Movie Dialogue Clustering for Character Comparison

Gary Shen
Lexington High School
MehtA+

Abdullah Arshad
Beaconhouse Potohar Campus
MehtA+

Muhammad Ahmed Imran
Beaconhouse Potohar Campus
MehtA+

Paul Sungjai Yoo
Thomas Jefferson HS for Sci & Tech
MehtA+

July 31, 2025

Abstract

Character dialogues in movies convey narrative roles and emotional depth, yet existing analyses often rely on single-feature approaches, such as sentiment or topic modeling, which fail to capture the multifaceted nature of dialogue styles across genres. This study addresses this limitation by applying k-means clustering to group 103 characters from 34 films in a custom dataset, `movie_dialogues.csv`, to uncover stylistic and thematic patterns for applications in scriptwriting, film studies, and recommendation systems. We selected KMeans for its scalability and ability to handle high-dimensional dialogue features, comparing three feature sets: stylistic (19 features), semantic (384-dimensional embeddings), and hybrid ($\sim$50 features). Experiment 1's 9-cluster configuration achieves balanced cohesion (44.44% accuracy, 33.33% diversity), grouping characters by stylistic traits like verbosity, while Experiment 2's semantic embeddings yield the highest net accuracy (35%) by capturing thematic roles (e.g., heroic vs. sarcastic dialogues). Qualitatively, clusters reveal archetypes like sci-fi protagonists or comedic sidekicks, enhancing narrative analysis. These results establish a robust framework for dialogue clustering, though hybrid features risk overfitting. Future work aims to refine feature selection and apply findings to automated script analysis tools.

1 Introduction

Movies captivate audiences through rich narratives, where character dialogues define personalities, emotions, and story arcs. Analyzing these dialogues can reveal patterns—such as a hero's rousing speeches or a villain's cryptic threats—that inform filmmaking, script analysis, and content recommendation systems. However, traditional methods like sentiment analysis or topic modeling often focus on isolated dialogue aspects, failing to integrate stylistic, emotional, and semantic elements. This gap limits the ability to group characters by their dialogue styles across diverse genres, hindering comprehensive narrative analysis.

This study tackles this challenge by applying k-means clustering to dialogues of 103 characters from 34 movies, sourced from a self-generated dataset, `movie_dialogues.csv`, scraped from imsdb.com. Our target audience includes researchers studying narrative structures, filmmakers crafting character-driven stories, and data scientists developing recommendation systems. We chose k-means clustering for its simplicity, scalability, and effectiveness in grouping high-dimensional data, enabling us to compare dialogue styles across genres like science fiction (*Star Wars*), animation (*Finding Nemo*), and drama (*The Godfather*).

We conducted three experiments: Experiment 1 uses 19 stylistic and sentiment features across 5, 9, 15, and 20 clusters; Experiment 2 employs 384-dimensional `all-MiniLM-L6-v2` semantic embeddings with 20 clusters; and Experiment 3 combines $\sim$50 stylistic, sentiment, and thematic features with 20 clusters. Performance is evaluated using accuracy (cohesive clusters), diversity (clusters spanning >5 movies), single-character clusters (overfitting), outliers (misclassified characters), and net accuracy (accuracy minus single-character clusters). Our contributions are:

1. A novel comparison of stylistic, semantic, and hybrid feature sets for dialogue clustering.
2. Identification of optimal cluster counts for stylistic analysis, balancing cohesion and granularity.
3. Demonstration of semantic embeddings' superiority in capturing thematic dialogue patterns, advancing cross-genre character comparison.

This work aims to provide a robust framework for dialogue-based character analysis, with applications in narrative studies and beyond.

2 Related Work

Clustering, a key unsupervised learning technique, groups similar data points based on features, with applications in text analysis, image segmentation, and social network analysis. KMeans clustering, used here, assigns data to clusters by minimizing within-cluster variance, offering scalability for high-dimensional data like movie dialogues [1]. Hierarchical clustering builds nested structures, ideal for hierarchical relationships but computationally intensive [**?**]. DBSCAN groups data by density, excelling with irregular clusters but sensitive to parameter settings [**?**]. Spectral clustering uses graph-based methods, suitable for complex relationships, yet requires careful tuning [**?**]. Each method has trade-offs, but KMeans balances efficiency and effectiveness for our dialogue dataset.

