of clusters in the reduced-dimensional space. This behavior is consistent across all tested configurations and cell types, as shown by the silhouette scores reported in Table 2, supporting the robustness of the clustering behavior across different architectural variations.

This result confirms the existence of a well-defined partition in the state space, where hidden states corresponding to sentences with the same intent are grouped into distinct regions. Furthermore, this partition can be analyzed without requiring the full-dimensional hidden state space. Instead, the structure of the state space can be captured by examining its intrinsic manifold through the top-*id* projections. Importantly, the clustering of hidden states into distinct regions based on intent is robust across different architectures and hyperparameter configurations. Regardless of the type of recurrent cell, embedding dimension, or hidden layer size, K-means clustering produces well-defined clusters, reinforcing the idea that intent-specific groupings are an inherent property of the learned state space.

7.3. Model Inference Mechanism: Sentences as Trajectories

In this section, we analyze how input sentences are processed as trajectories through the state space of a trained RNN, revealing structured and predictable behavior within the state space. Sentences evolve along paths, reaching distinct regions of the state space that can be numerically identified as clusters. These regions enable the RNN to classify sentences based on the semantic patterns captured in the final hidden states.

For each input token $\mathbf{x}_i$, the hidden state $\mathbf{h}_t$ is updated according to Equation 1, producing the next state $\mathbf{h}_{t+1}$. As a result, an input sequence $\mathbf{x}_1, \ldots, \mathbf{x}_T$ generates a corresponding sequence of hidden states $\mathbf{h}_1, \ldots, \mathbf{h}_T$, which describe a trajectory traversing the state space of the RNN. The hidden states can be projected onto the principal components of the *id*-dimensional state space, as described in Equation 3. Figure 6 (a) illustrates example trajectories for three sentences, each associated with a distinct intent. Each hidden state $\mathbf{h}_i$ is marked with a bullet, while the initial state $\mathbf{h_0}$, representing the zero-value vector before any token is injected, is represented as a black square. Lines connecting the states emphasize the sequential movement through the state space. Specific examples include: red trajectory ("*get_weather*" intent) "*What is the weather forecast for Garrison*"; brown trajectory ("*play_music*" intent) "*Play twenties on Groove Shark*" and purple trajectory (*"rate_book" intent*) "*Rate Mus of Kerbridge a one*". The corresponding input tokens are labeled next to the hidden state they generate. As tokens are processed sequentially, the orbits diverge from the origin and move towards distinct, peripheral regions of the state space.

This behavior generalizes across all intents and sentences, as shown in Figures 6 (b) and (c), respectively. Here, individual bullets have been removed for a cleaner representation, and arrows have been added to emphasize the sense of movement along the trajectories. These trajectories appear to direct the hidden states toward specific regions of the state space depending on the intent associated with the sentence. The final hidden state $\mathbf{h}_T$, located at the endpoint of each trajectory, is particularly significant as it determines the prediction of the network. Figure 6 (d) shows that the endpoints of these trajectories (i.e. the final hidden state) form distinct clusters, each corresponding to a particular intent. To numerically confirm the existence of these clusters, we applied a K-means algorithm (with 7 clusters) to the final hidden states. The results for different RNN configurations are summarized in Table 2. Across all tested configurations and cell types, silhouette scores greater than 0.75 confirm the presence of as many clusters as intents, demonstrating that this clustering behavior is robust and consistent regardless of architectural variations.

