KRED: 新闻推荐算法

一、实体表示层

新闻文章中的实体可以对应到知识图谱中的实体（注意这里只用到了实体，没有关系）。

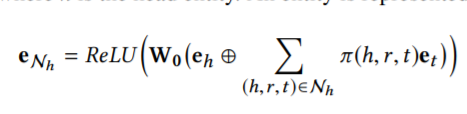
知识图谱：有向图，(h,r,t)集合

使用TransE学习嵌入实体和关系向量e（即TransE中实体矩阵E和关系矩阵R），将学习到的结果作为特性引入。

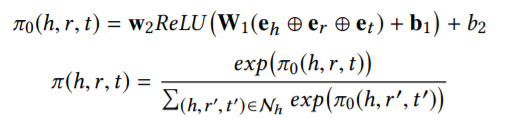
再考虑到实体邻居的价值，我们提供采用KGAT的思想进行实体表示

KGAT：知识图注意网络

因此，找到和h相关的所有三元组，则h的嵌入表示为：



“+”是表示拼接，π是控制从邻居学到多少比例的嵌入，通过两层神经网络得到（relu+sigmoid）。



可以学习更多hop的邻居，但是性价比很低，不考虑。

二、背景嵌入层

考虑信息的背景信息，分三部分增加嵌入：

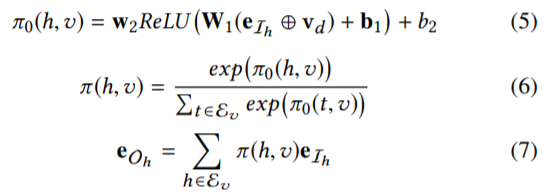
1. 是否出现在标题（1or2）

2. 频率（上限20）

3. 类别（eg公司/食物/…）

三、信息蒸馏层

借鉴Transformer（没有自注意力或多头注意力，性价比低，可能是因为没有NLU复杂）





生成的向量即为KDV。

多任务（硬共享）：

多任务效果好的原因：考虑新闻分类问题，两个user2item可以辅助判断item2item（类似扩大数据集）。

User2item推荐：输入向量包括用户u和文档v，预测分数g(u+v)，+表示连接，g表示神经网络，user向量生成（历史点击文档进入一层神经网络生成注意力分数，将该分数作为权重合并到文档中，生成用户向量）。

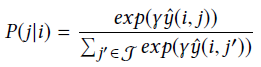
Item2item推荐：一层神经网络，128维，tanh激活函数，cosine相似度。

分类问题、新闻局部性问题、流行度预测问题：一层神经网络

Loss函数：

Item2user、user2user（参考TransE的算法过程）：

找5个反例，最大化正样例在六个样例中的概率（）：



最大似然求loss。

其它loss函数：

多分类问题：交叉熵

注：为了避免引入新的超参数来组合不同任务的损失函数，我们采用两阶段的方法进行多任务训练。在第一阶段，我们每隔几个小批量交替训练不同的任务。在第二阶段，我们只包含目标任务的数据，以最终确定特定于任务的模型。

