BENEMÉRITA UNIVERSIDAD AUTÓNOMA DE PUEBLA

FACULTAD DE CIENCIAS DE LA COMPUTACIÓN

DOCTORADO EN INGENIERÍA DEL LENGUAJE Y DEL CONOCIMIENTO

DETERMINACIÓN DEL NÚMERO DE CENTROIDES EN EL ALGORITMO IMPROVED-FPAC PARA EL AGRUPAMIENTO DE DOCUMENTOS DE TEXTO

TESIS PRESENTADA PARA OBTENER EL GRADO DE
DOCTORADO EN INGENIERÍA DEL LENGUAJE Y DEL CONOCIMIENTO

Presenta: Inti Sandino Magallón Juan-Qui

Directora de Tesis: Dra. Darnes Vilariño Ayala
Asesor: Dr. José Francisco Martínez Trinidad

PUEBLA, PUE. DICIEMBRE 2025

II

Resumen

En esta tesis abordamos el problema de cómo mejorar la calidad del agrupamiento de documentos de texto usando el algoritmo Improved-FPAC. Es importante destacar la necesidad de categorizar documentos de texto de manera eficiente para facilitar la toma de decisiones y la búsqueda de información en grandes volúmenes de datos. Los algoritmos tradicionales, como K-Means, a menudo no representan adecuadamente la estructura de los datos, ya que utilizan un solo centroide por grupo, lo que puede ser insuficiente en situaciones reales. Para solucionar esto, nuestro trabajo propone un método para determinar el número de centroides por cluster considerando las características específicas del corpus, el número de documentos y el vocabulario, previo a realizar el agrupamiento de documentos y un algoritmo que evalúa en cada iteración el número de centroides para cada grupo para Improved-FPAC. Tanto el método como el algoritmo propuesto se validan experimentalmente y muestran mejoras significativas en la calidad de los agrupamientos sin aumentar demasiado el tiempo de ejecución. Estas alternativas se adaptan mejor a las particularidades de cada conjunto de datos, superando las limitaciones de los valores fijos de centroides utilizados en estudios previos. La investigación presenta una mejora importante al algoritmo Improved-FPAC, ofreciendo una forma más adaptativa y eficiente de determinar el número de centroides, lo que resulta en una mejor calidad en el agrupamiento de documentos de texto.

Índice general

Índice de figuras

Índice de cuadros

Capítulo 1

Introducción

El presente trabajo se enmarca en el área de la **Ingeniería del Lenguaje y del Conocimiento**, disciplina dedicada al desarrollo de métodos y herramientas para procesar, analizar y comprender la información textual, transformando datos no estructurados en conocimiento significativo para la toma de decisiones. Esta disciplina se orienta tanto a la mejora de los procesos de interpretación semántica como a la construcción de modelos que faciliten la representación y organización del conocimiento.

Dentro de este contexto, uno de los problemas prácticos más relevantes es el agrupamiento de documentos, que consiste en clasificar de manera automática documentos de texto en conjuntos homogéneos basados en su similitud. Este proceso tiene aplicaciones en ámbitos como la recuperación de información, la categorización automática, la generación de resúmenes y la organización del conocimiento en entornos especializados, permitiendo a los usuarios identificar patrones y relaciones en la información de forma eficiente.

Dentro de los métodos de agrupamiento de documentos, uno de los algoritmos más utilizados es **K-Means**. Este algoritmo realiza la clasificación de un conjunto de documentos en K grupos, es decir, en grupos o conjuntos homogéneos de documentos que comparten características similares. Inicialmente el algoritmo K-means genera aleatoriamente un número K de centroides, o los selecciona aleatoriamente de la muestra disponible de documentos. Estos centroides sirven como referencia o representantes de los grupos a construir. Posteriormente, el algoritmo K-means asigna cada documento al grupo cuyo centroide es el más cercano, según una medida de

similitud (por ejemplo, la distancia euclidiana o la similitud del coseno). Posteriormente, con los grupos construidos, el algoritmo K-Means re-calcula los centroides con base en los documentos asignados a cada grupo (generalmente como el promedio), y el proceso se repite hasta que la clasificación se estabiliza de acuerdo a un criterio de paro. Entre las bondades de **K-Means** destacan su simplicidad, su facilidad de implementación y su alta eficiencia, lo que lo convierte en una herramienta popular para el análisis y la organización de textos, permitiendo identificar patrones y relaciones de manera rápida y clara.

Entre los algoritmos más exitosos para el agrupamiento de documentos basados en **K-Means** (Abualigah et al. (2020); Agarwal et al. (2022); Alghamdi & Selamat (2019); Cozzolino & Ferraro (2022); Dobrakowski et al. (2021); Eligüzel et al. (2022); Inje et al. (2023); Kim et al. (2020); Malik et al. (2022); Pandey & Shukla (2023); Ponnusamy et al. (2022); Song et al. (2015); V & S (2023); Yong & Liew (2023)), el algoritmo **Improved-FPAC** ha reportado los mejores resultados en términos de la calidad de los agrupamientos. La principal innovación de Improved-FPAC es que, en lugar de asignar directamente cada documento a un grupo mediante el cálculo de la distancia a un único centroide (como en el K-Means tradicional), utiliza dicho centroide como consulta para recuperar documentos a través de listas invertidas. En este enfoque, cada grupo se representa mediante un conjunto de diez centroides, lo que permite realizar una recuperación más fina y eficiente de los documentos. Sin embargo, el algoritmo Improved-FPAC tiene una limitación importante, utiliza de forma preestablecida el valor fijo de diez para el número de centroides, sin considerar las particularidades específicas del corpus a agrupar. Al utilizar un valor fijo para el número de centroides por grupo, se omite la posibilidad de ajustar este parámetro a las características particulares de cada corpus.

Esta investigación atiende a la necesidad de desarrollar una alternativa para determinar el número de centroides a utilizar en el algoritmo Improved-FPAC, lo que reduce la dependencia de un único valor predefinido de número de centroides por grupo. En este sentido, resulta esencial desarrollar alternativas que sean capaces de adaptarse de manera automática o semiautomática a las particularidades del corpus, evitando la fijación arbitraria de este parámetro. Así, se busca mejorar la calidad del agrupamiento de documentos, ofreciendo una solución adaptable a la heterogeneidad de los corpus y respondiendo a las demandas actuales en la organización y análisis de documentos de texto.

Además, la mejora en la calidad del agrupamiento tiene implicaciones directas en áreas como la recuperación de información, la categorización automatizada y la generación de resúmenes, donde una representación más fiel de la estructura semántica de los documentos facilita la identificación de patrones y relaciones relevantes. De esta manera, la investigación no solo aborda un reto teórico, sino que también contribuye al avance en la aplicación práctica de técnicas de procesamiento del lenguaje natural y análisis de datos, respondiendo a las demandas actuales en entornos especializados y en la toma de decisiones basada en información textual.

1.1. Planteamiento del problema

Dentro de los algoritmos de agrupamiento de documentos basados en K-Means se han desarrollado variantes que buscan mejorar tanto la eficiencia como la calidad de los agrupamientos de documentos. La variante más destacada es el algoritmo Improved-FPAC. Sin embargo, como ya se ha mencionado en Improved-FPAC el número de centroides por grupo es fijo, lo que puede no adaptarse adecuadamente a la variabilidad de los datos en diferentes corpus.

El desafío principal es desarrollar una alternativa que permita determinar el número de centroides en el algoritmo Improved-FPAC para el agrupamiento de documentos.

1.2. Objetivos

Objetivo general

Desarrollar un algoritmo para determinar el número de centroides a utilizar en Improved-FPAC para el agrupamiento de documentos tal que permita mejorar la calidad de los agrupamientos.

Objetivos específicos

1. Analizar la influencia de las características intrínsecas del corpus (tamaño, diversidad léxica y distribución de clases) en la calidad del agrupamiento de documentos.
2. Establecer la relación entre los parámetros del corpus y la cantidad de centroides necesaria para representar adecuadamente la estructura interna de los textos.
3. Diseñar y desarrollar un criterio o estrategia que permita ajustar la selección de centroides de forma automática o semiautomática en función de las características del corpus.

1.3. Preguntas de investigación

Una vez identificadas las limitaciones del algoritmo actual de agrupamiento de Improved-FPAC, establecemos las siguientes preguntas de investigación:

1. ¿De qué manera influyen las características intrínsecas del corpus (como el tamaño, la diversidad léxica y la distribución de clases) en la calidad del agrupamiento de documentos?
2. ¿Cuál es la relación entre las características del corpus (tamaño, diversidad léxica y número de categorías) y la cantidad de centroides necesaria para representar adecuadamente la estructura interna de los textos?
3. ¿Es posible diseñar un algoritmo que, ajustando de forma automática o semiautomática la selección de centroides o puntos representativos, mejore la calidad de los agrupamientos?

1.4. Hipótesis

Es posible desarrollar un método para determinar el número de centroides por grupo para el algoritmo Improved-FPAC que mejore la calidad de los agrupamientos.

1.5. Estructura de la tesis

La presente tesis se organiza en varios capítulos, cada uno de los cuales aborda aspectos fundamentales del estudio. A continuación, se describe la estructura general del trabajo, detallando el contenido y los objetivos de cada capítulo:
El Capítulo 1 presenta el contexto y la motivación del estudio, destacando la relevancia del agrupamiento de documentos en el ámbito de la ingeniería del lenguaje y del conocimiento. Se expone el problema de investigación, se formulan los objetivos generales y específicos, y se plantean las preguntas de investigación y la hipótesis. El capítulo 2 establece las bases conceptuales que sustentan la investigación. Se describen algunas definiciones relacionadas con el agrupamiento de documentos y las métricas de evaluación del agrupamiento de documentos, entre las que se incluyen F-Score, Índice Rand, Información Mutua Normalizada(NMI), Pureza, Precision y Recall. El capítulo 3 realiza una revisión los trabajos relacionados más relevantes en el área del agrupamiento de documentos, utilizando la recuperación de documentos. El capítulo 4 describe dos alternativas para determinar el número de centroides por grupo en el algoritmo Improved-FPAC. El capítulo 5 presenta los experimentos para evaluar las propuestas introducidas en el capítulo anterior en distintos corpus de documentos públicos estándar, utilizando las métricas descritas en el Marco Teórico para evaluar la calidad del agrupamiento. Además, se discuten los hallazgos en relación con las fortalezas y posibles limitaciones. El capítulo 6 sintetiza los principales hallazgos del estudio y presenta las conclusiones generales. Se discuten las implicaciones teóricas y prácticas del trabajo, las contribuciones al conocimiento en el área del procesamiento de información textual y el agrupamiento, y se proponen recomendaciones para futuras investigaciones, destacando posibles aplicaciones del método desarrollado y se identifican áreas para seguir profundizando en el desarrollo de métodos para mejorar la calidad en el agrupamiento de documentos.

Capítulo 2

Marco Conceptual

El agrupamiento de documentos es una técnica que organiza automáticamente un conjunto de textos en grupos, de modo que los documentos dentro de cada grupo sean más similares entre sí que con aquellos de otros grupos Jain et al. (1999). Para esto, usualmente se transforman los textos en representaciones numéricas utilizando, el modelo de espacio vectorial y técnicas de ponderación como TF-IDF, lo que permite comparar documentos mediante medidas de similitud como la similitud del coseno Salton et al. (1975); Schutze et al. (2008). Este enfoque resulta fundamental en áreas como la recuperación de información y la minería de datos Aggarwal & Zhai (2012).

El agrupamiento es un proceso de división de un conjunto de datos u objetos en un conjunto de subclases significativas, llamadas grupos. Formalmente, dada una colección de n objetos descritos por un conjunto de p atributos, el objetivo del agrupamiento es derivar una división útil de los n objetos en un número de grupos. Un agrupamiento es una colección de objetos similares entre sí, basados en los valores de sus atributos Arco et al. (2006).

El objetivo del agrupamiento de documentos es formar una colección o grupo de subconjuntos que cumplan:

1. Los documentos que pertenecen al mismo grupo deben ser tan similares como sea posible.

2. Los documentos que pertenecen a grupos diferentes deben ser tan diferentes como sea posible.

Para comprender mejor el agrupamiento de documentos, es necesario entender algunos conceptos clave y relevantes que se enumeran a continuación:

1. **Representación de documentos**: Diferentes modelos de representación, como el modelo vectorial, el modelo de espacio semántico y los modelos basados en redes neuronales, capturan diferentes aspectos de los documentos y permiten medir su similitud Tan et al. (2005). Estos modelos son cruciales para transformar los textos en una forma que pueda ser analizada y procesada por algoritmos de agrupamiento. En el modelo vectorial, cada documento se expresa como un vector de características (usualmente ponderado) que permite calcular la similitud —por ejemplo, mediante la similitud del coseno— entre documentos.

2. **Medidas de similitud**: Diversas medidas, como la distancia euclidiana, la similitud del coseno y la similitud de Jaccard, se utilizan para calcular la similitud entre documentos Schutze et al. (2008). La similitud del coseno, por ejemplo, mide el coseno del ángulo entre dos vectores, donde la dirección del vector es más importante que su magnitud. Estas medidas permiten a los algoritmos identificar qué documentos están relacionados entre sí, formando así agrupamientos significativos.

3. **Algoritmos de agrupamiento**: Constituyen una rama fundamental del aprendizaje no supervisado, cuyo objetivo es identificar estructuras ocultas dentro de un conjunto de datos sin la necesidad de contar con etiquetas o categorías previamente definidas. En esencia, estos algoritmos buscan organizar los datos en grupos, de manera que los elementos pertenecientes a un mismo grupo presenten una alta similitud entre sí y, al mismo tiempo, una marcada diferencia con respecto a los elementos de otros grupos Tan et al. (2016).

4. **Evaluación del agrupamiento**: Se utilizan diversas métricas, como el índice de Rand ajustado, la pureza y la medida Fowlkes-Mallows, para evaluar los grupos generados Jain et al. (1999). Estas métricas permiten comparar diferentes algoritmos y configuraciones de parámetros.

2.1. Representación de Documentos

La representación de documentos es el proceso mediante el cual se transforma un texto en una forma de modo que pueda ser procesado y analizado por algoritmos

computacionales. Este proceso es esencial en áreas como la recuperación de información, la minería de datos y, en particular, en el agrupamiento de documentos, donde la representación influye directamente en la capacidad para identificar similitudes y diferencias entre documentos.

Existen diversos enfoques para representar documentos, entre los que se destacan:

- **Modelo de Bolsa de Palabras (Bag-of-Words, BoW)**: Considera cada documento como una colección (o bolsa) de palabras, sin tener en cuenta el orden ni la estructura gramatical.
- **Modelo de Espacio Vectorial**: El modelo de espacio vectorial representa cada documento como un vector en un espacio de alta dimensión, en el que cada dimensión corresponde a un término del vocabulario extraído del corpus. Este enfoque, basado en el modelo de Bolsa de Palabras (BoW), facilita la comparación entre documentos mediante el uso de métricas de similitud, como la similitud del coseno o la distancia euclidiana.

  El modelo vectorial Salton et al. (1975) se desarrolló originalmente para la indexación automática. Bajo este modelo, una colección de n documentos con m términos únicos se representa como una matriz de término-documento de $m \times n$, donde cada documento es un vector de m dimensiones. Aunque el modelo en sí está bien establecido y forma la base para la discusión posterior, ha habido un interés reciente en representaciones alternativas de documentos.

  Se han utilizado varios esquemas de ponderación de términos, incluida la frecuencia de términos binarios y la frecuencia de términos simples (es decir, cuántas veces aparecen las palabras en el documento). En el esquema más popular, los vectores de documentos se componen de pesos que reflejan la frecuencia de los términos en el documento multiplicada por el inverso de su frecuencia en toda la colección ($tf \times idf$). La suposición es que las palabras que aparecen con frecuencia en un documento pero rara vez en toda la colección tienen un alto poder de discriminación. Bajo todos estos esquemas, es típico normalizar los vectores de documentos a la unidad de longitud.

