AliGraph: An Extremely Large Scale Graph Representation Learning in Practice

Xiaoyong Liu
Hongxia Yang
Wei Lin

Alibaba Group

Part 1：About Alibaba

Our Mission

让天下没有难做的生意
To make it easy to do business anywhere
Taobao
TMALL
聚
AliExpress
AliExpress
alibaba.com
Alibaba Group
1688.COM

Our Vision

MEET

Enabling millions of commercial and
social interactions

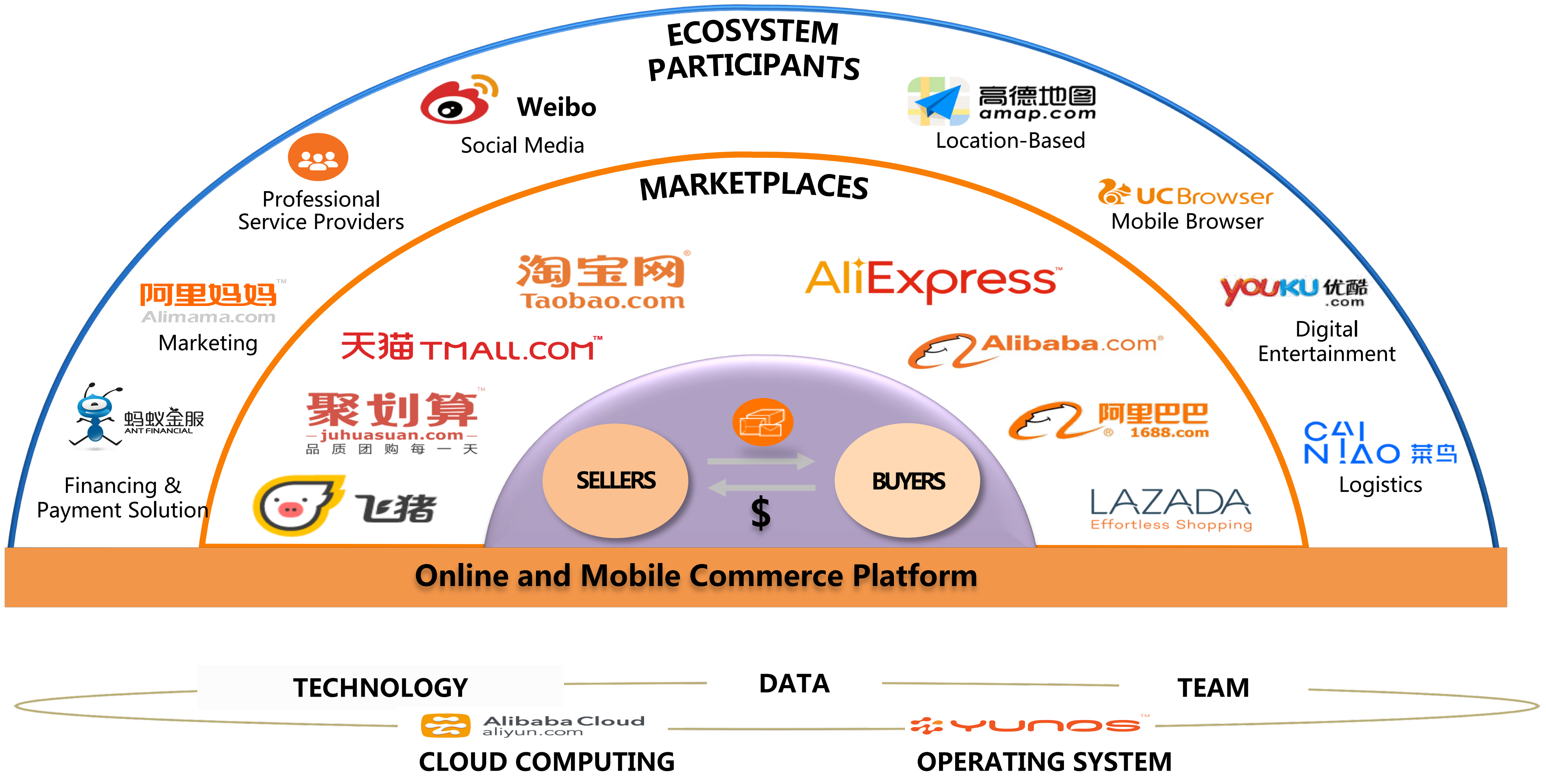
WORK

Empowering our customers with data and infrastructure to manage their business

LIVE

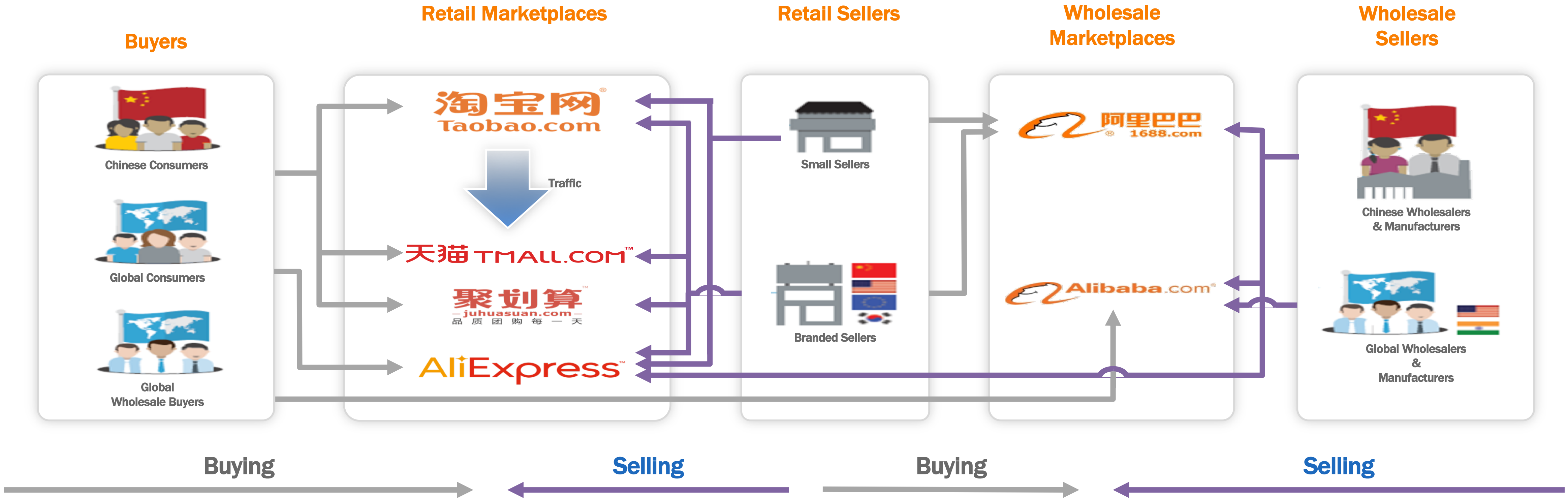
To become central to the everyday lives of our customers

Alibaba Ecosystem



Our Marketplaces are Networked

Alibaba Group 阿里巴巴集团



Cross Promotion Across Our China Retail Marketplaces

Strong Synergies Between Our Retail & Wholesale Marketplaces

Part 2：Why GE and Two works

Why Graph and Graph Embedding?

- Graph computing models are very popular in big data companies, especially IT companies, as they are the most straightforward solutions to many practical problems

- Traditional recommendation and CTR/CVR prediction problems can be equivalently modeled with the attributed user-item bipartite graphs

- Objective functions are more general: global optimization vs conditional independent optimization

- Introducing high-level proximity samples and their modeling brings both risks (e.g., noise) and benefits (e.g., more generalization and exploration, predictive graph changes)

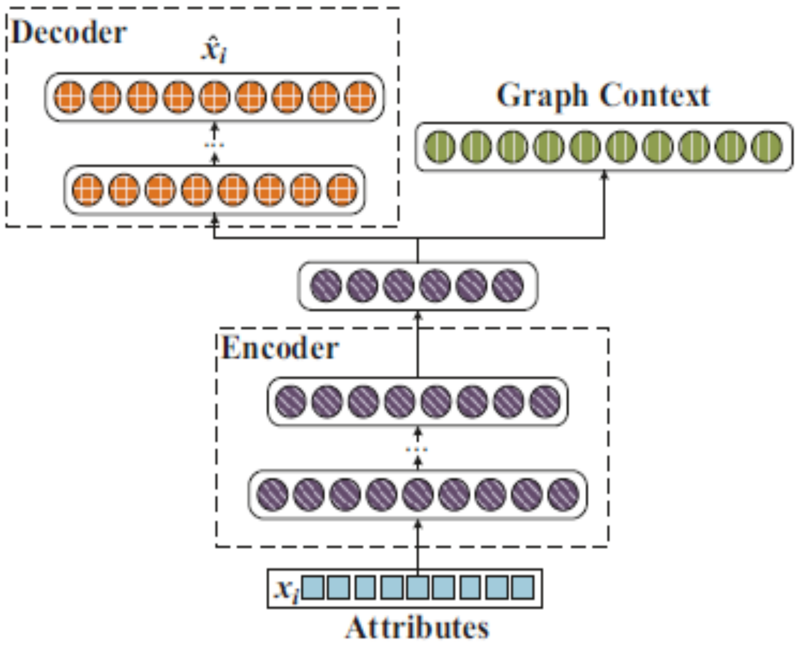
- Pure deep learning is mature, graph embedding that combines both deep learning and graph computing integrating end-to-end learning with inductive reasoning, which is expected to solve the relational reasoning that deep learning cannot perform[1].

[1] Relational inductive biases, deep learning and graph networks, Battaglia etc, arxiv, 2018.

Representative Work: Graph Embedding in Fraud Detection[1]

Challenges of mobile fraud detection:

- Billions of mobile logging records per day
- Device ids are missing
- Devices are abnormal:
    - Dual sim cards dual standby: one device has two imei, imsi[2] with random switching
    - Replace the sim card: one device corresponds to multiple imsi
    - System Restore: One device corresponds to multiple device ids
    - Simulator: One device corresponds to a large number of device ids
    - Cottage machine: a large number of devices share one same id
- Devices contain rich attributes



We propose a distributed deep learning-based representation learning model, ANRL, which maps nodes in the graph to representation vectors in low-dimensional space to facilitate subsequent processing of related tasks

- Combining both network structure and node attribute information to better maintain network characteristics
- Neural networks can mine deeper correlations between the two

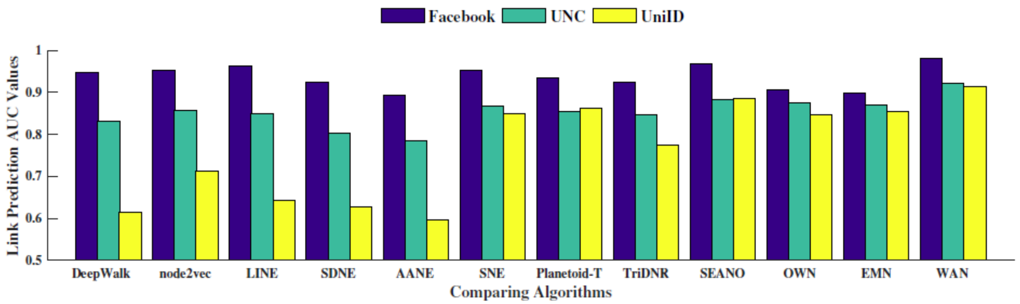
[1] **ANRL: Attributed Network Representation Learning via Deep Neural Networks, IJCAI 2018**
[2] **imei: international mobile equipment identity; imsi: international mobile subscriber identity**

Representative Work: Graph Embedding in Fraud Detection

Alibaba Group 阿里巴巴集团

- Objective Functions:

$$
\begin{aligned}
\mathcal{L} &= \mathcal{L}_{sg} + \alpha \mathcal{L}_{ae} + \beta \mathcal{L}_{reg} \\
&= -\sum_{i=1}^{n} \sum_{c \in C} \sum_{-b \le j \le b, j \ne 0} \log \frac{\exp(\mathbf{u}_{i+j}^{\mathrm{T}} \mathbf{y}_i^{(K)})}{\sum_{v=1}^{n} \exp(\mathbf{u}_v^{\mathrm{T}} \mathbf{y}_i^{(K)})} \\
&+ \alpha \sum_{i=1}^{n} \|\hat{\mathbf{x}}_i - T(v_i)\|_2^2 + \frac{\beta}{2} \sum_{k=1}^{K} (\|\mathbf{W}^{(k)}\|_F^2 + \|\hat{\mathbf{W}}^{(k)}\|_F^2)
\end{aligned}
$$



- Existing methods generally incorporate attribute information into representation learning by matrix decomposition or by adding auxiliary nodes. This shallow level model can't capture the deep inner relationship between the two.

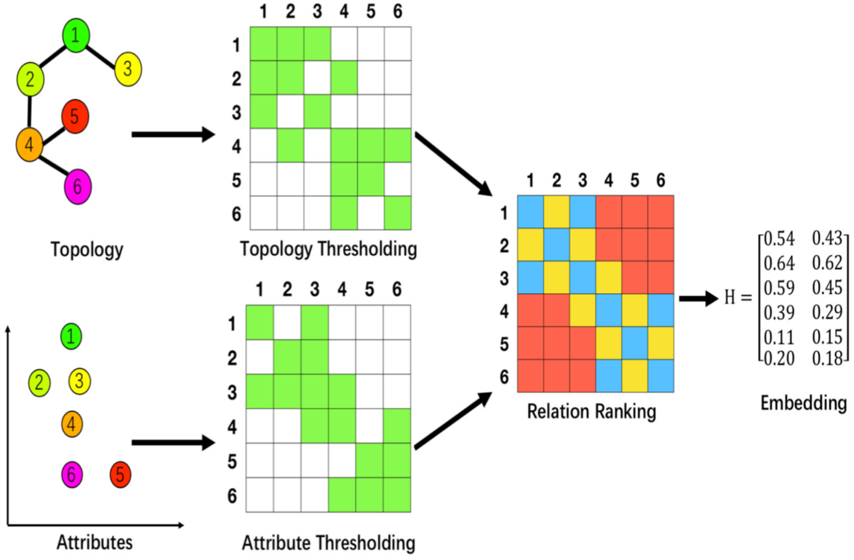
- Compared with the existing state-of-the-art method, our model ANRL has a certain degree of improvement in tasks such as ink prediction and node classification by using the deep learning model.

Datasets	Citeseer		Pubmed		Fraud Detection	
Evaluation	Micro-F1	Macro-F1	Micro-F1	Macro-F1	Micro-F1	Macro-F1
SVM	0.667	0.626	0.856	0.855	0.725	0.719
autoencoder	0.630	0.565	0.792	0.800	0.732	0.726
DeepWalk	0.583	0.534	0.809	0.795	0.509	0.464
node2vec	0.607	0.561	0.815	0.802	0.571	0.519
LINE	0.542	0.512	0.766	0.749	0.659	0.654
SDNE	0.569	0.528	0.699	0.677	0.662	0.656
AANE	0.579	0.541	0.784	0.765	0.654	0.643
SNE	0.632	0.615	0.803	0.797	0.662	0.654
Planetoid-T	0.656	0.594	0.851	0.847	0.692	0.693
TriDNR	0.633	0.587	0.843	0.824	0.686	0.685
SEANO	0.713	0.662	0.859	0.848	0.703	0.704
ANRL-OWN	0.652	0.606	0.842	0.845	0.724	0.720
ANRL-EMN	0.716	0.668	0.865	0.867	0.733	0.731
ANRL-WAN	**0.729**	**0.673**	**0.876**	**0.871**	**0.759**	**0.755**

Representative Work: Graph Embedding in Entity Recognition[1]

Challenges of mobile entity recognition:

- Data contains large-scale sparse network structure information and rich attribute information
- Both information can be used to describe the relationship among the nodes, however distance measures of the two are probably not monotonically increasing or decreasing of the same nodes
- Consider a non-positive relationship, we can capture more information to help improve the learning performance



We propose to investigate attributed network embedding through taking uncorrelation between topology structure and attribute into account and make the following contributions.

- We recognize the uncorrelation phenomenon between topology structure and attribute which affects the actual proximity among nodes in the embedding space.
- We propose a Personalized node Relation Ranking Embedding (PRRE) model such that the resulting network embedding can capture the uncorrelation as well as preserve good proximity among nodes.
- Sampling with Mini-batch Gradient Descent for efficient iteration and update.

[1] **PRRE: Personalized Relation Ranking Embedding for Attributed Networks, CIKM 2018**

Representative Work: Graph Embedding in Entity Recognition

- Objective function and algorithm framework

$$\mathcal{J}(H,\theta_T,\theta_A) = \prod_{i\in\mathcal{V}}\Bigg(\prod_{p\in P}\prod_{a\in A} P(p \geq_i a|\theta_A,\theta_T,H) \prod_{a\in A \prod_{n\in N}} P(a \geq_i n|\theta_A,\theta_T,H)\Bigg).$$

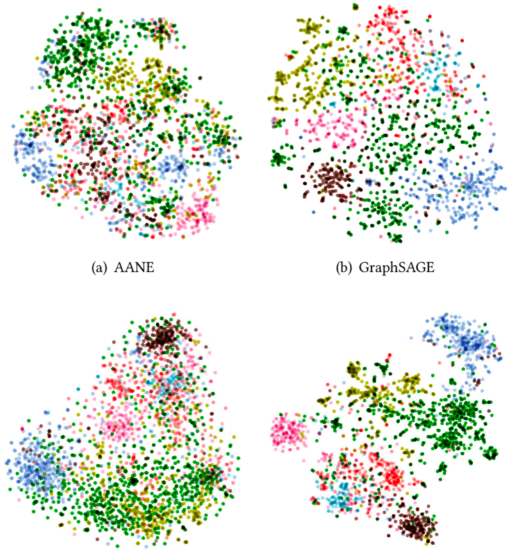
$$\mathcal{J}(H,\theta_T,\theta_A) = ln\mathcal{J}(H,\theta_T,\theta_A) - \lambda_h \sum_{i\in\mathcal{V}} ||h_i||^2$$

$$= \sum_{i\in\mathcal{V}}\Bigg(\sum_{p\in P}\sum_{a\in A}\frac{1}{1+g(\theta_T,\theta_A)} ln\frac{\sigma_{ip}-\sigma_{ia}+1}{2} + \sum_{a\in A}\sum_{n\in N}\frac{1}{1+g(\theta_T,\theta_A)} ln[\frac{\sigma_{ia}-\sigma_{in}+1}{2}]\Bigg) - \lambda_h \sum_{i\in\mathcal{V}} ||h_i$$

Algorithm 1 PRRE for attributed networks

Input: $G = \{\mathcal{V}, E, A\}, d$
Output: $H \in R^{n\times d}$

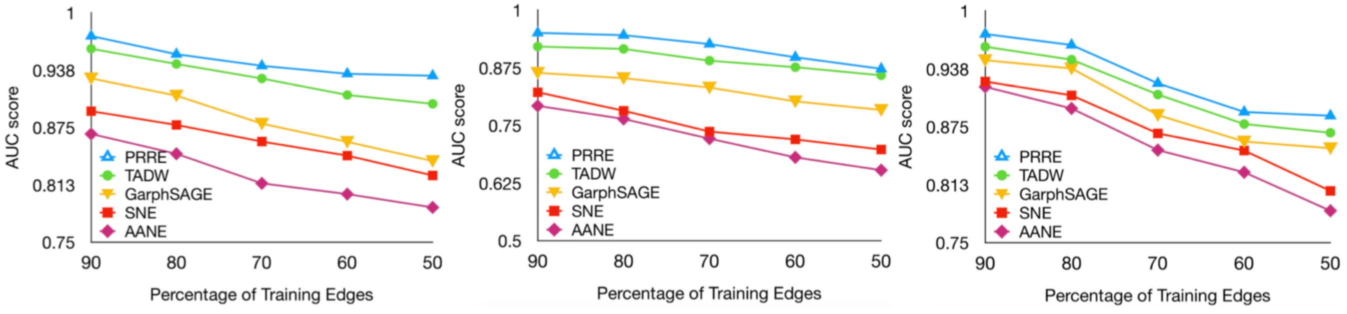
Compute similarity matrix $S_A \in R^{n\times n}$ and $S_T \in R^{n\times n}$ using selected similarity measure
Initialize θ_A, θ_T and $H \sim U(0,1)$
while t < max_iter and $\triangle\mathcal{J} < \epsilon$ **do**
    Sample batch B with size s,
    for $v_i \in \mathcal{V}$ **do**
        Compute Positive/Ambiguous/Negative pairs using current thresholds θ_T, θ_A
    end for
    Compute the gradients for h_i, h_p, h_a, h_n by Equation [13], [14], [15], [16].
    Update H by Equation [19]
    Compute the gradients for θ_T, θ_A by Equation [17],[18].
    Update θ_T, θ_A by Equation [19]
end while



