2018

第十一期

360互联网技术训练营

360算法技术解密与实践

图文信息流推荐中基于词向量的至简实践

司建锋
360信息流产品部

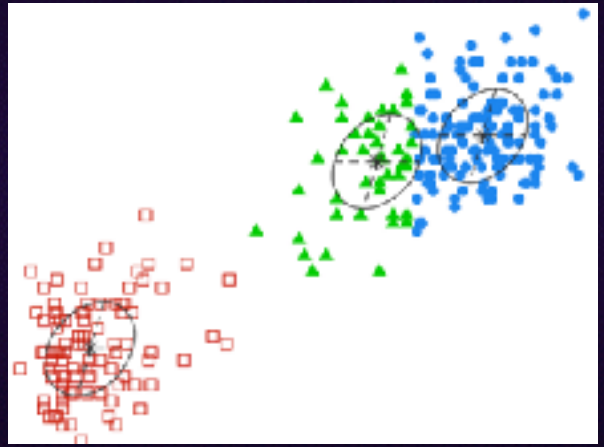
360技术

目录

- 背景介绍
- 技术方案
- 应用案例
- 总结

背景介绍

- 信息流标签系统
  - 面向资讯文本的NLP相关基础服务;
  - 语义级别理解和精准刻画;
    - 分类
    - 语义聚类
    - 关键词
    - 地域
    - 等等



背景介绍

- 信息流标签系统
  - Tag体系建设
    - 完备
    - 与时俱进
    - 可运营
  - 文本相关性
    - 聚焦

背景介绍

- 文本相关性
  - A) 深度语义匹配技术在360搜索的应用实践
  - B) 图文信息流推荐中基于词向量的至简实践
  - C) NMT 生成式广告触发模型
  - D) 强化学习在排序问题中的应用

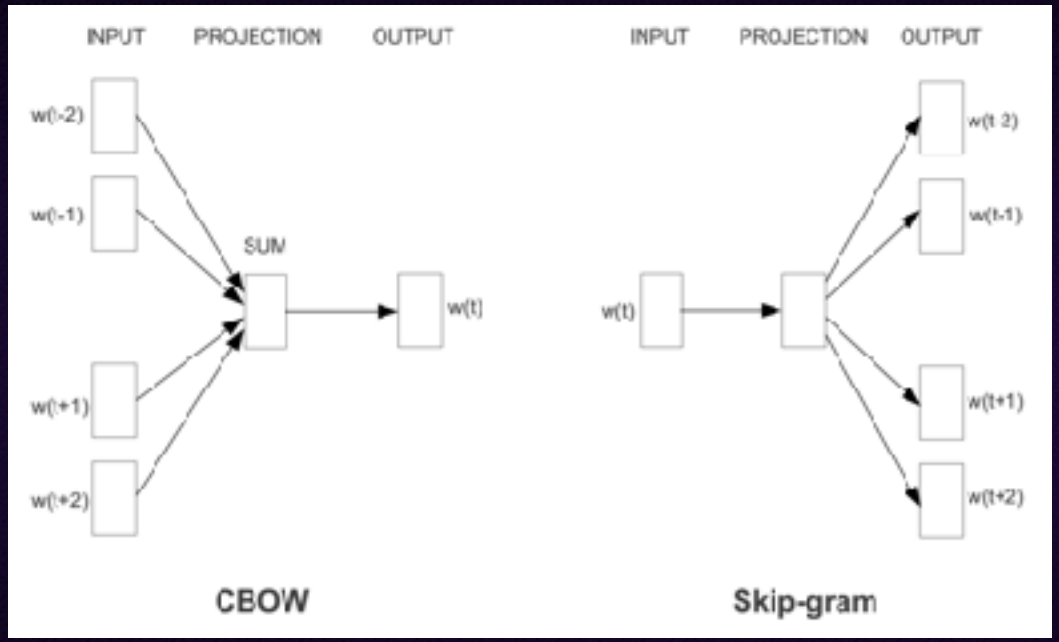
	A	B	C	D	
A	1	0.78132242	**0.817534268**	0.660987079	0.753281
B	0.78132242	1	0.718142271	0.650507927	0.716658
C	0.817534268	0.718142271	1	**0.610729158**	0.715469
D	0.660987079	0.650507927	0.610729158	1	0.640741

