People's Democratic Republic of Algeria

Ministry of Higher Education and Scientific Research

AKLI MOHAND OULHADJ UNIVERSITY of BOUIRA

Faculty of Sciences and Applied Sciences

Department of Computer Science

Master's thesis

in Computer Science

specialty: *ISIL*

Theme

Multi view clustering of scientific documents

Supervised by

- M. BAL Kamel

Performed by

- Chekroune Amina
- Cherfaoui Asma

2025/2026

Acknowledgements

First, we would like to thank Allah for giving us strength, patience, and help to complete this work and our academic journey.

We would like to express our sincere thanks to our supervisor, Mr. Kamel Bal, for his guidance, valuable advice, and continuous support throughout this project.

We also thank all the teachers of the Information Systems and Software Engineering (ISIL) department for their teaching and support during our studies.

We are grateful to the members of the jury for accepting to evaluate this work and for their time and attention.

We would like to thank our families for their support, encouragement, and patience throughout our studies.

Finally, we thank ourselves for the effort and work we have done during this academic journey. We hope this work will be a positive step toward our future careers.

Dedication

I would like to dedicate this work to all those who have been part of my academic journey in one way or another.

This journey was not easy. It required effort, patience, and consistency, especially during moments when motivation was low and challenges felt overwhelming. Despite the difficulties, I continued step by step until reaching this stage.

I dedicate this achievement especially to my family, and in particular to my mother, who has been my greatest source of support throughout this path. Her care, encouragement, patience, and prayers were always a constant source of strength. She stood by me in every stage without hesitation and believed in me even during the most difficult moments.

I also dedicate this work to myself, for continuing despite fatigue and challenges, for choosing perseverance over giving up, and for reaching the end of this academic journey.

This success is the result of effort, patience, and the sincere support of those who stood by me throughout this journey.

And I conclude with this reminder:

وَمَا تَوْفِيقِي إِلَّا بِاللَه

Chekroune Amina

Dedication

وَقُلِ اعْمَلُوا فَسَيَرَى اللَّهُ عَمَلَكُمْ وَرَسُولُهُ وَالْمُؤْمِنُونَ وَسَتُرَدُّونَ إِلَىٰ عَالِمِ الْغَيْبِ وَالشَّهَادَةِ فَيُنَبِّئُكُمْ بِمَا كُنْتُمْ تَعْمَلُونَ

The journey was neither short nor easy, nor was the path paved with facilities and comfort. Yet, all praise is due to Allah, by whose grace good deeds are completed and goals are achieved.

With all my love and gratitude, I dedicate the fruit of my success and graduation:

To the one whose name I carry with pride, the light that illuminated my path and the lamp whose glow will forever shine in my heart. To the one who devoted years of effort so that I could climb the stairs of success. To the one whose sacrifices and devotion gave me strength and confidence.

My father, my support, and my hero.

To the angel of my life, the woman who taught me the true meaning of giving. To the one beneath whose feet lies Paradise, whose prayers accompanied me through every hardship. To the extraordinary woman who longed to witness this day and whose eyes reflected pride and hope in every step I took.

My beloved mother, my inspiration, and my safe haven.

To those who have always stood beside me, whose support has been an endless source of strength. To the warmth of my family, the comfort of my soul, and the joy of my heart:

My dear siblings, Esmahan, Youcef, Tasnim, and Younes.

To everyone who offered me a helping hand, sincere advice, encouragement, or a kind word throughout this journey.

To my beloved family.

And to myself... To the person who endured sleepless nights, overcame moments of doubt, and never gave up despite the challenges. To the one who turned difficulties into determination and dreams into reality. Today, I close a chapter of effort and perseverance and open a new chapter filled with hope, ambition, and pride. All praise and thanks belong to Allah for what I have achieved and for what is yet to come.

Finally, all gratitude is due to Allah, at the beginning and at the end.

وَآخِرُ دَعْوَاهُمْ أَنِ الْحَمْدُ لِلَّهِ رَبِّ الْعَالَمِينَ

Cherfaoui Asma

Abstract

الملخص

أدى النمو السريع في المنشورات العلمية إلى توفر كميات هائلة من الوثائق الرقمية، مما جعل تنظيمها وتحليلها مهمة معقدة. تعتمد طرق التجميع التقليدية عادةً على تمثيل واحد للوثائق، وهو ما لا يسمح بالتعبير الكامل عن تعقيد البيانات العلمية، وغالبًا ما يؤدي إلى نتائج غير مثالية.

في هذا السياق، برز التجميع متعدد الرؤى كنهج واعد يستغل عدة تمثيلات مكملة لنفس الوثيقة من أجل تحسين جودة التجميع.

في هذا العمل، قمنا بدراسة وتنفيذ إطار للتجميع متعدد الرؤى موجه للوثائق العلمية. يعتمد النهج المقترح على تمثيلات دلالية متقدمة مدمجة مع تقنيات تقليل الأبعاد، بالإضافة إلى استغلال العلاقات الهيكلية بين الوثائق. كما يتم تطبيق خوارزميات تجميع مناسبة لكل نوع من البيانات، مع اعتماد استراتيجية دمج لدمج المعلومات المكملة المستخلصة من مختلف الرؤى.

تم تقييم جودة العناقيد الناتجة باستخدام مقاييس كمية. وتُظهر النتائج التجريبية أن دمج عدة رؤى يساهم بشكل ملحوظ في تحسين تماسك العناقيد وزيادة درجة الفصل بينها مقارنة بالأساليب أحادية الرؤية.

الكلمات المفتاحية: التجميع متعدد الرؤى، تجميع الوثائق العلمية، التمثيل النصي، تحليل الاستشهادات، تقليل الأبعاد، التمثيلات (التضمينات) الدلالية، دمج العناقيد، خوارزمية كمانس للتجميع

Résumé

La croissance rapide des publications scientifiques a conduit à la disponibilité de grandes quantités de documents numériques, rendant leur organisation et leur analyse une tâche complexe. Les approches traditionnelles de clustering reposent généralement sur une seule représentation des documents, ce qui ne permet pas de capturer pleinement la complexité des données scientifiques et conduit souvent à des résultats sous-optimaux.

Dans ce contexte, le Multi-View Clustering est apparu comme une approche prometteuse exploitant plusieurs représentations complémentaires du même document afin d'améliorer la qualité du clustering.

Dans ce travail, nous étudions et mettons en œuvre un cadre de clustering multi-vue pour les documents scientifiques. L'approche proposée repose sur des représentations

sémantiques avancées combinées à des techniques de réduction de dimension, ainsi que sur l'exploitation des relations structurelles entre les documents. Des algorithmes de clustering appropriés sont appliqués à chaque type de données, et une stratégie de fusion est adoptée pour intégrer les informations complémentaires issues des différentes vues.

La qualité des clusters obtenus est évaluée à l'aide de mesures quantitatives. Les résultats expérimentaux montrent que la combinaison de plusieurs vues améliore significativement la cohésion et la séparation des clusters par rapport aux approches mono-vue.

Mots-clés : Multi-vue clustering, clustering de documents scientifiques, représentation textuelle, analyse des citations, réduction de dimension, embeddings sémantiques, fusion de clusters, clustering K-means

Abstract

The rapid growth of scientific publications has led to the availability of large volumes of digital documents, making their organization and analysis a challenging task. Traditional clustering approaches generally rely on a single representation of documents, which fails to fully capture the complexity of scientific data and often leads to suboptimal results.

In this context, Multi-View Clustering has emerged as a promising approach that exploits multiple complementary representations of the same document in order to improve clustering quality.

In this work, we study and implement a multi-view clustering framework for scientific documents. The proposed approach relies on advanced semantic representations combined with dimensionality reduction techniques, as well as the exploitation of structural relationships between documents. Appropriate clustering algorithms are applied to each type of data, and a fusion strategy is adopted to integrate the complementary information from different views.

The quality of the obtained clusters is evaluated using quantitative metrics. The experimental results demonstrate that combining multiple views significantly improves cluster coherence and separation compared to single-view approaches.

Keywords: Multi-View Clustering, Scientific Document Clustering, Text Representation, Citation Analysis, Dimensionality Reduction, Semantic Embeddings, Cluster Fusion, K-Means Clustering

Contents

List of Figures

List of Tables

Abbreviations list

BERT	Bidirectional Encoder Representations from Transformers
CHI	Calinski–Harabasz Index
DBSCAN	Density-Based Spatial Clustering of Applications with Noise
DBI	Davies–Bouldin Index
Doc2Vec	Document embedding model
GloVe	Global Vectors for Word Representation
LDA	Latent Dirichlet Allocation
LSI	Latent Semantic Indexing
node2vec	Node embedding model
PMC	PubMed Central
PubMed	Public Medical Database
SciBERT	Scientific Bidirectional Encoder Representations from Transformers
TF-IDF	Term Frequency–Inverse Document Frequency
UMAP	Uniform Manifold Approximation and Projection
Word2Vec	Word embedding model

General Introduction

In recent years, the scientific field has experienced rapid growth in the volume of scientific publications available in digital form. Large databases now contain a vast number of articles across various disciplines. This significant increase in data has made the processes of organization, analysis, and knowledge extraction increasingly challenging, especially due to the limitations and inefficiency of manual processing when dealing with large-scale information.

In this context, document clustering is considered one of the most important unsupervised learning techniques. It aims to divide documents into homogeneous groups such that documents within the same cluster are similar to each other and dissimilar to those in other clusters. This approach facilitates data exploration, improves organization, and enhances access to relevant information.

However, most traditional methods rely on a single representation of documents, typically based on textual content. This limitation prevents capturing the complex relationships that exist between scientific documents. In reality, such documents are not limited to textual content only, but also include multiple interconnected dimensions that reflect their underlying knowledge structure.

To address these challenges, more comprehensive approaches have emerged, among which Multi-View Clustering has gained significant attention. This approach exploits multiple representations of the same document, allowing a more integrated and richer view of the data, which ultimately improves clustering quality.

Problem Statement

Despite the significant progress in document processing techniques, clustering scientific documents still faces several challenges, particularly when dealing with complex and multidimensional data. Relying on a single representation often leads to the loss of important information, negatively affecting the accuracy and effectiveness of clustering results.

Furthermore, scientific documents are characterized by the diversity of their associated information sources, making their comprehensive representation a difficult task.

Considering this, the primary research question explored in this study is expressed as follows:

> How can the clustering of scientific documents be improved by exploiting multiple representations of the same document in an integrated manner?

Objectives of the Study

This work aims to:

- Study the concept of Multi-View Clustering and highlight its importance in analyzing scientific documents
- Analyze different document representation methods and identify their characteristics
- Identify the limitations of relying on a single representation in the clustering process
- Explore mechanisms for combining multiple representations within a unified framework
- Improve clustering quality by exploiting multiple sources of information
- Obtain more accurate clusters that better reflect the true meaning of scientific documents

Thesis Structure

This manuscript is organized as follows:

- **Chapter 1: Text Document Clustering**

  This chapter presents the fundamental concepts of document clustering, focusing on text-based representations and classical clustering techniques.

- **Chapter 2: Clustering of Scientific Documents**

  This chapter discusses the specific characteristics of scientific documents and the challenges associated with their clustering.

- **Chapter 3: Foundations of Multi-View Clustering**

  This chapter introduces the theoretical foundations of Multi-View Clustering and reviews existing approaches for combining multiple data representations.

- **Chapter 4: Multi-View Clustering for Scientific Documents (Case of Biomedical Articles)**

  This chapter presents the proposed multi-view clustering methodology, including data representation, fusion strategies, clustering algorithms, and evaluation metrics.

- **Chapter 5: Evaluation and Experiments**

  This chapter provides a comprehensive evaluation of the proposed multi-view clustering approach, presenting the experimental setup, datasets, evaluation metrics, and a detailed analysis of the obtained results.

Chapter 1

Text Document Clustering

1.1 Introduction

The increasing amount of digital textual data has made its organization and analysis an essential task in various domains such as information retrieval and data mining. Among the most important techniques used for this purpose is text document clustering, which aims to automatically group similar documents into meaningful clusters.

However, due to the unstructured nature of textual data, this process requires several fundamental steps, including data preparation, representation, similarity evaluation, and clustering. This chapter introduces the basic concepts and methods underlying text document clustering, providing a foundation for more advanced approaches.

1.2 What is Clustering?

Clustering is an unsupervised machine learning technique that aims to group a set of data points into clusters such that objects within the same cluster are more similar to each other than to those in other clusters. Unlike supervised learning, clustering does not rely on predefined labels; instead, it discovers hidden patterns and structures within the data.

In the context of text mining, clustering is used to organize large collections of documents into meaningful groups based on their content. This process facilitates tasks such as document organization, topic discovery, information retrieval, and recommendation systems.

According to [1], clustering plays a fundamental role in text mining because it enables the automatic structuring of unorganized textual data into coherent groups, thereby improving data analysis and knowledge extraction.

1.3 Clustering Process

The clustering process consists of a sequence of essential steps that transform raw textual data into meaningful groups of similar documents. As illustrated in Figure 1.1, this process ensures that unstructured text is progressively cleaned, represented, and analyzed in order to produce accurate and interpretable clusters.

First, text preprocessing aims to clean and normalize the raw textual data by removing noise and preparing it for analysis.

Second, feature extraction transforms the processed text into numerical vectors using representation models such as TF-IDF (see Section 1.4.2).

Third, similarity measurement evaluates the degree of similarity between documents using distance or similarity metrics.

Fourth, clustering algorithms group documents into clusters based on their similarity, using methods such as K-means or hierarchical clustering.

Fifth, cluster formation results in the creation of groups of documents that share similar characteristics or topics.

Finally, evaluation and interpretation assess the quality of the clustering results and extract meaningful insights from the formed clusters.

The clustering process can be summarized through the following steps:

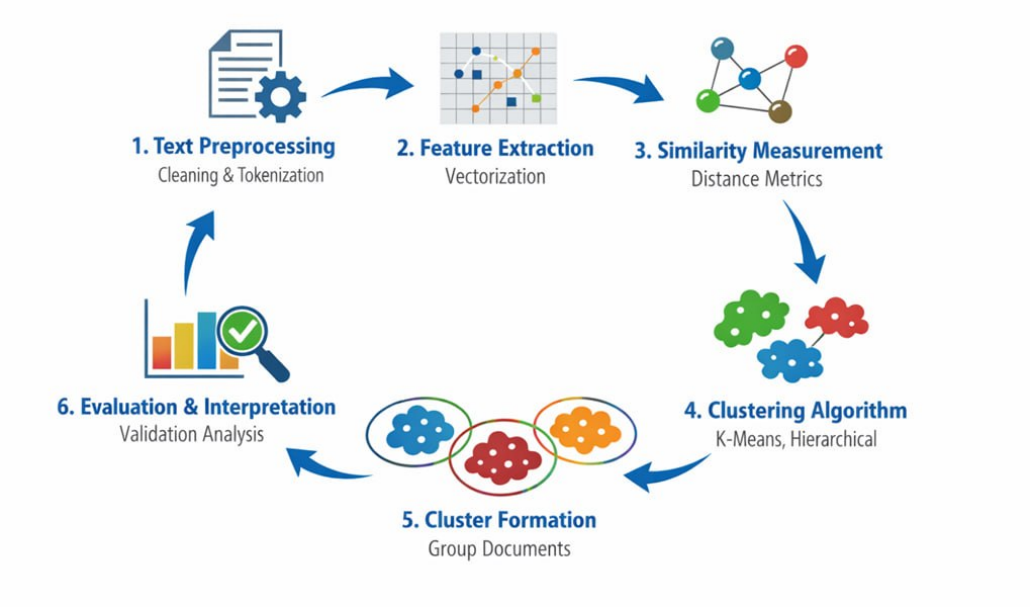

Figure 1.1: Text document clustering process

1.4 Document Representation Models

In text mining and document clustering, textual data must be transformed into a numerical representation that can be processed by machine learning algorithms. Since text documents are inherently unstructured, effective representation models are required to convert textual information into mathematical forms. These representations allow algorithms to measure similarities between documents and to perform tasks such as classification, clustering, and information retrieval.

According to [1], document representation is a fundamental step in text mining because most data mining algorithms operate on structured numerical data rather than raw text. Therefore, textual documents must be converted into feature vectors that capture the statistical properties of the terms contained in the documents.

Several models have been proposed in the literature for representing textual documents. Among the most widely used approaches are the Bag-of-Words model, TF-IDF weighting, the Vector Space Model, and Latent Semantic Indexing (LSI). These methods transform documents into vector representations where each dimension corresponds to a term in the vocabulary.

1.4.1 Bag-of-Words (BoW)

The Bag-of-Words model is one of the simplest and most commonly used approaches for representing textual documents. In this model, a document is represented as a collection of words without considering grammar, syntax, or word order. Instead, the representation focuses only on the frequency of words appearing in the document[1].

Formally, if we consider a vocabulary $V = \{t_1, t_2, \ldots, t_n\}$, each document can be represented as a vector:

$$d = (f_1, f_2, \ldots, f_n)$$

where f_i denotes the frequency of the term t_i in the document.

This representation leads to the creation of a term-document matrix, where rows correspond to documents and columns correspond to terms in the vocabulary. The entries of the matrix represent the number of occurrences of each term in each document. Despite its simplicity, the Bag-of-Words model has proven to be effective for many text mining tasks such as document classification and clustering.

However, the BoW representation has certain limitations because it ignores semantic relationships between words and treats each word independently.

1.4.2 TF-IDF Weighting

While the Bag-of-Words model uses raw term frequencies, not all words contribute equally to the meaning of a document. Some words appear frequently across many documents and therefore carry little discriminative power. To address this issue, the TF-IDF weighting scheme is commonly used[2] .

TF-IDF stands for Term Frequency – Inverse Document Frequency, and it measures the importance of a term in a document relative to a collection of documents.

Term Frequency (TF)

The Term Frequency (TF) measures how often a term appears in a document and can be defined as:

$$tf(t, d) = \frac{f(t, d)}{\sum_{t'} f(t', d)} \tag{1.1}$$

where:

- $f(t, d)$ is the number of occurrences of term t in document d,
- $\sum_{t'} f(t', d)$ is the total number of terms in document d.

Inverse Document Frequency (IDF)

The Inverse Document Frequency (IDF) measures how important a term is across the corpus. Several formulations exist:

$$idf(t) = \log\left(\frac{N}{df(t)}\right) \tag{1.2}$$

$$idf(t) = \log\left(\frac{N}{1 + df(t)}\right) \tag{1.3}$$

$$idf(t) = \log\left(1 + \frac{N}{df(t)}\right) \tag{1.4}$$

where:

- N is the total number of documents in the corpus,
- $df(t)$ is the number of documents containing the term t.

TF-IDF Combination

The TF-IDF weight is obtained by combining TF and IDF:

$$\text{TF-IDF}(t, d) = tf(t, d) \times idf(t) \tag{1.5}$$

Words that occur frequently in a document but rarely across the corpus receive higher TF-IDF weights. In contrast, words appearing in many documents obtain lower weights because they provide little information about document content. As noted by [2], TF-IDF weights are effective indicators of term importance and are widely used in information retrieval and text mining applications.

Based on TF-IDF weighting, each document can be represented as a vector in a multi-dimensional space, where each dimension corresponds to a term from the vocabulary and each value corresponds to its TF-IDF weight.

1.4.3 Latent Semantic Representation Models

Latent semantic representation models aim to capture hidden semantic structures in text by reducing dimensionality or modeling underlying topics. The main approaches include:

- **Latent Semantic Indexing (LSI)**

  Although the Vector Space Model is effective for representing text documents, it often suffers from problems such as high dimensionality and vocabulary mismatch. Different words may express similar meanings, while the same word may have multiple meanings.

  To address these limitations, Latent Semantic Indexing (LSI) was proposed as a dimensionality reduction technique that captures hidden semantic relationships between terms and documents[3].