数据集：

MSN新闻（2019-1-15~2019-1-28）

少于5个的删掉

总量：665,034 users, 24,542 news articles, 1,590,092 interactions

文章平均字数701

一份文件包含24个实体，在知识图谱中的1-hop覆盖了3314628个实体，1006个关系和71727874个triples。

Base1：bert lda+dssm naml （手工加入知识信息）

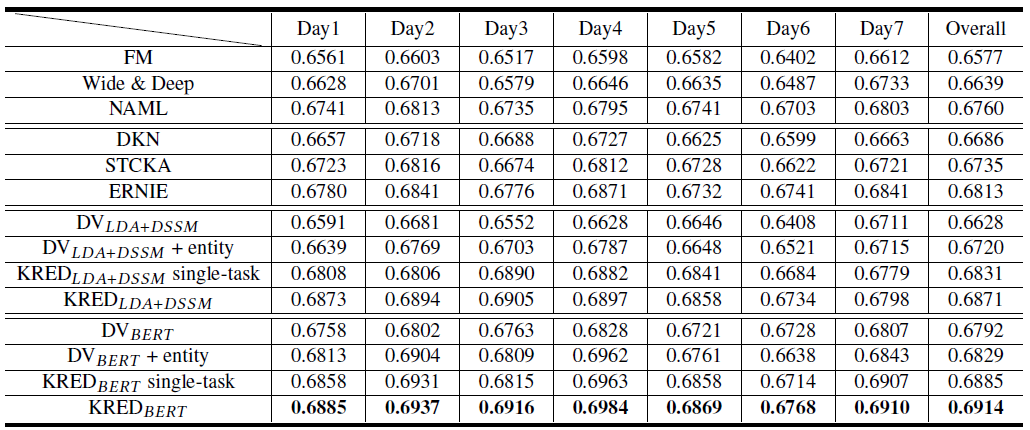
Base2：dkn stcka ernie （已经加入了知识信息）

Base2-2：LDA+dssm配置的KRED BERT配置的KRED

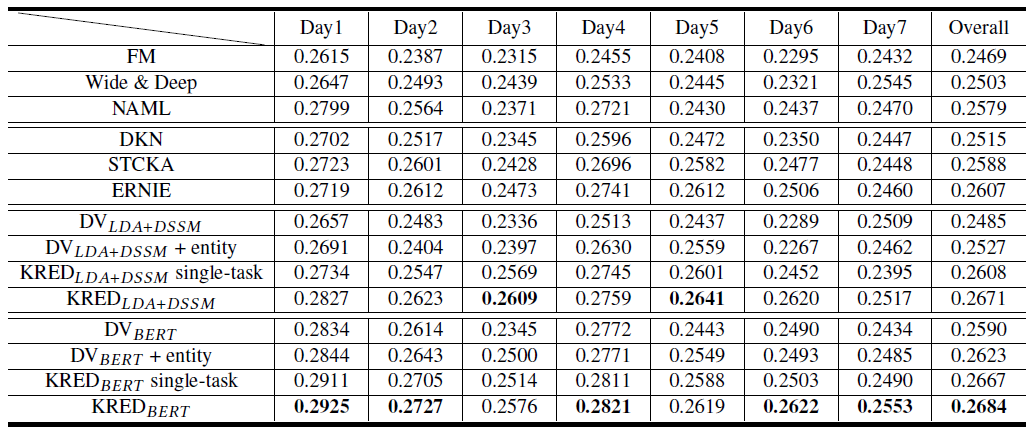
个性化推荐base：FM Wide+Deep

评价指标：

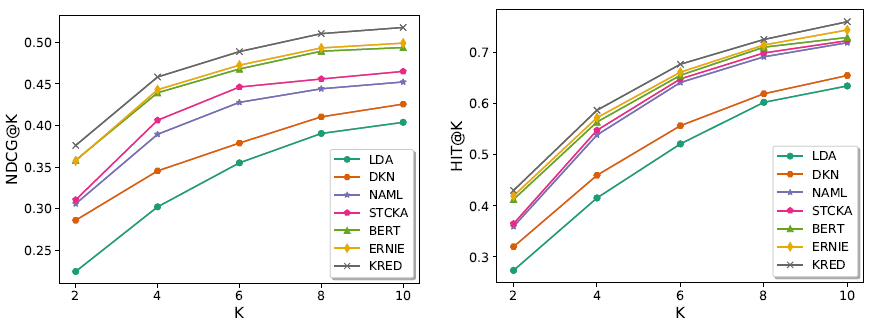
AUC(acc+macro-f1)



