Стабилизация обучения слотовых распутанных представлений для реальных данных

Фролов В.А.
frolov.vladimir@phystech.edu

Московский физико-технический институт
Центр когнитивного моделирования МФТИ

Научный руководитель к.ф.-м.н. А.И. Панов

2024 – Научно-исследовательский семинар

Введение

Цель: Разработка перспективного метода в задаче стабилизации обучения slot-based моделей на основе исследования различных подходов к реконструкции сцен и обнаружению объектов на ней.

Задачи:

- изучение и запуск базового решения;
- интегририрование интересных подходов в существующую модель;
- анализ результатов, релевантный исследуемой проблеме.

Исследуемая проблема:

1. масштабирование объектно-ориентированных методов на сложные данные реального мира, в том числе видео.

Список литературы

1. *M. Seitzer, M. Horn, A. Zadaianchuk, D. Zietlow, T. Xiao, C.J. Simon-Gabriel, T. He, Z. Zhang, B. Schölkopf, T. Brox, F. Locatello.* Bridging the Gap to Real-World Object-Centric Learning // In ICLR 2023.
2. *G. Singh, Y. Kim, S. Ahn.* Neural Systematic Binder // In ICLR 2023.
3. *D. Kirilenko, V. Vorobyov, A. Kovalev, A. Panov.* Object-Centric Learning with Slot Mixture Module // In ICLR 2024.
4. *C. Kim, W. Han, D. Ju, S.J. Hwang.* EAGLE: Eigen Aggregation Learning for Object-Centric Unsupervised Semantic Segmentation // In CVPR 2024.
5. *G. Hinton, O. Vinyals, J. Dean.* Distilling the Knowledge in a Neural Network // In NIPS 2014 Deep Learning Workshop.

План доклада

1. Общая постановка задачи
2. **DINO** and **S**lot **A**ttention **U**sing **R**eal-world data (DINOSAUR)
3. **N**eural **S**ystematic **B**inder (NSB)
4. **S**lot **E**igen **A**ggregation (SEA)
5. **S**lot **M**ixture **M**odule (SMM)
6. Результаты экспериментов

Общая постановка задачи обучения ООП

Задано:

- множество объектов Ω, где $|\Omega| = T$ и $\omega_t \in \Omega$;
- множество признаков $\mathbb{X} \in \mathbb{R}^{H \times W \times 3}$;
- отображение $\varphi : \Omega \to \mathbb{R}^{H \times W \times 3}$, обозначим $\mathbf{x}_t = \varphi(\omega_t)$ и $\forall \mathbf{x}_t,\ \mathbf{x}_t \subset \mathbb{X}$.

Пусть на множестве признаков задана функция $f_{\mathbf{enc}}\big(\mathbf{x}\big)$, а на множестве латентных векторов – $f_{\mathbf{dec}}\big(\mathbf{x}^*\big)$:

$$f_{\mathbf{enc}} : \mathbb{X} \to \mathbb{X}^*,$$

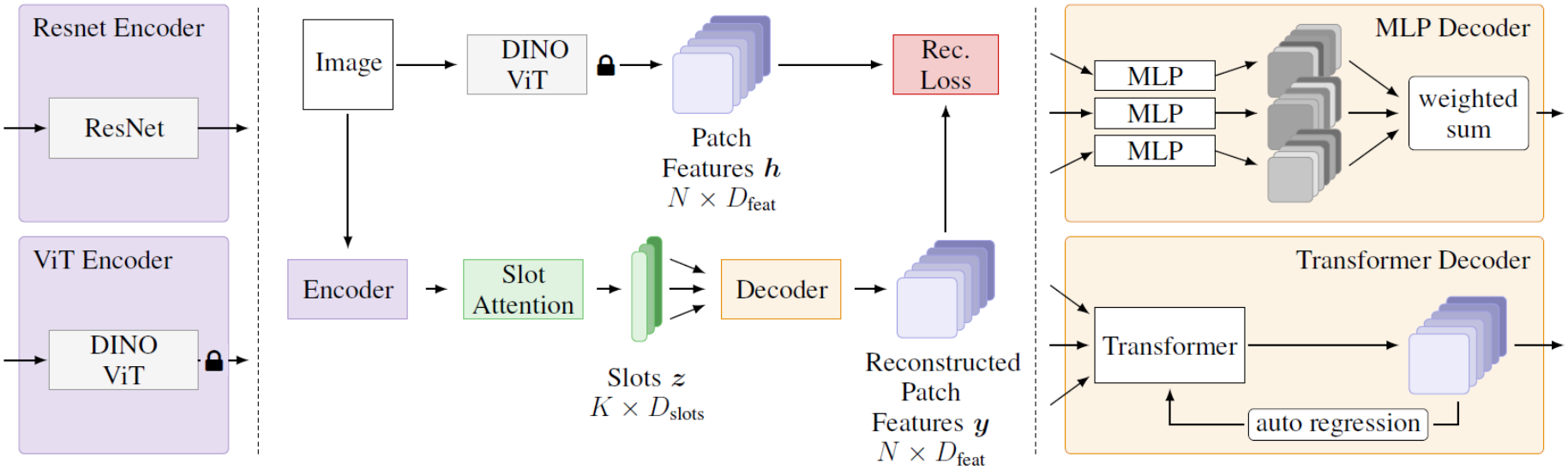
$$f_{\mathbf{dec}} : \mathbb{X}^* \to \hat{\mathbb{X}} \in \mathbb{R}^{H \times W \times 3},$$

