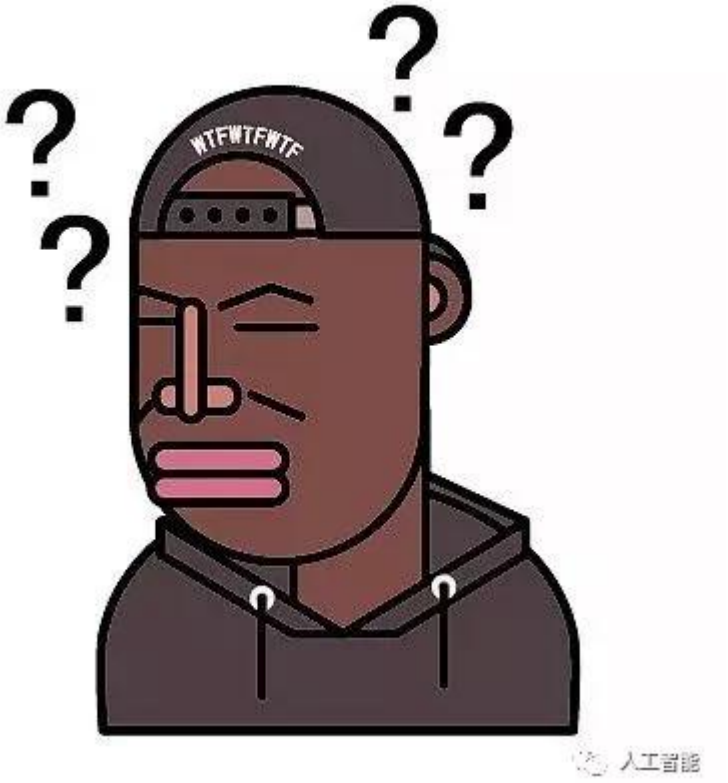
机器学习“剧透”权游大结局：三傻最先领盒饭，龙妈、小恶魔笑到最后

人工智能2019-04-16

千呼万唤，权利的游戏最终季终于开！播！了！

和大部分权游粉一样，文摘菌一整个上午的朋友圈都刷的战战兢兢，生怕被剧透。但是没有想到，还是“被“看到了大结局！而且，给我剧透的还是个算法。



这波最强剧透来自慕尼黑工业大学。早在最终季开播前，这所大学的计算机科学的同学就接到了一个特殊的作业：用机器学习，预测这一季谁最有可能坐稳铁王座。

这个听起来很有趣的项目用到了一种颇为残酷的算法——生存机会算法。具体的生存率预测，是通过寿命数据分析得到的结果。这种科学研究技术在医疗上已经有广泛的应用，例如用来检验治疗方法和并发症对癌症患者的影响。

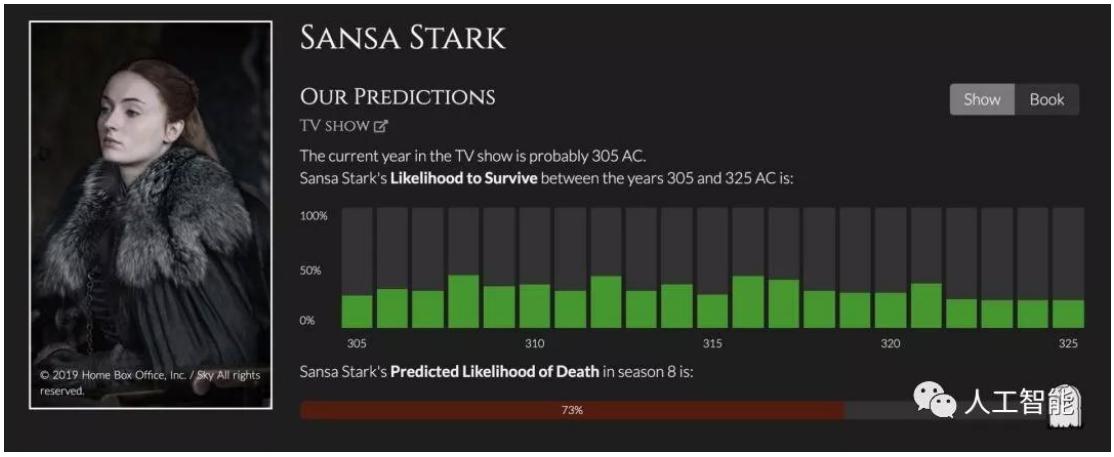
其实，生存机会算法是慕尼黑工业大学每学期 Javascript 研讨会的一部分，这个课题激起一届又一届学生的研究兴趣。他们开发了一个应用程序，并创造了一套人工智能的算法来计算每个人物的死亡率。早在 2016 年，第六季播出前，该专业的学生就准确预测了 Snow 的复活。

