[Supplementary] SEMA: Semantic Attention for Capturing Long-Range Dependencies in Egocentric Lifelogs

Pravin Nagar[1] K.N Ajay Shastry[2] Jayesh Chaudhari[2] Chetan Arora[2]
[1]University of Maryland, College Park
[2]Indian Institute of Technology, Delhi

In this Supplementary Material, we provide the following details omitted in the main text:

- Section 1: Mathematical Formulation of `SEMA`
- Section 2: Generating Pseudo-labels
- Section 3: Qualitative analysis of the *UTE* and *Epic Kitchens* datasets
- Section 4: Confusion Matrix
- Section 5: Attention Matrix Visualization
- Section 6: Details of *Epic Kitchens* Dataset
- Section 7: Annotation Procedure
- Section 8: Evaluation Metrics
- Section 9: Hardware Requirements

1. Mathematical Formulation of `SEMA`

`Transformer` network [11] proposed the self-attention to model the global dependencies in long sequences. Once the input sequence $\mathbf{X}$ of length N is linearly projected as query $\mathbf{Q} = \{\mathbf{q}_i \mid \mathbf{q}_i \in \mathbb{R}^m, i \in [N]\}$, key $\mathbf{K} = \{\mathbf{k}_i \mid \mathbf{k}_i \in \mathbb{R}^m, i \in [N]\}$, and value $\mathbf{V} = \{\mathbf{v}_i \mid \mathbf{v}_i \in \mathbb{R}^m, i \in [N]\}$, where m is the query, key, and value dimensions, then the self-attention mechanism is given as follows:

$$\text{Att}(\mathbf{Q}, \mathbf{K}, \mathbf{V}) = \mathbf{A}_{N\times N}\mathbf{V}_{N\times m}, \quad \mathbf{A}_{N\times N} = \text{softmax}\left(\frac{\mathbf{Q}_{N\times m}\mathbf{K}^T_{N\times m}}{\sqrt{m}}\right), \tag{1}$$

Here $\mathbf{A} \in \mathbb{R}^{N\times N}$ is the *attention matrix* and the element (i, j) is the dot product of the row i of $\mathbf{Q}$, with row j of $\mathbf{K}$. We can equivalently denote it as $\mathbf{A}(i, j) = \kappa(\mathbf{q}_i, \mathbf{k}_j)$, where $\kappa : \mathbb{R}^m \times \mathbb{R}^m \to \mathbb{R}^+$ is the kernel function.

The above style of learning an attention matrix is not scalable for extremely long egocentric lifelogs due to quadratic complexity with respect to number of samples, N, and the feature dimension, m. Kernel approximation is a powerful technique to make kernel methods scalable by projecting the input features into a new space where dot products approximate the kernel well. Formally, given a kernel κ, kernel approximation methods seek to find a nonlinear transformation $\phi : \mathbb{R}^m \to \mathbb{R}^k$, for any $q_i, k_j \in \mathbb{R}^m$

$$\kappa(\mathbf{q}_i, \mathbf{k}_j) = \phi(\mathbf{q}_i^\top)\phi(\mathbf{k}_j). \tag{2}$$

1.1. Mathematical Results

We first prove the following two mathematical results before using them in our formulation.

Lemma 1. *For a random vector $w \in \mathbb{R}^m$ sampled from a Gaussian distribution with zero mean and identity covariance matrix (I_m), and vectors $x, y \in \mathbb{R}^m$, we have:*

$$exp\left(\frac{\|x+y\|^2}{2}\right) = \mathbb{E}_{w\sim\mathcal{N}(0,I_m)} exp(w^T(x+y)) \tag{3}$$

Proof.

$$\exp\left(\frac{\|x+y\|^2}{2}\right) = \exp\left(\frac{\|x+y\|^2}{2}\right)\cdot 1 \tag{4}$$

$$= \exp\left(\frac{\|x+y\|^2}{2}\right)\frac{1}{(2\pi)^{m/2}}\int \exp\left(\frac{-\|w-(x+y)\|^2}{2}\right)dw \tag{5}$$

Since w is a Gaussian distributed vector in $\mathbb{R}^m$ with zero mean and identity covariance matrix, the second term represents the total probability and hence should be 1.