где $\mathbb{X}^* \in \mathbb{R}^d$ – представление множества признаков в более компактном латентном пространстве размерности d; $\hat{\mathbb{X}}$ – признаковые описания объектов на выходе, которые мы хотим приблизить к исходным. Для удобства обозначим $f_{\mathbf{dec}}\big(\mathbb{X}^*\big) = \mathcal{P}$.

Заметим, что это задача регрессии, причём такая, что на промежуточном этапе, который будет являться узким местом в модели, может решаться задача классификации или кластеризации для разделения сущностей в объекте.

Оптимизационная задача:

$$\mathcal{Q} = \min_{\mathbf{p}_t \in \mathcal{P}} \sum_{t=1}^{T} \|\mathbf{x}_t - \mathbf{p}_t\|_2^2.$$



Для каждого изображения $\mathbf{x}_t$, где $t \in \{1, ..., T\}$, ставится задача кодирования и реконструкции признаков $\hat{\mathbf{f}}_t$, для чего строится отображения $f_{\mathbf{enc}}$:

$$\mathbf{f}_t^* = f_{\mathbf{enc}}(\mathbf{x}_t) \in \mathbb{R}^{N \times D_{\mathbf{f}}},$$

где $\mathbf{f}_t^*$ – представление признаков изображения $\mathbf{x}_t$; N – количество токенов; $D_{\mathbf{f}}$ – размерность признаков.

---

https://openreview.net/pdf?id=b9tUk-f_aG

Задаётся K слотов, представляющих отдельные объекты или их части. Набор слотовых представлений для $\mathbf{f}_t^*$ обозначим $\mathcal{S}_t = \{s_t^1, ..., s_t^K\} \in \mathbb{R}^{K \times D_\mathbf{s}}$. Мы хотим итеративно обновлять эти представления с помощью cross-attention:

$$\mathcal{A}_t = \underset{K}{\text{softmax}}\left[\frac{1}{\sqrt{D_\mathbf{s}}} q(\mathcal{S}_t) \cdot k(\mathbf{f}_t^*)^T\right] \in \mathbb{R}^{K \times N}, \quad \mathcal{U}_t = \frac{1}{\mathcal{W}_t} \sum_{k=1}^{K} \mathcal{A}_{k,t} \cdot v(\mathbf{f}_{k,t}^*) \in \mathbb{R}^{K \times D_\mathbf{s}},$$