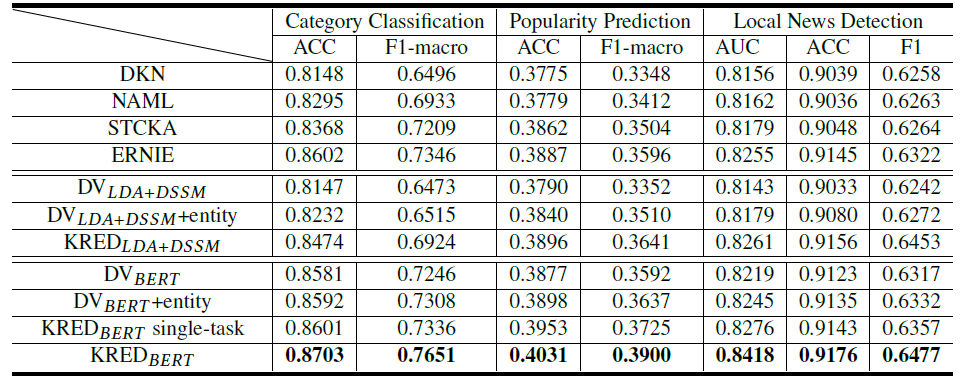
NDCG



个性化推荐（左NDCG 右HR）

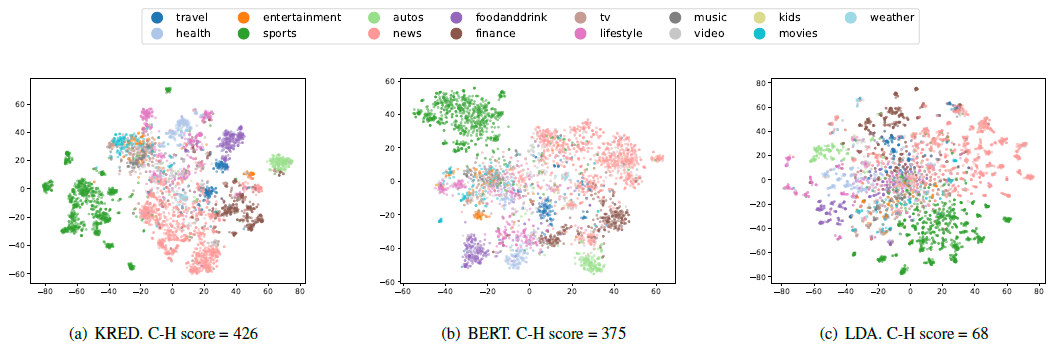


新闻分类和预测



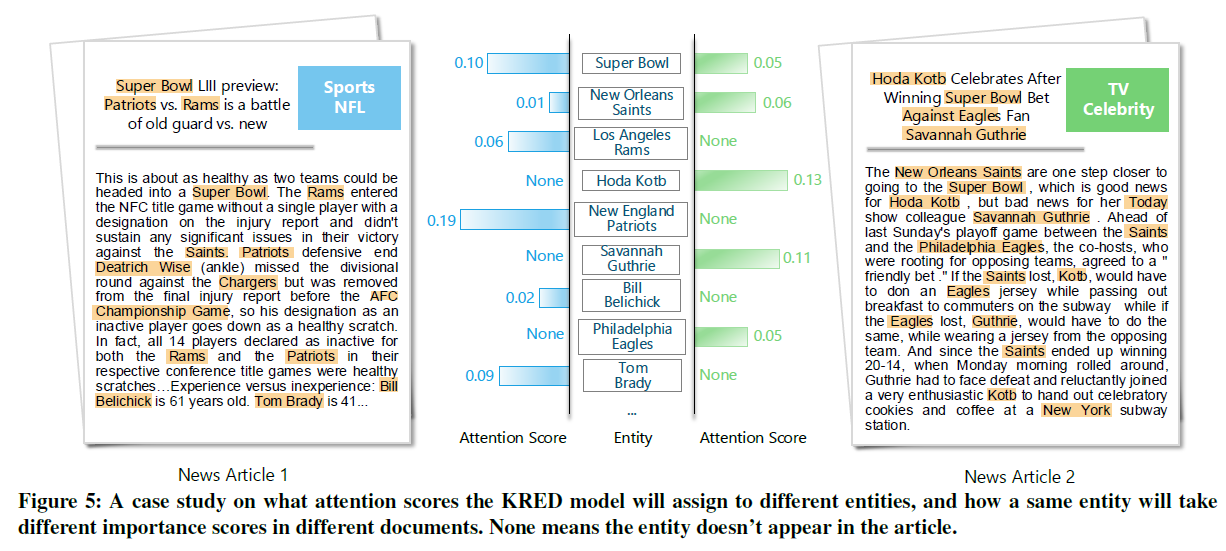
模型有效性（分布更好）

t-SNE algorithm



我们可以观察到，具有相同类别的文档倾向于聚集在一起，并且KRED模型产生的嵌入呈现出比BERT和LDA模型更好的分布模式。

Attention分数：

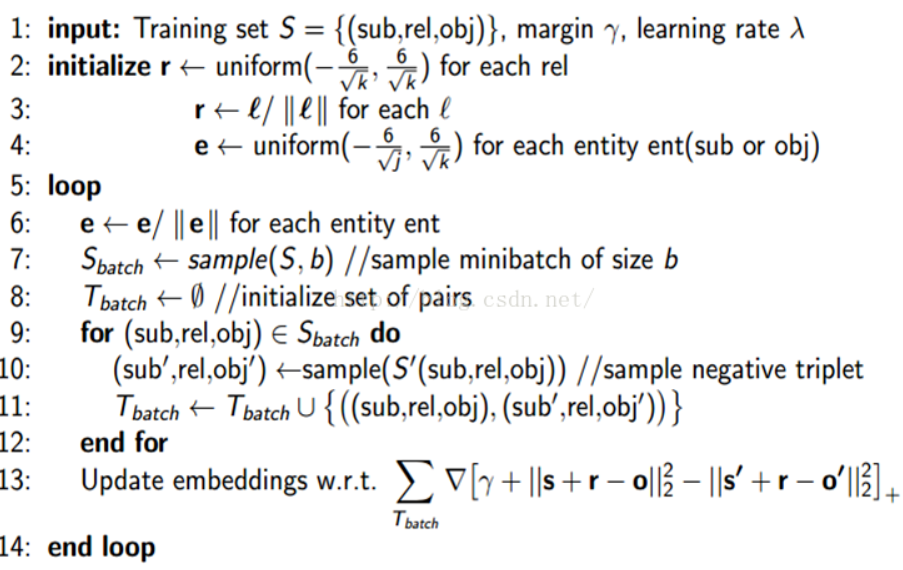


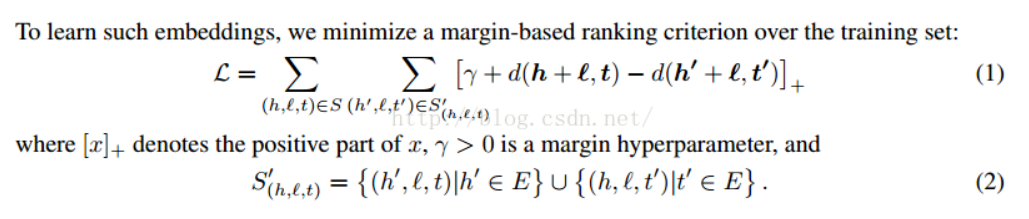
较高的Calinski-Harabasz得分与具有更好定义的聚类的模型有关，KRED、BERT和LDA模型的得分分别为426、375和68。

TransE: 2013-nips