$$\exp\left(\frac{\|x+y\|^2}{2}\right) = \exp\left(\frac{\|x+y\|^2}{2}\right)(2\pi)^{-m/2}\int \exp\left(\frac{-\|w-(x+y)\|^2}{2}\right)dw \tag{6}$$

$$= (2\pi)^{-m/2}\int \exp\left(\frac{\|x+y\|^2-\|w-(x+y)\|^2}{2}\right)dw \tag{7}$$

$$= (2\pi)^{-m/2}\int \exp\left(\frac{\|x+y\|^2-(w^Tw+\|x+y\|^2-2w^T(x+y))}{2}\right)dw \tag{8}$$

$$= (2\pi)^{-m/2}\int \exp\left(\frac{-(w^Tw-2w^T(x+y))}{2}\right)dw \tag{9}$$

$$= (2\pi)^{-m/2}\int \exp\left(\frac{-\|w\|^2}{2}\right)\exp(w^T(x+y))dw \tag{10}$$

$$= \mathbb{E}_{w\sim\mathcal{N}(0,I_m)}\exp(w^T(x+y)) \tag{11}$$

Hence proved. □

Lemma 2. *For $x, y \in \mathbb{R}^m$, we have: $exp(x^Ty) = \kappa(x,y) = \phi(x)\phi(y)$, where:*

$$\phi(x) = \mathbb{E}_{w\sim\mathcal{N}(0,I_m)}\left[exp\left(-\frac{\|x\|^2}{2}\right)exp(w^Tx)\right], \tag{12}$$

$$\phi(y) = \mathbb{E}_{w\sim\mathcal{N}(0,I_m)}\left[exp\left(-\frac{\|y\|^2}{2}\right)exp(w^Ty)\right], \tag{13}$$

and w is a Gaussian distributed vector in $\mathbb{R}^m$ with zero mean and identity covariance matrix (I_m).

Proof.

$$\exp(x^Ty) = \exp\left(\frac{1}{2}(-x^Tx-y^Ty+x^Tx+y^Ty+x^Ty+y^Tx)\right) \tag{14}$$

$$= \exp\left(\frac{1}{2}(-\|x\|^2-\|y\|^2+(x+y)^T(x+y)\right) \tag{15}$$

$$= \exp\left(\frac{1}{2}(-\|x\|^2-\|y\|^2+\|x+y\|^2)\right) \tag{16}$$

$$= \exp\left(\frac{-\|x\|^2-\|y\|^2}{2}\right)\exp\left(\frac{\|x+y\|^2}{2}\right) \tag{17}$$

Using Lemma 1 to replace second term in the R.H.S.

$$= \exp\left(\frac{-\|x\|^2 - \|y\|^2}{2}\right)\mathbb{E}_{w\sim\mathcal{N}(0,I_d)}\exp(w^T(x+y)) \tag{18}$$

$$= \mathbb{E}_{w\sim\mathcal{N}(0,I_m)}\left[\exp\left(w^T x - \frac{\|x\|}{2}^2\right)\exp\left(w^T y - \frac{\|y\|}{2}^2\right)\right] \tag{19}$$

$$= \phi(x)\phi(y) \tag{20}$$

where $\phi(x)$ and $\phi(y)$ are as given by Equations 12 and 13 respectively. Hence proved. □

1.2. Softmax Kernel Approximation using Semantic Kernel

We can use Lemma 2 to write the the attention matrix $\mathbf{A}$ as *softmax-kernel* as follows:

$$\mathbf{A}(x,y) = \exp(x^\top y) = \kappa(x,y) = \phi(x)\phi(y), \tag{21}$$

where we have ignored the scaling factor of softmax. We have also ignored $\sqrt{m}$-normalization which can be equivalently done by normalizing query and key matrices accordingly.

Instead of fixed random Fourier feature transform using random vector w as proposed in [2,3,8] to approximate the *kernel* $\kappa(x,y)$, we use representative frames ($\mathbf{R}$). The proposed semantic kernel (ϕ_{sem}) defined as below projects the data into a semantically meaningful space:

$$\phi_{\text{sem}}(x) = \sum_{R_i\in\mathbf{R}} \exp\left(-\frac{\|x\|^2}{2}\right)\exp(R_i^T x), \tag{22}$$

where $\mathbf{Q} \overset{iid}{\sim} \mathcal{D}$ (a standard normalized input distribution) and $\mathbf{R} \in \mathbb{R}^{k\times m}$, $\mathbf{R} \subset \mathbf{Q}$. Here, we pretend that the feature vectors of representative frames are sampled from zero mean, unit covariance Gaussian. Intuitively, the semantic kernel reduces the rank of attention matrix from N to k by projecting into the sapce of representative frames.