In natural language processing (NLP), dialogue analysis leverages tools like NLTK [2] and TextBlob [3] for stylistic features (e.g., sentence length, part-of-speech ratios), VADER for sentiment analysis [4], and `all-MiniLM-L6-v2` for semantic embeddings [5]. Empath [6] enriches analysis with emotional and topical categories. Prior movie dialogue studies often focus on single-feature analyses, limiting their ability to capture combined stylistic and semantic patterns. Our work advances this by comparing multiple feature sets, offering a comprehensive approach to clustering dialogues across genres.

3 Methodology

3.1 Dataset

The dataset, `movie_dialogues.csv`, comprises dialogues from 103 characters across 34 movies, scraped from imsdb.com using a custom HTML parser. It spans genres: science fiction (6 movies, e.g., *Star Wars*, *Interstellar*), action (7 movies, e.g., *Black Panther*), animation (4 movies, e.g., *Finding Nemo*), drama (5 movies, e.g., *The Godfather*), and others (12 movies, e.g., *Titanic*). Each movie contributes 3–5 characters (average: 3.03), selected by dialogue length. Dialogue lengths range from 100 to over 10,000 words, reflecting character prominence (e.g., protagonists like Han Solo vs. supporting roles like Threepio). Figure 1 illustrates dialogue length distribution, highlighting dataset diversity.

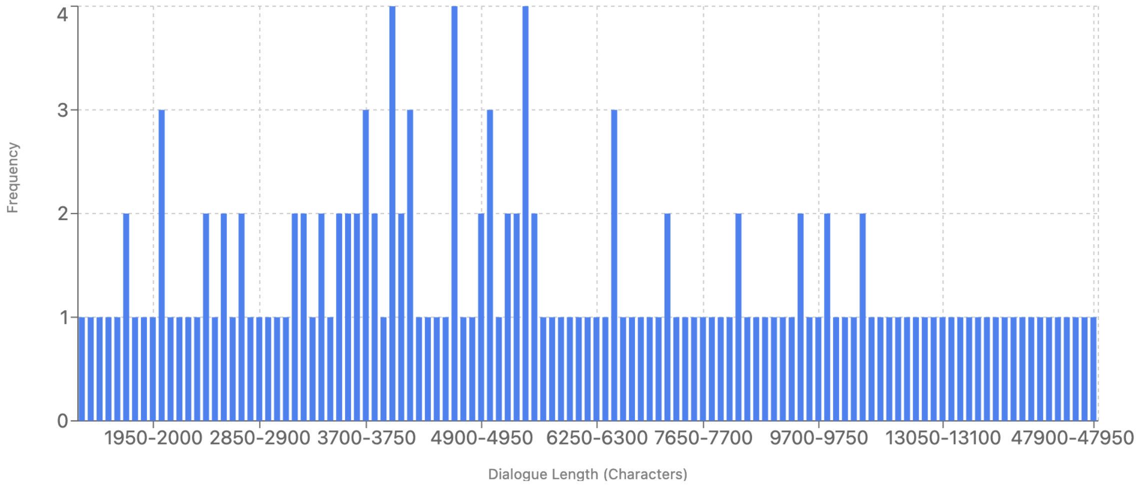

Figure 1: Histogram of dialogue lengths for 103 characters.

Preprocessing Dialogues were tailored for each experiment. Experiment 1 used NLTK to tokenize dialogues, extracting 19 stylistic and sentiment features (e.g., `avg_sent_len`, `polarity`). Experiment 2 concatenated dialogues per character for `all-MiniLM-L6-v2` embeddings (384 dimensions). Experiment 3 extracted ∼50 features, combining lexical, sentiment, Empath, and thematic keywords. All features were standardized with `StandardScaler`, and PCA was applied for 2D visualization.

3.2 Model Construction

K-means clustering sorts characters into k groups by finding the center (centroid) of each group and assigning characters to the closest one, minimizing the total distance within groups:

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

where k is the number of clusters, C_i is the i-th cluster, x is a character's dialogue features, and μ_i is the centroid. Imagine sorting characters into buckets based on how they talk: with 5 clusters, you might get broad groups like "heroes" (e.g., T'Challa's inspiring speeches) or "villains" (e.g., Vader's menacing lines); with 20 clusters, you get finer groups like "witty rogues" (e.g., Han Solo) or "wise mentors" (e.g., Gandalf). We tested three experiments to find the best grouping method.