  La mayoría de los algoritmos de agrupamiento utilizan una representación de

espacio vectorial de una forma u otra, aunque debe tenerse en cuenta que no se codifica información sobre el orden de las palabras, razón por la cual el modelo vectorial a veces se denomina "bolsa de palabras.º "modelo de diccionario". Esta representación permite que los documentos sean tratados como conjuntos de características independientemente del orden en que aparezcan en el texto, lo que facilita la comparación y el análisis.

- **Representaciones Distribuidas**: En enfoques más recientes se han utilizado técnicas de aprendizaje profundo para representar documentos mediante vectores densos (embeddings), tales como Word2Vec, GloVe o BERT. Estas representaciones capturan relaciones semánticas y contextuales entre palabras, proporcionando vectores que reflejan de forma más rica la semántica del documento.

2.2. Medidas de Similitud

Las medidas de similitud son fundamentales en el procesamiento y análisis de documentos, ya que permiten cuantificar la relación entre dos textos representados en forma numérica. Estas medidas son esenciales en tareas como la recuperación de información, la clasificación y la agrupación de documentos, ya que permiten identificar qué textos comparten un contenido similar. En el contexto de técnicas de agrupamiento (agrupamiento), su aplicación resulta clave para asegurar que los documentos con mayor similitud sean ubicados en el mismo grupo. A continuación se revisan las medidas más utilizadas en el agrupamiento de documentos.

2.2.1. Similitud Coseno

La similitud del coseno es una de las medidas más utilizadas en el análisis de textos. Esta medida se basa en calcular el coseno del ángulo entre dos vectores que representan los documentos en un espacio vectorial. Dado que en este modelo cada documento se expresa mediante un vector de características (usualmente ponderado con TF-IDF), la similitud del coseno se define como:

$$\text{Similitud del Coseno} = \frac{\vec{A} \cdot \vec{B}}{\|\vec{A}\| \times \|\vec{B}\|}$$

donde $\vec{A}$ y $\vec{B}$ son los vectores de dos documentos.

La similitud de coseno es especialmente útil para documentos de texto, ya que ignora la magnitud del vector y se enfoca en su orientación, es decir, considera el patrón de ocurrencia de los términos más que la cantidad total de veces que aparecen. Esto permite comparar documentos en función de la proporción en que se utilizan los términos, sin verse afectado por las diferencias de longitud entre ellos. Esta propiedad es particularmente relevante en el análisis de textos, donde el tamaño de los documentos puede variar considerablemente Salton et al. (1975).

2.2.2. Índice de Jaccard

La similitud de Jaccard es otra métrica empleada, especialmente cuando los documentos se representan como conjuntos de términos. Esta medida se define como el cociente entre el tamaño de la intersección y el tamaño de la unión de los conjuntos:

$$\text{Similitud de Jaccard} = \frac{|A \cap B|}{|A \cup B|}$$

El índice de Jaccard resulta particularmente útil en escenarios donde es relevante considerar la presencia o ausencia de términos sin ponderar sus frecuencias, proporcionando una visión basada en la coincidencia de elementos Jaccard (1901).

2.2.3. Similitud de Dice

La similitud de Dice se define como:

$$\text{Similitud de Dice} = \frac{2 \times |A \cap B|}{|A| + |B|}$$

donde $|A|$ y $|B|$ representan el número de elementos (términos) en cada documento, y $|A \cap B|$ es el tamaño de la intersección de ambos conjuntos. Esta métrica enfatiza la coincidencia de términos al duplicar el peso de la intersección, lo que puede proporcionar una medida sensible en situaciones donde los conjuntos son pequeños o presentan poca superposición Dice (1945).

2.2.4. Coeficiente de Sørensen

El coeficiente de Sørensen, frecuentemente considerado equivalente a la similitud de Dice, se expresa de manera similar:

$$\text{Coeficiente de Sørensen} = \frac{2 \times |A \cap B|}{|A| + |B|}$$

Ambas medidas ofrecen valores entre 0 y 1, donde 1 indica que los conjuntos son idénticos y 0 que no tienen elementos en común. Esta métrica es especialmente útil en escenarios en los que la frecuencia de los términos es menos relevante que la mera presencia o ausencia de estos Sorensen (1948).

2.3. Algoritmos de agrupamiento

Existen diversos algoritmos para el agrupamiento de documentos, que se pueden clasificar en:

1. **Particionales:** Este método divide el conjunto de documentos en un número fijo de grupos, asignando cada documento a un grupo de forma exclusiva. Un ejemplo clásico es el algoritmo **K-Means** MacQueenet al. (1967).

2. **Jerárquicos**: Se construye una estructura en forma de árbol (dendrograma) que representa las relaciones de similitud entre documentos. Puede ser aglomerativo (comenzando con cada documento como grupo individual y fusionándolos progresivamente) o divisivo (iniciando con un único grupo y dividiéndolo en subgrupos). Un ejemplo de este es el algoritmo Hierarchical Agglomerative agrupamiento (HAC) Jain & Dubes (1988).

3. **Basado en densidad**: Estos métodos identifican regiones densas en el espacio de características y las consideran grupos, mientras que los puntos en regiones menos densas se clasifican como ruido o outliers. Ejemplos populares son DBSCAN y OPTICS Ankerst et al. (1999); Kriegel et al. (2011).

4. **Basado en modelos**: Los algoritmos de agrupamiento basados en modelos asumen que los datos provienen de una combinación de distribuciones estadísticas subyacentes, y su objetivo es estimar la probabilidad de que cada instancia pertenezca a uno u otro grupo. Un ejemplo representativo es el modelo de Gaussian Mixture Models (GMM), que utiliza el algoritmo

de Expectation-Maximization para ajustar una mezcla de distribuciones gaussianas a los datos, permitiendo asignaciones probabilísticas en lugar de determinísticas como en K-means. Otra técnica destacada es Latent Dirichlet Allocation (LDA), utilizada especialmente en minería de textos, donde cada documento se considera una combinación de temas latentes, modelados como distribuciones sobre palabras. Además, variantes bayesianas como el Bayesian Gaussian Mixture Model incorporan inferencia bayesiana para estimar automáticamente el número de grupos, permitiendo una mayor flexibilidad y robustez frente a la incertidumbre en la estructura de los datos. Este tipo de enfoques resulta especialmente útil cuando los grupos tienen formas complejas o se superponen en el espacio de características Tan et al. (2016).

5. **Espectrales**: Los algoritmos de agrupamiento espectrales se basan en el análisis del espectro —es decir, los valores y vectores propios— de una matriz de similitud o de adyacencia construida a partir de los datos. Estos métodos transforman el problema de agrupamiento en un problema de particionamiento de grafos, donde los nodos representan documentos y las aristas indican la similitud entre ellos. A través de la descomposición espectral de dicha matriz, es posible proyectar los datos a un espacio de menor dimensión que conserva la estructura de conectividad, facilitando la identificación de grupos con formas no lineales o distribuciones complejas que los algoritmos tradicionales no logran capturar. Entre los algoritmos más representativos de este enfoque se encuentra el agrupamiento Spectral, que utiliza los vectores propios del Laplaciano del grafo para reducir la dimensionalidad antes de aplicar un algoritmo como K-means sobre la nueva representación. Esta metodología ha demostrado ser efectiva en escenarios donde los grupos no son convexos ni separables linealmente Von Luxburg (2007).

6. **Agrupamiento difuso (fuzzy):** El agrupamiento difuso (fuzzy clustering) se diferencia de los métodos tradicionales al permitir que cada instancia pertenezca simultáneamente a múltiples grupos, asignando un grado de pertenencia a cada uno. Esta aproximación refleja con mayor precisión la ambigüedad inherente en muchos conjuntos de datos reales, especialmente cuando las fronteras entre grupos no son claramente definidas. En lugar de una clasificación estricta, se construye una matriz de pertenencia en la que cada valor indica la afinidad relativa de un documento respecto a cada grupo.

Uno de los algoritmos más representativos de este enfoque es Fuzzy C-Means (FCM), que extiende el algoritmo K-means introduciendo funciones de pertenencia difusa y un parámetro de control de difusidad que ajusta el grado de solapamiento entre grupos. Este tipo de agrupamiento es especialmente útil en dominios como el procesamiento de lenguaje natural o la bioinformática, donde los objetos pueden compartir características comunes con múltiples categorías Bezdek et al. (1984).

Dentro de los algoritmos de agrupamiento el algoritmo K-means uno de los más utilizados en la agrupación de documentos y el objeto de estudio de nuestra investigación Agarwal et al. (2022); Bezdan et al. (2021a); Cozzolino & Ferraro (2022); Dobrakowski et al. (2021); Dodda & Babu (2024); Eligüzel et al. (2022); Haji et al. (2023); Inje et al. (2023); Liu et al. (2024); Purohit et al. (2023); Sharma et al. (2023). K-means se basa en la idea de que un punto central puede representar un grupo. En particular, para K-means se usa la noción de un centroide, que es el punto medio o mediano de un grupo de puntos Jain & Dubes (1988); Kaufman & Rousseeuw (2009). Para medir la distancia entre el centroide y los objetos que integran el grupo, el algoritmo K-Means comúnmente utiliza la distancia euclidiana. En el caso de los datos de tipo texto, es típico utilizar una medida de similitud coseno en lugar de la distancia euclidiana Dhillon & Modha (2001).

El algoritmo K-Means es un algoritmo de agrupamiento (clustering) ampliamente utilizado en el campo de aprendizaje automático y minería de datos. Su objetivo principal es particionar un conjunto de datos en grupos, de tal manera que los puntos de datos en el mismo grupo sean más similares entre sí que con aquellos en otros grupos MacQueenet al. (1967).

En este algoritmo se asigna un conjunto de datos a k grupos, donde cada grupo está representado por su centroide, definido como el promedio de los puntos que lo conforman. El valor de k debe establecerse previamente y corresponde al número de grupos que se desean identificar.

El procedimiento inicia con la selección aleatoria de k centroides dentro del espacio de agrupamiento. A continuación, cada punto del conjunto de datos se asigna al grupo cuyo centroide esté más cercano, utilizando comúnmente la distancia euclidiana como medida de similitud. Una vez que todos los puntos han sido asignados, se

recalculan los centroides de cada grupo con base en la media de los puntos asignados a ellos.

Este proceso de asignación y actualización se repite de manera iterativa. En cada iteración, los puntos pueden cambiar de grupo dependiendo de la posición de los nuevos centroides. El algoritmo continúa hasta que se alcanza un criterio de convergencia, como la estabilización de los centroides o la ausencia de cambios en las asignaciones.

Aunque es un algoritmo simple y fácil de implementar, el tiempo de ejecución está dominado por el cálculo del centroide del agrupamiento más cercano a cada punto, un proceso que toma $O(kn)$ tiempo por iteración, donde n representa el número de puntos en el conjunto de datos y k el número de grupos. Este tiempo se debe a la necesidad de calcular la distancia entre cada punto y cada uno de los k centroides en cada iteración. Para casos generales, el algoritmo puede tener una complejidad de $n^{O(kd)}$, donde d es la dimensionalidad del espacio de características. Este tiempo puede incrementarse significativamente a medida que aumenta la cantidad de datos (n), el número de agrupamientos deseados (k) o la complejidad del espacio vectorial (d) Vattani (2009).

2.4. Evaluación del agrupamiento

Para evaluar la calidad de los agrupamientos generados por un algoritmo de agrupamiento, especialmente cuando se dispone de una clasificación externa o “real” de referencia, se utilizan métricas como el índice de Rand (RI), la precisión, la exhaustividad (*recall*) y el F-Score. Estas métricas se basan en los valores de una matriz de confusión, la cual compara las asignaciones realizadas por el algoritmo (clase predicha, grupo asignado por el algoritmo de agrupamiento) con las categorías verdaderas a las que pertenecen los elementos (clase real). En este contexto, la **clase real** corresponde a la etiqueta correcta del dato según un conocimiento externo validado, mientras que la **clase predicha** es el grupo asignado por el algoritmo.

		Clase Predicha	
		Positivo	**Negativo**
Clase Real	**Positivo**	Verdadero Positivo (VP)	Falso Negativo (FN)
	Negativo	Falso Positivo (FP)	Verdadero Negativo (VN)

Cuadro 2.1: Matriz de Confusión

Las métricas evalúan la calidad de un clustering contando las concordancias y las discordancias entre los pares de datos, tanto en las etiquetas predichas por el algoritmo de agrupamiento como en las etiquetas de las clases reales. Se analizan pares de puntos x y y, y se clasifican de la siguiente manera:

- **Verdadero Positivo:** Los puntos x y y pertenecen a la misma clase real y al mismo grupo construido por el algoritmo de agrupamiento. (Acierto en la unión).
- **Falso Positivo:** Los puntos x y y pertenecen a clases reales distintas y están juntos en un grupo construido por el algoritmo de agrupamiento. (Error de unión: el algoritmo los junta incorrectamente).
- **Falso Negativo:** Los puntos x y y están en la misma clase real y el algoritmo de agrupamiento los pone en diferentes grupos. (Error de separación: el algoritmo los separa incorrectamente).
- **Verdadero Negativo:** Los puntos x y y pertenecen a distintas clases reales y el algoritmo de agrupamiento los etiqueta en distintos grupos. (Acierto en la separación).

A partir de estos cuatro conteos (VP, FP, VN y FN) se calculan las métricas de evaluación empleadas en este trabajo, con las que se evalúa la calidad del agrupamiento.

Índice Rand: Es una medida del porcentaje de decisiones correctas tomadas por el algoritmo, considerando la matriz de confusión se define de la siguiente forma:

$$RI = \frac{VP + VN}{VP + FP + FN + VN}$$

Recall: Es una métrica que mide la capacidad del algoritmo para recuperar correctamente los elementos que deberían estar en un mismo grupo. En otras palabras, indica qué proporción de las verdaderas asociaciones existentes en los datos fueron identificadas por el algoritmo. Un valor alto de Recall significa que el modelo logró reunir la mayoría de los elementos que efectivamente pertenecen juntos, mientras que un valor bajo refleja que dejó fuera una parte importante de esas asociaciones reales. Se define de la siguiente manera:

$$Recall = \frac{VP}{VP + FN}$$

Precisión: Es una métrica que mide la exactitud de las agrupaciones realizadas por el algoritmo. Indica qué proporción de los elementos que el modelo colocó en un mismo grupo realmente pertenecen a la clasificación real. Un valor alto de Precisión significa que la mayoría de las asociaciones hechas por el algoritmo fueron correctas, mientras que un valor bajo refleja que muchas de las agrupaciones predichas incluyen elementos que en realidad no deberían estar juntos. Se define como:

$$Precision = \frac{VP}{VP + FP}$$

F-Score: Se calcula a partir de la precisión y recall de la prueba, donde la precisión es el número de resultados positivos verdaderos dividido por el número de todos los resultados positivos, incluidos los que no se identificaron correctamente, y el recuerdo es el número de resultados positivos verdaderos dividido por el número de todas las muestras que deberían haber sido identificadas como positivas. Por lo tanto, su valor está ligado a estas dos variables y se define de la siguiente manera:

$$F - Score = \frac{2 \times Precision \times Recall}{Precision + Recall}$$

Además de las métricas derivadas de la matriz de confusión, como el Índice Rand, el Recall, la Precisión y el F-Score, existen otras medidas que permiten evaluar la calidad del agrupamiento desde una perspectiva distinta, sin depender directamente de la correspondencia documento a documento entre clases reales y predichas. Estas métricas, denominadas **Pureza** e **Información Mutua Normalizada (NMI)**, analizan la relación global entre las particiones obtenidas por el algoritmo y las clases de referencia, considerando la distribución de los documentos dentro de cada grupo.