  LSI relies on a matrix decomposition technique known as Singular Value Decomposition (SVD). Given a term-document matrix A, the decomposition can be expressed as:

$$A_{m\times n} = U_{m\times k}\Sigma_{k\times k}V^T_{k\times n} \tag{1.6}$$

  A simplified form of this decomposition is:

$$A = U\Sigma V^T \tag{1.7}$$

  where:

  - $A_{m\times n}$ is the term-document matrix.
  - $U_{m\times k}$ represents the term-concept matrix.
  - $\Sigma_{k\times k}$ is a diagonal matrix containing the top k singular values.
  - $V^T_{k\times n}$ represents the document-concept matrix.
  - k is the reduced number of latent dimensions ($k \ll n$).

By retaining only the largest singular values, LSI projects documents into a lower-dimensional semantic space where latent relationships between terms can be captured. This transformation helps improve document similarity estimation and reduces noise caused by irrelevant terms.

- **Latent Dirichlet Allocation (LDA)**

  Latent Dirichlet Allocation (LDA) is a probabilistic generative model used for discovering hidden thematic structures in large collections of text documents. Unlike traditional vector-based models, LDA represents each document as a mixture of latent topics, where each topic is characterized by a probability distribution over words[4].

  LDA assumes that documents are generated through a stochastic process in which:

  – Each document is composed of a mixture of multiple topics.

  – Each topic is defined as a distribution over a fixed vocabulary.

  – Each word in a document is generated by first selecting a topic, and then selecting a word from the corresponding topic distribution.

  The probability of a word given a document is defined as:

$$p(w \mid d) = \sum_{z=1}^{K} p(w \mid z)\, p(z \mid d) \tag{1.8}$$

  where $p(z|d)$ represents the topic distribution for a document and $p(w|z)$ represents the word distribution within a topic.

  By modeling documents in this way, LDA is able to uncover latent semantic structures without requiring labeled data. This makes it particularly useful for tasks such as topic modeling, document clustering, and information retrieval.

1.4.4 Neural Embedding Models

Recent advances in natural language processing have introduced neural embedding models, which provide dense and semantically meaningful representations of textual data.

Unlike traditional models such as Bag-of-Words or TF-IDF, which produce sparse representations, neural embeddings map words and documents into continuous vector spaces where semantic relationships are preserved.

These models are particularly effective in capturing contextual and semantic similarities between words and documents, making them highly suitable for modern text mining and clustering applications.

- **Word Embedding Models: Word2Vec and GloVe**

  Word embedding models such as Word2Vec and GloVe aim to represent words as dense, low-dimensional vectors in a continuous vector space. These representations are learned based on the distributional hypothesis, which states that words appearing in similar contexts tend to have similar meanings.

  Word2Vec, introduced by [5], uses neural networks to learn word representations by predicting surrounding words (Skip-gram) or predicting a word from its context (CBOW). Similarly, GloVe [6] constructs word embeddings by leveraging global word co-occurrence statistics.

  These models are capable of capturing semantic relationships and linguistic regularities, such as analogies (e.g., king – queen ≈ man - woman). However, they generate static embeddings, meaning that each word has a single representation regardless of context.

- **Document Embedding Model: Doc2Vec**

  Doc2Vec, proposed by [7], extends the Word2Vec framework to learn vector representations for entire documents. It introduces an additional vector representing the document, which is trained alongside word vectors to capture the global semantics of the text.

  This approach produces dense and informative document-level representations that are useful for tasks such as document clustering and classification. However, the effectiveness of Doc2Vec depends on the availability of large and diverse training corpora.

- **Contextual Embedding Models: Transformer-based Approaches**

More recently, transformer-based models such as BERT [8] and GPT [9] have significantly advanced text representation techniques. These models generate contextual embeddings, where the representation of a word dynamically changes depending on its surrounding context.

Unlike static embeddings, contextual models capture deeper semantic and syntactic relationships, enabling superior performance across a wide range of NLP tasks. They are particularly effective in handling polysemy and contextual ambiguity.

However, these models require substantial computational resources and are often more complex to interpret compared to traditional approaches.

1.4.5 Comparison of Document Representation Models

The table below presents a comparison of document representation models according to different criteria.

Model	**Typical Use Case**	**Complexity**	**Handles Word Order**	**Captures Semantics**
Bag-of-Words (BoW)	Basic information retrieval, text classification	Low	No	No
TF-IDF	Search engines, keyword extraction	Low	No	Limited
Latent Semantic Indexing (LSI)	Topic clustering, information retrieval	Medium	No	Yes (latent semantics)
Latent Dirichlet Allocation (LDA)	Topic modeling, document analysis	Medium	No	Yes (topic-based)
Word2Vec / GloVe	Semantic similarity, word representation	Medium	No	Yes
Doc2Vec	Document clustering, document representation	Medium	No	Yes
BERT / GPT	Advanced NLP tasks, contextual understanding	High	Yes	Yes (deep contextual semantics)

Table 1.1: Comparison of document representation models

1.5 Text Preprocessing Techniques

Text preprocessing is a crucial step in text mining and natural language processing. Raw textual data often contains noise, irrelevant information, and inconsistencies that may negatively affect the performance of text analysis algorithms. Therefore, preprocessing techniques are applied to clean, normalize, and transform text into a structured representation that can be efficiently processed by machine learning and data mining algorithms.

According to [1], preprocessing helps reduce data dimensionality and improve the quality of textual representations used in tasks such as document clustering and classification.

Several preprocessing techniques are commonly applied in text mining, including tokenization, stop-word removal, stemming, and lemmatization.

1.5.1 Tokenization

Tokenization is the process of breaking a text document into smaller units called tokens. These tokens typically correspond to words, but they may also represent sentences, characters, or sub-words depending on the analysis task. Tokenization is considered the first step in most text preprocessing pipelines because it transforms raw text into manageable units that can be analyzed computationally.

For example, the sentence "Text mining techniques are widely used in data analysis" can be tokenized into the following list of tokens: [text, mining, techniques, are, widely, used, in, data, analysis]. Once the text is tokenized, each token can be analyzed separately and used to construct document representation models such as the Bag-of-Words model or TF-IDF vectors.

Tokenization facilitates subsequent processing steps by enabling the identification and manipulation of individual words within a document. As noted by [10], tokenization plays a fundamental role in natural language processing because most computational models operate on sequences of tokens rather than raw text strings.

1.5.2 Stop-Word Removal

Stop-word removal is another important preprocessing technique in text mining. Stop words are common words that appear frequently in a language but carry little semantic

meaning for text analysis tasks. Examples of stop words in English include words such as *the, is, and, of,* and *in*. Since these words appear in many documents, they provide limited information for distinguishing between documents.

Removing stop words helps reduce the size of the vocabulary and improves the efficiency of text mining algorithms. It also enhances the quality of document representation by focusing on the most informative words in the text. For instance, the sentence "The process of text mining is very important in data analysis" becomes "process text mining important data analysis" after removing stop words.

According to [10], eliminating stop words can significantly reduce noise in textual data and improve the performance of various natural language processing tasks.

1.5.3 Stemming

Stemming is a text normalization technique that reduces words to their root or base form by removing suffixes or prefixes. The goal of stemming is to group together different forms of a word so that they can be treated as the same term during analysis. For example, the words *computing*, *computed*, and *computation* may all be reduced to the stem *comput*. This process helps reduce the number of unique terms in the vocabulary and improves the efficiency of text mining algorithms.

Stemming algorithms are commonly used in information retrieval and text mining systems in order to simplify word representations while preserving their essential meaning. By reducing morphological variants of words to a common base form, stemming improves the matching process between documents and queries and reduces the dimensionality of textual data [2].

1.5.4 Lemmatization

Lemmatization is another technique used to normalize textual data. Unlike stemming, which relies on simple heuristic rules, lemmatization uses vocabulary and morphological analysis to convert words into their canonical or dictionary form, known as the lemma. For example, the word *running* is reduced to *run*, while *better* is transformed into *good*.

Lemmatization generally produces more accurate and linguistically meaningful results than stemming because it considers the grammatical structure of words. However, it

requires more computational resources and access to lexical databases such as dictionaries or linguistic corpora.

According to [10], lemmatization is particularly useful in applications that require a deeper understanding of language structure, such as semantic analysis and advanced natural language processing tasks.

1.6 Similarity Measures

In text mining and document clustering, measuring the similarity between documents is a fundamental task. After transforming textual documents into numerical representations, such as vectors using techniques like Bag-of-Words or TF–IDF, it becomes necessary to evaluate how similar or different two documents are. Similarity measures provide mathematical methods to quantify the degree of resemblance between document vectors. These measures are widely used in information retrieval, document clustering, and recommendation systems to determine relationships between textual data [1].

Several similarity and distance metrics have been proposed in the literature to compare documents represented as vectors. Among the most commonly used measures in text mining are Cosine Similarity, Euclidean Distance, and Jaccard Similarity. Each of these measures evaluates similarity from a different perspective and is suitable for particular types of data representations.

1.6.1 Cosine Similarity

Cosine similarity is one of the most widely used similarity measures in text mining and information retrieval. It evaluates the similarity between two document vectors by measuring the cosine of the angle between them in a multidimensional vector space. The idea behind this measure is that two documents are considered similar if the angle between their vector representations is small. Cosine similarity values range from 0 to 1, where a value close to 1 indicates a high degree of similarity between documents, while a value close to 0 indicates little or no similarity [2].

Mathematically, the cosine similarity between two vectors A and B is defined as follows:

$$\text{cosine}(A, B) = \frac{A \cdot B}{\|A\| \, \|B\|} \tag{1.9}$$

where $A \cdot B$ represents the dot product of the two vectors and $\|A\|$ and $\|B\|$ denote their magnitudes. This measure is particularly effective for text documents because it focuses on the orientation of the vectors rather than their magnitude, which reduces the impact of document length on similarity calculations [1].

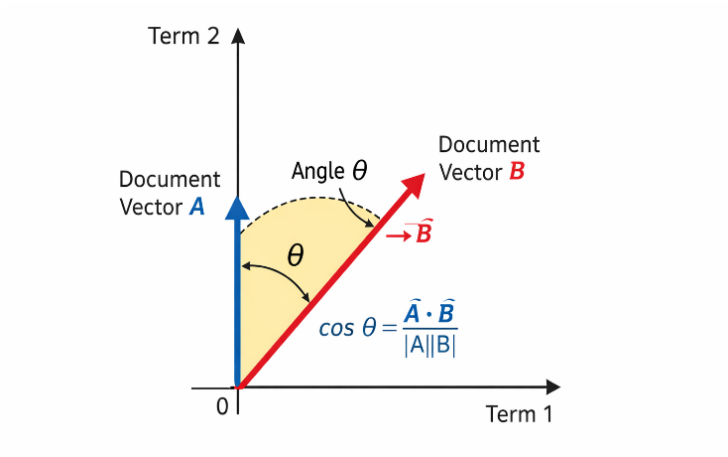

Figure 1.2: Cosine similarity between two document vectors

1.6.2 Euclidean Distance

Euclidean distance is a classical distance metric commonly used in machine learning and clustering algorithms. It measures the straight-line distance between two points in a multidimensional space. In the context of document representation, each document is represented as a vector, and the Euclidean distance calculates how far apart these vectors are from each other.

The Euclidean distance between two vectors A and B is defined as:

$$d(A, B) = \sqrt{\sum_{i=1}^{n} (A_i - B_i)^2} \tag{1.10}$$

A smaller distance indicates that the documents are more similar, whereas a larger distance suggests greater dissimilarity. Although Euclidean distance is widely used in clustering algorithms such as k-means, it may be sensitive to variations in document length when applied to text data [11].

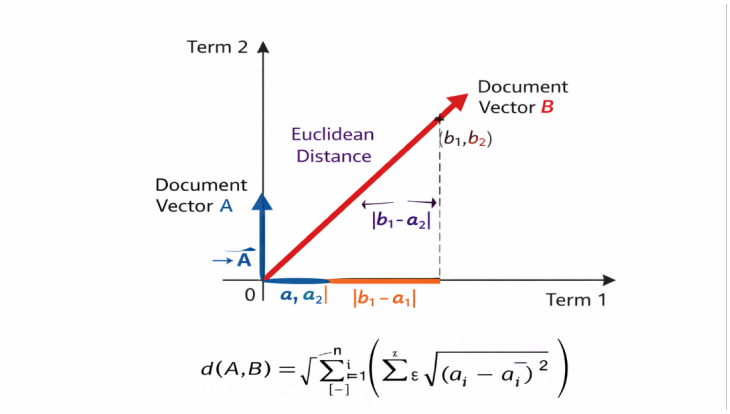

Figure 1.3: Euclidean distance between two document vectors

1.6.3 Jaccard Similarity

Jaccard similarity is another important metric used to measure similarity between two sets. Unlike cosine similarity and Euclidean distance, which operate on numerical vectors, the Jaccard coefficient is typically applied to sets of terms. It measures the ratio between the intersection and the union of two sets.

The Jaccard similarity between two sets A and B is defined as:

$$J(A, B) = \frac{|A \cap B|}{|A \cup B|} \tag{1.11}$$

where $|A \cap B|$ represents the number of common elements between the two sets, and $|A \cup B|$ represents the total number of unique elements in both sets. The value of the Jaccard coefficient ranges from 0 to 1, where 1 indicates that the two sets are identical and 0 indicates that they share no common elements. This measure is particularly useful for comparing documents based on the presence or absence of terms [1].

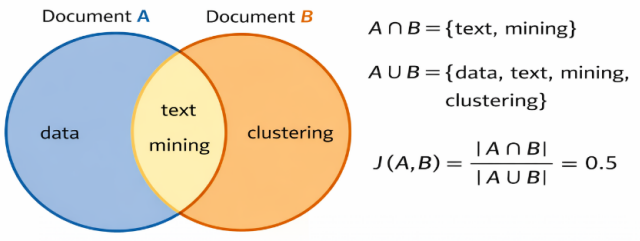

Figure 1.4: Jaccard similarity between two documents based on their term sets

1.7 Classical Clustering Algorithms

In the context of document clustering, textual data are typically represented as vectors using models such as Bag-of-Words or TF-IDF. Once documents are represented in a vector space, clustering algorithms can be applied to identify groups of semantically related documents. As noted by Han, Kamber, and Pei (2011), clustering techniques aim to partition data objects into meaningful groups based on similarity or distance measures [11].

Several clustering algorithms have been proposed in the literature. Among the most widely used classical methods are K-means clustering, hierarchical clustering, DBSCAN, and spectral clustering. These algorithms differ in their assumptions, computational complexity, and the types of clusters they can discover.

1.7.1 K-means Clustering

K-means is one of the most popular and widely used clustering algorithms due to its simplicity and efficiency. The algorithm partitions a set of data points into k clusters, where each cluster is represented by a centroid corresponding to the mean of the data points assigned to that cluster.

The algorithm operates iteratively. Initially, k centroids are randomly selected. Each data point is then assigned to the nearest centroid based on a distance measure, commonly the Euclidean distance. After the assignment step, the centroids are recalculated as the mean of the points within each cluster. This process continues until the cluster assignments no longer change or until a predefined convergence criterion is reached.

The objective of the K-means algorithm is to minimize the within-cluster sum of squares, which can be expressed as:

$$J = \sum_{i=1}^{k} \sum_{x \in C_i} \|x - \mu_i\|^2 \tag{1.12}$$

where C_i represents the set of data points belonging to cluster i, and μ_i is the centroid of that cluster.

K-means is computationally efficient and scalable to large datasets, making it widely used in document clustering tasks. However, the algorithm has certain limitations, including sensitivity to the initial selection of centroids and the requirement to specify the number of clusters in advance [11].

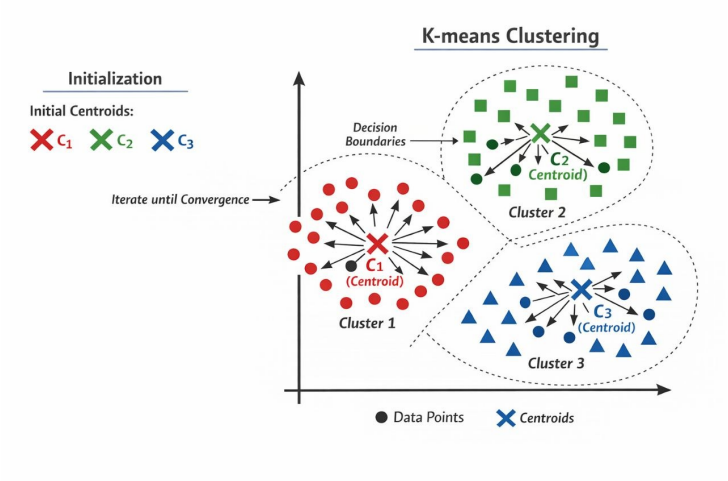

Figure 1.5: K-means clustering process with data points grouped around centroids.

Based on this principle, the k-means algorithm can be formally described as follows [11]:

Algorithm 1: K-means Clustering Algorithm

Input: Dataset $D = \{t_1, t_2, \ldots, t_n\}$, number of clusters K

Output: A set of K clusters

Initialize K cluster centroids $m_1, m_2, \ldots, m_K$ randomly;

repeat

  Assign each data point t_i to the cluster with the nearest centroid;

  Update each centroid m_j by computing the mean of all points assigned to cluster j;

until *convergence criterion is satisfied*;

1.7.2 Spectral Clustering

Spectral clustering is a more advanced clustering technique that relies on concepts from graph theory and linear algebra. Instead of directly applying clustering algorithms to the original data space, spectral clustering first constructs a similarity graph that represents relationships between data points.

In this graph, each node corresponds to a document, and edges represent the similarity between documents. The algorithm then computes the eigenvectors of a graph Laplacian matrix derived from the similarity matrix. These eigenvectors are used to project the data into a lower-dimensional space where clustering can be performed more effectively.

Spectral clustering is particularly useful when clusters have complex structures that cannot be easily separated using traditional methods such as K-means. It has been successfully applied in various domains, including image segmentation and document clustering [12].

The following figure illustrates the main steps of the spectral clustering process:

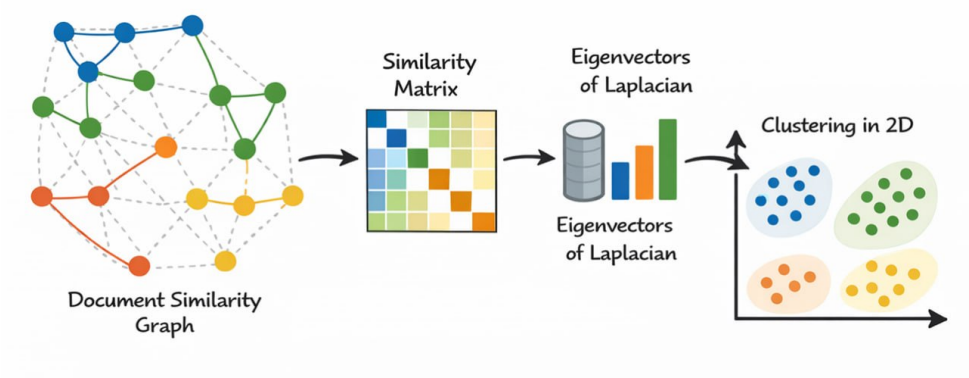

Figure 1.6: Spectral clustering process

Based on this process, the spectral clustering algorithm can be summarized as follows [12]:

Algorithm 2: Spectral Clustering Algorithm

Input: Dataset $S = \{s_1, s_2, \ldots, s_n\}$, number of clusters k

Output: A partition of the dataset into k clusters

Construct the similarity (affinity) matrix A:

$$A_{ij} = \exp\left(-\frac{\|s_i - s_j\|^2}{2\sigma^2}\right), \quad i \neq j \tag{1.13}$$

Compute the diagonal degree matrix D:

$$D_{ii} = \sum_j A_{ij}$$

Compute the normalized Laplacian matrix:

$$L = D^{-1/2} A D^{-1/2}$$

Compute the first k eigenvectors of L and form matrix $X \in \mathbb{R}^{n \times k}$;

Normalize each row of X to obtain matrix Y;

Apply K-means on the rows of Y;

Assign each point to the corresponding cluster;

1.7.3 Hierarchical Clustering

Hierarchical clustering is another widely used clustering technique that builds a hierarchy of clusters rather than producing a single partition of the data. The result of hierarchical clustering is typically represented as a tree-like structure known as a dendrogram, which illustrates how clusters are merged or divided at different levels of similarity.

