EMBEDDING-BASED RETRIEVAL IN FACEBOOK SEARCH

分享人:胡作梁

CHALLENGES

- The huge scale imposes challenges on both training of embeddings and serving of embeddings. In Facebook search, the search intent does not only depend on query text but is also heavily influenced by the user who is issuing the query and the context where the searcher is.

- Search engine usually needs to incorporate both embedding-based retrieval and term matching based retrieval together to score documents in the retrieval layer.

MODELING

In modeling, we proposed **unified embedding**, which is a two sided model where one side is search request comprising query text, searcher, and context, and the other side is the document. To effectively train the model, we developed approaches to mine training data from search log and extract features from **searcher, query, context, and documents**. For fast model iteration, we adopted a **recall metric** on an offline evaluation set to compare models.

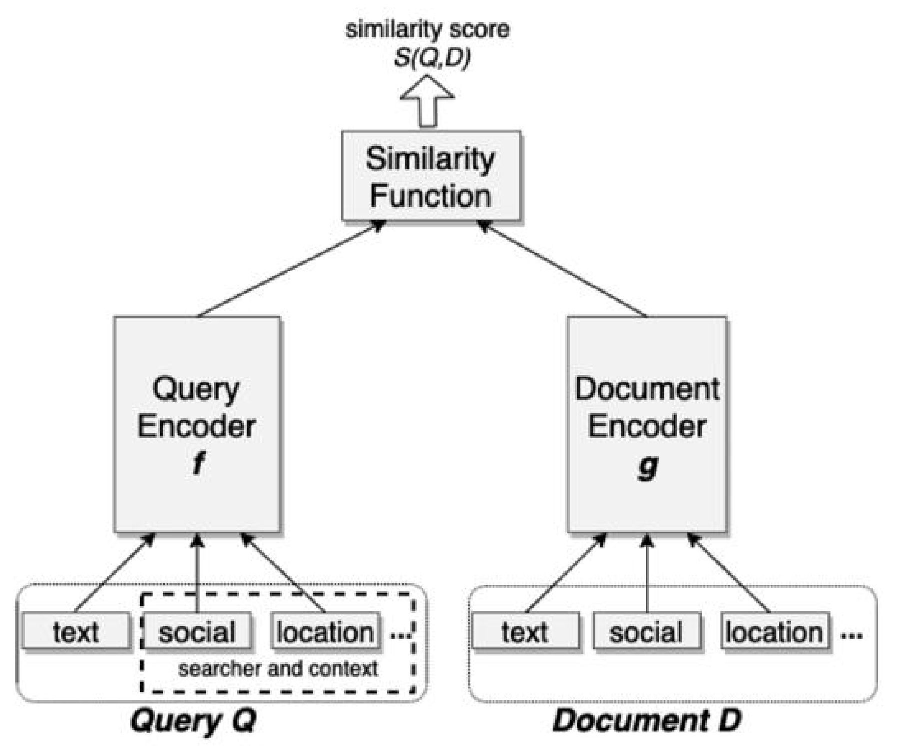


Figure 2: Unified Embedding Model Architecture

SERVING

we need to develop ways to effectively and efficiently serve the model in the retrieval stack. While it is straightforward to build a system combining the candidates from existing retrieval and embedding KNN, we found it is suboptimal because of several reasons: 1) it has huge performance cost from our initial experiment; 2) there is high maintenance cost because of dual index; 3) the two candidate sets might have significant overlap which makes it inefficient overall. Thereafter, we developed **a hybrid retrieval framework** to integrate embedding KNN and Boolean matching together to score documents for retrieval.

OPTIMIZATION

Search is a multi-stage ranking system where retrieval is the first stage, followed by various stages of ranking and filtering models. To wholly optimize the system to return those new good results and suppress those new bad results in the end, we performed laterstage optimization. In particular, we incorporated embeddings into ranking layers and built a training data feedback loop to actively learn to identify those good and bad results from embedding-based retrieval