Pureza: La medida de pureza evalúa el grado en que los documentos agrupados en un mismo grupo pertenecen a una única clase. Para calcularla, cada grupo

se asocia con la clase más frecuente dentro de él, y se mide la proporción de documentos correctamente asignados respecto al total. Su valor oscila entre 0 y 1, donde 1 indica una separación perfecta entre clases y grupos. Esta métrica tiende a favorecer a agrupamientos con un mayor número de grupos, ya que la pureza aumenta conforme los grupos se fragmentan. Se define como:

$$Pureza = \frac{1}{N} \sum_{k} \max_{j} |C_k \cap L_j|$$

donde:

- N es el número total de documentos del conjunto de datos.
- k es el número total de clústeres producidos por el algoritmo.
- C_k denota el *k-ésimo clúster* (el conjunto de documentos asignados al grupo k, con $k \in \{1, \ldots, k\}$).
- L_j denota la *j-ésima clase real* o etiqueta de referencia del corpus (con j recorriendo todas las clases disponibles).
- $|C_k \cap L_j|$ es la *cardinalidad* (número de elementos) de la intersección entre el clúster k y la clase real j; es decir, cuántos documentos del clúster k pertenecen realmente a la clase j.
- $\max_j$ indica que, para cada clúster C_k, se toma el *máximo* sobre todas las clases reales L_j: esto selecciona la clase mayoritaria dentro de C_k.

Es decir, para cada clúster C_k se cuenta cuántos de sus documentos pertenecen a su clase real más frecuente ($\max_j |C_k \cap L_j|$); luego se suman esos valores sobre todos los clústeres y se normaliza por N. El resultado mide qué tan puros son los clústeres respecto a las clases verdaderas: cuanto más alta la pureza, mayor es la proporción de documentos en cada clúster que comparten la misma clase real.

Información Mutua Normalizada (NMI). La métrica NMI mide la correspondencia entre los grupos generados y las clases reales del conjunto de datos, a partir de los conceptos de teoría de la información. Evalúa cuánta información comparten ambas particiones (clases y grupos), normalizando el valor entre 0 y 1 para permitir comparaciones entre distintos experimentos. Un valor de NMI cercano a 1 indica una

alta correspondencia entre la agrupación obtenida y la estructura de clases original. Su formulación se expresa como:

$$NMI = \frac{2 \times I(L;C)}{H(L) + H(C)}$$

donde $I(L;C)$ representa la información mutua entre las clases L y los grupos C, mientras que $H(L)$ y $H(C)$ son las entropías individuales de las clases y los grupos, respectivamente Vinh et al. (2009). Esta métrica es ampliamente utilizada en la evaluación de algoritmos de agrupamiento porque equilibra la pureza y la homogeneidad, ofreciendo una medida más robusta de la calidad global del agrupamiento.

En síntesis, este capítulo establece los conceptos que sustentan el desarrollo de la investigación doctoral. Se abordaron los principios del agrupamiento de documentos de texto, la representación vectorial de documentos y las medidas de similitud, así como las métricas de evaluación más utilizadas. En el contexto de esta investigación, estos elementos resultan esenciales para comprender los mecanismos mediante los cuales se evalúa la calidad y la representatividad de los agrupamientos de documentos generados. De esta manera, este capítulo proporciona el soporte conceptual para desarrollar la investigación sobre la determinación del parámetro que especifica el número de centroides por grupo a utilizar en el algoritmo Improved-FPAC.

Capítulo 3

Trabajo Relacionado

En los últimos años, el agrupamiento de documentos de texto ha sido ampliamente estudiado, dando lugar a una gran cantidad de propuestas que buscan mejorar la calidad, eficiencia y escalabilidad de los algoritmos tradicionales. Entre estas propuestas, destacan los enfoques basados en distancia —las variantes de K-Means, HAC o DBSCAN— así como aquellos fundamentados en modelos probabilísticos y de aprendizaje profundo, que permiten representar con mayor precisión las relaciones semánticas entre los documentos Agarwal et al. (2022); Bezdan et al. (2021b); Cozzolino & Ferraro (2022); Garg & Gupta (2018); Liu et al. (2024); Qiao et al. (2024). Esta diversidad de estrategias refleja el interés continuo de la comunidad científica por mejorar la calidad de los agrupamientos, ajustando los algoritmos a la naturaleza de los documentos y al crecimiento exponencial de la información digital.

Dentro de este amplio panorama, han cobrado especial relevancia los métodos de agrupamiento basados en recuperación de información (IR-based clustering), los cuales sustituyen el cálculo explícito de distancias por consultas (*queries*) a índices invertidos para determinar la similitud entre documentos y centroides. Este paradigma ha demostrado ser eficiente en el agrupamiento de documentos, ya que permite aprovechar las estructuras de búsqueda optimizadas de los sistemas de recuperación. En este contexto, los algoritmos de la familia **FPAC (Fast Partitional Clustering)** y sus variantes han reportado resultados sobresalientes en términos de precisión y tiempo de ejecución, consolidándose como una alternativa efectiva frente a los métodos tradicionales basados en distancias Ali et al. (2024); Bejos et al. (2020); Ganguly (2018); Hirsch et al. (2025); Rajagopal et al. (2022).

Este algoritmo sigue la misma estrategia general de K-Means: selecciona aleatoriamente k centroides (uno para cada grupo), pero, a diferencia de K-Means que calcula la distancia o similitud con el centroide, FPAC emplea la recuperación de documentos mediante listas invertidas, donde cada centroide se utiliza como una consulta que recupera, a través de la operación TOP(x), los documentos más similares en el espacio vectorial. De esta manera, se evita el cálculo explícito de similitudes entre cada documento y todos los centroides, reduciendo de forma sustancial el costo computacional característico de K-Means. Al aprovechar la estructura de listas invertidas, FPAC logra una ejecución significativamente más eficiente sin comprometer la calidad del agrupamiento.

Entre las alternativas propuestas para mitigar el problema de la alta dimensionalidad, destaca el uso de listas invertidas como estrategia para optimizar los procesos de recuperación y agrupamiento de documentos. En este contexto, se han presentado diversas mejoras, entre las cuales sobresale el algoritmo de agrupamiento rápido FPAC (Fast PArtitional Clustering) propuesto por Ganguly (2018). Este algoritmo sigue la misma estrategia general de K-Means: selecciona aleatoriamente k centroides (uno para cada grupo), pero, a diferencia de K-Means que calcula la distancia o similitud con el centroide, FPAC emplea la recuperación de documentos mediante listas invertidas, donde cada centroide se utiliza como una consulta que recupera, a través de la operación TOP(x), los documentos más similares en el espacio vectorial. De esta manera, se evita el cálculo explícito de similitudes entre cada documento y todos los centroides, reduciendo de forma sustancial el costo computacional característico de K-Means. Al aprovechar la estructura de listas invertidas, FPAC logra una ejecución significativamente más eficiente sin comprometer la calidad del agrupamiento.

El algoritmo **FPAC (Fast PArtitional Clustering)** consiste de los sguientes pasos:

1. **Selección de los centroides iniciales:** se eligen los centroides procurando que sean diferentes entre sí. El primer centroide se selecciona de manera aleatoria, y los siguientes se obtienen mediante consultas que evitan incluir documentos similares a los ya elegidos, garantizando así una mayor diversidad temática entre los grupos.

2. **Asignación de documentos a los grupos:** una vez definidos los centroides, cada uno se utiliza como consulta sobre un índice invertido que devuelve

una lista con los documentos más similares. Se tiene una lista por cada centroide. Si un documento aparece en una sola lista se incorpora al grupo correspondiente al centroide que lo haya obtenido en su lista. Si un documento aparece en varias listas, se asigna al grupo con el mayor utilizando el **Puntaje de Recuperación Normalizado** que es una métrica que mide la utilidad de un resultado de búsqueda basándose en su posición en la lista recuperada de documentos. Para cada documento recuperado, este puntaje puede interpretarse como el grado de similitud respecto del centroide que lo recuperó, considerando su posición en la lista de resultados. El valor del puntaje se normaliza en el intervalo $[0, 1]$, de modo que valores cercanos a 1 indican un alto grado de similitud del documento con respecto al centroide, mientras que valores cercanos a 0 representan un bajo grado de similitud Ganguly (2018); Liuet al. (2009). En caso de que el documento no sea recuperado por ningún centroide, se le asigna aleatoriamente a uno de ellos, dado que su relación con los demás es mínima.

3. **Actualización de los centroides:** en esta fase, FPAC emplea una **heurística de centralidad** que sustituye el cálculo del promedio vectorial por un criterio más simple y eficiente. El nuevo centroide c_i de cada grupo C_i se selecciona como el documento que contiene el mayor número de términos únicos dentro del conjunto de documentos asignados al grupo. La idea es que un documento con una cobertura más amplia de términos únicos puede representar mejor el contenido global del grupo y servir como punto de referencia para la siguiente iteración.

4. **Iteración del proceso:** los pasos de asignación y actualización se repiten de forma iterativa hasta que las asignaciones se estabilizan o se alcanza el número máximo de iteraciones establecido.

No obstante, a pesar de su eficiencia y buen desempeño, el algoritmo presenta una oportunidad de mejora relacionada con la selección de los centroides para la recuperación de los documentos. En particular, la selección de un solo centroide por grupo puede generar una cantidad considerable de documentos no asignados a ningún grupo, debido a que el centroide c_i seleccionado no siempre representa adecuadamente a todos los documentos pertenecientes al grupo C_i. Como resultado, el sistema de recuperación basado en c_i puede recuperar un número limitado de documentos, provocando que aquellos que no contengan c_i como centroide sean

asignados de forma aleatoria o que se produzca una alta coincidencia de términos comunes en la intersección de los centroides correspondientes a los k diferentes grupos. Esta limitación abre la posibilidad de desarrollar enfoques más flexibles que ajusten dinámicamente la representatividad de los centroides y mejoren la cobertura de recuperación durante el proceso de agrupamiento.

Debido a estas limitaciones, se propuso una mejora orientada a reducir la pérdida de representatividad y cobertura frente al algoritmo tradicional. Este algoritmo, conocido como **Improved-FPAC**, fue presentada por Bejos et al. (2020), que introduce la heurística denominada *L-packing*. A diferencia de FPAC, este algoritmo no utiliza un único centroide para representar a cada grupo, sino más de un centroide almacenados en L listas independientes. Para ello, los documentos que pertenecen al grupo C_i se dividen en L subconjuntos disjuntos, cada uno de los cuales genera su propio centroide. Esta estrategia permite que un grupo sea cubierto por múltiples centroides. Sin embargo, la implementación de esta heurística conlleva una degradación en el tiempo de cómputo respecto del algoritmo FPAC original, debido al aumento en el número de centroides y cálculos requeridos en cada iteración.

Dado que el algoritmo Improved-FPAC constituye la base sobre la cual se desarrolla la presente investigación, a continuación se describe con mayor detalle su funcionamiento. Este algoritmo representa un punto intermedio clave en la evolución de los métodos de agrupamiento basados en recuperación de información, al extender al algoritmo de agrupamiento FPAC mediante el uso de múltiples centroides por grupo y un proceso de recuperación para construir los agrupamientos sin calcular las distancias entre pares de documentos. Comprender su estructura, sus mecanismos de selección de centroides y su impacto en la calidad del agrupamiento resulta esencial para contextualizar las mejoras introducidas posteriormente en esta investigación doctoral.

3.1. Improved-FPAC

Como se comentó en la sección de introducción, este algoritmo se basa en FPAC y sigue un enfoque de recuperación de información para construir los grupos. Dado que en en esta investigación doctoral se estudia el problema de determinar el número de centroides por grupo que se utilizarán en Improved-FPAC, en el resto de la

sección, revisaremos las ideas principales de este algoritmo.

Improved-FPAC consta de 5 pasos:

1. **Selección de los centroides iniciales:** la selección de los k centroides iniciales se realiza de la misma forma que en el algoritmo FPAC.

2. **Construcción de los grupos iniciales:** al igual que el paso anterior, este paso se realiza de la misma forma que en FPAC.

3. **Selección de múltiples centroides:** una vez que Improved-FPAC ha construido los primeros k grupos iniciales, calcula l centroides por grupo, donde l es un parámetro del algoritmo. Dado un grupo C_i, los documentos que pertenecen a C_i se dividen en l subconjuntos disjuntos almacenados en l listas. Para esto, Improved-FPAC recorre la lista de documentos asignados al grupo y los distribuye uno por uno en cada lista disjunta hasta que se terminen todos los documentos del grupo. Estos subconjuntos permiten cubrir el grupo con varios subconjuntos de documentos, y en cada subconjunto se calcula un centroide como el vector promedio de los vectores de documentos en el subconjunto correspondiente. De esta manera, Improved-FPAC obtiene l centroides por cada grupo C_i; i = 1, ..., k.

4. **Asignación de no-centroides con múltiples centroides:** para asignar un documento no-centroide d a un grupo, para cada grupo C_i, Improved-FPAC utiliza cada uno de los l centroides como consulta para obtener la lista recuperada l para el grupo C_i. Luego, para un documento no-centroide d, al igual que en FPAC, se calcula el Puntaje de Recuperación Normalizado para cada centroide en cada grupo C_i. El documento d se asigna al grupo C_i, donde la suma de los puntajes de recuperación normalizados sea mayor. De la misma manera que en la construcción de los grupos iniciales, los documentos que no son recuperados por ningún centroide se asignan aleatoriamente a cualquiera de los k grupos.

5. **Condición de parada:** la selección de múltiples centroides y la asignación de no centroides con múltiples centroides se repite hasta que el número de documentos que cambian de grupo entre una iteración y otra no sea mayor que un porcentaje definido como la condición de parada; en Bejos et al. (2020) se utilizó un 10 %. A continuación, podemos ver el diagrama de Improved-FPAC.

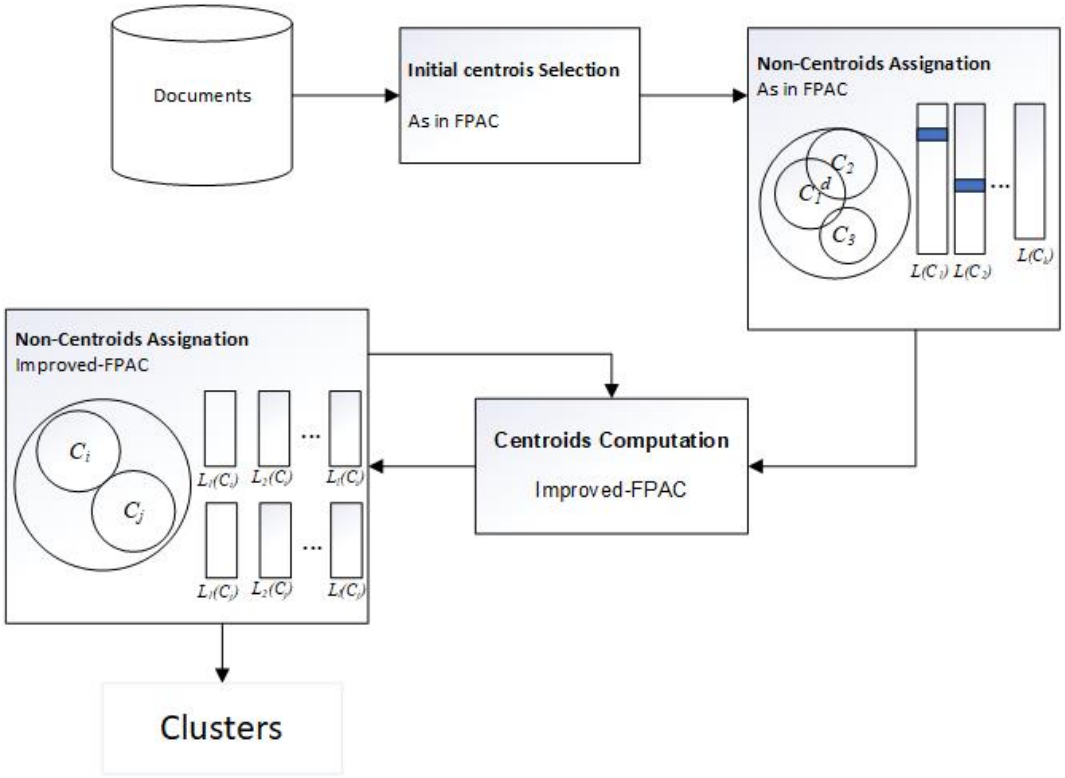

Figura 3.1: Diagrama Improved-FPAC.