技术方案

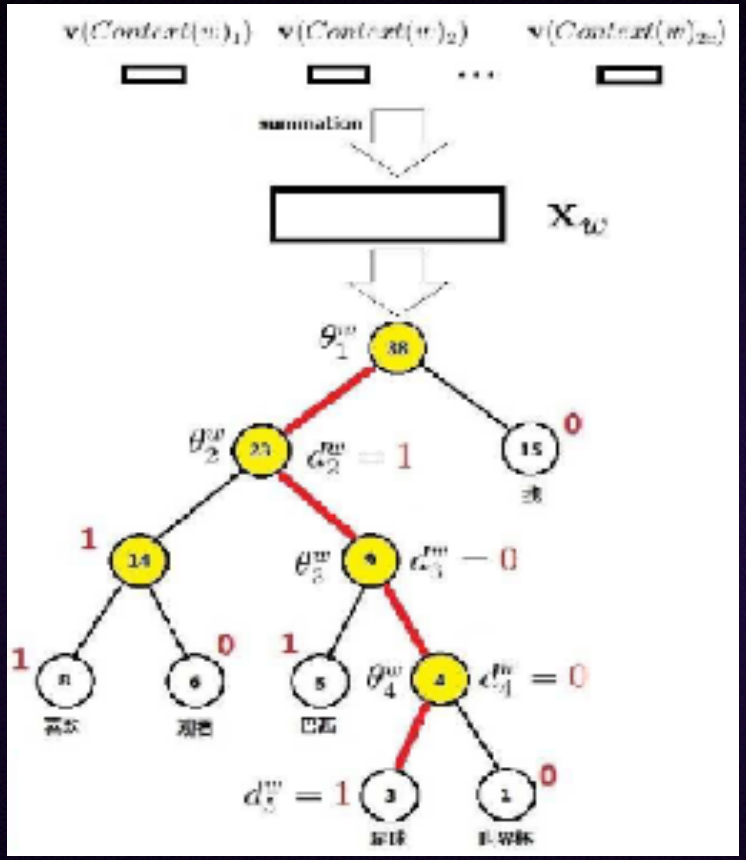
- Embedding based solution
  - Word/Sentence Embedding
  - User Embedding
- 向量空间检索
  - FAISS

Word Embedding

- *Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient Estimation of Word Representations in Vector Space. In Proceedings of Workshop at ICLR, 2013.*



Word Embedding



$$g(w) = \prod_{u \in \{w\} \cup NEG(w)} p(u|Context(w)),$$

$$p(u|Context(w)) = \begin{cases} \sigma(\mathbf{x}_w^\top \theta^u), & L^w(u) = 1; \\ 1 - \sigma(\mathbf{x}_w^\top \theta^u), & L^w(u) = 0. \end{cases}$$

$$p(u|Context(w)) = \left[\sigma(\mathbf{x}_w^\top \theta^u)\right]^{L^w(u)} \cdot \left[1 - \sigma(\mathbf{x}_w^\top \theta^u)\right]^{1-L^w(u)},$$

$$g(w) = \sigma(\mathbf{x}_w^\top \theta^w) \prod_{u \in NEG(w)} \left[1 - \sigma(\mathbf{x}_w^\top \theta^u)\right],$$

