Clustering

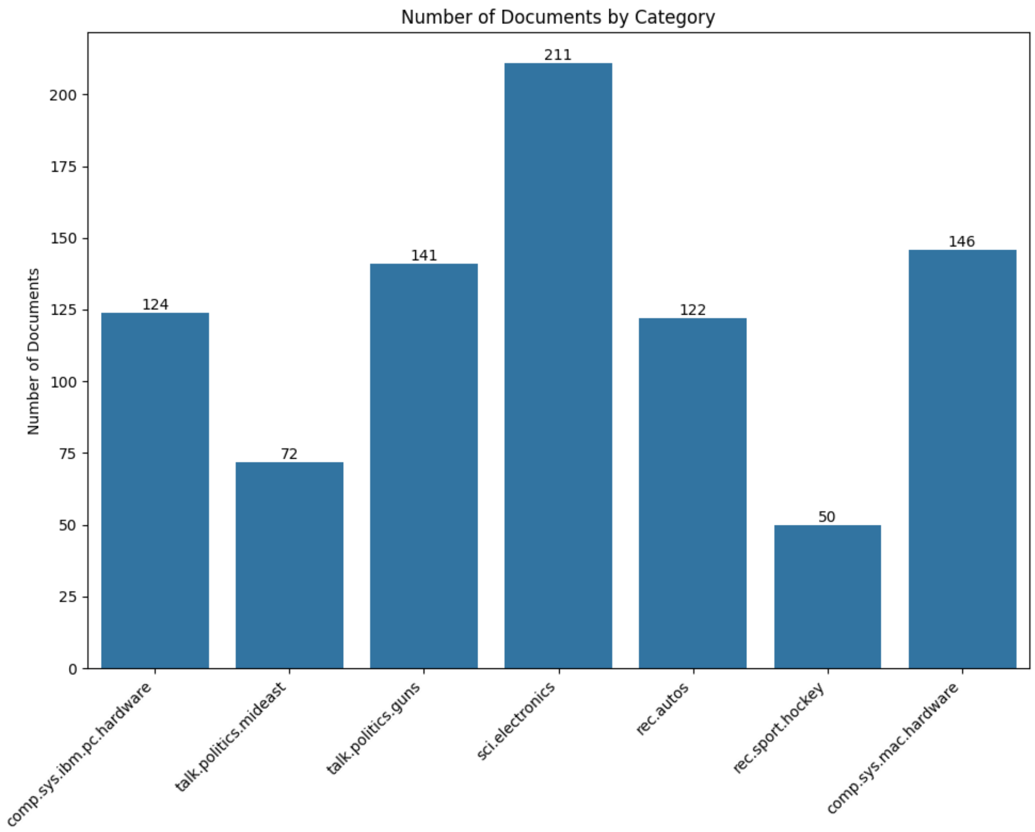
Objective: Apply clustering algorithms to text representations obtained from a subset of the 20 Newsgroup collection, evaluate the clustering results using external evaluation measures, and analyze the quality and characteristics of the clusters formed.