(a) AANE (b) GraphSAGE

(c) SNE (d) PRRE

Nodes are mapped into the 2-D space using t-SNE with learned embeddings



(a) Citeseer (b) Cora (c) Pubmed

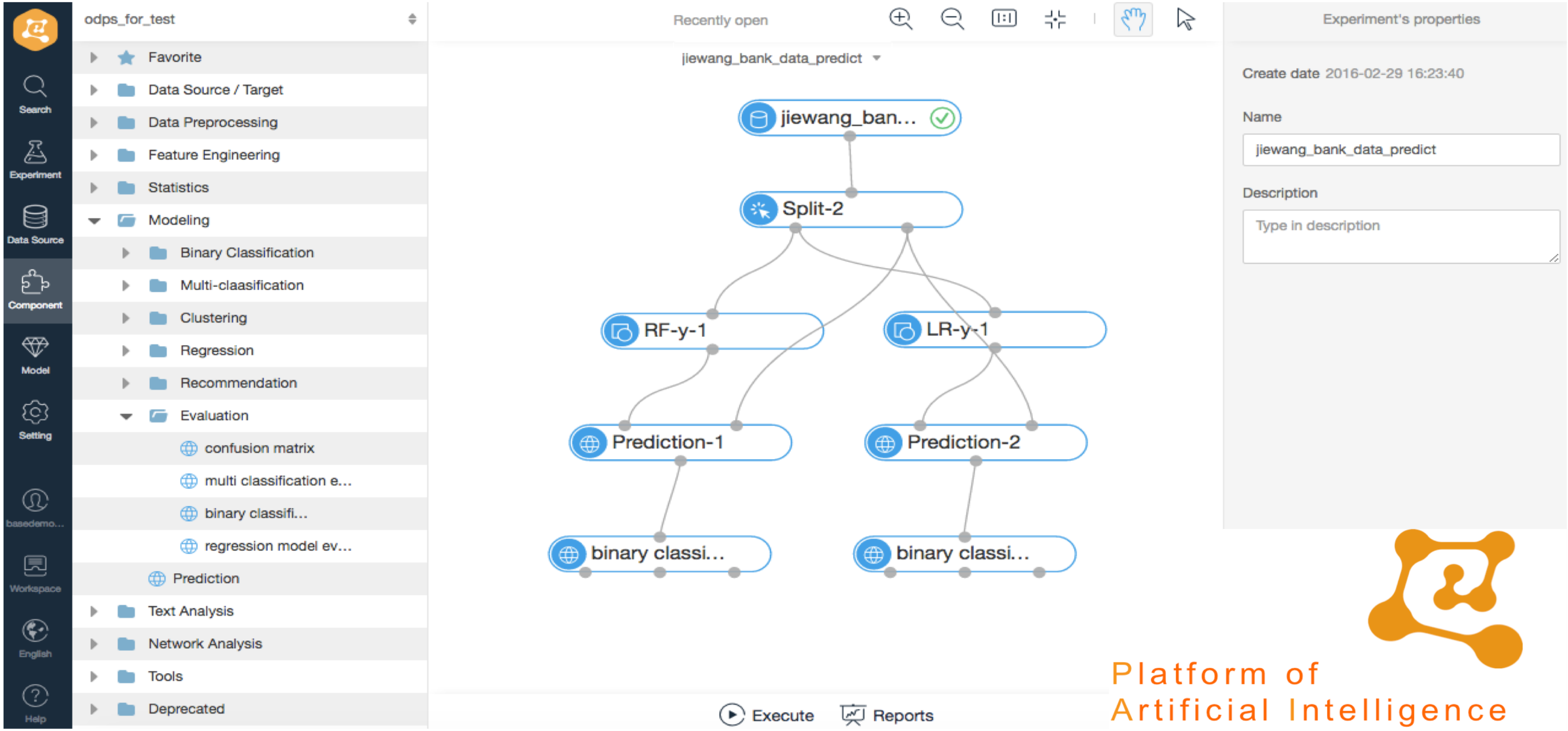
Other Graph Embedding Representative Works

Alibaba Group 阿里巴巴集团

1. Subgraph-augmented Path Embedding for Semantic User Search on Heterogeneous Social Network, WWW, 2018.
2. ANRL: Attributed Network Representation Learning via Deep Neural Networks, IJCAI, 2018.
3. Adversarial Detection with Model Interpretation. **KDD, 2018.**
4. SPARC: Self-Paced Network Representation for Few-Shot Rare Category Characterization. **KDD, 2018.**
5. Mobile access record resolution on large-scale identifier-linkage graphs. **KDD, 2018.**
6. Interactive Paths Embedding for Semantic Proximity Search on Heterogeneous Graphs. **KDD, 2018.**
7. PRRE: Personalized Relation Ranking Embedding for Attributed Network. 27th ACM International Conference on Information and Knowledge Management (CIKM), 2018.
8. Heterogeneous Embedding Propagation for Large-scale E-Commerce User Alignment, 2018 IEEE International Conference on Data Mining (ICDM), 2018
9. Local Algorithm for User Action Prediction Towards Display Ads. 23rd ACM SIGKDD Conference on Knowledge Discovery and Data Mining **(KDD), 2017**.
10. Hybrid Framework for Text Modeling with Convolutional RNN. 23rd ACM SIGKDD Conference on Knowledge Discovery and Data Mining **(KDD), 2017**.
11. Bayesian Heteroscedastic Matrix Factorization for Conversion Rate Prediction. 26th ACM International Conference on Information and Knowledge Management (CIKM), 2017.
12. Will Triadic Closure Strengthen Ties in Social Networks?, ACM Transactions on Knowledge Discovery from Data (TKDD), 2017.

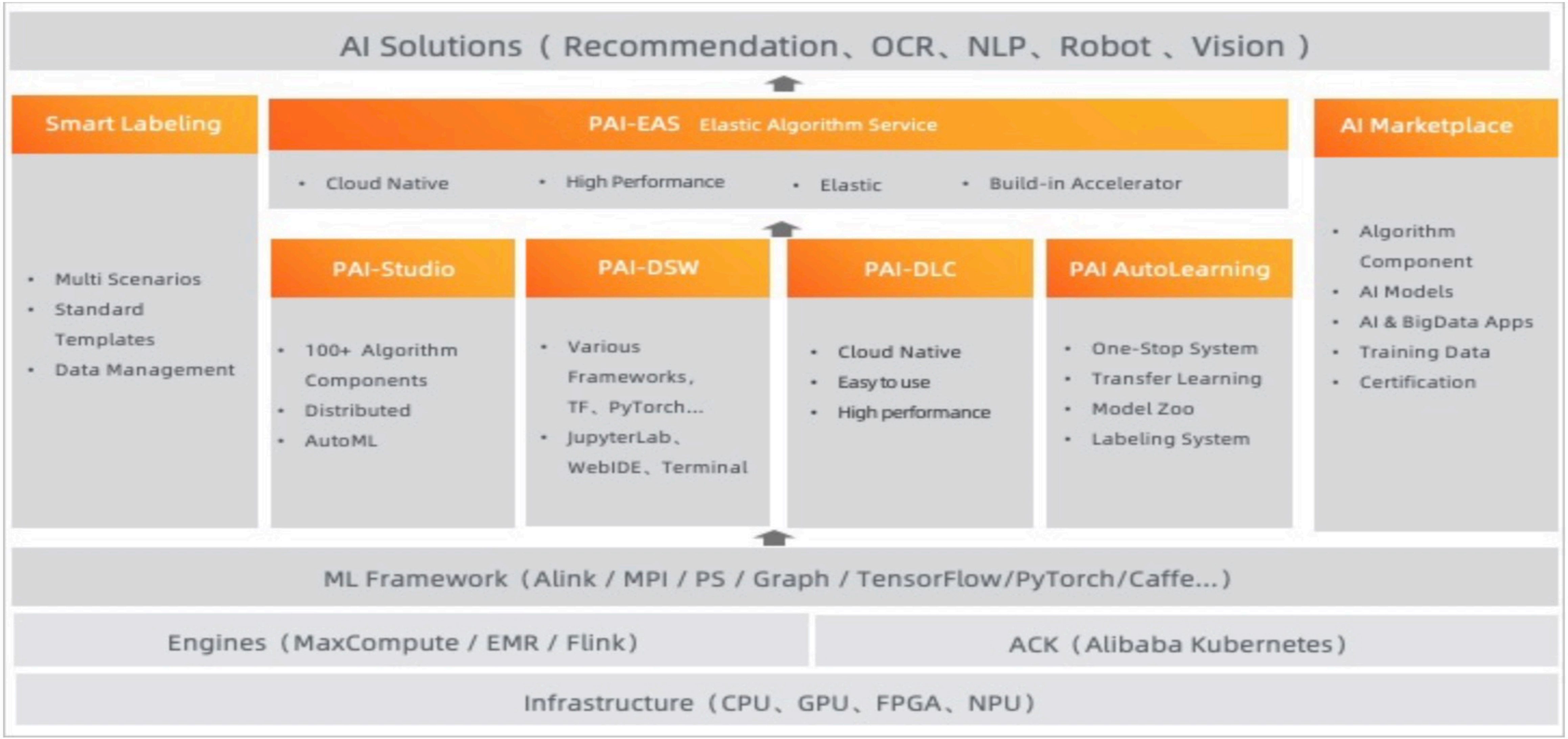
Part 3：PAI - AliGraph Algorithm System

PAI : Alibaba Machine Learning Platform of AI



Platform of
Artificial Intelligence

PAI Architecture



https://www.alibabacloud.com/product/machine-learning

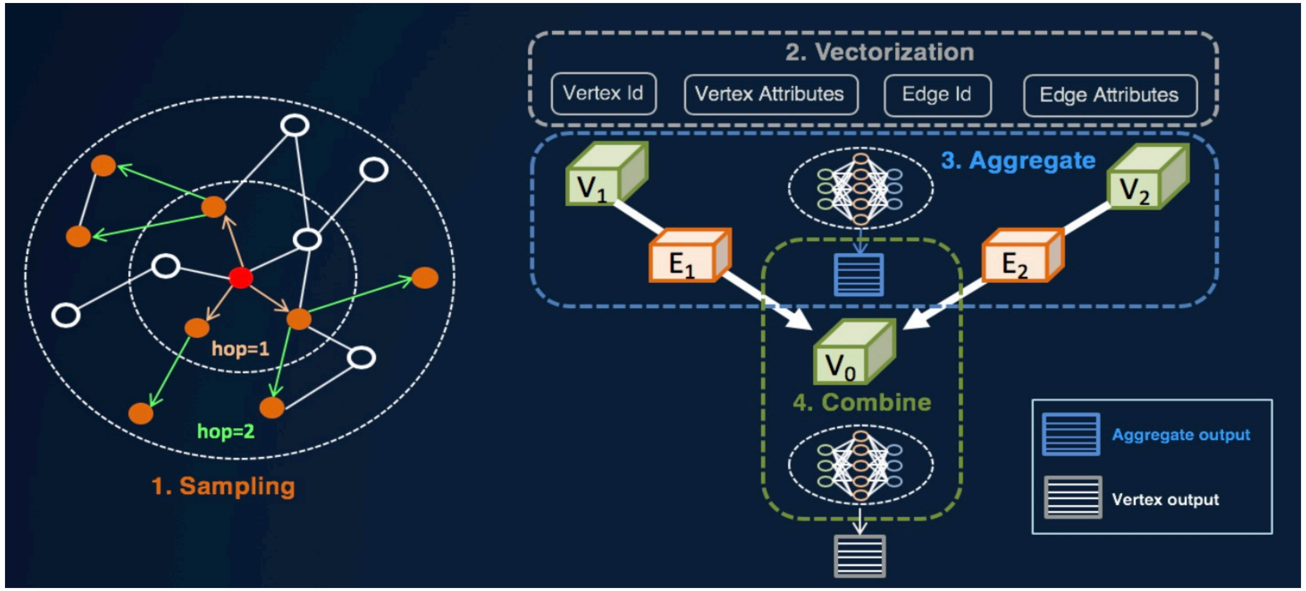
PAI-AliGraph Algorithm Framework

Algorithm 1: GNN Framework

Input: network $\mathcal{G}$, embedding dimension $d \in \mathbb{N}$, a vertex feature $\mathbf{x}_v$ for each vertex $v \in \mathcal{V}$ and the maximum hops of neighbors $k_{max} \in \mathbb{N}$.

Output: embedding result $\mathbf{h}_v$ of each vertex $v \in \mathcal{V}$

```
h_v^(0) ← x_v
for k ← 1 to k_max do
    for each vertex v ∈ V do
        S_v ← SAMPLE(Nb(v))
        h'_v ← AGGREGATE(h_u^(k-1), ∀u ∈ S)
        h_v^(k) ← COMBINE(h_v^(k-1), h'_v)
    normalize all embedding vectors h_v^(k) for all v ∈ V
h_v ← h_v^(k_max) for all v ∈ V return h_v as the embedding result for all
  v ∈ V
```

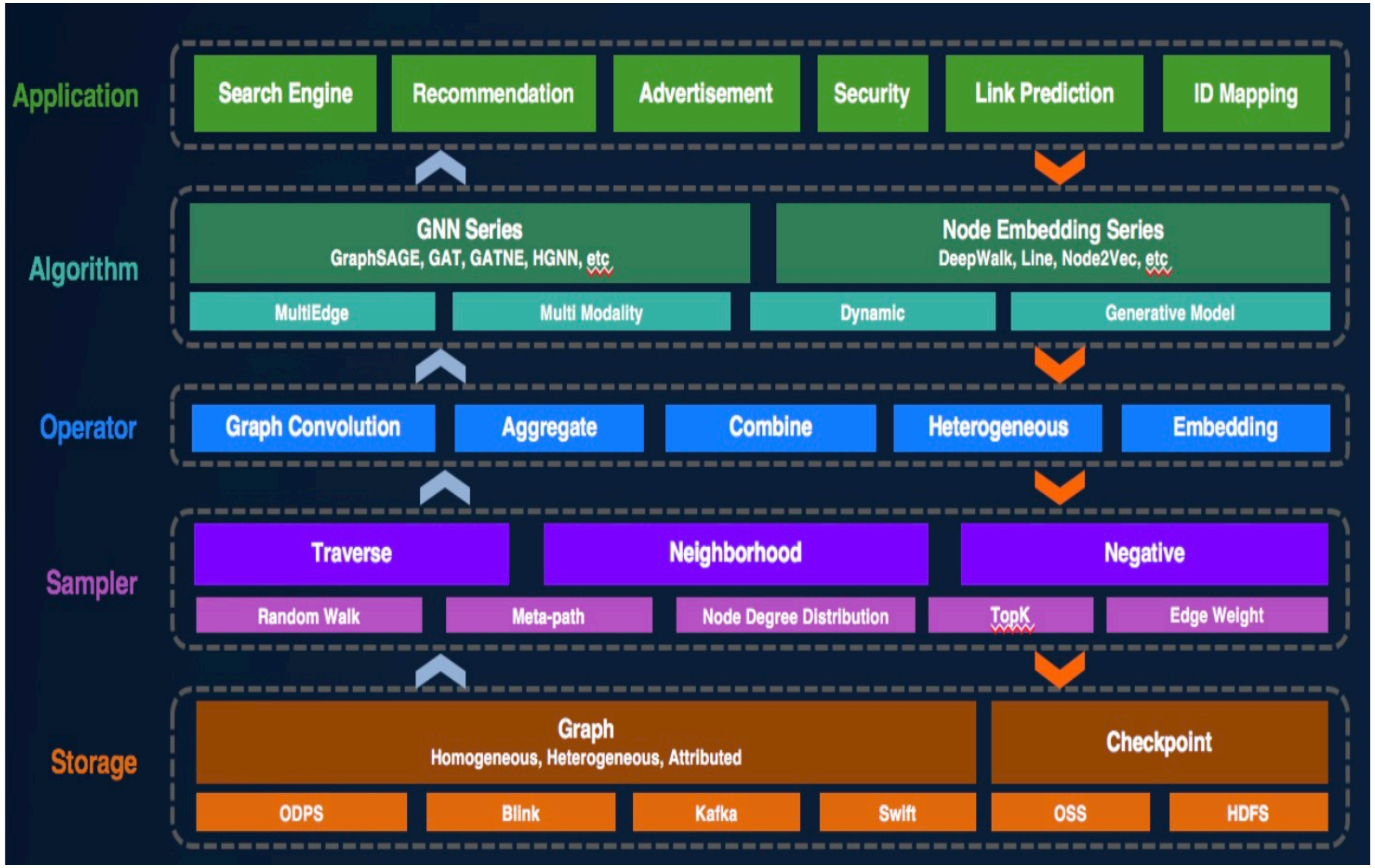


- Business needs to carry out more model innovations based on the AliGraph Algorithm Framework

- Need more flexible programming framework to support algorithm innovation
  - PAI Tensorflow

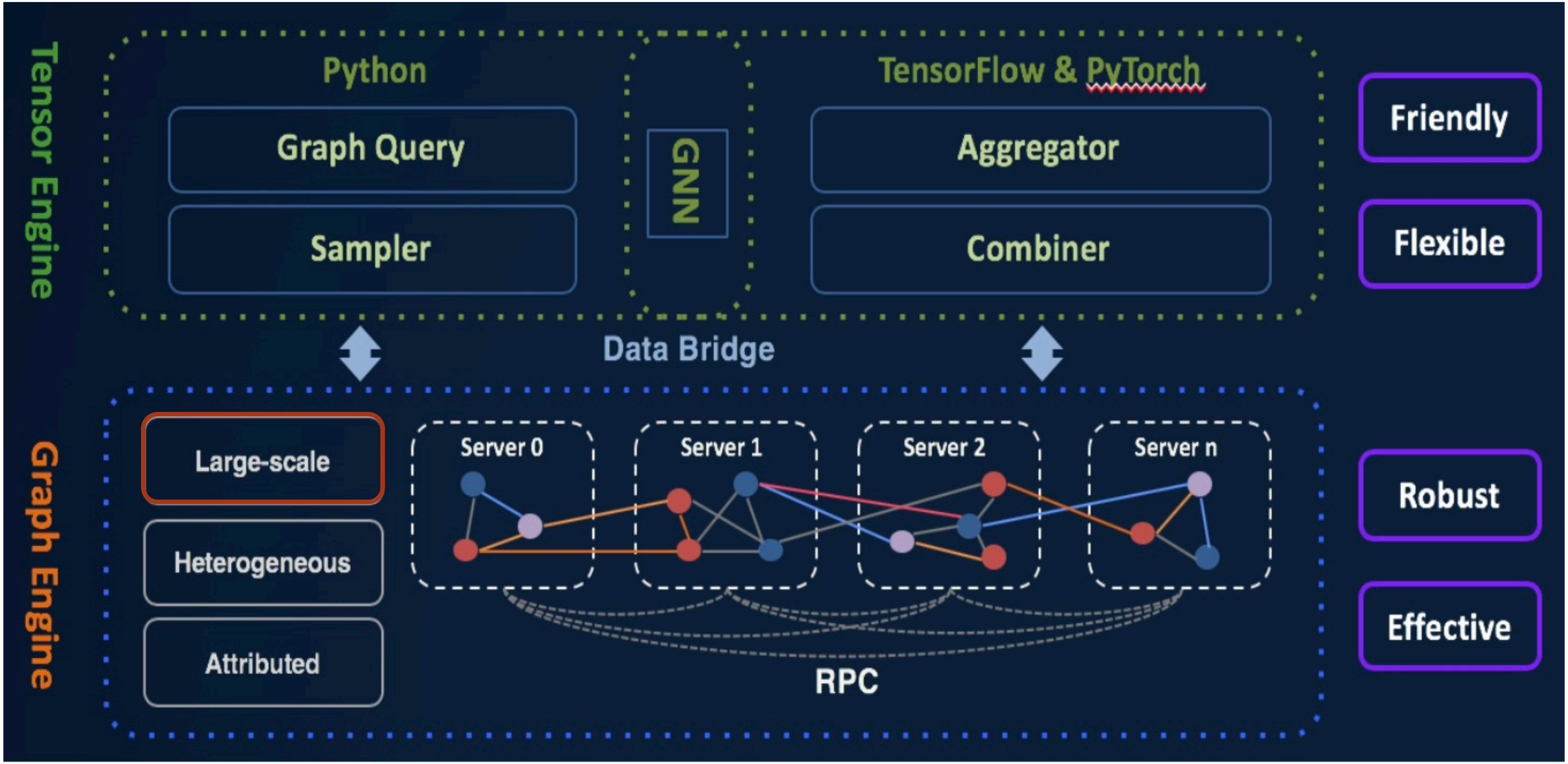
AliGraph-System Overview

- Graph storage + sampling + operator need to be tightly combined for deep learning together