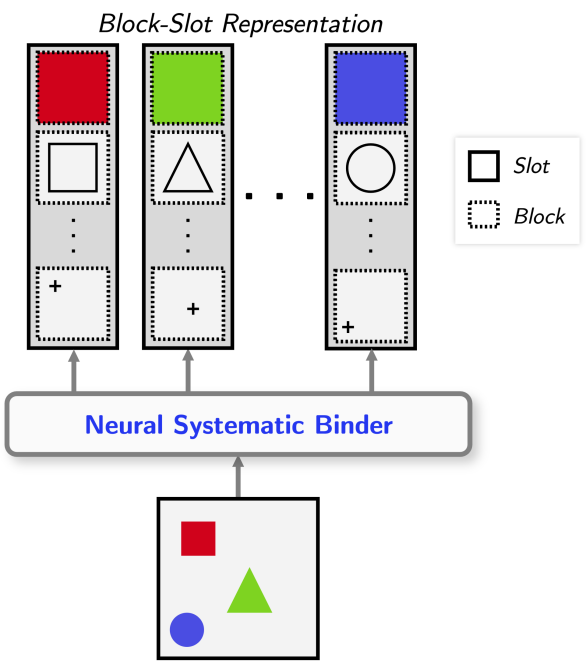
$$\hat{s}_t^k = \text{GRU}(u_t^k, s_t^k) \in \hat{\mathcal{S}}_t, \qquad \hat{s}_t^k \leftarrow \hat{s}_t^k + \text{MLP}[\text{LN}(\hat{s}_t^k)],$$

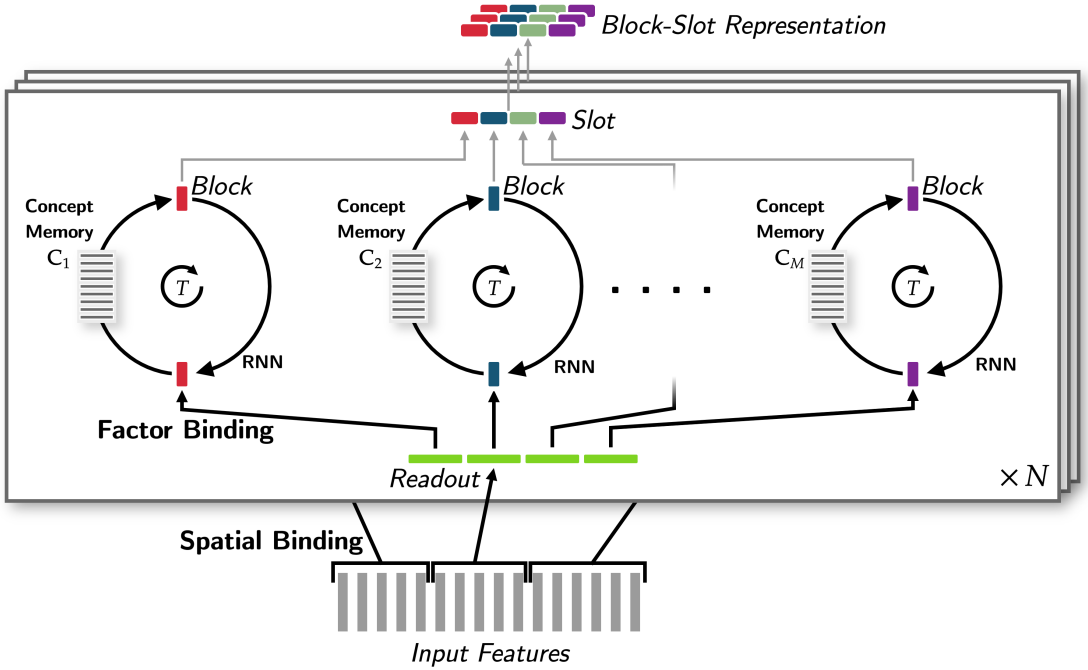
где $\mathcal{W}_t = \sum_{k=1}^{K} \mathcal{A}_{k,t}$; k, q, v – обучаемые линейные проекции $\mathbf{f}_t^*$, $\mathcal{S}_t$ и $\mathbf{f}_t^*$, соответственно, которые отображаются в общем пространстве размерности $D_\mathbf{s}$. Обновлённые слотовые представления используются для реконструкции признаков $\mathbf{f}_t^*$ и предсказания альфа-маски, для чего строится отображение $f_\mathbf{dec}$:

$$\hat{\mathbf{m}}_t^k, \hat{\mathbf{f}}_t^k = f_\mathbf{dec}(\hat{s}_t^k), \quad \hat{\mathbf{m}}_t = \underset{K}{\text{softmax}}(\hat{\mathbf{m}}_t^k), \quad \hat{\mathbf{f}}_t = \sum_{k=1}^{K} \hat{\mathbf{m}}_t \odot \hat{\mathbf{f}}_t^k.$$

