

DSSM

Learning Deep Structured Semantic Models for
Web Search using Clickthrough Data

Po-Sen Huang, Xiaodong He

CIKM, 2013

Outline

□ 研究背景

□ 模型

- Word Hashing
- 框架

□ 对比实验

- 评价标准
- 对比模型
- 实验结果

□ 评价

- 本文自身优点
- 不足及改进模型

DSSM-研究背景

- 面向信息检索领域：文档和查询项的相似度计算
- Lexical matching: TF-IDF/BM25
 - 语义鸿沟
 - 向量表示高维稀疏
- Latent semantic models: LSA/PLSA/LDA
 - 无监督学习，独立于文档和查询项的评分机制
- Bi-Lingual Topic Models (BLTMs) 和DPM (Discriminative Projection Models)：利用Clickthrough data的监督学习模型，但前者是利用EM算法的次优模型，后者涉及矩阵运算，计算耗时

DSSM-word Hashing

- 为解决词表高维的问题，降低单词表示维度
- N-gram: 将文本内容按照字节顺序进行大小为N的滑动窗口操作，最终形成长度为N的字节片段序列。n-gram中的gram根据粒度不同，有不同的含义，可以是字粒度，也可以是词粒度的。
- Bigram:
单词“apple”，字符粒度下，n的取值为2
它的bigram有：“#a”，“ap”，“pp”，“pl”，“le”，“e#”
- Trigram:
单词“apple”，字符粒度下，n的取值为3
它的trigram有：“#ap”，“app”，“ppl”，“ple”，“le#”

DSSM-word Hashing

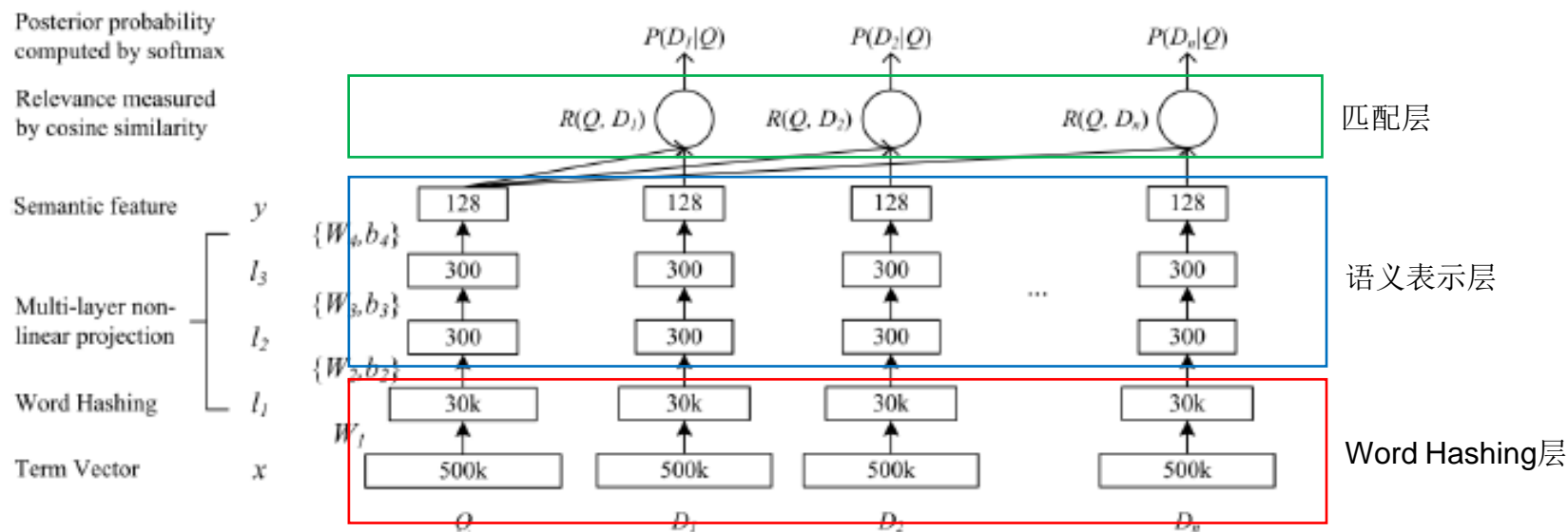
- 解决单词高维稀疏的问题
- 解决OOV的问题 (out-of-vocabulary)
- 符合单词形态学的特点
- 一种哈希算法，可能出现碰撞：500K个word可以降到30k维，冲突的概率为0.0044%

Word Size	Letter-Bigram		Letter-Trigram	
	Token Size	Collision	Token Size	Collision
40k	1107	18	10306	2
500k	1607	1192	30621	22

Table 1: Word hashing token size and collision numbers as a function of the vocabulary size and the type of letter ngrams.

