心法利器[7] | 漫谈语义相似度与语义向量表征

原创 机智的叉烧 CS的陋室 2020-11-08

收录于话题
#自然语言处理 14 #心法利器 15

猪突猛進

百石元 - 冴えない彼女の育てかた 第2卷 特典CD

【前沿重器】

全新栏目，本栏目主要和大家一起讨论近期自己学习的心得和体会，与大家一起成长。具体介绍：仓颉专项：飞机大炮我都会，利器心法我还有。

往期回顾

- 心法利器[2] | 统计语言模型使用反思
- 心法利器[3] | tf.keras自学笔记
- 心法利器[4] | tf.keras文本分类小例子
- 心法利器[5] | 聊自己非计算机专业做程序员的经验
- 心法利器[6] | python grpc实践

除了我之前讲的命名实体识别和文本分类，语义相似度应该是自然语言理解（NLU）里面又一大核心拼图，无论是机器翻译、搜索、对话等，都有很大的应用空间，之前其实或多或少也提到过，包括对一篇SIGIR论文的讲解：R&S[18] | SIGIR2018：深度学习匹配在搜索与推荐中的应用。这次给大家掰开揉碎展开谈谈这块吧。

语义相似度任务概述

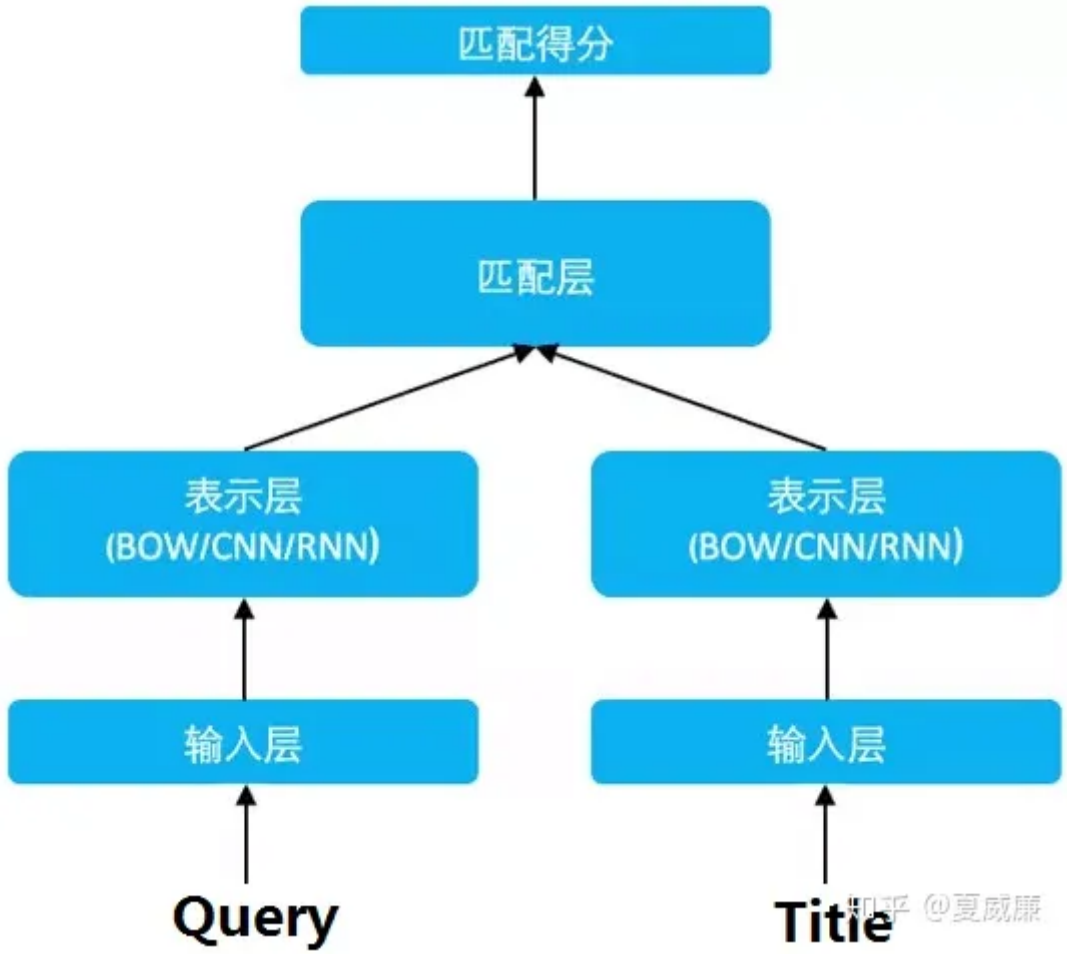
语义相似度，顾名思义，主要是为了衡量两个句子之间的相似度，来自天池新冠疫情相似句判定大赛的例子：

- 相似句：肺部发炎是什么原因引起的-肺部发炎是什么引起的
- 不相似句：肺部发炎是什么原因引起的-肺部炎症有什么症状

一般都会有非常明确的案例告诉我们，什么叫做相似，什么叫做不相似，这个有非常明显的场景愿意，还是上面那句话，在判断query意图上，如果是判断大粒度意图的话（是否是医疗问句）那就是相似句了，如果是小粒度（症状意图、病因意图）那这两句就不相似。

