Unsupervised behavioral profiling of deception (bluffing) using structured and embedding features with clustering algorithm

Kevin S. Bertoldo[1], Rita M. S. Julia[1], Elaine R. Faria[1], Marcelo Z. do Nascimento[1]

[1]Faculdade de Ciência da Computação – Universidade Federal de Uberlândia (UFU)
Postal Code 38.400-902 – Uberlândia – MG – Brasil

{kevinbertoldo, rita, elaine, marcelo.nascimento}@ufu.br

Abstract. Introduction*: This work investigates the use of Machine Learning techniques to analyze bluffing behavior in the social deduction game Ultimate Werewolf.* ***Objective:*** *To identify and characterize behavioral profiles of bluffing players, addressing challenges such as temporality, mixed data types, and the non-exclusive nature of behavioral patterns.* ***Methodology or Steps****: A pipeline that combines data preprocessing, Large Language Model for bluff classification, feature engineering with temporality and interaction attributes, and clustering. Multiple experimental scenarios were evaluated, including structured features, embedding-based representations, and hybrid approaches.* ***Results:*** *Although not achieving higher internal validation scores, experiments show that combining feature filtering with embedding-based representations and Gower distance leads to more coherent and interpretable clusters. These findings indicate that bluffing behavior is not homogeneous, but instead organized into identifiable patterns, highlighting the effectiveness of the proposed approach for analyzing behavior in conversation-driven games.*
Keywords *Machine Learning, Social Deduction Games, Role-Playing Games (RPG), Bluff, Player Behavior Modeling*

1. Introduction

Machine learning (ML) is widely regarded as a central component of past, present, and emerging technologies [Harley e Sparkman 2019]. It can be applied to problems ranging from simple tasks, such as linear regression to model the relationship between variables [Maulud e Abdulazeez 2020, Hope 2020], to complex scenarios that require systems to learn appropriate data representations [LeCun et al. 2015]. These applications span mathematical, statistical, and computational studies, including, for example, predicting new cases of coronavirus disease (COVID-19) using regression models [Rath et al. 2020], as well as more advanced tasks such as diagnosis, monitoring, and treatment supported by ML and artificial intelligence (AI) [Mottaqi et al. 2021].

In this context, games have proven to be a suitable and challenging case study to evaluate the effectiveness of modern ML techniques. Different game genres provide diverse environments for experimentation. A few examples are: Grid-based 2D Games with a highly cooperative and competitive environment introduce multi-agent interactive scenarios (e.g. The Battle of Polytopia, Overcooked); Pixel-based 2D Games where their simplicity leads to a perfect setting to use searching and learning techniques to generate levels (e.g. Mario, Angry Birds); Commercial Games introduce constraints, scalability challenges, and dynamic environments (e.g. StarCarft); Platform Games (including

board, card, and puzzle games) allow the evaluation of ML methods under varying rules, states, and sometimes partial observation through conversation (e.g. Werewolf Ultimate, Go) [Hu et al. 2024]. Amongst the Platform Games, a subgenre stands out, the role-playing game (RPG). RPGs are games in which players assume a role within a structured world, which may be fictional or non-fictional, and collaborate or compete to achieve specific objectives. Interaction through conversation, either with non-playable characters (NPCs) or other players, is a fundamental component of the gameplay. These characteristics make RPGs particularly suitable for studying communication-driven dynamics. Moreover, prior studies indicate that RPGs can foster social bonds, strengthen interpersonal relationships, and help individuals overcome shyness in group discussions [Jaques e Bisol 2019]. According to the authors in [Grande-de Prado et al. 2020], RPGs can benefit the players in many ways. It can be a playful and recreational activity that enhances reading, writing and mental calculation skills, while it also encourages empathetic and tolerant behavior.

Thus, for socialization and communication patterns, the conversation-driven RPG Ultimate Werewolf [Games 2024] provides a suitable case study. Its hidden roles and complex player interaction are native to social deduction games and pose a significant challenge to feature extraction, clustering decisions, and interpretability. Bluffing represents a particularly relevant interaction, as it involves deliberate attempts to deceive and manipulate other players to achieve game objectives [Marwala e Hurwitz 2009]. These behaviors can significantly influence gameplay dynamics and may occur at different moments, adding temporal complexity to the analysis.

Within this scope, this work investigates clustering techniques and analyzes bluffing behavior in a deceptive, conversation-driven RPG. The main motivations for this work's case study are threefold: (i) the outstanding challenge of dealing with human language conversation-driven games, in which interpretation strongly depends on player behavior; (ii) to the best of our knowledge, the lack of prior work focusing on the analysis of bluffer behavior in RPG games; (iii) games have been increasingly utilized as a tool for learning enhancement in Education (serious games - SG). In this sense, the study of conversation-driven games can lead to further applications in Education, serving as a learning incentive for languages as well as socialization.