Оптимизационная задача:

$$\mathcal{L}_\text{rec} = \min_{\hat{\mathbf{f}}_t \in \mathcal{F}} \sum_{t=1}^{T} \|\mathbf{f}_t - \hat{\mathbf{f}}_t\|_2^2.$$





Постановка задачи не изменяется, но в дополнение к пространственной привязке слотов добавляется абстракция на привязку к характеристикам объектов (цвет, форма, положение), которые моделируются с помощью так называемых блоков. Формально, для каждго слота $s_t^k \in \mathcal{S}_t$ назначается M блоков размерности $D_{\mathbf{s}}$. Каждый слот $s_t^k \in \mathbb{R}^{Md}$ является объединением M блоков:

$$\mathcal{A}_t = \underset{K}{\text{softmax}}\left[\frac{1}{\sqrt{MD_{\mathbf{s}}}} q(\mathcal{S}_t) \cdot k(\mathbf{f}_t^*)^T\right], \quad \mathcal{U}_t = \frac{1}{\mathcal{W}_t}\sum_{k=1}^{K} \mathcal{A}_{k,t} \cdot v(\mathbf{f}_{k,t}^*) \in \mathbb{R}^{K \times MD_{\mathbf{s}}}.$$

https://openreview.net/pdf?id=ZPHE4fht19t

Далее каждое слотовое представление разбивается на M сегментов одинаковых размеров и к каждому такому сегменту $u_t^{k,m}$ применяется уточняющая RNN:

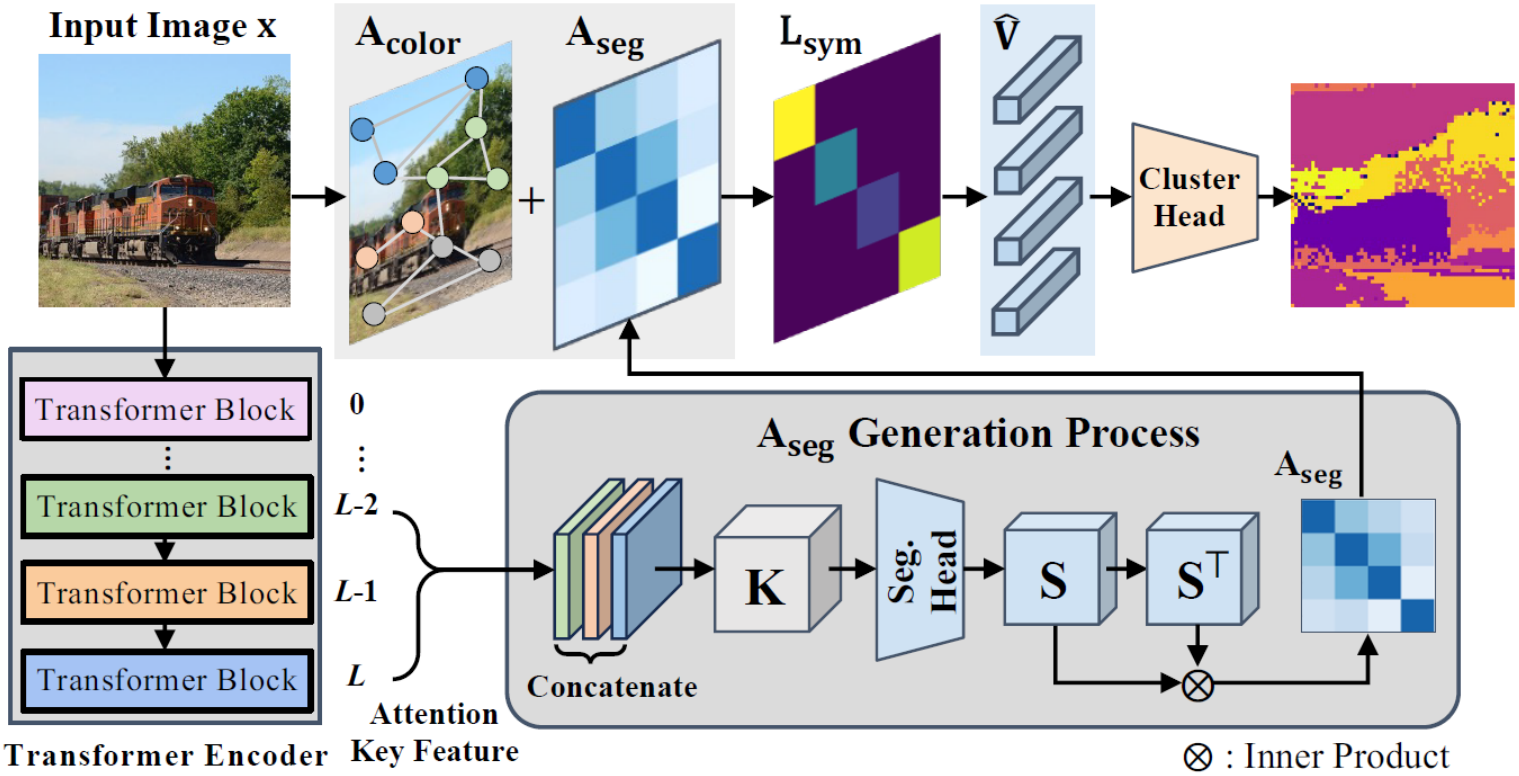
$$\hat{s}_t^{k,m} = \mathrm{GRU}(u_t^{k,m}, s_t^{k,m}) \in \hat{\mathcal{S}}_t, \qquad \hat{s}_t^{k,m} \leftarrow \hat{s}_t^{k,m} + \mathrm{MLP}[\mathrm{LN}(\hat{s}_t^{k,m})].$$