废话不多说，先来看看这个厉害的算法预测的最终季大结局。

根据算法，龙妈 DaenerysTargaryen（丹妮莉斯·塔格利安）生存的几率最高，达到了 99%，小恶魔 TyrionLannister（提利昂·兰尼斯特）也有 97%的存活率。

存活率排名一览

除了死亡率可能性最高的波隆和魔山，这个被七大王国最聪明的男人（TyrionLannister）预言最长命的“三傻”(SansaStark），她的死期也被预测的明明白白，死亡率高达 73%。

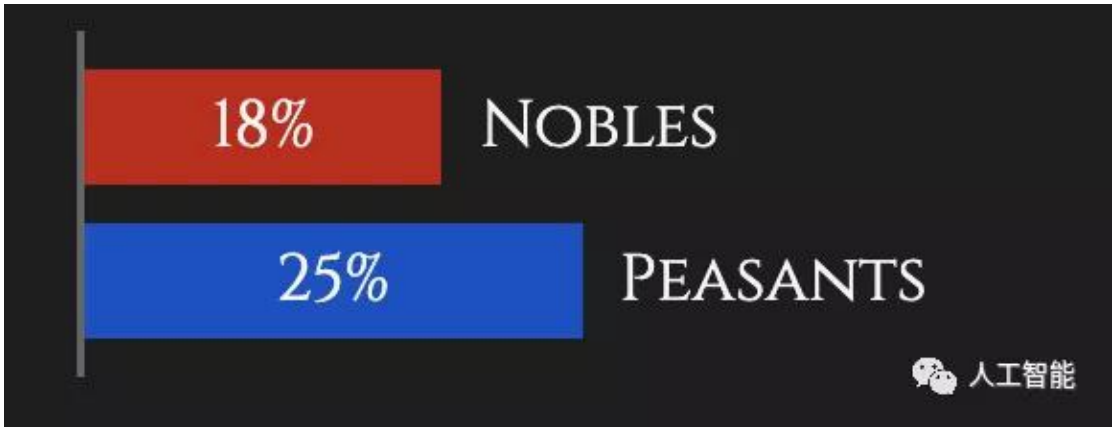


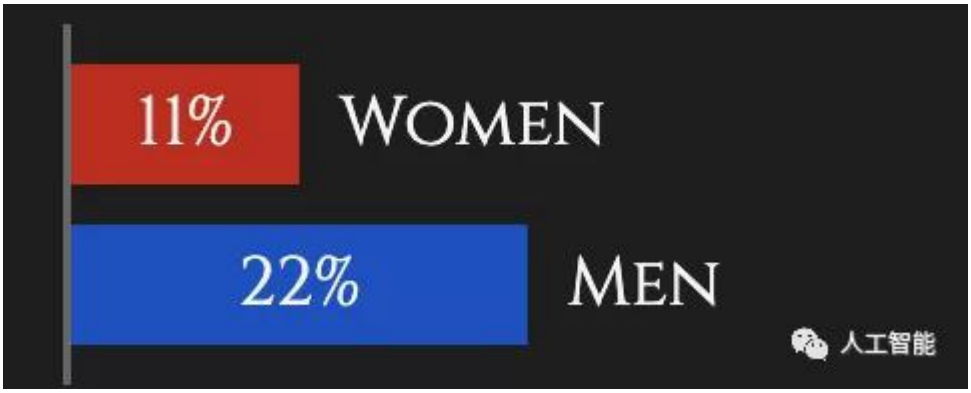
最强剧透如何操作？