EBR SYSTEM

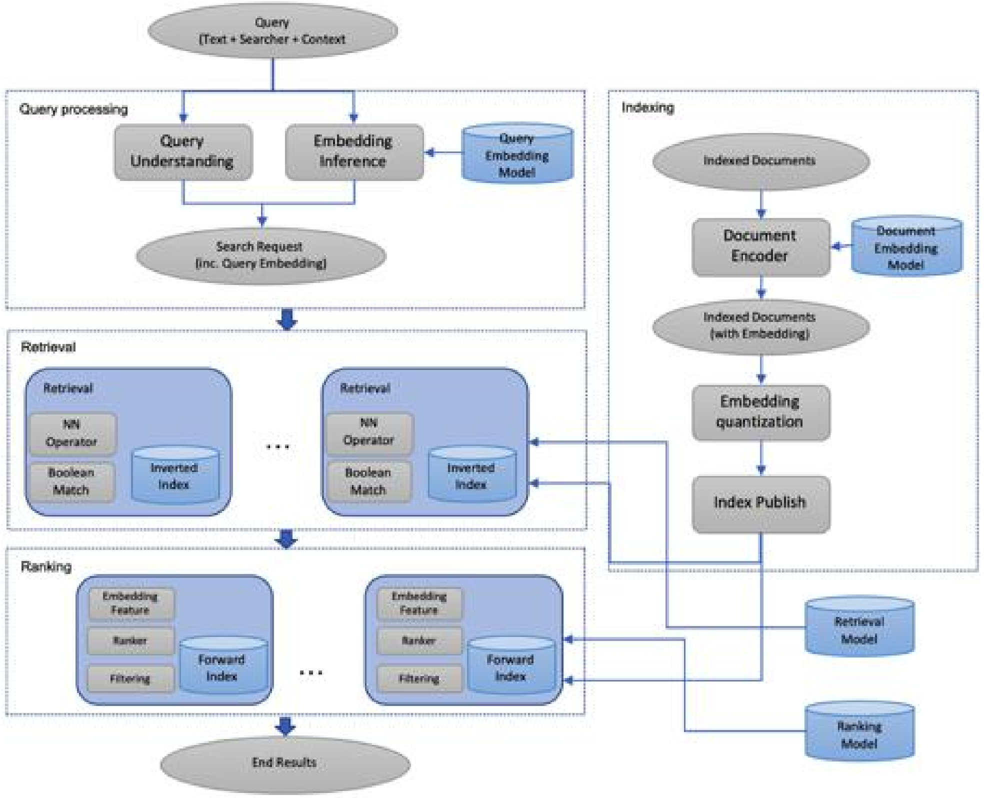
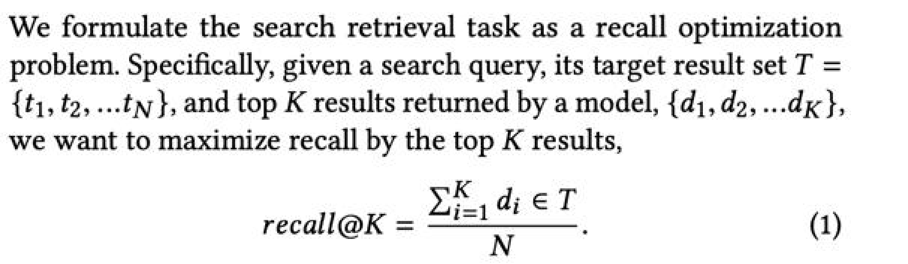