There are two main approaches to hierarchical clustering:

- Agglomerative (bottom-up) clustering, where each document initially forms its own cluster, and clusters are iteratively merged based on similarity.
- Divisive (top-down) clustering, where all documents initially belong to a single cluster, which is then recursively divided into smaller clusters.

The merging or splitting decisions are typically based on distance measures and linkage criteria such as single linkage, complete linkage, or average linkage. Hierarchical clustering is particularly useful for exploratory data analysis because it provides a visual representation of the clustering structure. However, it can be computationally expensive when applied to large datasets [1].

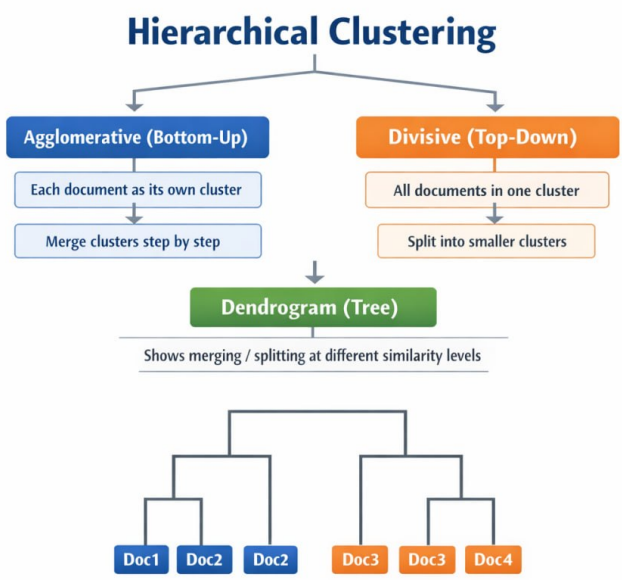

Figure 1.7: Hierarchical clustering process with dendrogram representation

1.7.4 DBSCAN

DBSCAN (Density-Based Spatial Clustering of Applications with Noise) is a density-based clustering algorithm designed to discover clusters of arbitrary shape in datasets containing noise. Unlike partitioning algorithms such as K-means, DBSCAN does not require the number of clusters to be specified in advance[13].

The algorithm groups together points that are closely packed in regions of high density while marking points located in low-density regions as noise or outliers. DBSCAN relies on two main parameters:

- ε (epsilon), which defines the radius of the neighborhood around a data point.
- MinPts, which specifies the minimum number of points required to form a dense region.

Points that have at least MinPts neighbors within the ε radius are considered core points and form the basis of clusters. Points that fall within the neighborhood of core points are assigned to the corresponding cluster, while isolated points are classified as noise.

One of the main advantages of DBSCAN is its ability to detect clusters with irregular shapes and to handle noise effectively. However, the performance of the algorithm strongly depends on the appropriate selection of its parameters [11].

The following figure illustrates how DBSCAN identifies clusters based on density, highlighting core points, border points, and noise:

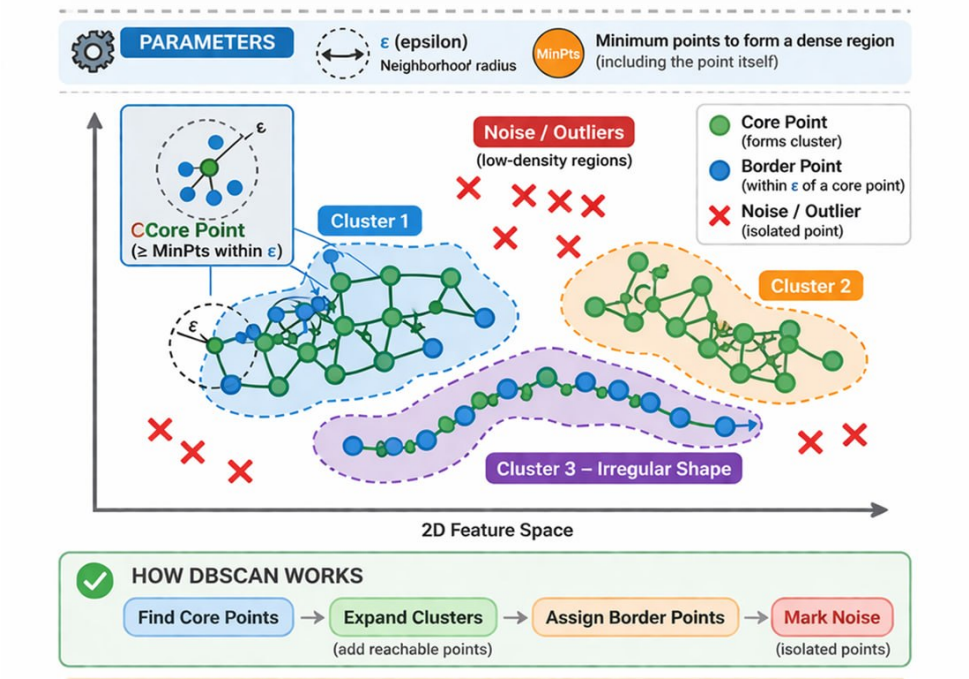

Figure 1.8: DBSCAN clustering process based on density

Based on this principle, the DBSCAN algorithm can be formally described as follows [13]:

Algorithm 3: DBSCAN Clustering Algorithm

Input: Dataset D, neighborhood radius ε, minimum number of points $MinPts$

Output: Set of clusters

```
Mark all points in D as UNVISITED;
foreach point p ∈ D do
    if p is UNVISITED then
        Mark p as VISITED;
        Retrieve neighbors of p within radius ε;
        if number of neighbors < MinPts then
            Mark p as NOISE;
        else
            Create a new cluster C;
            Add p to C;
            foreach neighbor q of p do
                if q is UNVISITED then
                    Mark q as VISITED;
                    Retrieve neighbors of q;
                    if number of neighbors ≥ MinPts then
                        Add these neighbors to the neighbor list;
                if q is not assigned to any cluster then
                    Add q to cluster C;
```

1.8 Challenges of Text Clustering

Text clustering is a complex task due to the intrinsic characteristics of textual data. Unlike structured numerical data, text data is typically high-dimensional, sparse, and semantically ambiguous. These challenges significantly affect the performance and accuracy of clustering algorithms.

1.8.1 High Dimensionality

One of the main challenges in text clustering is the high dimensionality of textual data. Each unique word in the corpus is treated as a feature, which leads to a very large feature space.

As highlighted in recent studies, text data can contain thousands or even millions of features depending on the vocabulary size. This high dimensionality increases computational complexity and reduces the efficiency of clustering algorithms.

Moreover, high-dimensional data often leads to the curse of dimensionality, where the distance between data points becomes less meaningful, making it difficult to accurately measure similarity.

1.8.2 Vocabulary Mismatch

Another major challenge is vocabulary mismatch, which refers to the variability in word usage across documents.

Different words may express the same meaning (synonymy), while the same word may have multiple meanings (polysemy). For example, two documents discussing the same topic may use different terms, leading to low similarity despite their semantic closeness.

Traditional models such as Bag-of-Words fail to capture these semantic relationships, as they treat words independently without considering context. This limitation negatively affects clustering quality and leads to inaccurate grouping of documents.

1.8.3 Sparsity Problem

Text data is inherently sparse, meaning that most documents contain only a small subset of the total vocabulary.

As a result, the term-document matrix contains many zero values, making the data representation sparse. This sparsity reduces the effectiveness of similarity measures and makes it difficult for clustering algorithms to detect meaningful patterns.

Studies have shown that sparse text representations can lead to poor generalization and reduced clustering performance, especially when using traditional methods such as K-means.

These challenges highlight the limitations of traditional text clustering approaches

and justify the need for more advanced techniques such as semantic representations and multi-view clustering [14].

1.9 Conclusion

In this chapter, the fundamental concepts of text document clustering have been introduced, covering the essential steps involved in processing and organizing textual data, from representation and preprocessing to similarity measurement and clustering techniques.

These elements provide a solid foundation for understanding how textual data can be structured and analyzed effectively. They also serve as a basis for exploring more advanced approaches in the following chapters.

Chapter 2

Clustering of scientific documents

2.1 Introduction

The rapid growth of scientific publications has resulted in large-scale and highly interconnected document collections, making their organization a critical challenge in fields such as information retrieval, knowledge discovery, and research trend analysis.

Traditional manual classification approaches are no longer sufficient to manage the continuous expansion of digital repositories. As a result, automatic methods for structuring and exploring scientific literature have become increasingly necessary.

Among these methods, clustering has emerged as an effective unsupervised technique for grouping documents based on their similarity. In the context of scientific literature, it enables the discovery of underlying thematic structures without relying on predefined categories.

However, clustering scientific documents remains challenging due to their specific characteristics, including structured formats, specialized vocabulary, and citation relationships. These aspects require advanced methods capable of capturing both textual and structural information.

This chapter provides an overview of clustering techniques for scientific documents. It introduces the main concepts, examines document characteristics and similarity measures, and discusses key challenges and limitations.

2.2 Scientific Documents: Specificities and Characteristics

A scientific document is a formal and structured piece of writing designed to communicate research findings, methodologies, and academic contributions. These documents typically follow standardized formats, including sections such as abstract, introduction, methodology, results, and references, which facilitate the dissemination and evaluation of scientific knowledge [15].

Scientific documents are characterized not only by their structured organization but also by the use of domain-specific terminology and precise language. Additionally, they are often interconnected through citation networks, where each document references prior work, forming complex relationships within the scientific community.

2.2.1 Structural Characteristics of Scientific Documents

Scientific documents represent the primary medium for communicating research findings within the scientific community. They follow a structured format that ensures clarity, reproducibility, and accessibility of scientific knowledge.

According to [15], scientific documents generally follow a structured format that includes several essential components:

- **Title**

  The title provides a concise description of the research topic and reflects the main focus of the study. A well-formulated title should be precise and informative, allowing readers to quickly understand the subject of the document.

- **Authors**

  Authors are the researchers who contributed to the work presented in the paper. Their names usually appear below the title, together with their institutional affiliations, providing both credit and responsibility for the reported research.

- **Abstract**

  The abstract provides a brief summary of the research, including its objectives, methodology, and main findings. It allows readers to quickly understand the purpose

and contributions of the study.

- **Keywords**

  Keywords represent the main topics addressed in the document and help improve the indexing and retrieval of scientific publications in digital libraries.

- **Introduction**

  The introduction presents the research context and defines the problem addressed in the study. It also outlines the objectives and motivations of the research.

- **Methodology**

  The methodology section describes the methods, techniques, or datasets used to conduct the research, enabling other researchers to understand and reproduce the study.

- **Results and Discussion**

  This section presents the findings obtained from the research and interprets their significance by comparing them with previous studies.

- **References**

  The references section lists the scientific sources cited in the document, allowing readers to consult the original works.

2.2.2 Specificities of Scientific Documents

Scientific documents have several characteristics that distinguish them from other types of textual data. These specificities influence how scientific literature is analyzed, organized, and processed in tasks such as information retrieval, text mining, and document clustering.

- **Structured Nature of Scientific Documents**

  Scientific documents follow a well-defined and standardized structure, including sections such as abstract, introduction, methodology, and results. Each section serves a specific purpose and contains different types of information. This structured organization distinguishes scientific documents from other types of textual data and facilitates their analysis and processing [15].

- **Specialized Scientific Vocabulary**

  Scientific documents contain specialized terminology specific to particular research domains. Researchers use precise technical terms to describe methods, theories, and results. This specialized vocabulary helps communicate complex ideas clearly within the scientific community [2].

- **Formal and Objective Writing Style**

  Scientific documents are written in a formal and objective style, focusing on clarity, precision, and neutrality. Authors avoid ambiguity and subjective expressions, ensuring that the information is communicated accurately and can be understood by the scientific community [15].

- **Large-Scale and Network-Based Nature**

  Scientific publications exist in very large collections and are often interconnected through various relationships. These documents form large networks of knowledge, where papers are linked through citations, collaborations, and shared research topics. Due to this large-scale and interconnected nature, scientific literature is often analyzed using network-based approaches [16].

- **Citation-Based Semantic Connectivity**

  Scientific documents are connected through citation relationships. When a paper cites another work, it indicates that the cited research is relevant to the current study. These citation links create semantic connections between documents and help reveal the structure of scientific knowledge and research communities [16].

2.3 Text-Based Approaches for Scientific Document Clustering

Scientific documents present unique linguistic characteristics that distinguish them from general-purpose text collections. They contain domain-specific terminology, dense technical jargon, complex abbreviations, and highly structured content such as abstracts, methodology sections, and reference lists. These properties make scientific document clustering particularly challenging, as effective representation models must go beyond

surface-level lexical matching and capture deep semantic relationships within specialized vocabularies.

Building upon the representation models introduced in Chapter 1, this section focuses on their application in the specific context of scientific document clustering. The approaches are presented according to their level of semantic expressiveness.

2.3.1 Vector Space and Keyword-Based Approaches

Early approaches for clustering scientific documents rely on the vector space model (VSM), where each document is represented as a weighted term vector. In this representation, term weights are commonly computed using TF-IDF to emphasize discriminative scientific terms while reducing the impact of frequent but less informative words. Document similarity is typically measured using cosine similarity.

The effectiveness of this model was first demonstrated by Salton et al. [17], who showed that representing documents as term vectors enables efficient indexing and clustering. More recent work, such as Kim and Gil [18], confirms that TF-IDF remains effective for scientific document classification, where it is used to extract representative keywords from abstracts and cluster papers using algorithms such as K-means.

However, despite its simplicity and scalability, the vector space model suffers from limitations in scientific corpora, particularly due to synonymy and polysemy, which can reduce clustering accuracy in multidisciplinary contexts.

2.3.2 Dimensionality Reduction and Latent Semantic Models

To address vocabulary mismatch and high dimensionality in vector space representations, dimensionality reduction techniques are used to capture latent semantic structures in scientific documents. These approaches provide more compact and meaningful representations, forming the basis of models such as LSI and LDA.

Latent Semantic Indexing (LSI). Proposed by Deerwester et al. [3], LSI applies Singular Value Decomposition (SVD) to the term-document matrix to capture latent semantic relationships between terms and documents. By projecting documents into a lower-dimensional space, LSI reduces noise caused by synonymy and polysemy, which are common in scientific writing. As a result, documents that share similar underlying

concepts, even if expressed using different vocabularies, are mapped closer in the latent space.

In the context of document clustering, LSI has been shown to significantly improve similarity measurement and clustering performance. For instance, Han et al. [19] demonstrated that applying LSI increases cosine similarity values between documents from approximately 0.02–0.03 to 0.19–0.35, leading to more effective grouping of related documents.

This makes LSI particularly useful for clustering scientific publications based on conceptual similarity rather than exact term overlap.

Latent Dirichlet Allocation (LDA). Introduced by Blei et al. [4], LDA is a probabilistic generative model in which each document is represented as a mixture of latent topics, while each topic is defined by a distribution over words. This representation is well-suited for scientific documents that often cover multiple research themes.

In scientific document clustering, LDA enables the discovery of hidden thematic structures and facilitates grouping publications according to their underlying topics. This has been demonstrated in studies such as Yau et al. [20], where topic modeling was used to cluster scientific documents using metrics such as precision, recall, and F-score.

Overall, LDA provides a more semantic and flexible representation than keyword-based approaches, improving clustering performance. Together with LSI, it produces compact and meaningful features that enhance clustering results.

2.3.3 Word and Document Embedding-Based Approaches

Neural embedding models provide dense vector representations that are well suited to the complexity of scientific language, where meaning is often expressed through specialized terminology and multi-word concepts.

Word2Vec [5] learns word representations based on local context using neural networks. It captures semantic relationships between words by placing those that occur in similar contexts closer in the embedding space. In the context of document clustering, its effectiveness has been demonstrated by Walkowiak et al. [21], where word embedding models were successfully used to represent documents and improve clustering quality using evaluation metrics such as AMI.

GloVe [6] leverages global word co-occurrence statistics to learn vector representations

that capture both semantic and syntactic relationships. By incorporating global corpus information, it is more robust to uneven term distributions commonly found in scientific texts. Recent studies such as Canon et al. [22] have shown that combining GloVe with Word2Vec embeddings improves clustering performance in academic text analysis by providing richer semantic representations.

Doc2Vec [7] extends word embedding approaches to the document level by learning fixed-length vector representations for entire documents. This enables capturing the global thematic structure of scientific papers, which is essential for clustering tasks. Its effectiveness has been demonstrated in studies such as Chen and Sokolova [23], where Word2Vec and Doc2Vec were applied to the analysis of scientific and medical texts, highlighting their ability to capture semantic relationships at both word and document levels.

These embedding representations are commonly used as input features for clustering algorithms such as K-means or hierarchical clustering, enabling more semantically meaningful grouping of scientific documents.

2.3.4 Transformer-Based Contextual Embedding Approaches

Recent advances in scientific document clustering are largely driven by transformer-based models, which generate contextual embeddings capable of capturing complex semantic relationships.

BERT [8] produces contextual representations by modeling bidirectional dependencies between words. While highly effective for capturing semantic meaning, its direct application to clustering tasks requires adaptation to generate meaningful document representations. Recent studies such as Hosokawa et al. [24] have demonstrated that BERT-based models can be successfully applied to cluster scientific documents by leveraging semantic similarity between abstracts, achieving performance comparable to or exceeding that of human experts.

SciBERT [25] extends BERT by being trained specifically on scientific publications. This domain adaptation enables it to better capture scientific terminology and writing conventions. In the study by Hosokawa et al. [24], SciBERT achieved superior clustering performance compared to general-purpose BERT models, highlighting the importance of domain-specific pretraining for scientific document clustering.

Despite their superior performance, transformer-based models require substantial com-

putational resources, which may limit their scalability for very large scientific datasets.

2.3.5 Comparative Summary of Text-Based Approaches for Scientific Documents

The table below presents a comparison of the main text-based approaches.

Approach	**Representation**	**Semantics**	**Strengths**	**Limitations**
VSM (TF-IDF)	Sparse vectors	Static	Simple, scalable	Synonymy, polysemy
LSI	Latent space (SVD)	Latent	Reduces synonymy/polysemy	Non-interpretable dimensions
LDA	Topic distributions	Latent	Interpretable topics	Sensitive to hyperparameters
Word2Vec	Word embeddings	Static semantic	Semantic similarity	No document representation
GloVe	Global embeddings	Static semantic	Local + global statistics	Context-independent
Doc2Vec	Document embeddings	Static semantic	Captures global meaning	Requires inference for new documents
BERT	Contextual embeddings	Deep contextual	Rich semantic representations	Computationally expensive
SciBERT	Scientific embeddings	Deep contextual	Domain-specific understanding	Computationally expensive

Table 2.1: Comparative Summary of Text-Based Approaches for Scientific Document Clustering

2.4 Citation-Based Approaches for Scientific Document Clustering

2.4.1 Citation Analysis Techniques

Citation analysis is a fundamental approach in scientometrics that studies how scientific documents reference each other. It provides insights into the intellectual structure of research fields, knowledge propagation, and relationships between documents [16]. By analyzing citation links, it becomes possible to identify thematic similarities and research communities.

Citation relationships are commonly represented as graphs, where documents are modeled as nodes and citation links as edges. These graph-based representations enable both quantitative analysis and visualization of scientific knowledge.

- **Citation Graphs**

  A citation graph is a directed graph $G = (V, E)$, where:

  - V represents the set of documents (nodes),
  - E represents the set of directed edges, where an edge $e_{ij} \in E$ exists if document d_i cites document d_j.