- **Experiment 1: Stylistic and Sentiment Features**
  - **Features**: 19 features, including sentence length, swear word count, adjective ratio, and sentiment scores (polarity, subjectivity) from NLTK, TextBlob, and VADER.
  - **Preprocessing**: Tokenized dialogues, computed features, standardized, visualized with PCA.
  - **Characteristics**: Groups characters by how they speak (e.g., talkative vs. concise). With 5 clusters, it might group "verbose leaders" (e.g., Gandalf) or "terse villains" (e.g., Vader); 9 or 20 clusters refine these into more specific styles, like "formal speakers."

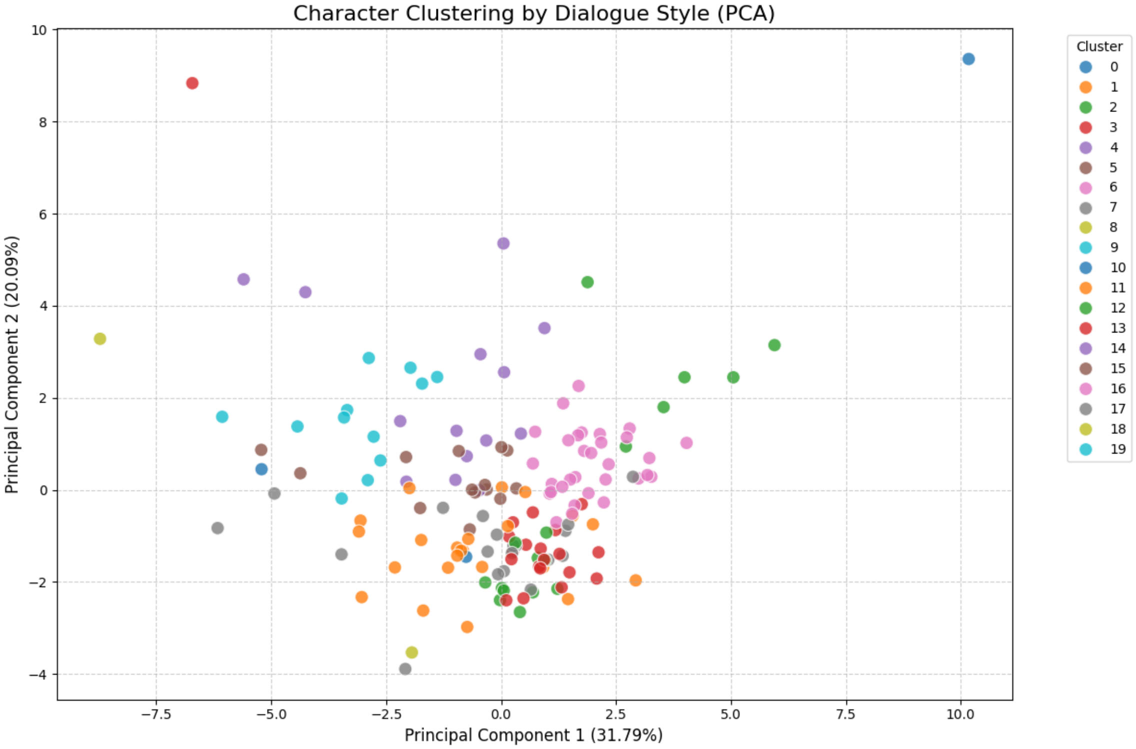

Figure 2: PCA scatter plot of Experiment 1's 20-cluster configuration, showing stylistic dialogue clusters.

- **Experiment 2: Semantic Embeddings**
  - **Features**: 384-dimensional `all-MiniLM-L6-v2` embeddings capturing dialogue meaning.
  - **Preprocessing**: Concatenated dialogues, encoded, standardized, PCA-visualized.
  - **Characteristics**: Groups characters by what they say (e.g., thematic roles). With 20 clusters, it separates "inspirational leaders" (e.g., T'Challa) from "sarcastic rogues" (e.g., Han Solo) or "comedic sidekicks" (e.g., Dory).

