Coarse-to-Fine Sparse Sequential Recommendation

Jiacheng Li[1], Tong Zhao[2], Jin Li[2], Jim Chan[2], Christos Faloutsos[2]
George Karypis[2], Soo-Min Pantel[2], Julian McAuley[2]
j9li@eng.ucsd.edu,{zhaoton,jincli,jamchan,faloutso,gkarypis,pantel,jumcaule}@amazon.com
[1]University of California, San Diego [2]Amazon, United States

SIGIR 2022

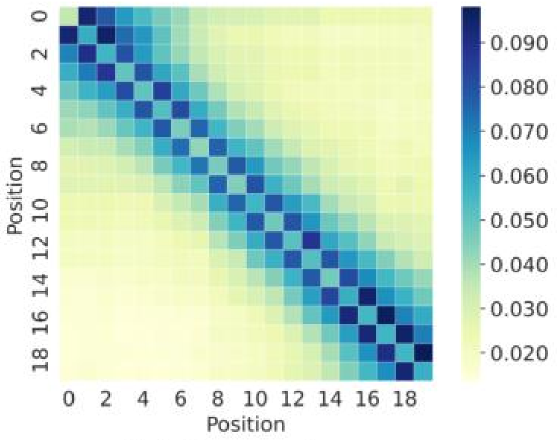
2022. 4. 12 • ChongQing

Leibniz-Institut für Sozialwissenschaften

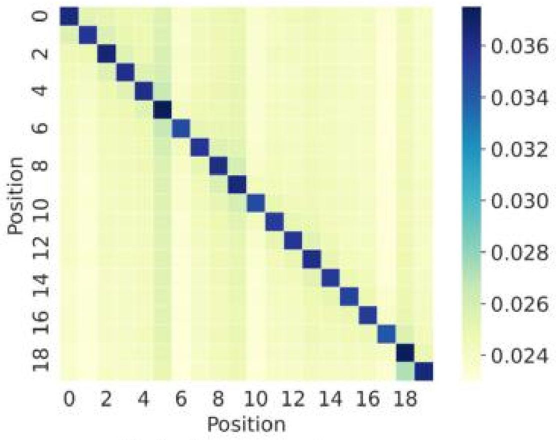
Reported by Gu Tang

Introduction

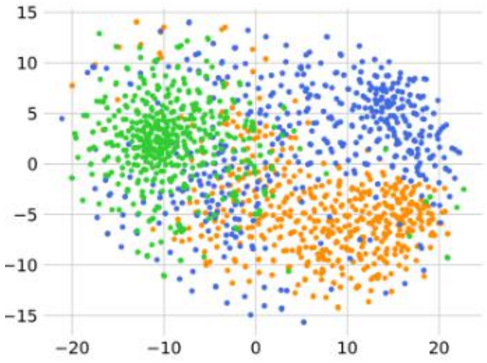
Figure 1: Illustration of a coarse-grained sequence (intents) and a fine-grained sequence (items).



