Human-Centric AI Training: Leveraging Feedback Through a Novel UI

Jingbo Yan[*1 *2] Seiji Yamada[*2 *1]

[*1] SOKENDAI [*2] National Institute of Informatics

Human-centered research has garnered significant insights and feedback from active human participants. However, effectively handling and utilizing this feedback to train AI agents remains a challenge. To ensure optimal performance in downstream tasks such as image classification or segmentation and to efficiently distill human input for large dataset tasks, We propose a novel UI designed for the rapid collection of feedback pertaining to clustering on a large scale. This feedback is subsequently incorporated into an end-to-end training model, facilitating the transfer of human metrics to the AI system. Unlike typical accuracy-oriented models, our approach emphasizes the interpretability of decision-making processes. It can provide insights into the prediction results obtained after the reference selective prototype, offering a unique perspective on the deep model controlling.

1. Introduction

The majority of existing research has concentrated on interactive interfaces akin to LLM designed for question-answering. This involves iterative rounds aimed at preserving context and correcting logical aspects to arrive at a conclusive result. We designed an interactive interface for users to observe and revise cluster results in four rounds, effectively handling must-link and cannot-link constraints for inter-cluster and intra-cluster distance measures by incorporating the COP-KMeans [5] algorithm. In this way, it is more related to topic modeling approaches to reading and classification than to question answering process. Additionally, after each iteration, the algorithm diffuses constraints to other images within the group, proving effective for large datasets.

Given the novel label collection approach, where each user finally obtains decent clustering results after providing feedback, we depart from literature like [2, 3], which deals with multi-agent cooperation problem. Instead, we adopt a majority voting system to determine the official training dataset containing human insights.

The deep learning models play a dual role in feature learning and task transfer. In our quest for a robust model, we focus on delving into feature learning methods, with a special emphasis on the layers within deep network architectures. When using this dataset for general image model training, resampling is a common practice. However, estimating importance weights for high-dimensional data is challenging. We employ the end-to-end Sinkhorn autoencoder technique. The VAE[4] infers the distribution of the latent space by generating images and subsequently aligns them with input images to approximate the true underlying distribution. In the latent space, the clustering outcomes are treated as groups of prototypes. Leveraging the concept of the transport problem, we compute the average for each group and randomly select prototypes, ensuring their similarity to the samples. Throughout the training process, the model learns to utilize these prototypes, facilitating the transfer of human metrics to the AI agent in an effective manner.

Contact: Jingbo Yan, NII, 2-1-2 Hitotsubashi, Chiyoda-ku, Tokyo 101-8430, JAPAN, enseiha@nii.ac.jp

2. Related Work

In terms of architecture, a VAE[4] is a pure generative model, meaning it cannot control what it generates. Leveraging optimal transport theory for unsupervised learning tasks, the Sinkhorn Autoencoder is distinguished by its ability to efficiently integrate optimal transport-based regularization, promoting structured latent spaces and enhancing interpretability in unsupervised learning. On the other hand, a conditional VAE (CVAE[6, 11]) can generate based on given labels, making it suitable for our COP-Kmeans results. Another end-to-end solution is deep clustering coupled with gumbel[11] to optimize discrete clusters. While it lacks interpretability, as the learning of visual words is not explicitly visible during vision task training.

3. Proposed Method

The model integrates an AI agent within the drag-and-drop experiment. It encompasses a demonstration phase to observe clustering results, a comparison phase where users engage in image reorganization interactions, and a component involving the calculation of must-link and cannot-link constraints through a diffusion algorithm. This comprehensive approach facilitates the seamless integration of human feedback into the AI system, fostering a synergistic interaction between the user and the intelligent agent.

Human insight learning is facilitated through the optimal transport problem, enhancing the agent's intelligence. The core of the agent's intelligence lies in the utilization of cluster prototypes. This comprehensive framework enables a seamless integration of human feedback and learning in the AI system, creating a synergistic interaction between the user and the intelligent agent.

3.1 GUI Design and Data Collection

To initiate our process, we apply watershed segmentation to images, converting them into rectangles and then dividing them into small, overlapping patches (each containing less than nine patches). This segmentation yields a multi-resolution representation spanning pixel, concept,

background, and object levels—an integral step for simulating neural network feature learning and aiding human comprehension. The utilization of overlapping patches enhances consistency and eliminates information gaps in the human-machine interface.