Although the application of ML to games has been extensively studied, the identification and differentiation of bluffing behavior remain relatively unexplored. This paper addresses this gap by focusing on the challenges associated with clustering such behaviors.

2. Related Work

This section organizes prior work into three main directions: player profiling in RPG environments, clustering methods for behavioral data, and deception games and Large Language Models (LLMs).

Player profiling in RPG environments In this context, [Drachen et al. 2014] analyze multiple clustering approaches using data from World of Warcraft. The article's author also does an Archetype Analysis, which is an important aspect for RPGs and should be considered when exploring them. Similar to this work, [Drachen et al. 2012] study

a distinct Fantasy MMORPG called TERA using a partitional algorithm; the author highlights that partitional methods capture general behavioral distributions, while SIVM better identifies extreme behavioral patterns. Aside from that, TERA's research also interprets the clustering results in Archetypes and shows that K-means and SIVM found distinct Archetypes, all depending on the MMORPG achievements. Still, these studies focus on action-based gameplay metrics rather than conversational interaction, leaving dialogue-driven behaviors such as deception largely unexplored.

Clustering methods for behavioral data From a broader perspective, [Bauckhage et al. 2015] provide a comprehensive overview of clustering techniques for game behavior analysis. The authors cover K-means and Spectral algorithms but also give substantial game context, guidelines, and examples, including their previous work [Drachen et al. 2012] as a relevant work for game behavior and proper cluster interpretation. Overall, existing literature demonstrates the effectiveness of clustering for player modeling and educational analytics.

Deception games and LLMs Recent work has increasingly explored social deduction games such as Mafia and Werewolf as environments for studying deception and social reasoning. For instance, [Yoo e Kim 2024] investigate deception detection using LLMs, demonstrating improved accuracy over traditional approaches. Similarly, [Lai et al. 2023] introduce a multimodal dataset for modeling persuasion strategies in Werewolf gameplay. Other studies have used these environments LLM reasoning and strategic interaction in multi-agent settings. However, existing approaches predominantly focus on supervised prediction tasks or model evaluation. In contrast, this work adopts an unsupervised perspective, aiming to identify behavioral profiles of bluffing players through clustering, combining structured features, temporal dynamics, and embedding-based representations. This gap motivates the present work.

3. Methodology

This work proposes a pipeline for clustering player behavior in a conversation-driven RPG, combining attribute engineering, LLM-based classification, and two distinct clustering algorithms (CA): HDBSCAN* [Malzer e Baum 2019] and Spectral [Ding et al. 2024].

Conversation-driven RPGs present a challenging environment due to multi-agent interactions, hidden roles, and dynamic behaviors. This study focuses on *bluffing* as the core behavioral signal to create the dataset and apply clustering and AI methods. Despite bluffs being a reduced scope of the dialogues, they present several challenges, such as: cross-checking information across different game logs and timestamps and finding an adequate strategy to correctly represent the aforementioned behaviors through attributes. The Ultimate Werewolf RPG was used as a case study, since it is a resourceful RPG enriched by the complexity of social interactions and the presence of roles. The following subsections present the game rules, the bluff scenarios mapped out, and a detailed view of each module that make up the implemented pipeline architecture illustrated in Figure 1.

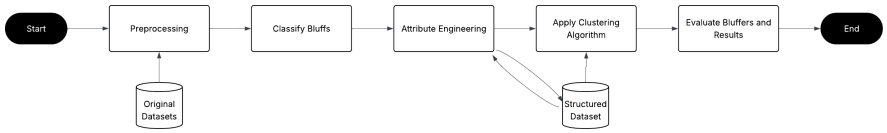

Figure 1. Flowchart with a representation of the methodology pipeline

3.1. Game Rules and Bluff Scenarios

The Ultimate Werewolf is a Round-based RPG focused on deception, deduction, and negotiation available on both tabletop and digital platforms through different distributors. In this game, each player assumes a role with unique objectives and abilities. The game ends when at least one player achieves their role's objective.

A round consists of a day and a night phase. A day is a discussion-driven phase full of social strategies for gathering or fabricating information to achieve the player role's final goal. At the end of this phase, all players vote someone out. The night is a phase in which players can use their unique role-based ability (protecting players, eliminating players, changing roles, etc.). Some of the abilities' consequences are then revealed to the players in the following round during the day phase. The Ultimate Werewolf game can be played in several modes, but two will be specified below, given that they form the datasets described in the following subsections.

- Standard mode: is conducted by a Moderator and composed of several rounds. The Moderator does not play the game and is responsible for keeping the turns organized.
- One Night mode: is a faster version of the standard game. It does not have a Moderator, has only one Werewolf, and consists of a night turn followed by a day turn.