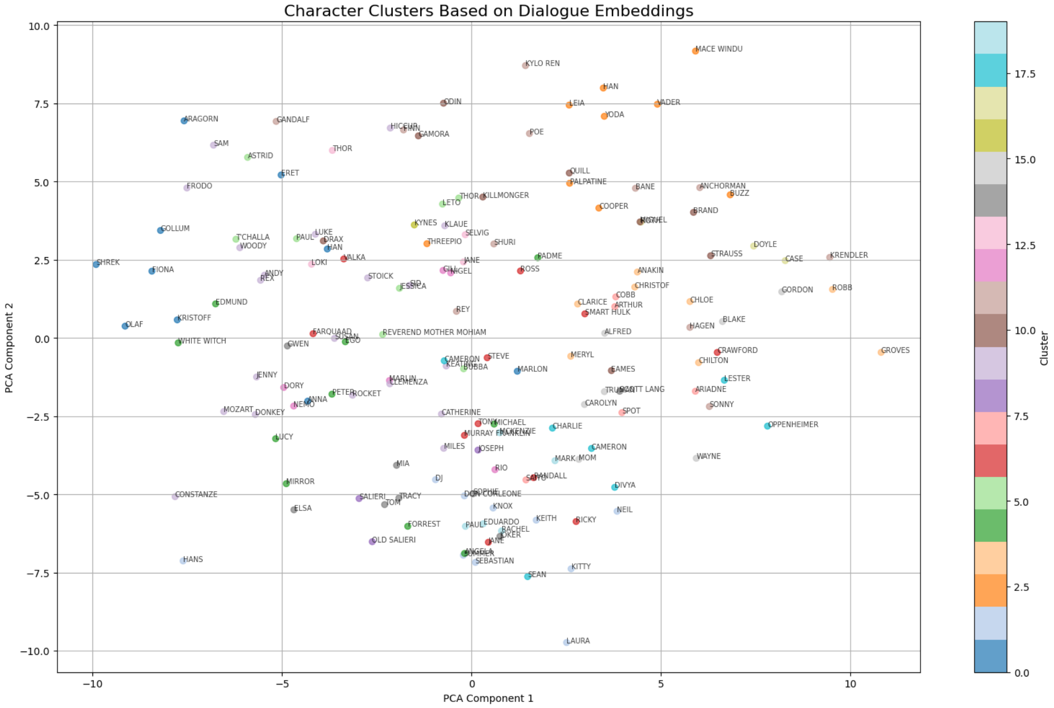

Figure 3: PCA scatter plot of Experiment 2's 20-cluster configuration, showing semantic dialogue clusters.

- **Experiment 3: Hybrid Features**
  - **Features**: ∼50 features combining lexical (e.g., text complexity), sentiment, Empath (e.g., anger score), and thematic keywords (e.g., hero-related words).
  - **Preprocessing**: Extracted features, standardized, PCA-visualized.
  - **Characteristics**: Balances how and what characters say but may overfit. With 20 clusters, it might create overly specific groups like "angry leaders" vs. "calm leaders."