算法具体怎么做的呢？简单来说，就是通过从原著和已播的剧集中提取人物角色、身份、性别、亲属数量、年龄、忠诚度、死亡概率等等来进行数据分析预测。

我们先拿 Sansa 的例子来简单解释一下。

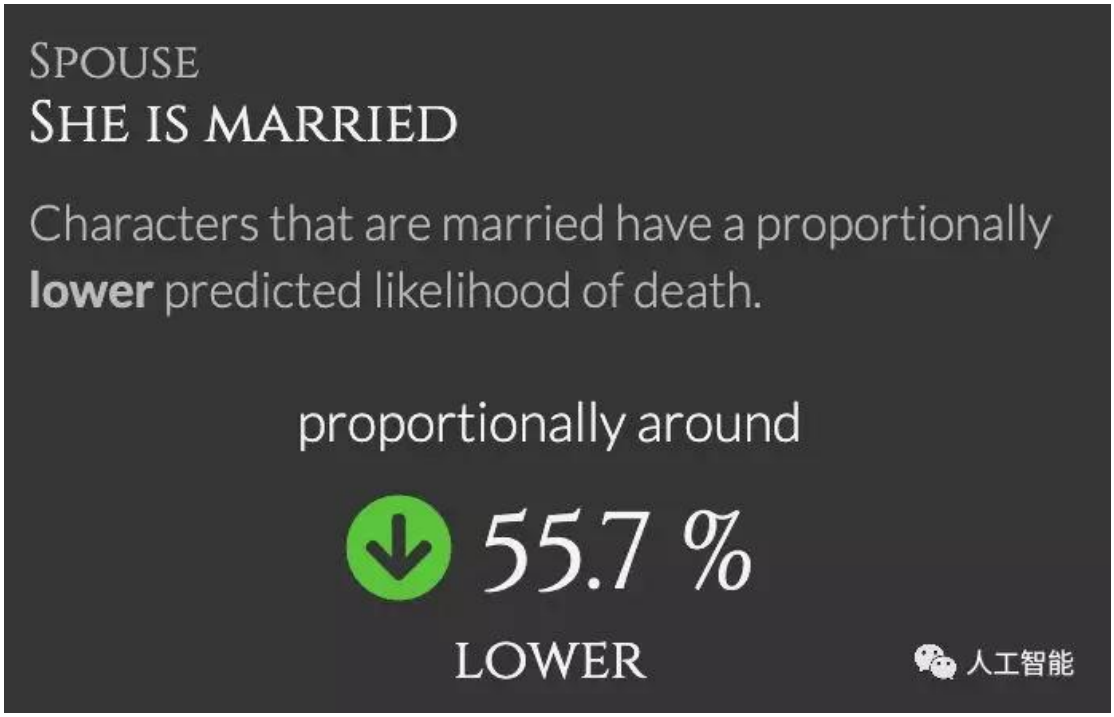
首先，在维斯特洛大陆，一般来说贵族是要比平民要活得久一点，而女性的角色也会比男性的要活的久一点。



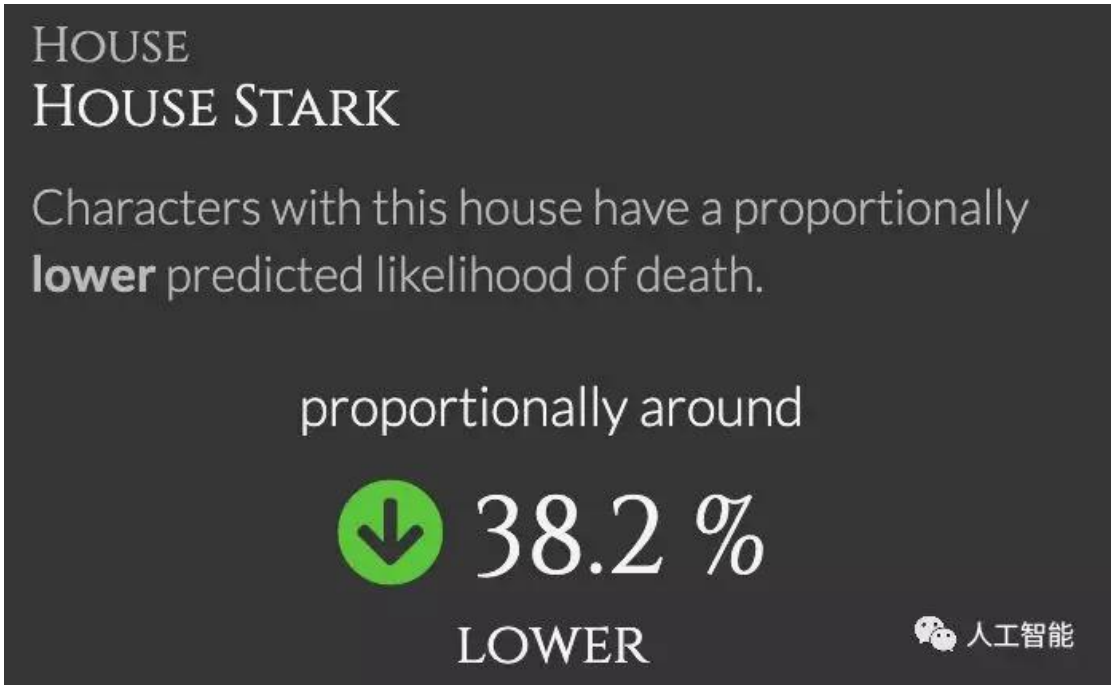


按主角来分，从北境之王的掌上明珠沦为最屈辱的贵族小姐再到临冬城女爵，主角光环下，死亡率一下降低 16%。

一般结过婚的女性也比较长命。虽然几段不幸的婚姻给 Sansa 带来惨痛的教训，但也完美的印证了“一切杀不死我的都将令我变强。”