As presented before, during the discussion phase (day phase), people tend to use many social artifices, bluff being one of them. Bluffing is defined as the act of intentionally providing misleading information to influence other players' decisions, a common strategy in games such as poker. In this work, two types of bluffs were mapped and used in feature engineering as a basis for clustering bluffers' behavior.

- Role bluff: when a player lies about their given role.
- Vote bluff: when a player lies about who they will be voting out. The time difference between the utterance and the vote is an important aspect of this type of bluff, as conversations continue over time.

3.2. Module 1: Preprocessing the Original dataset

The foundational dataset for this clustering process stores real disputes from the Ultimate Werewolf game obtained from the distinct datasets [Lai et al. 2023] and [Sun 2025]. The first contains 199 records extracted from two different video platforms (151 from YouTube and 48 from Ego4D). Further, this domain is divided into groups of folders, one composed of raw data instance files, and the other of processed data instance files.

The raw files are the core data for preprocessing and hold gameplay video identifiers; day round conversation transcripts separated by video timestamp and player

name in text files; and voting outcomes with players' roles and voting results in JSON files. Processed data instances were not used. The second dataset contains 9,505 JSON files, where each key provides a different set of information related to the game, such as transcripts, timestamps, voting results, eliminated players, and more.

The preprocessing was a data extraction work from the files from both datasets mentioned above to create a unified Pandas DataFrame in Python (136,673 records), where attributes were defined and processed into a temporary dataset with game metadata, player information (no Personally Identifiable Information contained), timestamps, and interaction data (sentences). This temporary dataset contains the initial core information necessary to define both considered bluff types (role and vote), which was used in the following module.

3.3. Module 2: Classify bluffs

Bluff is a common strategy in social games, police interrogations, and social life. It is an artifice capable of pressuring, captivating, or gathering information from others [Perillo e Kassin 2011]. Due to its potential and broad application, it proves to be a suitable candidate to mold distinct behaviors and profiles. It is important to highlight that bluff manipulation and analysis come with a challenge to overcome. Since the game has well-defined rounds with distinct available actions, the bluff and actions that need to be cross-checked occur at very separate times (time lag), which includes the complication of detecting and dealing with the bluffs.

That said, the temporary dataset was crucial because it gathered all the information necessary to pinpoint utterances where a bluff happened. Then, due to its capacity to analyze complex human conversations, the LLM DeepSeek (DS) was employed to classify bluff occurrences. A proper prompt engineering process gave DS the necessary extra capacity to be able to classify the dataset sentences as non-bluff, vote bluff, role bluff, or vote and role bluff [Hagendorff 2024].

The prompt consisted of a context statement defining how DS was supposed to act; an explanation of the conditions necessary for a role and/or a vote bluff to happen; points of attention and particularities related to the attributes; and 8-shot examples. Each example included a player name, utterance, initial role, final role, and the person voted by the given player, covering all possible scenarios and tricky utterances. The prompt was refined iteratively through empirical testing and DS self-evaluation strategies to improve classification consistency [Gu et al. 2026]. A manual inspection of a subset of classifications was performed to ensure the consistency and plausibility of the annotations. The final classification generated two new label attributes for each sentence: role bluff and vote bluff flags, which were added to the temporary dataset comprising a total of 13 attributes.

3.4. Module 3: Attribute Engineering

After bluff identification, the temporary dataset's 13 attributes went through feature processing, transforming into 35 attributes, which include behavioral metrics (e.g., accusations, denials), bluff-related features (e.g., number of bluffs, temporal distribution), interaction features (e.g., players mentioned, voting targets), and role-based attributes (e.g., role changes, werewolf flag). In sequence, the dataset was grouped by game,

round, and player. Within this new granularity, the bluffer occurrence was given by the existence of at least one sentence labeled as bluff, generating the Structured Dataset 1 (SD1) composed by the set of sentences where a bluffer was identified (3,782 records). Therefore, each record of SD1 contains 35 attributes that represent a bluffer.

SD1 is a representative dataset with 2603 rows classified as player rounds where only role bluffs were used, 973 rows where only vote bluffs were used, and 206 with both being used. Role-wise, 1957 rows were werewolf rounds, and 1825 were other roles. The unbalancing of the bluff type is expected, since the vote bluff can draw suspicions and is easier to be validated by the players.

SD1 normality was assessed using plotting exploration and the Shapiro–Wilk test [Field et al. 2012], which, with a 0.261 result and p-value of 5.124e-170, indicates that the data does not follow a normal distribution. However, the assumption of normality concerns the distribution of the residuals [Field et al. 2012]. The residuals show a normal distribution across the majority of the attributes or are slightly skewed (see Figure 2), endorsing the usage of ANOVA [Wang et al. 2022].