As shown in Figure 2, for prediction purposes, intermediate states are discarded, and only the final hidden state $\mathbf{h}_T$ is considered. Given a sentence, the logits for each class are computed by projecting $\mathbf{h}_T$ onto the readout vectors $\mathbf{r}_i$, corresponding to the rows of the readout matrix $\mathbf{W}$ as follows:

$$\mathbf{y} = \mathbf{y}_T = \mathbf{W}\mathbf{h}_T = [\mathbf{r}_1 | \ldots | \mathbf{r}_n]^T \mathbf{h}_T \tag{4}$$

This formulation enables the RNN to classify sentences based on the semantic patterns encoded in the final hidden states. For a sentence with true intent I and final state $\mathbf{h}_T$, correct prediction requires that $\mathbf{r}_I^T \mathbf{h}_T$ be greater than $\mathbf{r}_i^T \mathbf{h}_T$ for any $i \neq I$. To achieve this, the readout vector $\mathbf{r}_I$ must align as closely as possible with the final states $\mathbf{h}_{TI}$ of the cluster associated with intent I. Cosine similarity is a widely used measure of vector alignment [11], which captures the cosine of the angle between vectors. The values range from -1 (opposite direction) to 1 (perfect alignment), with 0 indicating orthogonality. Figure 6 (f) shows the cosine similarity between all pairs (r_i, c_j) for a trained GRU(emb: 16,hid: 16). Each r_i aligns closely with a single centroid, achieving similarity values exceeding 0.9. The visualization