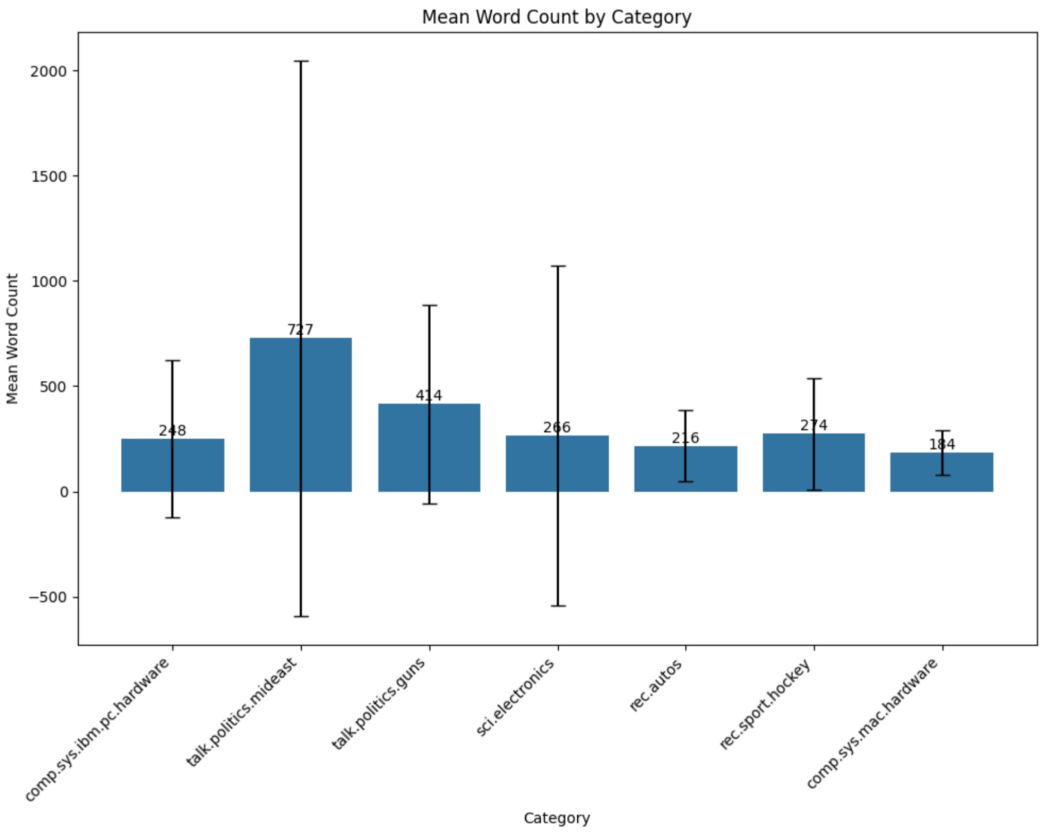
This research is publicly available at https://github.com/alvaro-francisco-gil/text-mining/tree/main/04_clustering

Collection Description

The analysis focused on a collection of text documents sourced from various newsgroups, each representing a distinct topic or category. A key initial step involved characterizing this collection by quantifying the distribution of documents across the different groups. Furthermore, the linguistic characteristics of the documents within each group were examined by calculating the average number of words per document and the standard deviation of word counts.

There is considerable variation in both the average length and the variability of document length across categories. For example, *talk.politics.mideast* has significantly longer posts on average and also the highest variability, while *comp.sys.mac.hardware* has the shortest average posts and lower variability.


Number of Documents by Category
Number of Documents
0
25
50
75
100
125
150
175
200
124
72
141
211
122
50
146
comp.sys.ibm.pc.hardware
talk.politics.mideast
talk.politics.guns
sci.electronics
rec.autos
rec.sport.hockey
comp.sys.mac.hardware


Mean Word Count by Category
Mean Word Count
−500
0
500
1000
1500
2000
248
727
414
266
216
274
184
comp.sys.ibm.pc.hardware
talk.politics.mideast
talk.politics.guns
sci.electronics
rec.autos
rec.sport.hockey
comp.sys.mac.hardware
Category

Gold Standard Formation

To enable external evaluation of the clustering results, a "gold standard" or ground truth was established. This gold standard leverages the inherent structure of the chosen dataset (a subset of the 20 Newsgroups collection), where each document is pre-assigned to a specific newsgroup category. These pre-existing labels, directly associating each document file or data entry with its corresponding group name (e.g., sci.electronics, rec.autos), serve as the definitive ground truth, allowing for objective measurement of the clustering algorithm's ability to rediscover these known groupings.

Preprocessing Phase

In the preprocessing phase, the text data underwent several transformations to enhance its quality and suitability for analysis. Initially, headers and extraneous metadata were stripped from the documents, focusing on the main content by removing lines before the first blank line, which typically contains email headers. To further clean the text, lines with email addresses and common signature patterns were identified and removed. Punctuation was eliminated using a translation table, and numbers were optionally removed to reduce noise. The text was then converted to lowercase to ensure uniformity. Tokenization was performed to split the text into individual words, followed by the removal of common English stop-words using the NLTK library. Finally, the text underwent optional word reduction through either lemmatization, which reduces words to their base form, or stemming, which reduces words to their root form. These preprocessing steps collectively prepared the text data for effective vectorization and subsequent analysis.