  The adjacency matrix A is defined as:

$$A_{ij} = \begin{cases} 1 & \text{if } d_i \text{ cites } d_j \\ 0 & \text{otherwise} \end{cases} \tag{2.1}$$

  This representation allows modeling citation networks and analyzing relationships between scientific documents [16].

- **Direct Citation**

  Direct citation evaluates the connection between two scientific documents based on the existence of a direct citation link. Two documents d_i and d_j are considered related if one document cites the other [16].

  The similarity between two documents based on direct citation can be defined as:

$$S_{i,j}^{dt} = \begin{cases} 1 & \text{if document } d_i \text{ cites } d_j \\ 0 & \text{otherwise} \end{cases} \tag{2.2}$$

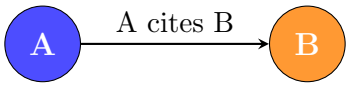

Figure 2.1: Direct Citation: document A cites document B

- **Co-Citation Analysis**

  Co-citation analysis examines how frequently two documents are cited together by other publications. Two documents are considered related when they are jointly cited by a third document [26].

  The similarity based on co-citation can be defined as:

$$S_{i,j}^{tco} = \frac{|C_i \cap C_j|}{|C_i \cup C_j|} \tag{2.3}$$

  where:

  – C_i and C_j represent the sets of documents that cite documents i and j, respectively.

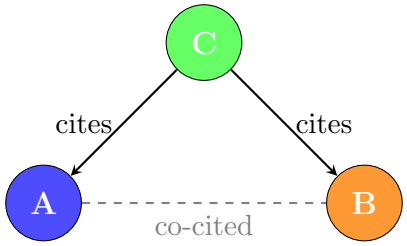

Figure 2.2: Co-Citation: document C cites both A and B

- **Bibliographic Coupling**

  Bibliographic coupling identifies the relationship between two documents by examining whether they share common references. Two documents are said to be coupled when they cite one or more references in common [27].

The similarity based on bibliographic coupling can be defined as:

$$S_{i,j}^{tb} = \frac{|C_i' \cap C_j'|}{|C_i' \cup C_j'|} \tag{2.4}$$

where:

– C_i' and C_j' represent the sets of documents cited by documents d_i and d_j, respectively.

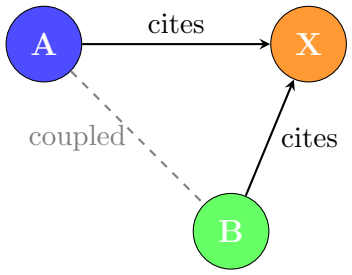

Figure 2.3: Bibliographic Coupling: A and B both cite X

2.4.2 Visualization Techniques

Citation analysis can be enhanced through visualization techniques that provide an intuitive representation of large-scale citation networks. In such representations, scientific documents are modeled as nodes, while citation relationships between them are represented as edges in a graph structure [16].

Several types of visualization are commonly used to analyze citation networks. First, direct citation graphs represent explicit citation relationships between documents, where a directed edge from one document to another indicates that the former cites the latter. Second, co-citation networks connect documents that are cited together by other publications. If two papers are frequently co-cited, they are likely to address related research topics or scientific contributions. Third, bibliographic coupling networks link documents that share common references, reflecting a shared knowledge base between them.

2.4.3 Comparison of Citation-Based Analysis Techniques

Technique	**Strengths**	**Limitations**	**Typical Applications**
Direct Citation	Simple and interpretable; reflects explicit document relationships	Sparse for recent publications; limited connectivity	Influence analysis and citation tracking
Co-Citation	Captures thematic and intellectual relationships	Requires sufficient citation data	Thematic clustering and community detection
Bibliographic Coupling	Effective for recent publications; independent of future citations	Sensitive to changes in citation patterns	Detection of emerging research topics

Table 2.2: Comparison of Main Citation-Based Analysis Techniques

2.4.4 Citation-Based Clustering Studies

Recent research has shown increasing interest in citation-based clustering for organizing and retrieving scientific documents. This approach has emerged as a powerful alternative to text-based methods for structuring scientific literature. Unlike lexical approaches, it exploits the structural relationships between documents through citation networks, enabling the discovery of intellectually coherent research communities. However, despite its advantages, citation-based clustering presents challenges related to scalability, algorithm selection, and the inherent sparsity or incompleteness of citation networks. As a result, recent studies have explored both classical network-based methods and hybrid approaches that integrate textual and citation information to improve clustering quality.

- **Citation Clusters for Information Retrieval**

  Bascur et al. [28] evaluated the effectiveness of citation-based clustering for academic information retrieval, particularly in systematic review tasks. Their approach

relied on constructing a large-scale citation network from the MEDLINE database and organizing it into a hierarchical structure of citation clusters using network-based clustering methods. The evaluation, conducted on 25 systematic reviews, showed that citation-based clustering is especially useful in recall-oriented scenarios, as it helps identify additional relevant documents. Overall, the study suggests that citation-based clustering should be seen as a complementary approach that improves retrieval coverage rather than as a replacement for text-based methods.

- **Systematic Comparison of Citation-Based Methods**

  Šubelj et al. [29] compared clustering methods for scientific publications based on citation relations. The study evaluated different algorithm families, including spectral methods, modularity optimization, map equation, and label propagation.

  The results showed that no single method consistently performs best, due to trade-offs between clustering quality and efficiency. However, map equation methods, particularly Infomap, achieved the best overall performance.

- **Large-Scale Clustering with the Leiden Algorithm**

  Traag et al. [30] introduced the Leiden algorithm for large-scale clustering of networks, including citation-based datasets. This method improves upon the Louvain algorithm by guaranteeing well-connected communities and producing more accurate and stable partitions. Moreover, its computational efficiency makes it suitable for large-scale bibliometric analysis.

- **Hybrid Citation-Text Approaches**

  Recent studies have increasingly focused on hybrid approaches that combine textual information with citation relations to improve the clustering of scientific documents. These methods aim to overcome the limitations of single-view approaches by integrating complementary sources of information.

  Yu et al. [31] proposed a hybrid self-optimized clustering model that combines citation-based similarity, derived from co-citation and bibliographic coupling, with text-based similarity based on TF-IDF and topological features. The integrated similarity is then used to cluster scientific publications using the Louvain algorithm. Experimental results demonstrated that the hybrid model produces more accurate

and meaningful clustering results compared to using citation or textual features alone.

- **Extended Citation Model**

  Zhang et al. [32] proposed an extended citation model for scientific document clustering that integrates multiple citation relations, including direct citation, co-citation, and bibliographic coupling, together with a textual similarity network. The model further incorporates important citation characteristics such as citation frequency, distribution, and proximity between cited documents, providing a more refined representation of document relationships. Experimental results demonstrated that this hybrid approach outperforms traditional single-source methods in terms of precision, recall, and F1-score.

- **Clustering at Scale with Citation Data**

  Recent studies have investigated the scalability of citation-based clustering methods on large scientific datasets. For example, Huong and Koch [33] evaluated clustering algorithms on a citation graph extracted from the Web of Science, comprising approximately 700,000 publications and 4.6 million citation links. The study compared spectral clustering, Louvain, and Leiden methods.

  The results showed that spectral clustering does not scale well to large datasets, failing to produce results within reasonable time, while Louvain and Leiden algorithms demonstrated high computational efficiency. However, meaningful clustering results require careful parameter tuning, particularly for large and heterogeneous citation networks.

2.4.5 Comparative Summary of Recent Citation-Based Clustering Studies

Study	**Data**	**Approach**	**Algorithms**	**Key Findings**
Bascur et al. [28]	MEDLINE	Citation-based clustering (IR)	Hierarchical, network-based	Enhances recall; supports systematic reviews
Šubelj et al. [29]	Web of Science	Comparative citation-based analysis	Spectral, modularity, Infomap	No universal best method; performance varies
Traag et al. [30]	Large-scale networks	Community detection	Louvain, Leiden	Leiden improves stability and connectivity
Yu et al. [31]	Scientific publications	Hybrid (text + citation)	Louvain	Improves clustering quality
Zhang et al. [32]	Citation network	Extended citation + text	Graph-based	Improves precision, recall, and F1-score
Huong & Koch [33]	Web of Science	Large-scale evaluation	Spectral, Louvain, Leiden	Scalable methods; spectral fails; tuning required

Table 2.3: Comparative Summary of Recent Citation-Based Clustering Studies

2.5 Comparative Analysis of Text-Based and Citation-Based Approaches

Text-based and citation-based approaches represent two complementary paradigms for clustering scientific documents. While text-based methods exploit the semantic information contained in document content (e.g., titles, abstracts, and full text), citation-based approaches capture structural relationships through citation links between publications.

Text-based approaches enable the identification of thematic and semantic similarities between documents. In contrast, citation-based methods reveal the influence, connectivity, and knowledge flow within the scientific community. As a result, these two per-

spectives provide complementary information, and their combination has been shown to significantly improve clustering performance in scientific document analysis.

Aspect	**Text-Based**	**Citation-Based**
Information Source	Textual content (titles, abstracts, full text)	Citation links between documents
Similarity Basis	Semantic and thematic similarity	Structural and relational similarity
Main Advantage	Captures topical and semantic relationships between documents	Captures influence and connectivity within the scientific community
Main Limitation	May ignore structural relationships and is sensitive to vocabulary variability	May ignore textual content and is affected by citation sparsity
Typical Methods	TF-IDF, LDA, Word2Vec, Doc2Vec	Direct citation, co-citation, bibliographic coupling

Table 2.4: Comparison between Text-Based and Citation-Based Approaches

2.6 Challenges in Scientific Document Clustering

Clustering scientific documents presents several challenges due to the complex nature of textual and bibliographic information contained in scholarly publications. These challenges can significantly affect the performance of clustering algorithms and the quality of the resulting document groups.

2.6.1 High-Dimensional Text

Scientific documents are typically represented using textual features extracted from titles, abstracts, or full texts. Since the vocabulary of a document collection may contain tens or even hundreds of thousands of terms, document representations become highly

dimensional and sparse. This phenomenon leads to the well-known curse of dimensionality, which can negatively affect similarity measures and clustering performance [2].

2.6.2 Vocabulary Variability

Another challenge arises from the variability of vocabulary used by different authors. The same concept can often be expressed using different terms or expressions, making it difficult to accurately capture semantic similarity between documents using purely lexical representations [1].

2.6.3 Semantic Ambiguity

Words may have multiple meanings depending on the context in which they are used. This phenomenon, known as semantic ambiguity or polysemy, can cause documents with similar terms to refer to different topics, thereby complicating the clustering process [1].

2.6.4 Citation Sparsity

Although citation relationships provide valuable information about connections between scientific publications, citation networks are often sparse. Many documents may contain only a limited number of citations, making it difficult to reliably estimate structural relationships between documents [16].

2.6.5 Multi-Topic Papers

Scientific papers frequently address multiple research topics within a single document. As a result, a document may belong to more than one thematic group, which complicates the clustering task and makes it difficult to assign documents to a single cluster [2].

2.7 Limitations of Single-View Clustering

Single-view clustering approaches for scientific documents present several limitations. These methods typically rely on a single representation of documents, such as textual features extracted from titles, abstracts, or full texts. However, relying on only one type of information is often insufficient to capture the complex nature of scientific publications.

Scientific documents can be represented through multiple views, including textual content, citation links, and author information. When clustering is based on a single representation, part of this information is ignored, leading to incomplete document representations.

Moreover, textual representations usually result in high-dimensional and sparse feature spaces, which can negatively affect similarity measures and reduce the effectiveness of clustering algorithms [2, 1].

These limitations have motivated the development of multi-view clustering approaches, which integrate multiple sources of information in order to improve clustering performance [34].

2.8 Conclusion

In this chapter, we explored scientific document clustering by presenting the main characteristics of scientific documents and the different approaches used for their analysis, including both text-based and citation-based methods.

We also highlighted several challenges, such as high-dimensional data, vocabulary variability, semantic ambiguity, citation sparsity, and multi-topic documents, which make the clustering process more complex.

These challenges reveal the limitations of single-view approaches, thereby motivating the use of more advanced methods such as multi-view clustering, which will be introduced in the next chapter.

Chapter 3

Foundations of Multi-View Clustering

3.1 Introduction

The rapid expansion of data generated by modern information systems has led to an increasing diversity of data representations. In many real-world applications, objects are described by multiple heterogeneous feature sets originating from different sources or modalities. Traditional clustering techniques generally assume a single homogeneous feature space, which limits their ability to capture the complexity of such data.

Multi-view learning has emerged as an effective paradigm to address this limitation by exploiting multiple representations of the same data. In this context, multi-view clustering aims to discover meaningful group structures without labeled data. By integrating complementary information from different views, it improves clustering accuracy, enhances robustness to noise, and provides more reliable results.

Unlike single-view clustering, where each instance is described by a single feature vector, multi-view clustering considers multiple views that provide partial but complementary information. Their integration enables a more comprehensive understanding of the data structure. As a result, multi-view clustering has gained attention in domains such as document analysis, multimedia processing, bioinformatics, and social network analysis.

This chapter introduces the fundamental concepts of multi-view clustering. It presents the problem definition, key principles such as complementarity and consensus, major algorithm categories, and fusion strategies. It also briefly reviews deep multi-view clustering and discusses its advantages and challenges.

3.2 Definition of Multi-View Clustering

Multi-view clustering refers to unsupervised learning methods designed to cluster data represented by multiple heterogeneous feature sets, known as *views*. These views originate from different sources or extraction processes and describe the same data instances from different perspectives.

According to [35], multi-view data consist of multiple heterogeneous representations that share common semantic information while exhibiting distinct properties. Each view alone is often insufficient to capture the full data structure, as it provides only partial information. Traditional single-view clustering methods, or simple feature concatenation, often fail to effectively exploit such data due to ignoring view heterogeneity.

[36] highlight that multi-view clustering relies on the assumption of *view consistency*, where the true cluster assignment should be consistent across views. Based on this assumption, algorithms aim to find clustering solutions that agree across views while leveraging their complementary information.

Furthermore, [37] describe multi-view clustering as an integrative framework that combines multiple representations to reduce noise and uncover more meaningful patterns than single-view approaches.

In summary, multi-view clustering is an unsupervised approach that partitions data into coherent groups by jointly exploiting multiple views, ensuring consistency while integrating complementary information to better reveal the underlying data structure.

3.3 Principles of Multi-view Clustering (MvC)

The process of Multi-view Clustering (MvC) is founded on two fundamental principles that explain its effectiveness and guide its modeling and application, namely the complementary principle and the consensus principle. These principles clarify the underlying assumptions of MvC and determine how multiple sources of information can be exploited effectively.

3.3.1 Complementary Principle

The complementary principle assumes that employing multiple views to describe the same data objects allows for a more comprehensive and accurate representation compared

to relying on a single view. As stated in the survey:

"multiple views should be employed in order to describe data objects more comprehensively and accurately" [35].

Although each individual view may be sufficient to accomplish a specific task, different views often contain complementary information. For instance, in image analysis, one view may capture texture features, while another describes edges or global structure, making the integration of these views essential for a deeper understanding of the data structure. This principle highlights the importance of exploiting non-overlapping information across different views to enhance clustering quality [35].

3.3.2 Consensus Principle

The consensus principle focuses on maximizing the consistency and agreement among different views by seeking a shared representation that reflects the common characteristics across all views. According to Yang and Wang (2018), this principle aims to reduce discrepancies between hypotheses derived from each view individually, as minimizing disagreement across views leads to improved overall clustering performance.

This principle constitutes the foundation of many multi-view learning algorithms, such as co-training methods, which aim to reinforce mutual agreement between different views in order to achieve a more stable and reliable clustering structure.

In summary, the complementary and consensus principles play a central role in addressing multi-view clustering problems. The former ensures the exploitation of diverse information, while the latter guarantees coherence among different views. Considering both principles jointly is a key condition for fully leveraging the potential of Multi-view Clustering [35].

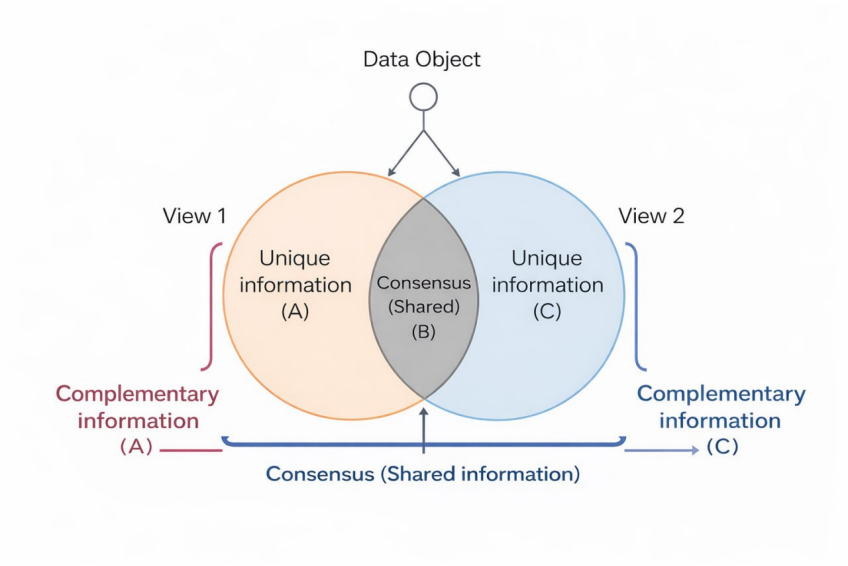

Figure 3.1: Schematic representation of complementary and consensus principles

3.4 The General Procedure of Multi-View Clustering

Despite the diversity of mathematical formulations and optimization strategies proposed in the literature, most Multi-View Clustering (MVC) methods follow a unified conceptual pipeline. The central objective is to jointly exploit multiple heterogeneous feature spaces describing the same set of instances in order to learn a consistent and robust clustering structure [35]. This general procedure can be organized into four main stages: view-specific representation, representation learning, multi-view fusion, and final clustering.

Step 1: View-Specific Data Representation

The process begins by constructing multiple views, where each view corresponds to a distinct feature space derived from different modalities or sources. Each representation captures a specific perspective of the same objects. For example, in scientific document clustering, one view may encode textual information (e.g., TF-IDF features), another may represent citation relationships as a graph structure, while a third may incorporate metadata or semantic embeddings. These heterogeneous representations form the foundation of multi-view analysis.

Step 2: Representation Learning / Dimensionality Reduction

Multi-view data are often high-dimensional and noisy. Therefore, many MVC approaches incorporate a representation learning stage to extract compact and discriminative latent features for each view. Techniques such as subspace learning, matrix factorization, graph-based embedding, and deep learning models are frequently employed to uncover the intrinsic structure of each view while reducing redundancy and noise.

Step 3: Multi-View Information Fusion

The fusion stage constitutes the core component of MVC. The goal is to integrate multiple representations while preserving both shared (consensus) information and complementary information across views. Fusion strategies are generally categorized into:

- Feature-level (early) fusion, where features are concatenated prior to clustering;
- Decision-level (late) fusion, where clustering is performed separately on each view and combined through consensus mechanisms;
- Joint learning frameworks, where view-specific representations and the consensus clustering are optimized simultaneously within a unified objective function.

Recent developments emphasize joint optimization models due to their ability to balance consistency and complementarity across views.

Step 4: Final Clustering

After obtaining either a unified latent representation or a fused similarity matrix, a clustering algorithm such as K-means or spectral clustering is applied to generate the final partition. The resulting clusters reflect the integrated structural information from all views.

The general MVC framework provides a structured mechanism for integrating heterogeneous data sources through sequential representation construction, principled fusion, and consensus-driven partitioning. This process improves clustering robustness, reduces sensitivity to noisy views, and enhances the interpretability and stability of the final clusters [35].

3.5 Fusion Strategies in Multi-View Clustering

Fusion strategies represent a key component in multi-view or multi-source clustering, as they define how information originating from different data views is integrated during the clustering process. According to Rappoport and Shamir [37], integration strategies can be categorized into three main types depending on the stage at which the fusion occurs: early integration, late integration, and intermediate integration.