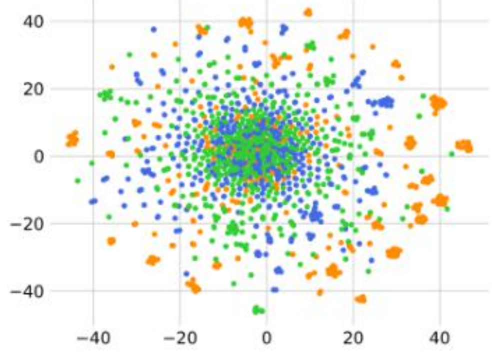
(a) *Dense* dataset



(b) *Sparse* dataset



(c) Frequent items



(d) Infrequent items

Method

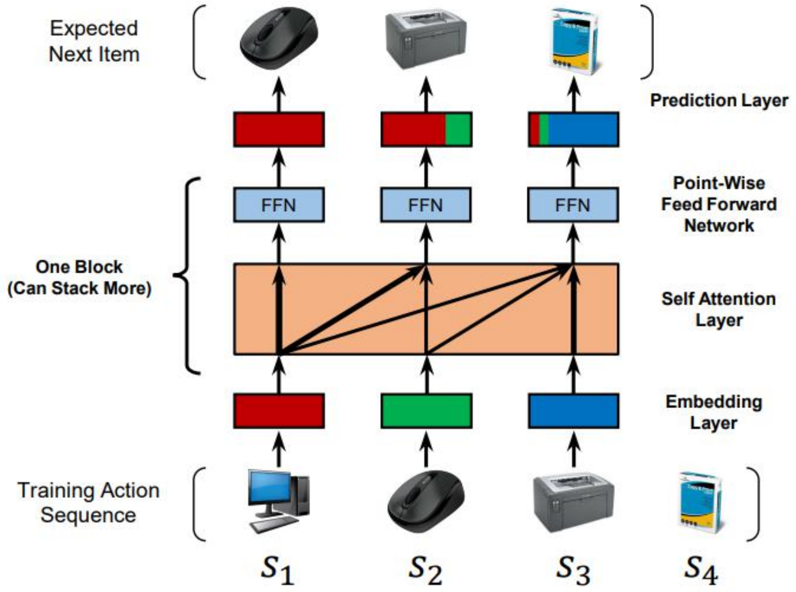


Figure 1: A simplified diagram showing the training process of SASRec. At each time step, the model considers all previous items, and uses attention to 'focus on' items relevant to the next action.

Base model

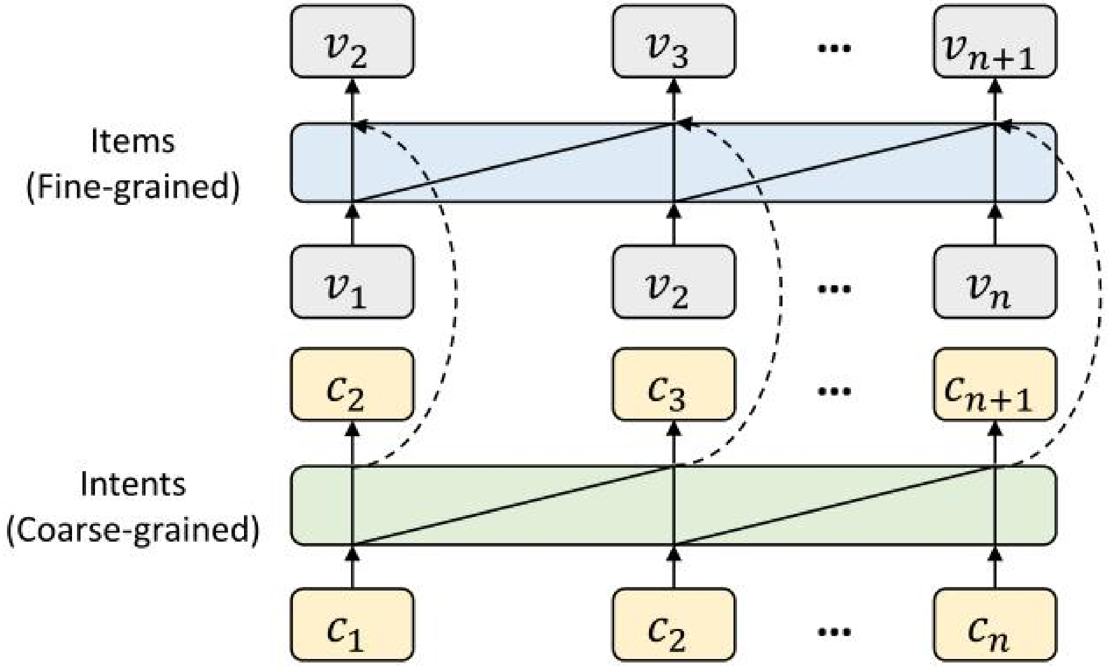


Figure 3: Framework illustration of CaFe.