Затем для каждого блока извлекается представление из концептуальной памяти $\mathbf{C}_m \in \mathbb{R}^{L \times D_\mathbf{s}}$:

$$s_t^{k,m} = \left[\underset{L}{\mathrm{softmax}} \left(\frac{s_t^{k,m} \cdot \mathbf{C}_m^T}{\sqrt{D_\mathbf{s}}} \right) \right] \cdot \mathbf{C}_m \in \mathbb{R}^{D_\mathbf{s}},$$

где $\mathbf{C}_m$ – набор из L обучаемых векторов (прототипов), связанных с конкретной характеристикой m.

Slot Eigen Aggregation



Данный подход заключается в продвинутой обработке признаков: формирование матрицы смежности $\mathbf{A}$ на основе цветового $\mathbf{A_{color}}$, а также семантического подобия $\mathbf{A_{seg}}$, симметричного нормализованного лапласиана $\mathbf{L_{sym}}$ и его собственное разложения для получения собственного базиса $\mathbf{V} \in \mathbb{R}^{N \times N}$.

---

https://openaccess.thecvf.com/content/CVPR2024/papers/Kim_EAGLE_Eigen_Aggregation_Learning_for_Object-Centric_Unsupervised_Semantic_Segmentation_CVPR_2024_paper.pdf

Матрица цветового подобия $\mathbf{A_{color}}$ вычисляется на основе евклидовых расстояний в RGB пространстве исходного изображения $\mathbf{x}_t$ с использованием радиально-базисной функции:

$$\mathbf{A_{color}}(p,q) = e^{-\frac{\|\ddot{\mathbf{x}}(p)-\ddot{\mathbf{x}}(q)\|_2^2}{2\sigma_c^2}}, \qquad \mathbf{A_{color}} \in \mathbb{R}^{N\times N}, \quad \sigma_c > 0,$$

где N – количество патчей; (p,q) – точки, между которыми считается функция расстояния.