$$G = \prod_{w \in \mathcal{C}} g(w)$$

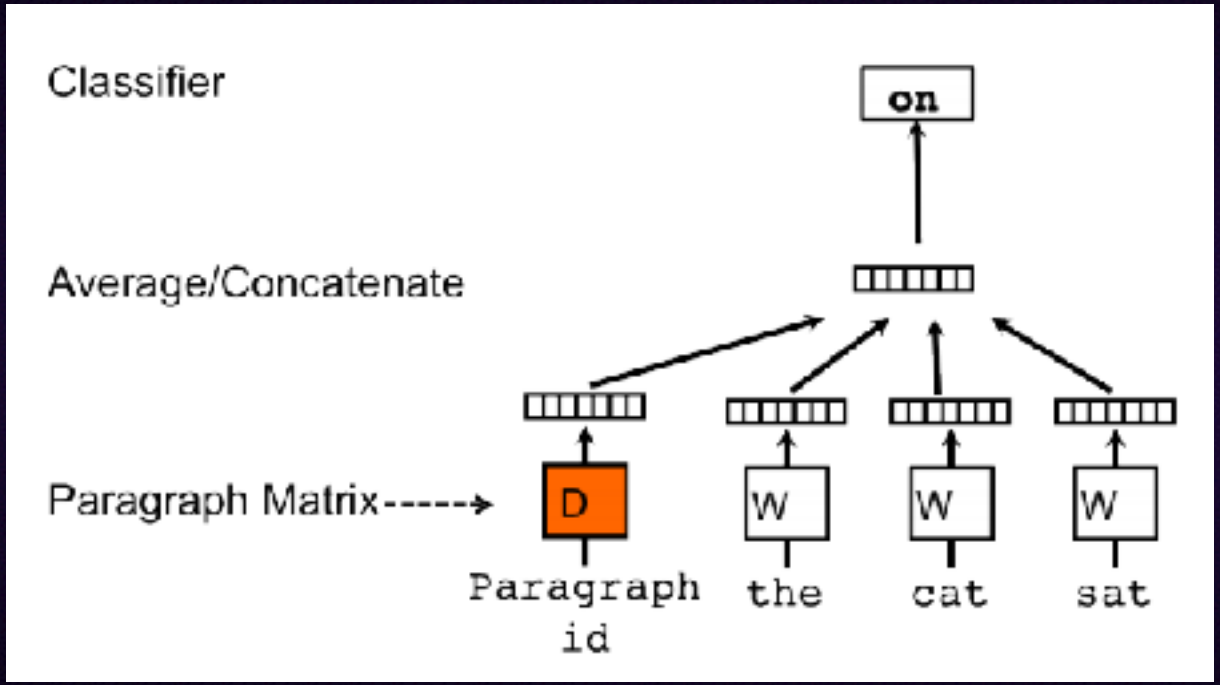
Sentence Embedding

- Within Sentence
  - CBOW/Skip-gram
  - Paragraph vector
  - SDAEs
  - sent2vec
- Between Sentences
  - SkipThought
  - FastSent
- Model Inference
  - Paragraph vector
  - SkipThought
  - SDAEs
- No Model Inference
  - CBOS/Skip-gram
  - FastSent
  - sent2vec

Sentence Embedding (PV-DM)

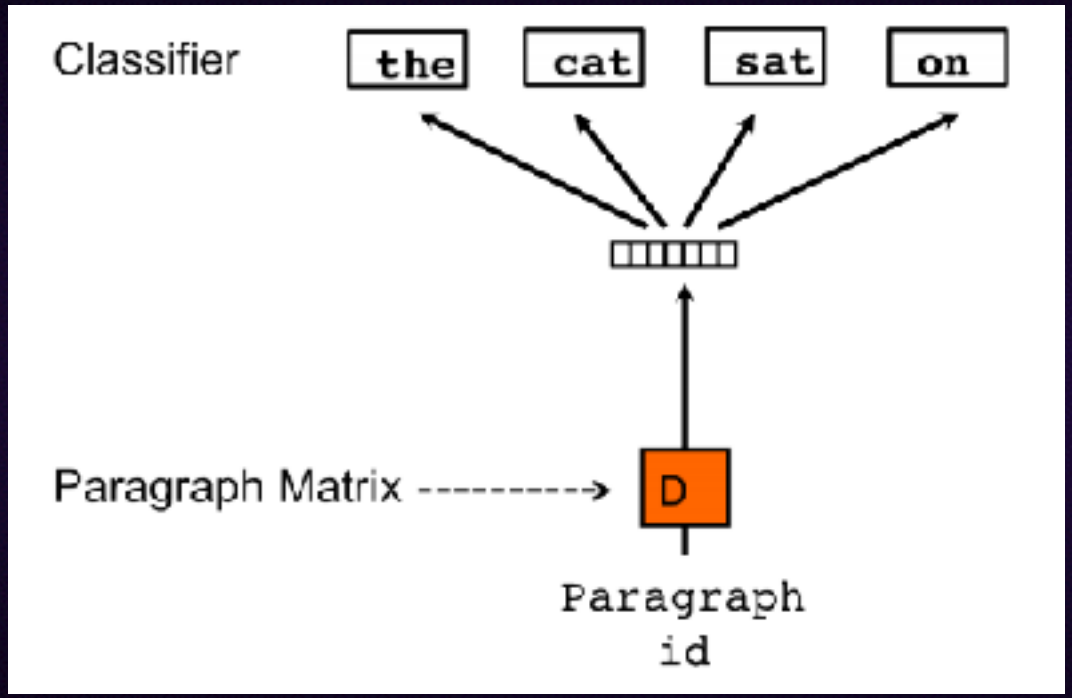
Quoc V. Le, and Tomas Mikolov, ``Distributed Representations of Sentences and Documents ICML", 2014.

Distributed Memory Model of Paragraph Vectors (PV-DM)



Sentence Embedding (PV-DBOW)