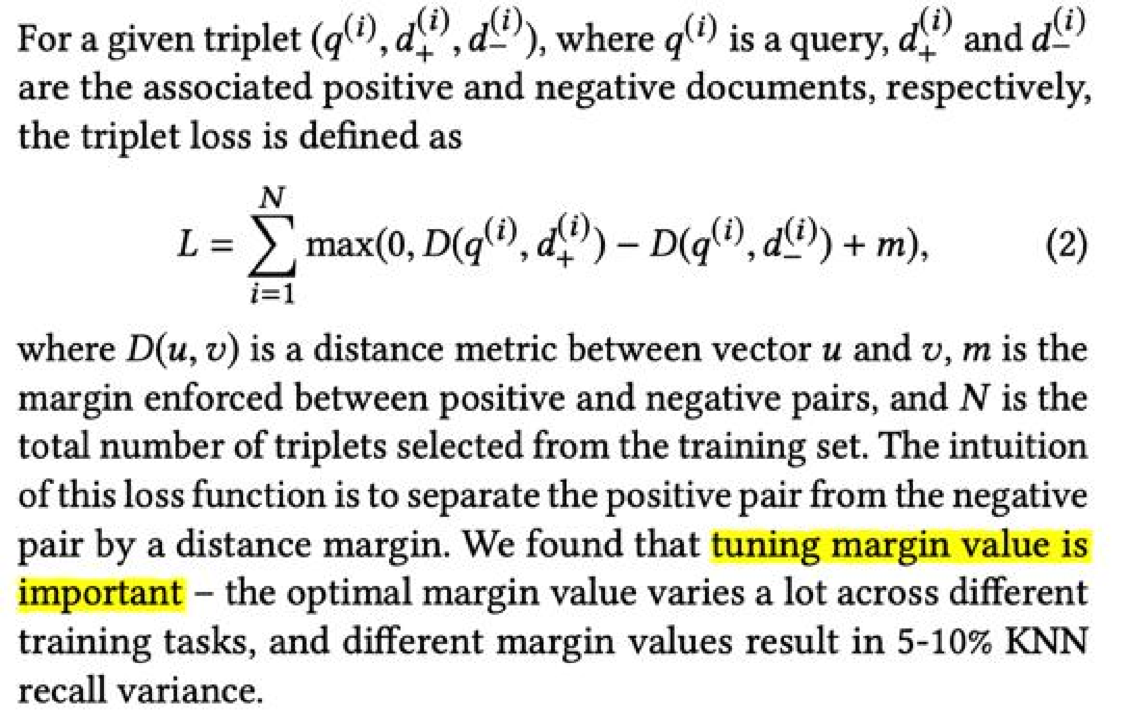
Figure 1: Embedding Based Retrieval System Overview

MODEL



We formulate the search retrieval task as a recall optimization problem. Specifically, given a search query, its target result set $T = \{t_1, t_2, \ldots t_N\}$, and top K results returned by a model, $\{d_1, d_2, \ldots d_K\}$, we want to maximize recall by the top K results,

$$recall@K = \frac{\sum_{i=1}^{K} d_i \in T}{N}. \qquad (1)$$



For a given triplet $(q^{(i)}, d_+^{(i)}, d_-^{(i)})$, where $q^{(i)}$ is a query, $d_+^{(i)}$ and $d_-^{(i)}$ are the associated positive and negative documents, respectively, the triplet loss is defined as

$$L = \sum_{i=1}^{N} \max(0, D(q^{(i)}, d_+^{(i)}) - D(q^{(i)}, d_-^{(i)}) + m), \qquad (2)$$

where $D(u, v)$ is a distance metric between vector u and v, m is the margin enforced between positive and negative pairs, and N is the total number of triplets selected from the training set. The intuition of this loss function is to separate the positive pair from the negative pair by a distance margin. We found that tuning margin value is important – the optimal margin value varies a lot across different training tasks, and different margin values result in 5-10% KNN recall variance.

- *easy triplets*：可以使*loss*为*0*的三元组，即容易分辨的三元组。
- *hard triplets*：*D(q,n)<D(q,p)*，即一定会误识别的三元组。
- *semi-hard triplets*：*D(q,p)<D(q,n)<D(q,p)+margin*，即处在模糊区域的三元组

