Clustering Textual Data
Literature Review, PoC, Winter 2024

Salveen Singh Dutt
01166213@pw.edu.pl

Karina Tiurina
01191379@pw.edu.pl

Patryk Prusak
01151475@pw.edu.pl

supervisor: Anna Wróblewska
Warsaw University of Technology
anna.wroblewska1@pw.edu.pl

Abstract

Clustering textual data, such as user-generated comments or reviews, is a critical task in understanding and organizing large datasets. This work explores machine learning techniques to cluster text data effectively, focusing on applications like categorizing Amazon reviews into meaningful groups. By investigating state-of-the-art methods in natural language processing such vectorization and similarity metrics, and leveraging unsupervised learning algorithms, we aim to uncover latent patterns within textual datasets. Our approach aims to demonstrate the potential for automating content categorization, providing actionable insights for e-commerce, social media analysis, and other domains.

1 Introduction

Clustering textual data, such as online reviews, into meaningful categories is a critical yet challenging task, especially in the absence of predefined labels. Reviews can be grouped based on sentiment, topics, or other latent patterns, with no single ”correct” clustering. This ambiguity complicates evaluation, as traditional clustering metrics often fail to capture the semantic nuances of text. Furthermore, encoding methods like TF-IDF or BERT embeddings influence how clusters form, adding another layer of complexity. To address these challenges, we propose a pipeline that preprocesses, encodes, clusters, and evaluates text data, with a focus on developing metrics that quantify clustering quality across multiple interpretations. This approach aims to enhance the reliability and interpretability of unsupervised clustering for large-scale textual datasets.

1.1 Research Questions

We aim to explore the following questions:

1. What are the current methodologies for clustering text data, and how effectively do they perform in different contexts?
2. What metrics and evaluation frameworks are used to measure the performance of text clustering, particularly in unsupervised settings
3. What novel approaches or adaptations can enhance the robustness, interpretability, and accuracy of text clustering techniques?

2 State-of-the-Art in Text Clustering and Performance Evaluation

2.1 Clustering Textual Data

Clustering textual data has advanced significantly with the advent of modern natural language processing (NLP) techniques. These methods predominantly leverage embedding-based representations, deep learning, and hybrid approaches to uncover latent patterns in text data. This section explores notable approaches that have revolutionized clustering methodologies, including embedding models, hybrid frameworks, and specialized clustering techniques.

2.1.1 Semantic Clustering with Sentence-BERT

Sentence-BERT (SBERT) [8] has significantly enhanced semantic clustering by creating sentence-level embeddings optimized for downstream clustering tasks. Unlike traditional BERT models, which are designed for token-level tasks, SBERT employs a siamese and triplet network structure to produce semantically meaningful embeddings.

In the study by Ortakci [7], researchers explored how different pooling strategies impact the clustering performance of SBERT embeddings. Pooling techniques like mean, max, and attention-based

pooling were examined to optimize the representation of sentence-level information. The study demonstrated that leveraging pooling techniques in conjunction with SBERT embeddings significantly improves clustering methods like k-means and agglomerative clustering, leading to better semantic coherence within clusters.

2.1.2 Deep Embedded Clustering (DEC)

Deep Embedded Clustering (DEC) integrates feature learning with clustering by jointly optimizing latent space representations and cluster assignments. DEC employs autoencoders to reduce the dimensionality of textual data and refine clustering objectives. This approach enhances clustering accuracy and efficiency by dynamically adapting the latent space to the clustering task.

Xie et al. [13] proposed DEC to analyze high-dimensional text corpora, such as social media datasets, where the inherent noise and sparsity of data present challenges. By reconstructing the input data while simultaneously refining cluster centers, DEC effectively captures the underlying semantic structure, outperforming traditional clustering techniques in terms of both interpretability and precision.

2.1.3 Clustering with Large Language Models (LLMs)

The emergence of large language models (LLMs), such as GPT-3 and GPT-4, has opened new possibilities for clustering textual data. These models generate embeddings that encapsulate deep contextual and semantic information, which can be leveraged for clustering. Furthermore, LLMs facilitate an iterative feedback process to refine and validate cluster assignments.

Brown et al. [2] demonstrated the use of GPT-based embeddings for clustering scientific literature and business reports, achieving state-of-the-art performance in grouping semantically similar texts. Additionally, LLMs have been employed for cluster interpretation, where they assist in generating concise summaries or representative labels for each cluster, making the results more actionable.

2.1.4 Domain-Specific Clustering: Keyphrase Extraction

Domain-specific clustering approaches, such as keyphrase-based clustering, enhance the granularity of textual analysis by grouping texts based on semantically meaningful phrases. Keyphrase extraction techniques identify important terms that serve as features for clustering, effectively reducing noise in the data.