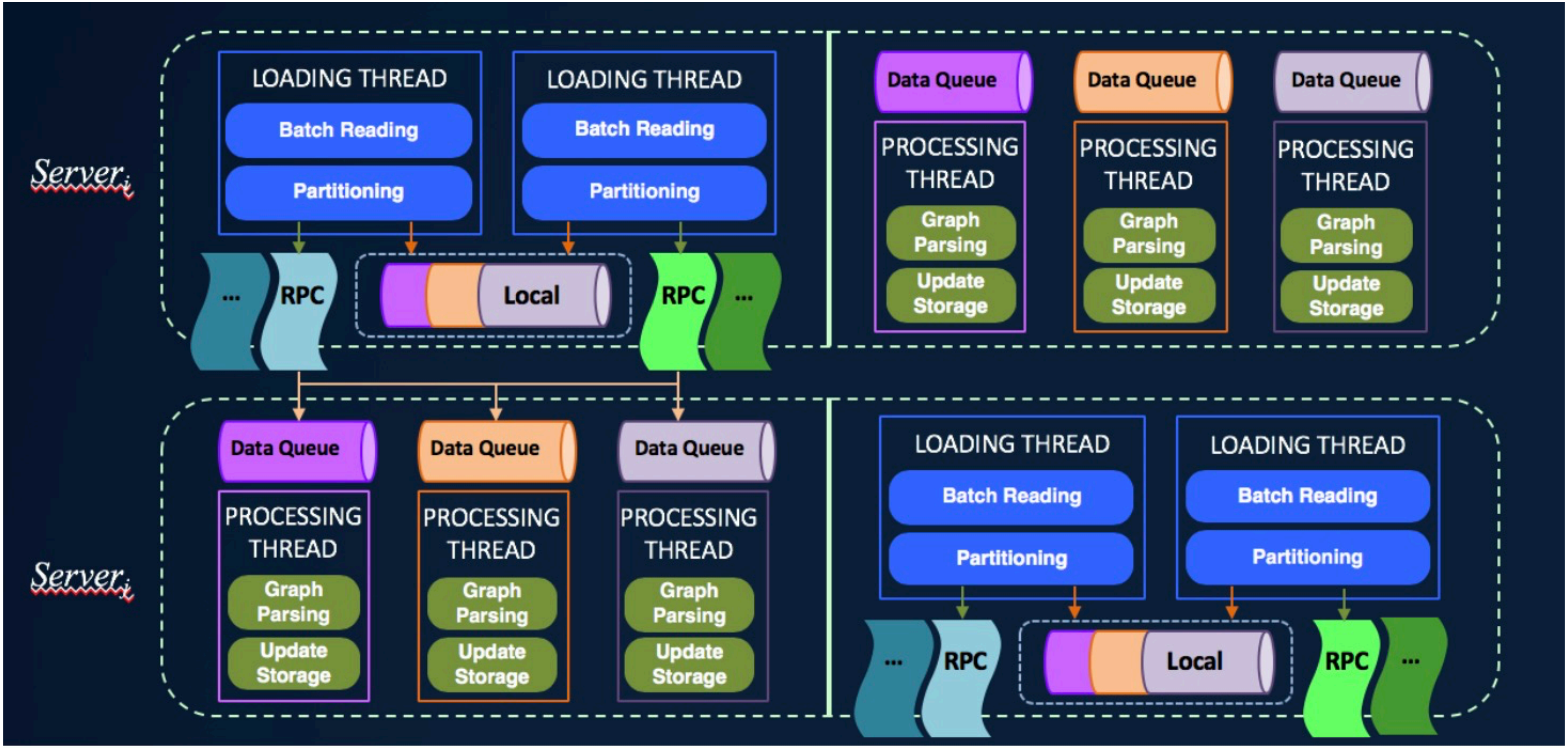


- Aligraph Algorithm Framework based on PAI-Tensorflow extension
  - Use TF's flexible composition method
  - Use TF's good design structure for effective expansion
  - Most of the existing algorithm innovations are based on TF

AliGraph-The Flow



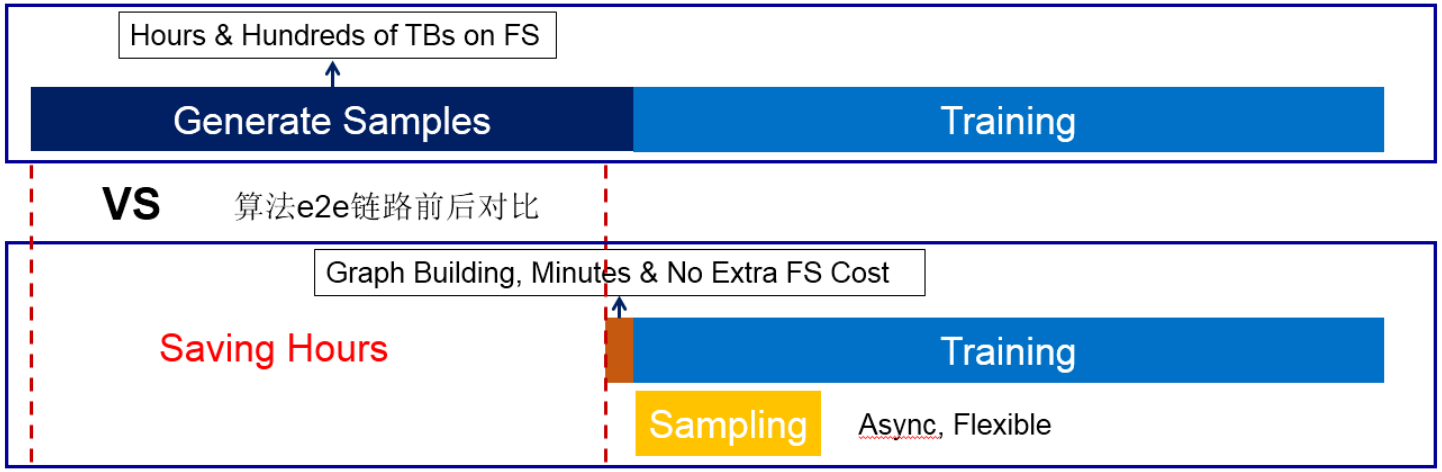
AliGraph-Graph Building



Sample

- Various sampling modes require flexible support
- Provide traverse, neighborhood, negative three ways
- Overlap sampling and calculation

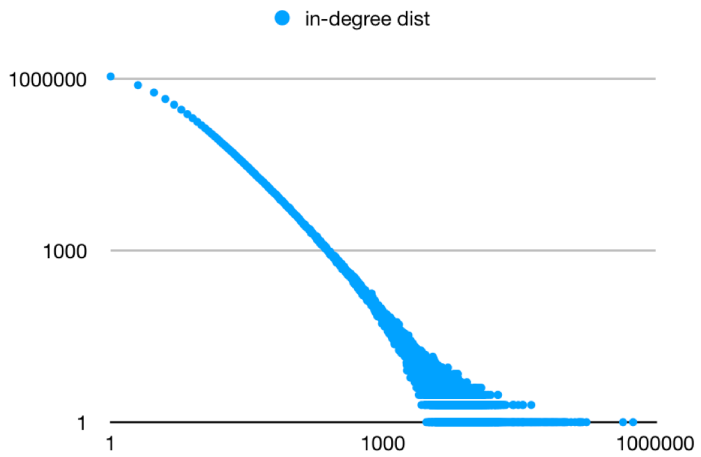
```
Define a TRAVERSE sampler as s1
Define a NEIGHBORHOOD sampler as s2
Define a NEGATIVE sampler as s3
...
def sampling(s1, s2, s3, batch_size):
    vertex = s1.sample(edge_type, batch_size)
    # hop_nums contains neighbor count at each hop
    context = s2.sample(edge_type, vertex, hop_nums)
    neg = s3.sample(edge_type, vertex, neg_num)
    return vertex, context, neg
```



Graph Storage (Memory Representation)

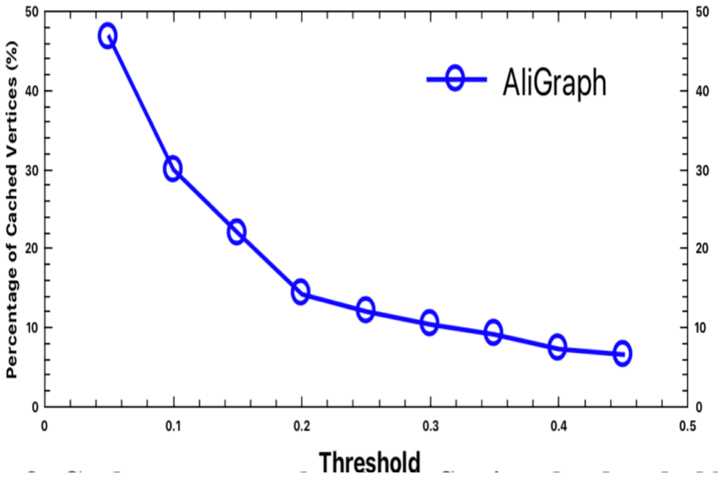
- Tens of billions of edge relations
- The adjacency relationship is the basis of the GNN algorithm, and locality needs to be fully utilized
- Tradeoff between data fragmentation and communication
  - Effective caching strategy: importance metric based on vertices
- Easy to traverse and sample by batch

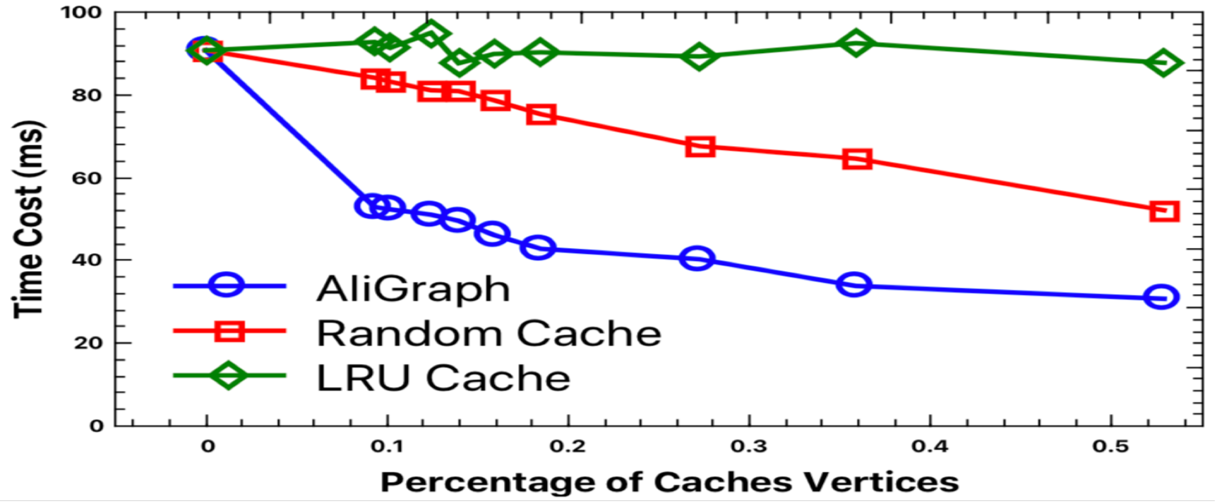
Cache/Memory Strategy



Input: graph $\mathcal{G}$, partition number p, cache depth h, threshold $\tau_1, \tau_2, \ldots, \tau_h$
Output: p subgraphs

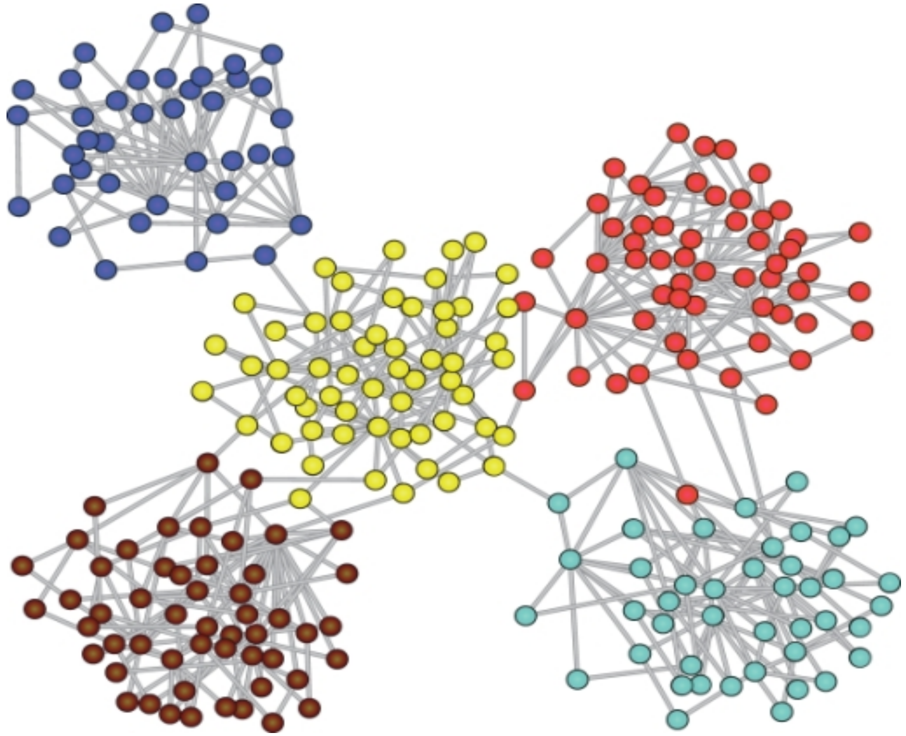
```
Initialize p graph servers
for each edge e = (u, v) ∈ ℰ do
    j = ASSIGN(u)
    Send edge e to the j-th partition
for each vertex v ∈ V do
    for k ← 1 to h do
        Compute D_i^(k)(v) and D_o^(k)(v)
        if D_i^(k)(v) / D_o^(k)(v) ≥ τ_k then
            Cache the 1 to k-hop out-neighbors of v on each partition where
              v exists
```

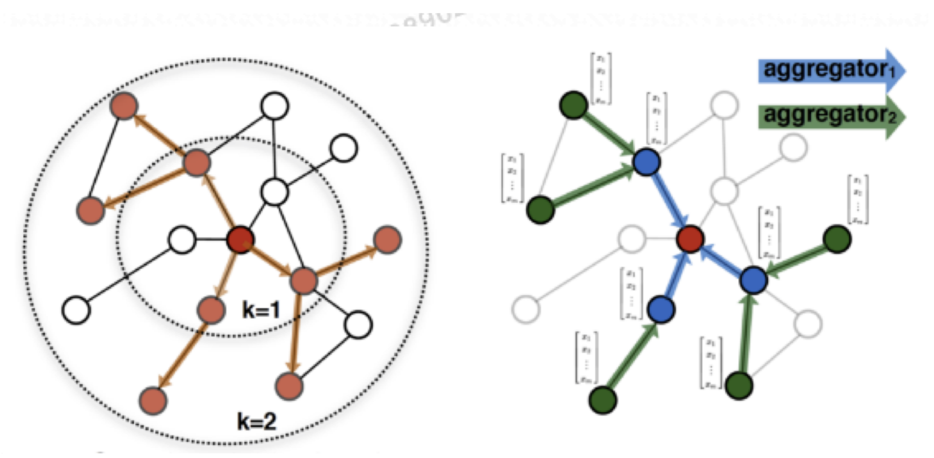




Cache acceleration: 50% faster than random method, 60% faster than LRU method

Operator Optimization





- Reduce double calculation by caching all levels of h
- Co-locate model parameters, all levels of h and the graph itself, greatly reduce network communication and delay
- Algorithm and System co-design: need to cooperate with training strategy and upper-level algorithm

AliGraph-Module Development Demo

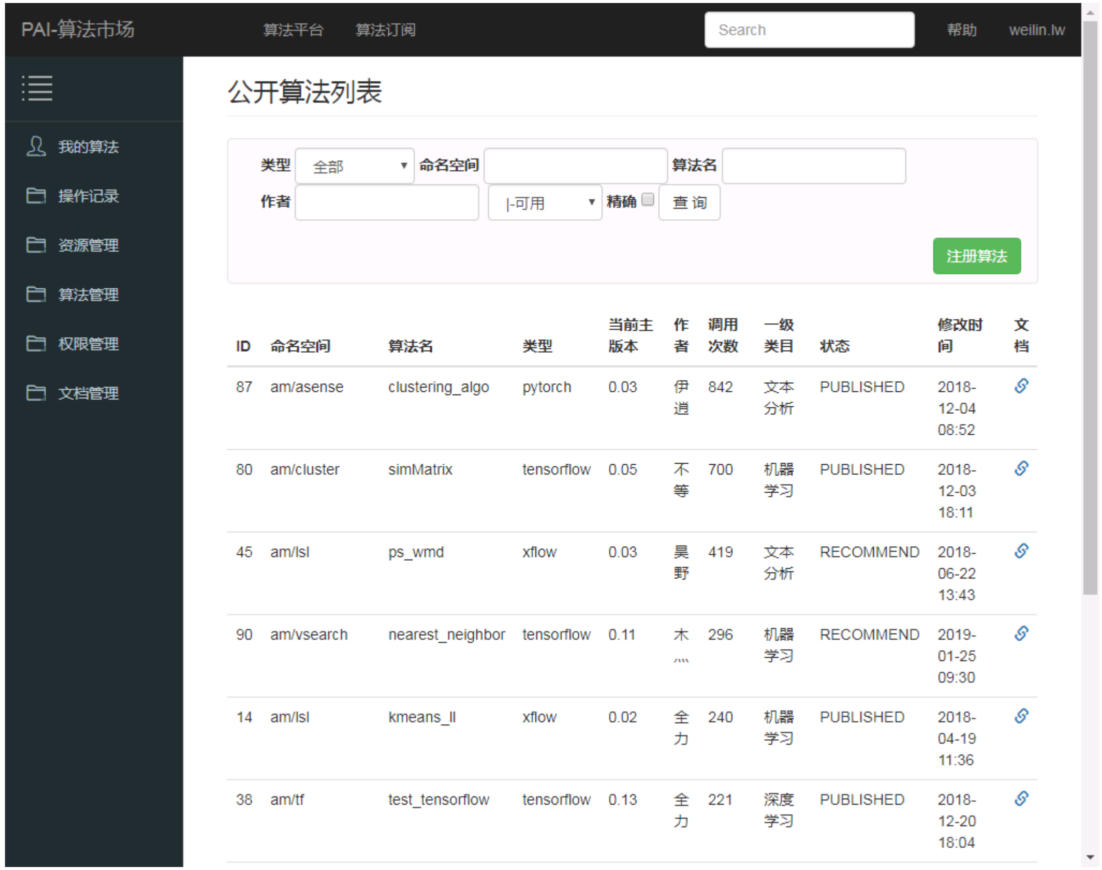
PrepareData
↓
BuildGraph
↓
GSL SampleStream
↓
FeatureVectorization
↓
GNN Module
↓
DefineLossFunction
↓
TrainingIteration
↓
ExportModule/Vector

example.py

```python
import graphlearn as gl
import graphlearn.python.nn.tfm as tfm
import tensorflow as tf

# Define graph object
g = gl.Graph()

# Add data source
g.node(i_path, 'i', decoder=gl.Decoder(attr_types=['float'] * 4, attr_dims=[10] * 4, labeled=True)) \
 .edge(i2i_path, ('i', 'i', 'i-i'), decoder=gl.Decoder()) \
 .init()

# Construct GSL Query
query = g.V('i').batch(10).alias('i') \
        .outV('i-i').sample(5).by('topk').alias('hop1') \
        .outV('i-i').sample(5).by('random').alias('hop2') \
        .values()
df = tfm.DataFlow(query)

# Construct Module
dims = np.array([4, 16, 8])
model = tfm.HomoEgoGraphSAGE(dims, bn_fn=None, active_fn=tf.nn.relu, droput=0.1)

# Module computing, Generate embedding
embeddings = model.forward(df.get_ego_graph('i'))

# Construct Node Classifer
nc = tfm.NodeClassifier(dims=[8, 4], class_num=2)
logits, loss = nc.forward(embeddings, eg.nodes.labels)

# Train
trainer = tfm.Trainer()
trainer.minimize(loss)

def trace(ret):
  print('loss = %f' % ret[1])

trainer.step_to_epochs(10, [logits, loss], trace)

g.close()
```

AliGraph-Algorithm Deployment and Share

- PAI provides algorithm warehouse for algorithm package, management and release



Part 4：Algorithm Warehouse

Sampling

Subgraph-augmented Path Embedding for Semantic User Search on Heterogeneous Social Network, WWW 2018.