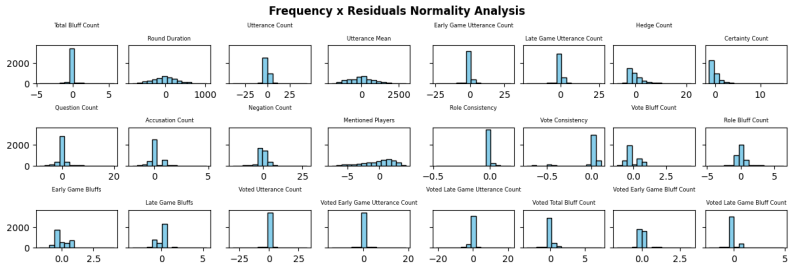

Figure 2. Residuals Normality Analysis

ANOVA Type II was used to understand the statistical relevance of each attribute over the main categorical attributes. Type II is a fit metric because the bluff types are unbalanced, even though the other attributes show a more balanced distribution [Øyvind Langsrud 2003].

Attributes with an ANOVA Type II p-value greater than 0.05 and a small F-value were classified as not significant and removed. These attributes were the round duration, the mean of the player's utterance length, the number of times the player said something indecisive or decisive, the players mentioned in sentences, and the vote consistency rate. This filtering resulted in a new relevant structured dataset with the same number of records as SD1, namely, Structured Dataset 2 (SD2)[1].

In addition, both structured datasets were represented using two methods. The first consisted of selecting a good normalization method for both numerical and categorical attributes. Thus, for numerical features, a rescaling between -1 and 1 was done by applying *StandardScaler* (SS) [de Amorim et al. 2023, Tan et al. 2018], which subtracts the mean (μ) from the value (x) and divides by the standard deviation (s) as $z = \frac{x-\mu}{s}$. SS is a fit technique for datasets with a close-to-normal distribution. For categorical features, *get_dummies* was applied to convert each value within each feature to a new

[1]Both SD1 and SD2 are available at `https://huggingface.co/datasets/KSBCode/werewolf-bluffers`

numerical binary column indicating whether that value was present in that row or not (1 or 0) [Tan et al. 2018].

The second method applied was Gower distance [Gower 1971, Ben Ali e Massmoudi 2013]. It is a distance measure that can be calculated between mixed-type attributes (categorical and numerical). It generates a similarity matrix by calculating the absolute difference between two values (x_1 and x_2) and dividing by the amplitude when numerical, as $dist = \frac{|x_1 - x_2|}{max - min}$, and as indicator for categorical attributes by assigning 1 when the value is present and 0 when it is not. It is a method fit for clustering that avoids the dummy transformations, saving memory and reducing dimensionality.

Recent studies have shown that embeddings are great for clustering activities [Cachay-Guivin 2024]. Therefore, to incorporate this aspect into the analysis, the sentences associated with SD1 and SD2 were encoded using two pre-trained sentence encoders to generate the embeddings and create the different scenarios for the experiments. The two encoders chosen are state-of-the-art, one being multilingual and the other generalist, namely, *intfloat/multilingual-e5-large* (Multi) [Wang et al. 2024], and *sentence-transformers/all-MiniLM-L6-v2* (AllMini) [Wang et al. 2020].

3.5. Module 4: Apply Clustering Algorithm

HDBSCAN* and Spectral are adopted as clustering methods. As a state-of-the-art CA, HDBSCAN* automatically determines the number of clusters in a dataset, making it well-suited for exploratory analysis. Furthermore, it combines the strengths of hierarchical and density-based approaches by handling variable-density clusters, identifying potential anomalies through outlier detection, providing soft clustering via membership probabilities, and adapting effectively to embedding-based and hybrid representations. While Spectral needs determination of the number of clusters and partitions a similarity graph instead of directly optimizing distances in the original feature space. This characteristic enables the algorithm to identify clusters with arbitrary shapes and to better capture complex behavioral relationships between players represented by heterogeneous numerical, categorical, and semantic features. Both algorithms' properties support a more informative and interpretable analysis within this work's context.

In terms of algorithm setup, the hyperparameter tuning of HDBSCAN* was performed via randomized search on a set of intervals where values fit the final dataset and the researchers' expected clusters based on the dataset knowledge, with a minimum cluster size varying from 5 to 40, minimum samples from 1 to 10, and cluster selection epsilon from 0.1 to 1.0. For Spectral, the number of clusters was determined by the best results obtained by HDBSCAN* and the expected number of clusters (2, 6, 7). Additionally, the Spectral internal strategy for assigning labels was *cluster_qr* [Damle et al. 2019] due to its better performance over *k-means* and *discretization* in both quality and speed, since it extracts labels directly from the eigenvectors.