En síntesis, el algoritmo Improved-FPAC representa un avance dentro de la línea de investigación de métodos de agrupamiento basados en recuperación de información. Su propuesta de utilizar múltiples centroides por grupo permitió mejorar la representatividad interna de los grupos sin comprometer de forma considerable la eficiencia computacional, lo que lo convierte en un referente directo para los trabajos que buscan optimizar la calidad del agrupamiento en colecciones textuales. De este modo, el análisis de Improved-FPAC no solo aporta contexto teórico, sino que establece el fundamento sobre el cual se construye la investigación doctoral en esta tesis sobre la determinación del número de centroides a utilizar en Improved-FPAC para el agrupamiento de documentos.

Capítulo 4

Métodos propuestos para determinar el número de centroides en el algoritmo Improved FPAC

Este capítulo presenta dos propuestas para determinar el número de centroides en el algoritmo Improved-FPAC para el agrupamiento de documentos de texto. La primera propuesta consiste en un método basado en características del corpus, denominado CCM-CD. La segunda propuesta es un algoritmo denominado Forward Improved-FPAC, que es una extensión del algoritmo Improved FPAC que ajusta el número de centroides mediante una búsqueda hacia adelante.

4.1. Método para Determinar el Número de Centroides Utilizando las Características del Corpus (CCM-CD)

Para proponer un método basado en las características del corpus que permita determinar el número de centroides en el algoritmo Improved-FPAC se analizaron experimentalmente los resultados de dicho algoritmo ejecutándolo con diferentes valores de l, con el objetivo de evaluar cómo la cantidad de centroides por grupo influye en la calidad general del agrupamiento. Este análisis experimental permitió determinar el valor de l que ofrece la mejor calidad para cada uno de los corpus mostrados en la tabla 4.1, dichos corpus corresponden a los mismos utilizados en los experimentos de Improved-FPAC, ya que son conjuntos de datos ampliamente

utilizados y con características variadas que permiten evaluar el rendimiento del método en distintos contextos. Se utilizó la métrica *F-Score* para identificar el valor de l que producía los mejores resultados para cada conjunto de datos, dado que es una medida ampliamente empleada en la evaluación de algoritmos de agrupamiento de documentos. Esta métrica permite además valorar de manera equilibrada la relación entre la *precisión* y el *recall*.

La Tabla 4.1 presenta las características de cada corpus utilizado, incluyendo el número de documentos ($Docs$), el número de palabras ($Words$) y el número de clases ($Class$). Este último valor no forma parte del proceso de agrupamiento; sin embargo, se incluye como referencia para facilitar el análisis en la comparación de los resultados obtenidos. La última columna muestra el número de centroides por grupo (l) que generó los mejores valores de *F-Score* para cada corpus, a partir de 20 ejecuciones realizadas con valores de l entre 10 y 200.

Cuadro 4.1: Mejor valor de l y características del corpus

Corpus	**Docs**	**Words**	**Class**	**l**
R52	9,100	18,989	52	40
R8	7,674	17,150	8	90
20newsgroups	18,248	69,120	20	130
Webkb	4,168	7,657	4	160
Hitech	2,301	22,119	6	170
BBCNews	2,225	20,749	5	180
Reviews	4,069	44,287	5	180
AGNews	127,599	41,337	4	190
HealthTweets	62,718	80,653	16	190
Webace20	3,900	11,606	20	90

A partir de los experimentos realizados con el algoritmo Improved-FPAC, se observó que en general mientras más documentos con vocabulario diverso haya en un grupo, mayor será el número de centroides necesarios para representarlo. Es importante destacar que, a mayor tamaño de un grupo, más documentos contendrá (y probablemente aumentará el número de términos), por lo que se requerirá un mayor número de centroides para representarlo adecuadamente.

Por otro lado, a medida que aumenta el número de grupos en los que se agrupan los documentos, en general, la cantidad de documentos por grupo disminuye Yuan et al. (2022); por tanto, se requerirá un número menor de centroides para representar los documentos dentro de cada grupo.

De lo anterior se infiere que el número de documentos y el tamaño del vocabulario ($Words$) están relacionados directamente con el número de centroides necesarios para representar los grupos. En cambio, el número de grupos (k) está relacionado inversamente con la cantidad de centroides requeridos para representar un grupo. Un valor mayor de k da lugar a más grupos, cada uno con menos documentos, y por lo tanto se necesitan menos centroides para representarlos. Por el contrario, si k es pequeño, habrá menos grupos, por lo que cada grupo contendrá más documentos, por lo que se requerirá un mayor número de centroides para representarlo. A partir de este análisis, se propone la siguiente expresión que relaciona las características del corpus como se ha explicado:

$$x = \frac{Docs + Words}{k} \tag{4.1}$$

Graficando los valores de la expresión 4.1 contra los valores de l que produjeron los mejores resultados en las 20 ejecuciones (al variar l entre 10 y 200 para cada corpus), se obtiene un conjunto de puntos en el plano (uno por cada corpus). A partir de estos puntos, se puede obtener una línea de tendencia que permite calcular el valor de l adecuado para el algoritmo Improved-FPAC en un corpus desconocido, a partir de sus características. La Figura 4.1 muestra dicha línea de tendencia, donde x representa el valor normalizado de la expresión 4.1 y y es el número de centroides por grupo (l) que produjo los mejores resultados para el corpus. Posteriormente, se ajustó una línea de tendencia logarítmica utilizando regresión basada en la suma de residuos al cuadrado Morgan & Tatar (1972).

La siguiente expresión representa la línea de tendencia obtenida en Microsoft Excel:

$$l = 48 \cdot \ln(x) + 298 \tag{4.2}$$

Esta línea de tendencia permite calcular el número de centroides por grupo que debe usarse en el algoritmo Improved-FPAC, a partir de las características del corpus.

En la práctica, aplicar este método implica evaluar la expresión 4.2, donde x se

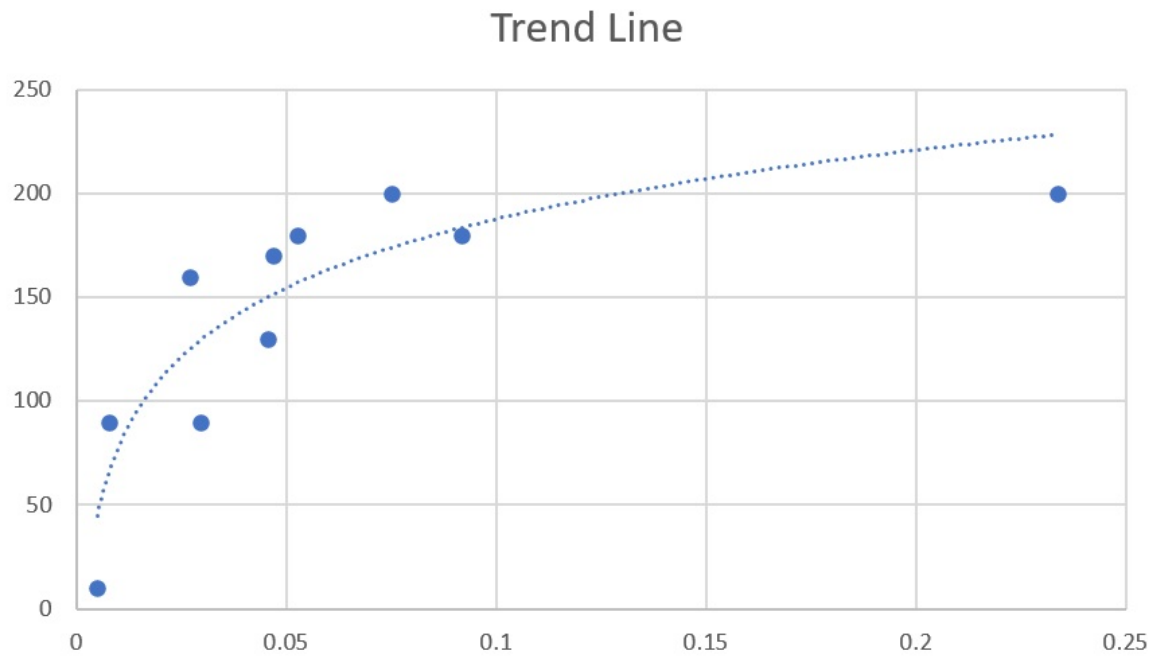

Figura 4.1: Línea de tendencia entre los valores de la expresión 4.1 (eje X) y el número de centroides por grupo l que generaron los valores de F-Score más altos (eje Y).

calcula a partir de las características del corpus. En la Figura 4.2 se muestra el diagrama de flujo del procedimiento para aplicar Improved-FPAC utilizando el método propuesto para determinar el número de centroides por grupo (parámetro l).

Es necesario mencionar que el valor obtenido con la expresión 4.2 debe truncarse a un número entero para definir el valor de l correspondiente a cada corpus. Además, si el valor calculado es menor que 10, se establece $l = 10$; y si es mayor que 200, se establece $l = 200$.

4.2. Algoritmo Forward Improved-FPAC

A continuación presentaremos el algoritmo *Forward Improved-FPAC*, el cual incorpora una búsqueda hacia adelante en el algoritmo *Improved-FPAC*, debido a que esta búsqueda permite incrementar progresivamente el número de centroides

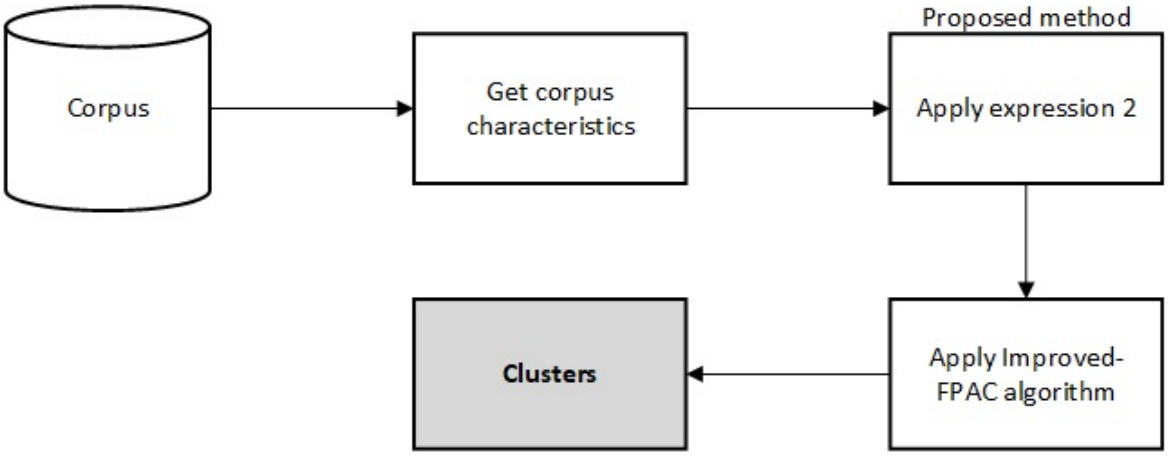

Figura 4.2: Diagrama de flujo del procedimiento para aplicar Improved-FPAC usando el método propuesto.

por grupo y evaluar en cada paso si se obtiene una mejora en la calidad del agrupamiento. De esta manera, se evita fijar de antemano un valor determinado para el número de centroides por grupo. Como se comentó en la sección anterior, Improved-FPAC reportó buenos resultados de agrupamiento utilizando diez centroides por grupo Bejos et al. (2020). Así, la idea detrás de nuestro algoritmo propuesto, Forward Improved-FPAC, es aprovechar el agrupamiento construido con Improved-FPAC, para ajustar el número de centroides mediante una búsqueda hacia adelante. **Forward Improved-FPAC** inicia el agrupamiento con 10 centroides por grupo, como se recomienda en Bejos et al. (2020). Una vez construido el agrupamiento inicial, se realiza una evaluación de la calidad del agrupamiento obtenido. Luego, se incrementa el número de centroides por grupo; se calculan los nuevos centroides, como se explica en el paso 3 de Improved-FPAC y construye el agrupamiento como se explica en el paso 4 de Improved-FPAC, se realiza una evaluación de la calidad del agrupamiento obtenido para determinar si el incremento de centroides mejora la calidad de los grupos. El incremento de número de centroides, la asignación de centroides y el agrupamiento se repiten de manera iterativa hasta que se cumpla el **criterio de paro**, es decir, cuando un nuevo incremento en el número de centroides no produce una mejora en la calidad del agrupamiento respecto al agrupamiento anterior con menor número de centroides. De esta manera, el agrupamiento del corpus se construye en función de los

mejores grupos que se pueden alcanzar con la búsqueda hacia adelante al variar el número de centroides por grupo. Este proceso implica evaluar la calidad de los grupos generados en cada paso de la búsqueda hacia adelante. Por lo tanto, en Forward Improved-FPAC proponemos evaluar la calidad del agrupamiento en cada paso de la búsqueda hacia adelante con base en el **Promedio del Puntaje de Recuperación Normalizado (Average NRS)** para cada grupo, el cual representa la proximidad promedio de los documentos a los centroides del grupo. Como se explica a continuación:

$$\text{squaresAverage\_NRS}_c = \frac{\sum_{i=1}^{n_c}(NRS_{i,c})^2}{n_c} \tag{4.3}$$

donde:

- c es el índice del grupo,
- n_c es el número de documentos en el grupo c,
- $NRS_{i,c}$ es el Puntaje de Recuperación Normalizado del documento i en el grupo c, el cual toma valores en $[0, 1]$.

También proponemos calcular la **Varianza del Puntaje de Recuperación Normalizado (Variance NRS)** dentro de cada grupo, que mide la dispersión de los documentos alrededor del centroide del grupo:

$$\text{Variance\_NRS}_c = \frac{\sum_{i=1}^{n_c}(NRS_{i,c} - \text{Average\_NRS}_c)^2}{n_c} \tag{4.4}$$

donde:

- c es el índice del grupo,
- $NRS_{i,c}$ es el Puntaje de Recuperación Normalizado del documento i en el grupo c,
- Average_NRS$_c$ es el Average NRS previamente calculado para el grupo c.

Una menor Varianza del Puntaje de Recuperación Normalizado indica que los documentos en un grupo están más cercanos al centroide. Esto es deseable porque implica que los documentos asignados a un grupo comparten mayor similitud, reduciendo la dispersión interna y mejorando la compacidad general de la estructura del grupo (un aspecto importante de un agrupamiento de alta calidad) Halkidi et al.

(2001); Jain & Dubes (1988); Xu & Wunsch (2005).

Por lo tanto, la **Calidad** del agrupamiento construido por nuestro algoritmo propuesto se calcula agregando las varianzas de todos los grupos y dividiéndolas entre el número de centroides por grupo:

$$\text{Clustering\_Quality} = \frac{\sum_{c=1}^{k} \text{Variance\_NRS}_c}{\text{Number of Centroids}} \tag{4.5}$$

donde:

- c es el índice del grupo,
- k es el número total de grupos,

La inclusión de un término de penalización (dividiendo entre el número de centroides) en nuestras métricas de calidad de agrupamiento busca equilibrar la cohesión interna y la complejidad del modelo. Sin dicha penalización, Forward Improved-FPAC podría favorecer el uso de muchos centroides, lo que reduciría artificialmente la varianza sin mejorar realmente la calidad del agrupamiento Halkidi et al. (2001); Sugar & James (2003); Tibshirani et al. (2001). Dividir entre el número de centroides por grupo desalienta el uso excesivo de centroides.