- Representative and negative sampling
- Borrow idea from importance sampling
- Extend node-wise to batch-wise sampling
- Type-dependent & -fusion sampling with self-normalization
- Computational complexity drops from O(|E|+|V|) to O(|V|)

Algorithm 2 Training type-fusion strategy with self-normalization (one batch)

Input: same as Algorihtm 1;
Output: same as Algorithm 1;

```
compute p and the sampler q by Eq.(16);
for each t ∈ T do
    sample n_t neighborhoods with the sampler q_t;
end for
for each v_i do
    for each t ∈ T do
        normalize {π(v_j)|v_j ∈ Ng(v_i, t)};
        compute the estimator ĝ_{i,t} by Eq. (17);
    end for
    compute the reconstructed h̃_i with {ĝ_{i,t}|t ∈ T} by Eq. (6);
    compute L_{EPS,k}(v_i) based on h̃_i, h_i;
end for
minimize L_k(H, Θ^(k)) and perform gradient updates;
output the optimized Θ^(k+1) and H.
```

Algorithm 1 Training with type-dependent strategy (one batch)

Input: target nodes $\{v_i\}$; neighborhood $\{v_j\}$; sampling size of each type $\{n_t|t \in T\}$; embeddings H; parameters $\Theta^{(k)}$.
Output: the optimized embedding H and parameters $\Theta^{(k+1)}$.

```
for each t ∈ T do
    compute p and the sampler q_t by Eq. (14);
    sample n_t neighbors with the sampler q_t;
end for
for each v_i do
    for each t ∈ T do
        compute the estimator ĝ_{i,t} by Eq. (9);
    end for
    compute the reconstructed h̃_i with {ĝ_{i,t}|t ∈ T} by Eq. (6);
    compute L_{EPS,k}(v_i) based on h̃_i and h_i;
end for
minimize L_k(H, Θ^(k)) and perform gradient updates;
output the optimized Θ^(k+1) and H.
```

Table 4: Micro/Macro-F1 scores for multi-class classification on Aminer. Excluding HEP, the best method is bolded and the second best is underlined. Percentages in parenthesis indicate the performance level, treating HEP-nil as 0% and HEP as 100%.

	Micro-F1				Macro-F1			
Sampling size	512 ~ 3%	1024 ~ 6%	2048 ~ 12%	4096 ~ 24%	512 ~ 3%	1024 ~ 6%	2048 ~ 12%	4096 ~ 24%
HEP-Nil	0.2084 (intuitive lower bound)				0.2027 (intuitive lower bound)			
HEP	0.9566 (intuitive upper bound)				0.9551 (intuitive upper bound)			
AS-GCN	0.2866 (10%)	0.2901 (11%)	0.2951 (12%)	0.3050 (13%)	0.2721 (9%)	0.2770 (9%)	0.2788 (9%)	0.2818 (10%)
Unif-TD	0.3089 (13%)	0.3934 (25%)	0.5227 (42%)	0.6861 (64%)	0.2864 (10%)	0.3679 (21%)	0.4978 (39%)	0.6642 (61%)
Unif-TF	0.2877 (11%)	0.4010 (26%)	0.5574 (47%)	0.7329 (70%)	0.2897 (11%)	0.3941 (25%)	0.5371 (44%)	0.7162 (68%)
Unif-TD-SN	0.2283 (3%)	0.2955 (12%)	0.4816 (37%)	0.7260 (69%)	0.1768 (-4%)	0.2286 (3%)	0.4431 (31%)	0.7209 (68%)
Unif-TF-SN	0.1631 (-6%)	0.2951 (12%)	0.4489 (32%)	0.7101 (67%)	0.1315 (-10%)	0.2655 (8%)	0.4302 (30%)	0.6954 (65%)
VarR-TD	0.3423 (18%)	0.4911 (38%)	0.5734 (49%)	0.6408 (58%)	0.3258 (16%)	0.4663 (34%)	0.5516 (46%)	0.6185 (55%)
VarR-TF	**0.6200 (55%)**	0.7472 (72%)	0.7931 (78%)	0.7920 (78%)	**0.6087 (54%)**	0.7388 (71%)	0.7812 (77%)	0.7822 (77%)
VarR-TD-SN	0.2353 (4%)	0.3245 (16%)	0.5210 (42%)	0.7078 (67%)	0.1999 (-1%)	0.2853 (10%)	0.5008 (39%)	0.6921 (65%)
VarR-TF-SN	0.5960 (52%)	**0.8124 (81%)**	**0.9008 (93%)**	**0.9311 (97%)**	0.5843 (50%)	**0.8170 (81%)**	**0.9047 (93%)**	**0.9327 (97%)**

Table 5: F1 and AUC scores for binary purchase prediction on Alibaba. Excluding HEP, the best method is bolded and the second best is underlined. Percentages in parenthesis indicate the performance level, treating HEP-nil as 0% and HEP as 100%

	F1-score				AUC			
Sampling size	512 ~ 2.5%	1024 ~ 5%	2048 ~ 10%	4096 ~ 20%	512 ~ 2.5%	1024 ~ 5%	2048 ~ 10%	4096 ~ 20%
HEP-Nil	0.3994 (intuitive lower bound)				0.5134 (intuitive lower bound)			
HEP	0.5793 (intuitive upper bound)				0.7777 (intuitive upper bound)			
AS-GCN	(unable to complete due to memory constraint)							
Unif-TD	0.3883 (-6%)	0.4353 (20%)	0.4692 (39%)	0.4891 (50%)	0.5999 (33%)	0.6616 (56%)	0.7158 (77%)	0.7463 (88%)
Unif-TF	0.4051 (3%)	0.4281 (16%)	0.4558 (31%)	0.4858 (48%)	0.6207 (41%)	0.6584 (55%)	0.6992 (70%)	0.7413 (86%)
Unif-TD-SN	0.3884 (-6%)	0.4322 (18%)	0.4470 (26%)	0.4920 (51%)	0.5982 (32%)	0.6635 (57%)	0.6873 (66%)	0.7500 (90%)
Unif-TF-SN	0.3928 (-4%)	0.4219 (13%)	0.4435 (25%)	0.4917 (51%)	0.6028 (34%)	0.6496 (52%)	0.6836 (64%)	0.7489 (89%)
VarR-TD	0.4476 (27%)	0.4518 (29%)	0.4852 (48%)	0.4743 (42%)	0.6829 (64%)	0.6969 (69%)	0.7314 (82%)	0.7317 (83%)
VarR-TF	**0.4774 (43%)**	0.4863 (48%)	**0.5019 (57%)**	0.5090 (61%)	**0.7374 (85%)**	**0.7466 (88%)**	**0.7513 (90%)**	0.7548 (91%)
VarR-TD-SN	0.4293 (17%)	0.4497 (28%)	0.4720 (40%)	0.5038 (58%)	0.6645 (57%)	0.6941 (68%)	0.7293 (82%)	0.7501 (90%)
VarR-TF-SN	0.4766 (43%)	**0.4896 (50%)**	0.4970 (54%)	**0.5097 (61%)**	0.7347 (84%)	0.7453 (88%)	0.7496 (89%)	**0.7551 (91%)**

Multiplex ANRL: Attributed Network Representation Learning via Deep Neural Networks, IJCAI 2018

Representation Learning for Attributed Multiplex Heterogeneous Network, KDD 2018

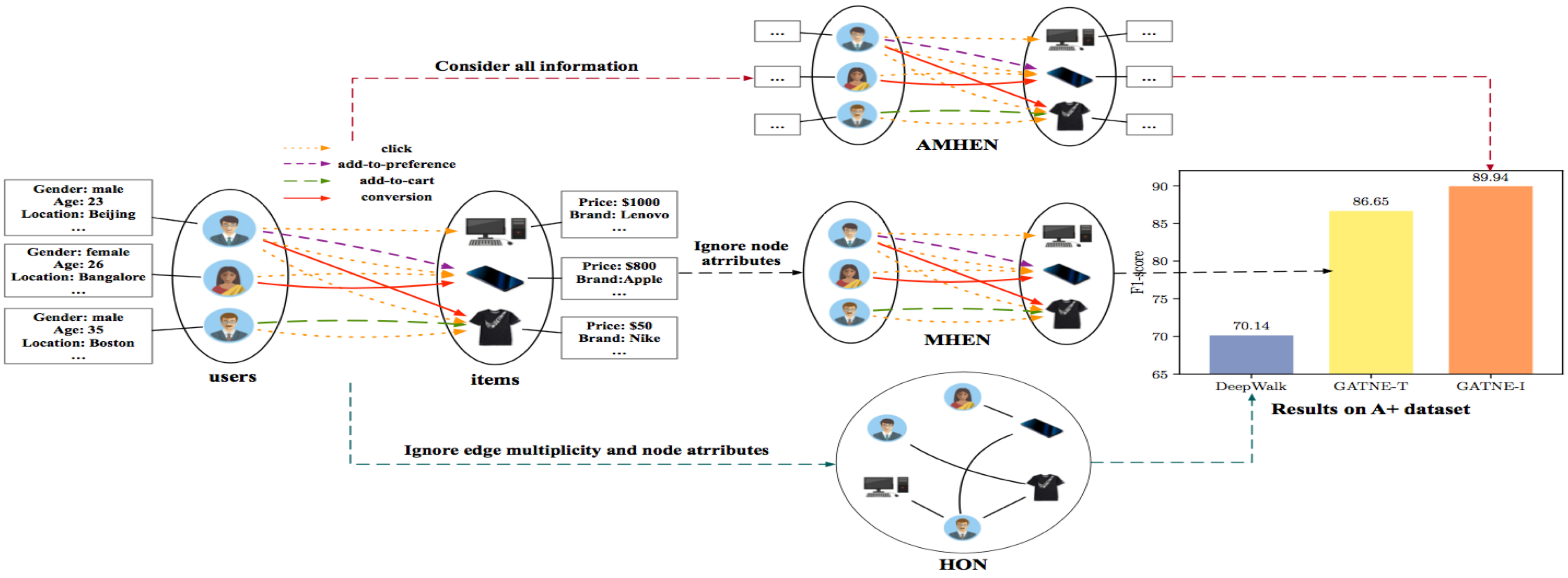


Figure 1: The left illustrates an example of an attributed multiplex heterogeneous network. Users in the left part of the figure are associated with attributes including gender, age, and location. Similarly, items in the left part of the figure include attributes such as price and brand. The edge types between users and items are from four interactions, including *click*, *add-to-preference*, *add-to-cart* and *conversion*. The three subfigures in the middle represent different ways of setting up the graphs, including HON, MHEN, and AMHEN from the bottom to the top. The right part shows the performance improvement of the proposed models over DeepWalk on the A+ dataset. As can be seen, *GATNE*-I achieves a +28.23% performance lift compared to DeepWalk.

- Transudative Model:

For *GATNE*-T, the overall embedding for node v_i on edge type r is:

$$\mathbf{v}_{i,r} = \mathbf{b}_i + \alpha_r \mathbf{M}_r^T \mathbf{U}_i \mathbf{a}_{i,r} = \mathbf{b}_i + \alpha_r \mathbf{M}_r^T \sum_{p=1}^{m} \lambda_p \mathbf{u}_{i,p},$$

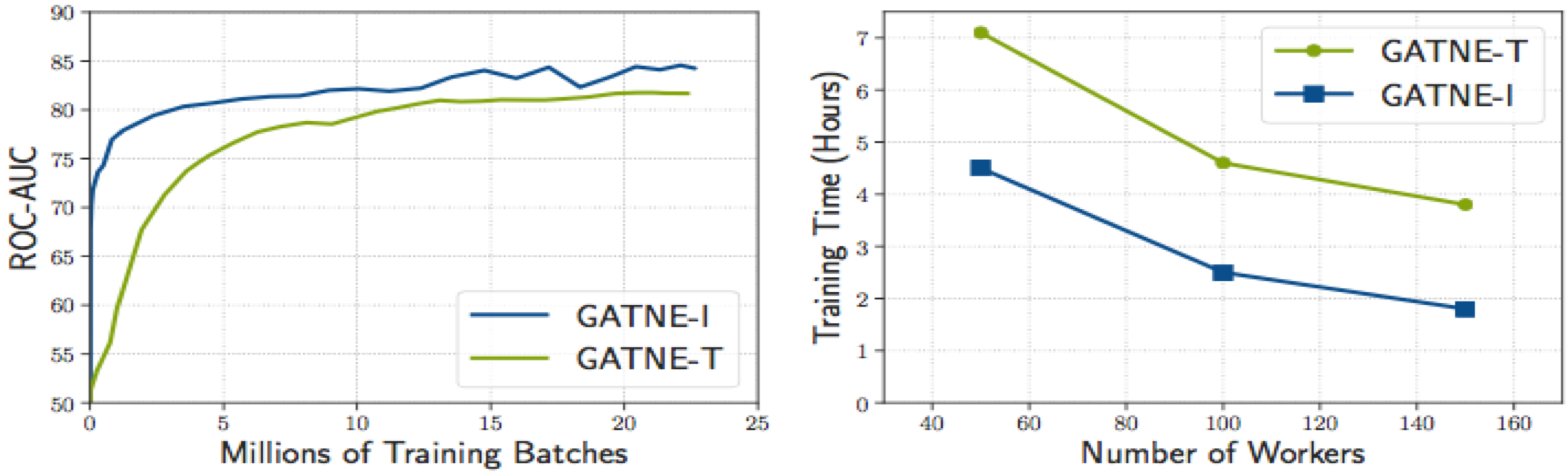
where λ_p denotes the p-th element of $\mathbf{a}_{i,r}$ and is computed as:

$$\lambda_p = \frac{\exp(\mathbf{w}_r^T \tanh(\mathbf{W}_r \mathbf{u}_{i,p}))}{\sum_t \exp(\mathbf{w}_r^T \tanh(\mathbf{W}_r \mathbf{u}_{i,t}))}.$$

- Inductive Model:

$$\mathbf{v}_{i,r} = \mathbf{h}_z(\mathbf{x}_i) + \alpha_r \mathbf{M}_r^T \mathbf{U}_i \mathbf{a}_{i,r} + \beta_r \mathbf{D}_z^T \mathbf{x}_i,$$

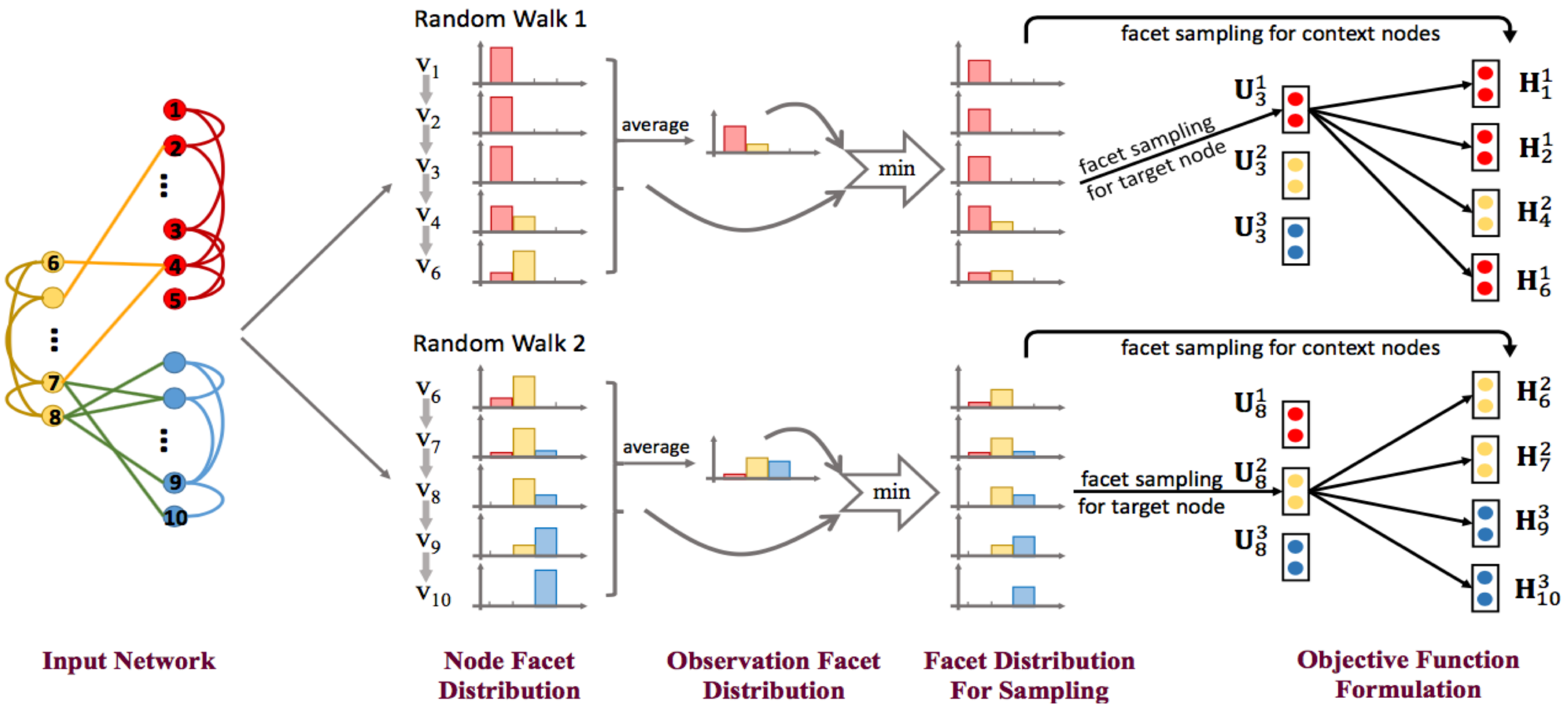
	Amazon			YouTube			Twitter			A+ dataset (small)		
	ROC-AUC	PR-AUC	F1	ROC-AUC	PR-AUC	F1	ROC-AUC	PR-AUC	F1	ROC-AUC	PR-AUC	F1
DeepWalk	94.20	94.03	87.38	71.11	70.04	65.52	69.42	72.58	62.68	59.39	60.62	56.10
node2vec	94.47	94.30	87.88	71.21	70.32	65.36	69.90	73.04	63.12	62.26	63.40	58.49
LINE	81.45	74.97	76.35	64.24	63.25	62.35	62.29	60.88	58.18	53.97	54.65	52.85
metapath2vec	94.15	94.01	87.48	70.98	70.02	65.34	69.35	72.61	62.70	60.94	61.40	58.25
ANRL	95.41	94.19	89.60	75.93	73.21	70.65	70.04	67.16	64.69	64.76	61.67	61.37
PMNE(n)	95.59	95.48	89.37	65.06	63.59	60.85	69.48	72.66	62.88	62.23	63.35	58.74
PMNE(r)	88.38	88.56	79.67	70.61	69.82	65.39	62.91	67.85	56.13	55.29	57.49	53.65
PMNE(c)	93.55	93.46	86.42	68.63	68.22	63.54	67.04	70.23	60.84	51.57	51.78	51.44
MVE	92.98	93.05	87.80	70.39	70.10	65.10	72.62	73.47	67.04	60.24	60.51	57.08
MNE	90.28	91.74	83.25	82.30	82.18	75.03	91.37	91.65	84.32	62.79	63.82	58.74
GATNE-T	**97.44**	**97.05**	**92.87**	**84.61**	81.93	**76.83**	**92.30**	91.77	**84.96**	66.71	67.55	62.48
GATNE-I	96.25	94.77	91.36	84.47	**82.32**	**76.83**	92.04	**91.95**	84.38	**70.87**	**71.65**	**65.54**



(a) Convergence (b) Scalability

Figure 3: (a) The convergence curve for *GATNE*-T and *GATNE*-I models on A+ dataset. The inductive model converges faster and achieves better performance than the transductive model. (b) The training time decreases as the number of workers increases. *GATNE*-I takes less training time to converge compared with *GATNE*-T.

