

Apollo自动驾驶入门课程第⑥讲 — 预测

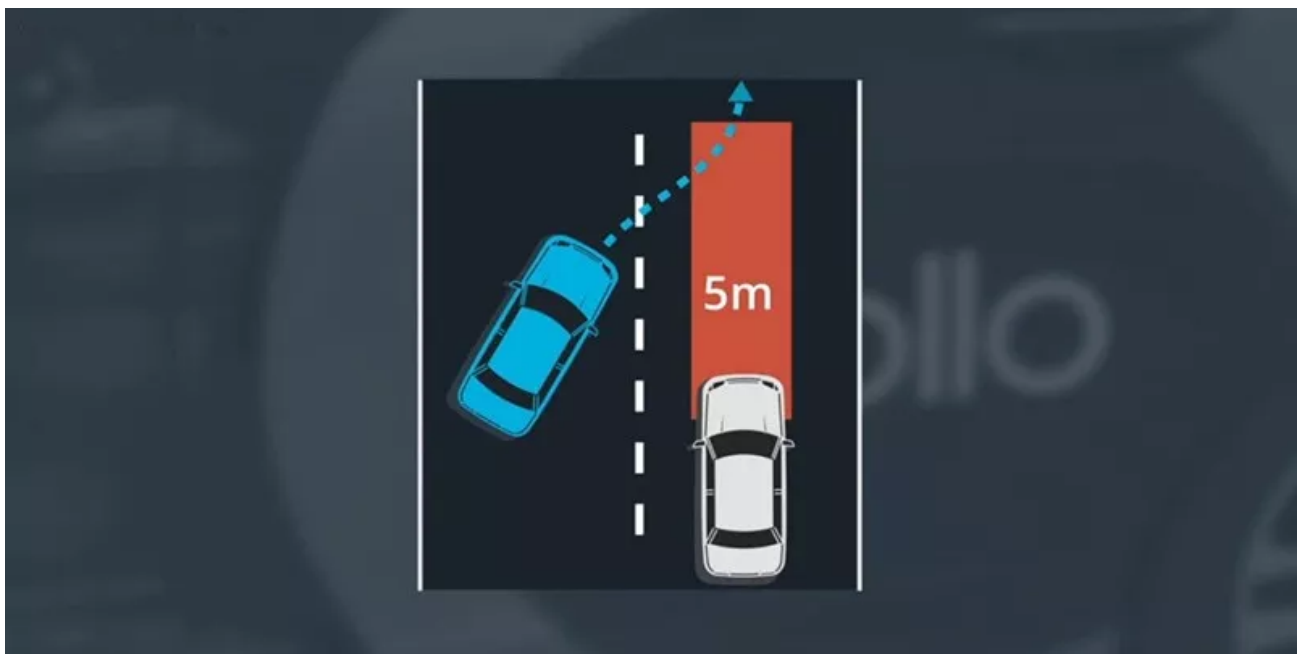


第六课，预测

1 | 简介

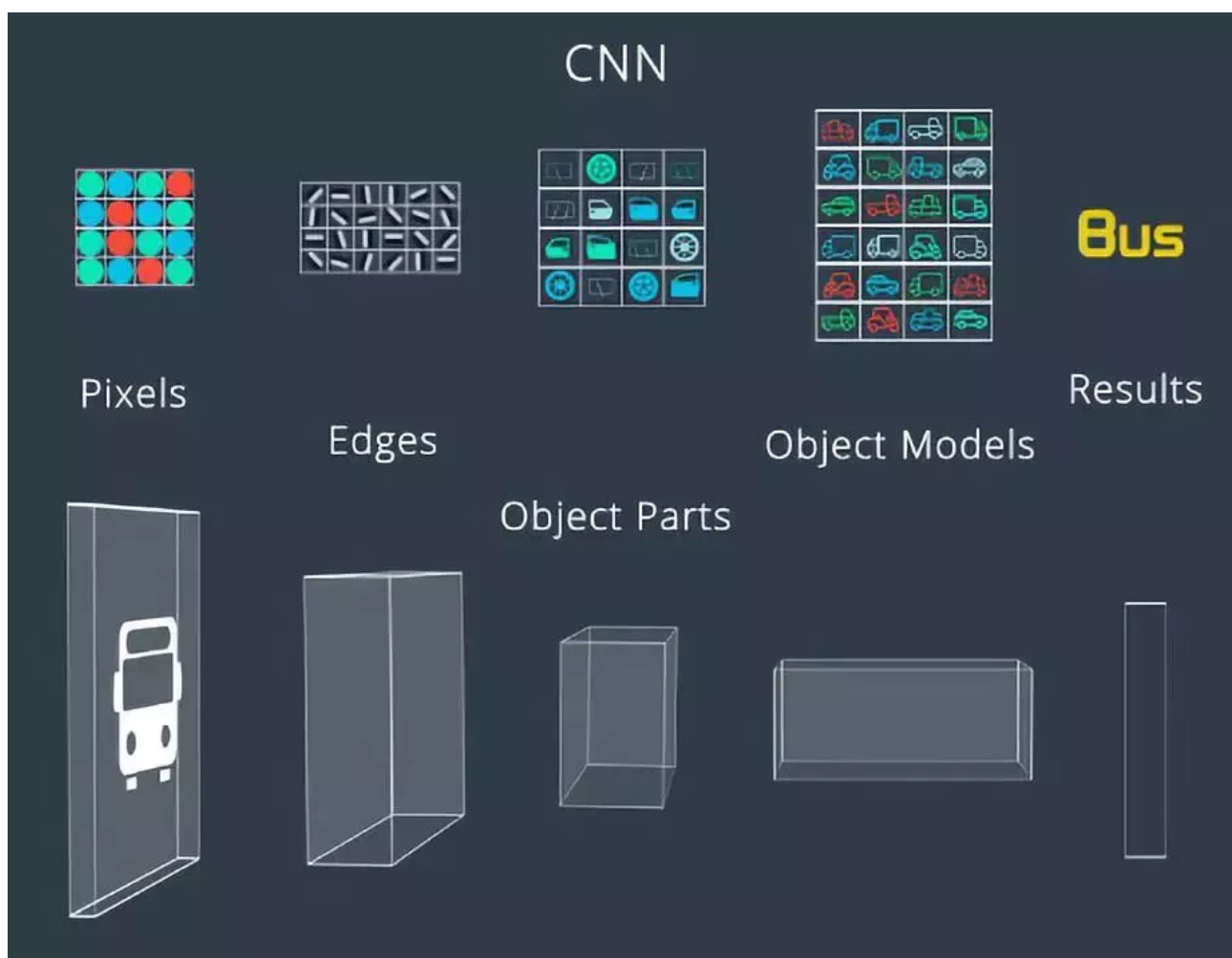
无人车是在许多物体间穿梭行驶，其中许多物体本身就是一直在移动的，比如像其他汽车、自行车、行人。无人车需要预测这些物体的行为，这样才能确保做出最佳决策。在环境中为所有其他的物体做出预测，这些共同形成了在一段时间内的预测路径，在每一个时间段内会为每一辆汽车重新计算预测他们新生成的路径，这些预测路径为无人车在规划阶段做出决策提供了必要信息。

预测路径有**实时性**的要求，实时性是指算法的延迟越短越好，一辆车如果是 60 千米/小时速度，那么它在每0.25 秒会行驶 5 米，所以需要确保无人车在行驶之前，前方5 米没有任何障碍物并且可以安全地穿行。



下一个目标是**准确性**。如果预测出相邻多车道的汽车想并入我们的车道，我们需要减速。而另一种情况是如果预测它会保持在自己的车道上行驶，我们需要做出的预测尽可能保持准确，这样才能帮助我们无人车做出很好的决策。

预测模块也应该能够学习新的行为，当路上有很多车辆，情况将变得复杂。开发出每种场景的静态模型是不可能完成的任务，所以预测模块能够学习新的行为，用这种方式可以使用多源的数据进行训练，使算法随着时间的推移而提升预测能力。