Method

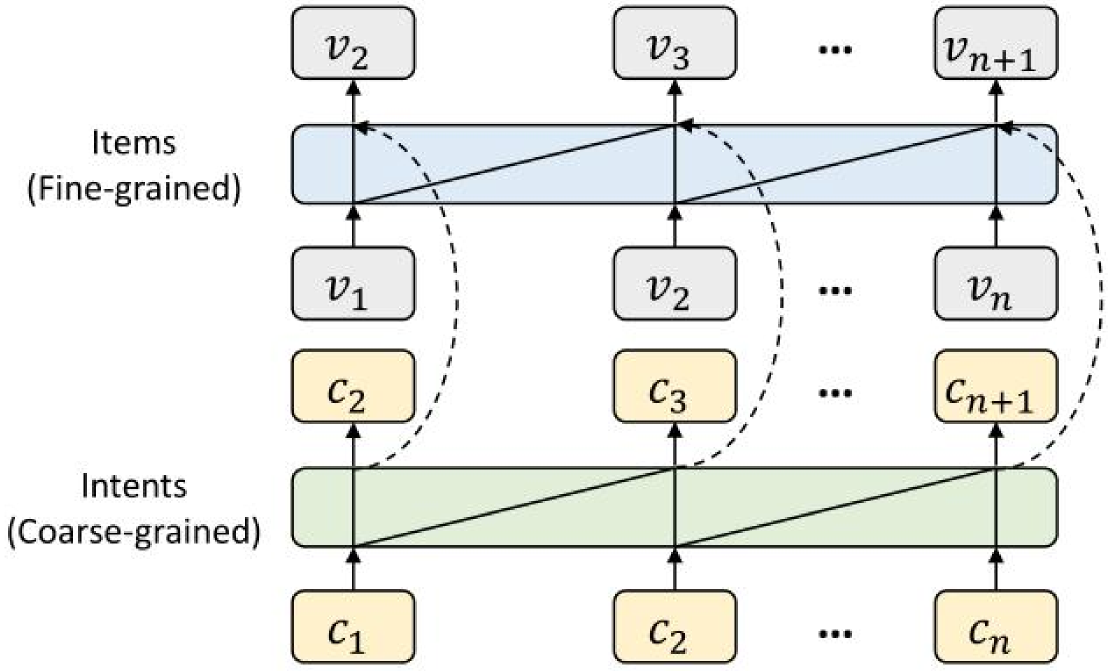


Figure 3: Framework illustration of CaFe.

2.2.1 Embedding. For an item set $\mathcal{V}$, an embedding table $\mathbf{E} \in \mathbb{R}^{d\times|\mathcal{V}|}$ is used for all items, whose element $\mathbf{e}_i \in \mathbb{R}^d$ denote the embedding for item v_i and d is the latent dimensionality. To be aware of item positions, SASRec maintains a learnable position embedding $\mathbf{P} \in \mathbb{R}^{d\times n}$, where n is the maximum sequence length. All interaction sequences are padded to n with a special 'padding' item. Hence, given a padded item sequence $S^v = \{v_1, v_2, \ldots, v_n\}$, the input embedding is computed as:

$$\mathbf{M}^v = \text{Embedding}(S^v) = [\mathbf{e}_1 + \mathbf{p}_1, \mathbf{e}_2 + \mathbf{p}_2, \ldots, \mathbf{e}_n + \mathbf{p}_n] \quad (1)$$

2.2.2 Transformer Encoder. The Transformer encoder adopts scaled dot-product attention [21] denoted as f_{att}. Given $\mathrm{H}_i^l \in \mathbb{R}^d$ is an embedding for v_i after the l^{th} self-attention layer and $\mathrm{H}_i^0 = \mathbf{e}_i + \mathbf{p}_i$, the output from multi-head (#head=M) self-attention is calculated as:

$$\mathbf{O}_i = \text{Concat}[\mathbf{O}_i^{(1)}, \ldots, \mathbf{O}_i^{(m)}, \ldots, \mathbf{O}_i^{(M)}]\mathbf{W}_O, \quad (2)$$

$$\mathbf{O}_i^{(m)} = \sum_{j=1}^{n} f_{\text{att}}(\mathrm{H}_i^l\mathbf{W}_Q^{(m)}, \mathrm{H}_j^l\mathbf{W}_K^{(m)}) \cdot \mathrm{H}_j^l\mathbf{W}_V^{(m)}, \quad (3)$$

where $\mathbf{W}_Q^{(m)}, \mathbf{W}_K^{(m)}, \mathbf{W}_V^{(m)} \in \mathbb{R}^{d\times d/M}$ are the m-th learnable projection matrices; $\mathbf{W}_O \in \mathbb{R}^{d\times d}$ is a learnable matrix to get the output $\mathbf{O}_i$ from concatenated heads. Our backbone SASRec model is a directional self-attention model implemented by forbidding attention weights between v_i and v_j ($j > i$).