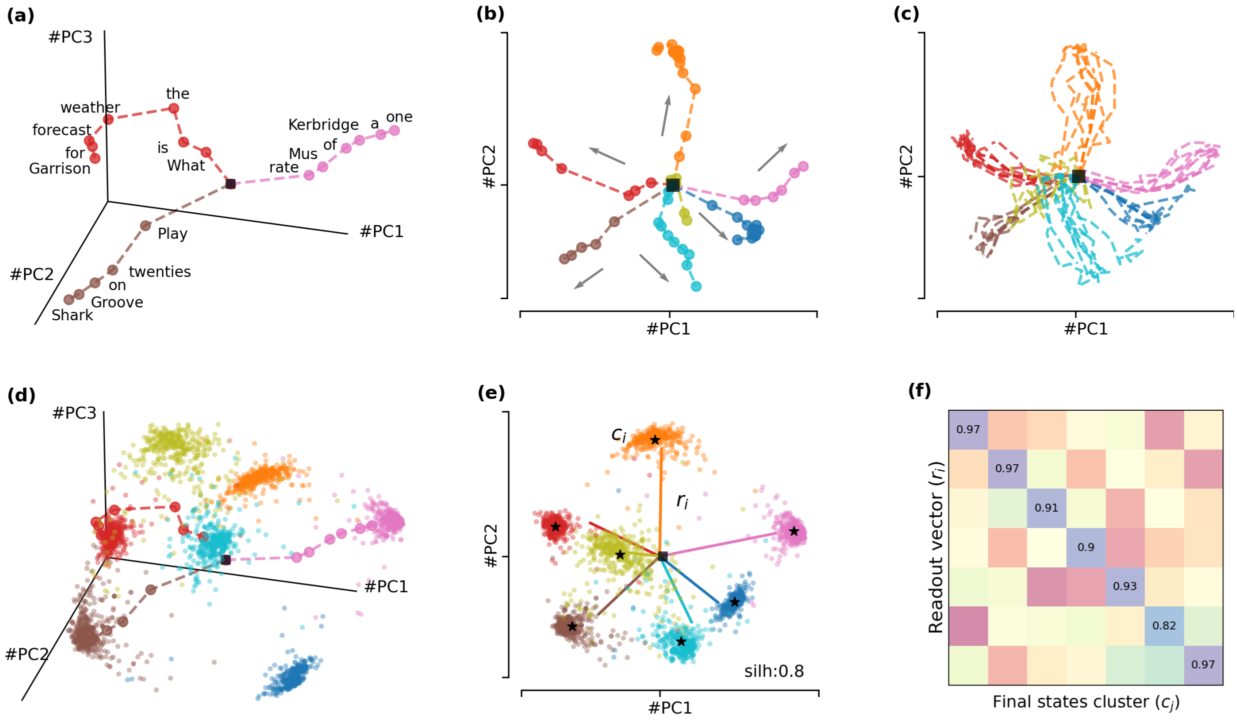

Fig. 6. State space visualizations of a GRU(emb:16,hid:16) trained on the SNIPS dataset, projected onto the top principal components. **(a)** Example trajectories for three individual sentences. Each point represents a hidden state corresponding to a token (labeled) in the sentence, a black square marks the initial hidden state. Dashed lines show transitions between states. **(b)** A representative trajectory for each intent, with arrows indicating the direction of movement. **(c)** Overlay of multiple trajectories sharing the same intent. **(d)** Three example trajectories embedded in the 3D state space, superimposed on the final hidden states of all test sentences. **(e)** Clusters of final states with centroids marked as black stars. Readout vectors (r_i) from the output layer are color-coded to match their corresponding clusters (c_i). The silhouette score for the clustering is 0.8. **(f)** Heatmap showing cosine similarities between final state cluster centroids and readout vectors. Only values above 0.5 are displayed.

in Figure 6 (e) confirms that the cluster centroids align with the readout vectors of the output layer, as evidenced by the cosine similarity heatmap in Figure 6 (f). This alignment illustrates how the RNN organizes its state space to facilitate classification, guiding sentence trajectories toward decision regions defined by the output layer. The predicted intent corresponds to the index i with the highest scalar value $\mathbf{r}_i^T \mathbf{h}_T$. In Figure 6 (e), the final hidden states are projected onto their principal components and colored by intent. The readout vectors r_i are also projected into this space using the same color scheme.

7.4. Cluster Spatial Characteristics

In this section, we analyze the spatial arrangements of the final hidden state clusters in the RNN state space. The results reveal a robust structure across different configurations, with cluster centroids located approximately equidistant from the initial hidden state and clusters remaining compact, enabling effective intent classification.

For each network configuration, we calculated the Euclidean distances d_i between the centroid of the i-th cluster and the initial hidden state $\mathbf{h_0}$. The distribution of these distances for two different GRU configurations is presented in Figure 7 (a). In all configurations, the standard deviation $\sigma(d)$ of these distances is significantly smaller than their mean $\bar{d}$, indicating low variability. This consistency suggests that cluster centroids are roughly equidistant from the initial state. Figure 7 (b) shows that the mean centroid distance $\bar{d}$ increases as the size of the hidden layer grows, while remaining largely unaffected by changes in embedding size. This trend reflects the network's ability to distribute information more broadly in the state space as its hidden layer capacity increases. The boxplot in Figure 7 (c) further highlights the low variability in centroid distances between different hidden layer sizes.

In addition, we measured the cluster radii R_i obtained as the average Euclidean distance between all points in the i-th cluster and its associated centroid c_i. The distribution of these radii for two distinct GRU configurations is shown in Figure 7 (d). Like centroid distances, the cluster radii increase with the growth of the hidden layer size, as illustrated in Figure 7 (e). As centroids move from the origin with larger hidden layers, the clusters expand proportionally. However, compared to centroid distances, the cluster radii presents a greater variability, as shown in Figure 7 (f). This greater dispersion reflects inherent differences in how hidden states from different intents are distributed within each cluster. As shown in Table 2, this behavior of distances and radii is consistent across a variety of configurations and cell types, further validating the robustness of the observed patterns in different network architectures.