3.2 Preparation

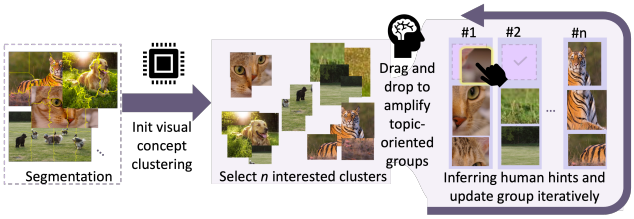

Figure 1: GUI design

The transformation of free-shaped segments into rectangles addresses the limitations of the original segmentation, which primarily captures texture, color, and pattern features. The resulting clustered form rectifies contextual information deficiencies, assisting users in distinguishing implied topics within each group. This action is especially crucial as the original segmentation often only captures texture, colour, and pattern features, such as animal fur, which can make it difficult for humans to distinguish the differences between each clustered group when organized in our user interface, as shown in Figure 1. The incorporation of a drag-and-drop interface empowers participants to rearrange image sets, thereby improving thematic coherence.

Despite these advancements, ambiguous relations between pairwise patches persist. To address this challenge, we design an algorithm for collecting constraints before implementing COP-Kmeans. Users contribute set-wise constraints, which are then simplified, and pairwise constraints are computed for COP-Kmeans. Additionally, we develop a constraints induction algorithm specifically for running semi-supervised COP-Kmeans.

3.3 Human-Aligned Variational Autoencoders

Given that participants may introduce subjective biases, unifying crowdsourced outcomes becomes challenging. In subsequent experiments, we exclusively involve the designer's input through the interactive system to eliminate unintended biases.

4. Experiments and Results

4.1 Dataset

We utilized the PascalVOC 2010 [1] training data and supplemented it with 135 images from the COCO [9] datasets. These additional images were labeled for various food items such as hot dog, pizza, sandwich, broccoli, banana, orange, apple, carrot, donut, and cake. This was done to enhance the dataset and create a food superclass. After segmentation, the dataset consisted of a total of 9,305 images. These data were processed using CLIP and VGG16 [10] to generate embeddings, which were subsequently transformed into a 2D space using t-SNE [8]. We initialized it with 20 groups using K-means. Due to the large number of images, only the 20 points closest to the center are displayed for each group.

To gather a wider range of data, we conducted this survey online. All participants were recruited from Yahoo! Crowd Sourcing and received a reward of 90 yen (approximately 60 cents). Through online experiments, we collected relevant questionnaires on the constraints used in COP-Kmeans , as well as comparative questionnaires regarding the thematic differences between groups before and after COP-Kmeans, and the satisfaction with the final expression of clusters.

4.2 User Operation Process

- Observe situations of zooming in and out and clustering. When the mouse hovers over an image, a border surrounds the entire group.

- After observing, a questionnaire 2 about the current representation of the group will pop up. Rate the initial impression of 20 clusters using a 7-point likert scale in the questionnaire.

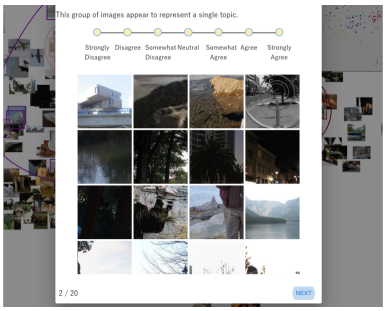

Figure 2: UI of clusters topic representation rating

- Select five groups that have a high degree of overlap. Implement shows in Figure 3.

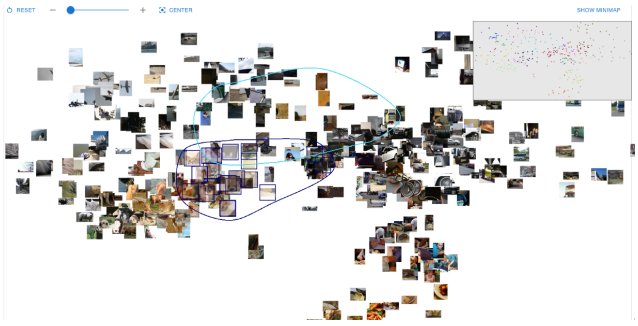

Figure 3: The clustering display webpage, users can zoom in & out.

- The five selected groups are displayed in columns, and users need to adjust the grouping of images by dragging and dropping. Each group displays 100 images that are closest to their respective centers. The backend service calculates pairwise constraints based on user's drag and drop actions. The constraints are then used to update the clustering using the COPKmeans

algorithm. The result is 20 new groups, each containing 20 images that are the nearest neighbors to the new center.