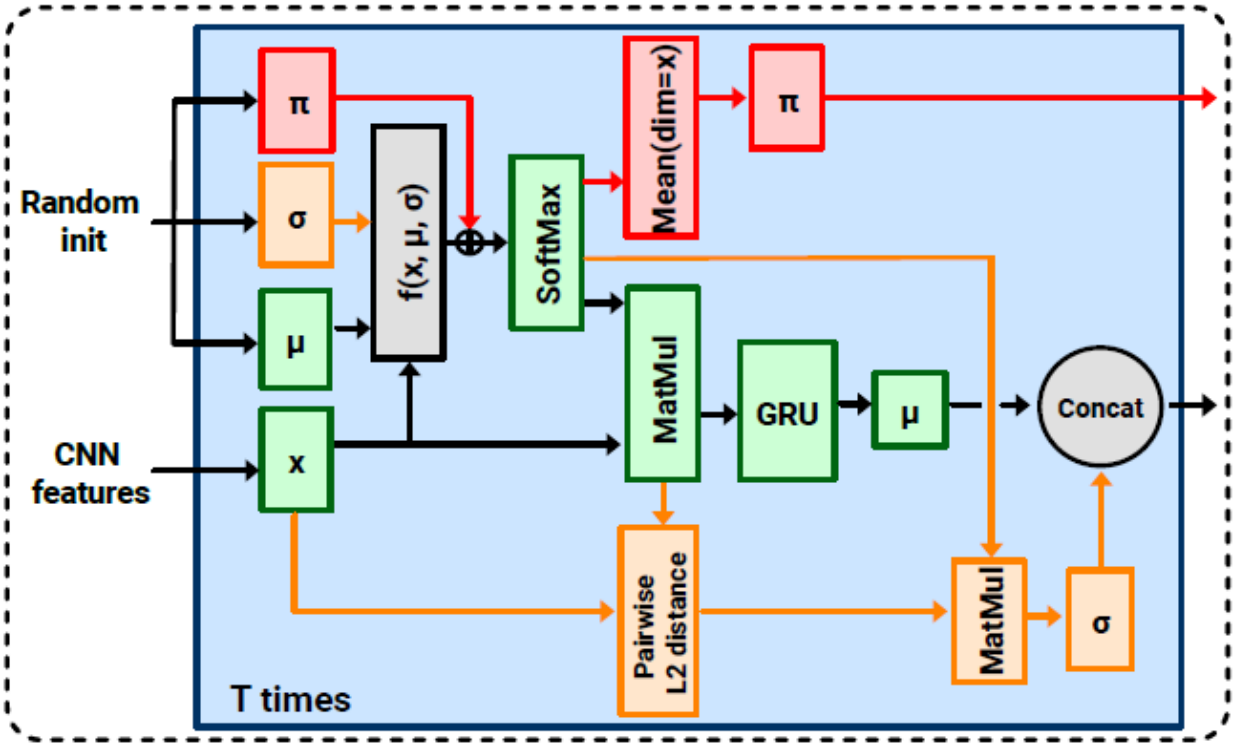
Матрица семантического подобия $\mathbf{A_{seg}}$ формируется как:

$$\mathbf{A_{seg}} = \mathbf{S}\mathbf{S}^T, \qquad \mathbf{A_{seg}} \in \mathbb{R}^{N\times N}, \quad \mathbf{S} \in \mathbb{R}^{H\times W\times D_S},$$

где $\mathbf{S}$ – агрегированные путём иерархической конкатенации ключевые признаки, полученные с кодировщика; D_S – размерность признаков.

Окончательная матрица смежности $\mathbf{A}$ представляет собой сумму вышеописанных матриц. Далее на её основе формируется симметричный нормализованный лапласиан $\mathbf{L_{sym}}$, который подвергается собственному разложению для извлечения k собственных векторов, формирующих набор $\hat{\mathbf{V}} \in \mathbb{R}^{N\times k}$:

$$\mathbf{A} = \mathbf{A_{color}} + \mathbf{A_{seg}}, \quad \mathbf{L} = \mathbf{D} - \mathbf{A}, \quad \mathbf{D}(i,i) = \sum_{j=1}^{N} \mathbf{A}(i,j), \quad \mathbf{L_{sym}} = \mathbf{D}^{-\frac{1}{2}}\mathbf{L}\mathbf{D}^{-\frac{1}{2}}.$$



Хотим построить вероятностную модель для экземпляра $\mathbf{x}_t$, порождённого смесью распределений с K компонентами и априорными весами $\pi \in \mathbb{R}^K$:

$$\mathbf{x}_t \sim p(\mathbf{x}_t|\theta) = \sum_{k=1}^{K} \pi_k p(\mathbf{x}_t|\theta_k), \quad P(\mathbb{X}|\theta) = \prod_{t=1}^{T} p(\mathbf{x}_t|\theta), \quad \sum_{k=1}^{K} \pi_k = 1,$$

https://openreview.net/pdf?id=aBUidW4Nkd

С помощью EM-алгоритма ищется K групп параметров θ_k, задающих каждый экземпляр $\mathbf{x}_t$, которые максимизируют оценку правдоподобия модели $P(\mathbb{X}|\theta)$.