Bougouin et al. [1] proposed a hybrid approach combining TF-IDF and SBERT embeddings to organize large-scale news datasets by thematic content. Their approach outperformed generic clustering methods, particularly in domains where domain-specific terminology is prevalent.

2.1.5 Hybrid Approaches Combining Graph Neural Networks and NLP

Recent advances also include hybrid frameworks that combine graph neural networks (GNNs) with textual embeddings for clustering. These methods construct graphs where nodes represent textual data points and edges represent semantic similarities derived from embeddings. Clustering is then performed on the graph structure, leveraging community detection algorithms.

Kipf and Welling [6] introduced a graph-based clustering approach to analyze academic publications, where citation networks and textual content are jointly modeled. This method achieves higher precision in identifying research clusters compared to standalone embedding-based methods.

2.1.6 Comparative Evaluation of Clustering Techniques

Empirical studies have systematically compared the performance of these approaches across various datasets, such as scientific abstracts, customer reviews, and social media posts. Metrics like silhouette score, Davies-Bouldin index, and clustering purity indicate that embedding-based methods, particularly those utilizing SBERT, consistently outperform traditional feature-based clustering techniques. However, hybrid methods and LLMs exhibit superior performance in domains with complex semantic structures or large-scale datasets.

These metrics highlight the challenges of assessing clustering quality in the absence of definitive labels. Hybrid evaluation frameworks that combine quantitative scores with human validation can provide deeper insights into clustering effectiveness.

3 Datasets

For this study, we utilize several publicly available datasets that provide rich sources of textual data.

The following datasets will be used for clustering and evaluation:

1. **Amazon Reviews:** The Amazon Reviews dataset is a comprehensive collection of product reviews across various categories, including electronics, books, and clothing. It includes text data, ratings, and metadata such as product category and review date. This dataset is particularly useful for sentiment analysis and product categorization tasks. We will use reviews along with their ratings to explore clustering based on both sentiment and content. The dataset is available at: `https://nijianmo.github.io/amazon/index.html`.

2. **The 20 newsgroups text dataset:** The newsgroups dataset contains around 18,000 texts combined into 20 different clusters. It is a relatively small dataset; split into train and test dataset, which makes it a perfect sample for preliminary models testing. The dataset can be downloaded using scikit-learn API [11]: `https://scikit-learn.org/stable/datasets/real_world.html#the-20-newsgroups-text-dataset`

3. **Yelp Reviews:** The Yelp Reviews dataset contains reviews from Yelp, covering businesses across different industries such as restaurants, hotels, and service providers. Each review includes the text, user ratings, and additional metadata like business information. This dataset offers a diverse set of texts that can be analyzed to uncover patterns in customer satisfaction and business categorization. The dataset is accessible at: `https://www.yelp.com/dataset`.

4. **Twitter Sentiment Analysis:** The Twitter Sentiment140 dataset includes millions of tweets that are labeled with sentiment annotations: positive, negative, or neutral. It is widely used for sentiment analysis tasks and provides a valuable resource for clustering tweet content. We will use this dataset to explore how sentiment and text features can influence clustering outcomes. The dataset is available at: `https://www.kaggle.com/datasets/kazanova/sentiment140`.

4 Solution Plan

4.1 Clustering Textual Data

For the clustering process, we propose two approaches.

1. Sentence-BERT (SBERT) Approach: The first approach will leverage Sentence-BERT (SBERT), a transformer-based model that generates semantically rich, dense embeddings for sentences or entire paragraphs. These embeddings capture deeper semantic relationships between sentences than traditional word embeddings. After obtaining the embeddings, we will experiment with various clustering algorithms, such as k-means, HDBSCAN and others. These algorithms will allow us to explore different clustering structures, with HDBSCAN in particular being useful for identifying clusters of varying densities, which is often ideal for textual data due to its inherent noise and varying distribution of content.

2. The second approach will combine Word2Vec embeddings with dimensionality reduction techniques such as UMAP (Uniform Manifold Approximation and Projection), autoencoders, etc, and clustering techniques such as k-means, HDBSCAN or etc. Word2Vec, a well-established word embedding technique, captures semantic relationships between words. By reducing the dimensionality of the word vectors with UMAP, we aim to visualize and cluster words or phrases in a lower-dimensional space. HDBSCAN will be employed to detect clusters of varying densities, which is particularly useful for discovering non-linear structures in textual data.

4.2 Evaluating Clustering Performance

4.2.1 Standard Quantitative Metrics

We will start by using well-established metrics from the state-of-the-art to evaluate clustering performance. These include:

- **F1-Score (F1S)** [4]: A well-known metric to balance between precision and recall. In the context of text clustering, it helps to ensure that clusters are both accurate and comprehensive.