Mixture GNN Is a Single Vector Enough? Exploring Node Polysemy for Network Embedding, KDD 2019



$$p(o|s(o), \mathcal{P}, \theta) = \prod_{v_j \in \mathcal{N}(v_i)} p(v_j|v_i, s(o)),$$

and each product factor is calculated as

$$p(v_j|v_i, s(o)) = \frac{\exp(<\mathbf{H}_j^{k_j}, \mathbf{U}_i^{k_i}>)}{\sum_{v,k} \exp(<\mathbf{H}_v^{k}, \mathbf{U}_i^{k_i}>)},$$

Hierarchical GNN

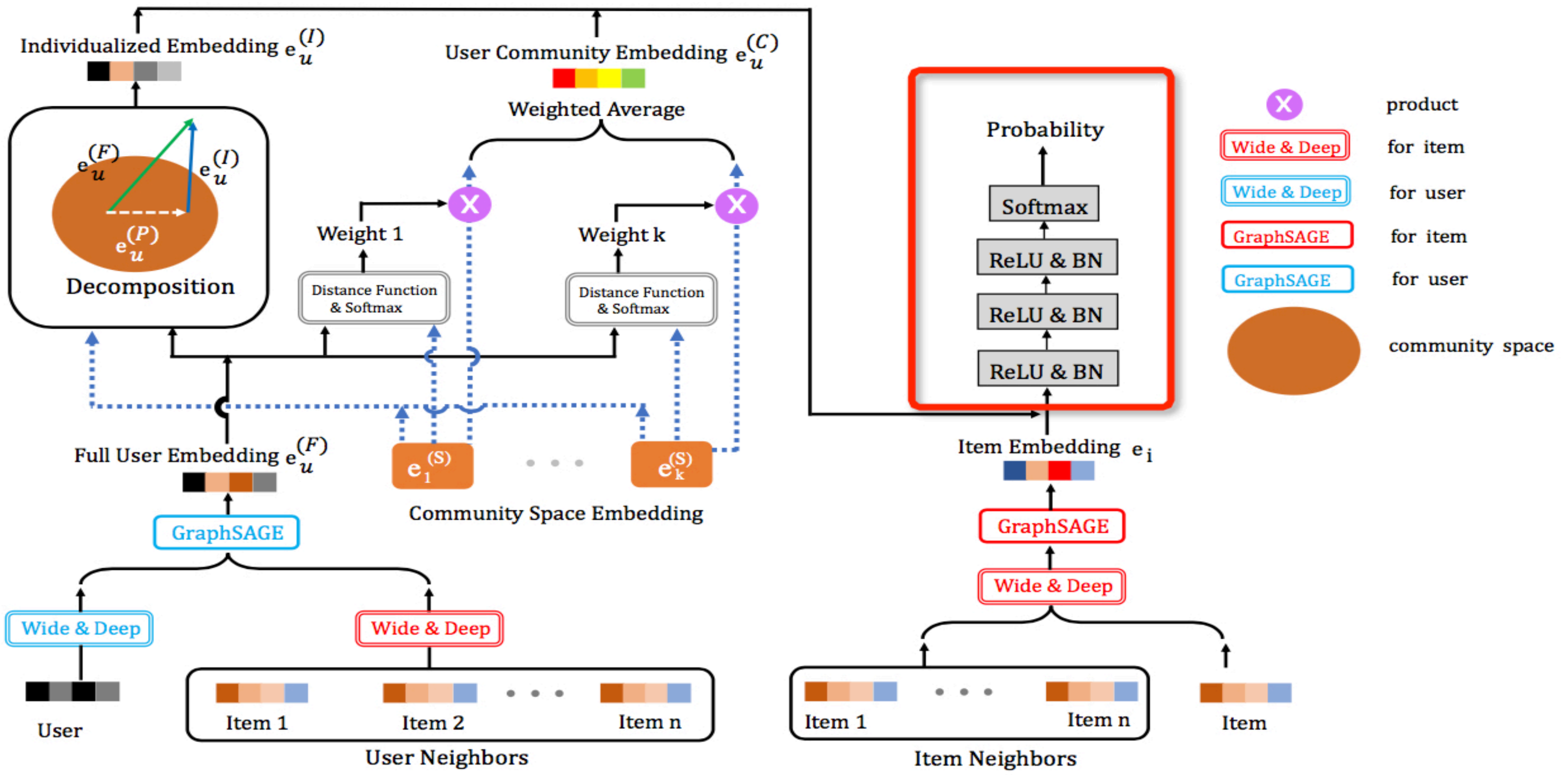


Figure 2: Framework of *Bi-HGNN*. Left side outputs *user-community* and *individualized* embeddings. Since user and item raw features are generally sparse, wide and deep is used here for feature encoding. Based on the user's historical behaviors, n items are selected as his/her neighbors to generate *full* user embedding with GraphSAGE. *User-community* embedding is derived through weighted average over users that belong to the specific community, where weight is determined by the distance between the user and *base community* embedding. *Full* embedding is decomposed into *user-community* and *individualized* embeddings, which is illustrated in the upper left of the figure within the dashed square. Right part represents the item embedding generation. With both the user and item embeddings, the classifier undergoes several fully connected neural networks and outputs a probability to determine whether to recommend the item to the user.

Evolving GNN Large Scale Evolving Graphs with Burst Detection, IJCAI 2019

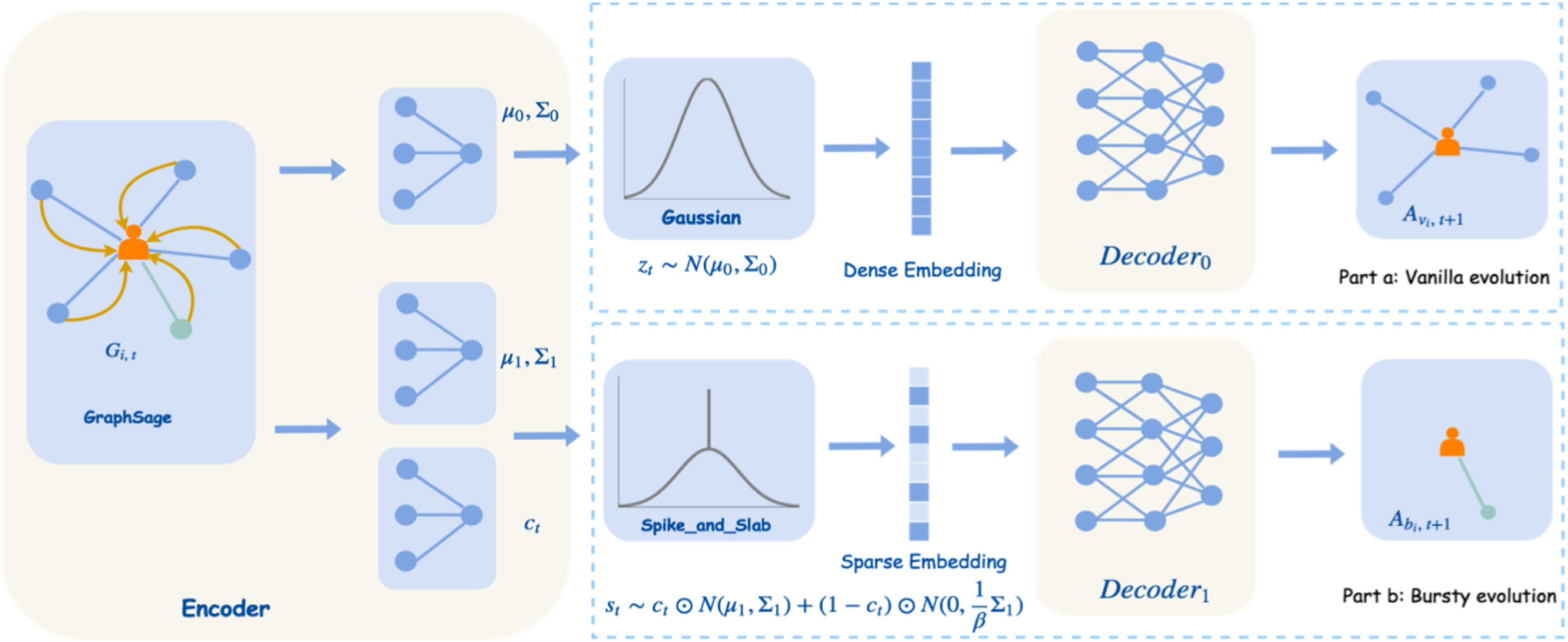


Figure 2: An illustration of the proposed framework *BurstGraph*. At time step t, the framework generates vanilla evolution and bursty evolution based on network structure G_t. Part a is an original VAE for vanilla evolution, where random variable $\boldsymbol{z}_t$ follows a Gaussian distribution. Part b is an extended VAE for bursty evolution, where random variable $\boldsymbol{s}_t$ follows a spike-and-slab distribution because of the sparsity of bursty links. The encoder for these two random variables $\boldsymbol{z}_t$ and $\boldsymbol{s}_t$ shares the same GraphSAGE to utilize the information from vertices and their neighbors.

Bayesian GNN

- BEM is proposed to bridge KG and BG seamlessly, with the consideration of the behavior-specific bias. This framework provides a new perspective of making a reasoning mechanism (cognitive graph).
- As a method, BEM is generic and flexible in that it can use any KG embeddings to correct any BG embeddings. On the contrary, it is potentially able to help KG embeddings acquire novel knowledge from the BG embeddings that does not exist in the knowledge graph.

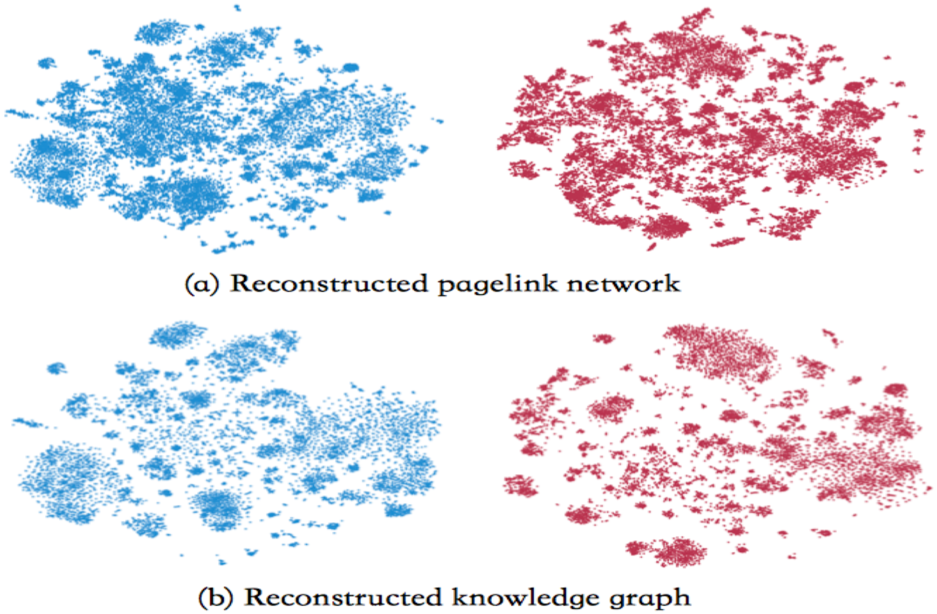


Figure 4: Reconstructed vertex graphs. The blue ones are original graphs and red ones are corrected graphs.

Supported Algorithms

Algorithms Proposed

Algorithm	Heterogeneous		Attributed	Dynamic	Large-Scale
	Node	Edge			
HEP	✓	✓	✓	✗	✗
AHEP	✓	✓	✓	✗	✓
GATNE	✓	✓	✓	✗	✓
Mixture GNN	✓	✓	✓	✗	✗
Hierarchical GNN	✓	✓	✓	✗	✗
Bayesian GNN	✗	✓	✓	✗	✗
Evolving GNN	✗	✓	✓	✓	✗

Part 5：Current Focus in Practice

Search-Recommendation Platform

Business Solution	Search/Recommendation/ Ads	Business Intelligent	Fraud Detection/ Risk Control	ID Mapping and Profile	Crowd Marketing
Service	Index/Rank	Training/Inference	Evolutional/Online updating	AutoML	Transfer Learning/ Reinforcement Learning
AI Algorithm	Natural Language Understanding	Image Processing	Video Processing	Machine Learning/ Deep Learning	Cognitive Graph
Data	Global User Profile	Global Seller Profile	Online Offline Info Integration	Ecommerce Cognitive Graph	Business Operation Knowledge
System	Distributed Computing	Stream Computing	Graph Computing	Online Learning and Inference	Heterogeneous Computing

Cloud Theme

from Single Product Recommendation to Group Meet Place、cultivate metal upgrade

Meeting Place for Personalized Groups

SPR

Reasoning

Tips

Knowledge Card

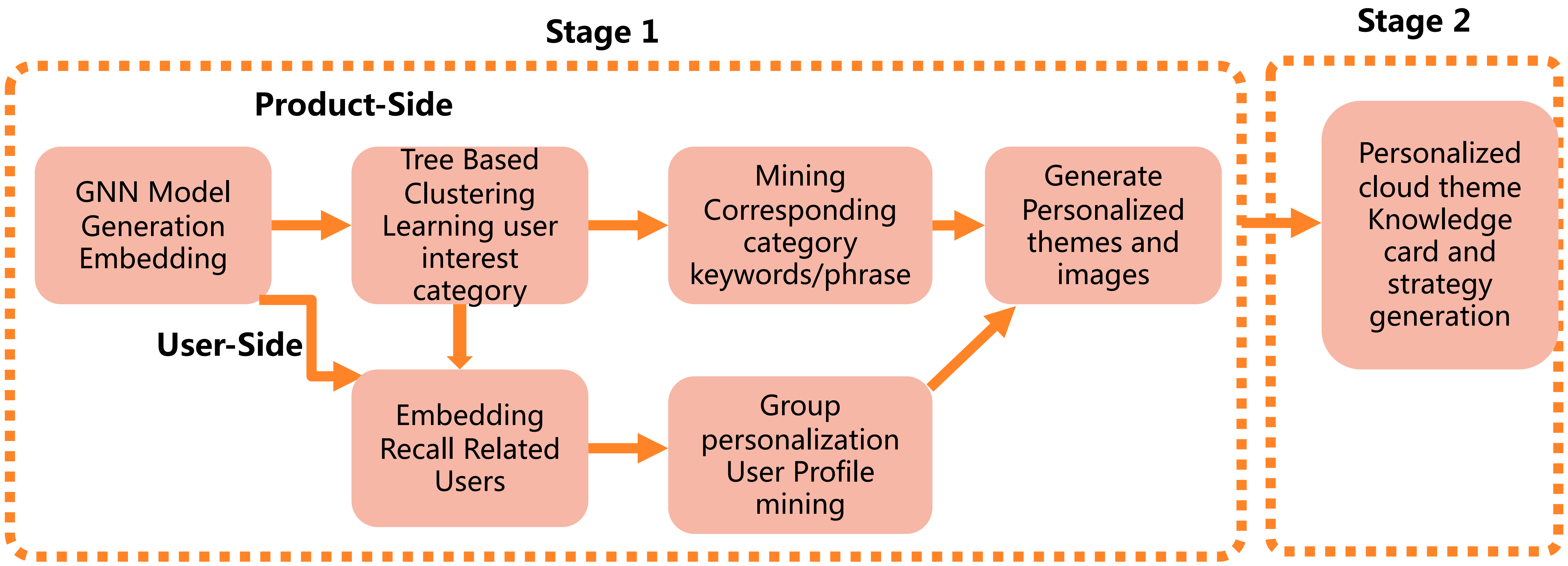
Ad Embedding On Scene

Video recommendation

Algorithm

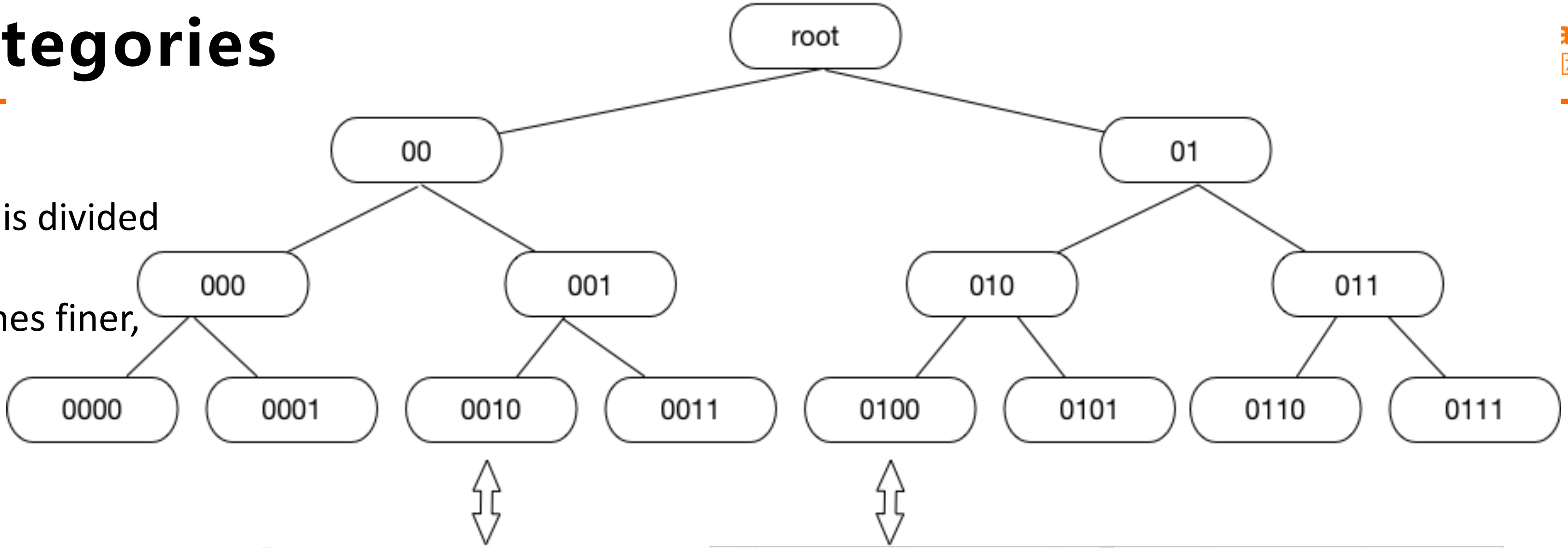
- Natural Language Understanding
- Image Processing
- Video Processing
- Machine Learning/ Deep Learning
- Graph Embedding

Cloud Theme The Flow



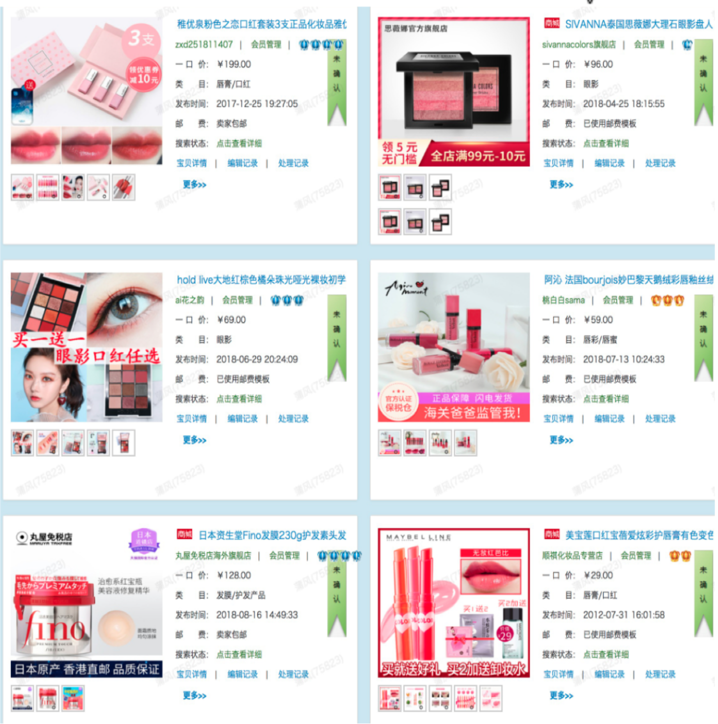
User Interest Categories

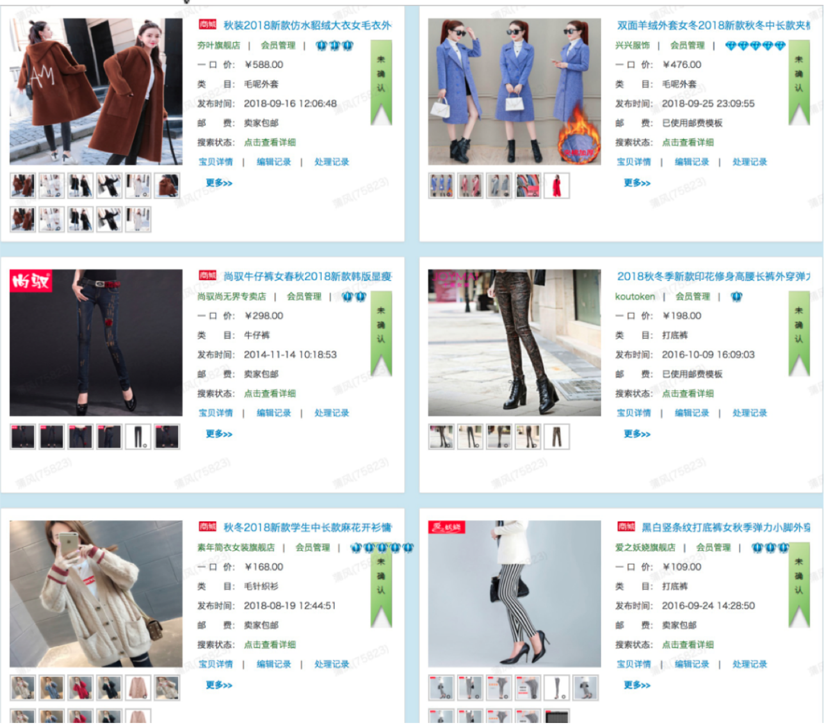
1. GCN Model generate embedding
2. Hierarchical clustering, each layer is divided
3. As the number of layers deepens, the granularity of the division becomes finer, and the semantics gradually emerge



Commodity Aggregation

Lipstick
Eye shadow
Lip gloss
Hair mask
......