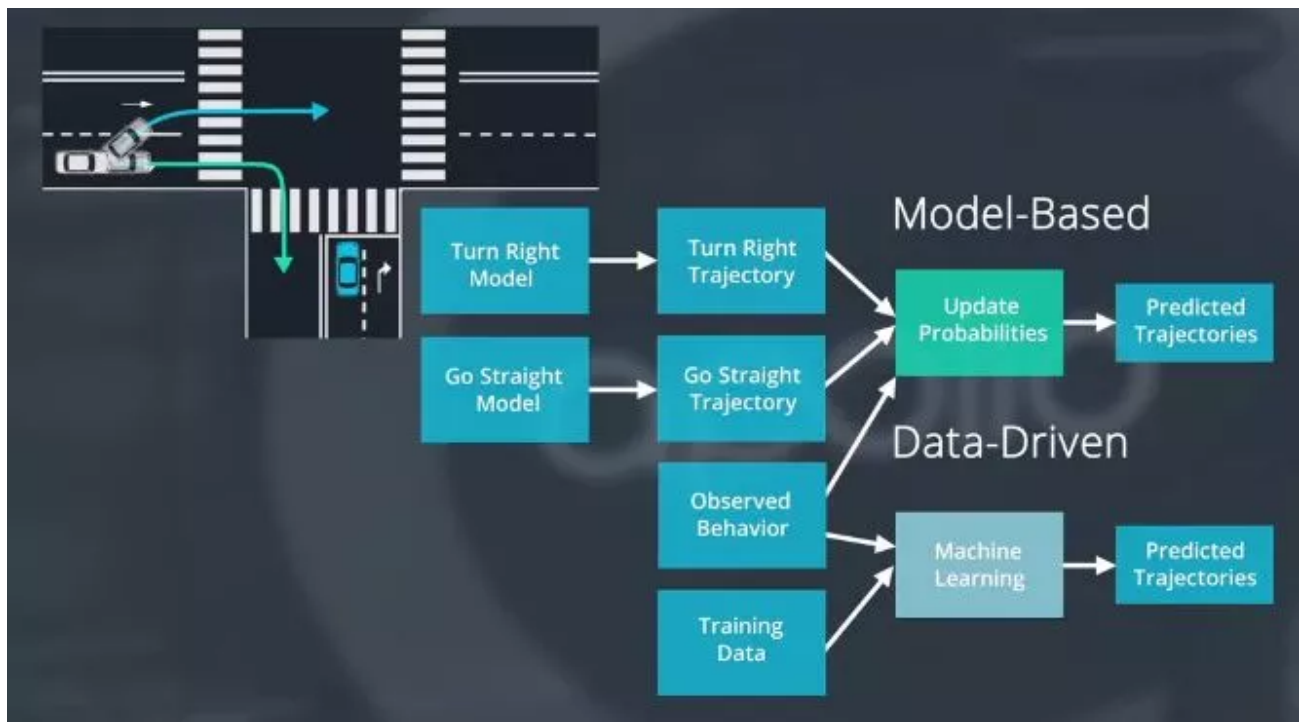


CNN根据其任务查找真正需要的特征，任务可能是图像检测、分类、分割、其他类型的目标。

2 | 不同的预测方式

我们将介绍两种不同的预测方式，分别基于**模型的预测**与**数据驱动预测**。

假设无人车来到一个T型路口并且看到一辆车从左面行驶而来，此时还不清楚这辆车是要右转还是直行，用基于模型的方法可以为此场景构建了两个候选的预测模型。一个模型描述了进行右转弯，用绿色轨迹表示，另一个模型描述了继续直行，用蓝色轨道表示。认为任意一种模式发生的概率都是相同的，所以有两个候选模型，每个模型都有自己的轨迹。继续观察移动车的运动，看它与哪一条轨迹更加匹配，如果看到车辆开始向左改变车道，我们会更加确信车辆最终会直行另一方面如果看到车在右转弯车道保持前行，我们会更加倾向于预测对车辆右转，这就是基于模型预测方法的工作原理。