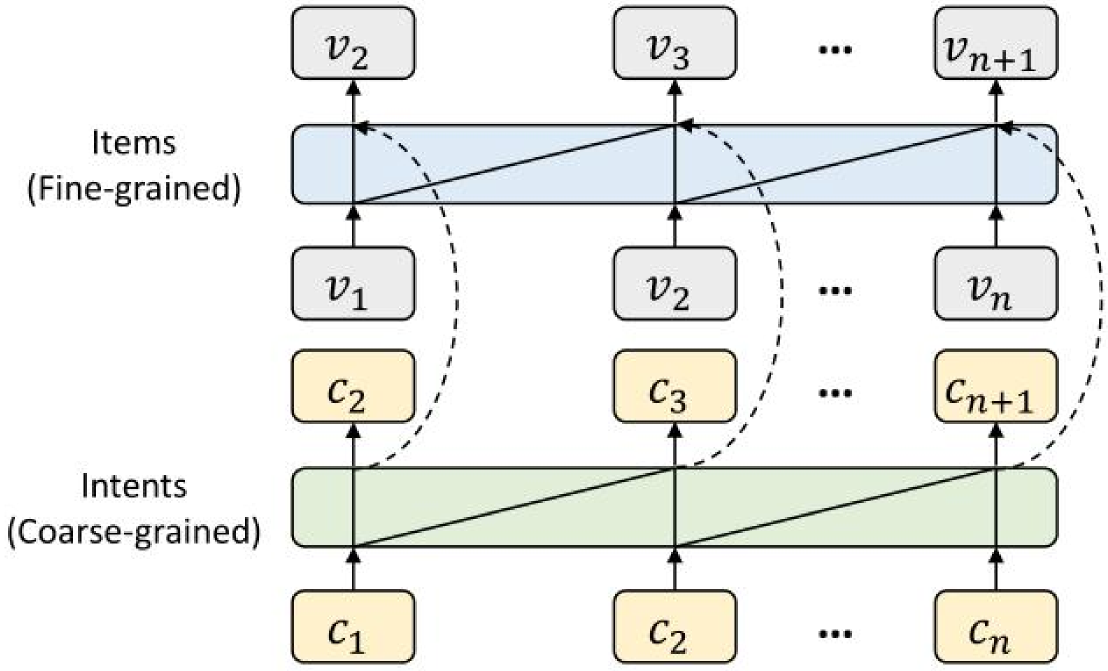


Figure 3: Framework illustration of CaFe.

where $\mathbf{W}_L^{(m)} \in \mathbb{R}^{d/M \times 1}$, $\mathbf{b}_L \in \mathbb{R}^1$, distance embedding $\mathbf{d}_{ij} \in \mathbb{R}^{d/M}$ is the $(n+i-j)$-th vector from distance embedding table $\mathbf{D} \in \mathbb{R}^{d \times 2n}$

$$\mathbf{M}^v = \text{Embedding}^v(S_u^v); \mathbf{M}^c = \text{Embedding}^c(S_u^c), \tag{4}$$

$$f_{\text{att}}(\mathbf{Q}_i, \mathbf{K}_j) = \frac{\exp(w_{ij}) \cdot \theta_{ij}}{\sum_{k=1}^{n} \exp(w_{ik}) \cdot \theta_{ik}}, w_{ij} = \frac{\mathbf{Q}_i \mathbf{K}_j^T}{\sqrt{d}} \tag{5}$$

$$\ln\theta_{ij} = (\mathrm{H}_i^l \mathbf{W}_Q^{(m)} + \mathrm{H}_j^l \mathbf{W}_K^{(m)} + \mathbf{d}_{ij})\mathbf{W}_L^{(m)} + \mathbf{b}_L \tag{6}$$

$$\mathbf{R} = \mathbf{R}^v + \mathbf{R}^c \tag{7}$$

$$r_{j,t}^c = \mathbf{R}_t^c \mathbf{E}_j^{cT}, r_{k,t}^v = \mathbf{R}_t \mathbf{E}_k^{vT} \tag{8}$$

$$\begin{aligned} \mathcal{L} &= \mathcal{L}_c + \mathcal{L}_v \\ &= -\sum_{S_u \in \mathcal{S}} \sum_{1 \le t \le n} \left[\log(\sigma(r_{y^c,t}^c)) + \sum_{c_j \notin S_u} \log(1 - \sigma(r_{c_j,t}^c)) \right] \\ &\quad - \sum_{S_u \in \mathcal{S}} \sum_{1 \le t \le n} \left[\log(\sigma(r_{y^v,t}^v)) + \sum_{v_k \notin S_u} \log(1 - \sigma(r_{v_k,t}^v)) \right] \end{aligned} \tag{9}$$

$$\begin{aligned} P(c_j, v_k | S_u^c, S_u^v, \Theta) &= P(c_j | S_u^c, \Theta) P(v_k | c_j, S_u^c, S_u^v, \Theta) \\ &= \sigma(r_{j,t}^c)\sigma(r_{k,t}^v) \end{aligned} \tag{10}$$

Table 1: Data statistics.