Now we compute $\mathbf{Q}' = \phi_{sem}(\mathbf{Q})$ and $\mathbf{K}' = \phi_{sem}(\mathbf{K})$, where $\mathbf{Q}', \mathbf{K}'^\top \in \mathbb{R}^{N\times k}$ are the factorization of attention matrix $\mathbf{A}$ and *exp* is applied element-wise. With this kernel trick, we can change the order of multiplication of query $\mathbf{Q}'$, key $\mathbf{K}'$ and value vectors $\mathbf{V}$.

$$\widehat{\text{Att}}_{sem}(\mathbf{Q},\mathbf{K},\mathbf{V}) = \mathbf{Q}'(\mathbf{K}'.\mathbf{V}) \tag{23}$$

This multiplication is characterized by the time complexity of $\mathcal{O}(Nkm)$ and space complexity of $\mathcal{O}(Nk + Nm + km)$ compared to $\mathcal{O}(N^2 + Nm)$ and $\mathcal{O}(N^2 m)$ of the self-attention [11] and allows us to scale it to long egocentric sequences.

2. Generating Pseudo-labels

For clustering, we use the core-set algorithm [9] to generate c-medoid indices using the latest embeddings generated from `SEMAFormer` for each sample. The core-set algorithm is an efficient approximation of the k-center problem [9]. These medoids are aligned/matched to the previously generated medoids, and medoids memory (comprises indices of medoids) is updated. Once the medoids memory is updated, the pseudo labels ($\widetilde{\mathbf{y}}$) are generated with the current embedding and the latest medoids. To initialize the medoids memory, we apply the core-set for input features.

3. Qualitative analysis of *UTE* and *Epic Kitchens* datasets

Similar to Fig. 2 in the main section, we also visualize the results obtained by `SEMA` for the *Epic Kitchens* and 'P04' sequence of *UTE* datasets in Fig. 1 and Fig. 2, respectively. Kindly refer to the figures for details.

4. Confusion Matrix

The confusion matrix in Fig. 3 validates the efficacy of the proposed framework and demonstrates that inter-class confusion is marginal for most of the activity patterns. However, due to high visual similarity, we perform poorly in a few classes. For example, 'Food at round table' confuses with 'walking in lab and chitchatting' because the two activities are adjacent and often confuses at boundaries, and 'in the classroom' confuses with 'walking in lab and chitchatting.'

