R&S[18] | SIGIR2018：深度学习匹配在搜索与推荐中的应用

原创 机智的叉烧 CS的陋室 2019-08-24

Not Afraid

Eminem - Not Afraid

往期回顾：

- ML&DEV[2] | 机器学习开发技能入门线路
- ML&DEV[1] | 机器学习数学基础入门线路
- NLP.TM[16] | SIGIR2019: 深度NLP在搜索系统中的应用
- NLP.TM[15] | 短文本相似度-CNN_SIM
- R&S[17] | 手把手搞推荐[6]：回顾整体建模过程

最近看到一篇报告，在SIGIR2018中发表，材料虽然只有PPT版本的PDF，但是阅读起来其实还是收获满满，这次给大家分享一下。

开始之前，先放两篇文章给大家看，与本文相关。

- 这次内容参考了知乎另一位大佬的内容，我把链接放在这里：https://zhuanlan.zhihu.com/p/38296950
- 我之前也分享过有关文本相似度的论文，与本文的深度学习匹配相关，所以放在这里：NLP.TM[15] | 短文本相似度-CNN_SIM

老规矩，我不会翻译或者把PPT讲一遍，我会挑里面的重点以及有意思的地方点出，详情大家直接看报告内容吧。想要资料的话，报告PPT关注我的公众号：“CS的陋室”，回复**"match-DL"**，即可得到PDF链接。

懒人目录：

概述

概述部分简单讨论了搜索和推荐的差别，此块不是本报告的重点，所以不是很深，但是感觉对理解两者区别有很大帮助，这里我结合PPT内容和个人理解给出我的解释吧：

- 用户在搜索中的意图远远比推荐要明确而且精准。
- 搜索会有明确的query可供分析，当然可有结合用户特征来推断，但是推荐只能靠用户特征来推断。
- 搜索强调意图以及相关性，推荐强调兴趣，后者还可以带有猜测性。
- 推荐需要对信息生产者和消费者同时满足，搜索需要更倾向于信息消费者。

传统的匹配方法

传统匹配本身也不是本文的重点，作者只是简单提到作为铺垫。

传统query-doc匹配模型

作者将传统方法分成5种思路：

- query规范化
- 词汇独立性（共现规则）
- 主题模型
- 隐空间
- 统计机器翻译

里面提的方法很多，其中包含非常经典的搜索匹配算法BM25、 tfidf、PageRank等，也有PLS（partial least square）、RMLS（regularized mapping to latent space）这两种线性相似度衡量模型，然后还提到一些比较经典的统计机器翻译模型SMT等。

