

082 | 基于深度学习的推荐模型之二：基于RNN的推荐系统

2018-04-11 洪亮劫

AI技术内参

[进入课程 >](#)



讲述：初明明

时长 06:44 大小 3.09M



周一我们看了一篇经典的文章，那就是尝试使用受限波兹曼机（RBM）来对推荐系统建模。这应该是最早把深度学习应用到推荐建模的典范。当然，RBM 虽然算是相对比较早的深度学习模型，但其本质上并没有很多后来提出的深度模型的特质，也没有真正发挥出深度学习模型的特有优势。

今天，我们结合几篇论文来看一类深度学习模型，那就是“**递归神经网络**”（Recurrent Neural Network），简称 **RNN**，在推荐系统中的应用。这类模型可以解决传统推荐模型中难以对时序信息进行建模的问题，拓宽了推荐系统的应用边界。

时序信息建模

要想说清楚 RNN 对于推荐系统的贡献，我们首先要来看一看为什么需要对时序信息进行建模。

在我们前面介绍的诸多推荐模型中，不管是矩阵分解，还是简单的基于流行度的推荐模型，或是其他更加复杂的张量分解等模型，其实都没有内置“时间”这个概念。

比方说，我们通过矩阵分解来对用户和物品的评分进行建模。在这个建模的场景里，用户物品评分矩阵是针对所有数据的，不管是用户昨天对某个物品进行评分还是今天，甚至是一年以前，所有数据都是在唯一一个矩阵里面加以表示。这么做的好处是，极大地简化了真实情况，不好的地方则是完全忽略了所有评分在时间上的发展。

其实早在 Netflix 大赛中，一些学者就在 Netflix 公布的数据集上发现，用户对于电影的喜爱程度，或者说评分数值，有非常明显的随时间变化而变化的趋势。文末我列的参考文献 [1]，这篇论文就是充分挖掘了时间信息从而带来了性能上的提升，如果你有兴趣的话，建议读一读这篇文章。

在深度学习模型，特别是 RNN 之前，如果我们希望对时间进行建模，从模型工具的角度上来说，我们都有哪些选择呢？

一种办法是可以尝试使用传统的“时序模型”（Time Series Models）。这一类模型在统计领域已经使用了较长时间，然而最大的问题就是，很多工具很难直接和我们提到的这些推荐模型进行嫁接。另外一个难点是在嫁接之后，模型的训练算法往往会变得异常复杂，这也给模型的普及和使用带来了许多障碍。

另外一种办法，就是尝试在现有的模型里通过特性（Feature）或者其他的方法，来让当前的模型能够对时间信息进行建模。这个思路其实是对矩阵分解进行了修改。这样做的好处就是可以根据自己的需要在某一个模型上进行必要的更改，然而这么做的先天性缺陷就在于提出来的修改往往只能针对某一个模型，而没有办法扩展到其他模型。

第三种做法是可以利用张量分解（Tensor Factorization）。我们直接把时间作为一个新的维度，因此可以对用户在某一个时间段对某个物品的评分进行建模，有一些工作是基于这个假设的。

不过，直接利用张量分解的最大问题是，张量本身并不是时序模型，利用张量对时序信息进行建模仅仅是因为时序信息大多时候可以被表达成为离散的数据，因此张量才在这里有了用

武之地。然而，因为张量无法直接对离散时序的两位数据点之间进行约束建模，比如时间点“昨天”和时间点“今天”可能在张量中占据两个不同的数据点，但是张量本身并不能把这两个数据点联系起来。也就是说，张量在“语义”（Semantics）上其实并不支持时序数据。

基于以上这些原因，我们需要有新的工具来对时序信息进行比较直接的建模，同时也能有相对容易的学习算法。

基于 RNN 的推荐模型

RNN 作为深度学习中有力的时序信息建模工具，随着深度学习的火热，被越来越多的学者用来解决我们刚才所说的这些问题。我们一起来看两篇论文，可以说这两篇文章是 RNN 在推荐领域应用的重要参考。

除了从工具的角度来讲为什么需要 RNN 以外，还有一个是从实际的场景出发，也就是在同一个“会话”（Session）中的推荐结果，这是一个比较突出的需要时序信息的场景，且传统的方法无法很好地解决。