TransE的直观含义，就是TransE基于实体和关系的分布式向量表示，将每个三元组实例（head，relation，tail）中的关系relation看做从实体head到实体tail的相加，通过不断调整h、r和t（head、relation和tail的向量），使（h + r） 尽可能与 t 相等，即 h + r = t，即找到实体矩阵E和关系矩阵R。

算法：





梯度下降采用SGD。

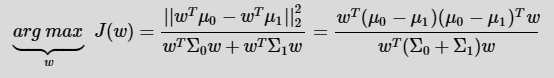
直观上，我们要前面的项（原三元组）变小（positive），后面的项（打碎的三元组）变大（negative）。就跟喂小狗一样，它做对了，就给骨头吃；做错了，就打两下。前面的项是对的（来自于训练集），后面的项是错的（我们随机生成的）。不同时打碎主体和客体，随机挑选一个打碎，另一个保持不变，这样才能够有对照性。

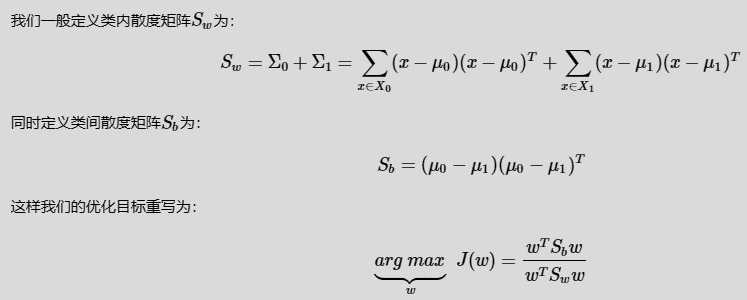
Code: <https://codechina.csdn.net/mirrors/wuxiyu/transe/-/blob/master/tranE.py>