Text Representations

In the text representation phase, various methods were employed to convert the preprocessed text data into numerical formats suitable for analysis. The Term Frequency-Inverse Document Frequency (TF-IDF) representation was utilized to capture the importance of words within documents relative to the entire corpus, providing a weighted vector for each document. This method highlights terms that are frequent in a document but rare across the dataset, enhancing the ability to distinguish between topics. Additionally, word embeddings were leveraged using the Word2Vec model, specifically the pre-trained "word2vec-google-news-300" embeddings. Two types of vector representations were derived from these embeddings: the average vector, which computes the mean of the word vectors in a document, and the sum vector, which aggregates the word vectors. These representations encapsulate semantic information and relationships between words, offering a rich and nuanced understanding of the text data. Together, these methods provide a comprehensive framework for analyzing and clustering the text documents.

Clustering Algorithms

To thoroughly explore the underlying structure within the dataset, we employed a selection of four distinct clustering algorithms, representing different methodological families. These include the widely-used partitional method K-means, the bottom-up Agglomerative hierarchical clustering, the graph-based Spectral Clustering capable of handling complex shapes, and the hybrid Bisecting K-means approach. By applying these diverse techniques across various data representations, we aim to gain a comprehensive understanding of the potential groupings within the text data and identify which methodologies are most effective for this specific task.

1. K-means Clustering

K-means is a foundational *partitional* clustering algorithm. It aims to divide the dataset into a pre-defined number k*k* of distinct, non-overlapping clusters. The algorithm works iteratively by assigning each data point to the cluster whose mean (centroid) is nearest, and then recalculating the centroid for each cluster based on the points assigned to it. This process continues until the cluster assignments no longer change significantly. K-means was chosen because it is computationally efficient, easy to understand, and serves as a strong baseline for comparison. It performs well when clusters are roughly spherical and equally sized.

2. Agglomerative Clustering

Agglomerative Clustering belongs to the family of *hierarchical* clustering methods. It follows a bottom-up approach, starting with each data point as an individual cluster. In successive steps, it merges the pair of clusters that are "closest" according to a chosen linkage criterion (e.g., Ward, average, complete) until only one cluster remains or a specified number of clusters is reached. This method was included because it does not require the number of clusters to be specified beforehand (though we are using a fixed number here) and can reveal nested structures or hierarchies within the data. The resulting dendrogram can also offer insights into the data's structure at different levels of granularity.

3. Spectral Clustering

Spectral Clustering is a *graph-based* technique that leverages the properties of the data's similarity graph. It transforms the data into a lower-dimensional space using the eigenvectors (the "spectrum") of the graph Laplacian matrix. Standard clustering methods, often K-means, are then applied in this transformed space. Spectral clustering was chosen because it is particularly effective at identifying non-convex cluster shapes and can perform well even when clusters are connected by thin structures, which often pose challenges for algorithms like K-means that rely on distance in the original feature space. It excels when the underlying structure resembles a manifold.

4. Bisecting K-means

Bisecting K-means can be seen as a hybrid approach, combining elements of *partitional* and *divisive hierarchical* clustering. It starts with all data points in a single cluster and iteratively selects a cluster to split into two using the basic K-means algorithm. This process is repeated until the desired number k*k* of clusters is obtained. Typically, the cluster with the largest variance or size is chosen for splitting at each step. It was included as an alternative to standard K-means, as it can sometimes produce better results or be less sensitive to the initial placement of centroids by focusing the partitioning effort iteratively.

Evaluation Metrics

In clustering analysis, evaluating the quality of the results is crucial to understanding how well the algorithm has captured the underlying structure of the data. Unlike supervised learning, clustering lacks explicit ground truth labels, making evaluation more challenging. To address this, we employ a combination of external validation metrics that compare the clustering results to known labels. These metrics provide insights into the accuracy, consistency, and representativeness of the clusters, ensuring a comprehensive assessment of clustering performance across different algorithms and data representations.