Woolen coat
jeans
Leggings Wool
sweater
......

Cloud Theme Title Auto-Generation

Now:

1. thousands of online cloud themes, mostly manual verification, many unmaintained
2. The main and subtitles of the cloud theme are too plain to attract users to click

Future:

1. Automatically generate cloud topics through algorithms, produce them in batches by algorithms, and automatically iteratively optimize.
2. Learn to produce scene-oriented, attractive main and subtitles.

   ex:

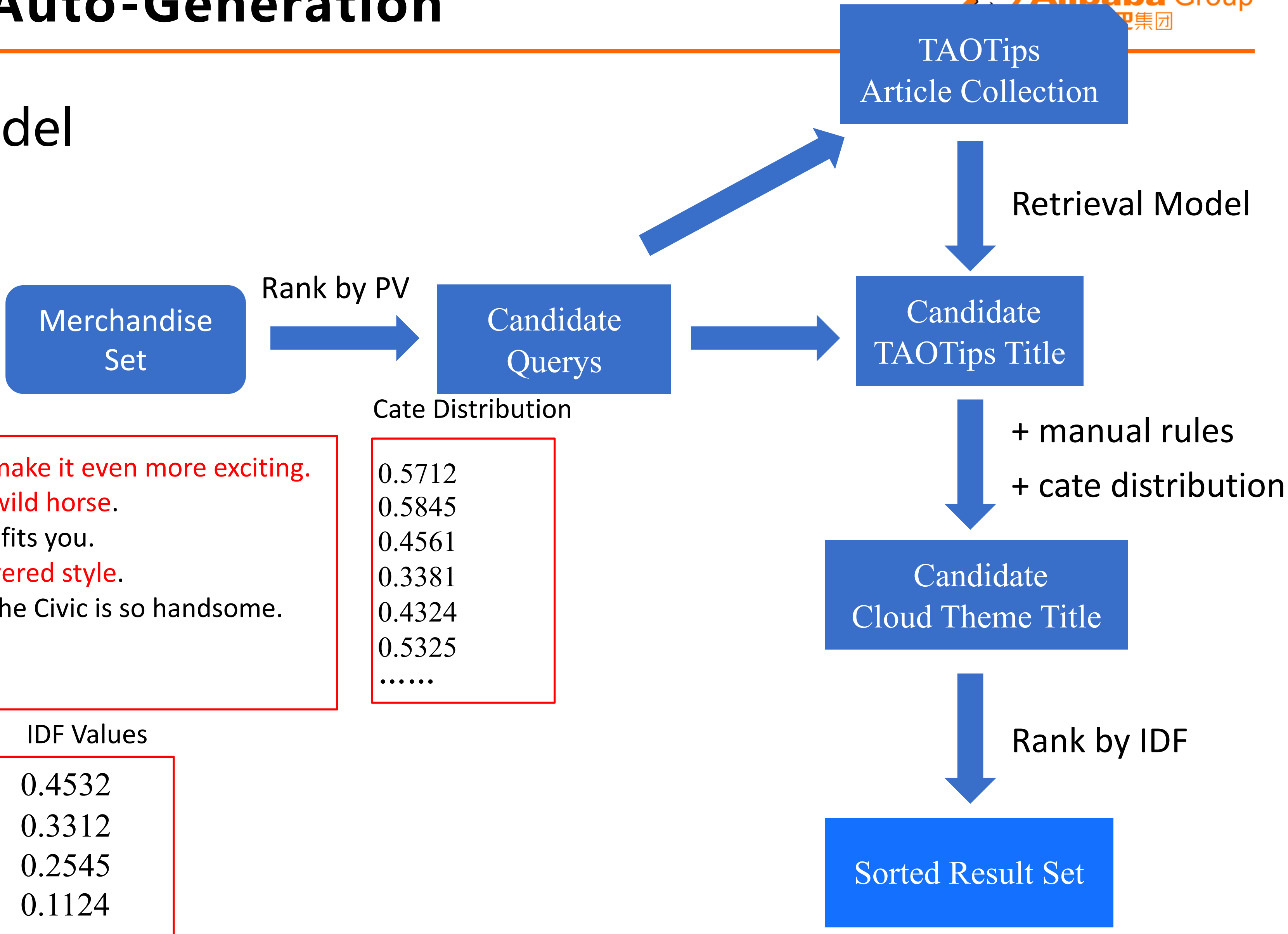
   1st cloud theme
   Main Title: The coolest "digital" baby
   Subtitle: Immersive game experience

   2nd cloud theme
   Main Title: Baby winter Heater
   Subtitle: Provide warm sleep for your baby

Cloud Theme Title Auto-Generation

Method1: retrieval model

Civic 10th gen modification
Civic Modification
Angkerra Modification
Fit modification
……



Appearance of 10th-gen Civic is modified to make it even more exciting.
The new Civic changes, turn your car into a wild horse.
Nothing is born perfect, a Civic modified car fits you.
The new Civic is modified to play a high-powered style.
The 205-horsepower grocery shopping car, the Civic is so handsome.
Upgrade the Civic to create a savage calf.
……

Cate Distribution

0.5712
0.5845
0.4561
0.3381
0.4324
0.5325
……

Turn your car into a wild horse
Make the car more exciting
Nothing is born perfect
create a savage calf
……

IDF Values

0.4532
0.3312
0.2545
0.1124
……

Cloud Theme Title Auto-Generation

Alibaba Group 阿里巴巴集团

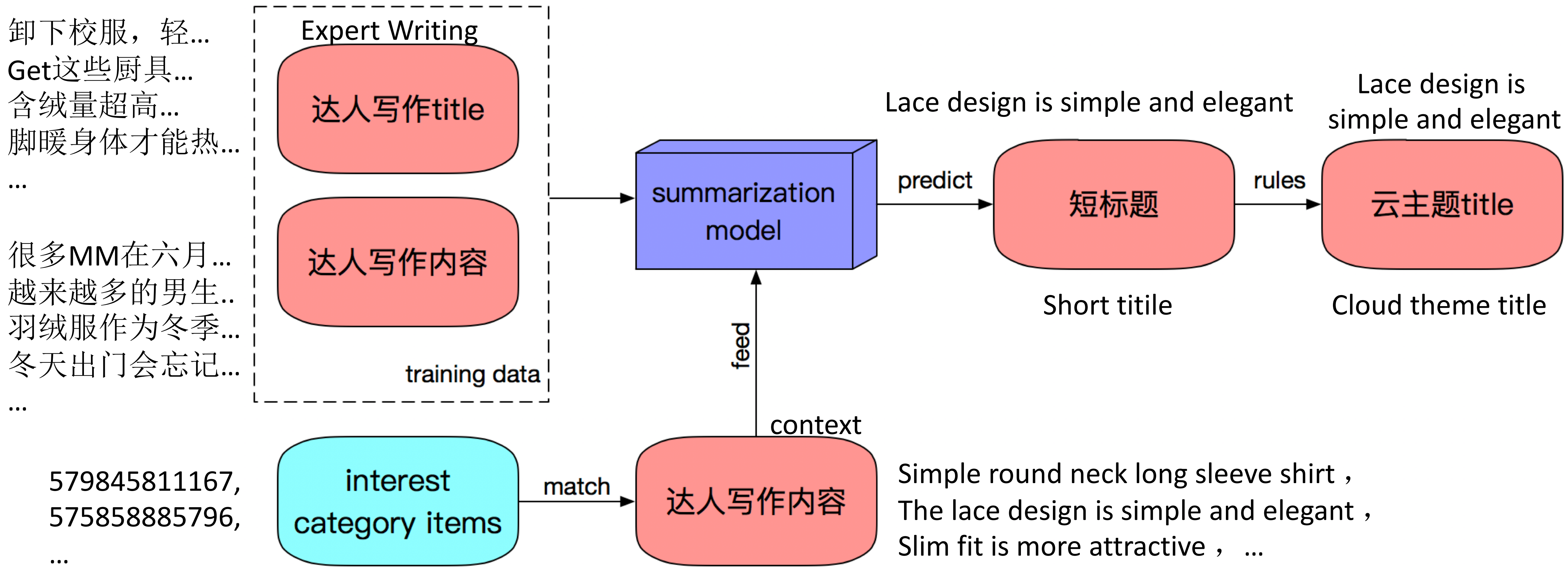
		Main Title	SubTitle	Category
1	序列	主标题	副标题	类目
1019	101010011	你的装嫩神器	完美发挥显示器的性能	显示器 键盘 散热器/风扇 Hifi音箱 机箱 DIY兼容机 鼠标垫/贴/腕垫 功放 显卡 游戏实物周边
1020	101010011	让你一步到位	笔记本玩游戏不方便	显示器 键盘 散热器/风扇 Hifi音箱 机箱 DIY兼容机 鼠标垫/贴/腕垫 功放 显卡 游戏实物周边
1021	101010011	真正的桌面PC	身临其境的游戏体验	显示器 键盘 散热器/风扇 Hifi音箱 机箱 DIY兼容机 鼠标垫/贴/腕垫 功放 显卡 游戏实物周边
146	110000	让宝宝有趣味的玩耍	夏天也可以用的尿不湿	连身衣/爬服/哈衣 内衣套装 棉袄/棉服 睡袋/防踢被 纸尿裤/拉拉裤/纸尿片 学步鞋 三轮车 套装 餐椅 唐装
147	110000	给宝宝更舒适的体验	婴儿的天然健康保护伞	连身衣/爬服/哈衣 内衣套装 棉袄/棉服 睡袋/防踢被 纸尿裤/拉拉裤/纸尿片 学步鞋 三轮车 套装 餐椅 唐装
148	110000	宝宝冬季的小暖炉	为宝宝提供温暖的睡眠	连身衣/爬服/哈衣 内衣套装 棉袄/棉服 睡袋/防踢被 纸尿裤/拉拉裤/纸尿片 学步鞋 三轮车 套装 餐椅 唐装

Baby’s small heater in winter

Provide warm sleep for your baby

Cloud Theme Title Auto-Generation

Method2: generative model



Cloud Theme Title Auto-Generation

	description	生成title
2	简约圆领长袖衬衫，蕾丝设计简约且优雅，修身的版型更为动人，3D印花更为休闲，很有文艺感。	蕾丝设计简约且优雅
3	眼下，外观小巧精美，看起来“萌萌哒”的多肉植物越来越受到养花爱好者喜爱，春天正是捣鼓这些可爱小植物的最佳时节。	多肉植物，给你最好的爱
4	袜子对于注重细节的女生来说很重要，不仅能让双脚得到细腻的呵护，还能通过个性搭配，穿出不一样的时尚新潮，助你秒杀路人甲乙丙丁，所以在众多的时尚单品中，袜子是你绝对不容忽视的女神标配哦~	袜子也要潮，穿出时尚新高度
5	白色是夏日里最常见的颜色，简约大气的白色非常百搭，让人看起来十分清爽，全白搭配亮眼又够chic，搭配亮色服饰穿，非常抢眼，助你轻松成为街头焦点！	简约白色，助你轻松成为街头焦点
6	不管是女生男生，每天都要摄取一定量的水是对身体有好处的，有一款心仪的杯子，连喝水都变成了一件开心的事情了，形形色色的杯子是不是都让你挑花了眼，小编给大家挑选了一些好看好用的杯子咯~	高颜值水杯，喝水也要萌萌哒
7	如果想穿得时尚又保暖，而且还很有型毛呢大衣当属明智之选，最容易穿出优雅气质来。穿上它，平淡中体现了一丝温馨和华贵，只要一件就能轻松抢尽眼球，让你成为寒风中暖融融的新淑女吧！	穿上毛呢大衣，暖融融的新淑女
8	很多男生可能觉得像剃须这种司空见惯每天都要做的小事，没必要浪费时间。但我会建议大家找一个阳光明媚的周末，尝试一次从洁脸、护理、上油、剃须、保养一条龙的剃须体验，我想你会因此爱上高大上的自己	让你高大上的剃须体验
9	炎炎夏日，小清新们的福利又来了，这么美的你怎么少得了仙气爆棚的连衣裙？无论是文艺复古系长裙还是元气满满的短裙，搭配着简单的碎花或者结合民族风，都有着独特的情调，所以说小清新连衣裙是永不过时的美衣，选对连衣裙，怎么搭配都美。	小清新连衣裙，怎么搭配都美
10	奥黛丽赫本曾经说过：“不涂口红的女人没有未来”。由此可见，口红对于女人来说是至关重要的。而口红囤积症是许多女孩的共有病症，大牌买多了钱包又受不了，其实很多口红平价又好用，不输大牌噢。	平价又好用，不输大牌的平价好物
11	非常经典的一款针织衫，船型领口的设计，显露出锁骨与脖颈曲线，修饰肩膀比例，凸显气质。流行的落肩，蝙蝠袖的宽松，任何身形都可以轻松驾驭。拼接不规则的多边形图案，特殊的面料更显个性。羊毛混纺针织面料，不易变形。	时尚百搭的针织毛衣
12	逢年过节总想给父母送点什么来聊表孝心，来补偿自己经常在外工作不能陪伴父母，但是很多时候送的礼物却有点鸡肋了。想不出送什么可以试试送一些比较实用的家电来分担家中家务的压力，让父母生活更舒心。	实用家电，送爸妈更省心