3.6. Module 5: Evaluate Bluffers and Results

The evaluation of the clustering method was conducted through both quantitative performance and qualitative interpretability of behavioral patterns. Several strategies were used in that sense, including a statistical summary of each cluster to observe

general variation over the attributes, the cluster evaluation metrics Silhouette Score [Tan et al. 2018] and Davies-Bouldin Index [Davies e Bouldin 1979], and Scatter plotting through dimensional reduction with both Principal Component Analysis (PCA) [Jolliffe 2002] and t-distributed Stochastic Neighbor Embedding (t-SNE) [van der Maaten e Hinton 2008].

For CA, Jaccard Similarity Coefficient [Murphy 1996], Adjusted Rand Index (ARI) [Hubert e Arabie 1985], and Normalized Mutual Information (NMI) [Manning et al. 2008] were used.

During the preparation of this manuscript, the authors used ChatGPT (OpenAI) as an auxiliary tool for language refinement and text revision. The use of this tool was limited to improving readability and organization, without contributing to the scientific content, experimental design, or interpretation of results. All content was reviewed and validated by the authors.

4. Experiments

4.1. Experimental Setup

Numerous test scenarios were defined by performing different combinations among the following elements: datasets (SD1 and SD2), embedding representations (Multi and AllMini), normalization methods (SS and Gower), hyperparameter sets and CA (HDBSCAN* and Spectral). These scenarios were designed to evaluate the impact of structured features, representation strategies, embeddings, and their combinations on clustering performance. The best combination was defined through internal validation metrics and cluster interpretability, the latter being assessed by considering cluster separability in attribute summaries and semantic coherence of sentences.

4.2. Evaluation Protocol

Clustering quality was assessed using internal validation metrics and complemented with qualitative inspection of cluster structure and interpretability. The Silhouette score, Davies-Bouldin index, and the HDBSCAN* mean cluster membership probability (i.e., the confidence that each record belongs to its assigned cluster) were used to assess cluster quality, followed by statistical summaries of cluster attributes, manual review of clusters, and visual validation through dimensionality reduction using PCA and t-SNE. Additionally, Jaccard, ARI, and NMI were evaluated between HDBSCAN* and Spectral best results.

5. Results and Discussion

Among the numerous scenarios evaluated, embedding-based representations combined with SD2 achieved higher cluster quality in HDBSCAN*, while Spectral achieved better results without embedding-based representations. Table 1 presents the ten combinations that produced the best results, five of them based on HDBSCAN* and the remaining ones based on Spectral.

Concerning HDBSCAN*, the best combination that was obtained involves feature filtering (SD2), embedding representations (AllMini) and Gower, whereas the second one involves SD1 and SS. It is noteworthy that the Silhouette score of the best combination

Table 1. 10 Clustering Results Subset

N°	Algorithm	Scenario	Strategy	N° Clusters	Silhouette	Davies-Bouldin	Mean Prob.
1	HDBSCAN*	SD1	StandardScaler	2	0.679	1.403	0.992
2	HDBSCAN*	SD1 + Multi	StandardScaler	2	0.678	1.404	0.992
3	HDBSCAN*	SD1 + AllMini	StandardScaler	2	0.675	1.406	0.992
4	**HDBSCAN***	**SD2 + AllMini**	**StandardScaler**	**2**	**0.721**	**1.408**	**0.940**
5	**HDBSCAN***	**SD2 + AllMini**	**Gower**	**6**	**0.511**	**1.562**	**0.941**
6	Spectral	SD1	StandardScaler	2	0.537	1.301	N/A
7	**Spectral**	**SD2**	**StandardScaler**	**7**	**0.622**	**2.255**	**N/A**
8	Spectral	SD2 + Multi	StandardScaler	6	0.613	2.458	N/A
9	**Spectral**	**SD2**	**Gower**	**6**	**0.553**	**0.995**	**N/A**
10	Spectral	SD2 + AllMini	Gower	6	0.518	1.057	N/A

was significantly superior (about 0.05) to that of the second. However, the best combination produced from Spectral has a higher Silhouette score when applying the SS strategy to SD2, but a rather unsatisfactory Davies-Bouldin value when compared with the Gower strategy. Although Table 1 highlights results with the highest internal validation scores, they yield less interpretable and behaviorally distinct clusters than the results with lower internal validation scores also highlighted, see Figure 3.