E-шаг:

$$P(\mathbb{X}, \mathbb{X}^*|\theta^*) = \prod_{t=1}^{T}\prod_{k=1}^{K}[\pi_k p(\mathbf{x}_t|\theta_k^*)]^{I(\mathbf{x}_t^*=\mathbf{x}_k^*)},$$

$$Q(\theta^*, \pi^*|\theta, \pi) = \mathbb{E}_{P(\mathbb{X}^*|\mathbb{X},\theta)}[\log P(\mathbb{X}, \mathbb{X}^*|\theta^*)],$$

$$p(\mathbf{x}_k^*|\mathbf{x}_t) = \frac{p(\mathbf{x}_k^*)p(\mathbf{x}_t|\theta_k)}{\sum_{k=1}^{K} p(\mathbf{x}_k^*)p(\mathbf{x}_t|\theta_k)} = \frac{\pi_k \mathcal{N}(\mathbf{x}_t|\mu_k, \Sigma_k)}{\sum_{k=1}^{K} \pi_k \mathcal{N}(\mathbf{x}_t|\mu_k, \Sigma_k)} = \gamma_{k,t},$$

где I – индикаторная функция.

M-шаг:

$$(\theta, \pi) = \arg\max_{(\theta^*, \pi^*)} Q(\theta^*, \pi^*|\theta, \pi),$$

$$\pi_k^* = \frac{\sum_{t=1}^{T}\gamma_{k,t}}{T},\ \mu_k^* = \frac{\sum_{t=1}^{T}\gamma_{k,t}\mathbf{x}_t}{\sum_{t=1}^{T}\gamma_{k,t}},\ \Sigma_k^* = \frac{\sum_{t=1}^{T}\gamma_{k,t}(\mathbf{x}_t - \mu_k)(\mathbf{x}_t - \mu_k)^T}{\sum_{t=1}^{T}\gamma_{k,t}}.$$

Результаты

	DINOSAUR	NSB	SEA	SMM
ARI	**45.12**	40.94	41.21	44.68
mBO	29.24	28.05	28.32	**29.32**

$$\mathrm{RI} = \frac{2(a+b)}{n(n-1)}, \qquad \mathrm{ARI} = \frac{\mathrm{RI} - \mathbb{E}[\mathrm{RI}]}{\max \mathrm{RI} - \mathbb{E}[\mathrm{RI}]},$$

где a — число пар объектов, имеющих одинаковые метки в обоих разбиениях (кластеризациях) $\mathbb{Y}$ и $\hat{\mathbb{X}}$; b — число пар объектов, имеющих различные метки в обоих разбиениях; n — число объектов в выборке.

$$\mathrm{mBO} = \frac{1}{TC} \sum_{t=1}^{T} \sum_{c=1}^{C} \frac{\mathbf{y}_{c,t} \cap \arg\max_{\hat{\mathbf{x}}_{c,t} \in \hat{\mathbb{X}}} \mathbf{y}_{c,t} \cap \hat{\mathbf{x}}_{c,t}}{\mathbf{y}_{c,t} \cup \arg\max_{\hat{\mathbf{x}}_{c,t} \in \hat{\mathbb{X}}} \mathbf{y}_{c,t} \cap \hat{\mathbf{x}}_{c,t}},$$

где C — количество классов; $\mathbf{y}_{c,t}$ — истинные метки для экземпляра с порядковым номером t в классе c; $\hat{\mathbf{x}}_{c,t}$ — оценки для экземпляра t в классе c.