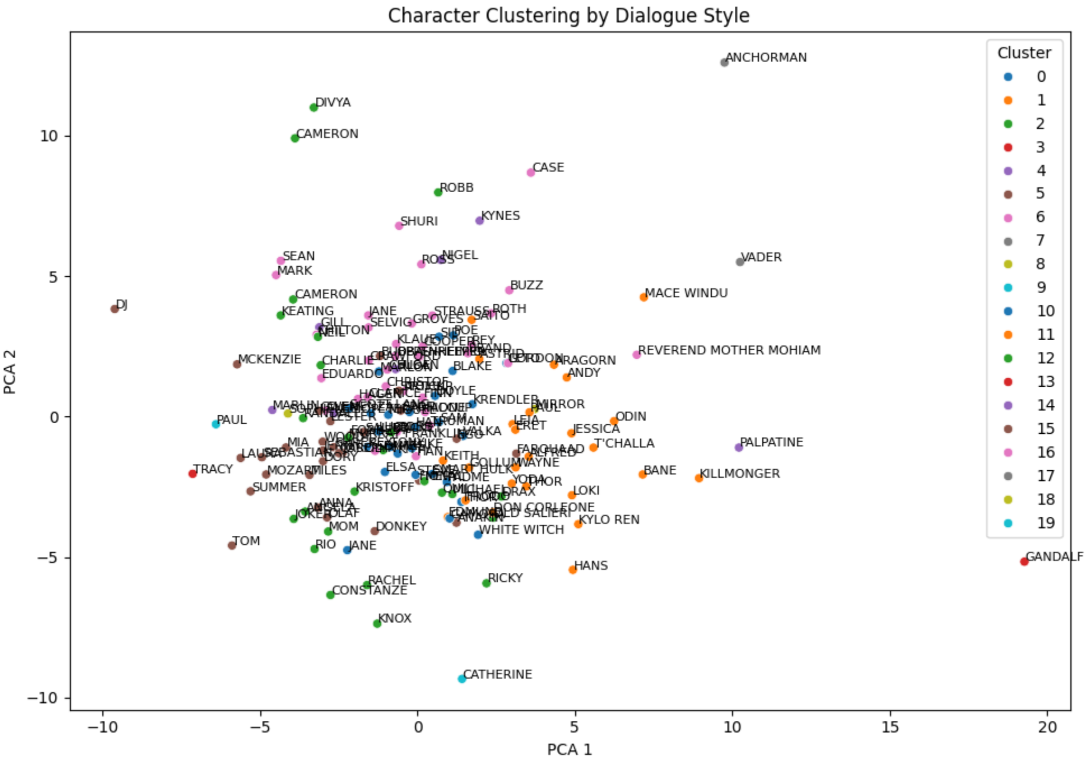

Figure 4: PCA scatter plot of Experiment 3's 20-cluster configuration, showing hybrid dialogue clusters.

4 Results and Discussion

We evaluated clustering with metrics: accuracy (percentage of cohesive clusters, where characters share similar dialogue styles), diversity (percentage of clusters with >5 movies, indicating genre variety), single-character clusters (percentage of clusters with one character, suggesting overfitting), outliers (percentage of misclassified characters), and net accuracy (accuracy minus single-character clusters, reflecting effective grouping). Results show how feature sets and cluster counts shape character groupings, with qualitative insights into cluster themes.

Comparison 1: Experiment 1 Across Cluster Numbers Experiment 1's 9-cluster configuration was optimal, achieving 44.44% accuracy and 33.33% diversity (Table 1). It grouped characters into stylistic categories, such as verbose leaders (e.g., Gandalf's eloquent speeches) or concise villains (e.g., Vader's sharp commands). The 5-cluster setup (60% net accuracy) was too broad, merging distinct styles (e.g., heroes like T'Challa with antiheroes like Loki), likely due to limited features capturing only broad patterns. The 20-cluster setup (15% single-character clusters) overfit, splitting similar characters (e.g., Han Solo and Quill), as stylistic features like sentence length lacked semantic context. Moderate cluster counts (9 or 15) balance cohesion and granularity, driven by features like adjective ratio and polarity.

Table 1: Experiment 1 Performance Across Cluster Numbers

Configuration	Clusters	Accuracy (%)	Diversity (%)	Single-Char. (%)	Net Accuracy (%)
Simplified	5	60.00	40.00	0.00	60.00
Broad	9	44.44	33.33	11.11	33.33
Nuanced	15	46.67	26.67	13.33	33.34
Granular	20	40.00	25.00	15.00	25.00