3.5.1 Early Integration

The first strategy, early integration, consists of combining data from multiple sources before performing clustering. In this approach, feature matrices from different views are concatenated into a single unified matrix, after which traditional single-view clustering algorithms are applied. As described by the authors, early integration "concatenates omic matrices to form a single matrix with features from multiple omics" [37]. Although this strategy is straightforward and easy to implement, it may suffer from several limitations, such as high dimensionality and the inability to account for different data distributions across views.

3.5.2 Late Integration

The second strategy is late integration, where each view is clustered independently and the resulting clustering solutions are combined afterward. According to Rappoport and Shamir [37], late integration involves clustering each data source separately and then integrating the clustering results to produce a final solution. This strategy preserves the individual characteristics of each view; however, it may fail to exploit shared information during the learning process itself.

3.5.3 Intermediate Integration

To address the limitations of early and late integration, the authors introduce intermediate integration, which builds a unified model that jointly incorporates all data sources during the learning phase. Instead of merging features or final results, this strategy integrates information directly within the model, allowing the discovery of more refined and

informative structures. As emphasized by Rappoport and Shamir [37], intermediate integration approaches attempt to build models that simultaneously exploit multiple sources in a shared framework.

In summary, fusion strategies in multi-view clustering can be understood as different levels of integration: before clustering (early integration), after clustering (late integration), or during the learning process (intermediate integration). The choice of fusion strategy directly influences the effectiveness of clustering by determining how information from multiple views is exploited.

To better illustrate the differences between fusion strategies,Table 3.1 summarizes their main characteristics, advantages, and limitations

Fusion Strategy	**Stage of Integration**	**Main Idea**	**Advantages**	**Limitations**
Early Integration	Before clustering	Concatenate features from multiple views into a single matrix	Simple implementation; allows use of traditional clustering algorithms	High dimensionality; ignores view-specific distributions
Late Integration	After clustering	Cluster each view independently and combine clustering results	Preserves individual view characteristics; flexible integration	Does not exploit shared information during learning
Intermediate Integration	During learning	Build a unified model integrating multiple views simultaneously	Captures shared and complementary information; more robust results	Higher model complexity; requires advanced optimization

Table 3.1: Comparison of Fusion Strategies in Multi-View Clustering

3.6 Categories of Multi-View Clustering Algorithms

Within the framework of categorizing Multi-View Clustering (MVC) algorithms, several studies propose a taxonomy of methodological approaches that differ according to the level and mechanism of information fusion across multiple views [35].

3.6.1 Co-Training Style Algorithms

The first category consists of co-training style algorithms, which are based on the principle of collaborative learning between views. In this approach, clustering results in each view are iteratively improved through the exchange of information across views. This direction relies on the assumption that each view contains complementary information that can progressively enhance the clustering performance of the other views through mutual supervision.

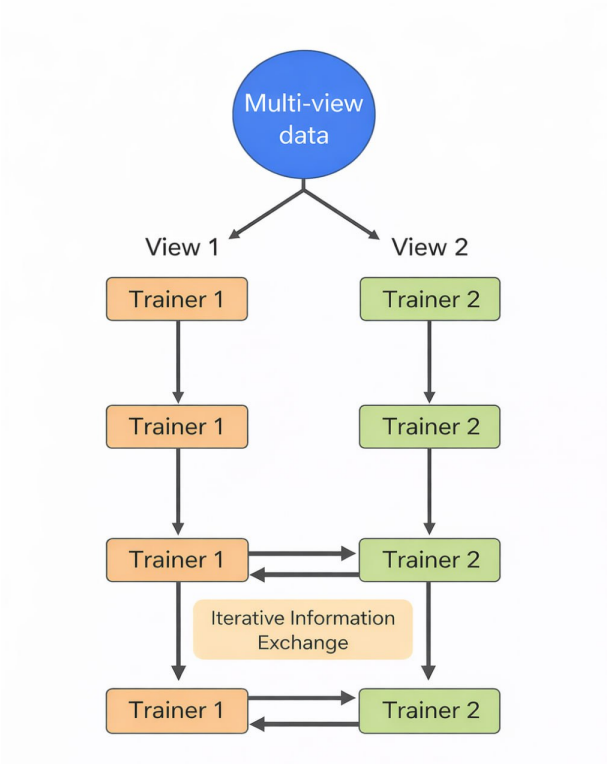

Figure 3.2: Schematic representation of the co-training process

3.6.2 Multi-Kernel Learning

The second category is multi-kernel learning, which represents each view using a specific kernel function and then combines these kernels into a unified kernel employed for clustering. This strategy enables the capture of nonlinear relationships within the data and ensures a balanced contribution of each view by learning appropriate weights during the integration process.

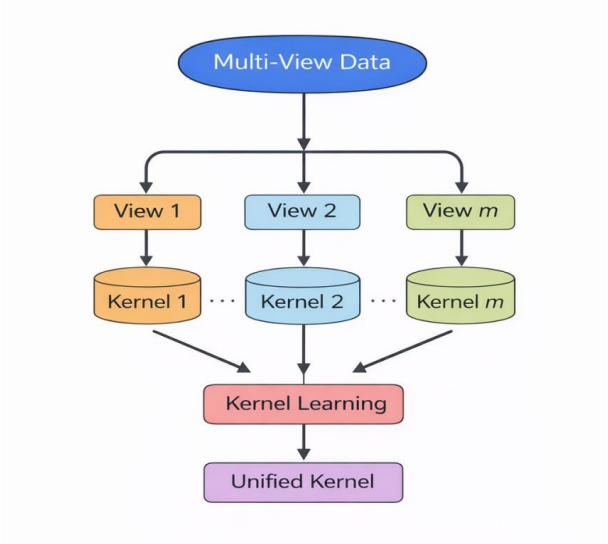

Figure 3.3: Schematic representation of the multi-kernel learning process

3.6.3 Multi-View Graph Clustering

The multi-view graph clustering category is based on constructing a graph for each view, where nodes represent data samples and edges reflect similarity relationships between them. The objective is to learn a fused graph that reflects the shared underlying structure of the data across all views. Once this integrated graph is obtained, techniques such as spectral clustering are applied to generate the final clustering results.

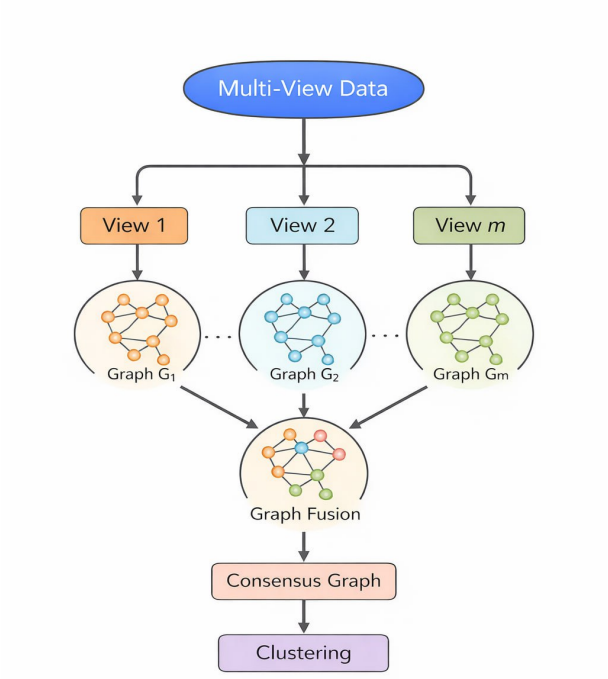

Figure 3.4: Schematic representation of the graph-based clustering process

3.6.4 Multi-View Subspace Clustering

In contrast, multi-view subspace clustering assumes that data from different views lie in shared low-dimensional subspaces. The goal is to learn a common latent representation that reduces the impact of noise while preserving the distinctive characteristics of each view.

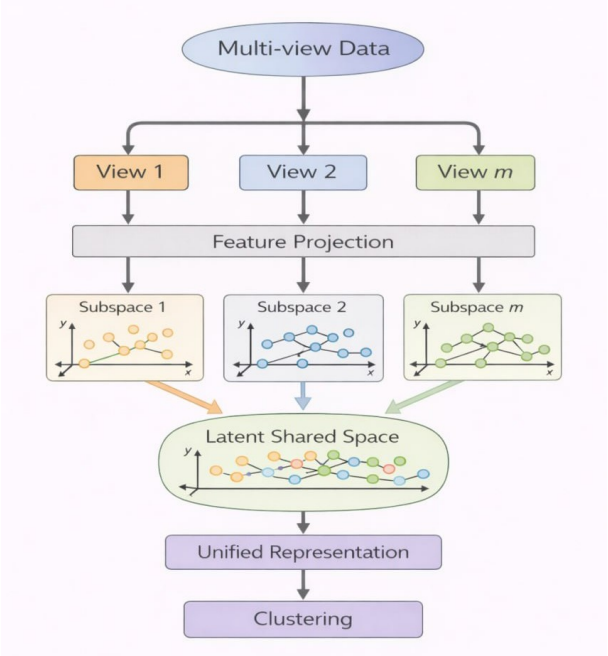

Figure 3.5: Schematic representation of the multi-view subspace clustering process

3.6.5 Multi-Task Multi-View Clustering

Finally, multi-task multi-view clustering combines the concepts of multi-view learning and multi-task learning. In this setting, multiple related clustering tasks—each possibly involving multiple views—are learned jointly. Knowledge sharing across tasks is exploited to improve overall clustering performance.

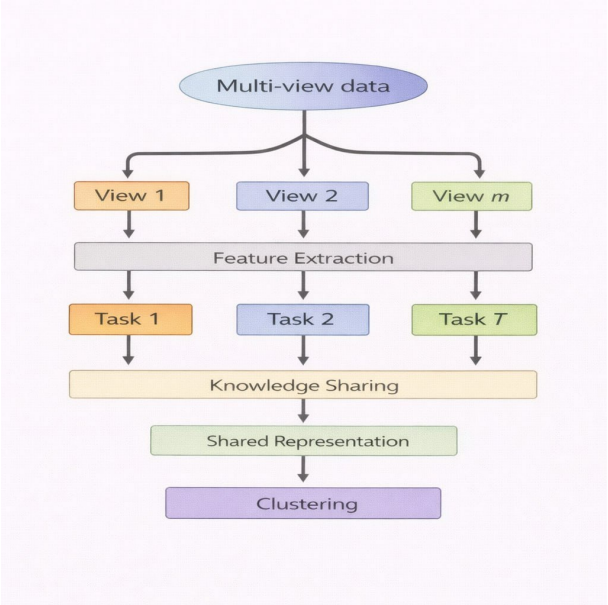

Figure 3.6: Schematic representation of multi-task multi-view clustering process

Overall, the distinction between these methodological directions does not lie in their ultimate objective—namely, obtaining a consensus clustering structure—but rather in the level at which information fusion is performed (representation level, kernel level, graph level, subspace level, or task level) and in the strategy adopted to achieve this consensus.

3.7 Deep Multi-View Clustering

Deep Multi-View Clustering (DMVC) has emerged as an important direction in unsupervised learning by integrating deep representation learning into the multi-view clustering framework. Traditional multi-view clustering methods often rely on linear assumptions or shallow mappings, which limit their ability to capture complex nonlinear structures in real-world data [38].

In contrast, deep learning provides powerful nonlinear modeling capabilities and enables automatic extraction of high-level features from heterogeneous data. As highlighted by Fu [39], DMVC combines deep learning and multi-view learning to improve feature extraction, information fusion, and clustering performance. This allows models to jointly optimize representation learning and clustering in an end-to-end manner.

3.7.1 Architectural Categorization of Deep Multi-View Clustering

DMVC methods can be categorized into four main families [39, 38]:

- **Autoencoder-Based Methods**

  These approaches use encoders to learn latent representations and decoders to reconstruct input data. The objective combines reconstruction loss and clustering constraints, enabling effective nonlinear feature learning and improved clustering performance.

- **Self-Representation-Based Methods**

  Based on subspace clustering, these methods represent each sample as a combination of others. Deep networks learn nonlinear transformations before constructing the self-representation matrix, capturing intrinsic data structures and improving cross-view consistency.

- **Contrastive Learning-Based Methods**

  These methods maximize similarity between views of the same instance and minimize similarity between different instances. They enhance representation quality and enforce semantic consistency across views [40].

- **Ensemble-Based Methods**

  Ensemble approaches combine multiple clustering results from different views or model initializations to improve robustness and stability. Consensus mechanisms help exploit complementary information effectively.

3.8 Challenges in Multi-View Clustering

Despite its advantages in exploiting complementary information from multiple representations, multi-view clustering (MVC) still faces several challenges. One major issue arises from the high-dimensional feature space resulting from the integration of multiple views. Although combining different representations enriches data description, it also increases computational complexity and may aggravate the curse of dimensionality [41].

Scalability is another important challenge, as many MVC algorithms incur high computational costs when applied to large-scale datasets. Therefore, developing efficient and scalable approaches is essential for handling large collections of multi-view data.

Another difficulty lies in the heterogeneity of views. Different views may vary in modality, scale, or quality, making their integration a complex task. Furthermore, redundancy or strong correlations between views can negatively affect clustering performance if not properly managed.

Finally, determining the relative importance of each view and selecting an appropriate fusion strategy remain challenging tasks. Since not all views contribute equally to clustering quality, adaptive weighting mechanisms and effective fusion strategies are often required.

3.9 Benefits of Multi-View Clustering

Compared to traditional single-view clustering methods, multi-view clustering (MVC) offers several advantages by exploiting complementary information from multiple data representations. Integrating different views enables the discovery of more robust clustering structures and improves clustering quality [42].

First, multi-view clustering provides a more comprehensive representation of the data. While single-view approaches capture only limited aspects of the data structure, multiple views offer complementary perspectives that better describe relationships between data points.

Second, combining multiple views helps reduce the impact of noise. If one view contains noisy or incomplete information, other views may still provide useful signals, allowing the clustering process to focus on consistent patterns across views.

Finally, multi-view clustering is particularly suitable for heterogeneous datasets where data naturally originates from multiple sources or modalities. By integrating these diverse representations, MVC enables a more effective analysis of complex data structures [42].

3.10 Conclusion

This chapter presented the fundamental concepts of multi-view clustering, including its definition, core principles, and main algorithmic categories. Different fusion strategies

were reviewed, together with an overview of deep multi-view clustering and its advantages over traditional approaches. In addition, key challenges and benefits of MVC were briefly discussed. Overall, this chapter provides the theoretical foundation necessary for understanding advanced multi-view clustering techniques developed in the following chapters.

Chapter 4

Multi-View Clustering for Scientific Documents (Case of Biomedical Articles)

4.1 Introduction

With the rapid growth of scientific publications, organizing and extracting meaningful knowledge from large collections of documents has become a challenging task. Traditional approaches that rely solely on textual content often fail to capture the complex relationships that exist between scientific articles, particularly those reflected through citation links.

In this context, multi-view clustering has emerged as an effective approach for improving clustering performance by integrating multiple sources of information. Scientific documents can naturally be represented from different perspectives, such as textual content and citation relationships, each providing complementary insights into the structure of the data.

In this chapter, we apply a multi-view clustering framework to a collection of biomedical scientific articles. The proposed approach combines textual representations and citation-based information in order to produce more coherent and informative clusters. By leveraging these multiple views, we aim to demonstrate the advantages of multi-view learning over single-view approaches in the context of scientific document clustering.

4.2 General Pipeline of the Proposed Approach

To operationalize the proposed multi-view clustering methodology, we designed a structured pipeline that integrates the different stages of data processing, representation, and fusion. This pipeline provides a clear overview of how heterogeneous information sources are transformed into coherent clusters of scientific documents.

The process begins with data extraction, where biomedical articles are collected along with their textual content and citation information. These raw inputs are then processed into two complementary views: the textual view, which captures semantic similarity through vector representations such as TF-IDF or embeddings, and the citation view, which models structural relationships using bibliographic coupling or co-citation analysis.

Once both views are established, the framework applies fusion strategies to integrate them. In the early fusion approach, feature representations from both views are concatenated prior to clustering, producing a unified feature space. In contrast, the late fusion approach combines the similarity matrices derived from each view through a weighted linear combination, before applying the clustering algorithm.

Finally, the fused data are subjected to clustering algorithms, yielding groups of documents that reflect both semantic and structural coherence. The resulting clusters are evaluated using quantitative metrics to assess cohesion and separation, ensuring that the framework produces meaningful and interpretable outcomes.

This pipeline, illustrated in Figure 4.1, serves as the backbone of our methodology. It demonstrates how multi-view clustering can systematically combine diverse sources of information to overcome the limitations of single-view approaches and enhance the organization of biomedical literature.

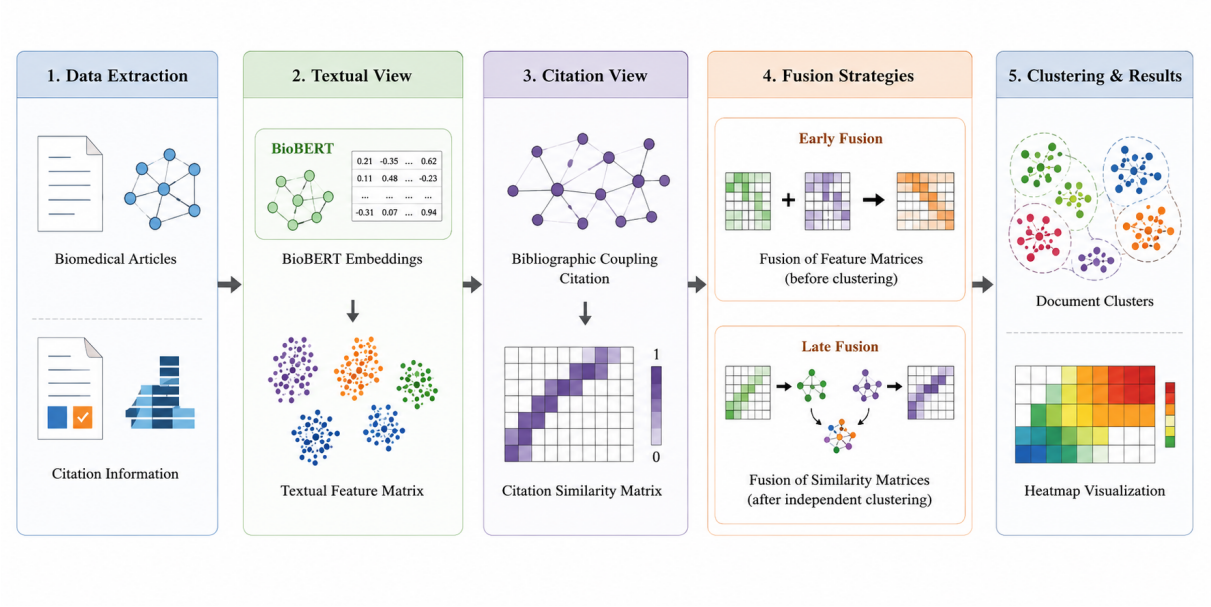

Figure 4.1: Multi-view clustering pipeline

4.2.1 Data Extraction and View Selection

The dataset used in this work was obtained from **PubMed Central (PMC)**, one of the largest open-access repositories of biomedical scientific literature. Each article in the dataset is identified by a unique **PMID** (PubMed Identifier). For each article, we collected the following information: title, abstract, list of references, and article identifier (PMID). The raw data was then structured into two separate dataframes: one containing the textual metadata of each article, and one containing the bibliographic citation relationships between articles.

Scientific documents can be represented through several views, including the Textual View, the Citation View, the Author Profile View, and other metadata views such as journal names or keywords.

These different views provide complementary perspectives on the same dataset, as each view reflects a specific aspect of the characteristics of scientific documents. However, the selection of views mainly depends on the context of the study, as well as on data availability and quality.