1. Adjusted Rand Index (ARI)

The Adjusted Rand Index (ARI) is an external validation metric that quantifies the agreement between the clustering results and the true labels, adjusted for chance. It evaluates all pairs of samples and determines whether they are consistently assigned to the same or different clusters in both the predicted and true labels. The ARI ranges from -1 to 1, where 1 indicates perfect agreement, 0 suggests random labeling, and negative values imply less agreement than expected by chance. This metric is particularly valuable because it is invariant to permutations of cluster labels, providing a robust measure of clustering accuracy relative to the ground truth.

2. B-cubed Precision, Recall, and F1 Score

B-cubed metrics offer a detailed evaluation of clustering performance by considering each data point's membership in clusters. B-cubed precision measures the proportion of correctly assigned points within a cluster, while recall assesses the proportion of correctly identified points for each true label. The B-cubed F1 score, the harmonic mean of precision and recall, provides a balanced view of clustering quality. These metrics are especially useful in scenarios with overlapping clusters or varying cluster sizes, as they account for the distribution of points across clusters and true labels.

3. Homogeneity, Completeness, and V-measure

Homogeneity, completeness, and V-measure are external metrics that assess the purity and coverage of clusters. Homogeneity measures whether each cluster contains only members of a single class, ensuring that clusters are internally consistent. Completeness evaluates whether all members of a given class are assigned to the same cluster, reflecting the extent to which clusters capture the entire class. V-measure, the harmonic mean of homogeneity and completeness, provides a balanced evaluation of clustering quality. These metrics help identify whether clusters are well-defined and representative of the true class structure.

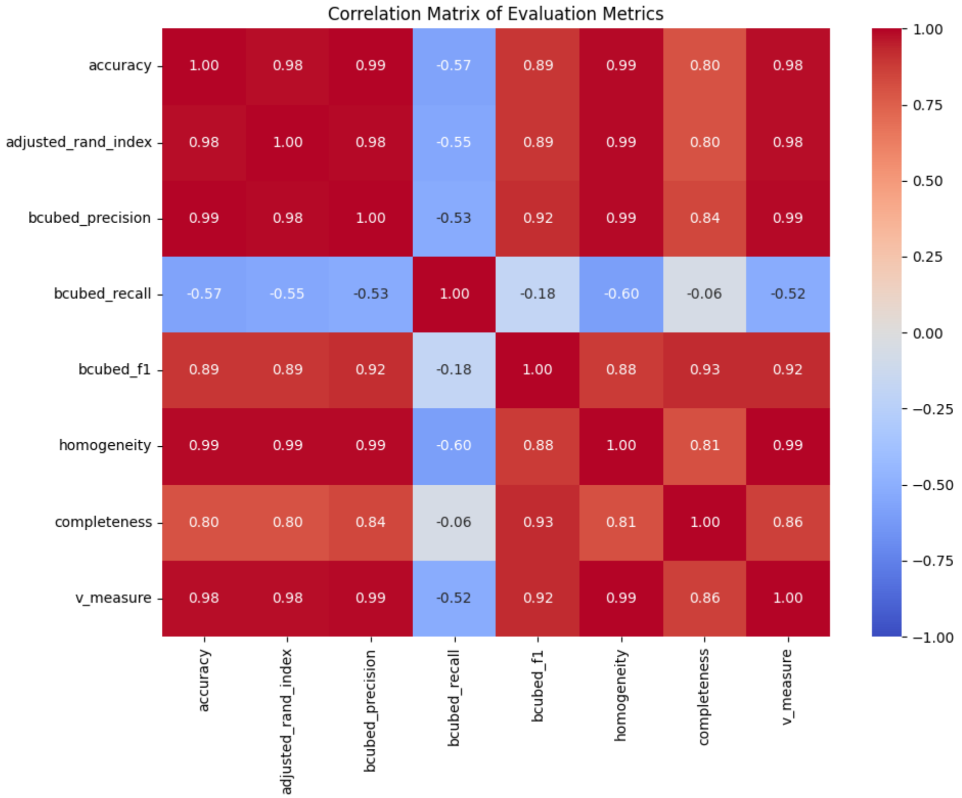
4. Accuracy After Reassignment

To calculate accuracy when cluster labels don't automatically match true categories, we first map each predicted cluster to the true category that appears most frequently among the points within that cluster. This creates a correspondence, allowing us to effectively relabel the algorithm's cluster assignments using the names of the true categories. Once predictions are relabeled according to this best-fit mapping, standard accuracy is calculated by finding the percentage of data points where the relabeled predicted category matches the actual true category.