当然的，这些模型其实可以尝试在现实场景中使用，由于简单的实现且模型简单，计算性能和结果性能压力小，是基线模型的尝试性方案。

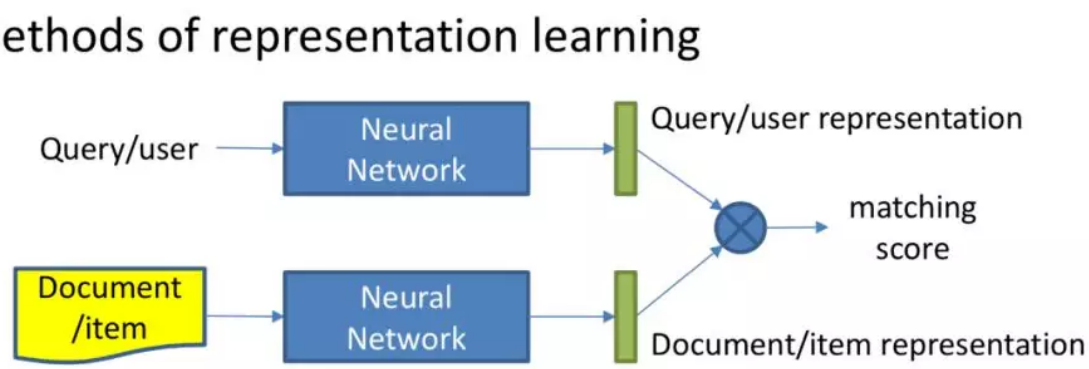
传统推荐匹配模型

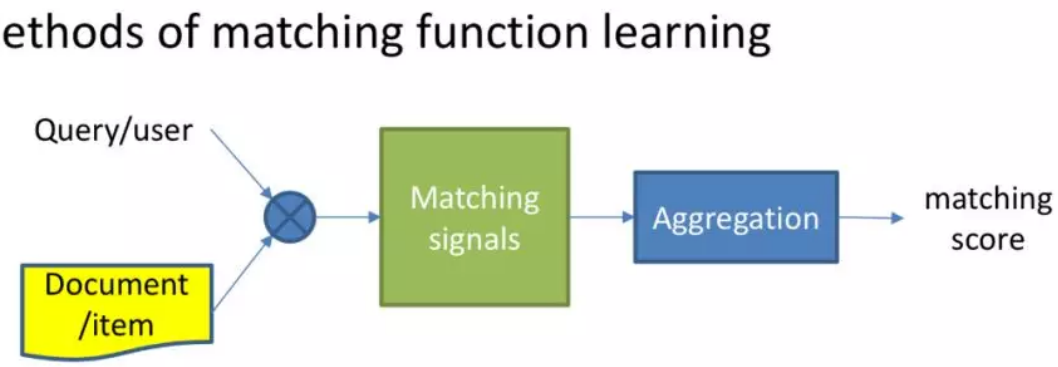
传统推荐匹配模型在现在其实很多人都会知道，基本就是大家能谈的，小至协同过滤，达到矩阵分解、FM等，大家也比较实习，此处不再详谈啦。

搜索篇

深度表示学习

作者认为，深度学习在语义匹配上的优势主要体现在两点，也如下图所示：





- 表示学习，从简单的onehot、词袋形式到分布表示，表达语义的能力更强。
- 匹配模型，从手工特征到自动化提取特征，损失函数逐步升级改善，融入更多匹配信息，再者考虑一些软性的匹配信息（例如连续型）。

DSSM

是语义匹配的开山之作，非常简单，但是在当时其实具有一定的启蒙性作用了，里面既有余弦距离这个经典语义匹配的痕迹，也有感知机这种比较新潮的模型，总结模型要点。

- 对query和doc分别进行，Letter-trigram：“#candy# #store#” --> #ca can and ndy dy# #st sto tor ore re#
- 分别对query和doc，根据整个句子的Letter-trigram转为onehot句子向量。
- 分别全连接层训练分布式句向量。
- 损失函数用softmax嵌套query和doc的余弦距离表示。

上述模型的优点其实非常明显，里面一些trick其实我们可以采用。

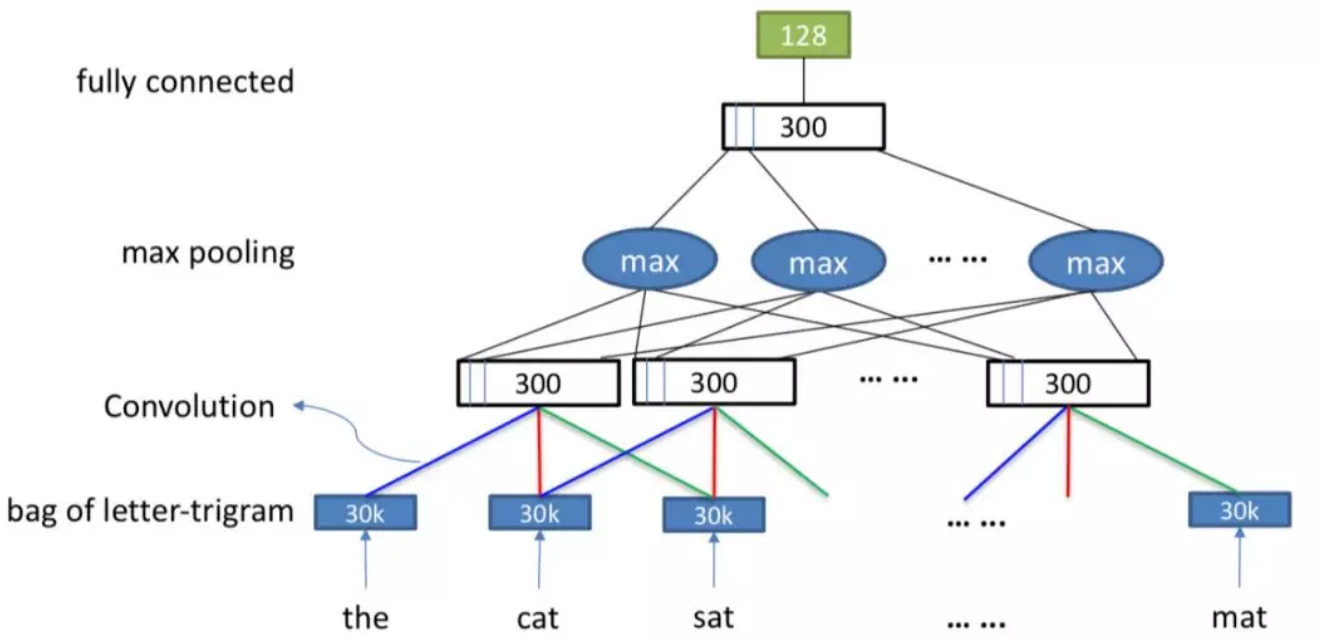
- Letter-trigram能有效降低词典大小（但似乎直接对英文没太大用，不过可以作为启发吧）
- 减少出现词典外词汇的问题，泛化能力提升
- 抗拼写错误