In this work, we selected two main views: the **Textual View** and the **Citation View**. This decision was made based on their ability to provide complementary semantic and structural information, as well as their relevance to the objectives of our clustering task.

- **Textual View:** The textual view relies on the content of scientific documents, including **titles, abstracts, and keywords**. As discussed in Chapter 2, these elements constitute essential components of a scientific document, as they provide a semantic representation of its content. In this work, we used this view to measure the semantic similarity between documents by analyzing textual content and extracting relevant linguistic and semantic features. Therefore, the textual view is considered one of the most important views in clustering tasks, especially when focusing on the semantic and knowledge-based content of documents.

- **Citation View:** The citation view is based on the relationships between scientific documents through bibliographic references. This view reflects the network structure of scientific knowledge by showing how research papers are interconnected. We adopted this view to model structural relationships between documents, which are independent of textual content, thus providing a complementary perspective to the textual view. This representation also helps in identifying scientific communities and shared research topics.

4.2.2 Textual View: Representation

The construction of the textual view starts by building a unified textual field for each scientific article. In our work, we rely only on the abstract as the main textual source.

This choice is motivated by the fact that titles and keywords are not always consistent or sufficiently informative across documents, while abstracts generally provide a more complete and structured description of the article content.

Afterward, a preprocessing phase is applied to clean and normalize the textual data before feature extraction. This includes:

- conversion to lowercase,
- removal of numerical values and punctuation,
- removal of extra spaces,
- stop-word removal,
- filtering of short tokens.

This preprocessing step results in a cleaned textual representation that is then fed into the BioBERT model. Although BioBERT is capable of processing raw text, applying basic cleaning beforehand helps reduce noise and ensures more consistent input representations.

Several textual representation methods were discussed in Chapter 1, including TF-IDF, LDA, LSI, and embedding-based models such as Word2Vec and BERT.

To construct the textual view of our framework, we focus on representing the semantic content of biomedical articles through their abstract fields. Each abstract is processed using **BioBERT** [43], a transformer-based language model pre-trained on large biomedical corpora such as PubMed and PMC. BioBERT converts each document into a high-dimensional embedding vector that captures contextual meaning and domain-specific terminology more effectively than traditional statistical models.

Because these embeddings are typically high-dimensional, a dimensionality reduction step is applied to facilitate visualization and subsequent analysis. We employ **UMAP** [44] (Uniform Manifold Approximation and Projection), which projects the BioBERT embeddings into a lower-dimensional space while preserving both local and global semantic structures. The resulting representation provides a compact and interpretable view of the corpus, where semantically related abstracts are positioned closer together in the reduced space.

Thus, our choice of BioBERT and UMAP is based on the following:

- improved representation of biomedical semantic information,
- effective dimensionality reduction,
- improved cluster separability.

This process, illustrated in Figure 4.2,constitutes the textual representation stage of our multi-view framework. It transforms raw biomedical text into meaningful semantic vectors, enabling the integration of textual information with other complementary views such as citation-based relationships.

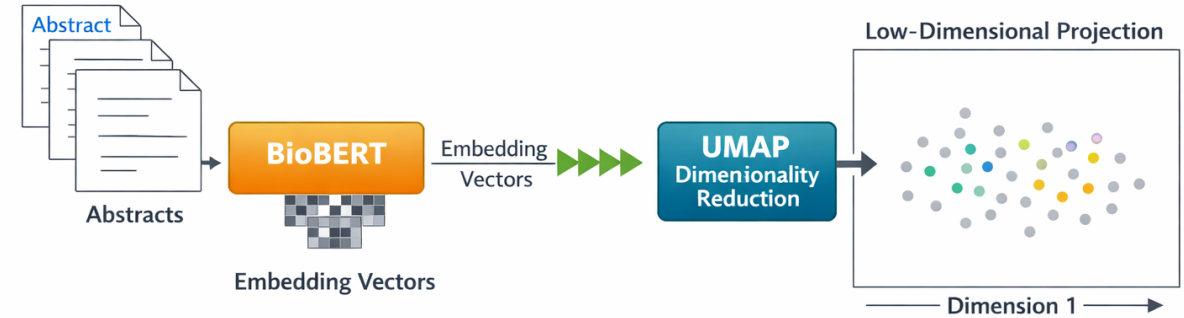

Figure 4.2: Textual representation pipeline using BioBERT and UMAP.

The semantic vectors produced by this pipeline are used to compute textual similarity between documents using **cosine similarity**, which measures the angle between two embedding vectors and is well-suited for dense high-dimensional representations. The resulting similarity matrix serves as input for the subsequent fusion stage.

4.2.3 Citation View: Representation

Several methods can model relationships between scientific documents, as discussed in Chapter 2: Direct Citation, Co-citation, Graph-based Approaches, and Bibliographic Coupling. Each defines similarity differently.

- **Direct Citation** links a document to another when it cites it directly. This relation is asymmetric and limited, reflecting a dependency rather than thematic proximity.
- **Co-citation** considers two documents similar if they are cited together by other papers. However, it depends heavily on incoming citations, which may be scarce for recent or less-cited works.
- **Graph-based Approaches** capture complex relationships but require intricate construction and entail higher computational cost.
- **Bibliographic Coupling**, in contrast, measures similarity through shared references. It provides a direct and stable similarity measure, even when citation counts are low.

Given these characteristics, **Bibliographic Coupling** was selected for our framework because it:

- is well-suited for scientific articles,
- does not depend on document popularity,
- provides a more stable and directly usable similarity measure.

In our case, in bibliographic coupling, we compute the similarity between two documents using the Jaccard index, which measures the overlap between their reference sets:

$$J(A, B) = \frac{|R_A \cap R_B|}{|R_A \cup R_B|} \tag{4.1}$$

where R_A and R_B denote the sets of references of documents A and B, respectively.

Figure 4.3 illustrates the main steps from citation extraction to the similarity matrix construction.

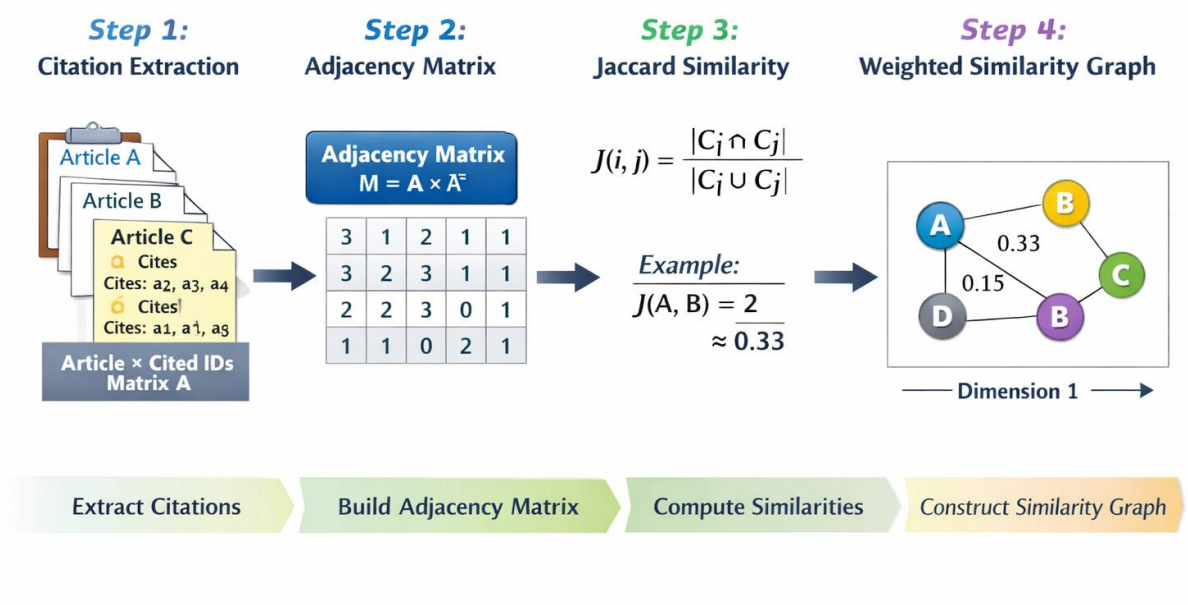

Figure 4.3: Citation view construction pipeline.

- **Step 1 – Extract Articles and Their References**

  Each article is associated with the list of documents it cites. For example:

Article	Cited IDs
A	a_1, a_2, a_3, a_4
B	a_2, a_3, a_5, a_6
C	a_1, a_7, a_8

This information forms the article–reference incidence matrix A, where rows represent articles and columns represent cited references.

- **Step 2 – Build the Adjacency Matrix**

  From the incidence matrix A, we construct the adjacency matrix M representing shared references between articles:

$$M = A \cdot A^T$$

  Each entry M_{ij} represents the number of common references between articles i and j.

- **Step 3 – Compute Jaccard Similarity**

  We transform the adjacency values into a normalized similarity measure using the Jaccard index:

$$J(i,j) = \frac{|C_i \cap C_j|}{|C_i \cup C_j|}$$

  Example for articles A and B:

$$C_A \cap C_B = \{a_2, a_3\} \Rightarrow |C_A \cap C_B| = 2$$

$$C_A \cup C_B = \{a_1, a_2, a_3, a_4, a_5, a_6\} \Rightarrow |C_A \cup C_B| = 6$$

$$J(A,B) = \frac{2}{6} \approx 0.33$$

- **Step 4 – Construct the Weighted Similarity Graph**

We use the Jaccard similarity matrix to build a weighted graph, where nodes represent articles and edges represent similarity values. Edges with weights below a chosen threshold (e.g., 0.01) are removed to retain only meaningful connections.

4.2.4 Fusion Strategy

Several fusion strategies were discussed in Chapter 3, namely Early Integration, Late Integration, and Intermediate Integration. In this work, we experimented with two fusion strategies: Early Fusion and Late Fusion.

- **Early Fusion Strategy:**

  In this strategy, we adopted an Early Fusion approach, where different representations are combined before applying the clustering algorithm. The main objective is to construct a unified feature space that integrates both semantic information extracted from text and structural information derived from citation relationships.

  We considered two complementary data views:

  – A textual view based on the **BioBERT** model, which generates contextual semantic embeddings for each document directly, without any additional dimensionality reduction (see Section 4.2.2).

  – A bibliographic view based on **Bibliographic Coupling**, using the **Jaccard similarity coefficient**, which measures the strength of relationships between documents based on shared references (see Section 4.2.3).

  The bibliographic view initially produces a similarity matrix rather than explicit feature vectors. Therefore, it cannot be directly combined with textual embeddings. To address this issue, we transformed the similarity matrix into feature-based representations using two different approaches.

  The first approach applies **Spectral Embedding** [45], which relies on eigen-decomposition of the similarity matrix to project documents into a lower-dimensional space while preserving the global structure of the citation graph. **L2 normalization** was further applied to ensure numerical stability and balanced feature scaling.

  The second approach uses **Node2Vec** [46], a graph representation learning method based on biased random walks. Documents are modeled as nodes in a graph, and

citation relationships are represented as weighted edges. Random walks are then performed to generate node sequences, from which low-dimensional embeddings are learned, capturing both local neighborhood relationships and global structural properties of the graph.

After obtaining both representations, **feature concatenation** was performed to fuse them into a single unified feature space. Since the resulting fused feature space is high-dimensional, **UMAP** [44] was applied to reduce dimensionality while preserving the local structure of the data and improving cluster separability. Finally, the reduced representation was used as input to the **K-Means clustering algorithm** (K = 20).

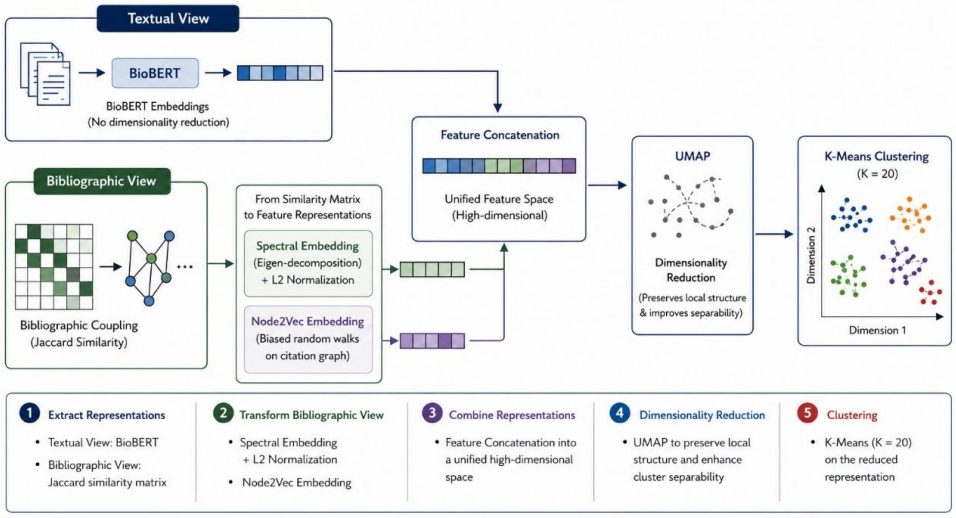

Figure 4.4: Early Fusion Framework for Textual and Bibliographic Views.

- **Late Fusion Strategy:**

  Unlike Early Fusion, Late Fusion combines views at the similarity level rather than at the feature level. This strategy preserves the intrinsic characteristics of each representation before integration. Both views were independently converted into similarity matrices:

  - **Textual Similarity Matrix:** Cosine similarity was computed using the BioBERT embeddings:

$$S_{text}(i,j) = \frac{v_i \cdot v_j}{\|v_i\|\|v_j\|} \tag{4.2}$$

– **Citation Similarity Matrix:** Similarity was computed using Bibliographic Coupling based on the Jaccard coefficient:

$$S_{cit}(i,j) = \frac{|R_i \cap R_j|}{|R_i \cup R_j|} \tag{4.3}$$

where R_i and R_j represent the sets of references cited by documents i and j.

– **Similarity Fusion:** Fusion was performed using a weighted linear combination:

$$S_{fusion} = \alpha \cdot S_{text} + (1-\alpha) \cdot S_{cit} \tag{4.4}$$

where $\alpha \in [0,1]$ controls the contribution of each view in the fusion process.

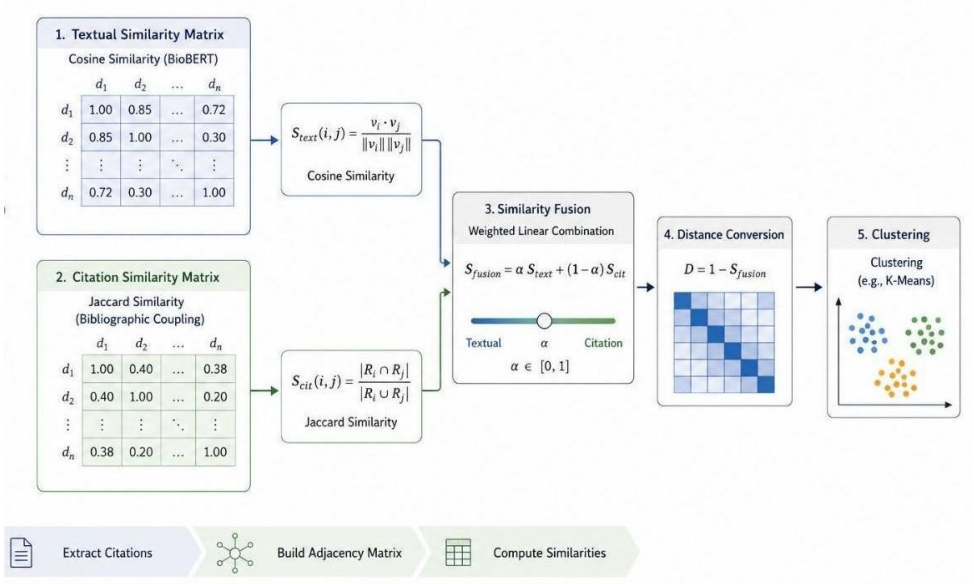

Figure 4.5: late Fusion Framework for Textual and Bibliographic Views.

4.2.5 Choice of Clustering Algorithm

Several clustering algorithms can be applied to the obtained representations, including *K-means*, *DBSCAN / HDBSCAN*, and *Spectral Clustering*.

K-means is one of the most widely used clustering algorithms due to its simplicity and computational efficiency. However, it requires the number of clusters to be specified in advance and assumes that clusters are approximately spherical and well-separated [11].

In contrast, density-based methods such as HDBSCAN are capable of detecting arbitrarily shaped clusters and handling noise points. Nevertheless, when applied to dense

embedding spaces, they may produce unstable results or classify a significant number of data points as noise[13] .

In our case, clustering is applied on representations obtained from early and late fusion. Early fusion is based on a feature matrix combining textual and citation information, while late fusion relies on similarity matrices derived from each view.

These representations are relatively structured, making them suitable for centroid-based methods. Therefore, we selected K-means for the following reasons:

- it is efficient and scalable,
- it performs well on structured representations,
- it provides stable clustering results,
- and it produces interpretable cluster assignments.

4.2.6 Clustering on Bibliographic Coupling Graph

For the bibliographic coupling view, the problem is naturally formulated as a graph-based clustering task, where documents are represented as nodes and similarities are represented as weighted edges.

Unlike embedding-based representations, this structure directly captures relational information derived from citation links. In this context, graph clustering methods are more appropriate than classical centroid-based approaches.

Therefore, the Louvain algorithm was used to cluster the bibliographic coupling graph. This choice is motivated by several properties: it automatically determines the number of clusters, it optimizes the modularity measure, it is highly effective for community detection in citation networks, and it is well-suited for sparse graph structures [47].

The modularity function optimized by the Louvain algorithm is defined as:

$$Q = \frac{1}{2m} \sum_{i,j} \left(A_{ij} - \frac{k_i k_j}{2m} \right) \delta(c_i, c_j) \tag{4.5}$$

where:

- A_{ij} is the weight of the edge between nodes i and j,
- k_i and k_j are the sum of edge weights connected to nodes i and j,

- m is the total weight of all edges in the graph,
- c_i and c_j are the communities of nodes i and j,
- $\delta(c_i, c_j)$ is an indicator function equal to 1 if both nodes belong to the same community and 0 otherwise [48].

4.3 Evaluation Metrics

Several internal evaluation metrics can be used to assess the quality of clustering results. In this work, we considered three widely used metrics:

- Silhouette Score
- Davies–Bouldin Index (DB)
- Calinski–Harabasz Index (CH)

Each of these metrics evaluates clustering quality from a different perspective.

- **Silhouette Score.** The Silhouette Score measures how well each data point fits within its assigned cluster compared to other clusters. It is defined as proposed by Rousseeuw [49]:

$$S(i) = \frac{b(i) - a(i)}{\max(a(i), b(i))} \tag{4.6}$$

where $a(i)$ is the average intra-cluster distance, and $b(i)$ is the distance to the nearest neighboring cluster. The score ranges from -1 to 1: values close to 1 indicate well-separated clusters, values close to 0 indicate overlapping clusters, and negative values indicate incorrect clustering. This metric simultaneously evaluates cohesion and separation.

- **Davies–Bouldin Index (DB).** The Davies–Bouldin Index evaluates cluster similarity based on intra-cluster dispersion and inter-cluster distance, as introduced by Davies and Bouldin [50]:

$$DB = \frac{1}{k} \sum_{i=1}^{k} \max_{j \neq i} \left(\frac{S_i + S_j}{M_{ij}} \right) \tag{4.7}$$

where S_i is the intra-cluster dispersion, M_{ij} is the distance between cluster centers, and k is the number of clusters. Lower values indicate better clustering quality.

- **Calinski–Harabasz Index (CH).** The Calinski–Harabasz Index is defined as proposed by Calinski and Harabasz [51]:

$$CH = \frac{\mathrm{Tr}(B_k)/(k-1)}{\mathrm{Tr}(W_k)/(N-k)} \tag{4.8}$$

  where $\mathrm{Tr}(B_k)$ is the between-cluster dispersion, $\mathrm{Tr}(W_k)$ is the within-cluster dispersion, N is the number of samples, and k is the number of clusters. Higher values indicate better clustering quality.