- **Normalized Mutual Information (NMI)**: A metric that measures the agreement between the predicted clustering and the true labels, adjusting for chance. NMI is commonly used to evaluate clustering in settings where the true clusters are known and provides a normalized score between 0 and 1, with 1 indicating perfect agreement.

- **Adjusted Rand Index (ARI)** [12]: Another widely-used metric that compares the clustering results with the ground truth, while correcting for the possibility of random clustering. The ARI ranges from -1 (no agreement) to 1 (perfect agreement).

- **Silhouette Score** [10]: This score assesses the quality of clustering by measuring how similar an object is to its own cluster compared to other clusters. A high silhouette score indicates well-separated clusters.

- **Davies-Bouldin Index** [5]: This is another clustering evaluation metric that measures the average similarity ratio of each cluster with the cluster that is most similar to it. A lower score indicates better clustering quality.

- **Homogeneity score (HS)** [9]: A metric which allows to evaluate clustering quality by measuring how uniform a cluster is concerning a single class or label. It assesses whether all texts within a cluster share the same ground truth label. A score of 1 indicates perfect homogeneity, while a score of 0 suggests complete randomness.

- **Calinski-Harabasz Index (CHI)** [3]: A metric evaluating cluster coherence and separation of a datasets without the ground truth label. A higher value indicates better-defined clusters, with greater separation and tighter cohesion.

These metrics will help quantitatively assess the clustering quality and alignment with the ground truth, ensuring that our models are performing accurately.

4.2.2 Novel Evaluation Using Large Language Models (LLMs)

In addition to the standard metrics, we will introduce a novel evaluation method by leveraging Large Language Models (LLMs). This method aims to provide a more qualitative and interpretive validation of the clusters. Specifically, we plan to:

- **Cluster Interpretability**: We will use LLMs to generate summaries or human-readable explanations for the clusters. This will allow us to validate whether the clusters have meaningful, coherent content that aligns with human intuition. For example, the model could generate a summary of each cluster's central theme, enabling a human evaluator to judge whether the cluster captures relevant and consistent aspects of the data.

- **Human Feedback and Adjustments**: By incorporating feedback from LLMs, we can refine and adjust clusters based on interpretations generated by the models. This process will be part of an iterative feedback loop that continuously improves the clustering outcomes.

- **Quality Assurance through LLMs**: LLMs will also assist in validating whether the clusters are semantically meaningful. For instance, the model could check whether the key topics within a cluster are related and ensure that any outliers or inconsistencies are flagged for review.

This approach brings a novel perspective to clustering evaluation, combining the strengths of quantitative metrics with the interpretability and flexibility of LLMs.

5 Proof of Concept (PoC)

5.1 Exploratory Data Analysis

As part of the PoC, we worked with two datasets from the list described earlier: 'The 20 newsgroups text dataset' and 'Amazon Reviews'.

5.1.1 The 20 newsgroups text dataset (EDA)

This datasets consists of two split subsets for training and testing. Number of rows is 11,314 and 7,532 respectively. All data is almost evenly divided into 20 clusters having 4-5% of data in each group.

Text lengths distribution is skewed with mean value significantly larger than the median. 75% of the data is lower than 980 characters with max length of around 15,000 symbols, which indicates extreme outliers in the dataset.

In order to exclude most common words and extra punctuation symbols, the following pre-processing was applied:

- Removing punctuation from the texts;
- Lowercase everything;
- Split by spaces;
- Removing "stop"/most common english words which do not influence the overall meaning of the texts.

Resulted tokens of both train and test subsets have similar distribution in terms of word usage.

Generally, most of the texts are either netral or positive. Less than 25% of the dataset have negative sentiment. Mean sentiment for both train and test is nuetral.

5.1.2 Amazon Reviews

The whole dataset is extremely large (over 233 million of reviews) and is split into groups by theme. For the PoC we selected 'Musical Instruments' subset which contains 1,511,675 of texts. This dataset has no labels on the clusters which makes the evaluation of the models more difficult.

The text lengths distribution is right-skewed containing major outliers. 75% of dataset contains less than 300 characters while the maximum is over 32 thousand.

For the texts pre-processing a similar approach as for 'the 20 news groups' was applied.

Most common words as well as the sentiment analysis indicate that reviews are generally positive with less then 25% of neutral and negative reviews.

5.2 Applied methods

We have applied various methods for text clustering including different embedding strategies and clustering models. Below we discuss the results of each method.

5.3 Word2Vec embeddings

Each transformed to tokens text was passed to Word2Vec of the gensim package to produce embedding vectors. The resulted vectors were passed directly to the well-known clustering methods, including K-Means, HDBScan, DBScan, Agglomerative clustering and Gaussian Mixture.

On the smaller dataset, the best method was HDBScan with 0.69 Silhoette Score. The lowest performance was shown by DBScan which was not able to divide dataset into clusters. Other models produced acceptable results ranging from 0.07 (GMM) to 0.47 (KMeans) Silhoette Score.