Datasets	#Interaction	#Item	#Intent	#Sequence	Ave. Length	Density
Amazon	5,370,171	1,910,226	1,392	131,248	40.9	2e-5
Tmall	14,460,516	1,788,758	9,999	131,086	110.3	6e-5

Table 2: Model comparision. The best results are bold and the best baselines are underlined.

		Item-only Methods					Intent-aware Methods					
Dataset	Metric	PopRec	SASRec	BERT4Rec	SSE-PT	LOCKER	NOVA	FDSA	BERT-F	LOCKER-F	CaFe	Improvement
Amazon	NDCG@5	0.0286	0.1418	0.1830	0.2108	0.2170	0.0281	0.0670	0.2199	<u>0.2436</u>	**0.3733**	+53.24%
	HR@5	0.0487	0.1844	0.2240	0.2501	0.2597	0.0475	0.1089	0.2676	<u>0.2947</u>	**0.4813**	+63.32%
	MRR	0.0485	0.1522	0.1956	0.2239	0.2297	0.0477	0.0857	0.2329	<u>0.2529</u>	**0.3656**	+44.56%
Tmall	NDCG@5	0.0360	0.0741	0.2753	0.2106	0.2961	0.0501	0.1083	0.2998	<u>0.3182</u>	**0.4290**	+34.82%
	HR@5	0.0596	0.1205	0.3673	0.2977	0.3872	0.0812	0.1685	0.3917	<u>0.4098</u>	**0.5152**	+25.72%
	MRR	0.0577	0.0948	0.2782	0.2173	0.2979	0.0716	0.1265	0.3014	<u>0.3189</u>	**0.4268**	+33.84%

Experiment

	Backbone (SASRec)	+(1) (FDSA)	+(1)(2)	+(1)(2)(3)	+(1)(2)(4)	+(1)(2)(3)(4) (CaFe)
NDCG@5	0.0741	0.1083	0.3045	0.3159	0.4254	**0.4290**
HR@5	0.1205	0.1685	0.3938	0.4066	0.5117	**0.5152**
MRR	0.0948	0.1265	0.3069	0.3172	0.4239	**0.4268**

Table 3: Ablation study on Tmall dataset. (1) fusing intents into item embeddings; (2) modeling intents explicitly; (3) local self-attention of item encoder; (4) inference with joint probability distribution of items and corresponding intents.

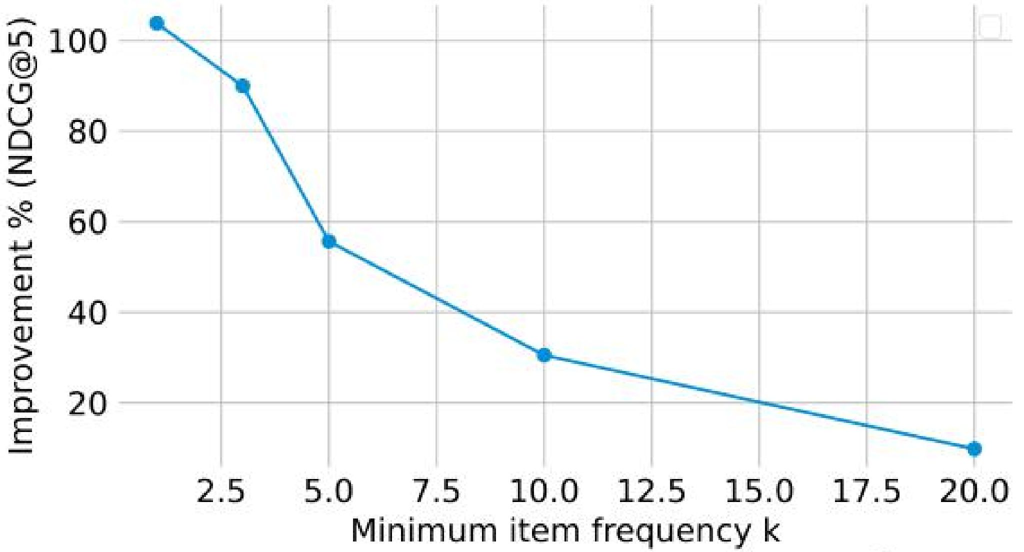


Figure 4: Improvement on Amazon compared to BERT4Rec.

Thanks