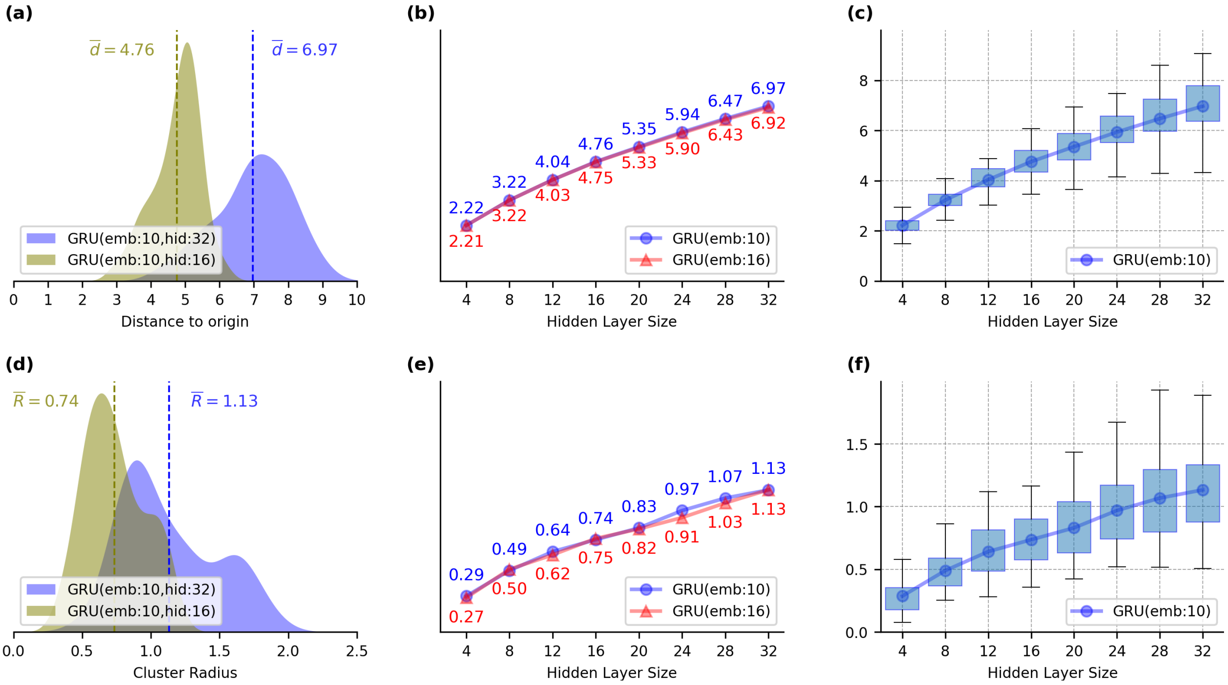

Fig. 7. Analysis of the spatial characteristics of final hidden states clusters for different GRU configurations. **(a)** Distribution of the distances (d) from each cluster centroid to the initial hidden state ($\mathbf{h_0}$), for each configuration. **(b)** Mean centroid distances $\bar{d}$ as a function of hidden layer size for two different embedding sizes. **(c)** Variability of centroid distances for GRU with embedding = 10 across hidden layer sizes.**(d)** Distribution of the clusters radii (R) for both configurations. **(e)** Mean cluster radii ($\bar{R}$) as a function of hidden layer size for two different embedding sizes. **(f)** Variability of cluster radii for GRU with embedding=10 across hidden layer sizes.

7.5. *Fixed Point Structure*

In the previous section, we show how sentences traverse a low-dimensional state space, guided by the RNN toward final state clusters. This structured behavior suggests the presence of an underlying fixed point topology in the network dynamics. To investigate this, we use the first-order approximation of the RNN dynamics [29] around an expansion point $(\mathbf{h}_e, \mathbf{x}_e)$ expressed as:

$$\mathbf{h}_t \approx \mathbf{F}(\mathbf{h}_e, \mathbf{x}_e) + \mathbf{J}^{rec}\mathbf{F}|_{(\mathbf{h}_e,\mathbf{x}_e)}\Delta\mathbf{h}_{t-1} + \mathbf{J}^{inp}\mathbf{F}|_{(\mathbf{h}_e,\mathbf{x}_e)}\Delta\mathbf{x}_t \tag{5}$$

where $\Delta\mathbf{h}_{t-1} = \mathbf{h}_{t-1} - \mathbf{h}_e$, $\Delta\mathbf{x}_t = \mathbf{x}_t - \mathbf{x}_e$ and $\{\mathbf{J}^{rec}\mathbf{F}, \mathbf{J}^{inp}\mathbf{F}\}$ are the Jacobian matrices of the update function $\mathbf{F}$ at the expansion point. Specifically, the *recurrent Jacobian* $\mathbf{J}^{rec}\mathbf{F}$ captures the local dynamics of the recurrent structure,