El algoritmo Forward Improved-FPAC propuesto es el siguiente:

1. **Agrupamiento inicial con Improved-FPAC**: El algoritmo comienza realizando un agrupamiento inicial utilizando el algoritmo Improved-FPAC, comenzando con 10 centroides por grupo como se sugiere en Bejos et al. (2020), hasta que el algoritmo converge o llega a un máximo de 5 iteraciones. Una vez realizado el agrupamiento, la calidad del agrupamiento construido se evalúa usando la expresión 4.5.

2. **Incrementar el número de centroides y construir un nuevo agrupamiento**: Se incrementa el número de centroides y se construye un nuevo agrupamiento con el número incrementado de centroides como en el paso 3 y 4 de Improved-FPAC).

3. **Comparación de calidad del agrupamiento**: La calidad del nuevo agrupamiento, con el número de centroides incrementado, calculada con la expresión

4.5, se compara con la calidad obtenida en la iteración anterior(con el número de centroides previo). Este paso determina si el aumento de centroides ha producido una mejora en la calidad del agrupamiento.

4. **Criterio de paro**: Si la calidad del último agrupamiento construido supera a la calidad de agrupamiento construido en la iteración anterior, el algoritmo regresa al segundo paso. Si no se obtiene una mejora, el proceso de búsqueda hacia adelante se detiene y se selecciona el mejor agrupamiento encontrado hasta el momento. En este paso se propone fijar un número máximo permitido de centroides, denominado *maxCentroids*, para limitar el espacio de búsqueda y evitar un crecimiento ilimitado. Este valor actúa como salvaguarda y típicamente se establece con base en observaciones empíricas o como una función del tamaño del corpus, como se explicará más adelante en el capítulo de experimentos.

Complejidad computacional de Forward Improved-FPAC

Para analizar la **complejidad computacional de Forward Improved-FPAC**, es necesario considerar que este algoritmo se desarrolla a partir de las bases establecidas por FPAC e Improved-FPAC. Por lo tanto, antes de examinar su complejidad, resulta indispensable describir brevemente la complejidad de sus predecesores, ya que Forward Improved-FPAC conserva la estructura fundamental de ambos, incorporando únicamente mecanismos adicionales de búsqueda hacia adelante y evaluación de calidad que afectan su costo de procesamiento.

El análisis de la complejidad computacional de los algoritmos basados en FPAC permite comprender su comportamiento en términos de eficiencia y escalabilidad frente a grandes colecciones de documentos. En el caso del algoritmo **FPAC (Fast Partitional Clustering Algorithm)** propuesto por Ganguly (2018), la reducción del costo computacional se logra mediante el uso de **listas invertidas** y la operación *TOP(x)*, que evita la comparación directa entre cada documento y todos los centroides, como ocurre en K-Means. Esta estrategia permite que la complejidad se exprese como $O(PKV/N \cdot \log(PKV/N))$, donde N representa el número de documentos, K el número de grupos, P la longitud promedio de los vectores (número de términos no nulos) y V el tamaño del vocabulario. Dado que en la práctica el cociente V/N suele ser menor que 1, la complejidad efectiva se aproxima a $O(PK\log(PK))$, lo que

ofrece una mejora significativa en comparación con el K-Means tradicional, cuya complejidad es $O(NKP)$.

En el caso del algoritmo **Improved-FPAC**, la complejidad computacional también considera el uso de varios centroides por grupo, a diferencia del FPAC original que utiliza únicamente uno por clúster. En este contexto, el número total de centroides empleados durante la ejecución del algoritmo se expresa como $K' = K \times n_c$, donde K corresponde al número de grupos, D al número de documentos y n_c al número de centroides por grupo. En consecuencia, la complejidad computacional de Improved-FPAC puede representarse como $O(D(K \times n_c)\log(DKn_c))$, lo que refleja un incremento proporcional al número de centroides utilizados, aunque mantiene la misma complejidad asintótica que FPAC.

En el caso de la complejidad computacional de Forward Improved-FPAC, considerando D como el número total de documentos del corpus, K el número de grupos y n_c el número de centroides por grupo en cada iteración. El algoritmo inicia ejecutando Improved-FPAC con $n_c = 10$ centroides por grupo. Durante este proceso, el costo principal surge de la consulta del índice invertido para calcular el puntaje de recuperación entre cada uno de los D documentos y los Kn_c centroides. Cada consulta requiere operaciones logarítmicas debido a las estructuras internas del índice invertido, por lo que la complejidad computacional del primer paso del algoritmo es:

$$O(D(Kn_c)\log(DKn_c)).$$

Una vez evaluada la calidad inicial, el algoritmo incrementa el número de centroides por grupo y repite el proceso de agrupamiento empleando nuevamente el paso 3 y 4 de Improved-FPAC. Al igual que en el paso anterior, el costo dominante proviene de las consultas a las listas invertidas para calcular los puntajes de recuperación entre los D documentos y los Kn'_c centroides de la iteración actual. En consecuencia, la complejidad computacional de este paso es:

$$O(D(Kn'_c)\log(DKn'_c)).$$

En la Comparación de la calidad del agrupamiento no se realizan consultas al índice invertido, ya que todos los valores del Puntaje de Recuperación Normalizado (PRN) fueron calculados en la fase previa y almacenados en una matriz de tamaño $D \times K$, donde cada fila corresponde a un documento y cada columna contiene el PRN del documento respecto a cada uno de los K grupos. El objetivo de este paso consiste

únicamente en calcular la varianza de los valores almacenados para cada clúster, recorriendo para ello las entradas correspondientes de la matriz. Dado que el cálculo de la varianza requiere recorrer una vez los D valores asociados a cada uno de los K grupos, el costo total de este paso es lineal respecto al tamaño de la matriz, es decir:

$$O(DK).$$

Al sumar la complejidad computacional de los pasos anteriores, el costo dominante de cada ciclo del proceso de Forward Improved-FPAC corresponde a la consulta al índice invertido necesaria para evaluar los D documentos frente a los Kn_c centroides de cada iteración, lo cual produce una complejidad de:

$$O(D(Kn_c)\log(DKn_c)),$$

Este resultado muestra que Forward Improved-FPAC mantiene la misma complejidad asintótica que Improved-FPAC.

Capítulo 5

Experimentos

El propósito de los experimentos mostrados en este capítulo es evaluar el desempeño de las dos alternativas propuestas para determinar el número de centroides en el algoritmo Improved-FPAC, CCM-CD y Forward Improved FPAC, con el fin de determinar si las estrategias propuestas permiten mejorar la calidad del agrupamiento y la eficiencia en el tiempo de ejecución del algoritmo Improved-FPAC. En este capítulo se plantean los experimentos, diseñados para medir, por un lado, la calidad de los grupos generados por cada algoritmo y, por otro, el costo computacional asociado a su ejecución.

5.1. Configuración de Experimentos

Los experimentos se realizaron sobre un conjunto de 20 corpus públicos (las características se muestran en la tabla 5.1) y emplea métricas ampliamente aceptadas en la literatura, tales como Rand Index (RI), Precision, Recall, F-Score, Purity y Normalized Mutual Information (NMI), junto con el tiempo total de ejecución. La finalidad de este estudio es comprobar si CCM-CD logra mejorar los resultados de Improved-FPAC mediante la estimación previa del número adecuado de centroides, y si Forward Improved-FPAC es capaz de superar a ambas alternativas mediante el ajuste dinámico del número de centroides. Los resultados obtenidos permitirán determinar la efectividad de las propuestas y verificar si ofrecen agrupamientos de mayor calidad y con tiempos de ejecución competitivos respecto a Improved-FPAC.

Se utilizó el algoritmo Improved-FPAC como referencia principal para evaluar las mejoras obtenidas por nuestra propuesta, ya que este es precisamente el

algoritmo que motivó la investigación doctoral de esta tesis. Además, Improved-FPAC constituye el punto de partida natural de la comparación porque, de acuerdo con los resultados reportados por Bejos et al. (2020), dicho algoritmo mostró obtener un desempeño superior al de diversos métodos de agrupamiento tradicionales y basados en recuperación de información, incluyendo aquellos contra los que fue comparado en su publicación original como FPAC True Centroids y FPAC descrito en Ganguly (2018). Del mismo modo, FPAC —su antecesor directo— demostró superar a otros algoritmos como K-Means y Skalable K-Means Centroids Bahmani et al. (2012); MacQueenet al. (1967), descritos en el estudio publicado Ganguly (2018), lo que posiciona a esta familia de métodos como una referencia sólida dentro del estado del arte.

5.1.1. Muestra

Para realizar los experimentos se utilizaron corpus de textos estándar etiquetados y publicados en la web y que han sido utilizados en los estudios previos relacionados a esta investigación, los cuales serán utilizados como entrada para la ejecución de cada uno de los algoritmos a comparar como para el método propuesto.

En la Tabla 5.1, se listan los corpus utilizados en nuestros experimentos para evaluar las mejoras obtenidas respecto a *Improved-FPAC*. Cada fila corresponde a un corpus distinto, mientras que las columnas muestran la siguiente información: el nombre del corpus, el número de documentos del corpus y el número de clases del corpus. Esta descripción detallada proporciona contexto para comprender la diversidad y la dificultad asociadas con cada corpus. Estos corpus, como ya se mencionó, fueron seleccionados porque constituyen conjuntos de datos públicos ampliamente empleados en la literatura de agrupamiento de documentos, lo que permite una comparación objetiva y replicable de los resultados. Además, presentan una diversidad en número de documentos, clases y vocabulario, lo que asegura una evaluación robusta de los algoritmos propuestos bajo distintos escenarios.

Cuadro 5.1: Corpus utilizados en los experimentos

Corpus	**Documentos**	**Clases**
03AGNews	89059	3

Continúa en la siguiente página

Corpus	Documentos	Clases
10DatasetClassification	1000	10
10newsgroups	9595	10
BBCNewsReviews10	6294	10
DocumentClasification	5485	8
German	10273	9
HealthTweets8	29141	8
IMDBMovieReviews	50000	2
MedicalDataset	14438	5
NewsCategory12	45846	12
NewsCategory16	57488	16
NewsCategory20	72924	20
NewsCategory4	13307	4
NewsCategory8	24045	8
Ohsumed	23166	23
R26	8578	26
R30	8679	30
Reuters10	1723	10
Reuters8	3410	8
TextClasificationOnEMail	9119	5

5.1.2. Métricas de evaluación

Para evaluar la calidad de los agrupamientos, empleamos múltiples métricas estándar y ampliamente utilizadas —F-Score, Rand Index (RI), Precisión, Recall, Pureza e Información Mutua Normalizada (NMI)—. Estas medidas se seleccionaron porque están bien establecidas en la literatura, son ampliamente reconocidas por la comunidad de investigación y permiten la comparación directa con trabajos previos Halkidi et al. (2002); Karypis et al. (2000); Manning (2008). El uso de estas métricas garantiza que nuestros resultados puedan evaluarse objetivamente y compararse con enfoques de agrupamiento relacionados. Estos resultados permitieron determinar la calidad de los agrupamientos obtenidos y verificar si los algoritmos propuestos superan las alternativas existentes. La intención del estudio es demostrar que nuestra versión arrojará agrupamientos de mejor calidad que los de estudios similares previos, realizando la medición con las variables antes mencionadas.

5.2. Evaluación de la calidad

En esta sección se presentan los experimentos realizados para evaluar la calidad de agrupamiento de *CCM-CD* Magallon Juan Qui et al. (2025) y *Forward Improved-FPAC*, las dos alternativas propuestas para detereminar el número de centroides en el algoritmo Improved-FPAC, comparados frente a *Improved-FPAC* Bejos et al. (2020).

La Tabla 5.2 presenta los valores promedio de Rand Index (RI), Precisión, Recall, F-Score, Pureza e Información Mutua Normalizada (NMI) obtenidos para cada corpus por los tres algoritmos evaluados: CCM-CD, Forward Improved-FPAC e Improved-FPAC. Para obtener estos resultados, cada algoritmo fue ejecutado de manera independiente sobre los 20 corpus seleccionados, aplicando en todos los casos el mismo esquema de preprocesamiento, la misma representación vectorial y los mismos parámetros base definidos previamente. Tras la ejecución de cada algoritmo sobre los corpus, se calcularon todas las métricas de evaluación mencionadas y se repitió el proceso 10 veces para cada corpus con cada algoritmo, para garantizar consistencia en los resultados obtenidos; posteriormente, se promediaron los resultados con el fin de minimizar variaciones producidas por inicializaciones o aleatoriedad inherente. Además, la tabla incluye el número promedio de centroides empleados por cada algoritmo, lo que permite analizar no solo la calidad del agrupamiento, sino también la cantidad de centroides utilizados para alcanzarla. Cada fila corresponde a un algoritmo aplicado a un corpus específico, mientras que las columnas indican la puntuación resultante de cada métrica de evaluación. El valor más alto de cada métrica dentro de un corpus se resalta en negritas para facilitar la comparación.

La Tabla 5.2 presenta los valores promedio de Rand Index (RI), Precisión, Recall, F-Score, Pureza e Información Mutua Normalizada (NMI) obtenidos para cada corpus y para: CCM-CD, Forward Improved-FPAC e Improved-FPAC. Además, la tabla incluye el número promedio de centroides empleados por cada alternativa, lo que permite analizar no solo la calidad de agrupamiento sino también el número de centroides requeridos para alcanzar dichos resultados. Cada fila corresponde a cada alternativa aplicada a un corpus específico, mientras que las columnas indican la puntuación de cada métrica de evaluación. El valor más alto de cada métrica dentro de un corpus se resalta en negritas para facilitar la comparación.

Cuadro 5.2: Valores promedio de RI, Precisión, Recall, F-Score, Pureza y NMI para cada corpus. El valor más alto de cada métrica dentro de un corpus se resalta en negritas.