出生在一个牛掰的家族也会延长寿命哦！史塔克家族的孩子虽然历经磨难，但除了“少狼主”外都还闯到了决赛圈。



忍辱负重的 Sansa 有命撑到第 8 季也是实属不易，但预测高达 73%的死亡率如何在剧中实现，我们可以拭目以待。

完整的人物清单及其生存机会等数据统计可以在以下网站在线获取。

https://got.show

“算法”与“数据”之歌

据这门课程的授课教师 BurkhardRost 介绍，虽然对权力的游戏里面人物角色生存率的预测只是来“幻想”数据，但是这种研究问题的方法早已被用于现实世界，并且对我们的日常生活产生了强大影响。类似的算法也能够在金融和医疗领域发挥作用。

“权力的游戏”世界互动地图的截图

数据提取

考虑实际情况，数据的最好来源是关于《权力游戏》的一系列维基百科介绍。在维基百科上基本囊括了 5 本书以及 8 季电视剧的内容，总计大约 2000 位角色的信息。除了提取角色的生存状态，即是否死亡，还需要其他的角色特征信息。

有了描述每一个角色特征的数据集，下一步是寻找能够判断角色是否死亡的特征集。

贝叶斯生存分析

模型的首要目标是使用贝叶斯推断相关方法来判断角色不同特征与存活率关系。模型假设，每一个都有一定的死亡概率。对于所有角色来说，“基本危险”到来的死亡概率都是相同的。演员之所以领盒饭，肯定他演的角色有“作死”特性。

例如，男人死亡概率为 60%，而呆在兰尼斯特家里可能会降低 50%。综合考虑这两个因素之后，就可以建立一个生存函数。

这个生存函数具体描述的是：在某个时间点，角色存活率。例如它可能告诉我们，乔恩·雪诺（JonSnow）活到 60 岁的概率是 45%，或者杰米·兰尼斯特（JamieLannister）被认为有 60%的几率能活到第八季。

使用带有 pymc3 封装的 MCMC 仿真来训练这个模型。选择下列几个特征进行分析：

- 家庭（House）
- 爱人（Lovers）
- 婚姻（Marriage）
- 地位（Titles）
- 主角/配角（Major/Minor character）

男性（Male）

-

-

神经网络（Neural Network）

- **Keras 建立模型**

  慕尼黑的同学使用了 Python 的 Keras 来建立模型。基本上是最简单的神经网络架构之一——前馈技术。这意味着，输入值是具有任意数量的实值维度的向量，然后通过“隐藏层”进行处理，最终输出也是数字向量。此外，这类神经网络由许多参数组成，参数会在训练过程中自动更改，因此网络输出也能尽可能接近给定的输入-输出关系。

  必须考虑如何将与角色相关的复杂信息转换为矢量。某些信息是标量信息，例如维基百科中角色的排名或其关系数。

  其他信息可能是一组预定义的值，例如角色出现的剧集。因此，需要创建一个与剧集维度相同的向量，如果角色出现在相应的剧集中则将维度设置为 1.0，否则为 0.0。这样，可以将不同种类的信息转换为矢量，并且这些矢量仅相互影响。最后，有 1561 个书籍数据的输入维度和 411 个显示数据。以下是使用的数据类型👇

- 原著：性别、页数多少、亲属数量、年龄、文化、房子、房屋区域、忠诚度、角色所属的著作、地点、标题

-

- 剧集：性别、内容多少、亲属数量、年龄、忠诚度、角色出现的剧集、标题

- 一般来说，“年龄大小”依然是导致角色死亡最重要的因素;毕竟，年纪越大，之前所遭受的危险就越大！这就是为什么角色的当前年龄（如前所述的单热矢量）也是神经网络输入的一部分。因为神经网络输出只是将“存活百分比”确定为 0 和 1 之间的数字，所以可以为单个角色创建大约 90 个不同的输入向量，例如可能的年龄就会有一个。如果该角色在该年龄仍然存活，则神经网络将为该输入向量预测 1.0，否则为 0.0。

总结一下，让我们看一下有关预测和神经网络的统计数据。首先，权游原著维基百科共包含 484 个可用的角色，其中 188 个用于训练（即已经死亡），剩下的 296 个还活着的角色创建了预测。最后，训练准确率达到了 88.75％，而最终验证准确率为 89.92％。类似地，可以从剧集维基百科提取 146 个可用角色，82 个用于训练，64 个用于预测。这里的最终训练准确率为 79.64％，最终验证准确率为 85.69％。

- END -