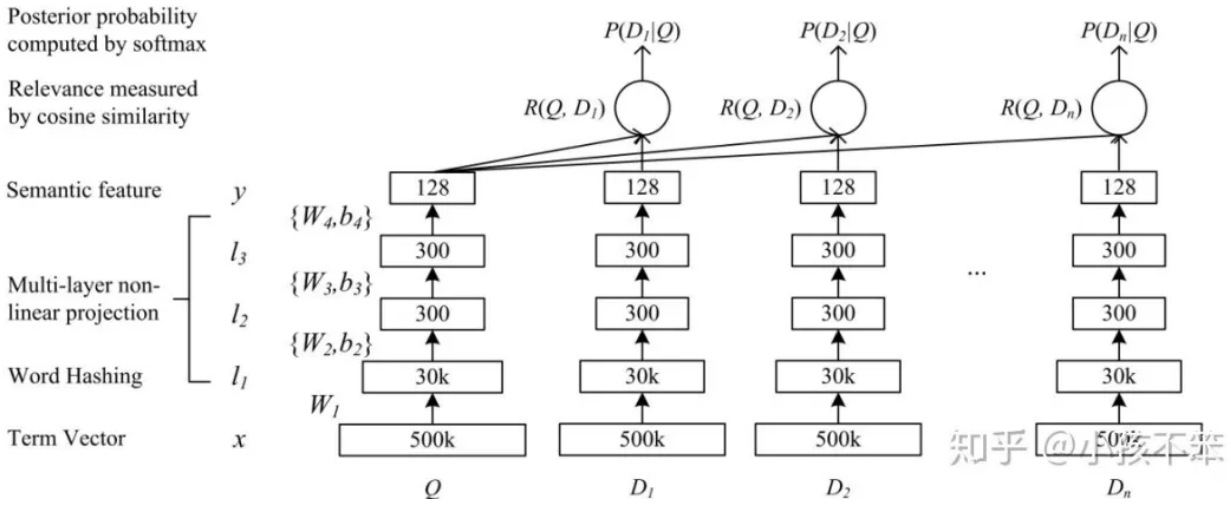
常规的语义相似度架构

语义相似度架构其实非常简单，简单地说就是两个句子进去，进行特征抽取，简单到BOW、TF-IDF，复杂的就是bert（跳过word2vector、elmo？？？），然后进行相关性匹配计算，这块也花样繁多，简单的就是余弦相似、欧氏距离等，复杂的可以矩阵相乘等，最终到达标签，完成一次语义相似度评估计算。OK，上图：



孪生网络

孪生网络本身是CV领域的东西，后来被NLP领域拿来直接用了，生根发芽，发扬光大，最经典的使用要数DSSM了。其实孪生网络的核心思想还是藏在训练方式上，一般的语义相似度我们都是用比较简单二分类来训练了，说白了就是预测出一个相似度概率和实际标签算交叉熵，而孪生网络，顾名思义，就是要考虑两个网络，具体考虑的思路，就非常像LTR里面的pair wise，但是更为暴力一些，pairwise是两者PK，而在DSSM里，会给到一个正样本（与原句相似）和N个负样本（与原句不相似）来共同构造成一个样本，具体的图就长这样：



这种方式的核心在于，以一个句子为核心来进行对比，能精准告诉模型“什么是对的，什么是错的”，边界清楚能让模型分的更好（SVM的思想），举个例子，我们都再说“人工智能”，我们不知道的一切都可以被戴上这个帽子，但是边界非常模糊，这让人容易被忽悠，这就是因为我们不知道什么不是“人工智能”，模型也是一样，给到清晰的边界，模型能分得更好，这个是“主动学习”的一大初衷。

表征的优化

DSSM终究试一次简单的尝试，但以这个为出发点，大家又围绕着DSSM做了很多工作，其中一大重点就是把DSSM这种base line级别的操作（文章用的是word hashing）升级为一些效果更好的操作：

- CNN-DSSM：也被叫做CLSM，把全连接层换成了CNN。
- LSTM-DSSM：把全连接层换成了LSTM，严谨的应该说是peephole LSTM。
- 比较新潮的Attention操作也偶有见到。self-attention和co-attention都有，attention能够一定程度的充当“关键词抽取”的作用，因此在特定场景（例如信息冗余较多的对话场景）效果提升有些明显。

有关attention的使用，推荐一篇论文Attention-Based Convolutional Neural Network for Modeling Sentence Pairs，也就是所谓的ABCNN，感觉把attention在这块的使用讲的比较好：

- BCNN（basic cnn）其实就是用cnn作为特征提取的关键步骤，只不过这里的卷积参数是双塔共享的。
- ABCNN-1针对两个句子的原始表征（没有进行CNN时），构造了attenton矩阵作为卷积输入的另一个通道。
- ABCNN-2在1的基础上，把attention的步骤放在了池化之前，该attention用来对卷积的结果进行reweight，然后再做池化。
- ABCNN-3是把上述两者进行了结合。