UNIFIED EMBEDDING MODEL

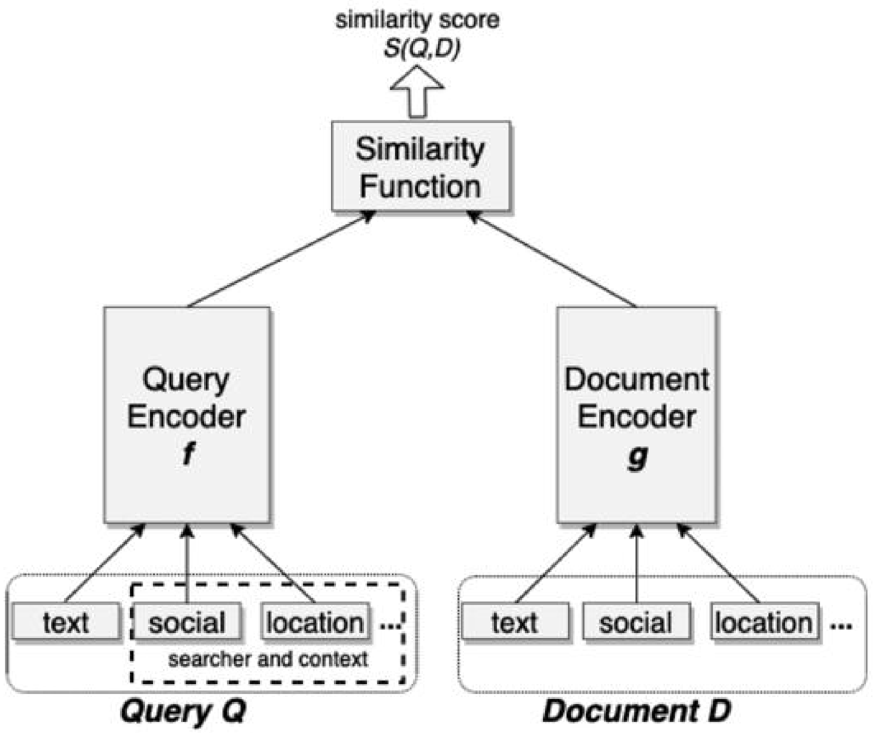
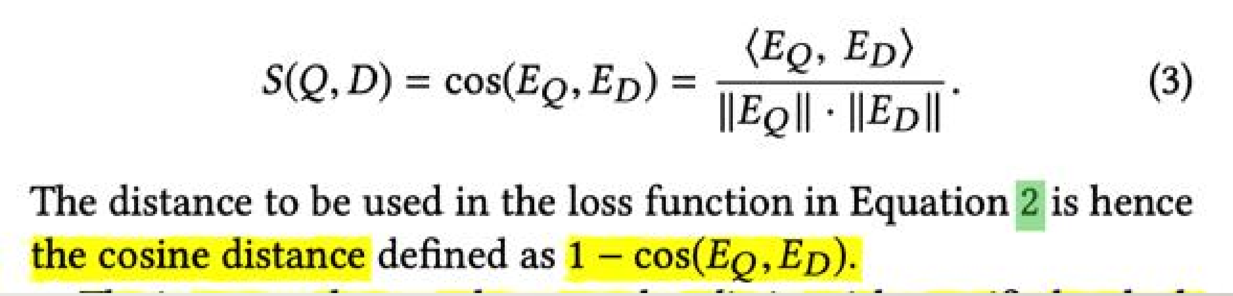


Figure 2: Unified Embedding Model Architecture



$$S(Q,D) = \cos(E_Q, E_D) = \frac{\langle E_Q,\ E_D \rangle}{\|E_Q\| \cdot \|E_D\|}. \tag{3}$$

The distance to be used in the loss function in Equation 2 is hence the cosine distance defined as $1 - \cos(E_Q, E_D)$.

FEATURE ENGINEERING

- **Text features**

character n-grams vs word n-grams

使用char级别的n-grams，两方面的好处：
1.词表小，容易学习。
2.不容易OOV(out-of-vocabulary)。

paper提出，使用char级别的n-grams效果比word级别的n-grams要好。此外，**用了char级别的n-grams后，再加上word级别的n-grams效果还有一定的提升。**

使用文本的embdeding匹配和倒排索引相比（Boolean term matching），有两点好处：1.有改写、纠错、变换和泛化的作用。2.有更好的文本理解能力，可以识别哪些term更重要、哪些term可省略。

Fuzzy text match: learn to match between query "kacis creations" and the Kasie’s creations page
Optionalization: "mini cooper nw", the model can learn to retrieve the expected group Mini cooper owner/drivers club by dropping "nw"

FEATURE ENGINEERING

- **Location features**
searcher city, region, country, and language; explicit group location tagged by the admin

- **Social embedding features**
trained a separate embedding model to embed users and entities(name for people entities or title for non-people entities) based on the social graph