DSSM-模型

- 通过搜索引擎中 Query 和 Title 的点击数据，用 DNN 将其表达为低维语义向量，并通过 cosine 距离来计算两个语义向量的距离，最终训练出语义相似度模型。
- 该模型既可以用来预测两个句子的语义相似度，又可以获得某句子的低维语义向量表达。
- 加了一层word Hashing，再用DNN训练



DSSM-模型

模型框架

Posterior probability
computed by softmax

Relevance measured
by cosine similarity

Semantic feature

y

Multi-layer non-
linear projection

l_3

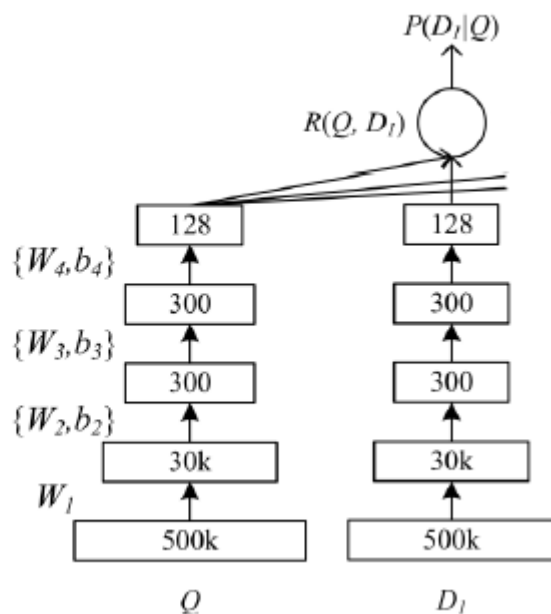
l_2

Word Hashing

l_1

Term Vector

x



$$R(Q, D) = \cosine(y_Q, y_D)$$

$$y = f(W_N l_{N-1} + b_N)$$

$$l_i = f(W_i l_{i-1} + b_i), i = 2, \dots, N - 1$$

$$f(x) = \tanh(x)$$

$$l_1 = W_1 x$$

Input: x

$$P(D|Q) = \frac{\exp(\gamma R(Q, D))}{\sum_{D' \in D} \exp(\gamma R(Q, D'))}$$

softmax 函数可以把**Query**与**Doc**的语义相似性转化为一个后验概率

$$Loss = -\log \prod_{(Q, D^+)} P(D^+|Q), D^+ \text{ 表示被点击的文档, 最大化被点击文档相关性的最大似然}$$

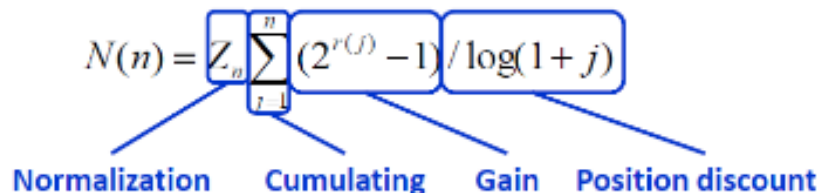
DSSM-实验准备

□ 数据准备:

查询项和文档title: 搜集16510英文query, 每个query对应15个title人工标注, 每个query-title打分是[0,4]

□ 评估方法: NDCG(Normalized Discounted Cumulative Gain, 归一化折损累积增益)

- 用来评价信息检索排名的好坏。越相关的文档排名越靠前
- $2^{r(j)} - 1$: 检索结果排在第j位置上的增益得分, $r(j)$ 相关度0~4
- $\log(1 + j)$: 考虑排序结果位置因素
- Z_n : 归一化因子, 是理想情况下检索结果排序累积增益结果
- $NDCG@k$

$$N(n) = Z_n \sum_{j=1}^n (2^{r(j)} - 1) / \log(1 + j)$$


Normalization Cumulating Gain Position discount

DSSM-对比模型

- 无监督模型：
 - ✓ TF-IDF/BM25: Lexical matching
 - ✓ WTM [word translation model]: 学习查询项单词和文档单词之间的映射关系
 - ✓ LSA/PLSA: 利用文档语料进行训练，将文档和单词映射到同一语义空间
 - ✓ DAE [deep auto-encoder]: 4层隐藏层（300Nodes），中间层（128Nodes），仅利用文档语料训练
- 监督模型：利用**Clickthrough data**
 - ✓ BLTM-PR: 双语主题模型，利用EM算法使查询项和文档属于同一隐含主题
 - ✓ DPM: 使用S2Net算法学习查询项和文档之间的映射矩阵