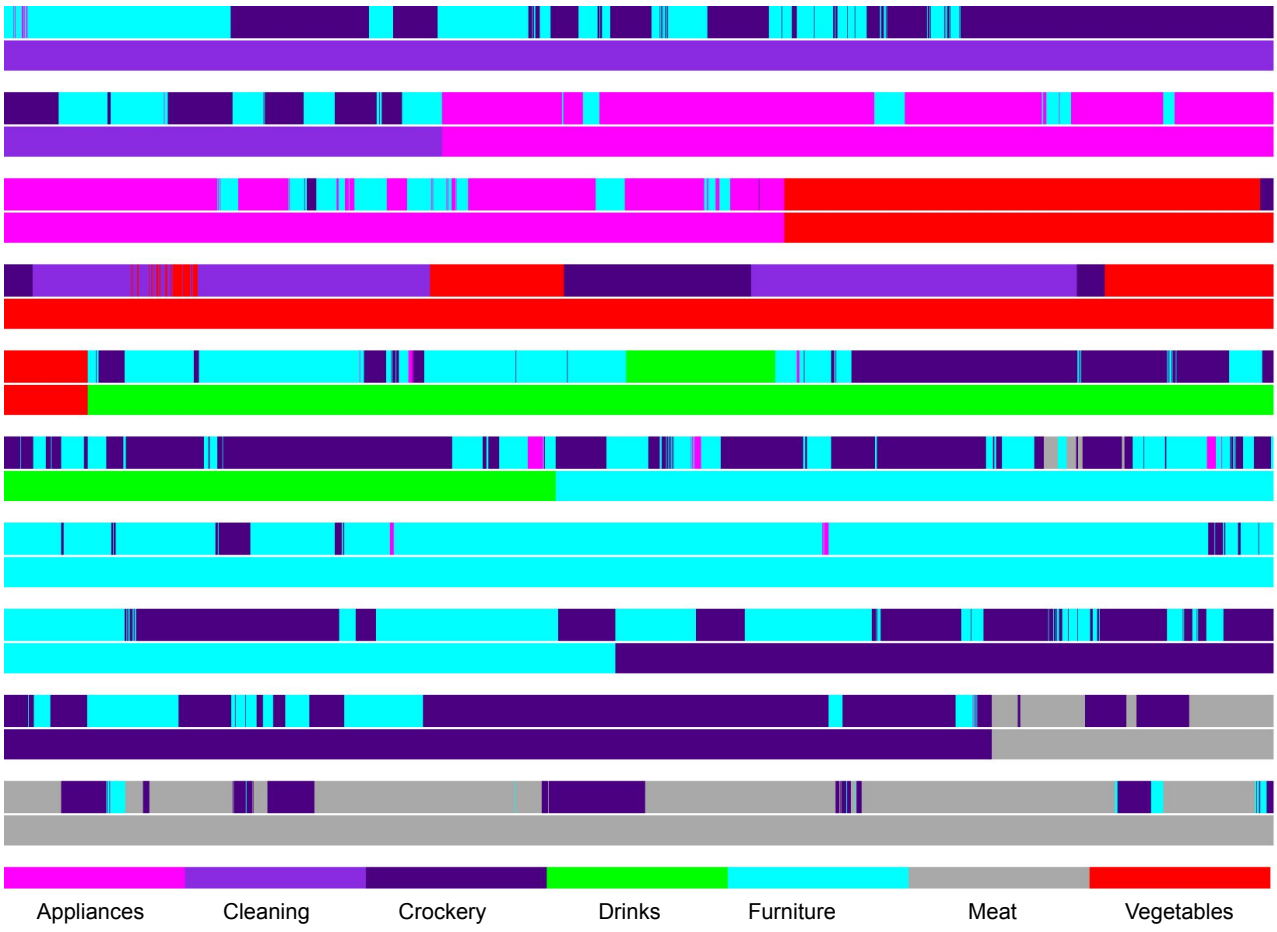

Figure 1. The figure demonstrates the visualization of a comparison between the predicted class and ground truth for *Epic Kitchens* dataset (for better visualization, we have divided the long sequence into 2000 frames for each row). The SEMA performs reasonably well even though most category classes overlap in the *Epic Kitchens* sequence. The 'appliances', 'crockery', 'furniture', and 'meat' categories show marginal misclassification. At the same time, 'cleaning' and 'drinks' shows huge misclassification due to high overlap with other categories. 'Cleaning' is misclassified into 'crockery' or 'Furniture', and 'drinks' is misclassified into 'furniture' and 'crockery.'

5. Attention Matrix Visualization

Fig. 4 shows the visualization of the attention map generated by SEMA. The SEMA does not compute the attention matrix. However, for visualization purposes, we have multiplied the Q' and K' to form an $N \times N$ attention matrix. For each frame (row in the figure), we pick the top 100 closest frame indexes. We observed that frames belonging to similar classes tend to attend similar frames and are distributed globally.

6. Details of *Epic Kitchens* Dataset

To demonstrate the proposed framework on the *Epic Kitchens* dataset [4], we synthesize a long video sequence (approx. 20k frames) using the *Epic Kitchens* dataset. This dataset is divided into high-level categories based on the occurrences of 'noun' classes. We pick equal video snippets of each category: 'appliances,' 'cleaning,' 'crockery,' 'drinks,' 'furniture,' 'meat,' and 'vegetables' across all subjects and concatenate them to form a long colossal sequence. The subset 'noun' classes selected for each category are listed in Table 1.