但是因为使用的还是词袋模型作为基础，所以语序没有考虑其中从而影响了语句信息提取，这是词袋模型的核心短板，要突破，最常见的方法就是利用CNN提取短期局部信息和利用RNN尽可能地提取长期整体信息。

CNN捕捉局部信息

在NLP领域，CNN具有很高的地位，因为他对局部信息的抽取非常敏感，一维卷积能够提取单个位点的关键信息，同时保持局部序列信息，但是一维卷积的局限性比较大，引入多维卷积可以体现N-gram特性。（ARC-I和CNTN都体现了类似的特性）

值得重点提到的是，Shen 等在2014年CIKM中发表了CDSSM，在DSSM的基础上引入了卷积层和池化层，效果相比CDSSM有一定的提升（我的感觉其实并不是特别大，因为其实letter-trigram已经能体现一定的局部信息），文章的简要示意图如下，该图由本次报告者给出，并没有采用本原论文的图，用以凸显CDSSM和DSSM的区别。



RNN捕捉序列信息

RNN甚至是RNN族模型，如GRU和LSTM，重点都在于提取序列信息，比较复杂的结构甚至能借助”门“等方式，提取重点信息记录，并在最终输出中体现，另外，使用双向RNN的方式甚至能有更好的效果（例如Palangi等提出的LSTM-RNN形式）。

匹配形式

说白了，就是最终怎么体现匹配程度，常见的，其实有三种形式。

- 点积。在分别对query和doc进行一系列的深度学习模型堆叠后，后去句向量，利用点积计算两者距离。
- 余弦距离。我的理解是点积的一种标准化版本。
- 多层感知机（全连接层）。个人感觉也是点积的变式，就是矩阵点积时新增了一个系数矩阵，或者说点积是多层感知机下的一种特殊情况，其实就是一个一层系数矩阵为单位矩阵的感知机，因此，可训练的多层感知机能使匹配计算更具有灵活性。

个人更加喜欢第三个，当然第二个我也没试过不知道和第三个比那个比较好，大概这样吧。

匹配度量学习

上面的重点在于更为精准地表达query和doc，但是在匹配上就显得非常简单，如果有更加合适的信息综合提取方法，应该能有更针对问题特异性的新方案产生，匹配结果会更加精准，因此在匹配度量一块有很多挖掘的空间。

ARC-II

Hu等在NIPS2014上提出的ARC-II模型。

- 句向量形式分别表达query和doc。
- 句向量一一组合拼接（concat）后一维卷积，共构成互动矩阵。
- 互动矩阵进行多次二维卷积，接入全连接层。

这个模型的优势在于有匹配矩阵，可解释性强，基本的卷积可以保留序列信息，但是unigram的信息体现的不是很多，另外个人觉得对query和doc有一定的长度约束，两者的长度差太远可能会出现问题。

