer,» Scikit Learn, [En línea]. Available: https://scikit-learn.org/stable/modules/generated/sklearn.feature_extraction.text.CountVectorizer.html.

[102] M. Grootendorst, «c-TF-IDF,» BERTopic, [En línea]. Available: https://maartengr.github.io/BERTopic/getting_started/ctfidf/ctfidf.html.

[103] J. Pedro, «Understanding Topic Coherence Measures,» Towards Data Science, 10 06 2022. [En línea]. Available: https://towardsdatascience.com/understanding-topic-coherence-measures-4aa41339634c.

[104] Inside Airbnb, «Inside Airbnb,» Inside Airbnb, [En línea]. Available: http://insideairbnb.com/.

[105] S. Han, «googletrans 3.0.0,» 14 junio 2020. [En línea]. Available: https://pypi.org/project/googletrans/.

[106] J. Ooms, «cld2: Google's Compact Language Detector 2,» 26 octubre 2022. [En línea]. Available: https://docs.ropensci.org/cld2/.

[107] NLTK Project, «Natural Language Toolkit,» 2023. [En línea]. Available: https://www.nltk.org/.

[108] R. Rehurek y P. Sojka, «gensim 4.3.0,» University of Malta, 2010. [En línea]. Available: https://pypi.org/project/gensim/.

[109] R. Walker, «GSDMM: Short text clustering,» 2017. [En línea]. Available: https://github.com/rwalk/gsdmm.

[110] M. Grootendorst, «BERTopic: Neural topic modeling with a class-based TF-IDF procedure,» 2022. [En línea]. Available: https://maartengr.github.io/BERTopic/index.html.