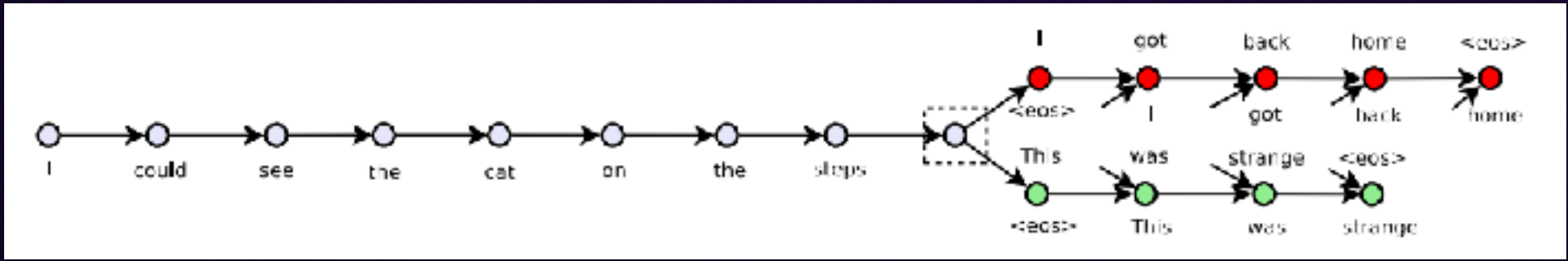
Distributed Bag of Words version of Paragraph Vector (PV-DBOW)



Sentence Embedding (SkipThought)

Ryan Kiros, Yukun Zhu, Ruslan Salakhutdinov, Richard S. Zemel, Antonio Torralba, Raquel Urtasun, and Sanja Fidler. "Skip-Thought Vectors." arXiv preprint arXiv:1506.06726 (2015).

I got back home. I could see the cat on the steps. This was strange.



Sentence Embedding (SkipThought)

Objective. Given a tuple (s_{i-1}, s_i, s_{i+1}), the objective optimized is the sum of the log-probabilities for the forward and backward sentences conditioned on the encoder representation:

$$\sum_t \log P(w_{i+1}^t | w_{i+1}^{<t}, \mathbf{h}_i) + \sum_t \log P(w_{i-1}^t | w_{i-1}^{<t}, \mathbf{h}_i) \quad (10)$$

im sure youll have a glamorous evening , she said , giving an exaggerated wink .
im really glad you came to the party tonight , he said , turning to her .

“ i ’ll take care of it , ” goodman said , taking the phonebook .
“ i ’ll do that , ” julia said , coming in .

Sentence Embedding (SDAEs/FastSent)

Hill, F. Cho, K. & Korhonen, A. 2016. Learning Distributed Representations of Sentences from Unlabelled Data. NAACL 2016

- Sequential Denoising Autoencoders (SDAEs)
  - Remove words
  - Swap words
- FastSent

On prediction:

$$\mathbf{s} = \sum_{w \in S} u_w .$$

$$\mathbf{s_i} = \sum_{w \in S_i} u_w$$

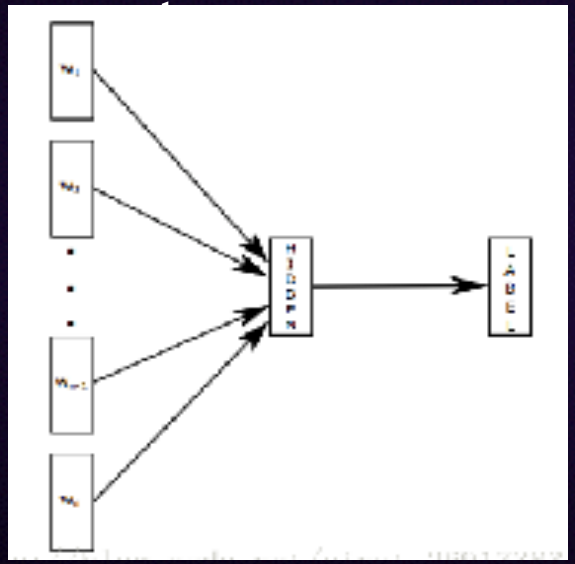
$$\sum_{w \in S_{i-1} \cup S_i \cup S_{i+1}} \phi(\mathbf{s_i}, u_w) . \quad \sum_{w \in S_{i-1} \cup S_{i+1}} \phi(\mathbf{s_i}, v_w)$$

Sentence Embedding (sent2vec)

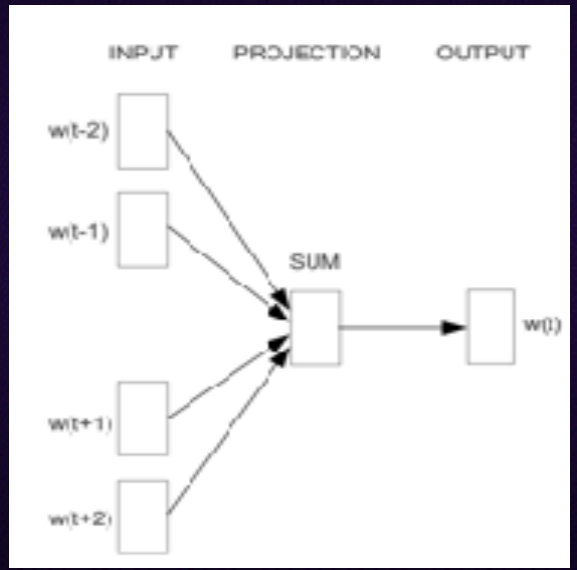
Matteo Pagliardini, Prakhar Gupta, Martin Jaggi, Unsupervised Learning of Sentence Embeddings using Compositional n-Gram Features NAACL 2018