Match-SRNN

IJCAI2016的论文，论文的内容很多，我会精读这篇论文，这里提一个比较关键的点：

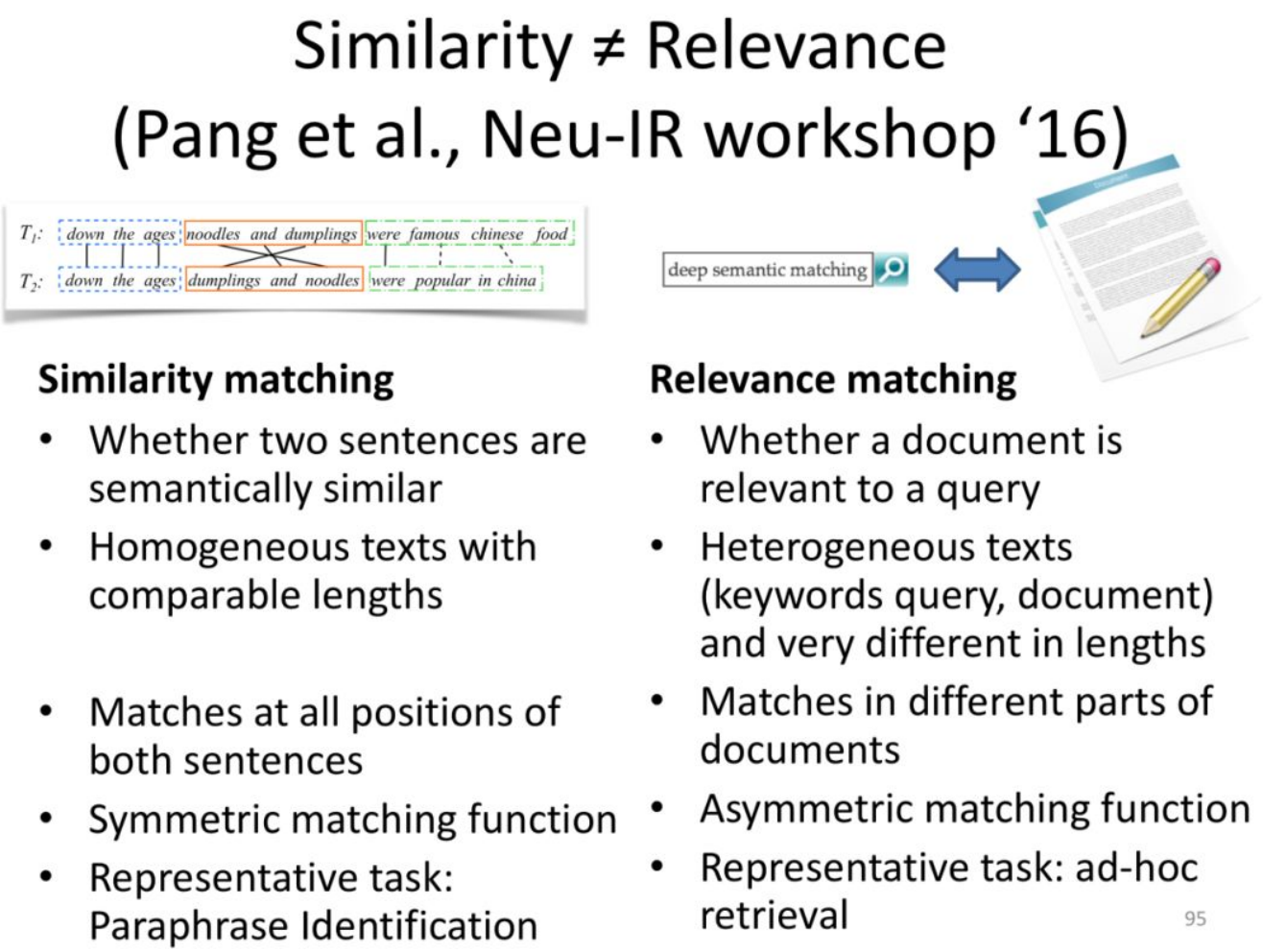
在构建完词汇级别的一一匹配互动矩阵后，利用类似动态规划的思想来计算两者的相似度，f[i][j]表示query的前i个词和doc的前j个词的匹配分，而这个f[i][j]则与前置位的几个数据有关，f[i-1][j]、f[i][j-1]、f[i-1][j-1]，于是此处的目标就是你找到一个合适的函数完成下面的映射：

f[i][j]=func(f[i-1][j],f[i][j-1],f[i-1][j-1])

这里就简单谈这两篇比较有亮点的论文，两者的亮点都在于探究如何匹配更能体现语义以及其匹配程度，深度学习层面的更新其实在匹配上并不一定有新意，例如加入attention，效果肯定会有提升，这其实是在做匹配问题时，探讨什么是匹配，这个哲学问题是非常需要思考的。

相关性的讨论

虽然前面说的都是匹配问题，但是其实是语义上的匹配，说白了更倾向于从相似的角度去衡量，而现在讨论的是相关，具体两者的区别作者给出了解释：



感觉醍醐灌顶啊有木有。

我来举个栗子吧，“明天天气是什么”对“晴转多云”，语义上肯定是不相似的，但是意思上是相关的，或者说从搜索层面就是匹配的。

有关相关性分析，作者将现在的研究现状分了两个大思路，一个是基于全局信息，一个是基于局部信息。

基于全局信息

基于全局信息的匹配模型，通过抽象query和doc的信息，进行匹配，求出累计分布 ，来计算相似度，优点在于对短query长doc的计算敏感度高（基于局部信息的其实也有这个优点额），鲁棒性也高，但是缺点在于放弃了对序列信息的关注。