5.4 Word2Vec with UMAP embeddings

Additionaly to Word2Vec, a UMAP (Uniform Manifold Approximation and Projection) dimensionality reduction was applied to bring embedding to a 2D space.

On a smaller dataset most of the models were performing similar as without dimensionality reduction. It makes UMAP a good alternative to plain Word2Vec due to the faster processing. However, on a greater dataset models with UMAP embeddings tend to produce a significantly greater amount of clusters (¿50) with smaller amount of items in each group.

5.5 SBERT embeddings Clustering

We have conducted tests using SBERT embeddings (In particular the DistilRoBERTa model).

5.5.1 Amazon reviews dataset

We took a subset of 15000 comments (due to computational limitations) and compared combinations of Dimensionality Reduction Techniques and clustering techniiques.

For Dimensionality Techiniques we compared UMAP, PCA and no. Also each reduction technique was compared with different dimension sizes: 50, 100, 200 and 300. dimensionality redcution. As for clustering we tried K-Means, HDBScan, Agglomerative Clustering, Spectral Clustering, Gaussian Mixture and DBScan.

After all the computations the Silhouette score was the best for DBScan with no dimensionality reduction. Find Table 1 for more information on Silhouette score for different techniques.

5.5.2 20 News Group

For 20 News Group we did tests only on few clustering techniques and dim reductions and found out that the best combination is to use HSBScan with UMAP reducing the embedding to 50 dimensions.

This gives 0.73 Silhouette score.

5.6 SBERT embeddings with SVM

Additionally to try how supervised learning would work on 20 News Groups, we trained an SVM on SBERT embeddings.

	Technique	Silhouette score
	none_dbscan	0.99999875
	pca_300_dbscan	0.9999986
	pca_200_dbscan	0.99768126
	pca_100_dbscan	0.9834989
	pca_50_dbscan	0.9795705
	umap_50_hdbscan	0.5505021
	umap_50_kmeans	0.34788093
!	umap_200_kmeans	0.33983523
	umap_300_kmeans	0.31991813
	umap_100_kmeans	0.3172989
	umap_200_agg	0.30400354
	umap_300_agg	0.2896424
	umap_50_gmm	0.2891189
	umap_200_hdbscan	0.2869405
	umap_200_gmm	0.28363466
	umap_200_spectral	0.28261667

Table 1: Best Clustering Results: SBERT For Amazon Reviews

The main finding was that SVM was best according to F1 score only with non-reduced embeddings. If we use UMAP, the accuracy decreases substantially. F1 Score for 20 News Group (Subset of 5000 rows) is 60%.

5.7 Future Work

- Utilize alternative embedding techniques, such as BERT and LLM embeddings (like Gemma), and compare the results against Sentence-BERT embeddings (SBERT).
- Introduce additional metrics for evaluating the effectiveness of clustering methods such as CHI (Calinski–Harabasz Index)
- Leverage LLMs for labeling a subset of Amazon reviews to aid in the supervised learning process.

References

[1] Adrien Bougouin, Florian Boudin, and Béatrice Daille. Keyphrase extraction for clustering purposes: Combining textual and semantic features. *Journal of Information Retrieval*, 2021.

[2] Tom B. Brown, Benjamin Mann, and Nick et al. Ryder. Language models are few-shot learners. *Advances in Neural Information Processing Systems (NeurIPS)*, 33, 2020.

[3] Tadeusz Caliński and Jerzy Harabasz. A dendrite method for cluster analysis. *Communications in Statistics-theory and Methods*, 3(1):1–27, 1974.

[4] Nancy Chinchor and Beth M Sundheim. Muc-5 evaluation metrics. 1993.

[5] David L. Davis and Donald W. Bouldin. A cluster separation measure. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-1(2):224–227, 1979.

[6] Thomas N. Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. *International Conference on Learning Representations (ICLR)*, 2017.

[7] Yasin Ortakci. Revolutionary text clustering: Investigating transfer learning capacity of sbert models through pooling techniques. *Journal of Advanced Text Clustering*, 2023.

[8] N Reimers. Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084*, 2019.

[9] Andrew Rosenberg and Julia Hirschberg. V-measure: A conditional entropy-based external cluster evaluation measure. In *Proceedings of the 2007 joint conference on empirical methods in natural language processing and computational natural language learning (EMNLP-CoNLL)*, pages 410–420, 2007.

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

[11] Scikit-learn. The real world datasets.

[12] Douglas Steinley. Properties of the hubert-arable adjusted rand index. *Psychological methods*, 9(3):386, 2004.

[13] Jianlong Xie, Ross Girshick, and Ali Farhadi. Unsupervised deep embedding for clustering analysis. *Proceedings of the International Conference on Machine Learning*, 2016.