DSSM-实验结果

#	Models	NDCG@1	NDCG@3	NDCG@10	
1	TF-IDF	0.319	0.382	0.462	Lexical matching
2	BM25	0.308	0.373	0.455	
3	WTM	0.332	0.400	0.478	
4	LSA	0.298	0.372	0.455	Unsupervised methods 只利用文档语料训练
5	PLSA	0.295	0.371	0.456	
6	DAE	0.310	0.377	0.459	
7	BLTM-PR	0.337	0.403	0.480	Supervised methods 利用点击数据
8	DPM	0.329	0.401	0.479	
9	DNN	0.342	0.410	0.486	Paper's methods
10	L-WH linear	0.357	0.422	0.495	
11	L-WH non-linear	0.357	0.421	0.494	
12	L-WH DNN	0.362	0.425	0.498	

Table 2: Comparative results with the previous state of the art approaches and various settings of DSSM.

DNN: 没有WH, 同DAE, 使用Clickthrough data, 输入40k-vocab

L-WH linear: 有WH, 不使用非线性激活函数

L-WH non-linear: 有WH, 使用激活函数如tanh

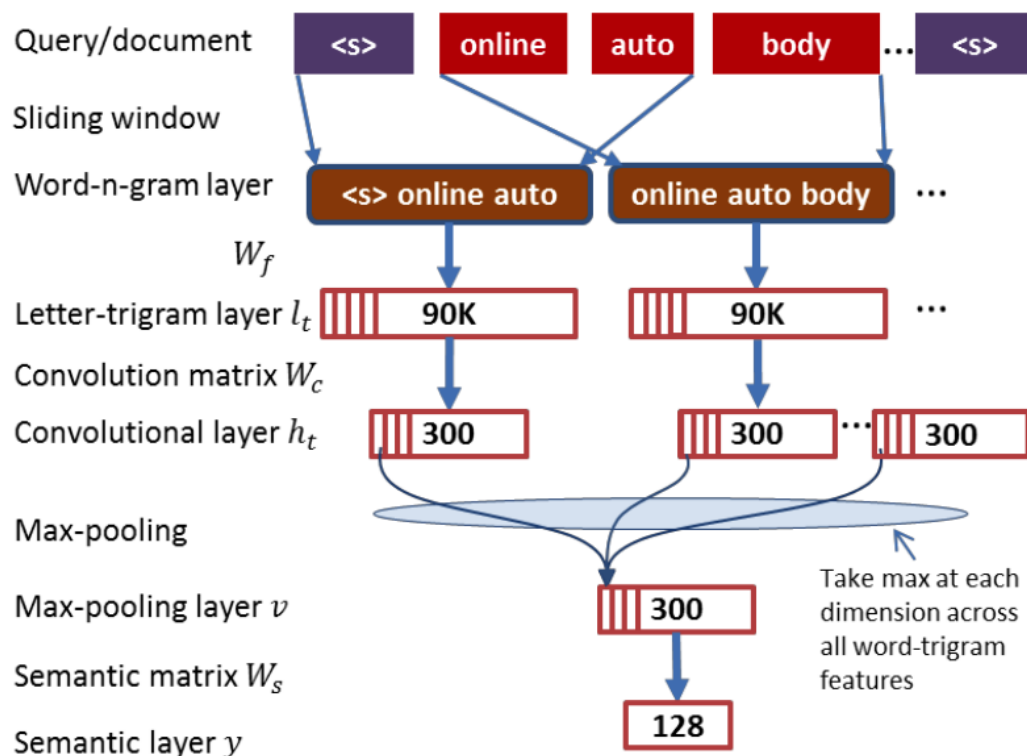
L-WH DNN: 有WH, 三层网络, 500k-vocab, 如模型图所示

DSSM-文章总结

- 充分利用Clickthrough data，直接对文档和查询项得分建模
- 借鉴语音识别思想，使用深度学习框架解决问题
- 针对大规模语料的单词表示稀疏问题，提出一种基于字符级别的word hashing机制，在不损失实验效果下，有效地降低了单词的表示维度。同时既克服OOV问题，可以提高模型的泛化能力
- 非常适合信息检索领域，充分利用Clickthrough data
- DSSM不仅可以计算文档和查询项的得分，而且能得到向量表示作为文档/查询项的语义表示