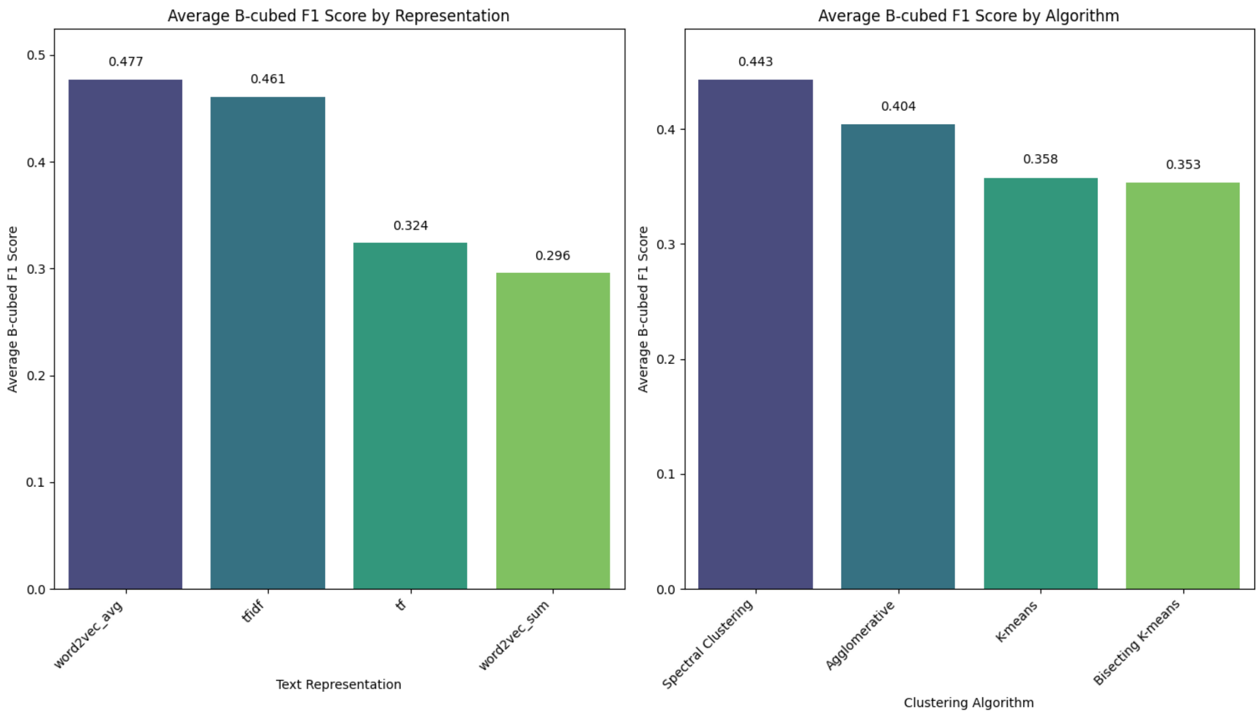
Results

Representation	Algorithm	Accuracy	ARI	Bcubed Precision	Bcubed Recall	Bcubed F1	Homogeneity	Completeness	V-measure
word2vec_avg	Spectral Clustering	**0.62**	**0.36**	**0.52**	0.66	**0.58**	**0.49**	0.58	**0.53**
tfidf	Agglomerative	0.53	0.25	0.42	0.74	0.53	0.38	**0.59**	0.46
tfidf	Spectral Clustering	0.50	0.30	0.43	0.67	0.53	0.43	0.58	0.49
word2vec_avg	Agglomerative	0.51	0.28	0.41	0.56	0.47	0.40	0.48	0.43
word2vec_avg	K-means	0.51	0.28	0.39	0.53	0.45	0.36	0.43	0.39
word2vec_avg	Bisecting K-means	0.48	0.25	0.39	0.44	0.41	0.37	0.39	0.38
tfidf	Bisecting K-means	0.52	0.24	0.39	0.41	0.40	0.36	0.36	0.36
tfidf	K-means	0.50	0.21	0.37	0.39	0.38	0.32	0.33	0.33
tf	Spectral Clustering	0.34	0.03	0.25	0.79	0.38	0.12	0.37	0.18
tf	Agglomerative	0.26	0.01	0.18	**0.94**	0.31	0.03	0.31	0.06
tf	K-means	0.26	0.01	0.18	0.94	0.31	0.03	0.32	0.05
tf	Bisecting K-means	0.26	0.01	0.18	0.94	0.31	0.03	0.29	0.05
word2vec_sum	Agglomerative	0.29	0.03	0.19	0.73	0.31	0.05	0.16	0.07