- Repeat the process of selecting groups and organizing images by dragging and dropping for four rounds.
- Conducting questionnaires, including one focused on determining whether the images in each group are topic-oriented, meaning they can be described by only one or two words. A similar question to the previous one pops up initially. It also encompasses inquiries about final clustering performance. Furthermore, there is a questionnaire that addresses constraints, data comprehension, and other critical metrics for Explainable Artificial Intelligence (XAI).

4.3 User Experiment Results

For all users finally finish all task, we recruited participants with 18∼99 age limitation. we recruited 91 participants; there were 79 males and 12 females ranging in age from 19 to 72 years for an average of 43.93 (SD = 10.99).

In this context, we've observed that if the initial topic representation falls short, the entire process requires more iterations, which can make user engagement more challenging and potentially lead to dropouts. Consequently, we chose to label and cluster the data using CLIP instead of directly applying k-means to the initial clustering. Despite the results indicating a decline in the quality of topic representation, it is important to note that this is not solely attributed to frustrations with the clustering update algorithm. Users consistently express agreement with every update. More specifically, 54 participants reported a significant increase in confidence, while 26 participants believed that it consistently happened when asked the question, 'How many rounds of grouping, after drag & drop, led to meeting your expectations?' Moreover, the comparison between initial and final clustering results also supports that this ultimately leads to improved clustering.

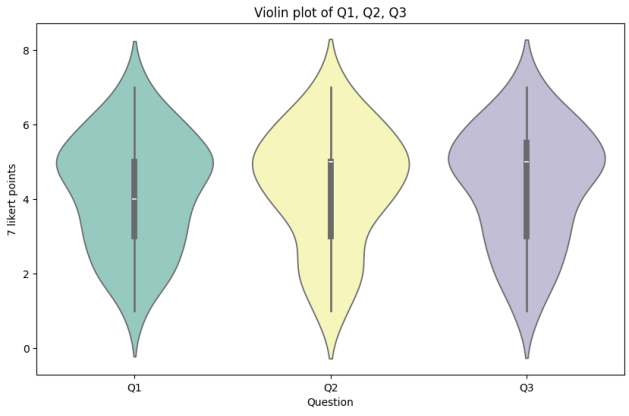

Figure 4: This shows how user suppose clustering result, Q1~Q3 describe in 4.3.

Q1~Q3: Questions after completing all four rounds of clustering updates is designed below.

- Do you think the final grouping is easier to understand than the initial grouping?
- Do you think the final grouping can classify better than the initial grouping?
- Does the final grouping result match your expectations and intuition more than the initial grouping result?

4.4 Model Training Results

4.4.1 Implement Details

The model is a VAE-U-Net liked encoder-decoder5 structure based on the ResNet architecture.

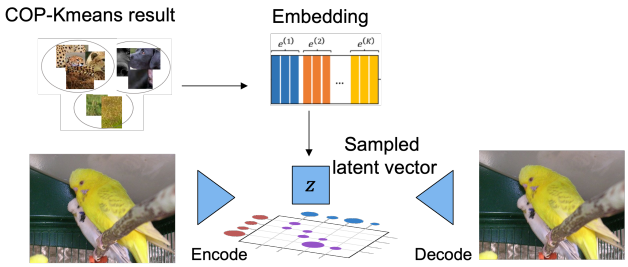

Figure 5: It starts with the COP-Kmeans result, showing clusters of images. These clusters are then transformed into an embedding, represented by a bar of colors. The embedding leads to a sampled latent vector 'Z', which is then encoded and decoded to reconstruct an image, as shown by the parrot's images at the bottom. The process demonstrates the encoding of image features into a lower-dimensional space and their subsequent reconstruction.

4.4.2 Quantitative Results

This Figure 6 shows a plot of the training loss for a Variational Autoencoder (VAE) model, measured by the Normalized Mean Squared Error (NMSE) and plotted on a logarithmic scale. The y-axis represents the NMSE on a log scale, indicating the error between the reconstructed output and the original input, while the x-axis represents the number of iterations or epochs through the training data.

The plot illustrates an initial sharp decrease in NMSE, indicating rapid reduction of reconstruction error. As training advances, NMSE diminishes at a slower pace, typical during convergence. Fluctuations arise due to optimization algorithm stochasticity, and "Smoothed NMSE" suggests the plot underwent smoothing for a clearer trend depiction.