A. Joulin, E. Grave, P. Bojanowski, T. Mikolov, Bag of Tricks for Efficient Text Classification, ACL 2017

FastTex



Sent2Vec



On prediction:

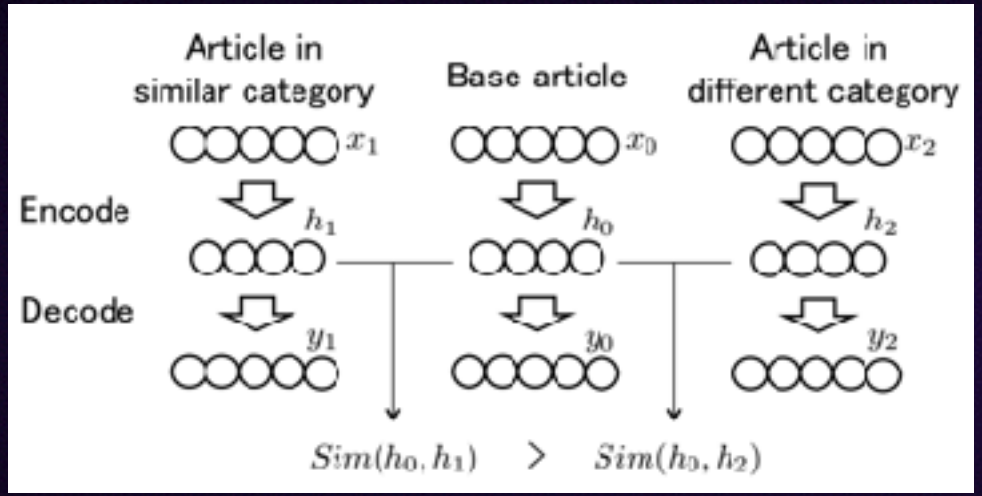
$$\mathbf{s} = \sum_{w \in S} u_w .$$

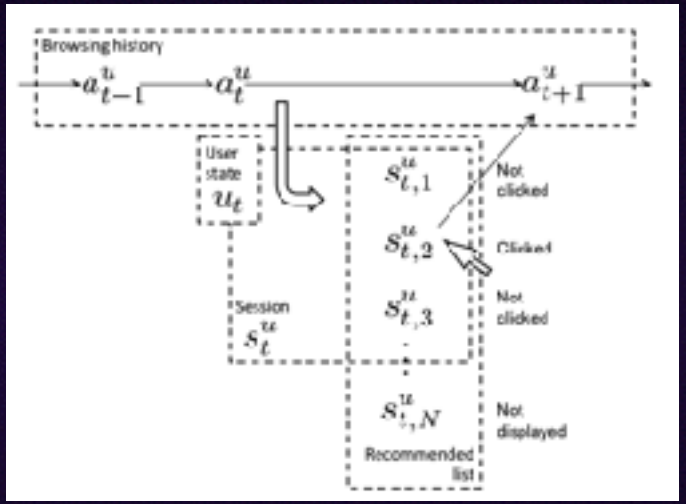
Sentence Embedding (sent2vec)

```
void FastText::sent2vec(Model& model, real lr,  const std::vector<int32_t>&
line){
 // …..
for (int32_t i=0; i<line.size(); ++i)    {
    if (uniform(model.rng) > dict_->getPDiscard(line[i])
            || dict_->getTokenCount(line[i]) < args_->minCountLabel)
      continue;
    context = line;
    context[i] = 0;
    dict_->addNgrams(context, args_->wordNgrams, args_->dropoutK,
model.rng);
    model.update(context, line[i], lr);
  }
}
```

User Embedding

- *Shumpei Okura ;Yukihiro Tagami ;Shingo Ono;Akira Tajima, Embedding-based News Recommendation for Millions of Users, KDD 2017*





StarSpace: Embed All The Things!

- *Ledell Wu, Adam Fisch, Sumit Chopra, Keith Adams, Antoine Bordes, Jason Weston, StarSpace: Embed All The Things, arXiv:1709.03856v5 [cs.CL] 21 Nov 2017*

FAISS 向量空间检索

- *Johnson, Jeff and Douze, Matthijs and Jegou, Herve, Billion-scale similarity search with GPUs, , arXiv preprint arXiv:1702.08734, 2017*
- *H. Jegou, M. Douze, and C. Schmid. Product quantization for nearest neighbor search. IEEE Trans. PAMI, 33(1):117–128, January 2011.*

- A library for efficient similarity search and clustering of dense vectors.