DSSM-不足及改进

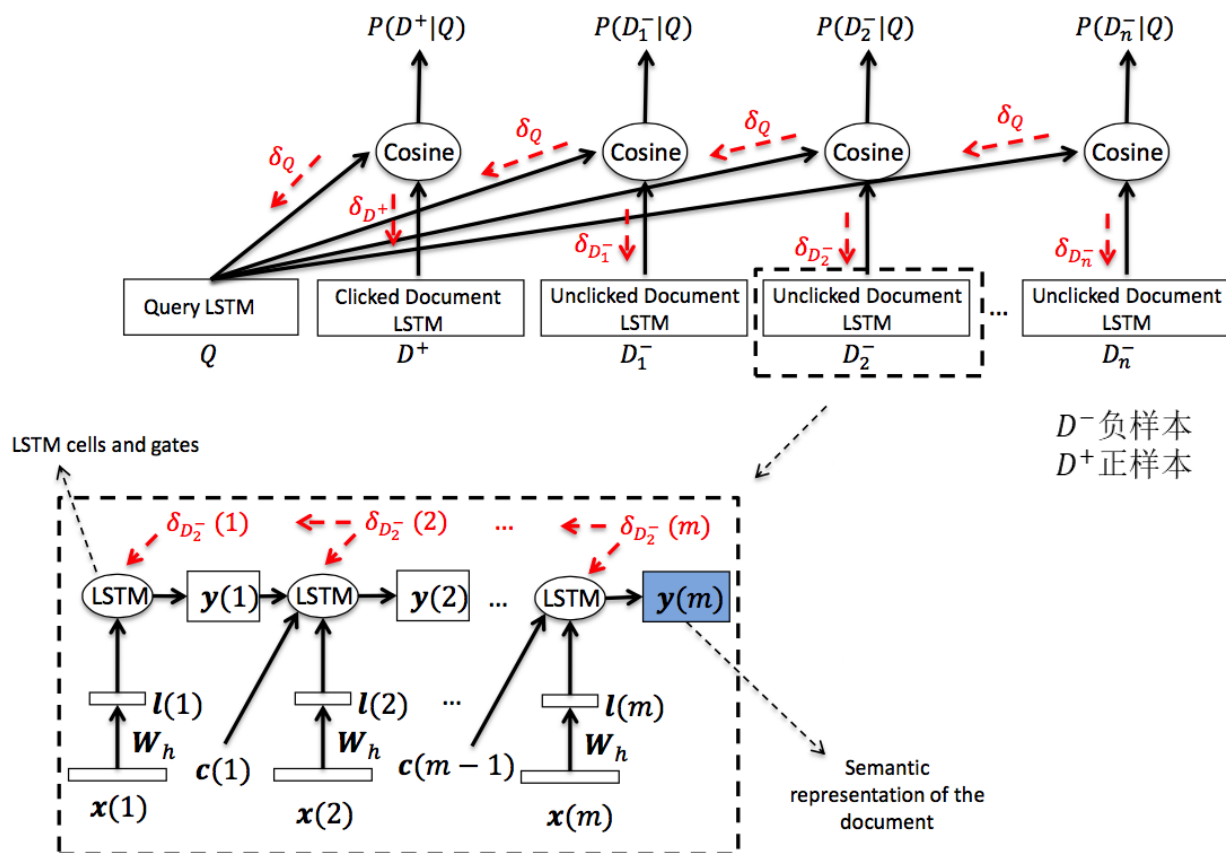
- ❑ 丢失上下文信息，结合滑动窗口和CNN弥补
- ❑ 改进模型:CDSSM (CLSM, Convolutional latent semantic model)