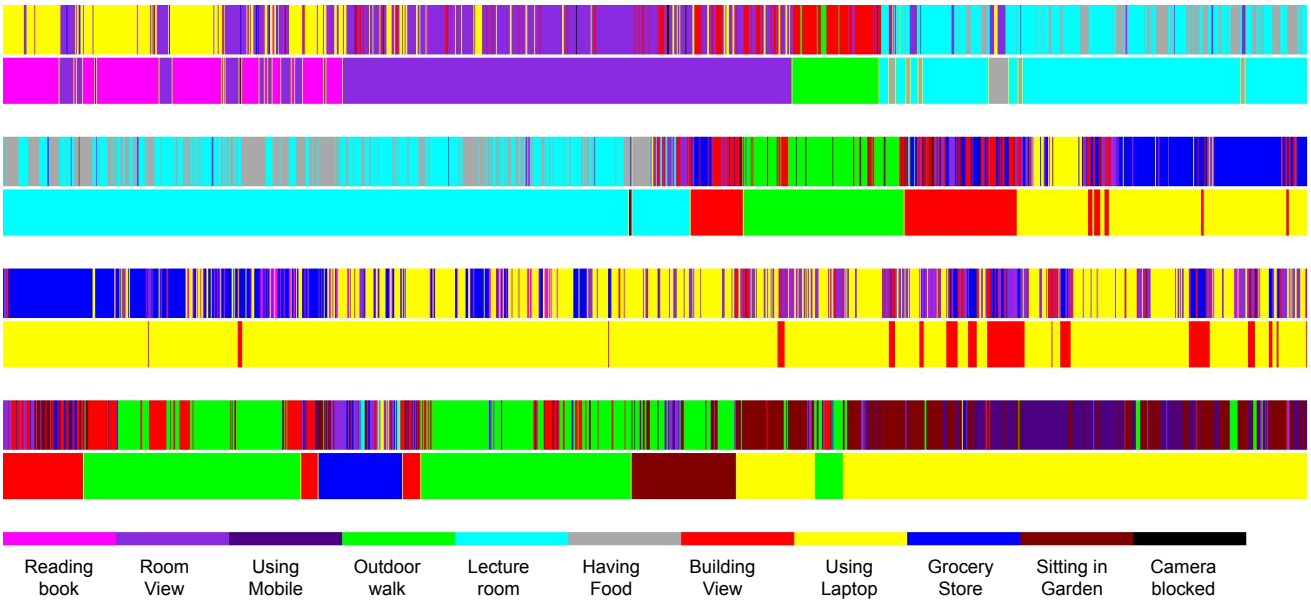

Figure 2. The figure demonstrates the visualization of a comparison between the predicted class and ground truth for 'P04' sequence of *UTE* dataset (for better visualization, we have divided the long sequence into 2000 frames for each row).

Category Id	Category Name	Nouns selected
1	Appliances	Washing Machine, Fridge
2	Cleaning	Cloth, Towel
3	Crockery	Plate, bowl
4	Drinks	Tea, Juice, Wine, Drink, Beer, Whisky
5	Furniture	Floor, Chair, Wall
6	Meat	Meat, Chicken, Sausage, Fish, Pork, Bacon, Beef
7	Vegetables	Onion, Potato, Carrot, Tomato, Mushroom, Cucumber, Vegetables

Table 1. Nouns selected corresponding to the categories for *Epic Kitchens* dataset.

7. Annotation Procedure

We recruited three participants from different backgrounds (ECE undergraduate, CSE undergraduate, and CSE graduate) for annotation. We have generated codebooks of each subject of the *EgoRoutine* [10] dataset separately and shared them with participants to annotate videos on the same granularity. Each *label file* comprises the activity number and the corresponding activity name (refer to Table 2 for the label file of subject-1 of *EgoRoutine* dataset). We share an annotation file with the participants, comprising two columns titled *start time* and the *activity number* for each day of the subject. The activities span for short to very long duration, so we just collect the activity's start time with its corresponding activity number (from the label file). Precisely, for a particular day of the photostream sequence, the user needs to start from the first fame of the sequence and identify the activity performed from the activity codebook shared. The timestamp of the frame and the activity number is filled in the two columns discussed. The timestamp can be obtained from the frame name itself. Each frame is named $XXXXXXXX\_HHMMSS\_XXX.jpg$, where $HHMMSS$ is the timestamp of the frame.

Table 3 demonstrates the details of the activity patterns used to annotate each subject. We can observe that the activity patterns are vast and similar to the real world. For each subject, the number and type of activity patterns differ significantly. The annotations also allow us to generate ground truth at multiple granularities as we annotated at high granularity. For example, we can always merge 'in metro', 'in cab' and 'in bus' activities to 'traveling' at low granularity.

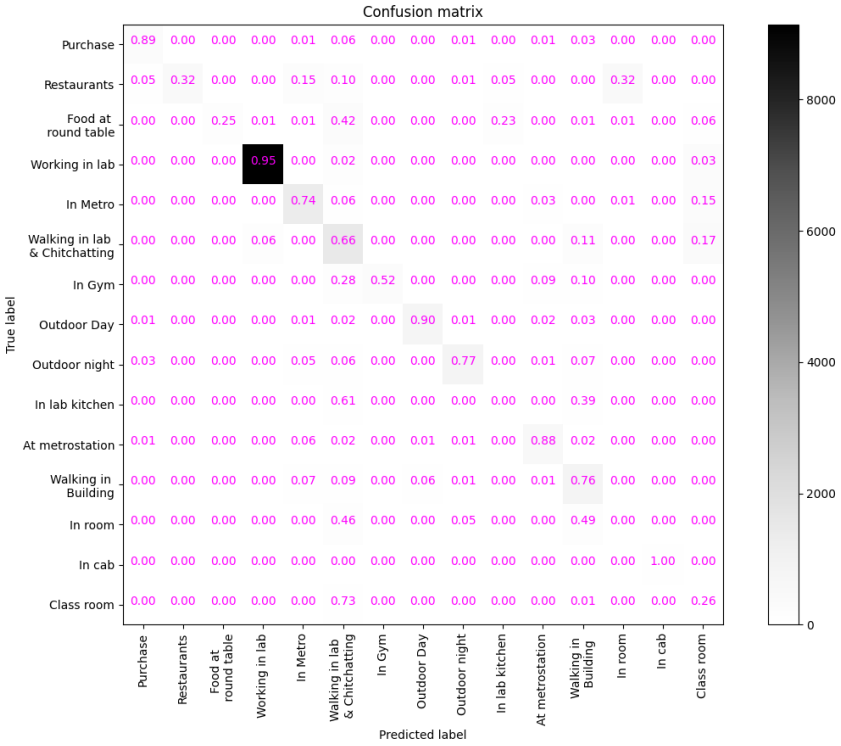

Figure 3. The confusion matrix demonstrates that inter-class confusion is marginal for most of the activity patterns.