- Approximate Nearest-Neighbor Search
  - Two levels of quantization
    - q_1 coarse quantization
    - q_2 fine quantization, which is built on the residual of q_1
    - $y \approx q(y) = q_1(y) + q_2(y - q_1(y))$
  - An inverted file system with the asymmetric distance computation (IVFADC)
    - Index on q_1

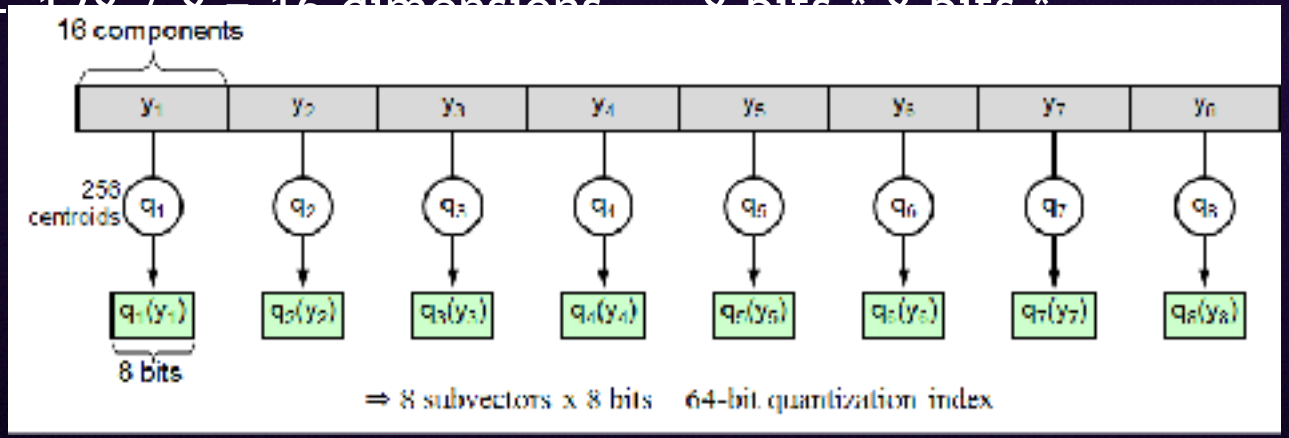
FAISS 向量空间检索：q_1

K-Means

$$L_{\mathrm{IVFADC}} = \underset{i=0:\ell \text{ s.t. } q_1(y_i)\in L_{\mathrm{IVF}}}{k\text{-argmin}} \|x - q(y_i)\|_2.$$

FAISS 向量空间检索：q_2

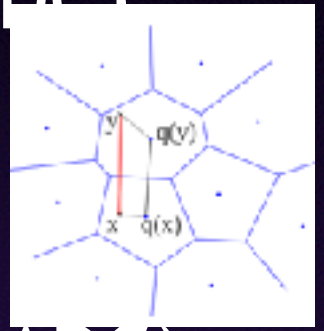
- Product quantizer
  - 2^2=4, 2^4=16, 2^8=256, …, 2^64=18446744073709551616 (=256^8)
  - 128 dimensions, 64 bits
  - 128 / 8 = 16 dimensions, 8 bits * 8 bits *



FAISS 向量空间检索: $Distance(x, y)$

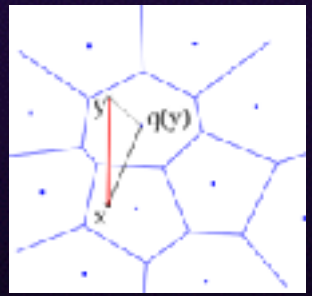
- Symmetric distance computation (SDC)

$$\hat{d}(x,y) = d(q(x), q(y)) = \sqrt{\sum_j d(q_j(x), q_j(y))^2},$$



- Asymmetric distance computation (ADC)