比较想介绍的是Xiong等在SIGIR17上发表的K-NRM模型，虽然简单但是用了很多我们可以借鉴的trick。

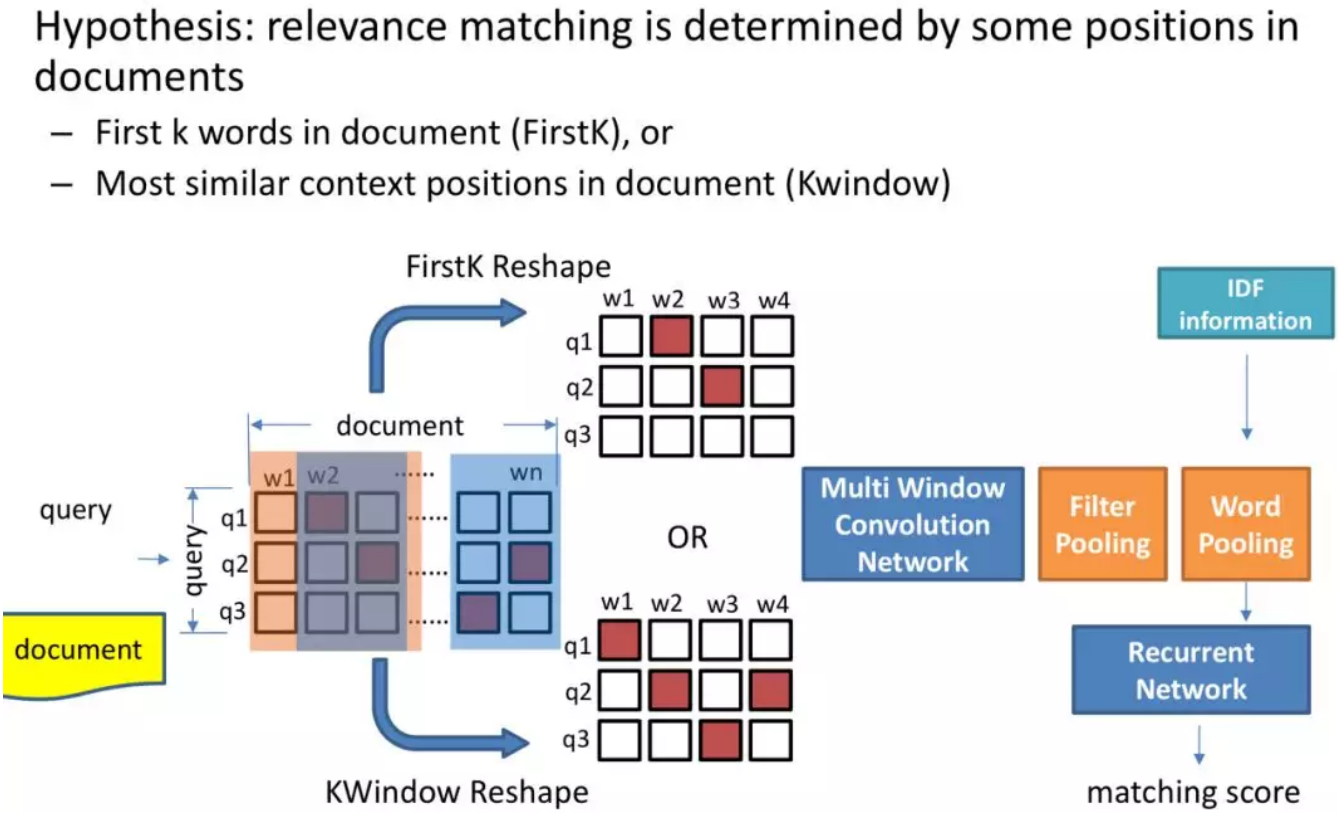
- 词嵌入+余弦距离构建相似度矩阵。词嵌入现在NLP的基操，通过词嵌入其实能体现模型对语义的表达，另外预先相似度也是本报告提到最多的相似度计算方式。

- 运用核池化做非线性的特征提取和组合。非线性核池化能软性提取关键信息，不会关注于一个位点，也不会关注整体情况的线性组合情况。

基于局部信息

基于局部信息在于关注doc中的局部关键信息，提取后和query进行匹配，优点在于过滤了不重要的信息而关注关键信息，同时还能够保留一定的序列信息。

这里想聊Hui等在2017年EMNLP中提出的PACRR模型。它的核心假设是关键的匹配信息在于doc特定位置的信息，因此可以用不同大小的卷积核获取各种粒度的匹配信号，池化和组合后，放入RNN进行计算，个人感觉这是一个CNN+RNN组合的新方法，非常具有新意，同时结合匹配的实际场景，非常有用。



推荐篇

推荐和搜索相比最鲜明的特点就是用户不会主动输入太多信息，因此我们需要从侧面去挖掘和发现，因此推荐系统会话很多心思在对用户和item的表征上。

和搜索相同，作者主要从表示学习和度量学习两个角度讨论有关深度学习匹配的思路，其实看里面的模型，大都是非常经典或者非常流行的方法，只是报告人的分类角度不同而已，其实这也是在文献综述中非常常用的技巧，通过特定的方式分析论文以表现观点。