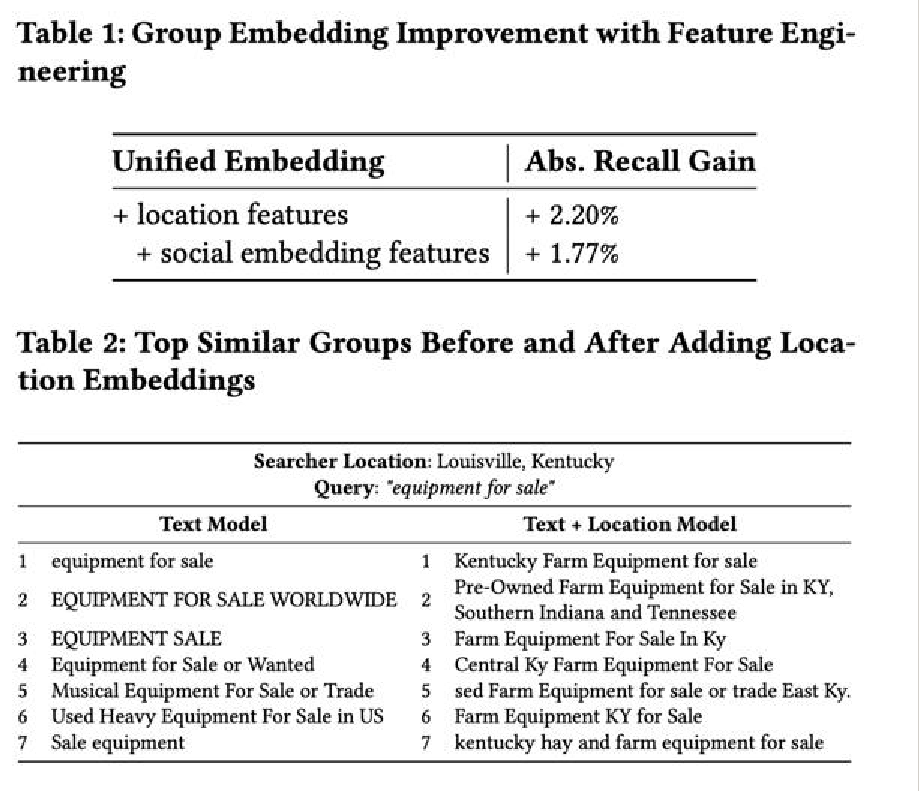


Table 1: Group Embedding Improvement with Feature Engineering

Unified Embedding	Abs. Recall Gain
+ location features	+ 2.20%
+ social embedding features	+ 1.77%

Table 2: Top Similar Groups Before and After Adding Location Embeddings

Searcher Location: Louisville, Kentucky
Query: *"equipment for sale"*

	Text Model		Text + Location Model
1	equipment for sale	1	Kentucky Farm Equipment for sale
2	EQUIPMENT FOR SALE WORLDWIDE	2	Pre-Owned Farm Equipment for Sale in KY, Southern Indiana and Tennessee
3	EQUIPMENT SALE	3	Farm Equipment For Sale In Ky
4	Equipment for Sale or Wanted	4	Central Ky Farm Equipment For Sale
5	Musical Equipment For Sale or Trade	5	sed Farm Equipment for sale or trade East Ky.
6	Used Heavy Equipment For Sale in US	6	Farm Equipment KY for Sale
7	Sale equipment	7	kentucky hay and farm equipment for sale

TRAINING DATA MINING NEGATIVES

- random samples: for each query, we randomly sample documents from the document pool as negatives.

- non-click impressions: for each query, we randomly sample those impressed but not clicked results in the same session as negatives.

实验结果表明，仅仅用第二种方法，效果非常差。 paper解释展示但用户没点击的样本，其实也是经过搜索系统的层层漏斗（例如很多相关性模块）筛选出来的才能展示到用户面前，这样的样本其实多多少少和query还是沾边的，或者说比较相关的。而对于召回任务来说，整个候选集其实有大量的和query毫不相关、不沾边的doc。这样子模型学习的样本，和真正要召回时的候选集的分布，就是完全不一样的了。

TRAINING DATA MINING POSITIVE

- clicks: it is intuitive to use clicked results as positives, since clicks indicates users' feedback of the result being a likely match to users' search intent.

- impressions: the idea is that we treat retrieval as an approximation to ranker but can execute fast. Thereafter, we want to design the retrieval model to learn to return the same set of results that will be ranked high by the ranker. In this sense, all results shown or impressed to the users are equally positive for retrieval model learning.