Activity Number	Activity Name
1	buying
2	having food in restaurant
3	having meeting and food at round table
4	working in lab
5	in metro
6	walking in lab and chitchatting
7	in gym
8	outdoor walking in day
9	outdoor walking in night
10	in lab kitchen
11	at metro station
12	walking in the building
13	in room
14	in cab
15	class room

Table 2. Activity labels for 'S1' of *EgoRoutine* dataset.

8. Evaluation Matrix

We use commonly used evaluation matrices titled Adjusted Mutual Information (AMI) and Normalized Mutual Information (NMI), and F1 score to demonstrate the results. They are as follows:

8.1. Adjusted Mutual Information (AMI)

Suppose that the sequence of length N is partitioned into predicted clusters $A = \{A_1, A_2, ...A_{K_p}\}$ and ground truth clusters $B = \{B_1, B_2, ..., B_{K_g}\}$, where, K_p an K_g are the number of clusters in the ground truth and predicted clusters. The clusters are pairwise disjoint i.e. $|\cup_{i=1}^{K_p} A_i| = |\cup_{i=1}^{K_g} B_i| = N$. Then the mutual information between two clusters can be defined as:

$$MI(A,B) = \sum_{i=1}^{K_p}\sum_{j=1}^{K_g} P_{AB}(i,j) log \frac{P_{AB}(i,j)}{P_A(i)P_B(j)} \tag{24}$$

where $P_{AB}(i,j) = \frac{|A_i \cap B_j|}{N}$ denotes the probability that frame belong to both the clusters $A_i \in A$ and $B_i \in V$, $P_A(i) = \frac{|A_i|}{N}$, and $P_B(j) = \frac{|B_j|}{N}$.

The expected mutual information can be defined as:

$$E\{MI(A,B)\} = \sum_{i=1}^{K_p}\sum_{j=1}^{K_g}\sum_{n_{ij}=(a+b-N)^+}^{min(a_i,b_j)} \frac{n_{ij}}{N} log(\frac{N.n_{ij}}{a_i b_j}) \times \frac{a_i! b_j! (N-a_i)!(N-b_j)!}{N! n_{ij}! (a_i - n_{ij})!(b_j - n_{ij})!(N - a_i - b_j + n_{ij})!} \tag{25}$$

where $(a_i + b_j - N)^+ = max(1, a_i + b_j - N)$, n_{ij} denotes the number of common frames in clusters A_i and B_j, $a_i = \sum_{j=1}^{K_g} n_{ij}$, and $b_j = \sum_{i=1}^{K_p} n_{ij}$.

From Equation 24 and 25, the AMI is defined as [7]:

$$AMI(A,B) = \frac{MI(A,B) - E\{MI(A,B)\}}{max\{H(A), H(B)\} - E\{MI(A,B)\}} \tag{26}$$

8.2. Normalized Mutual Information (NMI)

Similarly, from equation 24 and 25, the NMI is defined as [7]:

$$NMI(A,B) = \frac{MI(A,B)}{(H(A) + H(B))/2} \tag{27}$$

8.3. F-score

Similar to [1,5], we use the greedy approach [6] for a one-to-one mapping between the predicted clusters and ground truth clusters. The cost of assigning cluster i to class label j is computed as the F1 score weighted by population for class j when i is assigned to j. The precision (P_r), recall (R_r) and F-score (F_r) are defined as:

$$P_r = \frac{\text{TP}}{\text{TP} + \text{FP}} \quad \text{, and} \quad R_r = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

$$F_r = \frac{2 \times P_r \times R_r}{P_r + R_r} \times 100\% \tag{28}$$

where TP, FP, and FN represent the true positive, false positive, and false negative calculations.

9. Hardware Requirements

All the codes are executed on $A100$ GPU (40GB GPU memory) and the CPU (Intel Xeon Silver 4210) with 128GB RAM and 48 cores. It takes approximately half an hour to execute 3000 epochs.