但是吧，我其实并不想太过详细地谈很多比较强的模型，而是从中提取一些tricks供大家参考，有关论文我会以参考文献的方式备注在最后。

表示学习

作者以协同过滤为核心点讨论表示学习，一方面借助用户-item的互动矩阵来表征两者关系，另一方面，也会尝试借助一些外部信息来协助表征两者关系。

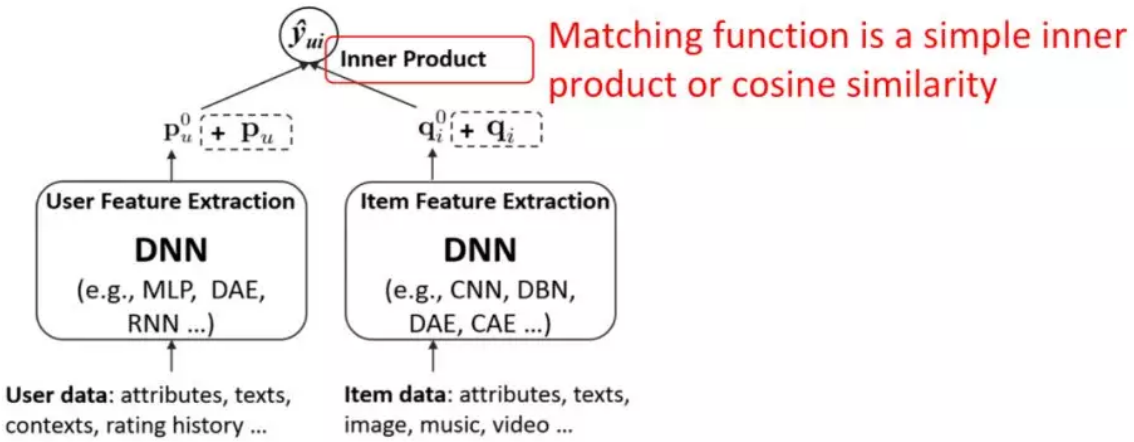
借助用户-item的互动关系信息方面，我看到了非常流行的deeoFM，但我更想拿CDAE来说事，即协同降噪自编码器，由Wu等在WSDM2016中发表。

- 输入其实非常朴素，用户ID和一些有关用户针对item的历史行为的记录。
- 自编码器的使用，实质上是MLP+MF的格式，多层感知机体现非线性，MF体现线性，复杂格式融合使模型更加灵活。
- 中间使用降噪自编码器，可以避免用户的一些异常行为（可被看作是噪音）对用户表征带来的影响。

而针对外部信息来表征两者关系，其实就是在上述基础上加上一些外部信息，例如用户的性别年龄职业地址等，item的标题主题内容发布时间等，甚至有商品图片等信息，反正就是脑洞无敌大，当然的信息越多，对推荐效果就会越有利，由于外部信息的形式各异，所以很难有说非常具有代表性的方法。

通用的外部信息，DCF（Deep Collaborative Filtering via Marginalized DAE）借助降噪自动编码器进行处理，而针对图像信息，DUIF（Deep User and Image Feature Learning）使用了AlexNet提取商品图片特征并借助MF等方式将信息整合在CKE（Collaborative Knowledge Base Embedding）则将item中的文本信息考虑其中，用nlp相关的方式提取好文本信息放入模型。

- A General framework to summarize the above works:



- Depending on the available data to describe a user/item, we can choose appropriate DNN to learn representation.

  E.g., Textual Attributes -> AutoRec, Image -> CNN, Video -> RNN etc.

其实说白了，我们其实就是要想方设法做两件事情：

- 表征好用户对item的历史行为。
- 挖掘用户和item中体现的特征，并将其作为特征纳入到模型中。

匹配度量学习

在这里作者将其分为两块，协同过滤和基于特征的模型，前者大家很熟悉，不过在基本的协同过滤上可能可以加上卷积层、机器翻译等神奇的操作，而基于特征的方面的，wide&deep和FM非常具有代表性。

He等在WWW2017中提出了神经协同过滤框架，尝试将协同过滤的主体替换为神经网络，这个思想其实很多人都从不同角度提到，但是这个框架的提出算是对一系列方法的总结。

- **NCF is a general framework that replaces the inner product with a neural network to learn the matching function.** $\hat{y}_{ui} = f(\mathbf{p}_u, \mathbf{q}_i)$