（没有自注意力或多头注意力，性价比低，可能是因为没有NLU复杂）

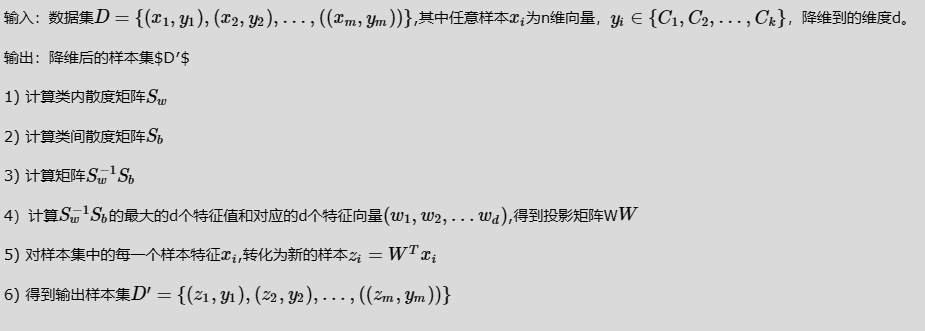
LDA(Special Case有监督的2分类)

由于是两类数据，因此我们只需要将数据投影到一条直线上即可。假设我们的投影直线是向量w,则对任意一个样本本,它在直线w的投影为,对于我们的两个类别的中心点,,在在直线w的投影为和。由于LDA需要让不同类别的数据的类别中心之间的距离尽可能的大，也就是我们要最大化,同时我们希望同一种类别数据的投影点尽可能的接近，也就是要同类样本投影点的协方差和尽可能的小。

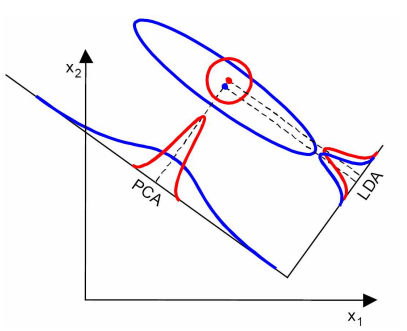
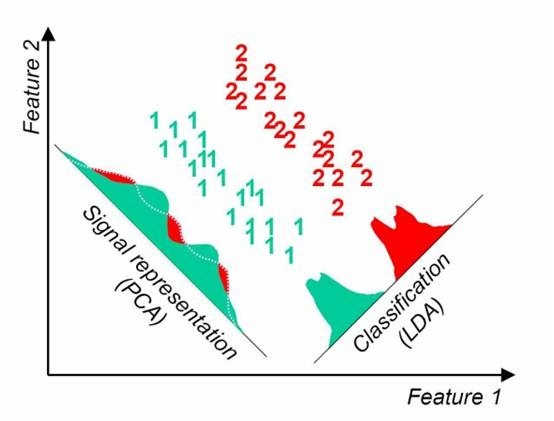




LDA算法：



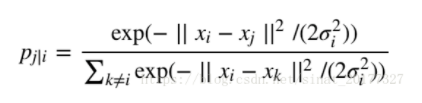
LDAvsPCA



SNE:

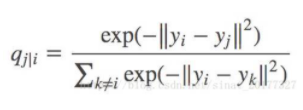
基本思想:若两个数据在高维空间中是相似的，那么降维至 2 维空间时它们应该离得很近。

高维空间中，j落在i的高斯分布下的邻域内的概率计算（越近，概率越大）：

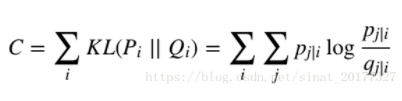


特别的，。

二维空间中（取），j落在i的高斯分布下的邻域内的概率计算（越近，概率越大）：



概率分布差异使用KL散度衡量



保证了局部性，即实际p较大，预测p（即q）较小，则loss很高

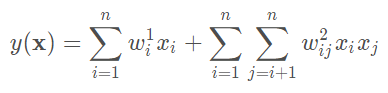
但是不保证全局性（这是高维到低维的固有损失）

RS：MF NCF FM=ID+MF GCN

知识RS：RippleNet

FM：

Rendle, Steffen. “Factorization Machines.” IEEE International Conference on Data Mining IEEE Computer Society, 2010:995-1000



核心思想：用MF求解w。