Id	#Activities	Name of Activity Patters
S1	15	buying, having food in restaurant, having meeting and food at round table, working in lab, in metro, walking in lab and chitchatting, in gym, outdoor walking in day, outdoor walking in night, in lab kitchen, at metro-station, walking in the building, in room, in cab, in class room
S2	25	in home kitchen, in balcony (tea), working on laptop at home, having food, walking in building, walking in day (walking outdoor), cycling, operating vending machine/ATM, at metro station, in metro, walking in night, purchasing, using mobile/kindle (in room), washroom, sitting at beach, in mall /hotel having food, bus, room view, using laptop in lab, using mobile/kindle but not in room, having tea (in room), walking in lab and chitchatting, in library, working on laptop at library, in museum
S3	16	room view, in kitchen, having food (room/restaurant/cafe), at metro station, in metro, in washroom, outdoor walk in day, walking in building/ taking printout from printer, walking in lab and chitchatting, working on laptop (watching movie on laptop), in class room/ attending presentation, using mobile, purchasing (in mall/food/bakery/watch), outdoor walk in night, at airport, in advisor's room
S4	31	in room (walking, kitchen), walking outdoor (day), walking in building, working on laptop or desktop (in room or lab), riding bike, in hospital waiting room, with doctor, having food/in restaurant, walking in lab and chitchatting, using mobile (outdoor/restaurant/airport), in classroom, in washroom, watching TV and using mobile in room, purchasing toys, veggies, fruits, at airport, walking outdoor in night, at metro station, in metro, driving car, in swimming pool, Blur images, in school, in plane, attending a presentation, coffe/tea break at conference/at lounge, giving presentation, in cab, hosting a conference as receptionist, in open-bus/bus, at poster, at beach and mountains
S5	24	in room, outdoor walk in day, walking in building, working on laptop, driving car, in metro, at metro station, walking in lab and chitchatting, class room/attending presentation in audi/conference), having food at (home/restaurant/in conference), outdoor walk in night, in cab, at airport, in flight, purchasing (on stores at airport/local shops/mall/tickets at bus station), in hotel room/ hotel, at conference venue/ lounge/ reception, in bus, on beach, archaeological zone, at poster, monument visit, bus station, watching television
S6	19	at home, outdoor walk in day, walking in building, working on laptop, walking in lab and chitchatting, purchasing (food, shoes, toys, books), having food (in lab, restaurant), washroom, outdoor walk in night, in kitchen, in classroom, in Bus, in hotel room, museum, in car, visiting a old township and mountains, at metro station, in metro, at circus
S7	25	at home, outdoor walk in day, walking in building, working on laptop, walking in lab and chitchatting, purchasing (food, cloths, sweets, fruits, in supermarket), having food (in lab, restaurant), at metro station, in metro, at fair, washroom, outdoor walk in night, blank frame, in bus, in hospital/clinic/medical facility, in classroom, at concert, meeting with professor, sitting in park (picnic), at conference venue, at poster, attending presentation, in kitchen, in car, walking in hill area/trekking

Table 3. Table demonstrates the details of the activity patterns used to annotate each subject. We can observe that the activity patterns are vast and similar to the real world. For each subject, the number and type of activity patterns differ significantly. The annotations also allow us to generate ground truth at multiple granularities as we annotated at high granularity. For example, we can always merge 'in metro', 'in cab' and 'in bus' activities to 'traveling' at low granularity.

References

[1] Bharat Lal Bhatnagar, Suriya Singh, Chetan Arora, and CV Jawahar. Unsupervised learning of deep feature representation for clustering egocentric actions. In *IJCAI*, 2017.

[2] Krzysztof Choromanski, Valerii Likhosherstov, David Dohan, Xingyou Song, Andreea Gane, Tamas Sarlos, Peter Hawkins, Jared Davis, Afroz Mohiuddin, Lukasz Kaiser, et al. Rethinking attention with performers. *arXiv preprint arXiv:2009.14794*, 2020.

[3] Krzysztof M Choromanski, Mark Rowland, and Adrian Weller. The unreasonable effectiveness of structured random orthogonal embeddings. *NIPS*, 2017.

[4] Dima Damen, Hazel Doughty, Giovanni Maria Farinella, Antonino Furnari, Evangelos Kazakos, Jian Ma, Davide Moltisanti, Jonathan Munro, Toby Perrett, Will Price, et al. Rescaling egocentric vision: collection, pipeline and challenges for epic-kitchens-100. *IJCV*, 2022.

[5] Kris M Kitani, Takahiro Okabe, Yoichi Sato, and Akihiro Sugimoto. Fast unsupervised ego-action learning for first-person sports videos. In *CVPR*, 2011.

[6] Harold W Kuhn. The hungarian method for the assignment problem. *Naval research logistics quarterly*, 1955.

[7] Xuan Vinh Nguyen, Julien Epps, and James Bailey. Information theoretic measures for clusterings comparison: is a correction for chance necessary? In *ICML*, 2009.

[8] Ali Rahimi and Benjamin Recht. Random features for large-scale kernel machines. *NIPS*, 2007.