Corpus/Alternativa	Avg L	RI	Precisión	Recall	F-Score	Pureza	NMI
03AGNews							
Forward Improved-FPAC	220	**0.8423**	**0.7619**	**0.7674**	**0.7646**	**0.8553**	**0.5989**
CCM-CD	164	0.8025	0.7022	0.7093	0.7058	0.8026	0.5238
Improved-FPAC	10	0.6418	0.4639	0.4748	0.4693	0.5786	0.1854
10DatasetClassification							
Forward Improved-FPAC	57.8	0.9083	0.5361	0.5836	0.5586	0.6684	0.6396
CCM-CD	59	**0.9174**	**0.5785**	**0.6264**	0.6013	**0.7012**	**0.6754**
Improved-FPAC	10	0.8804	0.4036	0.4308	0.4166	0.5442	0.4862
10newsgroups							
Forward Improved-FPAC	91	0.8884	0.4469	0.4660	0.4562	0.5781	0.4911
CCM-CD	137	**0.8961**	**0.4828**	**0.5016**	**0.4920**	**0.6127**	**0.5297**
Improved-FPAC	10	0.8709	0.3624	0.3795	0.3707	0.4941	0.4012
BBCNewsReviews10							
Forward Improved-FPAC	220	**0.8900**	**0.6313**	**0.5172**	**0.5664**	**0.7433**	**0.6833**
CCM-CD	119	0.8887	0.6243	0.5157	0.5627	0.7409	0.6787
Improved-FPAC	10	0.8674	0.5383	0.4517	0.4872	0.6767	0.6059
DocumentClasification							
Forward Improved-FPAC	140	**0.7290**	**0.7837**	0.3379	0.4711	0.7994	0.4439
CCM-CD	151	0.7192	0.7591	0.3183	0.4477	**0.8020**	**0.4601**
Improved-FPAC	10	0.7250	0.7342	**0.3595**	**0.4799**	0.7484	0.3572
German							
Forward Improved-FPAC	116	**0.8592**	**0.4433**	**0.4124**	**0.4272**	**0.5724**	**0.4115**
CCM-CD	94	0.8565	0.4321	0.4026	0.4168	0.5594	0.3951
Improved-FPAC	10	0.8414	0.3689	0.3462	0.3570	0.4984	0.3179
HealthTweets8							
Forward Improved-FPAC	220	0.8259	0.4246	0.3597	0.3895	0.5497	0.3598
CCM-CD	157	**0.8268**	**0.4294**	**0.3705**	**0.3978**	**0.5747**	**0.3714**
Improved-FPAC	10	0.8031	0.3371	0.2861	0.3095	0.4810	0.2350
IMDBMovieReviews							
Forward Improved-FPAC	10	0.5070	0.5069	0.5190	0.5129	0.5510	0.0105
CCM-CD	163	**0.5327**	**0.5320**	**0.5496**	**0.5406**	**0.6160**	**0.0497**
Improved-FPAC	10	0.5048	0.5047	0.5192	0.5118	0.5419	0.0072
MedicalDataset							
Forward Improved-FPAC	184	**0.6861**	**0.2983**	**0.2616**	**0.2787**	**0.4046**	**0.1106**
CCM-CD	159	0.6834	0.2921	0.2568	0.2733	0.4033	0.0968
Improved-FPAC	10	0.6821	0.2878	0.2518	0.2686	0.3931	0.0873
NewsCategory12							

Continúa en la siguiente página

Tabla 5.2 (continuación)

Corpus/Alternativa	Avg L	RI	Precisión	Recall	F-Score	Pureza	NMI
Forward Improved-FPAC	220	**0.7995**	**0.3640**	**0.1756**	**0.2369**	**0.5185**	**0.2543**
CCM-CD	158	0.7969	0.3516	0.1726	0.2315	0.5026	0.2360
Improved-FPAC	10	0.7870	0.2900	0.1390	0.1879	0.4481	0.1542
NewsCategory16							
Forward Improved-FPAC	220	**0.8564**	**0.3787**	**0.1947**	**0.2572**	**0.5076**	**0.2979**
CCM-CD	156	0.8520	0.3486	0.1827	0.2398	0.4825	0.2652
Improved-FPAC	10	0.8464	0.3009	0.1532	0.2031	0.4447	0.2115
NewsCategory20							
Forward Improved-FPAC	220	**0.8876**	**0.3189**	**0.1772**	**0.2278**	**0.4709**	**0.2857**
CCM-CD	155	0.8847	0.2960	0.1690	0.2152	0.4374	0.2518
Improved-FPAC	10	0.8796	0.2398	0.1322	0.1705	0.3882	0.2039
NewsCategory4							
Forward Improved-FPAC	220	**0.6901**	**0.5542**	**0.4181**	**0.47665**	**0.6837**	**0.2741**
CCM-CD	160	0.6772	0.5288	0.3991	0.4549	0.6560	0.2407
Improved-FPAC	10	0.6281	0.4317	0.3227	0.3694	0.5746	0.1054
NewsCategory8							
Forward Improved-FPAC	220	**0.8038**	**0.4133**	**0.3097**	**0.3541**	**0.5545**	**0.2947**
CCM-CD	158	0.7973	0.3892	0.2936	0.3347	0.5307	0.2680
Improved-FPAC	10	0.7765	0.3052	0.2252	0.2592	0.4566	0.1700
ohsumed							
Forward Improved-FPAC	220	**0.8877**	**0.1374**	**0.0810**	**0.1019**	**0.2613**	**0.1249**
CCM-CD	158	0.8871	0.1327	0.0788	0.0988	0.2545	0.1132
Improved-FPAC	10	0.8873	0.1286	0.0751	0.0948	0.2396	0.1031
r26							
Forward Improved-FPAC	52.2	0.7566	0.8249	0.1850	0.3013	0.7820	0.4692
CCM-CD	130	0.74341	0.80306	0.13328	0.2285	**0.79887**	**0.4767**
Improved-FPAC	10	**0.7679**	**0.8384**	**0.2271**	**0.3532**	0.7693	0.4559
r30							
Forward Improved-FPAC	42.8	0.7626	0.8353	0.1832	0.2990	0.7845	0.4760
CCM-CD	125	0.7521	0.8327	0.1376	0.2353	**0.8046**	**0.4913**
Improved-FPAC	10	**0.7781**	**0.8597**	**0.2396**	**0.3721**	0.7738	0.4717
Reuters10							
Forward Improved-FPAC	27	0.8470	0.2900	0.2930	0.2914	0.4315	0.3198
CCM-CD	103	**0.8688**	**0.3876**	**0.3789**	**0.3831**	**0.5238**	**0.4413**
Improved-FPAC	10	0.8423	0.2692	0.2735	0.2713	0.4104	0.2867
Reuters8							
Forward Improved-FPAC	10	0.8014	0.3494	0.3180	0.3328	0.4682	0.2868
CCM-CD	121	**0.8294**	**0.4458**	**0.3883**	**0.4149**	**0.5642**	**0.4341**
Improved-FPAC	10	0.7962	0.3305	0.2999	0.3143	0.4502	0.2588
TextClasificationOnEMail							

Continúa en la siguiente página

Tabla 5.2 (continuación)

Corpus/Alternativa	Avg L	RI	Precisión	Recall	F-Score	Pureza	NMI
Forward Improved-FPAC	10	0.6403	0.4190	0.2711	0.3291	0.5506	0.1145
CCM-CD	137	**0.6871**	**0.5295**	**0.3426**	**0.4159**	**0.6304**	**0.2595**
Improved-FPAC	10	0.6330	0.4011	0.2578	0.3138	0.5340	0.0894

La comparación entre las tres alternativas está basada exclusivamente en los valores promedio de RI, Precisión, Recall, F-Score, Pureza y NMI reportados en la Tabla 5.2, muestra que Forward Improved-FPAC obtiene las mejores puntuaciones en la mayoría de los corpus evaluados. En particular, el algoritmo alcanza el valor más alto en estas métricas en **11 de los 20 corpus analizados**, lo que evidencia un desempeño superior en comparación con CCM-CD e Improved-FPAC.

Por su parte, *CCM-CD* presenta la mejor calidad en 7 de los 20 corpus, superior al de *Forward Improved-FPAC*, donde *CCM-CD* es mejor en las métricas como NMI y Pureza, y en ocasiones la totalidad de los indicadores.

Finalmente, *Improved-FPAC* —que opera con $L = 10$— solo obtiene una mejor calidad en 2 corpus, en donde obtiene los mayores promedios de RI, Precisión, Recall y F-Score dentro de cada corpus. Fuera de estos corpus, sus promedios suelen situarse por debajo de *Forward Improved-FPAC* y *CCM-CD*.

Para determinar si las diferencias observadas en los promedios de las métricas corresponden a mejoras estadísticamente significativas y no a variaciones aleatorias entre ejecuciones, a continuación se aplica la prueba de rangos con signo de Wilcoxon sobre los valores de *F-Score* emparejados por corpus y por corrida. Esta prueba no asume normalidad y es adecuada para comparar dos algoritmos evaluados bajo las mismas instancias experimentales. Adoptamos un umbral de significancia de $p < 0.05$ y reportamos los p-valores junto con el algoritmo que obtiene el mayor promedio. Los resultados de las comparaciones *Forward Improved-FPAC vs. CCM-CD* y *Forward Improved-FPAC vs. Improved-FPAC* se presentan en las Tablas 5.3 y 5.4, respectivamente, mientras que la comparación *CCM-CD vs. Improved-FPAC* se muestra en la Tabla 5.5. Con base en esta prueba, se identificó el algoritmo mejor calificado para cada corpus analizado. Aunque la prueba de Wilcoxon se aplicó a todas las métricas de evaluación (RI, Precisión, Recall, Pureza, NMI y F-Score), los resultados fueron consistentes entre métricas, conduciendo a las mismas conclusiones. Por razones de espacio, solo reportamos los resultados de F-Score en el texto

principal, mientras que el conjunto completo de resultados de la prueba de Wilcoxon se encuentra disponible en línea [1].

Para los corpus donde la diferencia de desempeño entre los dos algoritmos fue estadísticamente significativa (*p-value*<0.05), el F-Score promedio más alto de la comparación se resalta en negritas, indicando el algoritmo que arrojó el mayor valor. Si no hay valores en negritas en una fila, significa que la prueba de Wilcoxon no encontró una diferencia estadísticamente significativa en los resultados de F-Score para ese corpus. Adicionalmente, el *p-value* también se resalta en negritas cuando indica que la diferencia es estadísticamente significativa.

La Tabla 5.3 presenta los resultados de la prueba de rangos con signo de Wilcoxon que compara los algoritmos Forward Improved-FPAC y CCM-CD utilizando valores de F-Score en 20 corpus. Esta tabla incluye cuatro columnas: el nombre del corpus, el F-Score promedio obtenido por Forward Improved-FPAC, el F-Score promedio obtenido por CCM-CD y el *p-value* correspondiente de la prueba de Wilcoxon.

Cuadro 5.3: Resultados de la prueba de Wilcoxon (F-Score) entre Forward Improved-FPAC y CCM-CD (diferencias significativas en negritas).

Corpus	Forward Improved-FPAC	CCM-CD	p-value
03AGNews	**0.76467**	0.70582	**0.01953**
10DatasetClassification	0.55864	**0.60137**	**0.04883**
10newsgroups	0.45627	0.49201	0.19336
BBCNewsReviews10	0.56645	0.56274	0.43164
DocumentClasification	**0.47113**	0.44770	**0.01953**
German	**0.42727**	0.41682	**0.00977**
HealthTweets8	0.38950	0.39780	0.49219
IMDBMovieReviews	0.51290	**0.54067**	**0.00195**
MedicalDataset	**0.27876**	0.27330	**0.01953**
NewsCategory12	0.23691	0.23157	0.06445
NewsCategory16	**0.25721**	0.23981	**0.02734**

Continúa en la siguiente página

[1] https://github.com/intisand/ForwardImproved-FPAC/blob/main/Experiment_wilcoxon_20_dataset.xlsx

Corpus	Forward Improved-FPAC	CCM-CD	p-value
NewsCategory20	**0.22789**	0.21525	**0.00391**
NewsCategory4	**0.47665**	0.45494	**0.01367**
NewsCategory8	**0.35414**	0.33475	**0.00977**
ohsumed	**0.10199**	0.09889	**0.01367**
r26	**0.30133**	0.22850	**0.00391**
r30	**0.29908**	0.23539	**0.04883**
Reuters10	0.29148	**0.38316**	**0.00195**
Reuters8	0.33280	**0.41495**	**0.00195**
TextClasificationOnEMail	0.32918	**0.41598**	**0.00195**

En términos generales, los resultados de la prueba de Wilcoxon indican un desempeño favorable de *Forward Improved-FPAC* frente a *CCM-CD* en una mayoría de corpus, pero con excepciones claras. En 11 de 20 conjuntos, *Forward Improved-FPAC* obtuvo el mayor F-Score y la diferencia fue estadísticamente significativa $p \leq 0.02734$). Estos casos sugieren que la búsqueda hacia adelante para ajustar el número de centroides por grupo ofrece una ventaja consistente.

Por otro lado, *CCM-CD* superó significativamente a *Forward Improved-FPAC* en 5 corpus $p \leq 0.00195$). En estos escenarios, el esquema de determinación de centroides de *CCM-CD* parece funcionar mejor en colecciones con menor número de clases.

Finalmente, en 4 corpus no se observaron diferencias estadísticamente significativas (*10newsgroups*, *BBCNewsReviews10*, *HealthTweets8*, *NewsCategory12*), lo que apunta a un comportamiento comparable entre ambos métodos. En conjunto, los hallazgos muestran un patrón 11–5–4 (victorias de *Forward Improved-FPAC*–victorias de *CCM-CD*–empates), respaldando a *Forward Improved-FPAC* como alternativa preferente cuando se prioriza la calidad promedio del agrupamiento.

Cuadro 5.4: Resultados de la prueba de Wilcoxon (F-Score) entre Forward Improved-FPAC e Improved-FPAC (diferencias significativas en negritas).

Corpus	Forward Improved-FPAC	Improved-FPAC	p-value
03AGNews	**0.76467**	0.46934	**0.00195**
10DatasetClassification	**0.55864**	0.41666	**0.00195**
10newsgroups	**0.45627**	0.37071	**0.00195**
BBCNewsReviews10	**0.56645**	0.48726	**0.00195**
DocumentClassification	0.47113	0.47996	0.84570
German	**0.42727**	0.35709	**0.00195**
HealthTweets8	**0.38950**	0.30955	**0.00195**
IMDBMovieReviews	0.51290	0.51188	0.23242
MedicalDataset	**0.27876**	0.26864	**0.00977**
NewsCategory12	**0.23691**	0.18794	**0.00195**
NewsCategory16	**0.25721**	0.20310	**0.00195**
NewsCategory20	**0.22789**	0.17050	**0.00195**
NewsCategory4	**0.47665**	0.36941	**0.00195**
NewsCategory8	**0.35414**	0.25923	**0.00195**
ohsumed	**0.10199**	0.09484	**0.00195**
r26	0.30133	0.35324	0.16016
r30	0.29908	**0.37216**	**0.01367**
Reuters10	**0.29148**	0.27135	**0.00195**
Reuters8	**0.33280**	0.31431	**0.00195**
TextClasificationOnEMail	**0.32918**	0.31386	**0.00195**

En conjunto, los resultados evidencian una ventaja clara de *Forward Improved-FPAC* frente a *Improved-FPAC* en términos de F-Score. En 16 de los 20 corpus evaluados, *Forward Improved-FPAC* obtuvo promedios significativamente superiores ($p \leq 0.00977$). Estas ganancias respaldan la eficacia del esquema de búsqueda hacia adelante para ajustar dinámicamente el número de centroides por grupo y adaptarse a la estructura del corpus.

Existen tres casos sin diferencias estadísticamente significativas: En ellos, los promedios son cercanos y no permiten afirmar la superioridad de un método sobre el otro. Finalmente, solo en un corpus *Improved-FPAC* supera significativamente a *For-*

ward Improved-FPAC ($p = 0.01367$).

En síntesis, la evidencia apoya a *Forward Improved-FPAC* como una mejor opción frente a Improved-FPAC y CCM-CD cuando se busca obtener una mejor calidad promedio del agrupamiento, conservando robustez ante variaciones entre corpus.

Cuadro 5.5: Resultados de la prueba de Wilcoxon (F-Score) entre CCM-CD e Improved-FPAC (diferencias significativas en negritas).

Corpus	CCM-CD	Improved-FPAC	p-value
03AGNews	**0.70582**	0.46934	**0.00195**
10DatasetClassification	**0.60137**	0.41666	**0.00195**
10newsgroups	**0.49201**	0.37071	**0.00195**
BBCNewsReviews10	**0.56274**	0.48726	**0.00195**
DocumentClasification	0.44770	0.47996	0.10547
German	**0.41682**	0.35709	**0.00195**
HealthTweets8	**0.39780**	0.30955	**0.00195**
IMDBMovieReviews	**0.54067**	0.51188	**0.00195**
MedicalDataset	**0.27330**	0.26864	**0.00195**
NewsCategory12	**0.23157**	0.18794	**0.00195**
NewsCategory16	**0.23981**	0.20310	**0.00195**
NewsCategory20	**0.21525**	0.17050	**0.00195**
NewsCategory4	**0.45494**	0.36941	**0.00195**
NewsCategory8	**0.33475**	0.25923	**0.00195**
ohsumed	**0.09889**	0.09484	0.01953
r26	0.22850	**0.35324**	**0.00195**
r30	0.23539	**0.37216**	**0.00195**
Reuters10	**0.38316**	0.27135	**0.00195**
Reuters8	**0.41495**	0.31431	**0.00195**
TextClasificationOnEMail	**0.41598**	0.31386	**0.00195**

Por otro lado, los resultados muestran una ventaja consistente de *CCM-CD* sobre *Improved-FPAC* en la mayoría de los corpus. En 17 de 20 conjuntos, el F-Score promedio es mayor para *CCM-CD* y las diferencias son estadísticamente significativas en casi todos los casos ($p \leq 0.01953$).