DINOSAUR



NSB

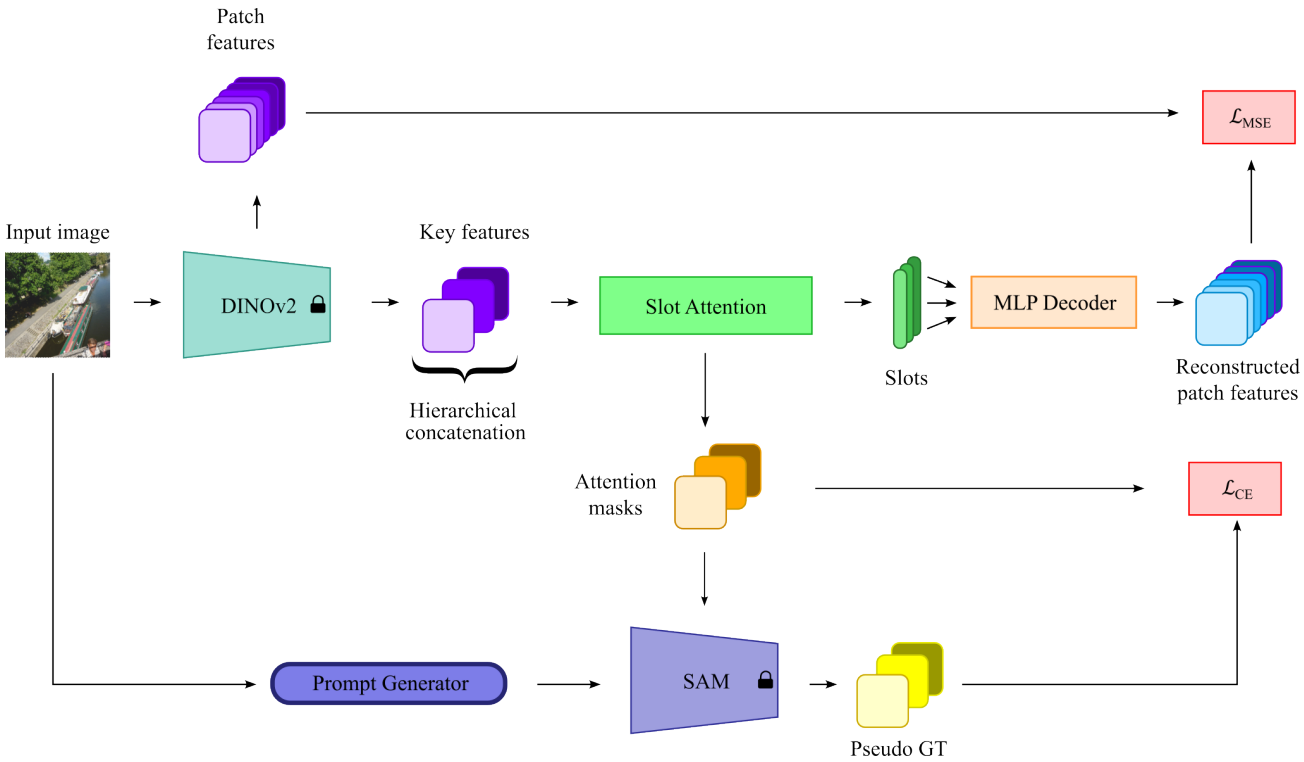


SEA



SMM





После итеративного уточнения слотов $\mathcal{S}_t$ и масок внимания $\mathcal{A}_t$ хотим задаться некоторым эталоном того, что хотелось бы сгенерировать $\overline{\mathcal{A}}_t$, и обуславливаться на него. Для получения таких подсказок вводится отображение $f_{\mathbf{SAM}}$:

$$\overline{\mathcal{A}}_t = f_{\mathbf{SAM}}(\mathcal{A}_t), \qquad \mathcal{L}_{MSE} = \sum_{t=1}^{T} \|\mathbf{f}_t - \hat{\mathbf{f}}_t\|_2^2, \qquad \mathcal{L}_{CE} = -\sum_{t=1}^{T} \overline{\mathcal{A}}_t \log \mathcal{A}_t,$$

$$\mathcal{L} = \mathcal{L}_{MSE} + \mathcal{L}_{CE}.$$

Заключение

Планируется:

1. Попробовать техники регуляризации, основанные на информационном потенциале, моделировать не совсем нормальные распределения для SMM.
2. Для SEA провести очередной анализ написанного кода и попытаться извлекать из собственных векторов информацию именно об объектах, а не о сцене в целом.
3. Продолжать работу в weakly supervised парадигме с guidance loss и интеграцией SAM в объектно-ориентированные модели.

Спасибо за внимание!

frolov.vladimir@phystech.edu