4.5 Qualitative Results

Figure 7 depicts a visual comparison of data clustering before and after a training process. In the upper part of the image, we see a scatter of various small images grouped together but not distinctly separated. The images are connected by lines of different colors, which might represent different relationships or categories. The layout seems somewhat disorganized, indicating that this is the state of the data before training. In the lower part of the image, the same small images are now more clearly grouped into distinct clusters, each encircled by colored outlines. The clusters are more defined and separated from each other, sug-

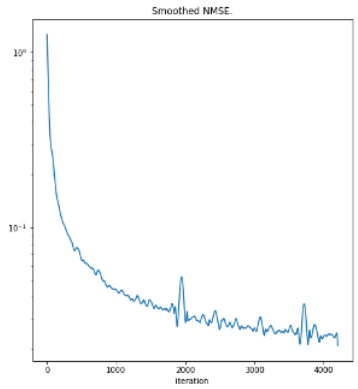

Figure 6: The overall trend in the plot indicates an improvement in the model's performance over time, eventually reaching stability.

gesting that the training process has successfully categorized the images into groups based on their features or similarities.

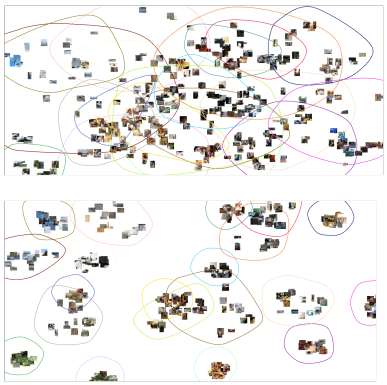

Figure 7: It demonstrates the effect of a training algorithm on organizing and classifying visual data.

5. Conclusion and Future Work

Our approach seamlessly integrates human-machine feature learning, enabling a comprehensive evaluation of interpretability and various machine learning metrics via the loss function. The architecture operates without relying on semantic labels. Nonetheless, the controlled sample distribution in the latent space strongly suggests user thinking signals. Beyond revealing patterns in user behavior during drag-and-drop interactions, we delve into the semantic structure of image data, resulting in interpretable visual outcomes.

To refine our model, we aim for a more direct representation of the interaction between user-labeled data and the training data flow within the learning model. Emphasizing the distinguishability of the data distribution in the final explanation is paramount for further improvement. Additionally, we are planning to collect diverse user feedback through crowdsourcing, enhancing the exploration of human tags and refining the presentation of final explanations.

References

[1] Everingham, M., et al. (2010). The Pascal Visual Object Classes (VOC) Challenge. *International Journal of Computer Vision*, 88, 303-338.

[2] Chaimontree, S., Atkinson, K., & Coenen, F. (2012). A multi-agent based approach to clustering: Harnessing the power of agents. In *Agents and Data Mining Interaction: 7th International Workshop on Agents and Data Mining Interation, ADMI 2011, Taipei, Taiwan, May 2-6, 2011, Revised Selected Papers, 7*, 16-29. Springer Berlin Heidelberg.

[3] Jubair, M. A., Mostafa, S. A., Mustapha, A., et al. (2022). A Multi-Agent K-Means Algorithm for Improved Parallel Data Clustering. *JOIV: International Journal on Informatics Visualization*, 6(1-2), 145-150.

[4] Kingma, D. P., & Welling, M. (2013). Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*.

[5] Bradley, P. S., Bennett, K. P., & Demiriz, A. (2000). Constrained k-means clustering. *Microsoft Research, Redmond*, 20(0), 0.

[6] Kingma, D. P., Mohamed, S., Jimenez Rezende, D., and Welling, M. (2014). Semi-supervised learning with deep generative models. In *Advances in Neural Information Processing Systems, 27*.

[7] Shen, Y., Shen, Z., Wang, M., Qin, J., Torr, P., and Shao, L. (2021). You never cluster alone. In *Advances in Neural Information Processing Systems, 34*, 27734–27746.

[8] Van der Maaten, L., and Hinton, G. (2008). Visualizing data using t-SNE. *Journal of Machine Learning Research*, 9(11).

[9] Lin, T. Y., Maire, M., Belongie, S., et al. Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V*, 740-755. Springer International Publishing, 2014.

[10] Simonyan, K., and Zisserman, A. (2014). Very Deep Convolutional Networks for Large-Scale Image Recognition. *arXiv preprint arXiv:1409.1556*.

[11] Shen, Y., Shen, Z., Wang, M., Qin, J., Torr, P., & Shao, L. (2021). You never cluster alone. In *Advances in Neural Information Processing Systems, 34*, 27734–27746.