Matching function based on NN

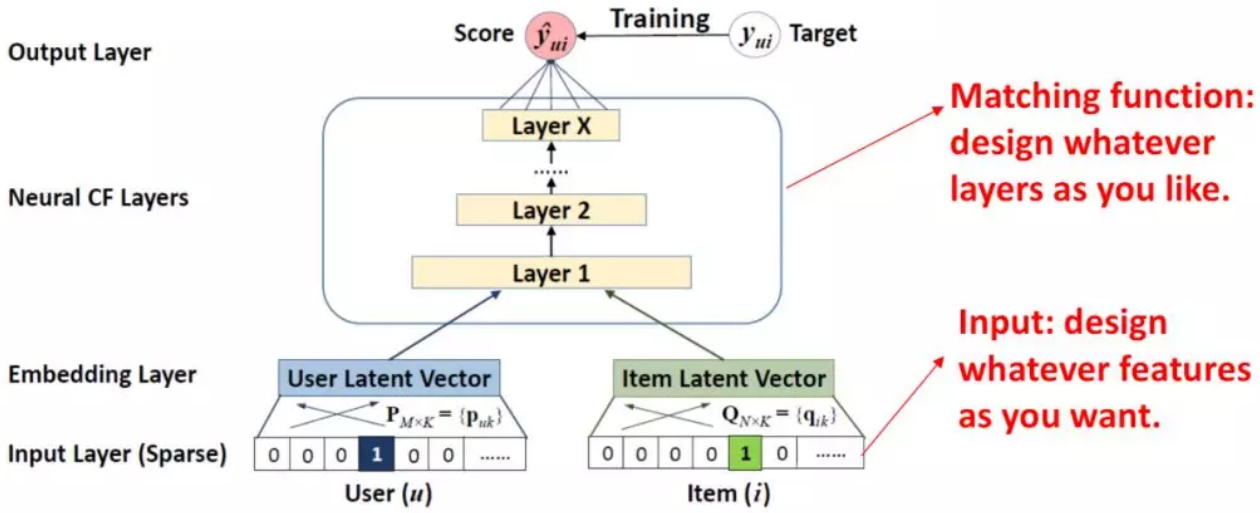


Figure 2: Neural collaborative filtering framework

128

虽然如此，作者单独强调了一个颇为重要的信息，那就是卷积层在处理乘性特征时效果不佳。

有意思的是，机器翻译的思想居然能用到匹配度量学习上，实现对未来用户兴趣的预测（He et al, Recsys’ 17）。

- Focused on next-item recommendation
  - Third-order relationship between <user, current item, next item>
  - Define relation vector as the current item:

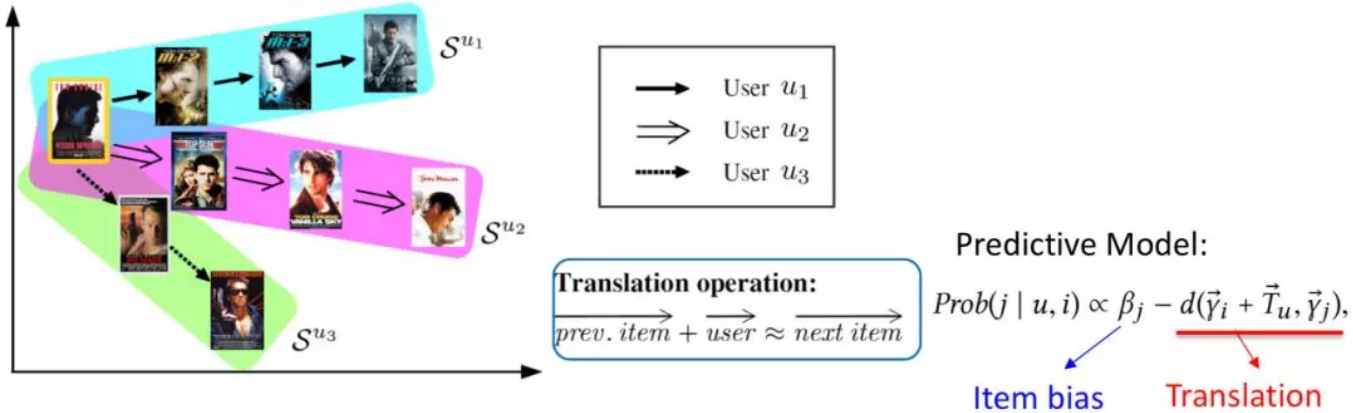
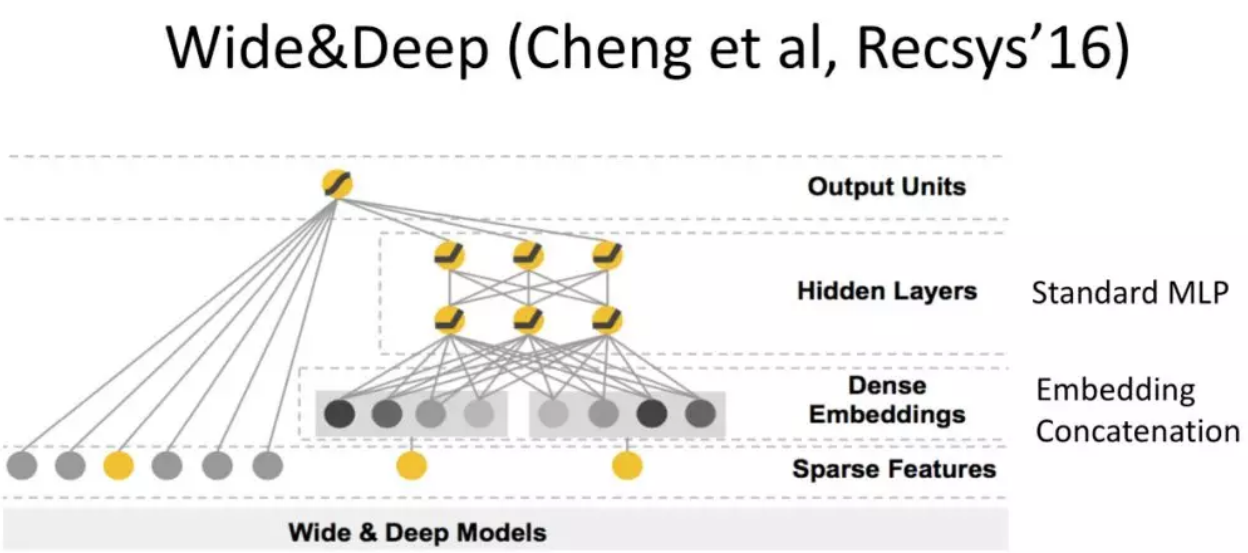