在这方面比较早的尝试，来自于在 ICLR2016 上发表的一篇论文《使用递归神经网络的基于会话的推荐》（Session-based recommendations with recurrent neural networks）[2]。这里面的基本思路其实很直观，就是在推荐系统中使用某种 RNN 模型，在这篇论文里使用的是 **GRU**（Gated Recurrent Unit）。我们在这里不展开 RNN 或者 GRU 的定义细节以及这些模型里面的特征。我们从比较高的维度来讲一讲 RNN 的建模思路。

RNN 的输入是当前的一个物品，然后 RNN 需要输出的是对于下一个物品的预测，同时为了能够对这些在时间序列上的物品进行建模，RNN 内部也要维持一个隐含的状态序列，这个状态序列会随着时间的变化而变化。不同的 RNN 模型在以下内容的实现上有所不同，包括如何实现这个隐含的状态序列，以及如何对从状态序列到输入和输出物品的建模。

总的来说，这篇论文就是直接把目前在自然语言处理中经常使用的 GRU 拿到了推荐领域，并且做了一些微小的修改。从实验结果来看，如果我们对会话进行建模的话，效果要明显好于没有对会话进行建模。

再介绍一个比单对会话进行建模更进一步的工作，是发表于 WSDM2017 上的一篇论文《递归推荐网络》（Recurrent Recommender Networks）[3]。我们前面提到了，矩阵分解的最大问题就是学习到的用户以及物品向量无法囊括时间的信息。在这篇论文里，作者

们假定用户的隐向量，物品的隐向量都会随着时间的变化而变化，而在某一个时刻的物品评分仅仅是那一个时刻的用户隐向量和物品隐向量的一个结合。在传统的模型里，要描述这种隐向量随着时间的变化而变化的思路就会充满难度。

在这篇论文里，一个重要的贡献就是使用了一种叫 **LSTM** (Long Short-Term Memory) 的 RNN 模型，来对这些用户的隐变量随着时间的变化而变化进行建模。我们在这里也不展开 LSTM 的基本定义。总体来说，LSTM 也是对某一组输入变量和输出变量在时间上的关系进行建模，同时利用内部隐状态序列来对输入和输出变量进行建模。

小结

今天我们聊了如何利用 RNN 对推荐系统中的时序信息进行建模。

一起来回顾下要点：第一，我们讨论了为什么对时序信息建模是非常必要的；第二，我们结合两篇论文，聊了利用 RNN 对时序信息建模的例子。

最后，给你留一个思考题，除了会话信息或者用户的喜好是比较明显的时序信息以外，推荐系统中还有哪些时序信息的应用呢？

欢迎你给我留言，和我一起讨论。

参考文献

1. Yehuda Koren. [Factor in the neighbors: Scalable and accurate collaborative filtering](#). ACM Trans. Knowl. Discov. Data 4, 1, Article 1 (January 2010), 24 pages, 2010.
 2. Balázs Hidasi, Alexandros Karatzoglou, Linas Baltrunas, and Domonkos Tikk. [Session-based recommendations with recurrent neural networks](#). International Conference on Learning Representations (2016), 2016.
 3. Chao-Yuan Wu, Amr Ahmed, Alex Beutel, Alexander J. Smola, and How Jing. [Recurrent Recommender Networks](#). Proceedings of the Tenth ACM International Conference on Web Search and Data Mining (WSDM '17). ACM, New York, NY, USA, 495-503, 2017.
-

AI 技术内参

你的360度人工智能信息助理

洪亮劼

Etsy 数据科学主管
前雅虎研究院资深科学家



新版升级：点击「👤 请朋友读」，10位好友免费读，邀请订阅更有**现金**奖励。

© 版权归极客邦科技所有，未经许可不得传播售卖。页面已增加防盗追踪，如有侵权极客邦将依法追究其法律责任。

上一篇 081 | 基于深度学习的推荐模型之一：受限波兹曼机

下一篇 083 | 基于深度学习的推荐模型之三：利用深度学习来扩展推荐系统

精选留言 (1)

写留言



林彦

2018-04-13



放入购物车，购买，评论，收藏是不是也属于用户的喜好？与客服的联系互动，用户的分享，与其他用户的互动我觉得是时序信息。