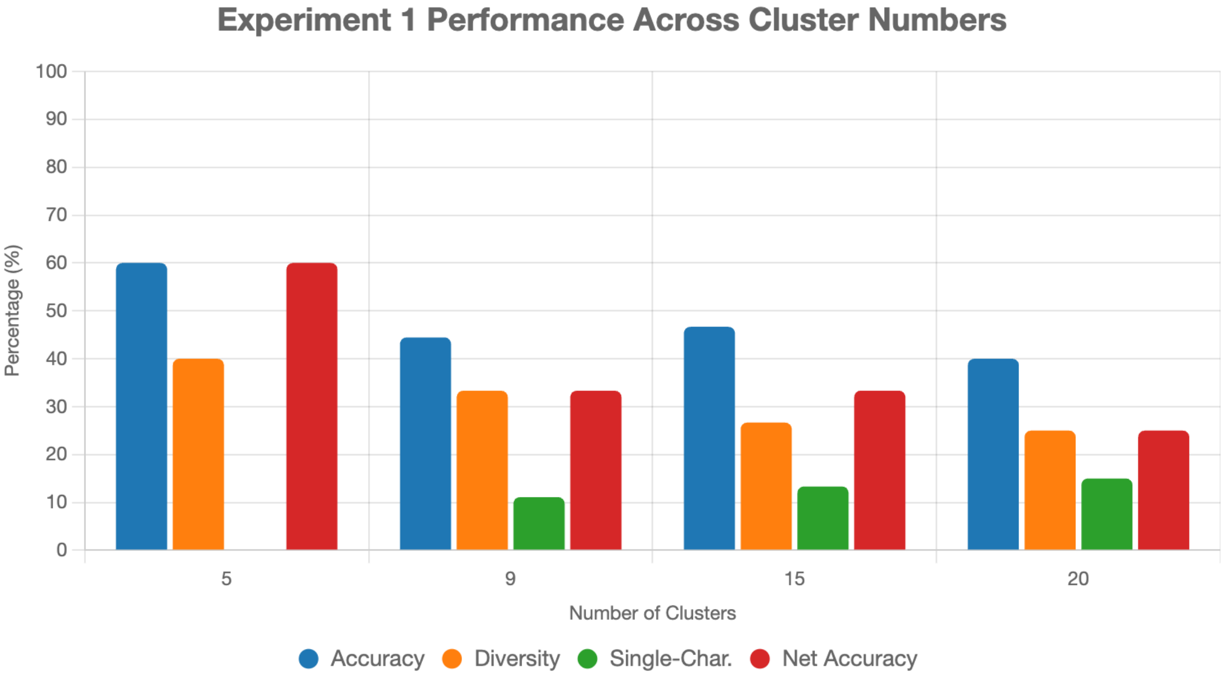

Figure 5: Experiment 1 performance across 5, 9, 15, and 20 clusters, showing trade-offs in accuracy and diversity.

Comparison 2: Experiments 1, 2, and 3 (20 Clusters) Experiment 2, using semantic embeddings, achieved the highest net accuracy (35%, Table 2), forming clusters like sci-fi protagonists (e.g., Han Solo, Leia in Cluster 2) or adventure characters (e.g., Dory, Marlin in Cluster 12). Its low single-character cluster rate (5%) reflects robust generalization, as embeddings capture thematic meaning (e.g., heroism, sarcasm) across genres. Experiment 3's high accuracy (65%) was undermined by overfitting (40% single-character clusters), likely due to excessive features (e.g., combining polarity and Empath scores) creating overly specific clusters. Experiment 1's stylistic focus (25% net accuracy) missed thematic depth, grouping characters by surface traits (e.g., verbosity) rather than roles. Semantic embeddings excel because they encode contextual meaning, aligning characters like T'Challa and Paul (heroic leaders) despite different genres.

Table 2: Performance with 20 Clusters

Experiment	Features	Accuracy (%)	Single-Char. (%)	Outlier (%)	Net Accuracy (%)
Experiment 1	19	40.00	15.00	30.00	25.00
Experiment 2	384	40.00	5.00	25.00	35.00
Experiment 3	50	65.00	40.00	15.00	25.00