En conjunto, la evidencia respalda a *CCM-CD* como una alternativa preferente frente a Improved-FPAC, cuando el objetivo principal es maximizar la calidad promedio de agrupamiento (F-Score). Sin embargo, los casos en que *Improved-FPAC* resulta superior (o no es significativamente diferente) sugieren que la elección del método puede beneficiarse de considerar las características del corpus (número de clases, número de documentos y términos) y el comportamiento observado en validaciones preliminares.

5.3. Evaluación de tiempo de ejecución

El análisis de tiempo de ejecución complementa la evaluación de calidad al ofrecer una medida del costo computacional asociado a cada alternativa. En esta sección se compara el tiempo de ejecución de *Forward Improved-FPAC*, *CCM-CD* e *Improved-FPAC* bajo la misma configuración experimental descrita en la Sección 5.1. El tiempo reportado corresponde al promedio (en segundos) de 10 ejecuciones independientes para cada corpus, con cada algoritmo. Este análisis permite identificar el costo de la ejecución de cada alternativa, así como el efecto práctico de utilizar un número de centroides por grupo (L) fijo o ajustándolo durante la construcción del agrupamiento. A continuación, la Figura 5.1 resume los tiempos promedio por corpus y la Tabla 5.6 detalla, además, el L promedio utilizado por cada método.

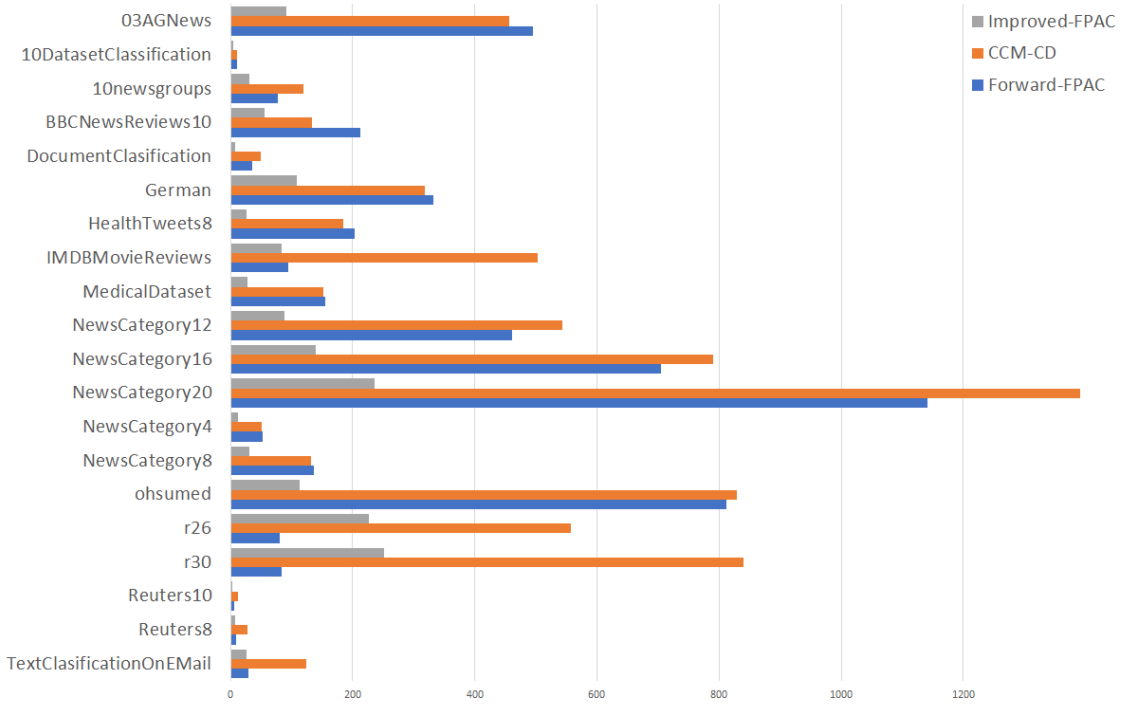

Figura 5.1: Tiempo de ejecución promedio por corpus para Forward Improved-FPAC, CCM-CD e Improved-FPAC.

En general, la Figura 5.1 muestra que Improved-FPAC usualmente logra tiempos de ejecución más cortos para la mayoría de los corpus, mientras que Forward Improved-FPAC y CCM-CD tienden a requerir sustancialmente más tiempo, particularmente para corpus más grandes o complejos. En varios casos, el tiempo de ejecución de Forward Improved-FPAC es entre 3 y 5 veces mayor que el de Improved-FPAC, mientras que CCM-CD puede ser hasta un orden de magnitud más lento. No obstante, estos tiempos de ejecución se mantienen aceptables en la práctica para escenarios de agrupamiento de documentos *offline*, donde el agrupamiento se realiza típicamente una sola vez y luego se reutiliza para tareas de recuperación o análisis.

Cuadro 5.6: Tiempo de ejecución promedio y número de centroides por grupo para cada corpus.

Corpus	Forward Improved-FPAC		CCM-CD		Improved-FPAC	
	Centroides	Tiempo	Centroides	Tiempo	Centroides	Tiempo
03AGNews	220	495.5	164	455.6	10	91.2
10DatasetClassification	57.8	10.3	59	10.3	10	4.7
10newsgroups	91	77.5	137	118.9	10	30.6
BBCNewsReviews10	220	212.5	119	133.1	10	55.2
DocumentClasification	140	34.8	151	49.4	10	7.6
German	116	332.3	94	317.5	10	108
HealthTweets8	220	202.8	157	184.8	10	26.1
IMDBMovieReviews	10	94.2	163	502.5	10	84
MedicalDataset	184	154.4	159	152.6	10	27.4
NewsCategory12	220	460.9	158	543.3	10	89
NewsCategory16	220	704.3	156	790.7	10	140.3
NewsCategory20	220	1141.3	155	1392.7	10	235.8
NewsCategory4	220	53	160	50.3	10	12
NewsCategory8	220	136.1	158	131.4	10	31.5
ohsumed	220	812.1	147	828.3	10	112.8
r26	52.2	80.2	130	558	10	226.2
r30	42.8	83.8	125	839.7	10	252
Reuters10	27	5.6	103	12.7	10	3.6
Reuters8	10	8.6	121	28.3	10	7.6
TextClasificationOnEMail	10	28.7	137	124.1	10	26.8
Total Runtime		5,128.9		7,224.2		1,572.4

La Tabla 5.6 presenta el número promedio de centroides y el tiempo de ejecución promedio obtenidos para cada uno de los 20 corpus evaluados, considerando las tres alternativas de agrupamiento: Forward Improved-FPAC, CCM-CD e Improved-FPAC.

Las diferencias en el tiempo de ejecución de las tres alternativas pueden atribuirse al diseño de cada algoritmo. Improved-FPAC utiliza un número pequeño y fijo de centroides por grupo, lo que reduce la carga computacional. Forward Improved-FPAC, por otro lado, emplea una estrategia de búsqueda hacia adelante que ajusta iterativamente el número de centroides, aumentando así el costo computacional pero mejorando la calidad del agrupamiento. CCM-CD, aunque no realiza ajustes durante la ejecución, muestra un mayor tiempo total porque todas sus iteraciones se llevan a cabo con el mismo número de centroides por grupo, que puede ser alto. En con-

traste, Forward Improved-FPAC inicia sus primeras iteraciones con un bajo número de centroides por grupo que aumenta gradualmente; como resultado, el tiempo de ejecución de las iteraciones iniciales es menor que el de CCM-CD.

En términos de tiempo de ejecución, Forward Improved-FPAC requirió un promedio de 256.4 segundos por corpus, con un tiempo total de 5,128.9 segundos. Aunque este tiempo de ejecución es mayor que el de Improved-FPAC, Forward Improved-FPAC ofreció consistentemente una calidad de agrupamiento superior a través de los corpus. Si bien Improved-FPAC requirió menos tiempo de ejecución (78.6 segundos por corpus en promedio, 1,572.4 segundos en total), esta eficiencia se obtuvo a expensas de una menor calidad de agrupamiento. Finalmente, CCM-CD fue el método más costoso en tiempo, alcanzando un promedio de 361.2 segundos por corpus y un tiempo total de 7,224.2 segundos; no obstante, dicho costo adicional conlleva beneficios tangibles: en varios corpus consiguió los mejores valores de Pureza, NMI y F-Score.

5.4. Observaciones finales

En este capítulo se presentaron y analizaron los resultados experimentales de las alternativas propuestas para determinar el número de centroides por cluster *Forward Improved-FPAC* y *CCM-CD* sobre el mismo conjunto de corpus, considerando métricas externas de calidad (RI, Precisión, Recall, F-Score, Pureza y NMI) y el tiempo de ejecución. Los promedios por corpus mostraron que *Forward Improved-FPAC* ofrece, con mayor frecuencia, la mejor calidad global; *CCM-CD* resulta competitivo e incluso superior en un conjunto de corpus. La validez de estas diferencias se corroboró mediante pruebas de Wilcoxon, donde *Forward Improved-FPAC* superó significativamente a *Improved-FPAC* en la mayoría de los corpus y mostró un balance favorable frente a *CCM-CD*, mientras que este último obtuvo ventajas significativas en varios conjuntos específicos. Finalmente, en términos generales, los resultados de tiempo muestran que Forward Improved-FPAC incrementa el costo computacional debido a su proceso iterativo de ajuste del número de centroides; no obstante, este incremento se mantiene dentro de rangos manejables en la práctica. En promedio, Forward Improved-FPAC requiere entre 3 y 5 veces más tiempo que Improved-FPAC, pero este aumento resulta razonable considerando las mejoras en la calidad del agrupamiento que el método logra de manera consistente. Así,

el balance entre costo temporal y calidad favorece su uso en escenarios donde la precisión del agrupamiento es prioritaria y los tiempos adicionales de ejecución no representan una limitación significativa. Por su parte, CCM-CD resulta ser la alternativa más costosa, ya que de acuerdo a nuestros experimentos, tiende a trabajar con un mayor número de centroides desde el inicio de su ejecución. En conjunto, estos resultados confirman que las mejoras en calidad obtenidas por Forward-FPAC implican un costo moderado, y que CCM-CD ofrece una mejora en calidad respecto a Improved-FPAC, pero con un mayor tiempo de ejecución que Forward Improved-FPAC.

A diferencia del método CCM-CD, que calcula un valor para el número de centroides antes de comenzar el agrupamiento y lo usa para construir el agrupamiento aplicando Improved-FPAC, en cambio, Forward Improved-FPAC comienza con un número bajo de centroides y este valor se incrementa progresivamente durante la ejecución mediante una búsqueda hacia adelante. Esto tiene un efecto importante en el costo computacional general del algoritmo. Aunque Forward Improved-FPAC realiza múltiples iteraciones con un número variable de centroides, las primeras iteraciones tienen un bajo costo, ya que de acuerdo a nuestros experimentos, el número de centroides involucrado puede ser menor que la estimación calculada por CCM-CD. En consecuencia, aunque Forward Improved-FPAC realiza ajustes iterativos aumentando progresivamente el número de centroides, su costo computacional total puede ser menor que el de CCM-CD. Esto se debe a que CCM-CD usa directamente un número fijo y potencialmente alto de centroides desde el inicio. Dado que este número permanece constante durante todo el proceso, el costo se acumula linealmente y puede superar el costo de Forward Improved-FPAC, que se detiene tan pronto como no se observan más mejoras.

Capítulo 6

Conclusiones y trabajo futuro

6.1. Propósito y alcance

La finalidad de esta investigación fue investigar sobre la Determinación del Número de Centroides en el Algoritmo Improved-FPAC Para el Agrupamiento de Documentos de Texto, un algoritmo que de acuardo a la literatura ha mostrado buenos resultados para el agrupamiento de documentos. Como producto de la investigación se desarrollaron dos alternativas —*CCM-CD* y *Forward Improved-FPAC*—. A lo largo del estudio, ambas alternativas se contrastaron empíricamente contra *Improved-FPAC* en varios corpus estándar y bajo seis métricas externas (RI, Precisión, Recall, F-Score, Pureza y NMI), complementando el análisis con pruebas de Wilcoxon y evaluación de tiempos de ejecución.

6.2. Síntesis de hallazgos

Los resultados muestran que las alternativas propuestas, Forward Improved-FPAC y CCM-CD, son las que con mayor frecuencia alcanzan la mejor calidad de agrupamiento. Sin embargo, su desempeño no es uniforme en todos los corpus, sino que depende de las características del conjunto de datos. En particular, Forward Improved-FPAC tiende a obtener los mejores resultados en corpus grandes o cuando se requiere construir un número elevado de agrupamientos, donde el ajuste dinámico del número de centroides permite capturar mejor la estructura de los agrupamientos (por ejemplo, 03AGNews, NewsCategory12/16/20, NewsCategory8 o HealthTweets8). En contraste, CCM-CD resulta más competitivo en corpus pequeños o de dimensión moderada —como Reuters10, Reuters8, TextClassificationOnE-

mail o 10DatasetClassification— donde el número estimado de centroides calculado mediante el modelo de regresión es suficiente para representar adecuadamente la distribución de los documentos sin necesidad de explorar múltiples número de centroides. Esta diferenciación sugiere que Forward Improved-FPAC es más adecuado para corpus extensos o con mayor número de términos, mientras que CCM-CD constituye una alternativa eficiente cuando se trata de colecciones más reducidas o con menor variabilidad en su contenido.

Por su parte, *Improved-FPAC* conservó tiempos de ejecución considerablemente inferiores a las alternativas propuestas. En prácticamente todos los corpus, su tiempo fue entre **3 y 6 veces menor** que el de *Forward Improved-FPAC* y entre **4 y 7 veces menor** que el de *CCM-CD*, lo que confirma que la ausencia de ciclos adicionales y el uso fijo de un número reducido de centroides permiten mantener un costo computacional mucho más bajo. Sin embargo, esta ganancia en eficiencia se ve acompañada de una pérdida significativa de calidad frente a las dos alternativas propuestas, diferencia que fue confirmada mediante la prueba de Wilcoxon ($p < 0.05$) aplicada sobre las métricas de evaluación.

6.3. Contribución

Partiendo de *Improved-FPAC* como base, esta investigación contribuye con dos alternativas para la determinación del número de centroides para el algoritmo Improved-FPAC para mejorar la calidad de los agrupamientos:

- **Forward Improved-FPAC**: incorpora una estrategia de ajuste progresivo del número de centroides por grupo, logrando en promedio los mayores valores **en las seis métricas de calidad** (F-Score, RI, Precisión, Recall, Pureza y NMI) en un conjunto amplio de corpus. Esta mejora se debe al ajuste dinámico del número de centroides, que permite definir o estructurar mejor los agrupamientos de documentos durante la búsqueda secuencial.
- **CCM-CD**: estima un número de centroides a partir de un modelo de regresión que, en varios corpus, produce resultados **competitivos o superiores en las seis métricas**, superando sistemáticamente a *Improved-FPAC* y, en algunos casos, igualando o superando a *Forward Improved-FPAC*. Este comportamiento confirma que una estimación previa del número de centroides puede ser efectiva para mejorar la calidad del agrupamiento.

