Two Tower Model with Self-Supervision Learning

Friday, March 15, 2024 3:50 PM

Two Tower Model Issues:

① A few popular items takes most clicks

② Large number of items get a few clicks.

③ The representation vector of popular items is learned well

However, the vector of unpopular items is poorly learned.

Take large fraction

How to solve? data argumentation

Two Tower Model Training:

Batch Examples:

users	items
#1	#1
#2	#2
⋮	⋮
#N	#N

Positive Samples: (user ID)

$(1,1), (2,2), \ldots (N,N)$ (N: item ID)

Negative Samples:

$(1,2), (1,3) \cdots (1,N)$

$(2,1), (2,3) \cdots (2,N)$

$(N,1) \quad (N,2) \cdots (N,N-1)$

# of positive samples: N # of negative: $N(N-1)$

objective: make $\cos(a_i, b_+)$ large; make $\cos(a_i, b_-)$ small

Two Tower Model with Self-Supervision Learning

Friday, March 15, 2024 3:50 PM

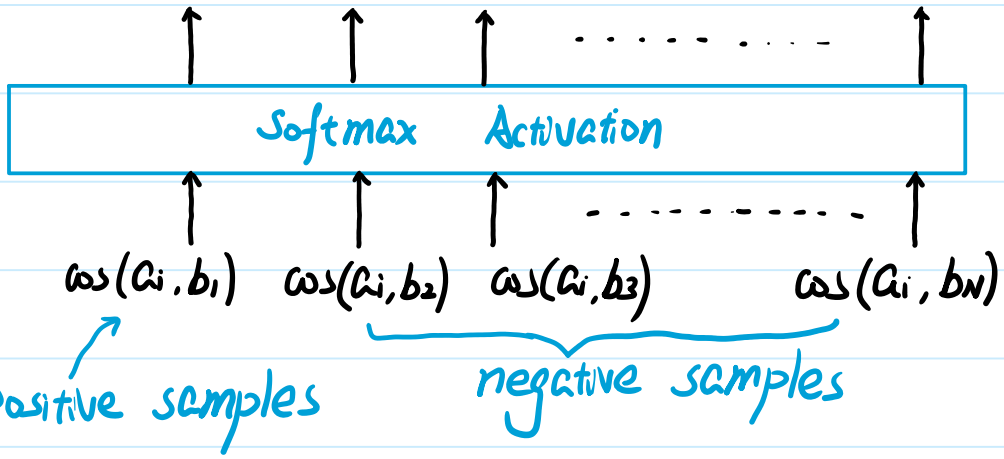
loss function:

observation: y_i: 1, 0, 0, ... 0

positive sample

negative sample

prediction: P_i: $P_{i,1}$ $P_{i,2}$ $P_{i,3}$ ... $P_{i,N}$



Cross Entropy Loss (y_i, P_i)

$$L = -\log P_{i,i} = -\log \frac{\exp(\cos(a_i, b_j))}{\sum_{j=1}^{n} \exp(\cos(a_i, b_j))}$$

Probability of item j being selected:

$$P_j \propto (\#\text{ of click})$$

Estimated like from user i to item j: $\cos(a_i, b_j)$

Corrected like: $\cos(a_i, b_j) - \log P_j$

Two Tower Model with Self-Supervision Learning

Friday, March 15, 2024 3:50 PM

Training Process:

① Randomly select N users and their interacted items

$(a_1, b_1), (a_2, b_2), \cdots (a_N, b_N)$ → one batch

② Loss function:

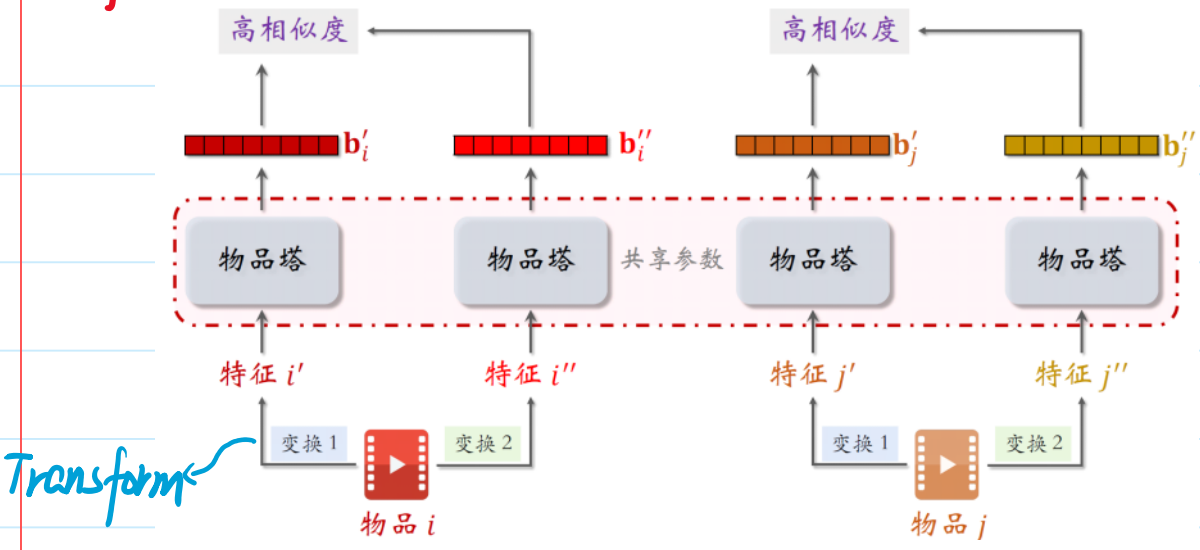
$$L_{main}[i] = -\log \frac{\exp[\cos(a_i, b_i) - \log P_i]}{\sum_{j=1}^{n} \exp[\cos(a_i, b_j) - \log P_j]}$$

for user i

③ gradient descent:

$$\frac{1}{N} \sum_{i=1}^{N} L_{main}[i]$$

Self-supervision:



(Picture from Shusen Wang on Youtube/Bilibili)

① make $\cos(b_i', b_i'')$ and $\cos(b_j', b_j'')$ large

② make $\cos(b_i', b_j')$ and $\cos(b_i', b_j'')$ small

same item: high similarity; different item: low similarity

Two Tower Model with Self-Supervision Learning

Friday, March 15, 2024 3:50 PM

Transform #1: Random mask

① select some categorical features and mask them

e.g. item #1: genre = (photo, beauty, travel)

→ mask: genre = ("missing")

Transform #2: Drop out.

① randomly drop part of features

e.g. item # genre = (photo, beauty, travel)

→ drop: genre = (photo)

Transform #3: Complementary:

e.g. items: (ID, genre, hashtag, city)

two groups { (ID, hashtag) / (genre, city)

each item has two representation:

(ID, "missing", hashtag, "missing") → item tower

("missing", genre, "missing", city) → item tower

vector representation similar

Two Tower Model with Self-Supervision Learning

Friday, March 15, 2024 3:50 PM

Transform #4: mask correlated features.

① gender: U = {male, female, neutral}

② genre: V = {cosmetics, digit, football, tech}

③ u = male and v = digit probability is large

④ u = male and v = cosmetics probability is small.

mutual information:

$$MI(U,V) = \sum_{u \in U} \sum_{v \in V} P(u,v) \cdot \log \frac{P(u,v)}{P(u) \cdot P(v)}$$

k features → k × k matrix for MI

① random select one feature as "seed feature"

② get most relevant k/2 features

③ mask "seed" and k/2 features

Pros: works best

Cons: implementation is difficult; hard to maintain

Overall: NOT efficie

Two Tower Model with Self-Supervision Learning

Friday, March 15, 2024 3:50 PM

Model Training: for self supervision

① uniformly select m items (NOT based on # of clicks)

② perform two types of transformations

item tower generates two groups of vectors

$(b_1', b_2', \cdots b_m')$ $(b_1'', b_2'', \cdots b_n'')$

③ loss function for item i:

$$L_{self}[i] = -\log \frac{\exp(\cos(b_i', b_i''))}{\sum_{j=1}^{m} \exp(\cos(b_i', b_j''))}$$

$\cos(b_i', b_j'')$: i and j similarity

④ objective function:

$$\frac{1}{m} \sum_{i=1}^{m} L_{self}[i]$$

Self supervision + Two Tower Model:

① random sample n (user, item) from clicked items

② uniform sample m items from all items

③ gradient descent:

$$\frac{1}{n} \sum_{i=1}^{n} L_{main}[i] + \alpha \frac{1}{m} \sum_{j=1}^{m} L_{self}[j]$$

two tower | hyper parameter | self supervision