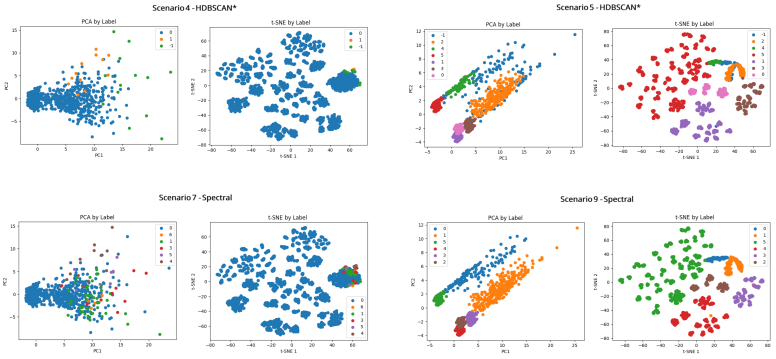

Figure 3. Highlighted Results - PCA and t-SNE Visualizations

When defining the attributes for SD1 and SD2, empirical research indicated a group of six to seven clusters for the bluffers' behavior, and the statistical summary report shows impressive results in terms of cluster interpretation when we consider scenarios 5 and 9 in comparison to 4 and 7. Scenario 4 resulted in only two clusters with high intra-cluster variability and limited interpretability, as both were dominated by the Werewolf role. The same behavior is observed in scenario 7, five out of seven clusters were dominated by the Werewolf role and all having a high number of role types per cluster. In contrast, scenario 5 and 9 produced six distinct clusters (plus one for outliers in HDBSCAN*) with clearly differentiated behavioral patterns.

The results in both HDBSCAN* and Spectral Gower scenarios showed similar clusters as proven by a 0.986 Jaccard score, 0.990 ARI, 0.965 NMI, and comparable statistical summaries. In depth, these scenarios' utterance and bluff distribution tends more to the late game, which indicates that the second half is heavier on discussion and deception. Clusters 3 and 5 are the ones where players use more communication resources such as denials, questions, and accusations; and show a higher usage of role bluffs in comparison to vote bluff. The opposite happens with cluster 1. Cluster 2 is the one with a more balanced summary across all attributes, never being the one with the highest or

lowest mean. However, the outliers in HDBSCAN* are the ones with the most distinct summaries, containing high means (μ) and standard deviations (σ), showing that usually players do not tend to exaggerate in the game by talking or bluffing too much.

In terms of role, the clusters show even more interesting behavior associated to the metrics. 5 out of the 6 clusters present lower variability of roles, showing that role is an important aspect of bluffying. It is worth noting that all non-werewolf clusters have Werewolf as the most frequently voted role, suggesting a tendency toward effective identification of deceptive behavior. See Table 2 for a subset of the statistical summary.

Table 2. Statistical Summary Subset

N°	Cluster Name	Algorithm	Utterance (Utt)		Early Game Utt		Late Game Utt		Questions		Denials		Accusations		Role Bluff		Vote Bluff		End Role	
			μ	σ	μ	σ	μ	σ	μ	σ	μ	σ	μ	σ	μ	σ	μ	σ	Distinct	Top
1	Accusatory Bluffers	HDBSCAN*	1.263	0.489	0.197	0.398	1.067	0.607	1.025	0.569	3.266	1.633	0.092	0.300	0.479	0.507	0.606	0.489	1	Hunter
		Spectral	1.263	0.489	0.197	0.398	1.067	0.607	1.025	0.569	3.266	1.633	0.092	0.300	0.479	0.507	0.606	0.489	1	Hunter
2	Moderated Deceivers	HDBSCAN*	1.504	0.911	0.248	0.535	1.256	0.831	1.256	0.693	4.050	2.053	0.146	0.390	0.531	0.507	0.519	0.503	1	Villager
		Spectral	1.504	0.911	0.248	0.535	1.256	0.831	1.256	0.693	4.050	2.053	0.146	0.390	0.531	0.507	0.519	0.503	1	Villager
3	General Bluffers	HDBSCAN*	30.665	11.529	12.011	6.237	18.655	7.521	5.960	3.821	11.676	6.320	0.763	1.051	1.759	1.473	0.482	0.822	13	Robber
		Spectral	32.110	13.391	12.902	7.392	19.208	8.260	6.172	4.095	12.448	7.416	0.831	1.130	1.825	1.698	0.525	0.845	13	Robber
4	Informational Bluffers	HDBSCAN*	1.565	1.030	0.467	0.531	1.098	1.015	1.329	0.705	4.456	2.085	0.220	0.442	0.743	0.463	0.306	0.466	3	Seer
		Spectral	1.566	0.837	0.464	0.527	1.076	0.868	1.318	0.669	4.459	2.086	0.220	0.442	0.742	0.464	0.307	0.467	2	Seer
5	Chaotic Deceivers	HDBSCAN*	28.205	9.086	11.282	5.489	16.923	5.857	5.410	2.987	9.859	5.239	0.692	0.872	1.513	1.102	0.436	0.676	1	Werewolf
		Spectral	31.391	14.036	12.947	8.138	18.444	7.818	6.113	3.838	11.179	6.557	0.874	1.185	2.205	1.838	0.523	0.908	1	Werewolf
6	Strategic Deceivers	HDBSCAN*	1.299	0.536	0.439	0.520	0.860	0.704	1.121	0.602	3.711	1.821	0.182	0.404	0.872	0.384	0.165	0.373	1	Werewolf
		Spectral	1.303	0.566	0.440	0.520	0.864	0.723	1.122	0.602	3.710	1.821	0.182	0.404	0.872	0.383	0.165	0.373	1	Werewolf
-1	Outliers	HDBSCAN*	36.598	18.008	15.780	10.227	20.818	10.005	6.970	4.776	14.144	9.141	1.106	1.416	2.583	2.341	0.667	1.024	11	Werewolf