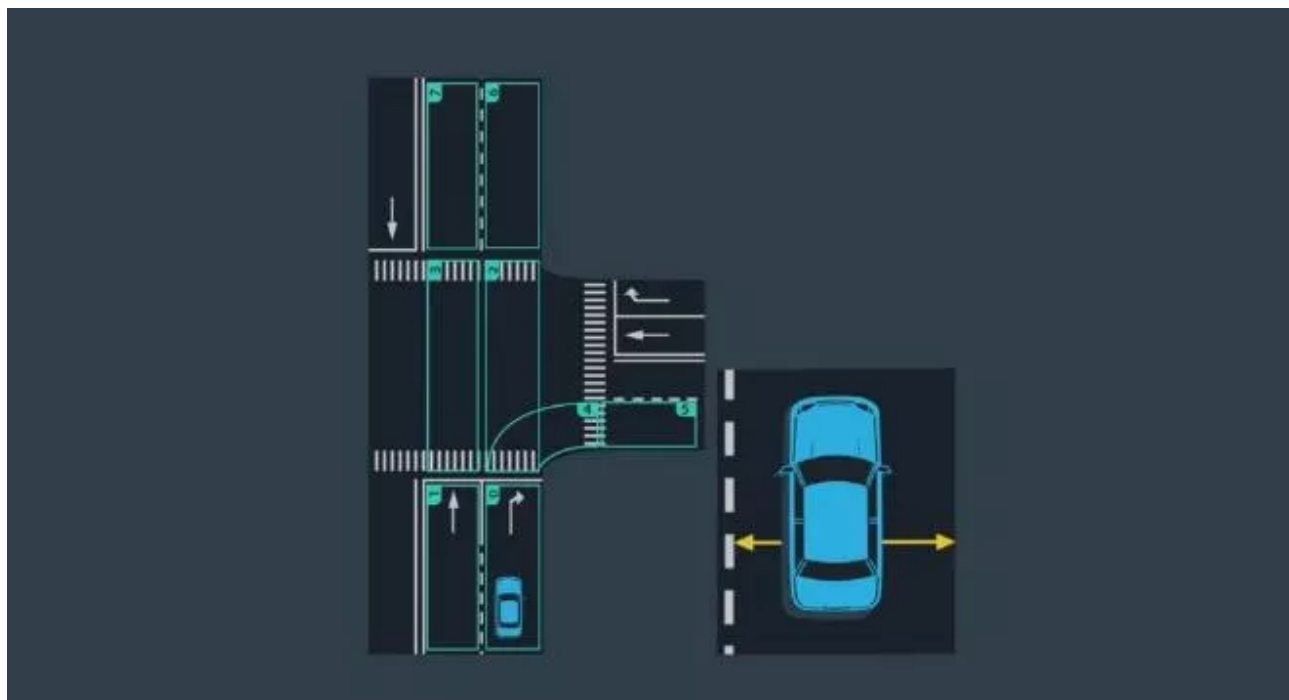


数据驱动预测使用机器学习算法，通过观察结果来训练模型，可以在现实世界中利用此模型去做出预测。数据驱动方法的优点是训练数据越多，模型效果越好。基于模型的方法的优点在于它的直观，并且结合了现有的物理知识以及交通法规还有人类行为多方面知识。

3 | 基于车道序列的预测

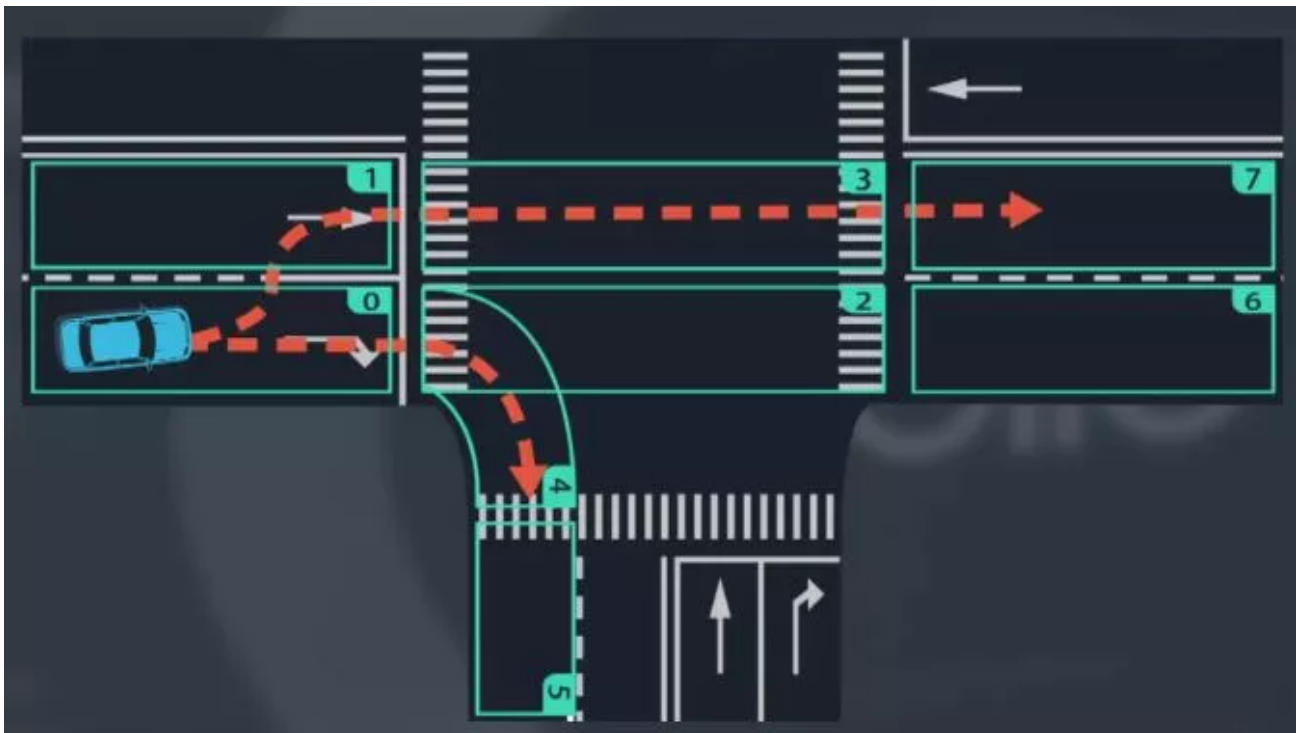
Apollo 提供了一种叫**基于车道序列**的方法，为了建立车道序列，首先将道路分成多个部分，每一部分都覆盖了一个易于描述车辆运动的区域。比如如图是一个部分区域的十字路口。为了预测，我们更关心车辆如何在这些区域内转换，而不是在某个区域内的具体行为。可以将车辆的行为划分为一组有限的模式组合并将这些模式组合描述为车道序列，例如直行汽车的运动可以描述为车道序列是 0-1-3-7。