Cloud Theme Main Image Auto-Generation

Cloud theme cover image

↑

Template Puzzle

↑

TOP5/TOP7 Popularity selection from the cutout

↑

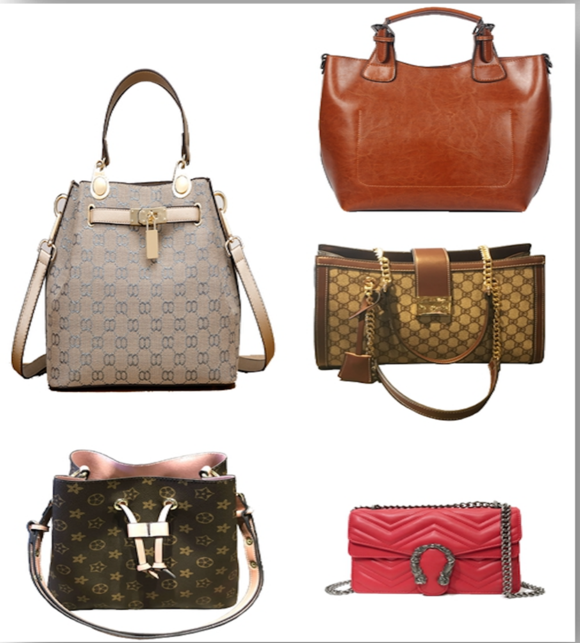
Commodity Cutout

↑

Detect and filter non-white background product images

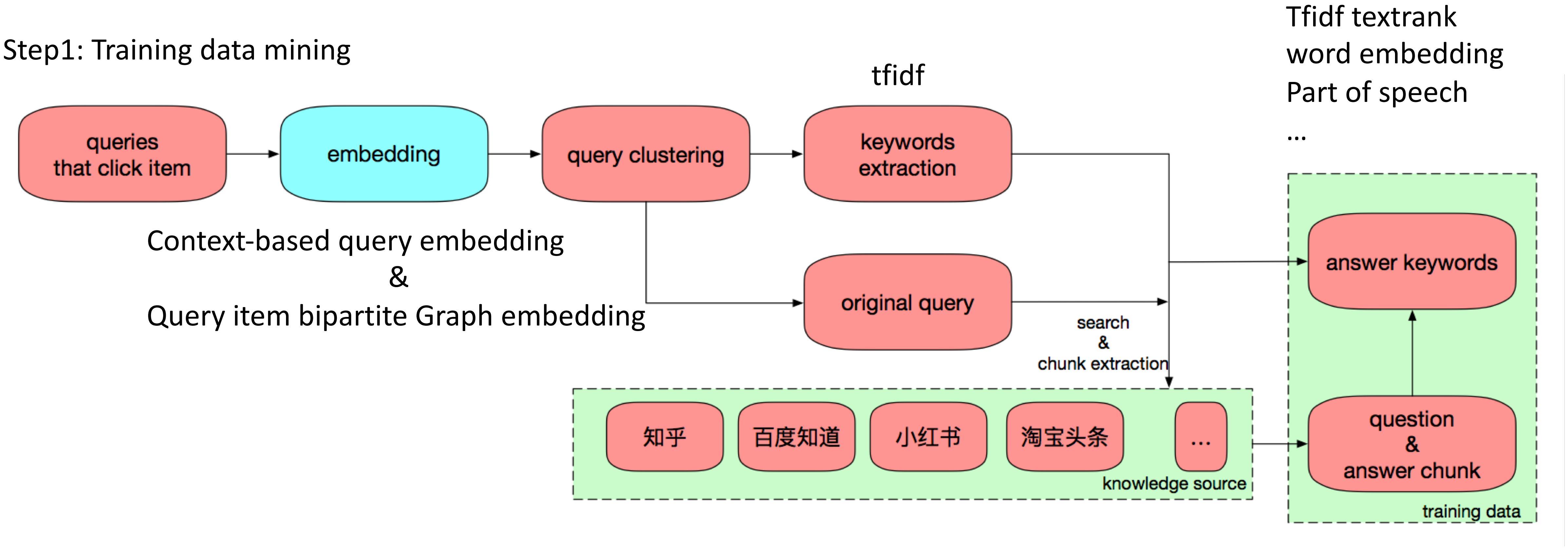
↑

Extract the TopN product map of each cloud-themed product



Cloud Theme Knowledge Card/Guide Generation

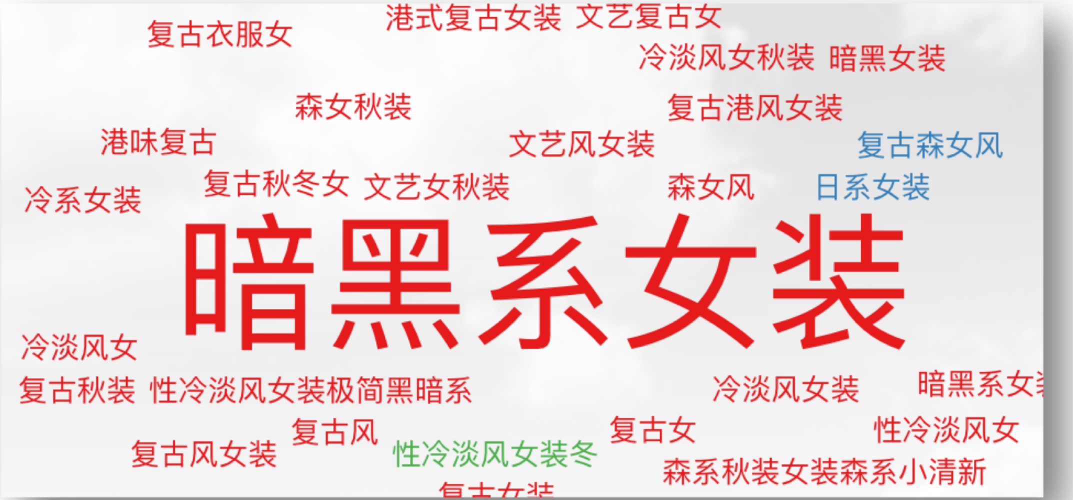
Step1: Training data mining

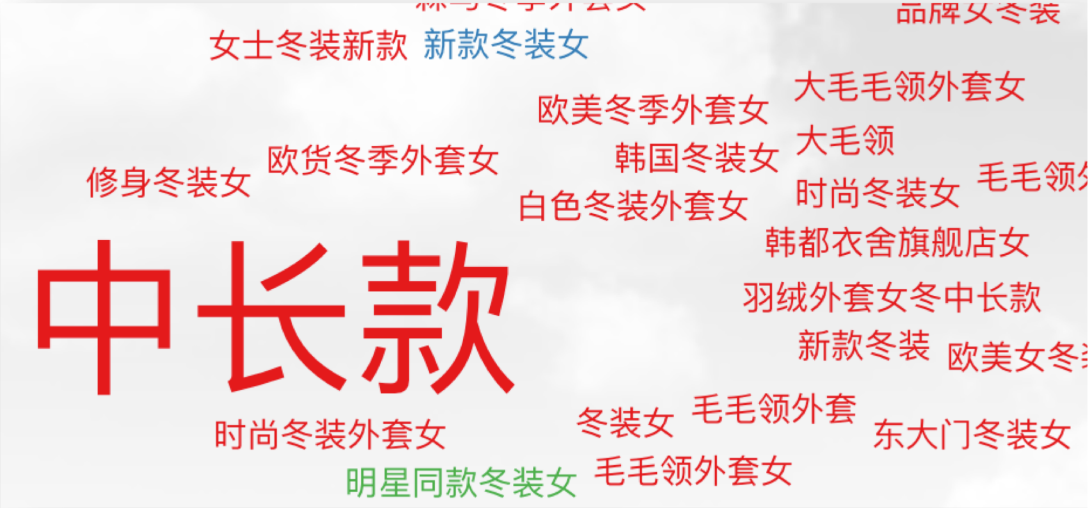


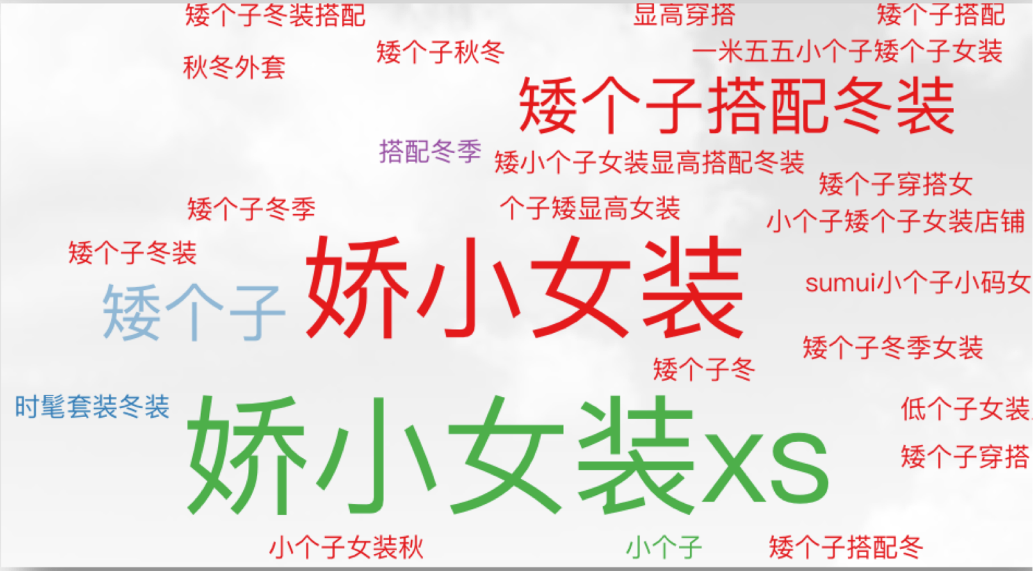
Cloud Theme Knowledge Card/Guide Generation

Alibaba Group
阿里巴巴集团

Knitwear category: Query clustering result display



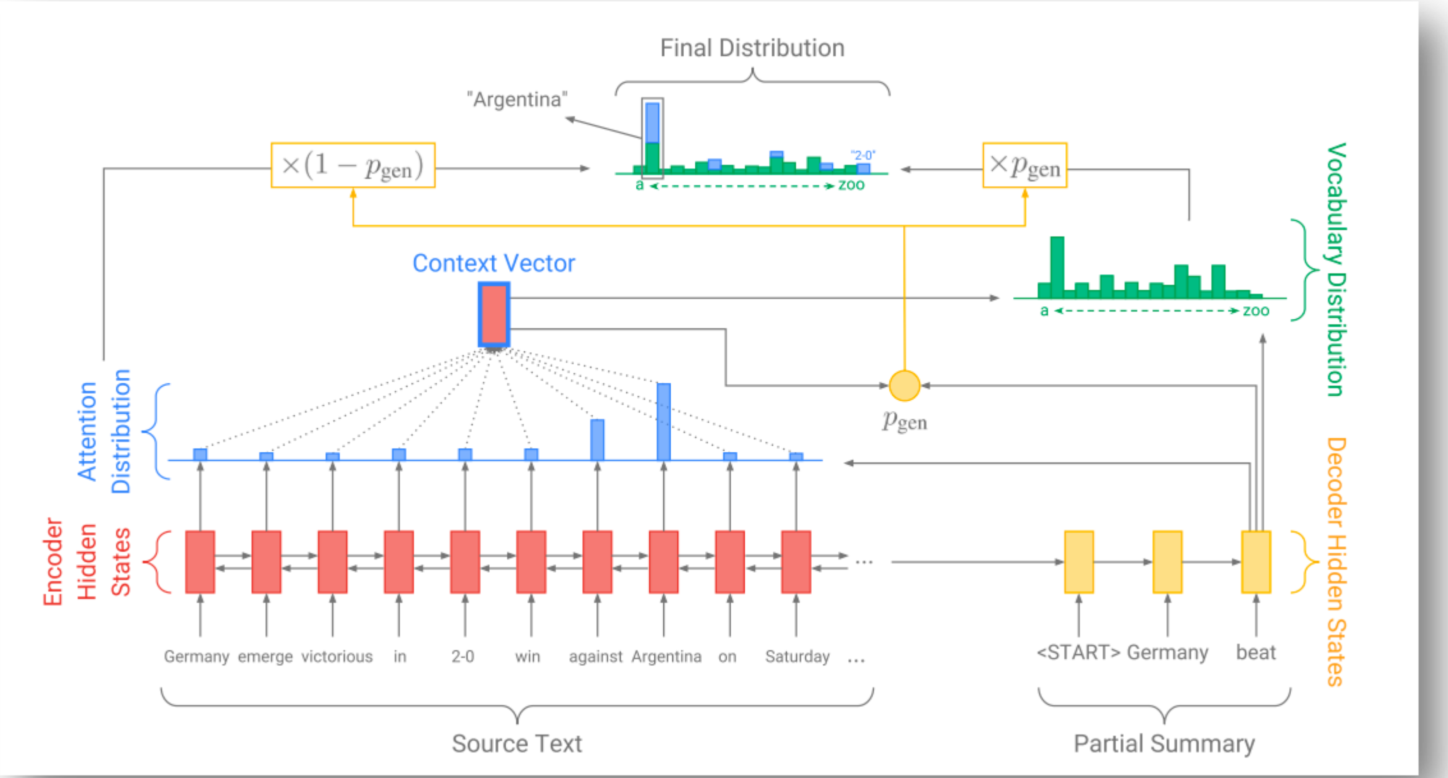




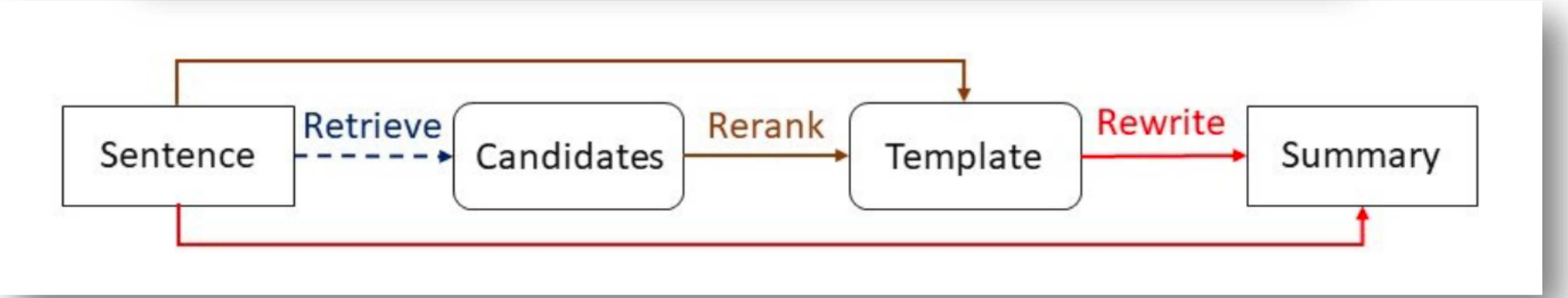
Cloud Theme Knowledge Card/Guide Generation

NLG model

PGN:



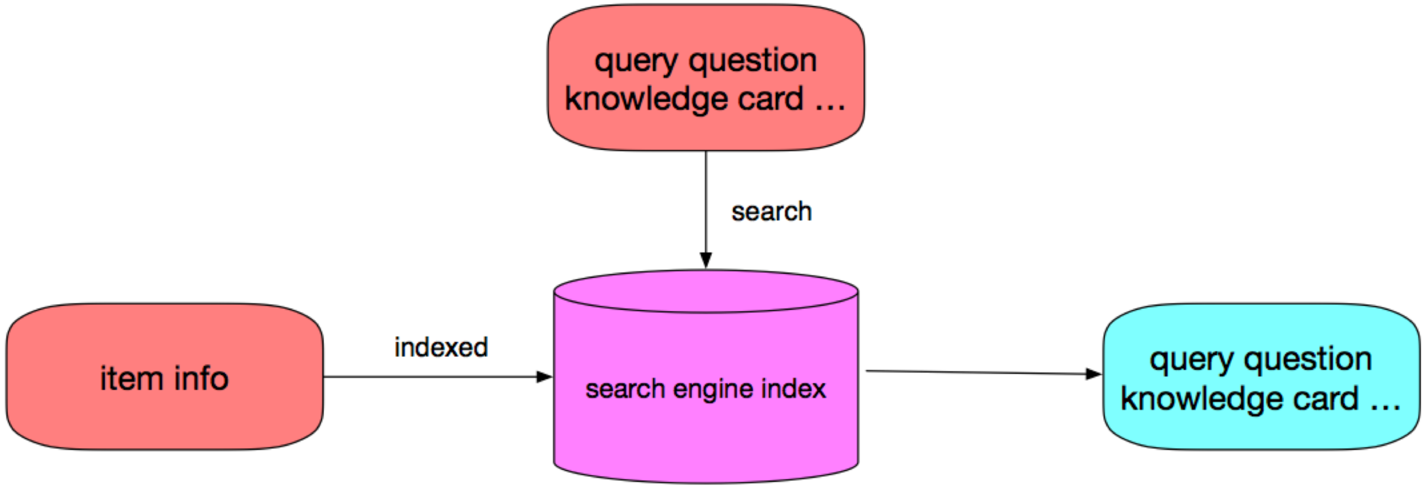
Soft template



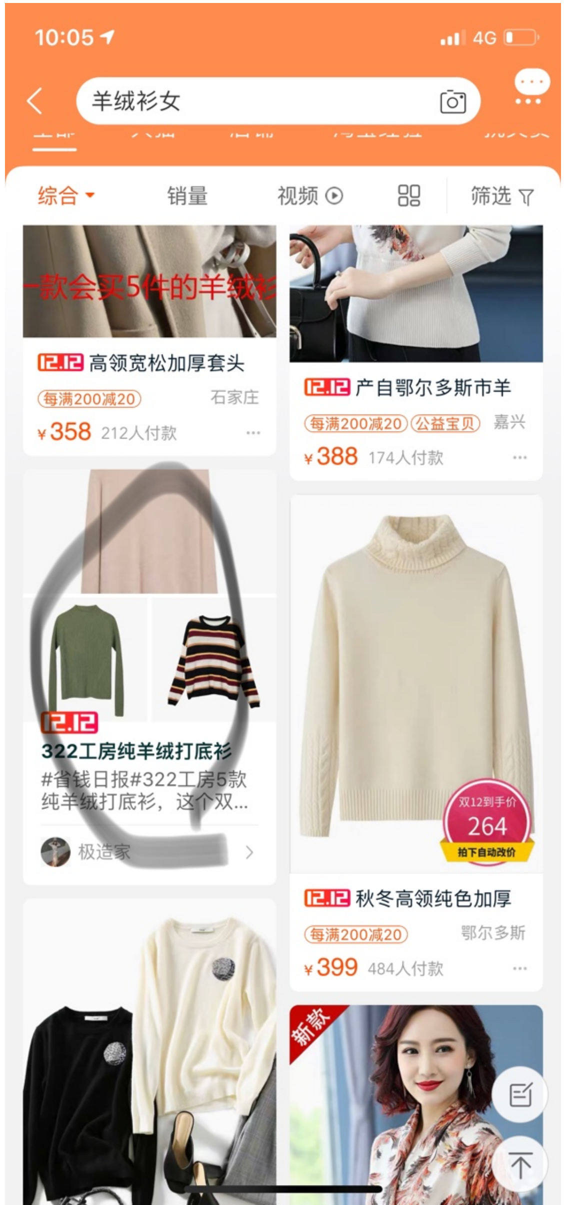
Cloud Theme Knowledge Card/Guide Generation

Product embedding
Currently, items are retrieved through query cluster, keywords, question, and knowledge card;

Query cluster: Autumn suit, female student, fresh, college style...
Keywords: campus, in the Mood for Love, golden years
Question: Do you know the style of female students?
Knowledge card: We are at the border of immaturity and maturity. If we want to be a campus goddess, of course we have to choose what suits us. To become a dazzling scenery, we should live the most beautiful look. Come to the small society of university...



Cloud Theme Knowledge Card/Guide Generation



Part 6：What is next?

Current Focus and Challenges

- Recommendation pipeline for Alibaba: billions of users and products with over trillions of edges and extremely rich attributes
    - Algorithm + System Co-optimization

- Graph embedding & inference for attributed heterogeneous network
    - ✓ Heterogeneous nodes with attributes
    - ✓ Different types of edges, easy to integrate knowledge (Knowledge Graph and Graph Embedding)
    - ✓ For each specific task, best choice of edge type & node attributes

- Multi-modality & Scalable Bayesian Deep Learning

- Introducing high quality data labels
    - ✓ High quality side information, such as high valuable users and merchandise
    - ✓ Targeted positive and negative samples, node & edge importance weights

- Online Inference
    - ✓ New data inflow, real-time graph update
    - ✓ Can handle sequential information

- AliGraph Echo System On Cloud

AliGraph-Open Source

Welcome to Try out

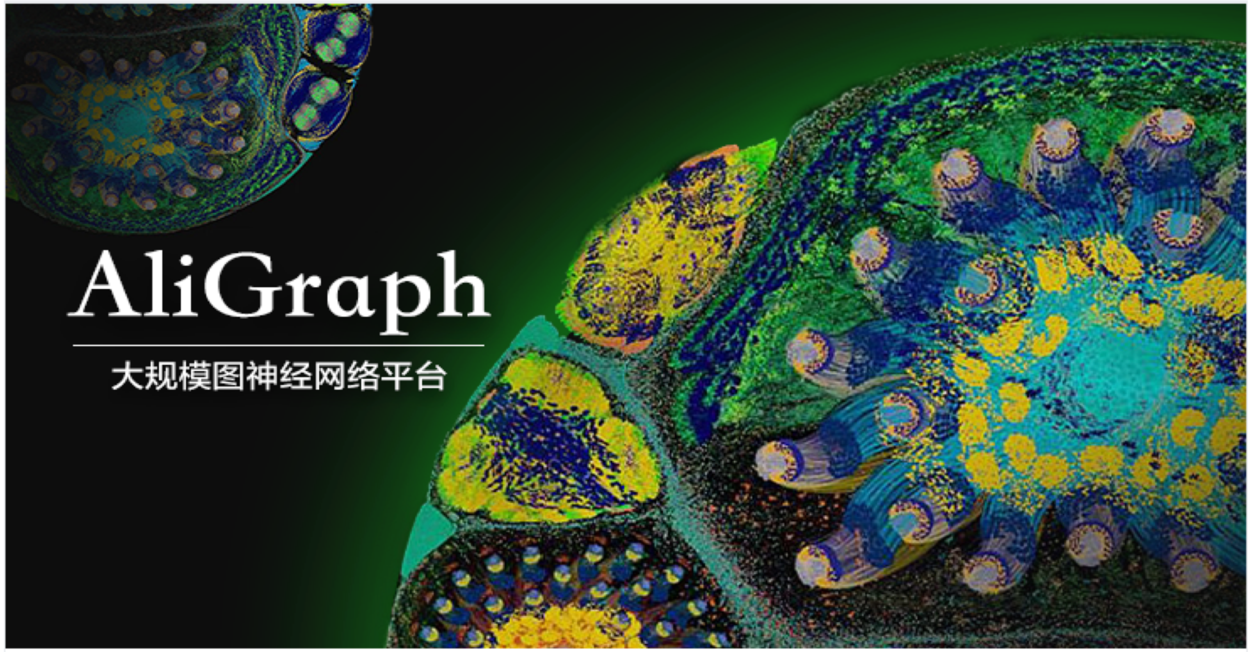
SYSTEM+ALGORITHM

Code:

https://github.com/alibaba/graph-learn

Q/A：

graph-learn@list.alibaba-inc.com

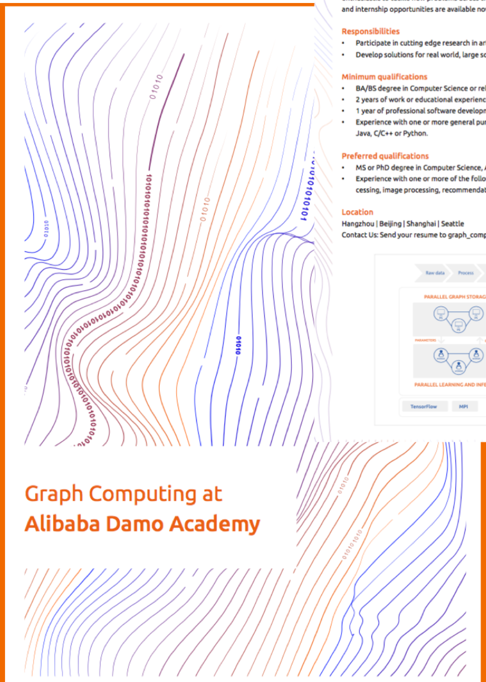


AliGraph：大规模图神经网络平台

AliGraph是新一代大规模图神经网络平台，其认知计算模型比起现有的深度学习技术有了突破性进展，被誉为人工智能2.0模型。

Paper:

AliGraph: A Comprehensive Graph Neural Network Platform, VLDB 2019



Graph Computing at Alibaba Damo Academy

Extremely large scale graphical model has been playing an increasingly important role in big data companies. In particular, graph inference combined with deep learning (e.g., graph embedding) has achieved successful phased results in many of Alibaba's business scenarios. The data of the Alibaba ecosystem is extremely rich and varied, covering everything from shopping, travel, entertainment, and payment. We are working on the development of a new generation of graph learning platform that can efficiently perform inference analysis on billions of nodes and trillions of edges to be deployed in fields including recommendation, warehouse management, risk control, social networks, among many others that affect the core business of Alibaba.