6. Conclusion

This work demonstrated that combining feature engineering, embedding-based representations, and hierarchical density-based clustering enables the identification of meaningful behavioral patterns in conversation-driven RPGs. In particular, the use of Gower distance and embedding-based representations resulted in clusters that better captured the temporality and strategic aspects of bluffers' behavior. While for algorithms that does not handle high-dimensional data well, such as Spectral, solely using a data representation strategy that best fits the problem was enough for meaningful results. The findings suggest that bluffing is not uniformly distributed across players, but instead emerges through distinct behavioral profiles associated with communication patterns, role dynamics, and game phases. These results highlight the importance of interpretability over purely metric-based evaluation in behavioral clustering tasks.

Bluff identification relying on LLM-generated annotations and no ground-truth behavioral taxonomy existing for bluffing styles are some limitations of this study. Although prompt engineering, self-evaluation strategies, and manual inspection were employed to improve consistency, classification errors may propagate to the feature engineering stage and consequently influence the resulting behavioral clusters. This work contributes to the literature by proposing a complete pipeline for unsupervised behavioral profiling of bluffing players, integrating LLM-based bluff detection, temporal feature engineering, mixed-data representations, and clustering analysis.

Future work includes external validation through expert annotation, investigation of more clustering categories, and extending the approach to other conversation-driven game environments.

7. Acknowledgments

We would like to express our sincere gratitude to the National Council for Scientific and Technological Development (CNPq), Grant #302833/2025-0, and to Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES), for financial support.

References

Bauckhage, C., Drachen, A., e Sifa, R. (2015). Clustering game behavior data. *IEEE Transactions on Computational Intelligence and AI in Games*, 7(3):266–278.

Ben Ali, B. e Massmoudi, Y. (2013). K-means clustering based on gower similarity coefficient: A comparative study. In *2013 5th International Conference on Modeling, Simulation and Applied Optimization (ICMSAO)*, pages 1–5.

Cachay-Guivin, A. (2024). Evaluation on embeddings application for spanish automatic text clustering. *Ingeniare. Revista chilena de ingeniería*, 32.

Damle, A., Minden, V., e Ying, L. (2019). Simple, direct and efficient multi-way spectral clustering. *Information and Inference: A Journal of the IMA*, 8(1):181–203.

Davies, D. L. e Bouldin, D. W. (1979). A cluster separation measure. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-1(2):224–227.

de Amorim, L. B., Cavalcanti, G. D., e Cruz, R. M. (2023). The choice of scaling technique matters for classification performance. *Applied Soft Computing*, 133:109924.

Ding, L., Li, C., Jin, D., e Ding, S. (2024). Survey of spectral clustering based on graph theory. *Pattern Recognition*, 151:110366.

Drachen, A., Sifa, R., Bauckhage, C., e Thurau, C. (2012). Guns, swords and data: Clustering of player behavior in computer games in the wild. In *2012 IEEE Conference on Computational Intelligence and Games (CIG)*, pages 163–170.

Drachen, A., Thurau, C., Sifa, R., e Bauckhage, C. (2014). A comparison of methods for player clustering via behavioral telemetry. *CoRR*, abs/1407.3950.

Field, A., Miles, J., e Field, Z. (2012). *Discovering Statistics Using R*. SAGE Publications, 1 edition.

Games, B. (2024). Ultimate werewolf.

Gower, J. C. (1971). A general coefficient of similarity and some of its properties. *Biometrics*, 27(4):857–871.

Grande-de Prado, M., Baelo, R., García-Martín, S., e Abella-García, V. (2020). Mapping role-playing games in ibero-america: An educational review. *Sustainability*, 12(16).