Figure 1: *TransRec* as a sequential model: Items (movies) are embedded into a ‘transition space’ where each user is modeled by a *translation* vector. The transition of a user from one item to another is captured by a user-specific translation operation.

而在基于特征的匹配度量学习中，作者提到了3个关键点：

- 原始特征向量大都稀疏，因此有必要进行embedding。
- 特征之间的互动关系需要被关注，同时捕捉方法也非常重要。

wide&deep是非常经典的一种方法，借助两种方式对相近的特征体系进行深度和浅层的抽取，浅层抽取在于广度和可解释性，用的是简单的线性回归，而深度的在于产生难以被挖掘的特征信息， 目前这个方法仍为众多场景所使用。



而在FM上，作为经典模型，我这里也简单谈到自己的观点：

R&S[7] | 深度讨论FM和FFM：不仅是推荐

当然了，在后续还有人提出了deepFM，将深度学习纳入其体系下，甚至有NFM（神经FM）等操作，从作者角度，FM的变式实验效果其实不错，但是具体在应用场景，还需要时间的检验。

总结

终于写完了，感觉这应该是我写的技术文里花的时间最长的一次，除了本报告外，我还读了一些里面提到的论文，并从中提取了一些关键点，感觉收获还是不小的吧，希望大家读完也能有收获，里面很多技巧，其实是我们作为算法、数据科学等工工作人员锁需要的，尤其是针对特定的场景，我们需要给出特定的解决方案。

现在来简单归纳一下吧。

- 针对匹配问题，其实主要考虑两个方面内容，匹配对象两者的表征和度量两者相似度的方法。
- 报告中大部分方法到了匹配下游都是有监督学习，有标注数据有利于让模型理解何为你定义的相似。
- Letter-trigram似乎可以抽取到英文单词中的词根词缀，中文似乎能有这方面的技巧？（笔画嵌入似乎雷声大雨点小，不知道有没有更好的方法）
- 一维卷积和多维卷积在NLP中的理解。
- 互动矩阵构建有多种方法，都可以去了解一下。
- 有关相似和相关之间的思考。
- 核池化。
- 自编码器以及其变式在很多观点看来是比较积累的，但是在匹配这块，似乎得到了一些青睐。
- 流行模型必有其流行的理由，结果导向来看就是效果好，归因的话，个人认为是找到了一种合适的整理特征方法，他们在输入的角度是平行的，但是根据模型的训练，会自动地通过权重体现其重要性。

推荐阅读材料

为了方便大家进一步深入了解，我整理了一些本文提到的一些重要论文和材料，欢迎大家阅读。

[1] 报告文献：Xu J, He X, Li H .Deep Learning for Matching in Search and Recommendation. SIGIR 2018.

[2] 机器翻译信息检索： Adam Berger and John Lafferty. Information Retrieval as Statistical Translation. SIGIR 1999.

[3] DSSM：Huang P S, He X, Gao J, et al. Learning deep structured semantic models for web search using clickthrough data. CKIM 2013.

[4] CDSSM：Shen Y, He X, Gao J, et al. A Latent Semantic Model with Convolutional-Pooling Structure for Information Retrieval. CKIM 2014.

[5] RNN捕捉序列信息：Palangi H, Deng L, Shen Y, et al. Deep sentence embedding using long short-term memory networks: Analysis and application to information retrieval. TASLP 2016.

[6] ARC-II：Hu B, LuShen Y, He X, Gao J, et al. A latent semantic model with convolutional-pooling structure for information retrieval. NLPS 2014.

[7] Match-SRNN：Wan S, Lan Y, Guo J, et al. A Deep Architecture for Semantic Matching with Multiple Positional Sentence Representations. IJCAI 2016.

[8] K-NRM：Chenyan Xiong, Zhuyun Dai, Jamie Callan, Zhiyuan Liu, Russell Power. End-to-End Neural Ad-hoc Ranking with Kernel Pooling. SIGIR 2017.

[9] PACRR：Hui K, Yates A, Berberich K, et al. PACRR: A Position-Aware Neural IR Model for