实验结果表明，两种方法效果基本一致。用了点击的样本做正样本后(或正样本增强)，再增加展示的样本，也没有额外的收益。

LATER-STAGE OPTIMIZATION

Embedding as ranking feature

目前的搜索系统是根据目前的倒排索引优化，可能很多向量检索回来的结果，整个搜索系统都没见过。所以需要对系统，根据向量检索做一些调整，以便能更好的将向量检索的结果应用到搜索系统中：

将emb应用上层的排序系统的ranking feature，这样有两个好处：1.让上层的排序系统，更好的识别ANN检索回来的结果（例如ANN-score特征权重比较大）。 2.向量检索的embedding提供了很好的语义信号。在多种尝试中（cosine、Hadamard product、裸的embedding），用cosine的效果最好

Training data feedback loop

向量检索的结果recall高，但和倒排索引相比**precision可能差（即排序能力可能差）**。为了提高向量检索的precision，可以记录一些融合向量检索后的排序结果，送给人工标注是否相关，然后将人工标注结果加入向量检索模型的训练样本中，重训模型。这种方法一般可以同时提高precision和recall。

ADVANCED TOPICS
HARD MINING

Hard negative mining (HNM): When analyzing our embedding model for people search, we found that the top K results from embeddings given a query were usually with the same name, and the model did not always rank the target results higher than others even though the social features are present.

- Online hard negative mining;
- Offline hard negative mining.

Hard positive mining: To maximize the complementary gain by embedding based retrieval, one direction is to identify new results that have not been retrieved successfully by the production yet but positive. To this aim, we mined potential target results for failed search sessions from searchers' activity log.

ADVANCED TOPICS HARD MINING

Online hard negative mining;

online部分，训练样本输入的是正样本的<query, doc>对，负样本就是batch内其他query的doc，然后用这batch-size - 1个负样本里最难的负样本作为最终的负样本集合，来计算模型的loss和更新参数，其他负样本忽略。这个技术其实叫hardest negative，**本文指出，这是提升模型效果非常重要的一个策略。**paper里指出，最多选最难的2个负样本，多选了会损失模型效果。但是这种方法始终还是有局限，可能还是产生不了任意难度的负样本（显存的原因，batch-size也不能无限开大）。

ADVANCED TOPICS
HARD MINING

Offline hard negative mining

1.对于每个query，得到top-K最匹配的doc
2.根据难的负样本选择策略选择难的负样本
3.用难的负样本重训模型

该paper指出，仅仅用难的负例，效果没办法超越online随机负例。在offline的hard negative mining中发现，**用最难的负样本效果不是最好的。而是用排在101-500的是最好的**。第二个策略是融合一些简单的样本还是有必要的，**随机的容易负样本：难的负样本比例大概是100:1效果最好(是随机的容易的负样本更多)**。

EMBEDDING ENSEMBLE

Weighted Concatenatin

不同难度的模型独立打分，最终取Top K的分数依据是多模型打分的加权和（各模型的权重是超参，需要手工调整）

$$S_w(Q, D) = \sum_{i=1}^{n} \alpha_i \cos(V_{Q,i},\ U_{D,i}),$$

$$E_Q = (\alpha_1 \frac{V_{Q,1}}{\|V_{Q,1}\|}, \cdots, \alpha_n \frac{V_{Q,n}}{\|V_{Q,n}\|}),$$

$$E_D = (\frac{U_{D,1}}{\|U_{D,1}\|}, \cdots, \frac{U_{Q,n}}{\|U_{Q,n}\|}).$$

$$\Rightarrow$$

$$\cos(E_Q,\ E_D) = \frac{S_w(Q, D)}{\sqrt{\sum_{i=1}^{n} \alpha_i^2} \cdot \sqrt{n}}.$$

- 文章中指出，使用“曝光未点击”作为*hard negative*训练出来的*hard model*，离线指标好，但是线上没有效果
- 反而，使用挖掘出来的*hard negative*训练出来的*hard model*，与*easy model*融合的效果最好

Cascade Model

候选物料先经过easy model的初筛，再经过hard model的二次筛选，剩余结果再交给下游，更复杂的粗排或是精排。

根据文章中的经验，使用“曝光未点击”作hard negative训练出来的hard model同样没有效果，反而是挖掘出来的hard negative训练出来的hard model做二次筛选更加有效

QA?