Representation	Algorithm	Accuracy	ARI	Bcubed Precision	Bcubed Recall	Bcubed F1	Homogeneity	Completeness	V-measure
word2vec_sum	K-means	0.28	0.02	0.19	0.68	0.29	0.04	0.12	0.06
word2vec_sum	Bisecting K-means	0.28	0.02	0.19	0.66	0.29	0.04	0.11	0.06
word2vec_sum	Spectral Clustering	0.38	0.12	0.29	0.30	0.29	0.24	0.25	0.25

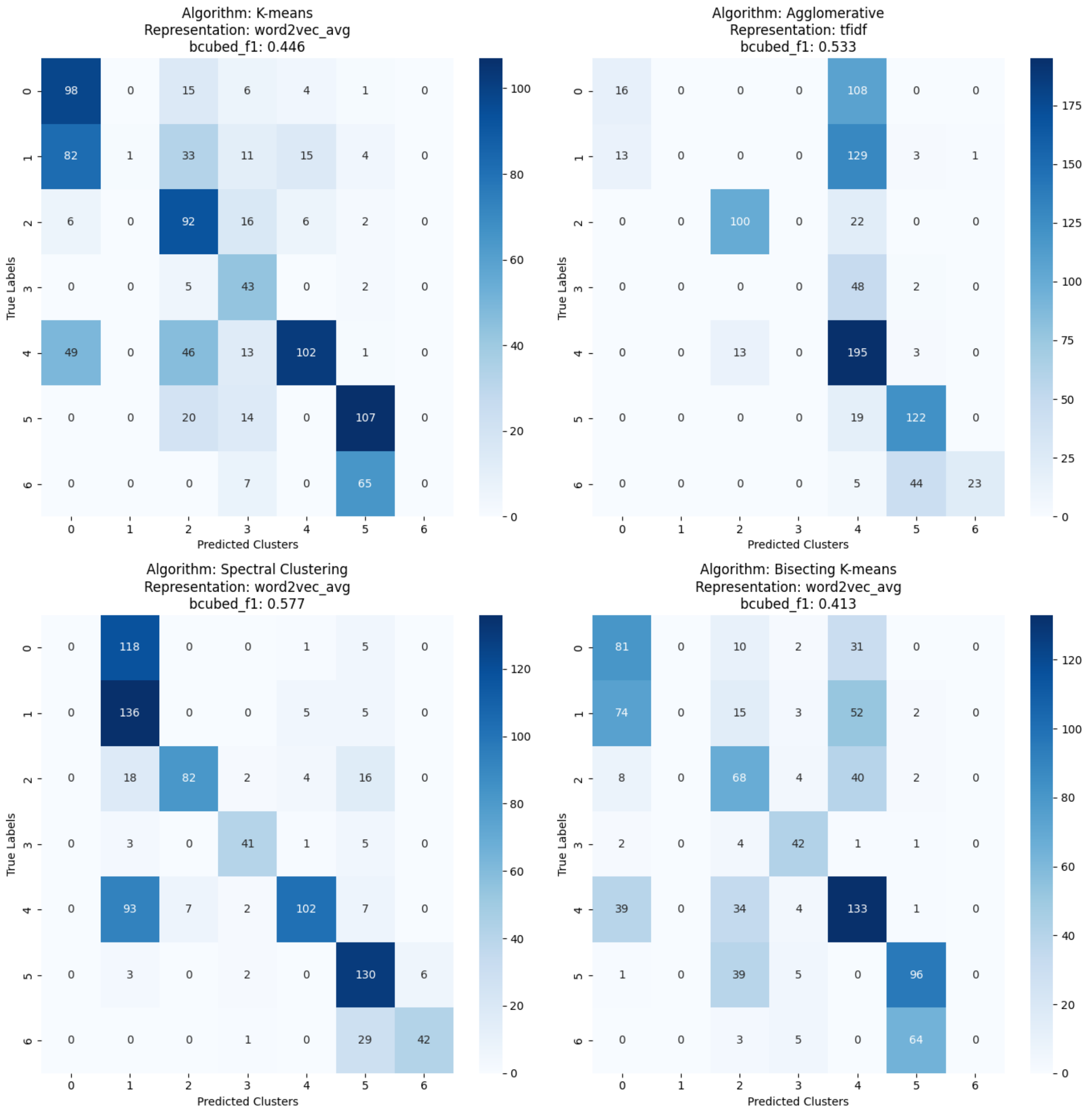


The results confirm that all evaluation metrics are strongly correlated, as expected, meaning that when good clustering occurs, it is reflected consistently across metrics such as accuracy, Adjusted Rand Index (ARI), Bcubed Precision, Recall, and F1. The best clustering performance was achieved using the Word2Vec Average representation combined with Spectral Clustering, which resulted in the highest accuracy of 0.62 and a Bcubed F1 score of 0.58, along with strong ARI (0.36) and V-measure (0.53). These results highlight the effectiveness of Word2Vec Average in capturing semantic relationships between words and Spectral Clustering's ability to identify complex patterns in high-dimensional data.



The average B-cubed F1 scores provide valuable insights into the effectiveness of different text representations and clustering algorithms in capturing the structure of the dataset. Among the representations, Word2Vec Average achieved the highest average B-cubed F1 score of 0.4766, indicating its strong ability to preserve semantic relationships and produce coherent clusters. This was closely followed by TF-IDF, which scored 0.4609, demonstrating its effectiveness in identifying important terms despite lacking semantic depth. On the other hand, simpler representations like TF and Word2Vec Sum performed significantly worse, with average B-cubed F1 scores of 0.3239 and 0.2957, respectively, highlighting their limitations in capturing meaningful patterns for clustering tasks.

When analyzing clustering algorithms, Spectral Clustering emerged as the most effective, achieving an average B-cubed F1 score of 0.4427, which aligns with its ability to handle complex data structures and relationships. Agglomerative Clustering followed with a score of 0.4037, benefiting from its hierarchical approach to grouping similar data points. However, centroid-based algorithms like K-means and Bisecting K-means showed lower performance, with scores of 0.3577 and 0.3530, respectively, reflecting their challenges in handling high-dimensional text data effectively.