[9] Ozan Sener and Silvio Savarese. Active learning for convolutional neural networks: A core-set approach. *arXiv preprint arXiv:1708.00489*, 2017.

[10] Estefania Talavera, Carolin Wuerich, Nicolai Petkov, and Petia Radeva. Topic modelling for routine discovery from egocentric photo-streams. *Pattern Recognition*, 2020.

[11] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NIPS*, 2017.

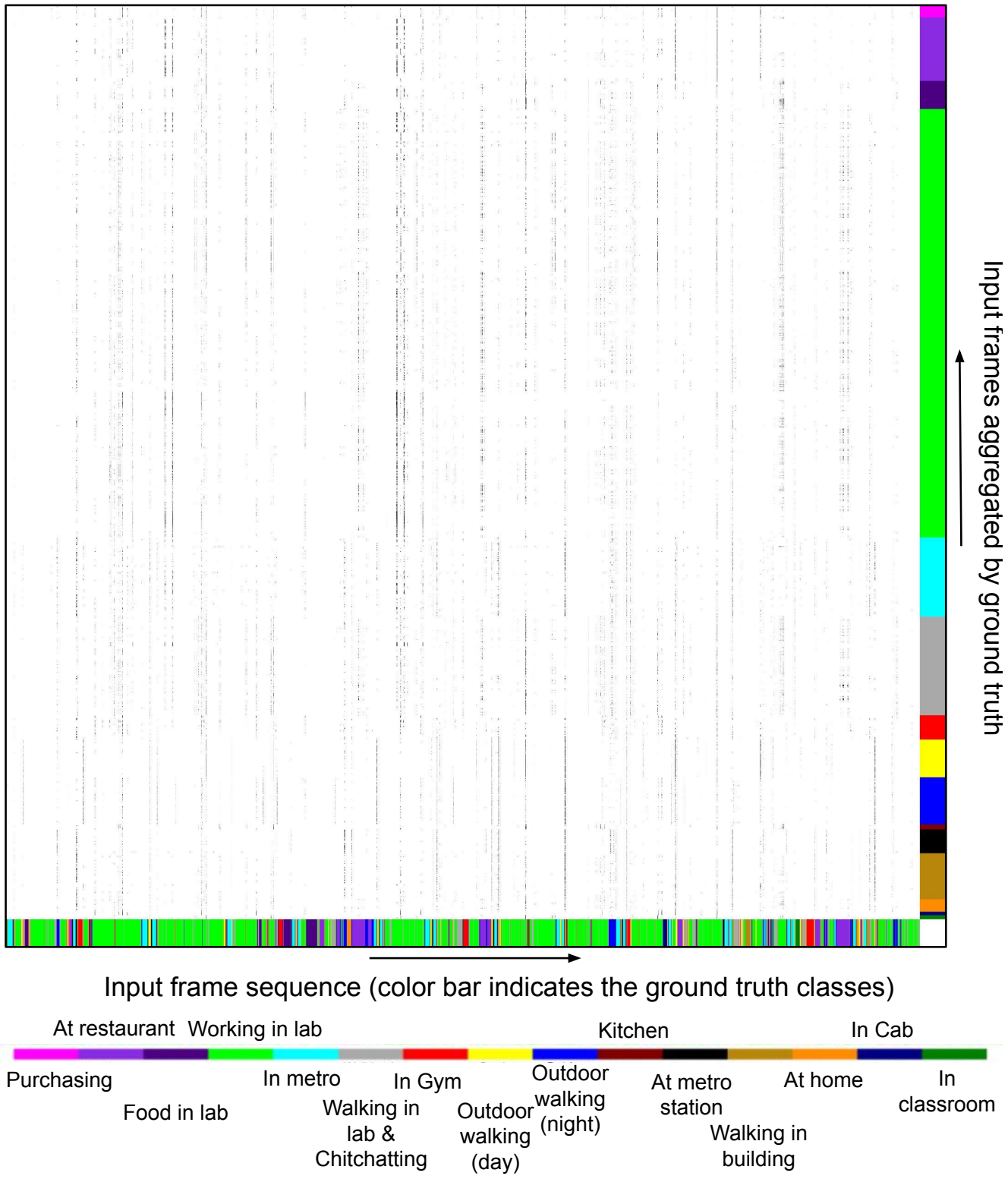

Figure 4. The visualization shows the attention map generated by the SEMA for 'subject 1' (all 14 days). The color bar on the X-axis and Y-axis shows the ground truth corresponding to the frame. The white patches in each row show the frames attended corresponding to the input frame. We sorted the frames by the ground truth classes (color bar in the column) to better view the attention of frames belonging to similar classes.