$$\tilde{d}(x,y) - d(x, q(y)) - \sqrt{\sum_j d(u_j(x), q_j(u_j(y)))^2},$$

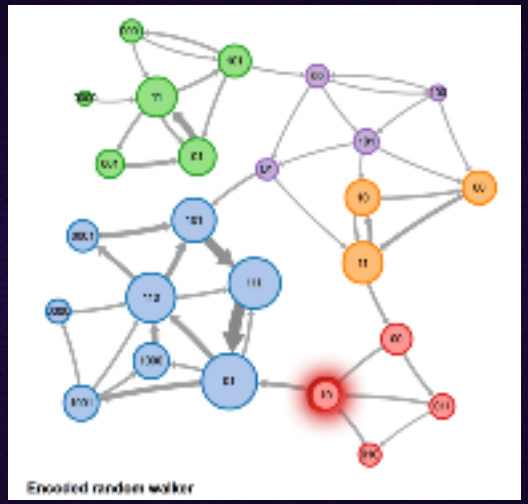


应用案例

- 语义簇
  - 1.0
  - 2.0
  - 3.0
  - 4.0
- 相关新闻
  - 1.0
  - 2.0

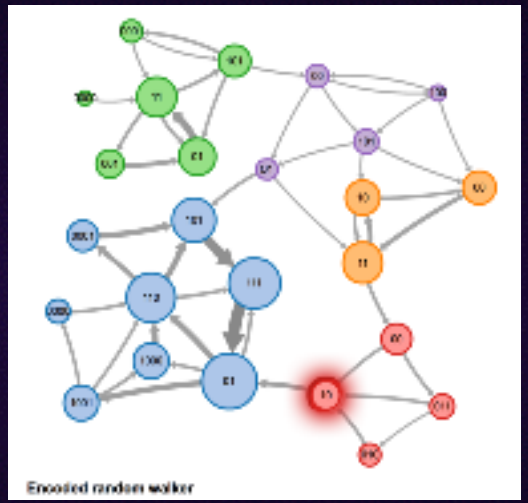
应用案例 (cluster 1.0)

- News titles + keyword， one-hot编码，计算cosine距离进行聚类
- 聚类算法：
  - infomap
- 优点：
  - 第一次引入聚类
- 缺点：
  - 存在语义鸿沟，解释性较差
  - 不方便更新， 维护成本高



应用案例 (cluster 2.0)

- 全量实体词聚类， 按词的word2vec向量计算cosine距离；
- 聚类算法：
  - infomap
- 优点：
  - 概念较稳定，一次成型
  - 可解释性强， 方便运营；
- 缺点：
  - 有广度，没深度
  - 容易出现超大簇（比如明星簇）



应用案例 (cluster 3.0)

- 千万级news titles， 通过sent2vec进行向量化；
- 聚类算法：
  - MPI + Fast-Kmeans (直接利用簇心向量做预测)
- 优点：
  - Sentence级别的语义空间，训练和预测保持一致
  - 聚类够精准(广度+深度)
  - 在固定语义空间，实现增量训练 (sent2vec模型日更新，定期引入新词/新簇)

应用案例 (cluster 4.0)

- cluster 4.0 是什么?

应用案例 (4.0)

应用案例 (聚类+tag体系)

语义簇	tag体系	相似度
science:3417:阿里云:云计算	科技|IT产业|云计算	0.899106
science:3798:python:爬虫	科技|基础理论|计算机技术	0.897999
science:5627:无人驾驶:自动驾驶	科技|人工智能|无人驾驶	0.896513
science:2061:共享单车:单车	科技|互联网|共享经济	0.886559
science:14214:比特币:数字货币	科技|区块链|数字货币	0.874244
science:15261:小米:华为	科技|通信|手机终端	0.869865
science:7419:数博:贵阳	科技|IT产业|大数据	0.869369
science:12100:索尼:富士	科技|数码|相机	0.86599
science:3796:联通:电信	科技|通信|运营商	0.865448
science:8070:勒索病毒:比特币	科技|互联网|互联网安全	0.86078
science:10326:ofo:摩拜	科技|互联网|共享经济	0.858769
science:2340:尼康:d850	科技|数码|相机	0.858523
science:15300:魅族:黄章	科技|通信|手机终端	0.857771
science:8499:储能:aes	科技|数码|相机	0.85741

应用案例 (相关新闻)

- 相关新闻
  - 1.0 (离线计算)
    - Title, content, keyword, 分类，Topic Modeling，规则，etc.
    - Hadoop MR
    - 维护成本高
  - 2.0 (在线计算)
    - Sentence Embedding
    - + FAISS
    - + 简单业务逻辑
      - 真的简单么?

应用案例 (相关新闻)

query: "美俄升级巡航导弹 中国实力如何？“
(美:0.324623 俄:0.521765 升级:0.211953 巡航导弹:0.783746 中国:0.383207 实力:0.289951 巡航:0.463009 导弹:0.732088)

TOP-K	相似度
title: "美俄升级巡航导弹 中国具备后发优势但缺实战检验",	0.92[illegible]69,
title: "美俄封锁也没用，中国兄弟就是多，俩伙伴助中国研制先进巡航导弹",	0.91[illegible]35,
title: "俄军核动力巡航导弹到底如何？50年前美苏都搞过，我军直接放弃了",	0.91[illegible]5,
title: "如果印度发射巡航导弹，中国的防空系统能拦截多少？",	0.90[illegible]72,
title: "轰-6K战力全面升级，可挂载6枚长剑10巡航导弹，西方不敢相信！",	0.8929706811904907,
title: "最小的防空导弹，重量仅2公斤将成为巡航导弹克星！",	0.8811995983123779,
title: "大国利器，DF41导弹第十次试射成功，打击能力好不逊色于美俄！",	0.7853902578353882,