Finally, we present the confusion matrices corresponding to the best results (according to the B-cubed F1 score) for each clustering algorithm. Overall, the results are promising; however, we consistently observe that around two clusters tend to be overpopulated. This suggests that the algorithm struggled to properly distinguish between certain classes. This issue may also stem from the dataset itself, as some classes are not clearly separable—for instance, two of them focus on political topics, which could lead to overlap.

Conclusions

In general, these techniques have demonstrated the ability to cluster news articles by topic in an unsupervised manner, without any prior knowledge. Accuracy, which is the easiest metric for humans to interpret, reached a promising value of 0.62 in the best case. Although there are alternative ways to calculate accuracy—such as using the Hungarian algorithm—we believe all of them would yield highly correlated results. Therefore, the majority class voting approach we used remains representative and appropriate.

It is also interesting to observe the discrepancies between the average and the sum of Word2Vec vectors. The average proved to be the best-performing representation, while the sum yielded the worst results. This difference could be attributed to the fact that summing vectors may amplify the influence of frequent or common words, thereby distorting the semantic meaning of the sentence or document, whereas averaging normalizes this effect and better captures the overall semantic context.

Regarding the clustering algorithms, despite their differing approaches, they all achieved relatively similar performance, with only about a 0.1 difference in B-cubed F1 scores between the best and the worst. This suggests that all algorithms are generally capable of identifying meaningful groupings, and that the choice of representation technique plays a more critical role in clustering performance. Nonetheless, Spectral Clustering tends to perform slightly better overall, likely due to its ability to model complex relationships and its relatively recent development compared to more traditional methods.