- **Tiempo de ejecución de las alternativas**: además de la calidad, se investigó detalladamente el costo computacional de cada algoritmo, encontrándose que *Improved-FPAC* es la alternativa más eficiente en tiempo debido a que utiliza un número fijo y reducido de centroides; *Forward Improved-FPAC* incrementa su costo por los ciclos adicionales que ejecuta para ajustar dinámicamente n_c; y *CCM-CD* presenta el mayor tiempo de ejecución, ya que suele operar desde el inicio con un número de centroides más alto. Estos resultados confirman que las alternativas con mejor calidad conllevan un **mayor costo temporal**.

6.4. Conclusión final

El trabajo desarrollado parte de un problema fundamental en los algoritmos de agrupamiento basados en recuperación de información: la determinación adecuada del número de centroides por grupo en el caso particular de Improved-FPAC, este parámetro tiene un impacto directo tanto en la calidad del agrupamiento como en el costo computacional, y hasta ahora no existía una alternativa para estimarlo o ajustarlo. Sobre esta base, la presente investigación se propuso diseñar y evaluar alternativas que permitieran mejorar la calidad del agrupamiento sin perder de vista la eficiencia del proceso.

Los resultados obtenidos demuestran que este objetivo se cumplió satisfactoriamente. Las dos variantes propuestas —*CCM-CD* y *Forward Improved-FPAC*— superan de manera consistente a *Improved-FPAC* en las seis métricas de calidad evaluadas (RI, Precisión, Recall, F-Score, Pureza y NMI). *Forward Improved-FPAC* destaca particularmente en corpus de gran tamaño o alta complejidad, donde su estrategia de búsqueda hacia adelante del número de centroides que permite encontrar una opción sin utilizar un entrenamiento previo. Por su parte, *CCM-CD* ofrece una alternativa viable, especialmente en corpus de menor tamaño, con la ventaja adicional de no requerir más iteraciones. Ambos algoritmos constituyen mejoras claras respecto al método de referencia.

En términos computacionales, el estudio también permite encontrar una relación entre calidad y tiempo de procesamiento. Mientras que *Improved-FPAC* sigue siendo la opción más eficiente computacionalmente, las alternativas propuestas introducen incrementos en el tiempo de ejecución que resultan razonables considerando las mejoras obtenidas en calidad. Esta relación entre calidad y costo computacional es

relevante para orientar la elección del algoritmo en función del tamaño y características del corpus a procesar.

Finalmente, las contribuciones de este trabajo abren diversas líneas de investigación futura. Entre ellas se encuentra la posibilidad de automatizar aún más la selección del número de centroides mediante enfoques estadísticos. En conjunto, los resultados obtenidos no solo fortalecen las capacidades de Improved-FPAC, sino que también ofrecen herramientas y rutas prometedoras para avanzar en la construcción de métodos de agrupamiento más robustos, precisos y adaptativos.

6.5. Trabajo futuro

A partir de los hallazgos y limitaciones identificadas, se delinean varias líneas concretas para continuar esta investigación:

Como trabajo futuro, proponemos en primer lugar estudiar el impacto de la inicialización de centroides en los algoritmos de agrupamiento basados en recuperación de información y también la actualización de centroides en este tipo de algoritmos. Esto implica modificar un elemento a la vez —por ejemplo, la estrategia de inicialización, la actualización de centroides o las heurísticas de recuperación— y observar su efecto específico en las métricas de evaluación. Este análisis controlado facilitará identificar los componentes más influyentes y proponer mejoras con mayor impacto.

En segundo lugar, resulta pertinente profundizar en *eficiencia y escalamiento*. La incorporación de paralelización en CPU/GPU y de cómputo distribuido, así como la evaluación de estructuras de índice y búsqueda eficiente (listas invertidas optimizadas, *block-max*/WAND y métodos de vecinos aproximados, ANN), puede reducir de forma sustancial los tiempos de ejecución sin sacrificar calidad, haciendo viable el procesamiento de colecciones más grandes.

En tercer lugar, proponemos simplificar la *consulta de recuperación* usando menos palabras sin perder capacidad para distinguir entre grupos. En concreto, se puede seleccionar solo los términos más informativos del centroide (por ejemplo, según su peso TF–IDF) y fijar un límite máximo de términos por consulta. Con ello se espera reducir accesos al índice y comparaciones, disminuyendo el tiempo de ejecución.

La evaluación debe medir cómo afecta esta reducción a la calidad del agrupamiento (F-Score, RI, Precisión, Recall, Pureza y NMI) y al tiempo, para encontrar el mejor equilibrio entre ambos.

Finalmente, proponemos una línea de *integración aplicada* orientada a evaluar la utilidad de los agrupamientos en tareas, como recuperación de información, recomendación y resumen, así como su transferencia a conjuntos de datos de uso real en industria o instituciones. Este enfoque permitirá validar el valor práctico de los métodos más allá de los indicadores de laboratorio y orientar su adopción en contextos operativos.

Referencias

Abualigah, L., Gandomi, A. H., Elaziz, M. A., Hussien, A. G., Khasawneh, A. M., Alshinwan, M., & Houssein, E. H. (2020). Nature-inspired optimization algorithms for text document clustering–a comprehensive analysis. *Algorithms*, *13*(12), 345.

Agarwal, N., Sikka, G., & Awasthi, L. K. (2022). A systematic literature review on web service clustering approaches to enhance service discovery, selection and recommendation. *Computer Science Review*, *45*, 100498.

Aggarwal, C. C. & Zhai, C. (2012). A survey of text classification algorithms. In *Mining text data* (pp. 163–222). Springer.

Alghamdi, H. M. & Selamat, A. (2019). Arabic web page clustering: A review. *Journal of King Saud University-Computer and Information Sciences*, *31*(1), 1–14.

Ali, W., Khamis, S., & Zakaria, W. (2024). A novel approach for parallel document clustering using an enhanced parallel wand algorithm. *International Journal of Intelligent Engineering & Systems*, *17*(6).

Ankerst, M., Breunig, M. M., Kriegel, H.-P., & Sander, J. (1999). Optics: Ordering points to identify the clustering structure. *ACM Sigmod record*, *28*(2), 49–60.

Arco, L., Bello, R., Mederos, J. M., & Pérez, Y. (2006). Agrupamiento de documentos textuales mediante métodos concatenados. *Inteligencia Artificial. Revista Iberoamericana de Inteligencia Artificial*, *10*(30), 43–53.

Bahmani, B., Moseley, B., Vattani, A., Kumar, R., & Vassilvitskii, S. (2012). Scalable k-means++. *arXiv preprint arXiv:1203.6402*.

Bejos, S., Feliciano-Avelino, I., Martínez-Trinidad, J. F., & Carrasco-Ochoa, J. (2020). Improved fast partitional clustering algorithm for text clustering. *Journal of Intelligent & Fuzzy Systems*, *39*(2), 2137–2145.

Bezdan, T., Stoean, C., Naamany, A. A., Bacanin, N., Rashid, T. A., Zivkovic, M., & Venkatachalam, K. (2021a). Hybrid fruit-fly optimization algorithm with k-means for text document clustering. *Mathematics*, *9*(16).

Bezdan, T., Stoean, C., Naamany, A. A., Bacanin, N., Rashid, T. A., Zivkovic, M., & Venkatachalam, K. (2021b). Hybrid fruit-fly optimization algorithm with k-means for text document clustering. *Mathematics*, *9*(16), 1929.

Bezdek, J. C., Ehrlich, R., & Full, W. (1984). Fcm: The fuzzy c-means clustering algorithm. *Computers & geosciences*, *10*(2-3), 191–203.

Cozzolino, I. & Ferraro, M. B. (2022). Document clustering. *Wiley Interdisciplinary Reviews: Computational Statistics*, e1588.

Dhillon, I. S. & Modha, D. S. (2001). Concept decompositions for large sparse text data using clustering. *Machine learning*, *42*(1), 143–175.

Dice, L. R. (1945). Measures of the amount of ecologic association between species. *Ecology*, *26*(3), 297–302.

Dobrakowski, A. G., Mykowiecka, A., Marciniak, M., Jaworski, W., & Biecek, P. (2021). Interpretable segmentation of medical free-text records based on word embeddings. *Journal of Intelligent Information Systems*, *57*, 447–465.

Dodda, R. & Babu, A. S. (2024). Text document clustering using modified particle swarm optimization with k-means model. *International Journal on Artificial Intelligence Tools*, *33*(01), 2350061.

Eligüzel, N., Çetinkaya, C., & Dereli, T. (2022). A novel approach for text categorization by applying hybrid genetic bat algorithm through feature extraction and feature selection methods. *Expert Systems with Applications*, *202*, 117433.

Ganguly, D. (2018). A fast partitional clustering algorithm based on nearest neighbours heuristics. *Pattern Recognition Letters*, *112*, 198–204.

Garg, N. & Gupta, R. (2018). Performance evaluation of new text mining method based on ga and k-means clustering algorithm. In *Advanced Computing and Communication Technologies* (pp. 23–30). Springer.

Haji, S. H., Jacksi, K., & Salah, R. M. (2023). A semantics-based clustering approach for online laboratories using k-means and hac algorithms. *Mathematics*, *11*(3).

Halkidi, M., Batistakis, Y., & Vazirgiannis, M. (2001). On clustering validation techniques. *Journal of intelligent information systems*, *17*, 107–145.

Halkidi, M., Batistakis, Y., & Vazirgiannis, M. (2002). Cluster validity methods: part i. *ACM Sigmod Record*, *31*(2), 40–45.

Hirsch, L., Hirsch, R., & Ogunleye, B. (2025). Document clustering with evolved multi-word search queries. *Evolutionary Intelligence*, *18*(2), 37.

Inje, B., Nagwanshi, K. K., & Rambola, R. K. (2023). An efficient document information retrieval using hybrid global search optimization algorithm with density based clustering technique. *Cluster Computing*, 1–17.

Jaccard, P. (1901). Étude comparative de la distribution florale dans une portion des alpes et des jura. *Bull Soc Vaudoise Sci Nat*, *37*, 547–579.

Jain, A. K. & Dubes, R. C. (1988). *Algorithms for clustering data*. Prentice-Hall, Inc.

Jain, A. K., Murty, M. N., & Flynn, P. J. (1999). Data clustering: a review. *ACM computing surveys (CSUR)*, *31*(3), 264–323.

Karypis, M. S. G., Kumar, V., & Steinbach, M. (2000). A comparison of document clustering techniques. In *TextMining Workshop at KDD2000 (May 2000)*, (pp. 428–439).

Kaufman, L. & Rousseeuw, P. J. (2009). *Finding groups in data: an introduction to cluster analysis*. John Wiley & Sons.

Kim, H., Kim, H. K., & Cho, S. (2020). Improving spherical k-means for document clustering: Fast initialization, sparse centroid projection, and efficient cluster labeling. *Expert Systems with Applications*, *150*, 113288.

Kriegel, H.-P., Kröger, P., Sander, J., & Zimek, A. (2011). Density-based clustering. *Wiley interdisciplinary reviews: data mining and knowledge discovery*, *1*(3), 231–240.

Liu, T.-Y. et al. (2009). Learning to rank for information retrieval. *Foundations and Trends® in Information Retrieval*, *3*(3), 225–331.

Liu, W., Sun, Y., Yu, B., Wang, H., Peng, Q., Hou, M., Guo, H., Wang, H., & Liu, C. (2024). Automatic text summarization method based on improved textrank algorithm and k-means clustering. *Knowledge-Based Systems*, *287*, 111447.

MacQueen, J. et al. (1967). Some methods for classification and analysis of multivariate observations. In *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, volume 1, (pp. 281–297). Oakland, CA, USA.

Magallon Juan Qui, I. S., Martinez Trinidad, J. F., Vilarino Ayala, D., & Carrasco Ochoa, J. A. (2025). Corpus Characteristics-Based Method to Centroids Number Determination for Clustering Text Documents. *Journal of Scientometric Research*, *14*(1), 319–330.

Malik, F., Khan, S., Rizwan, A., Atteia, G., & Samee, N. A. (2022). A novel hybrid clustering approach based on black hole algorithm for document clustering. *IEEE Access*, *10*, 97310–97326.

Manning, C. D. (2008). *Introduction to information retrieval*. Syngress Publishing,.

Morgan, J. & Tatar, J. (1972). Calculation of the residual sum of squares for all possible regressions. *Technometrics*, *14*(2), 317–325.

Pandey, K. K. & Shukla, D. (2023). Ndpd: an improved initial centroid method of partitional clustering for big data mining. *Journal of Advances in Management Research*, *20*(1), 1–34.

Ponnusamy, M., Bedi, P., Suresh, T., Alagarsamy, A., Manikandan, R., & Yuvaraj, N. (2022). Design and analysis of text document clustering using salp swarm algorithm. *The Journal of Supercomputing*, *78*(14), 16197–16213.

Purohit, K., Vats, S., Saklani, R., Kukreja, V., Sharma, V., & Yadav, S. P. (2023). Improvement in k-means clustering for information retrieval. In *2023 4th International Conference on Electronics and Sustainable Communication Systems (ICESC)*, (pp. 1239–1245).

Qiao, Y., He, S., Yang, Y., Carlson, P., & Yang, T. (2024). Approximate cluster-based sparse document retrieval with segmented maximum term weights. *arXiv preprint arXiv:2404.08896*.

Rajagopal, P., Aghris, T., Fettah, F.-E., & Ravana, S. D. (2022). Clustering of relevant documents based on findability effort in information retrieval. *International Journal of Information Retrieval Research (IJIRR)*, *12*(1), 1–18.

Salton, G., Wong, A., & Yang, C.-S. (1975). A vector space model for automatic indexing. *Communications of the ACM*, *18*(11), 613–620.

Schutze, H., Manning, C. D., & Raghavan, P. (2008). *Introduction to information retrieval*. Cambridge University Press.

Sharma, I., Sharma, A., Chaturvedi, R., Rajpurohit, J., & Kumar, M. (2023). Skiff: Spherical k-means with iterative feature filtering for text document clustering. *Journal of Information Science*, 01655515231165230.

Song, W., Qiao, Y., Park, S. C., & Qian, X. (2015). A hybrid evolutionary computation approach with its application for optimizing text document clustering. *Expert Systems with Applications*, *42*(5), 2517–2524.

Sorensen, T. (1948). A method of establishing groups of equal amplitude in plant sociology based on similarity of species content and its application to analyses of the vegetation on danish commons. *Biologiske skrifter*, *5*, 1–34.

Sugar, C. A. & James, G. M. (2003). Finding the number of clusters in a dataset: An information-theoretic approach. *Journal of the American Statistical Association*, *98*(463), 750–763.

Tan, P., Steinbach, M., & Kumar, V. (2005). *Introduction to data mining. Austin*.

Tan, P.-N., Steinbach, M., & Kumar, V. (2016). *Introduction to data mining*. Pearson Education India.

Tibshirani, R., Walther, G., & Hastie, T. (2001). Estimating the number of clusters in a data set via the gap statistic. *Journal of the royal statistical society: series b (statistical methodology)*, *63*(2), 411–423.

V, V. K. & S, S. (2023). Developing a conceptual framework for short text categorization using hybrid cnn- lstm based caledonian crow optimization. *Expert Systems with Applications*, *212*, 118517.

Vattani, A. (2009). K-means requires exponentially many iterations even in the plane. In *Proceedings of the twenty-fifth annual symposium on Computational geometry*, (pp. 324–332).

Vinh, N., Epps, J., & Bailey, J. (2009). Information theoretic measures for clusterings comparison: Variants. *Properties, normalization and correction for chance*, *18*.

Von Luxburg, U. (2007). A tutorial on spectral clustering. *Statistics and computing*, *17*, 395–416.

Xu, R. & Wunsch, D. (2005). Survey of clustering algorithms. *IEEE Transactions on neural networks*, *16*(3), 645–678.

Yong, K. S. & Liew, J. S. Y. (2023). The more. *Journal of Intelligent Information Systems*, 1–23.

Yuan, M., Zobel, J., & Lin, P. (2022). Measurement of clustering effectiveness for document collections. *Information Retrieval Journal*, *25*(3), 239–268.