应用案例 (相关新闻)

query: "陆毅带火的厨房收纳架，婆婆用了一次就说好，还说要买多几件",
(陆毅:0.474111 厨房:0.626293 收纳架:0.64394 婆婆:0.53074 收纳:0.732534
)

TOP-K	相似度
title: "给老公买它后做饭在也没抱怨了!厨房神器让你爱上厨房",	0.7115
title: "小媳妇下厨房，学会这两招，婆婆老公都 称赞！",	0.7012
title: "新婚夫妻厨房总是太杂乱，6款置物架让老公抢着要烧饭",	0.6952
title: "居家女人大爱的10款现代厨房必备用品，让 厨房收纳更整洁！",	0.6845
title: "越来越多人厨房不装吊柜了，以后都流行装这种，实用又聪明",	0.6838
title: "老婆这样收拾厨房，邻居见了都跟着学，太实用了！",	0.6791
title: "三种适合厨房使用的插座，用完厨房超整洁",	0.6790552735328674,

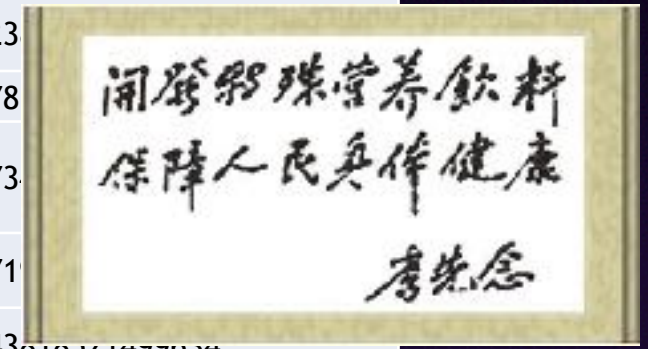
应用案例 (相关新闻)

key: "那个挂牌的猫咪警察终于上任了！",
(挂牌:0.554584 猫咪:0.528889 警察:0.480718 终于:0.299403 上任:0.589807)

TOP-K	相似度
title: “”失驾”, 还敢开车上路? 警察叔叔已经盯上你了",	0.5624265(
title: "西安一司机买假牌照被抓，警车上打110说遇到了假警察",	0.5458707(
title: "他们是警察，他们是父亲！听来自三位警察父亲的声音",	0.5455189(
key: “那个挂牌的猫咪警察终于上任了！” + top 正文	
title: "鹦鹉飞出门迷路了，抱住警察叔叔就哭：请带我回家",	0.6342902183532715,
title: "狗狗因长得太像熊猫，被警察登门拜访调查，尴尬误会让人笑喷",	0.6193301677703857,
title: "为什么警娃在警察爸爸手中服服贴贴？看完漫画就懂了！",	0.6145355701446533,
title: "因为体型太加能，长相凶悍，一出门就被警察盘查"	0.6142921447753906

应用案例 (相关新闻)

key: "李先念书法"，(李先念:0.876618 书法:0.6368)	
TOP-K	**相似度**
title: "实拍李先念故居和杨尚昆故居",	0.7618297934532166,
title: "毛主席：书法临帖最终要形成自己的风格",	0.7171080112457275,
title: "迟浩田书法，具有将军书法中的独有气度",	0.6823
title: "毛泽东主席书法欣赏",	0.6778
title: "他被称为游击天才 和许世友同级李先念给他当过政委 30岁跳河而死",	0.6773
title: "军阀韩复榘书法：一手好楷书，不是略通笔墨，而是精通书法",	0.6771
title: "看完高岗书法，再看看饶漱石书法，同为书法大家，差距却甚远",	0.6743
title: "十四位北大校长的书法，让书法家们汗颜！",	0.6519374251365662,
title: "实拍开国将军谷景生的书法作品，苍劲有力，浩然正气",	0.6481997966766357,
title: "书法家王献之临终愧休郗道茂",	0.6366733312606812,
title: "他是开国中将，手下没有多少兵，却为何能一次救了毛主席，一次救了共产党",	0.6279226541519165,



总结

- 基于各种各样有趣的embedding， 以及向量空间检索的支持，我们可以做很多事情。
  - 数据支持
  - 模型理解
  - 业务导向
  - 创意突破

相关性

A: 深度语义匹配技术在360搜索的应用实践
B: 图文信息流推荐中基于词向量的至简实践
C: NMT 生成式广告触发模型
D: 强化学习在排序问题中的应用

谢谢

360技术