In our case, these three metrics were used jointly in order to validate the obtained results from multiple perspectives. This approach allows us to reduce the bias associated with relying on a single metric, confirm the consistency of the results, and obtain a more robust and reliable evaluation.

4.4 Conclusion

This chapter presented the methodological framework of the proposed multi-view clustering approach for scientific documents, with a focus on biomedical articles. It described the complete processing pipeline, starting from data extraction and view selection to the construction of textual and bibliographic representations.

The chapter also discussed different fusion strategies, namely Early Fusion and Late Fusion, designed to integrate semantic and structural information from multiple views. In addition, it addressed the choice of clustering algorithms by distinguishing between centroid-based methods such as K-means, applied to embedding spaces, and graph-based methods such as the Louvain algorithm, applied to citation networks.

Finally, the evaluation metrics used to assess clustering quality, including the Davies–Bouldin index and the Calinski–Harabasz index, were introduced.

Overall, this chapter provides a coherent and comprehensive framework for multi-view document clustering by combining heterogeneous representations with appropriate fusion and clustering strategies.

Chapter 5

Evaluation and Experiments

5.1 Introduction

This chapter presents the experimental evaluation of the proposed approach. It aims to assess the effectiveness of the different design choices made in the previous chapters, including data representation, fusion strategy, and clustering methodology.

We first describe the experimental setup, including the dataset used and the evaluation metrics adopted to measure clustering quality. Then, we present and analyze the obtained results for the different components of the proposed framework.

In particular, we evaluate the impact of both textual and bibliographic representations, as well as the performance of the embedding methods and the clustering algorithm. The goal is to provide a comprehensive analysis of the behavior of the proposed approach and to validate its effectiveness in grouping scientific documents into meaningful clusters.

5.2 Dataset Presentation

The quality and reliability of a clustering framework strongly depend on the characteristics of the dataset used during experimentation. Since the objective of this work is to cluster scientific documents by exploiting both semantic and bibliographic information, a dataset containing complete scientific articles along with their citation relationships is required.

To this end, a collection of biomedical scientific articles was used in this study. The dataset was designed to provide both rich textual content and citation information, en-

abling the construction of complementary representations of documents. This dual nature makes it particularly suitable for multi-view clustering, where both semantic and structural aspects of the data are exploited.

5.2.1 Source and Description

The dataset was obtained from the PubMed database [52] and is closely related to the PubMed Central (PMC) repository [53], a large open-access digital archive maintained by the U.S. National Library of Medicine.

PubMed provides access to a vast collection of biomedical literature, while PMC offers full-text scientific articles in structured XML format. This structured representation facilitates the extraction of relevant information, including textual content (titles, abstracts, and full text) as well as bibliographic data such as references and citation links.

The availability of both textual and citation information makes this dataset particularly appropriate for constructing the two views considered in this work: the textual view and the bibliographic (citation) view.

5.2.2 Metadata Extraction

The extracted XML files were parsed using dedicated XML processing tools in order to retrieve scientific articles classified as *research-article*. For each article, several metadata fields were collected and stored for subsequent analysis.

The extracted metadata include:

- Article title;
- Publication year;
- Abstract;
- Keywords;
- Journal title.

Whenever available, the PubMed Identifier (PMID) was used as the unique identifier for each article. The PMID provides a reliable mechanism for linking metadata records

with citation information and ensures consistency throughout the different processing stages.

Following the extraction phase, a total of **13,645** scientific articles were successfully collected from the PMC repository.

5.2.3 Citation Extraction

In addition to textual information, citation relationships were extracted in order to construct the bibliographic representation of the dataset.

Citation links were identified by analyzing the in-text citation markers (*xref* tags) and matching them with the corresponding entries in the reference lists. The PMID identifier was used whenever possible to establish explicit citation links between articles.

For each citing article, all referenced articles were identified and transformed into citation relations. Furthermore, citation frequencies were preserved in order to capture the strength of citation relationships.

The citation extraction process generated **504,662** citation relations linking scientific articles within the dataset. These citation links constitute the basis for constructing the bibliographic coupling graph and the citation-based similarity matrices used in the clustering experiments.

5.2.4 Data Cleaning

Raw scientific datasets often contain duplicate records, missing identifiers, and inconsistent citation links. Therefore, several data-cleaning operations were performed to improve data quality and ensure the reliability of the experimental results.

The cleaning process included:

- Removal of duplicate articles based on PMID identifiers;
- Elimination of duplicated citation relations;
- Verification of consistency between citing and cited articles;
- Removal of invalid or incomplete records.

After cleaning, the metadata dataset contained **10,996** unique scientific articles, while the citation dataset retained **409,610** valid citation relations. This preprocessing stage ensured that all subsequent analyses were conducted on a coherent and reliable dataset.

```
===== DATASET STATISTICS =====
Total number of articles: 13645
Number of articles (after removing duplicates): 10996
Total number of citation relations: 504662
Number of citation pairs (after removing duplicates): 409610
```

Figure 5.1: Dataset statistics

	article_id	title	abstract	keywords	journal_title	cited_id
0	28197375	Tumor-infiltrating lymphocyte composition, org...	The clinical relevance of tumor-infiltrating l...	None	Unknown	26308244
1	28123890	Novel treatment strategy with autologous activ...	This proof-of-concept single-arm open-label ph...	None	Unknown	11698225
2	28123889	Distinct transcriptional changes in non-small ...	We recently completed a phase I/IIa trial of R...	Biomarker candidates; cancer immunotherapy; mR...	Unknown	25056707
3	28123900	Standard radiotherapy but not chemotherapy imp...	Introduction: Advanced non-small cell lung can...	None	Unknown	23972814
4	28197366	Compensatory upregulation of PD-1, LAG-3, and ...	Tumor-associated or -infiltrating lymphocytes ...	None	Unknown	16344461

Figure 5.2: Sample of extracted scientific articles from the dataset

5.3 Data Preprocessing

In this chapter, we briefly summarize the preprocessing stage, which was detailed in Chapter 4, in order to focus on its practical role in the experimental setup.

As a first step, articles without abstracts were removed to ensure data quality and consistency. This filtering step resulted in a reduced but more reliable dataset, where each document contains sufficient textual information for embedding generation.

For the **textual view**, abstracts of scientific articles were used as the main source of information.After basic cleaning and preprocessing, the texts were encoded using the BioBERT model. Stemming and lemmatization were not applied, as BioBERT preserves contextual information more effectively without them. This choice preserves the contextual and semantic information in the text. The output is a set of dense embedding vectors representing the semantic content of each document.

For the **bibliographic view**, citation relationships were used to construct a bibliographic coupling graph. Documents sharing common references were considered related, and their similarity was computed using the Jaccard coefficient. After filtering inconsistent records, the final graph contains **4,957 nodes** representing scientific articles and **201,996 edges** representing bibliographic coupling relationships. This reflects a dense structural connectivity pattern within the dataset.

Finally, similarity matrices were built for both views. The textual similarity matrix was computed using cosine similarity between BioBERT embeddings, resulting in a dense semantic structure. In contrast, the bibliographic similarity matrix, based on Jaccard similarity, is sparser and mainly reflects shared citation behavior.

5.3.1 Similarity Distribution Analysis

Figure 5.3 illustrates the distribution of similarity values for both representations.

The textual similarity matrix exhibits a relatively balanced distribution centered around moderate similarity values, indicating the presence of meaningful semantic relationships between documents. In contrast, the bibliographic similarity matrix is highly sparse, with most similarity values concentrated near zero.

This difference highlights the complementary nature of the two views. While the textual representation captures semantic information derived from document content, the bibliographic representation captures structural information derived from citation behavior. Combining these two sources of information is expected to improve clustering quality by exploiting both semantic and structural characteristics of scientific documents.

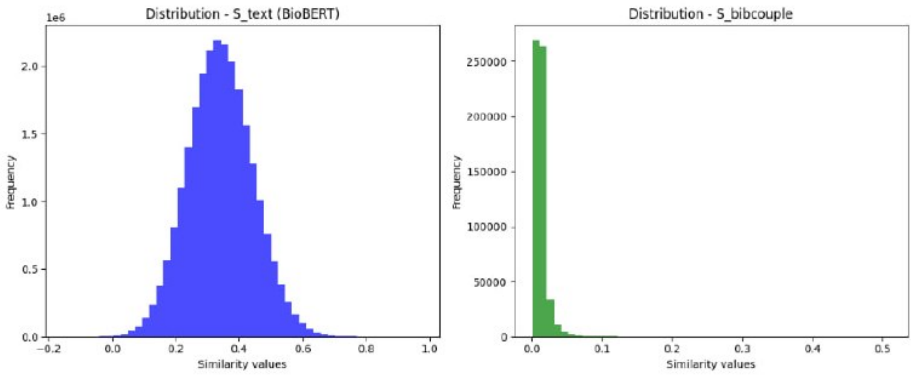

Figure 5.3: Distribution of similarity values for textual and bibliographic representations

5.4 Experimental Configurations

To evaluate the effectiveness of the proposed multi-view clustering framework, five experimental configurations were designed and compared. These configurations represent different ways of exploiting textual and bibliographic information, ranging from single-view approaches to multi-view fusion strategies.

The objective of this experimental design is to assess the contribution of each representation and to identify the most effective strategy for clustering scientific documents.

5.4.1 Configuration C1 – Text Only

The first configuration relies exclusively on the textual representation of scientific articles. In this setting, document abstracts were encoded using the BioBERT model to generate dense semantic embeddings.

Since BioBERT embeddings are high-dimensional, the UMAP dimensionality reduction technique was applied to obtain a more compact representation while preserving the local structure of the data. The resulting embeddings were then clustered using the K-Means algorithm.

This configuration serves as a textual baseline and allows us to evaluate the clustering performance obtained using semantic information alone.

Figure 5.4: Pipeline of Configuration C1 (Text Only)

5.4.2 Configuration C2 – Citation Only

The second configuration exploits only bibliographic information extracted from citation relationships.

A bibliographic coupling graph was constructed by connecting articles that share common references. Community detection was then performed using the Louvain algorithm, which automatically identifies groups of closely related documents within the citation network.

Unlike the textual configuration, this approach relies exclusively on structural information and ignores document content.

This configuration serves as a bibliographic baseline and enables the evaluation of citation-based clustering independently from textual information.

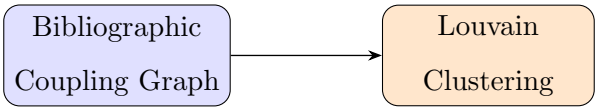

Figure 5.5: Pipeline of Configuration C2 (Citation Only)

5.4.3 Configuration C3 – Early Fusion with Spectral Embedding

The third configuration corresponds to the first multi-view approach based on Early Fusion.

The textual view was represented using BioBERT embeddings, while the bibliographic view was transformed into low-dimensional node representations using Spectral Embedding. This technique projects graph nodes into a vector space while preserving the global structure of the graph.

The two representations were concatenated at the feature level to produce a unified representation for each document. UMAP was then applied to reduce dimensionality before performing clustering with K-Means.

By integrating semantic and structural information before clustering, this approach exploits the complementary strengths of both views.

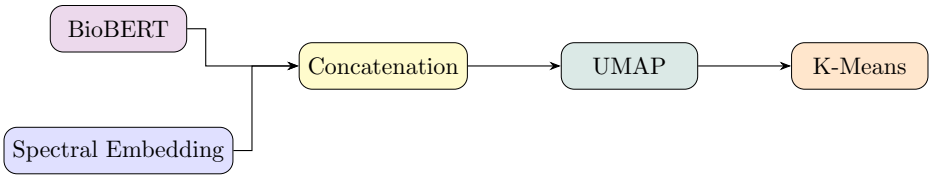

Figure 5.6: Pipeline of Configuration C3 (Early Fusion — Spectral Embedding)

5.4.4 Configuration C4 – Early Fusion with Node2Vec

This configuration follows the same Early Fusion strategy as C3, but replaces Spectral Embedding with Node2Vec for graph representation learning.

Node2Vec generates low-dimensional node embeddings by performing biased random walks on the bibliographic graph. This technique captures both local neighborhood information and global structural relationships between documents.

The generated graph embeddings are concatenated with BioBERT embeddings to obtain a combined representation. The fused vectors are then reduced using UMAP and clustered using K-Means.

This configuration allows comparison between two different graph embedding techniques within the Early Fusion framework.

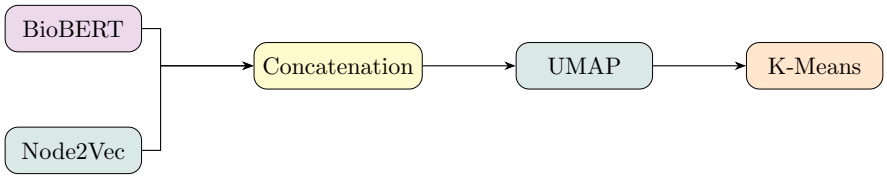

Figure 5.7: Pipeline of Configuration C4 (Early Fusion — Node2Vec)

5.4.5 Configuration C5 – Late Fusion

The fifth configuration implements a Late Fusion strategy.

Instead of combining feature vectors, this approach merges similarity information obtained independently from the textual and bibliographic views. First, a textual similarity matrix is generated using BioBERT embeddings, while a bibliographic similarity matrix is constructed from bibliographic coupling relationships. The two matrices are then combined through weighted fusion to produce a unified similarity representation. Finally, K-Means clustering is applied to the resulting fused representation.

Unlike Early Fusion, where information is integrated before clustering, Late Fusion preserves the independence of each view until the final fusion stage.

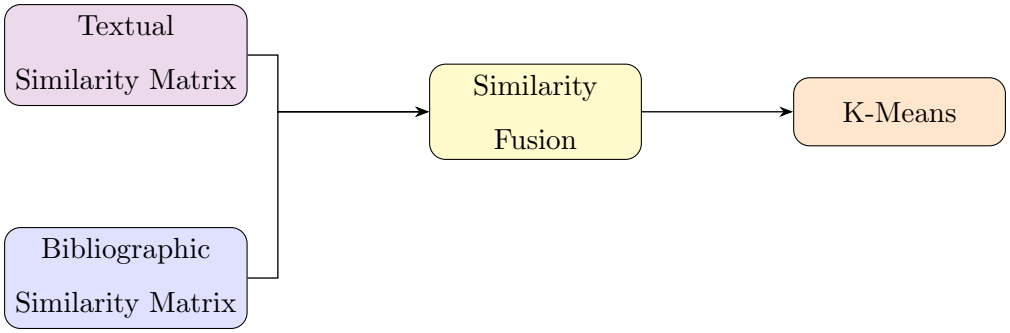

Figure 5.8: Pipeline of Configuration C5 (Late Fusion at the Similarity Matrix Level)

5.5 Experimental Environment and Tools

5.5.1 Execution Environment

All experiments in this work were conducted using **Google Colaboratory (Google Colab)** [54], a cloud-based development environment that allows the execution of Python code directly in a web browser. Google Colab provides free access to computational resources such as GPU and TPU accelerators, without requiring local installation or configuration. This makes it particularly suitable for machine learning and data-intensive experiments.

The implementation was carried out using the **Python programming language** [55], which is widely used in scientific computing, data analysis, and machine learning due to its simplicity, flexibility, and rich ecosystem of libraries.

Due to memory limitations, experiments were performed on a subset of 4,957 articles after filtering out records without abstracts.

5.5.2 Libraries and Tools

The proposed framework relies on several Python libraries that support data processing, machine learning, and graph-based modeling:

- **NumPy** [56]: A fundamental library for numerical computing in Python. It provides efficient multidimensional array structures and mathematical functions for scientific computation.
- **Pandas** [57]: An open-source Python library that provides a set of fundamental data structures and tools for statistical computing and data analysis. Its core data structure, the `DataFrame`, enables storing and manipulating structured data in a tabular format with support for mixed data types and built-in data alignment.
- **Scikit-learn** [45]: An open-source Python library that provides a wide range of machine learning algorithms for both supervised and unsupervised learning tasks. It is designed to be accessible, with an emphasis on ease of use, consistency, and performance, and is built on top of NumPy and SciPy.

- **SciPy** [58]: An open-source scientific computing library built on top of NumPy. It provides a comprehensive collection of numerical algorithms and mathematical tools, including optimization, integration, interpolation, linear algebra, and eigenvalue decomposition.

- **UMAP** [44]: A non-linear dimensionality reduction library that projects high-dimensional embeddings into a lower-dimensional space while preserving the local structure of the data.

- **Node2Vec** [46]: A graph representation learning algorithm based on biased random walks. It learns low-dimensional continuous feature representations for nodes in a graph by generating sequences of nodes through flexible random walks, capturing both local neighborhood structure and global connectivity patterns within the graph.

- **Transformers** [59]: An open-source library developed by Hugging Face that provides state-of-the-art implementations of Transformer-based architectures for natural language processing. It offers a unified API for accessing and using a wide variety of pretrained models, enabling researchers and practitioners to easily load, fine-tune, and deploy models for a broad range of natural language processing tasks.

5.6 Experimental Results

This section presents the experimental results obtained from the different clustering configurations introduced previously. The objective is to evaluate the effectiveness of the proposed representations and fusion strategies using the internal validation metrics . The experiments compare single-view approaches (Text Only and Citation Only) with multi-view approaches based on Early Fusion and Late Fusion.

5.6.1 Choice of K

We set the number of clusters to $\mathbf{K = 20}$ for all experiments. We determined this value by first applying the **Louvain** algorithm on the Bibliographic Coupling graph, which automatically identified **20 clusters** within the dataset. We then used this result as a

reference to set K for the **K-Means** algorithm in both the Early Fusion and Late Fusion pipelines, ensuring consistency across all experiments.

5.6.2 Text Only Results

The first baseline experiment was conducted using only the textual representation generated by BioBERT. The resulting embeddings were reduced using UMAP and clustered using K-Means.

Method	**Silhouette ↑**	**Calinski–Harabasz ↑**	**Davies–Bouldin ↓**
BioBERT + UMAP + K-Means	0.3547	2112.7307	1.15018

Table 5.1: Clustering evaluation results for the textual baseline configuration

5.6.3 Citation Only Results

The second baseline experiment relied exclusively on citation information. A bibliographic coupling graph was constructed and clustered using the Louvain community detection algorithm.

Method	**Silhouette ↑**	**Calinski–Harabasz ↑**	**Davies–Bouldin ↓**
Bibliographic Coupling + Louvain	0.0037	2.2541	17.4214

Table 5.2: Clustering results for the citation-only configuration (C2)

The citation-based representation produced significantly weaker results than the textual representation. The extremely low Silhouette Score and Calinski–Harabasz Index, together with the very high Davies–Bouldin Index, indicate poorly separated and highly overlapping clusters.

This behavior can be explained by the sparse nature of citation relationships in the selected dataset. Although citation information provides valuable structural knowledge, it appears insufficient when used as the sole source of information for clustering.

5.6.4 Early Fusion Results

The Early Fusion strategy combines textual and bibliographic representations at the feature level before clustering. Two graph embedding techniques were evaluated: Spectral Embedding and Node2Vec.

Method	**Silhouette ↑**	**Calinski–Harabasz ↑**	**Davies–Bouldin ↓**
Spectral Embedding	**0.5321**	**2808.55**	**0.7336**
Node2Vec	0.4150	1620.60	0.8671

Table 5.3: Clustering evaluation metrics – Early Fusion (K = 20)

Based on the evaluation metrics presented in Table 5.3, Spectral Embedding clearly outperforms Node2Vec across all evaluation criteria. It achieves a higher Silhouette Score (0.5321 vs 0.4150), indicating better cluster separation, and a significantly higher Calinski–Harabasz Index (2808.55 vs 1620.60), reflecting more compact and well-defined clusters. Additionally, it obtains a lower Davies–Bouldin Index (0.7336 vs 0.8671), suggesting less overlap and greater dissimilarity between neighboring clusters. These results consistently confirm that Spectral Embedding produces superior clustering quality in the early fusion setting with K=20

5.6.5 Late Fusion Results

The Late Fusion strategy combines the textual and bibliographic similarity matrices after independent processing of each view.