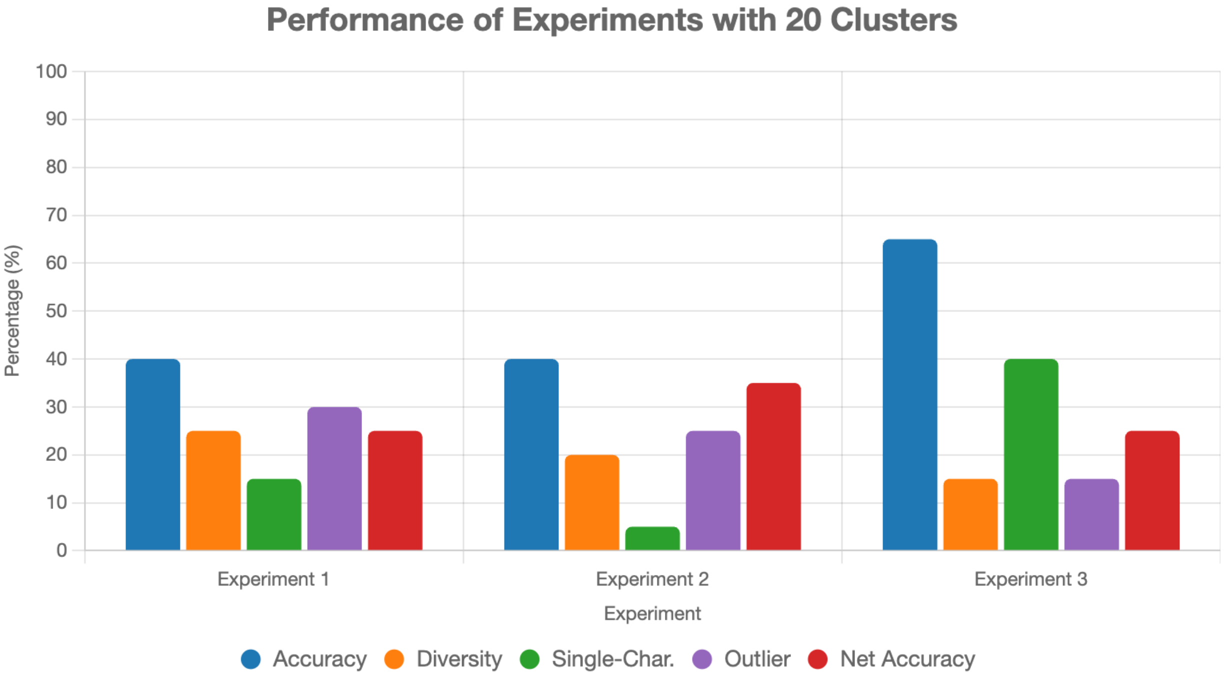

Figure 6: Performance of Experiments 1, 2, and 3 with 20 clusters, highlighting Experiment 2's superior net accuracy.

5 Conclusion and Future Work

This study establishes that semantic embeddings (Experiment 2) excel in clustering movie dialogues, achieving 35% net accuracy by capturing thematic roles (e.g., sci-fi heroes, comedic sidekicks), while stylistic features (Experiment 1, 9 clusters) balance cohesion and granularity for style-based grouping. These findings enable applications like automated script analysis, character archetype identification, and recommendation systems by revealing dialogue patterns across genres. Experiment 3's overfitting underscores the need to optimize feature selection. Future work could explore hierarchical clustering for nested dialogue structures, DBSCAN for irregular cluster shapes, or advanced embeddings (e.g., BERT-based models) to enhance semantic analysis. Validating results on larger, diverse datasets and integrating visual or tonal features could further improve narrative analysis tools.

6 Division of Labor

- Gary Shen: Webscraper engineering, dataset preprocessing, Experiment 1 development, comparative analysis, visualization.
- Abdullah Arshad: Experiment 3 development, feature selection, cluster coherence analysis.
- Muhammad Ahmed Imran: Experiment 2 development, visualization.
- Paul Sungjai Yoo: Dataset preprocessing, cluster number analysis, k-means model construction.

7 Acknowledgements

The authors would love to thank MehtA+, mentors Mr. Bhagirath Mehta and Mr. Arrun Sivasubramanian, and instructor Ms. Haripriya Mehta for their support and encouragement. We also acknowledge the developers of NLTK, TextBlob, VADER, SentenceTransformers, and Empath for their open-source tools.

References

[1] James MacQueen. Some methods for classification and analysis of multivariate observations. 1:281–297, 1967.

[2] Steven Bird, Ewan Klein, and Edward Loper. *Natural Language Processing with Python*. O'Reilly Media, 2009.

[3] Steven Loria. Textblob: Simplified text processing, 2018.

[4] C.J. Hutto and Eric Gilbert. Vader: A parsimonious rule-based model for sentiment analysis of social media text. In *Proceedings of the Eighth International AAAI Conference on Weblogs and Social Media*. AAAI Press, 2014.

[5] Nils Reimers and Iryna Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084*, 2019.

[6] Ethan Fast, Binbin Chen, and Michael S. Bernstein. Empath: Understanding topic signals in large-scale text. *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, pages 4647–4657, 2016.