换积木这种事情姑且只能算小打小闹，我们的目标不止于此，而是会——换大积木，BERT，它具有更为强力的BERT能力，效果自然也越好，sentence-bert就是对bert在语义相似度上的一个重要的使用。

向量化表征

语义相似度能让我们衡量两个句子的语义相似度，但是实际上我们的应用场景更多是“在句子库里面找到最接近给定句子的句子”，例如一个句子“明天天气”，我要找到整个库里面最接近的那个句子，例如“明天的天气”、“明天下雨吗”、“天气之子”等，如果简单的一个一个匹配去找，复杂度就是O(N)，这个数字无疑非常可怕，当库是千万级别大小的时候，这个搜索就变得非常低效，因此我们是需要特定的手段来解决这个问题的，这套解决方案就是——最近邻召回检索，这里依赖两个技术点，一个是基于语义相似度的句子向量化表征，另一个是最近邻向量索引。

最近邻向量索引

不知道大家对KNN有没有了解，在统计学习方法中，第一个讲的方法就是他，说白了就是找最接近给定样本的N个训练样本，他们大部分属于什么什么类，那这个给定样本就是什么类，而问题在于怎么找到最接近的N个样本，这实际上就是一个通过向量找向量的工作，在《统计学习方法》里面提出的是kd tree，而现在比较流行的主要是两个——同样是基于树的annoy和基于图的hnsw，经过一些评测（https://zhuanlan.zhihu.com/p/152522906），这两个方案都有比较领先的性能，网络上有比较明确的原理和实现方案，此处就不赘述啦~

向量化表征

向量化表征对语义相似度有了更高的要求，它不仅仅要求我们要算出两者的相似度，还要求在计算过程中，需要把句子表征都需要降维到向量级别，因为后续下游要通过最近邻向量完成搜索，这个问题其实还是比较简单的，常规的操作不外乎就是2种，平均池化和最大池化，在Sentence-BERT中又提到了一种，所以总共就是3种：

- 平均池化，对表征的矩阵整体进行平均池化，mean/average pooling，得到向量。
- 最大池化，对表征的矩阵整体进行最大池化，max pooling，得到向量。
- 直接用CLS位置的输出向量化作为整个句子的向量。

更多前沿操作

上面应该都是业界比较公认的比较基本的操作和理解，在此基础上带大家看一些比较前沿的一些使用方案和方法。

机器之心-百度NLP

针对传统匹配（如BM25）的多义同义词处理、句子结构复杂、匹配非对称等问题，在sim-net（即一一匹配计算相似度）的基础上，做了多粒度切词、高频判断语义引入等先验信息的方式来对提升效果，并在实验上针对正负样本比例、阈值判断等做了很多的尝试（文章具体没说）。

Facebook使用经验

在KDD2020里面facebook分享了自己在文本匹配方面的经验：Embedding-based Retrieval in Facebook Search。

首先在训练上，使用的是比较常见的三元组hinge loss。

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

至于模型，则是和DSSM类似的双塔结构，query和doc分别表征后进行相似度匹配，但是在特征上花费了一些功夫，新增了位置特征、社交特征，后两个都是个性化特征，大家就根据实际情况选用即可。

在训练上，负样本的构建一直是语义相似度的一个老大难问题，根据他们的经验，曝光未点击和数据库随机抽取一起用的效果会更好。

至于服务上，即最近邻召回，facebook则用的是自己开源的faiss引擎，内部搭载的是annoy索引机制。

另一方面，文章再提到了负样本构建，即Hard negative mining问题，一般是通过在线曝光未点击、离线相似度阈值范围抽取的方式挖掘。

文章还提到一个很有意思的提升方案，就是模型融合，简单地说就是模型预测的向量分别进行归一化后进行concat，然后进行检索。

蚂蚁金服语义相似度竞赛的经验

- 纠错和避免纠错问题。
- 特征工程的尝试，除了语义向量外，还可以加入长度、编辑距离、n-gram相似度、词汇统计特征、关键词、疑问词相似度、主题意图相似等。
- co-attention交互特征的使用。

小结

小结几个关键点吧

- 分别表征后进行交互仍然是主流，但是由于向量召回的存在，欧氏距离余弦距离仍然是主流，所以模型的优化核心就放在了如何更好地表征句子上。
- 训练方式基本就固定在了DSSM或者是孪生网络类似的联合训练上。

- 负样本的构造，HNM问题是提升效果的一个关键因素。
- 业界的尝试没有想象中的复杂，而是在业务和实际的操作中寻找经验和结论，这很大一部分原因和向量召回这一需求的约束有关。

我是叉烧，欢迎关注我！

叉烧，OPPO搜索算法工程师，主做Query理解，NLP方向。19届北科技统计学硕士（保研），17届北京科技大学信息与计算科学、金融工程双学位毕业，论文7篇，学生一作3篇，参与国家级及以上学术会议4次，优秀论文一次，国奖金。曾任去哪儿网大住宿事业部产品数据，美团点评出行事业部算法工程师。

微信个人公众号
CS的陋室

微信 zgr950123
邮箱 chashaozgr@163.com
知乎 机智的叉烧