Method	**Silhouette ↑**	**Calinski–Harabasz ↑**	**Davies–Bouldin ↓**
Late Fusion	0.0132	29.3182	4.9442

Table 5.4: Clustering evaluation metrics – Late Fusion

Contrary to expectations, the Late Fusion approach produced relatively poor clustering performance. Although it combines information from both views, the resulting clusters remain weakly separated.

One possible explanation is the significant difference in the distributions of the textual and bibliographic similarity matrices. As shown previously in the similarity distribution analysis, the textual similarities exhibit a relatively dense distribution, whereas the bibliographic similarities are highly sparse. This discrepancy may reduce the effectiveness of matrix-level fusion and introduce additional noise into the clustering process.

5.6.6 Overall Comparison

Method	**Silhouette ↑**	**Calinski–Harabasz ↑**	**Davies–Bouldin ↓**
Citation Only	0.0037	2.2541	17.4214
Late Fusion	0.0132	29.3182	4.9442
Text Only (BioBERT + UMAP + K-Means)	0.3547	2112.7307	1.15018
Early Fusion (Node2Vec)	0.4150	1620.60	0.8671
Early Fusion (Spectral Embedding)	**0.5321**	**2808.55**	**0.7336**

Table 5.5: Comparison of clustering performance across all methods

Based on the evaluation results, we found that **Early Fusion with Spectral Embedding** achieved the best overall clustering performance, with the highest Silhouette Score and the lowest Davies–Bouldin Index among all tested strategies. These results confirm that combining semantic and structural representations at the feature level, using eigen-decomposition-based embeddings, produces more compact and well-separated clusters.

5.6.7 UMAP Visualization

To provide a qualitative interpretation of the clustering results, UMAP was used to project the high-dimensional document representations into a two-dimensional space.

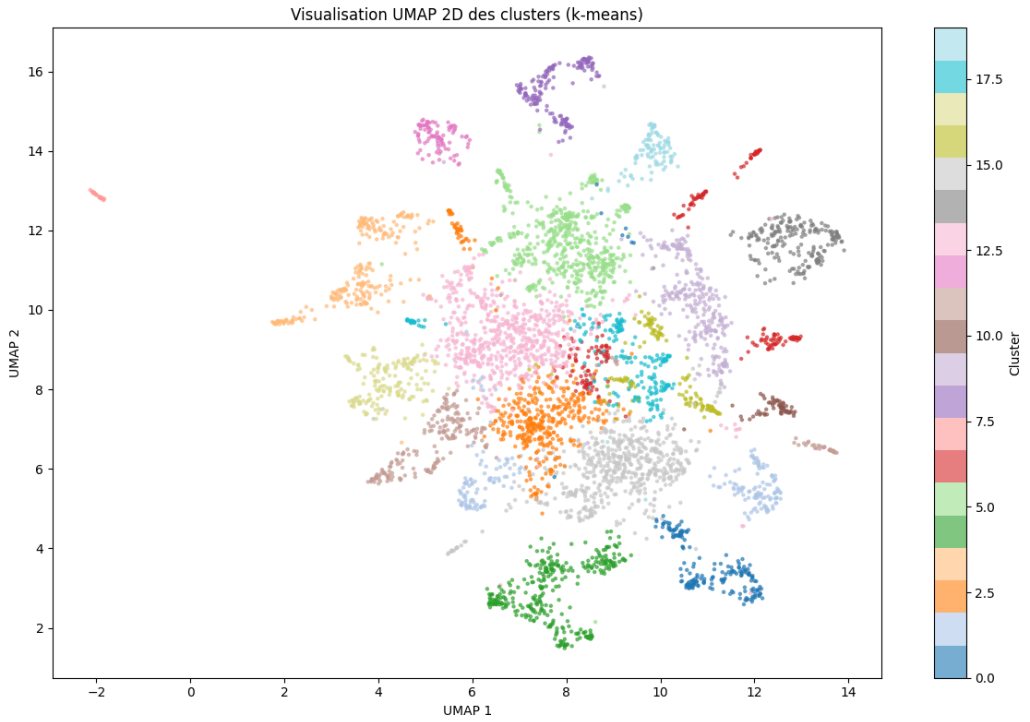

Figure 5.9: 2D cluster visualization – Early Fusion (Spectral Embedding)

Figure 5.9 presents the 2D UMAP visualization for the best-performing configuration, namely Early Fusion with Spectral Embedding (C3). The projection reveals the emergence of relatively well-defined clusters with clear spatial separation, particularly at the periphery, alongside a denser central region where thematic boundaries are less distinct. Compared to the single-view baselines — text-only (C1) and citation-only (C2) — the clusters appear more compact and better organized, which is consistent with the quantitative evaluation results.

These findings suggest that combining the semantic information extracted by BioBERT with the relational structure derived from bibliographic coupling yields a richer and more expressive document representation. Specifically, the textual modality captures thematic similarity based on abstract content, while the spectral embedding encodes inter-document relationships through shared reference proximity. Together, these two complementary sources of information produce a joint representation that provides a more effective and informative basis for clustering biomedical literature than either modality considered in isolation.

5.7 Discussion

5.7.1 Multi-View vs. Single-View

The results demonstrate that multi-view clustering consistently outperforms single-view approaches. Although the BioBERT-based textual representation achieves satisfactory performance, the citation-based representation alone yields limited results due to the sparsity of citation links between documents. By combining both views, the framework is able to leverage complementary semantic and structural information, leading to more coherent and better-separated clusters.

5.7.2 Early Fusion vs. Late Fusion

Among the evaluated fusion strategies, Early Fusion significantly outperforms Late Fusion. By integrating textual and bibliographic information prior to clustering, the model learns a more expressive and discriminative representation of documents, resulting in more compact cluster structures. In contrast, Late Fusion is less effective, as it fails to fully capture the interactions between heterogeneous similarity spaces derived from textual and citation data.

5.7.3 Spectral Embedding vs. Node2Vec

The comparison of graph embedding methods shows that Spectral Embedding consistently outperforms Node2Vec across all evaluation metrics. This result suggests that preserving the global structure of the citation graph is more beneficial for clustering than relying on local neighborhood information derived from random walks.

5.7.4 Limitations

This study has several limitations. First, experiments were conducted on a subset of 4,957 articles after filtering out records without abstracts, which may limit the generalizability of the results. Second, the evaluation relies exclusively on internal clustering metrics, as ground-truth labels are not available. Finally, the sparsity of the citation graph negatively impacts methods that depend on structural information.

5.7.5 Overall Conclusion

Overall, the results confirm that Early Fusion combined with Spectral Embedding provides the most effective configuration for scientific document clustering by jointly exploiting semantic and bibliographic information.

5.8 Conclusion

In this chapter, we presented the experimental evaluation of the proposed multi-view clustering framework. After describing the dataset, preprocessing steps, experimental environment, and evaluation metrics, several clustering configurations were compared.

The results showed that the textual representation based on BioBERT outperformed the citation-based representation when used independently. Furthermore, combining textual and bibliographic information improved clustering quality. Among all evaluated configurations, Early Fusion with Spectral Embedding achieved the best performance according to all evaluation metrics.

Overall, these findings confirm the effectiveness of integrating semantic and bibliographic information for scientific document clustering and demonstrate the advantages of multi-view approaches over single-view methods.

General Conclusion

In this thesis, we addressed the problem of scientific document clustering, which represents a major challenge in the current context characterized by the exponential growth of digital data. The main objective of this work was to explore and exploit approaches aimed at improving the quality of document grouping, while taking into account the richness and complexity of this type of data.

In the first phase, we studied the theoretical foundations of document clustering by analyzing different representation methods, similarity measures, and the main clustering algorithms. This study enabled us to identify the limitations of classical approaches, particularly those based on a single representation.

To overcome these limitations, we adopted an approach based on Multi-View Clustering, which allows the combination of multiple complementary sources of information. In this context, we proposed a methodology combining a textual view based on BioBERT and a bibliographic view based on Bibliographic Coupling, integrated through Early Fusion and Late Fusion strategies.

On the practical side, we implemented a complete processing pipeline including preprocessing, representation, dimensionality reduction, clustering, and evaluation. This process allowed us to analyze the impact of different methodological choices on clustering quality.

The obtained results show that integrating multiple views improves the coherence and relevance of clusters compared to single-view approaches. The use of appropriate techniques also contributed to better data structuring and more reliable extraction of document groups.

Finally, this work opens several promising research perspectives. It would be relevant to explore other fusion strategies, integrate additional information sources, or apply more

advanced models to further improve clustering performance.

In conclusion, this thesis highlights the importance of Multi-View Clustering as an effective approach for processing scientific documents, enabling better exploitation of the richness of information and improving the quality of clustering results.

Bibliography

[1] C.C. Aggarwal and C.X. Zhai. *Mining Text Data*. Springer New York, 2012. ISBN: 9781461432234. URL: https://books.google.dz/books?id=vFHOx8wfSUOC.

[2] Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. *Introduction to Information Retrieval*. Cambridge University Press, 2008.

[3] Scott Deerwester et al. "Indexing by latent semantic analysis". In: *Journal of the American Society for Information Science* 41.6 (1990), pp. 391–407.

[4] David M Blei, Andrew Y Ng, and Michael I Jordan. "Latent dirichlet allocation". In: *Journal of machine Learning research* 3.Jan (2003), pp. 993–1022.

[5] Tomas Mikolov et al. "Efficient estimation of word representations in vector space". In: *arXiv preprint arXiv:1301.3781* (2013).

[6] Jeffrey Pennington, Richard Socher, and Christopher D Manning. "Glove: Global vectors for word representation". In: *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*. 2014, pp. 1532–1543.

[7] Quoc Le and Tomas Mikolov. "Distributed representations of sentences and documents". In: *International conference on machine learning*. PMLR. 2014, pp. 1188–1196.

[8] Jacob Devlin et al. "Bert: Pre-training of deep bidirectional transformers for language understanding". In: *Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers)*. 2019, pp. 4171–4186.

[9] Alec Radford et al. "Improving language understanding by generative pre-training". In: (2018).

[10] D. Jurafsky and J.H. Martin. *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition*. Prentice Hall series in artificial intelligence. Pearson Prentice Hall, 2009. ISBN: 9780131873216. URL: https://books.google.dz/books?id=fZmj5UNK8AQC.

[11] J. Han, M. Kamber, and J. Pei. *Data Mining: Concepts and Techniques*. The Morgan Kaufmann Series in Data Management Systems. Morgan Kaufmann, 2011. ISBN: 9780123814807. URL: https://books.google.dz/books?id=pQws07tdpjoC.

[12] Andrew Ng, Michael Jordan, and Yair Weiss. "On spectral clustering: Analysis and an algorithm". In: *Advances in neural information processing systems* 14 (2001).

[13] Martin Ester et al. "A density-based algorithm for discovering clusters in large spatial databases with noise". In: *kdd*. Vol. 96. 34. 1996, pp. 226–231.

[14] Majid Hameed Ahmed et al. "Short text clustering algorithms, application and challenges: a survey". In: *Applied Sciences* 13.1 (2022), p. 342.

[15] Robert A. Day and Barbara Gastel. *How to Write and Publish a Scientific Paper*. 8th ed. Greenwood, 2016.

[16] Mark Newman. *Networks: An Introduction*. Oxford University Press, 2010.

[17] Gerard Salton, Anita Wong, and Chung-Shu Yang. "A vector space model for automatic indexing". In: *Communications of the ACM* 18.11 (1975), pp. 613–620.

[18] Sang-Woon Kim and Joon-Min Gil. "Research paper classification systems based on TF-IDF and LDA schemes". In: *Human-centric Computing and Information Sciences* 9.1 (2019), p. 30.

[19] Choonghyun Han and Jinwook Choi. "Effect of latent semantic indexing for clustering clinical documents". In: *2010 IEEE/ACIS 9th International Conference on Computer and Information Science*. IEEE. 2010, pp. 561–566.

[20] Chyi-Kwei Yau et al. "Clustering scientific documents with topic modeling". In: *Scientometrics* 100.3 (2014), pp. 767–786.

[21] Tomasz Walkowiak and Mateusz Gniewkowski. "Evaluation of vector embedding models in clustering of text documents". In: *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP 2019)*. 2019, pp. 1304–1311.

[22] MJP Canon, LL Maceda, and NM Flores. "Contextual understanding of academic-related responses based on enhanced word embeddings, clustering, and community detection". In: *Journal of Advances in Information Technology* 15.6 (2024), pp. 693–703.

[23] Qufei Chen and Marina Sokolova. "Specialists, scientists, and sentiments: Word2Vec and Doc2Vec in analysis of scientific and medical texts". In: *SN Computer Science* 2.5 (2021), p. 414.

[24] Ryoma Hosokawa et al. "Reference Classification Using BERT Models to Support Scientific-Document Writing". In: *JSAI International Symposium on Artificial Intelligence*. Springer. 2023, pp. 167–183.

[25] Iz Beltagy, Kyle Lo, and Arman Cohan. "SciBERT: A pretrained language model for scientific text". In: *Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (EMNLP-IJCNLP)*. 2019, pp. 3615–3620.

[26] Henry Small. "Co-citation in the scientific literature: A new measure of the relationship between two documents". In: *Journal of the American Society for information Science* 24.4 (1973), pp. 265–269.

[27] Maxwell Mirton Kessler. "Bibliographic coupling between scientific papers". In: *American documentation* 14.1 (1963), pp. 10–25.

[28] Juan Pablo Bascur et al. "Academic information retrieval using citation clusters: in-depth evaluation based on systematic reviews". In: *Scientometrics* 128.5 (2023), pp. 2895–2921.

[29] Lovro Šubelj, Nees Jan Van Eck, and Ludo Waltman. "Clustering scientific publications based on citation relations: A systematic comparison of different methods". In: *PloS one* 11.4 (2016), e0154404.

[30] Vincent A Traag, Ludo Waltman, and Nees Jan Van Eck. "From Louvain to Leiden: guaranteeing well-connected communities". In: *Scientific reports* 9.1 (2019), p. 5233.

[31] Dejian Yu et al. "Hybrid Self-Optimized Clustering Model Based on Citation Links and Textual Features to Detect Research Topics". In: *PLOS ONE* 12.10 (2017), e0187164. DOI: 10.1371/journal.pone.0187164.

[32] Shuai Zhang, Yangbing Xu, and Wenyu Zhang. "Clustering scientific document based on an extended citation model". In: *IEEE Access* 7 (2019), pp. 57037–57046.

[33] Vu Thi Huong and Thorsten Koch. "Clustering scientific publications: lessons learned through experiments with a real citation network". In: *arXiv preprint arXiv:2505.18180* (2025).

[34] Steffen Bickel and Tobias Scheffer. "Multi-View Clustering". In: *Proceedings of the Fourth IEEE International Conference on Data Mining (ICDM)*. IEEE, 2004, pp. 19–26. DOI: 10.1109/ICDM.2004.10095.

[35] Yan Yang and Hao Wang. "Multi-view Clustering: A Survey". In: *Big Data Mining and Analytics* 1 (June 2018), pp. 83–107. DOI: 10.26599/BDMA.2018.9020003.

[36] Abhishek Kumar and Hal Daumé. "A co-training approach for multi-view spectral clustering". In: *Proceedings of the 28th international conference on machine learning (ICML-11)*. 2011, pp. 393–400.

[37] Nimrod Rappoport and Ron Shamir. "Multi-omic and multi-view clustering algorithms: review and cancer benchmark". In: *Nucleic acids research* 46.20 (2018), pp. 10546–10562.

[38] Ruina Bai et al. "Deep multi-view document clustering with enhanced semantic embedding". In: *Information Sciences* 564 (2021), pp. 273–287. DOI: 10.1016/j.ins.2021.02.027. URL: https://doi.org/10.1016/j.ins.2021.02.027.

[39] Yijian Fu. "A Review of Deep Multi-View Clustering Methods". In: *Journal of Theory and Practice in Engineering and Technology* 2.2 (2025), pp. 1–5.

[40] Jie Chen et al. "Deep Multiview Clustering by Contrasting Cluster Assignments". In: *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. 2023, pp. 16752–16761. DOI: 10.1109/ICCV51070.2023.01536. URL: https://doi.org/10.1109/ICCV51070.2023.01536.

[41] Abdelmalik Moujahid and Fadi Dornaika. "Advanced unsupervised learning: a comprehensive overview of multi-view clustering techniques". In: *Artificial Intelligence Review* 58.8 (2025), p. 234.

[42] Fanghua Ye et al. "New Approaches in Multi-View Clustering". In: Aug. 2018. ISBN: 978-1-78923-526-5. DOI: 10.5772/intechopen.75598.

[43] Jinhyuk Lee et al. "BioBERT: a pre-trained biomedical language representation model for biomedical text mining". In: *Bioinformatics* 36.4 (2020), pp. 1234–1240.

[44] Leland McInnes, John Healy, and James Melville. "Umap: Uniform manifold approximation and projection for dimension reduction". In: *arXiv preprint arXiv:1802.03426* (2018).

[45] Fabian Pedregosa et al. "Scikit-learn: Machine learning in Python". In: *the Journal of machine Learning research* 12 (2011), pp. 2825–2830.

[46] Aditya Grover and Jure Leskovec. "node2vec: Scalable feature learning for networks". In: *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*. 2016, pp. 855–864.

[47] Vincent D Blondel et al. "Fast unfolding of communities in large networks". In: *Journal of statistical mechanics: theory and experiment* 2008.10 (2008), P10008.

[48] Wenyuan Liu et al. "Predicting the evolution of physics research from a complex network perspective". In: *Entropy* 21.12 (2019), p. 1152.

[49] Peter J Rousseeuw. "Silhouettes: a graphical aid to the interpretation and validation of cluster analysis". In: *Journal of computational and applied mathematics* 20 (1987), pp. 53–65.

[50] David L Davies and Donald W Bouldin. "A cluster separation measure". In: *IEEE transactions on pattern analysis and machine intelligence* 2 (1979), pp. 224–227.

[51] Tadeusz Caliński and Jerzy Harabasz. "A dendrite method for cluster analysis". In: *Communications in Statistics-theory and Methods* 3.1 (1974), pp. 1–27.

[52] National Center for Biotechnology Information. *PubMed*. https://pubmed.ncbi.nlm.nih.gov/. Accessed: 2026-05-20.

[53] National Center for Biotechnology Information. *PubMed Central (PMC)*. https://www.ncbi.nlm.nih.gov/pmc/. Accessed: 2026-05-20.

[54] Ekaba Bisong. "Google colaboratory". In: *Building machine learning and deep learning models on google cloud platform: a comprehensive guide for beginners*. Springer, 2019, pp. 59–64.

[55] G. Van Rossum and F.L. Drake. *Python 3 Reference Manual: (Python Documentation Manual Part 2)*. Documentation for Python. CreateSpace Independent Publishing Platform, 2009. ISBN: 9781441412690. URL: https://books.google.dz/books?id=KIybQQAACAAJ.

[56] Charles R Harris et al. "Array programming with NumPy". In: *nature* 585.7825 (2020), pp. 357–362.

[57] Wes McKinney et al. "Data structures for statistical computing in Python." In: *scipy* 445.1 (2010), pp. 51–56.

[58] Pauli Virtanen et al. "SciPy 1.0: fundamental algorithms for scientific computing in Python". In: *Nature methods* 17.3 (2020), pp. 261–272.

[59] Thomas Wolf et al. "Transformers: State-of-the-art natural language processing". In: *Proceedings of the 2020 conference on empirical methods in natural language processing: system demonstrations*. 2020, pp. 38–45.

Annexe