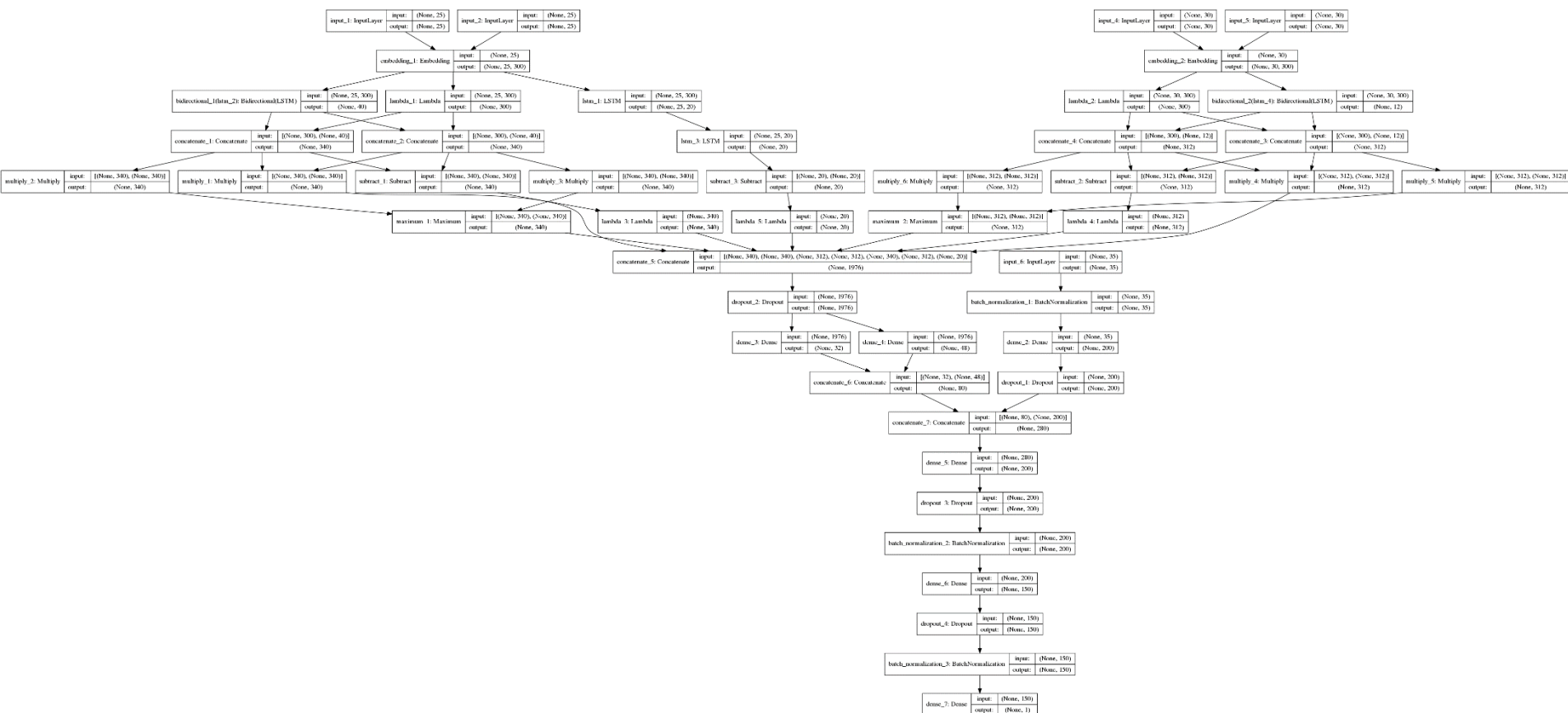
1. 使用指定滑窗大小对输入序列取窗口数据（称为word-n-gram）
2. 对于这些word-n-gram按letter-trigram进行转换构成representation vector(其实就是Word Hashing)
3. 通过卷积层提取了滑动窗口下的上下文信息
4. 使用max-pooling层来取那些比较重要的word-n-gram
5. 再过一次全连接层层计算语义向量
6. 最终输出128维 语义向量

DSSM-不足及改进

- 捕获较远距离上下文特征
- 改进模型:LSTM-DSSM



ATEC比赛模型



Reference

- Learning Deep Structured Semantic Models for Web Search using Clickthrough Data, 2013
- Shen, Yelong, et al. “A latent semantic model with convolutional-pooling structure for information retrieval.”. ACM, 2014
- Palangi, Hamid, et al. “Semantic modelling with long-short-term memory for information retrieval.” arXiv preprint arXiv:1412.6629 (2014).
- <http://kubicode.me/2017/04/21/Deep%20Learning/Study-With-Deep-Structured-Semantic-Model/>
-