To tackle these challenges, we're looking for scientists/engineers who have both a strong **algorithm design** background and/or strong **system implementation** skills, the latter stemming from the need for efficiently computing tremendous graphs in a distributed fashion. We need our scientists/engineers to be versatile, display leadership qualities and be enthusiastic to tackle new problems across the full-stack as we continue to push technology forward. Both full-time and internship opportunities are available now.

Responsibilities

- Participate in cutting edge research in artificial intelligence and machine learning applications.
- Develop solutions for real world, large scale problems.

Minimum qualifications

- BA/BS degree in Computer Science or related technical field or equivalent practical experience.
- 2 years of work or educational experience in Machine Learning or Artificial Intelligence.
- 1 year of professional software development experience.
- Experience with one or more general purpose programming languages including but not limited to: Java, C/C++ or Python.

Preferred qualifications

- MS or PhD degree in Computer Science, Artificial Intelligence, Machine Learning, or related technical field.
- Experience with one or more of the following: graph model, knowledge graph, risk control, natural language processing, image processing, recommendation systems, targeting systems, ranking systems or similar.

Location

Hangzhou | Beijing | Shanghai | Seattle

Contact Us: Send your resume to graph_compute@alibaba-inc.com

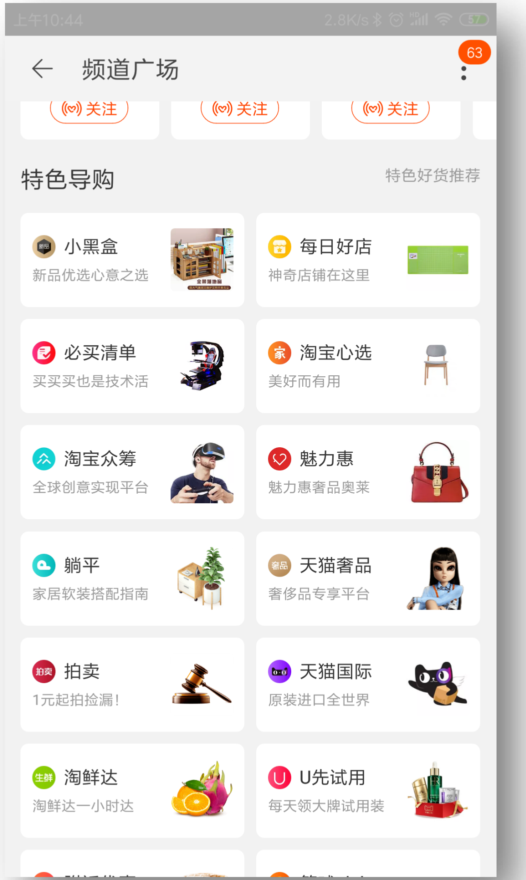
Thanks

Xiaoyong Liu : xiaoyong.liu@alibaba-inc.com
Hongxia Yang : yang.yhx@alibaba-inc.com
Wei Lin : weilin.wl@alibaba-inc.com

Part Appendix： Cognitive

Master Context

Channels Page



Taobao Experience

XiaoHongShu

Merchandise Context

WeiTao

Product Title

Product details

User Context

Evaluation

Asking Everyone

Query

Value, Application and Landing of NLG

Result Display:

中长款女童保暖呢子外套这款毛呢大衣,表面纯色和内里色调形成气质撞色,整体看上去尽显气质和个性,如果家中的宝宝正是一个特别有想法的孩子,个性十足,那么这款外套千万不能错过!作为基础款大衣,这款的不同之处就在与颜色,搭配黑色毛衣,炫酷一姐!

Mid-length girl's warm woolen coat, this woolen coat, the surface solid color and the inner tone form a temperament contrast, the overall look is full of temperament and personality, if the baby in the family is a particularly thoughtful child, full of personality, then Don't miss this coat! As a basic coat, the difference between this one lies in the color, with a black sweater, and a cool sister!

春装新款的中长款羊绒呢大衣,这款大衣采用优质的羊绒面料,柔软舒适,穿着舒适.双排扣的设计,简约大方,时尚百搭.搭配一条简单的牛仔裤,就可以美美的出门了.

The new mid-length cashmere coat for spring. This coat is made of high-quality cashmere fabric, soft and comfortable, and comfortable to wear. The double-breasted design, simple and elegant, fashionable and versatile. With a simple pair of jeans, you can go out beautifully .

小公主韩版加厚毛呢大衣这款黑色毛衣采用了经典的圆领设计,穿起来非常显气质,衣服的袖口和衣摆都做了螺纹收口,穿着更加舒适.衣身两侧的大口袋装饰,给衣服增加了一点童趣的味道,穿上很有气质哦.

Little princess Korean thick woolen coat. This black sweater adopts a classic round neck design, which is very elegant to wear. The cuffs and hem of the clothes have threaded closures, making it more comfortable to wear. The large pockets on both sides of the body are decorated, It adds a little childlike taste to the clothes, and it is very temperamental.

型色缤纷的春装新款.韩版气质的羊绒呢大衣,中长款不挑身材,3d立体剪裁,纽扣设计别致,贴布精致,不可或缺的一款.搭配尖头方扣浅口单鞋,绒面粗跟简直爱不释手,颜色也是很春天的颜色.

A new style of colorful spring clothes. Korean style of temperament cashmere coat, medium and long models do not pick up the figure, 3D three-dimensional tailoring, button design unique, delicate patch, an indispensable one. With pointed square button shallow shoes, velvet The thick noodles are simply too much to put it down, and the colors are also very spring colors.

Which one is Machine Generated?

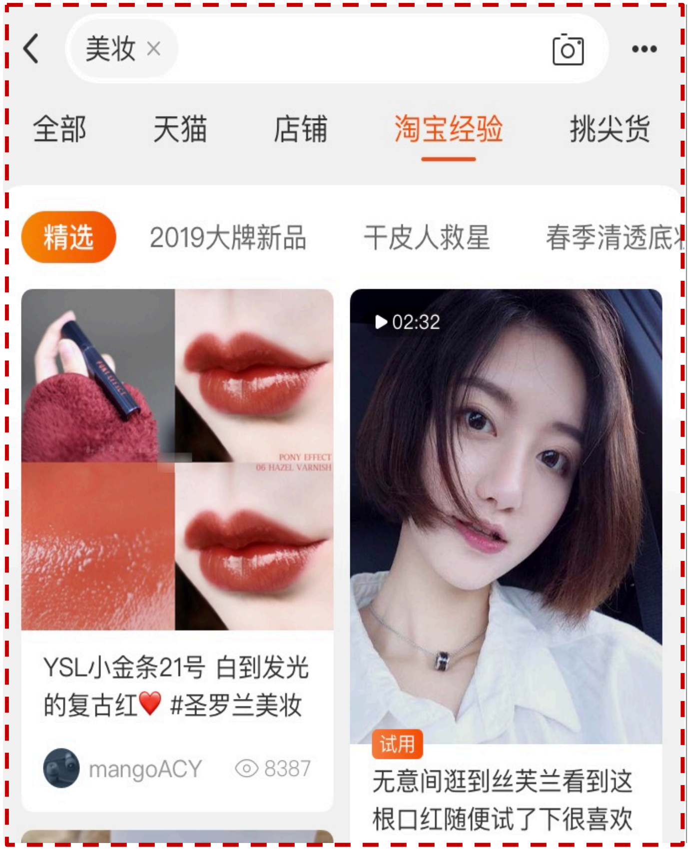
Note：Generate Chinese then translate to Englis

Short video front desk service

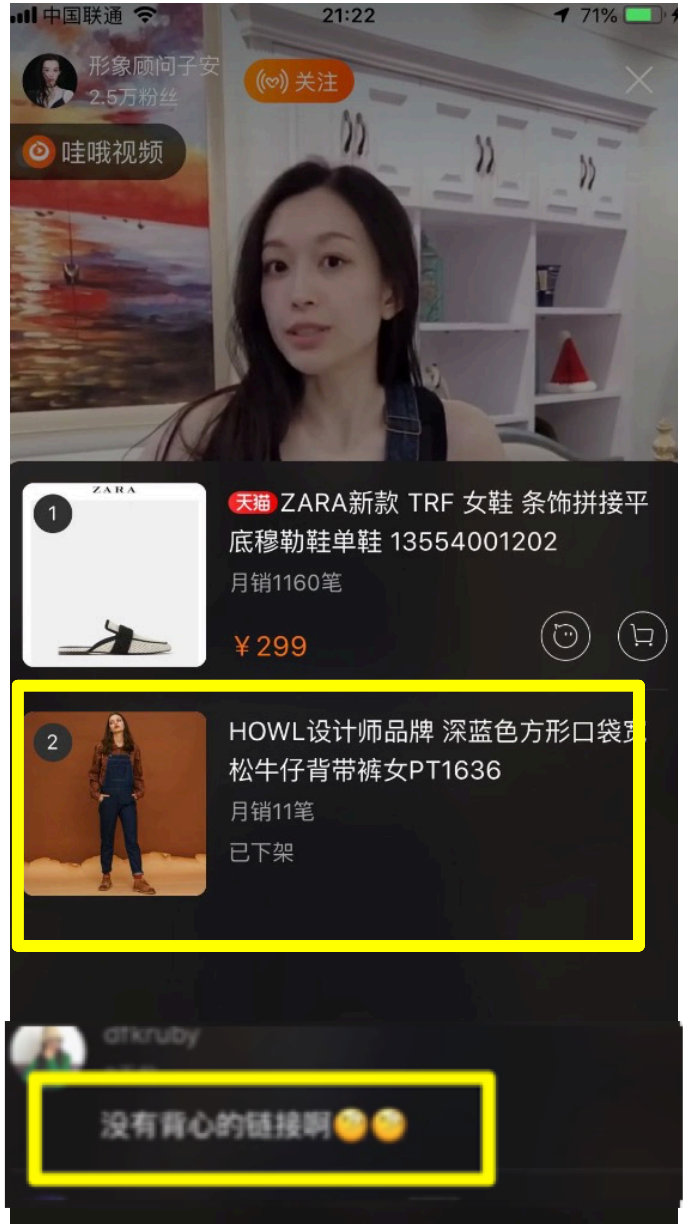
Guess you like the video recommendation

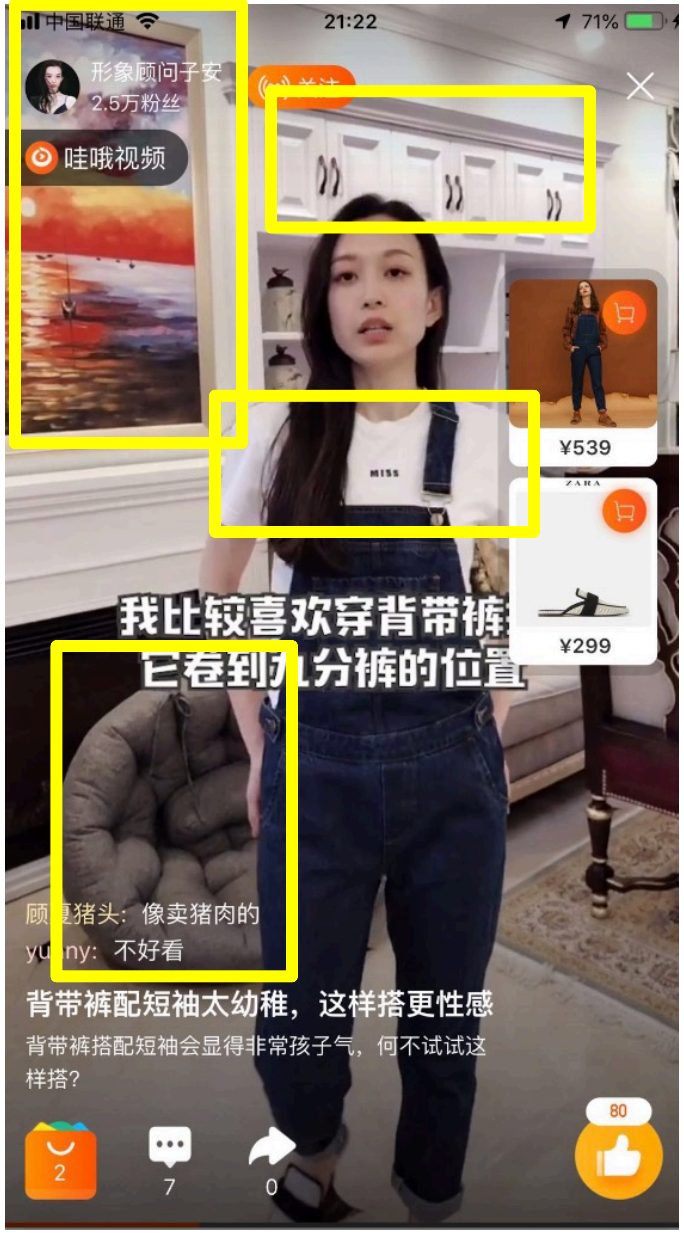
Taobao experience video content

Wow video



Video product detection



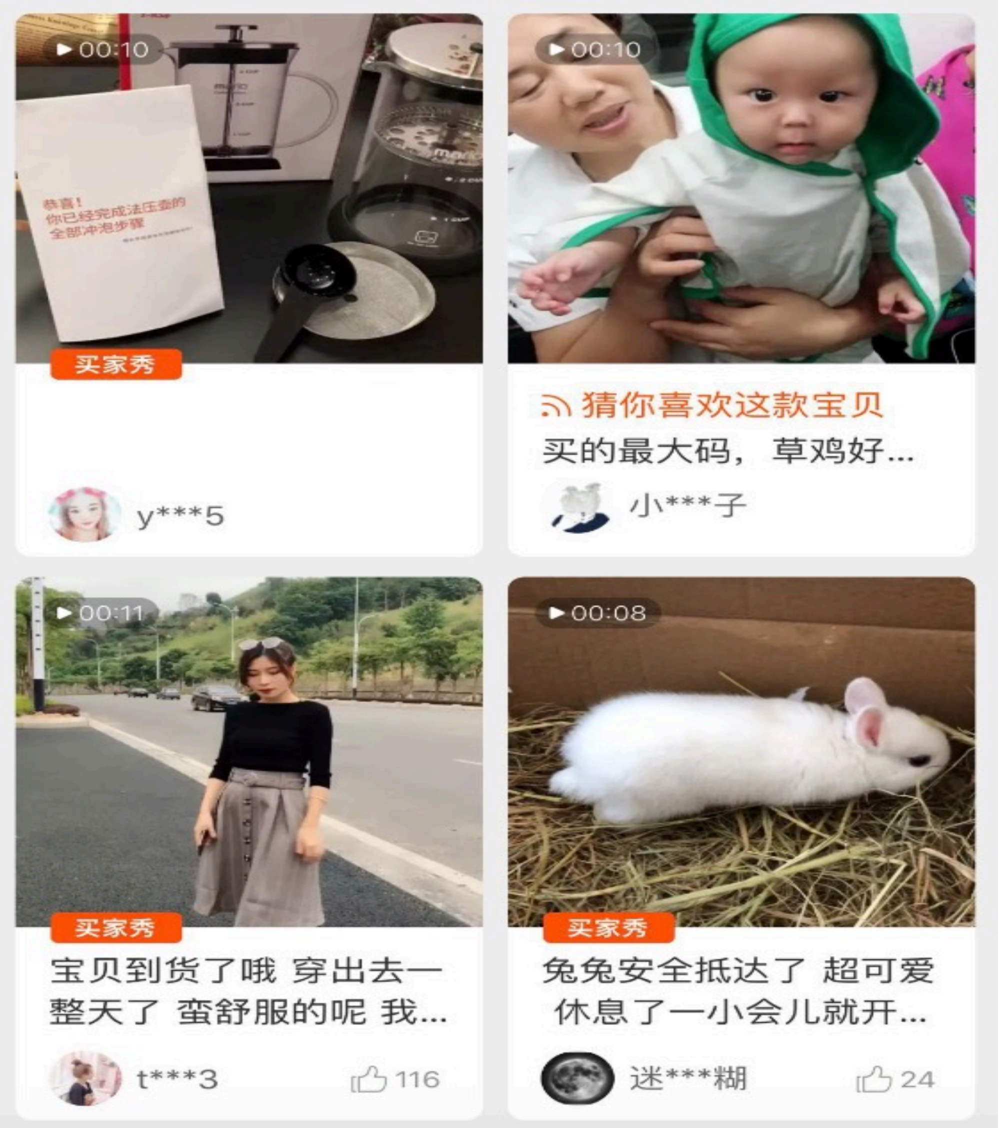


Video product detection---image target detection

- From the expert videos, the videos with higher products are selected as training data
- Generate a multi-product category data set with a single product image model
- Weak supervised object detection+ Video description infor
- Manual marking +few shot learning

Video summary generation-1

- Video text description is the basic input information for video understanding
- Evaluate a large number of undescriptive and unclear descriptions in the video



Video summary generation-2

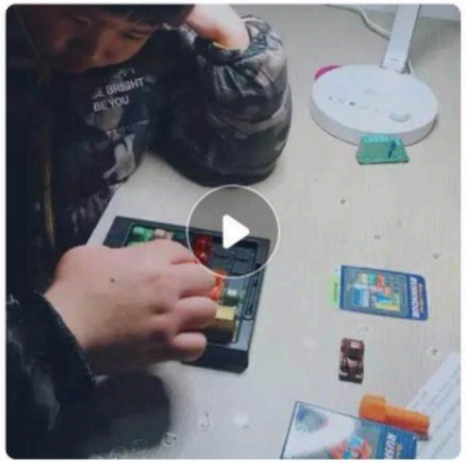
- Video + seq2seq model
- Video content + review text content + product attributes, produce short video abstracts
  - Long sentences short
  - Long short sentences
  - Add Product Subject
- Manually filter the training data set from the videos with high views and high likes of users

Kid Toy Album

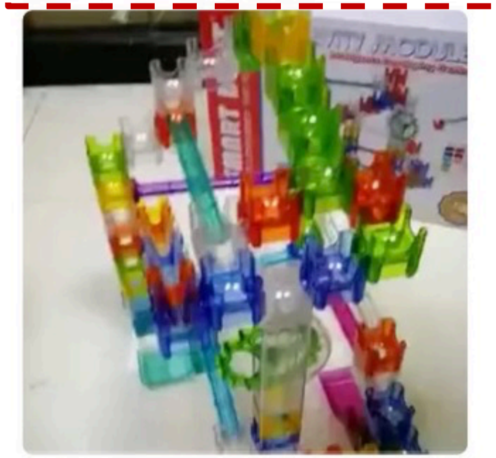
2019-03-22 颜色分类:豪华版 实测7岁以上 送迷宫

因为同学的推荐。又因为个人手机上玩过类似游戏。果断入手。宝贝很欢喜。既能开发小脑，又可以让其抛弃电子产品，提高专注力。杠杠滴！希望用不了多久，宝贝破解技能超过老母。

很好玩！除了图纸造型外，孩子还搭了些其他造型。店家发货速度快，质量跟专柜对比一样，应该是正品！重点是真的很好玩！已经推荐给周围的朋友了。



Develop the cerebellum to improve concentration toys

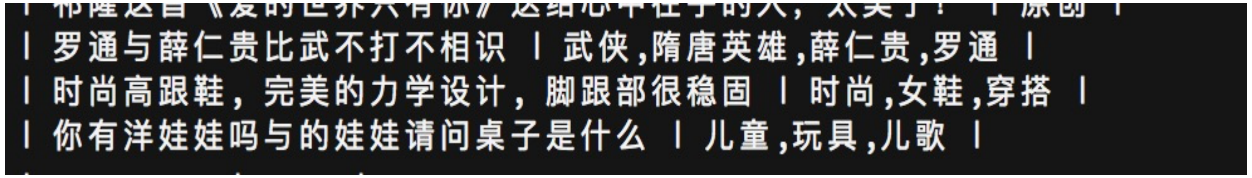


Variety of styles and counter quality

Video summary generation-3

- Youku： Film and television comprehensive video slices,

- Training data: high-quality short video pool with complete titles
- Tags Auxiliary title generation


罗通与薛仁贵比武不打不相识	武侠,隋唐英雄,薛仁贵,罗通
时尚高跟鞋，完美的力学设计，脚跟部很稳固	时尚,女鞋,穿搭
你有洋娃娃吗与的娃娃请问桌子是什么	儿童,玩具,儿歌

Personalized content themes

Alibaba Group 阿里巴巴集团

- Generate personalized content themes composed of products, graphics, and videos
- The video quality evaluation model automatically selects products from Youku short videos
  + hand-made short videos
- Gag recall, content-related recall
- Video sorting, video product mixing