Gu, J., Jiang, X., Shi, Z., Tan, H., Zhai, X., Xu, C., Li, W., Shen, Y., Ma, S., Liu, H., Wang, S., Zhang, K., Lin, Z., Zhang, B., Ni, L. M., Gao, W., Wang, Y., e Guo, J. (2026). A survey on llm-as-a-judge. *The Innovation*, 7(6):101253.

Hagendorff, T. (2024). Deception abilities emerged in large language models. *Proceedings of the National Academy of Sciences*, 121(24):e2317967121.

Harley, J. B. e Sparkman, D. (2019). Machine learning and nde: Past, present, and future. *AIP Conference Proceedings*, 2102(1):090001.

Hope, T. M. (2020). Chapter 4 - linear regression. In Mechelli, A. e Vieira, S., editors, *Machine Learning*, pages 67–81. Academic Press.

Hu, C., Zhao, Y., Wang, Z., Du, H., e Liu, J. (2024). Games for artificial intelligence research: A review and perspectives. *IEEE Transactions on Artificial Intelligence*, 5(12):5949–5968.

Hubert, L. e Arabie, P. (1985). Comparing partitions. *Journal of Classification*, 2(1):193–218.

Jaques, R. R. e Bisol, C. A. (2019). Role-playing games for common well-being in high school — notes from a study developed in southern brazil. *International Journal of Information and Education Technology*, 9(7):498–501.

Jolliffe, I. T. (2002). *Principal Component Analysis*. Springer, 2 edition.

Lai, B., Zhang, H., Liu, M., Pariani, A., Ryan, F., Jia, W., Hayati, S. A., Rehg, J., e Yang, D. (2023). Werewolf among us: Multimodal resources for modeling persuasion behaviors in social deduction games. In Rogers, A., Boyd-Graber, J., e Okazaki, N., editors, *Findings of the Association for Computational Linguistics: ACL 2023*, pages 6570–6588, Toronto, Canada. Association for Computational Linguistics.

LeCun, Y., Bengio, Y., e Hinton, G. E. (2015). Deep learning. *Nature*, 521(7553):436–444.

Malzer, C. e Baum, M. (2019). Hdbscan(ϵ): An alternative cluster extraction method for HDBSCAN. *CoRR*, abs/1911.02282.

Manning, C. D., Raghavan, P., e Schütze, H. (2008). *Introduction to Information Retrieval*. Cambridge University Press.

Marwala, T. e Hurwitz, E. (2009). A multi-agent approach to bluffing. In *Multiagent systems*. Citeseer.

Maulud, D. H. e Abdulazeez, A. M. (2020). A review on linear regression comprehensive in machine learning. *Journal of Applied Science and Technology Trends*, 1(2):140–147.

Mottaqi, M. S., Mohammadipanah, F., e Sajedi, H. (2021). Contribution of machine learning approaches in response to sars-cov-2 infection. *Informatics in Medicine Unlocked*, 23:100526.

Murphy, A. H. (1996). The finley affair: A signal event in the history of forecast verification. *Weather and Forecasting*, 11(1):3 – 20.

Perillo, J. T. e Kassin, S. M. (2011). Inside interrogation: The lie, the bluff, and false confessions. *Law and Human Behavior*, 35(4):327–337.

Rath, S., Tripathy, A., e Tripathy, A. R. (2020). Prediction of new active cases of coronavirus disease (covid-19) pandemic using multiple linear regression model. *Diabetes Metabolic Syndrome: Clinical Research Reviews*, 14(5):1467–1474.

Sun, L. (2025). slhleosun/werewolf$_gameplays$.

Tan, P.-N., Steinbach, M., Karpatne, A., e Kumar, V. (2018). *Introduction to Data Mining (2nd Edition)*. Pearson, 2nd edition.

van der Maaten, L. e Hinton, G. (2008). Visualizing data using t-sne. *Journal of Machine Learning Research*, 9:2579–2605.

Wang, L., Yang, N., Huang, X., Yang, L., Majumder, R., e Wei, F. (2024). Multilingual e5 text embeddings: A technical report.

Wang, W., Wei, F., Dong, L., Bao, H., Yang, N., e Zhou, M. (2020). Minilm: Deep self-attention distillation for task-agnostic compression of pre-trained transformers.

Wang, Y.-H., Zhang, Y.-F., Zhang, Y., Gu, Z.-F., Zhang, Z.-Y., Lin, H., e Deng, K.-J. (2022). Identification of adaptor proteins using the anova feature selection technique. *Methods*, 208:42–47.

Yoo, B. e Kim, K.-J. (2024). Finding deceivers in social context with large language models and how to find them: the case of the mafia game. *Scientific Reports*, 14(1):30946.

Øyvind Langsrud (2003). Anova for unbalanced data: Use type ii instead of type iii sums of squares. *Statistics and Computing*, 13(2):163–167.