这同样是一辆无人驾驶汽车如何观察物体的状态，除了位置、速度、朝向、加速度之外，无人车还需考虑车道段内物体的位置。例如预测模块会考虑从物体到车道线段边界的纵向和横向距离，还包含之前时间间隔的状态信息以便做出更准确的预测。

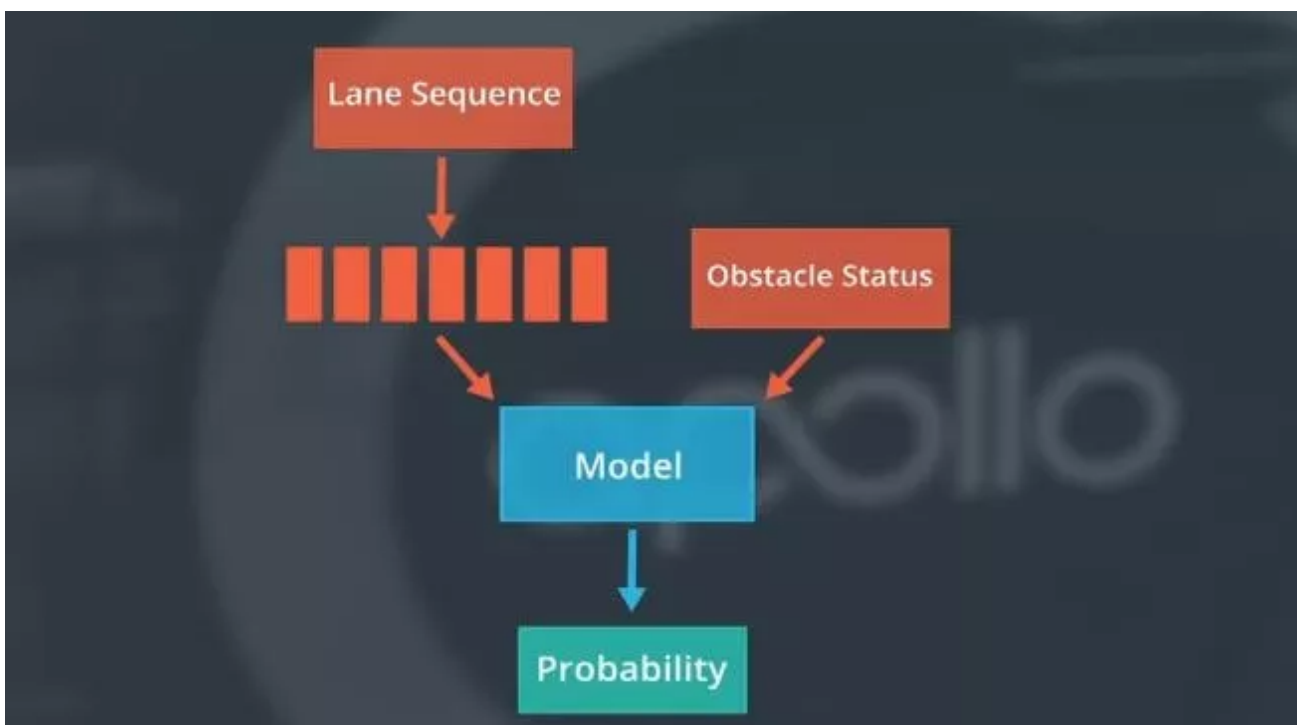


5 | 预测目标车道

使用车道序列框架的目标是为道路上的物体生成轨迹。先从一个稍微简单的问题开始，会预测车道线段之间的过渡。假设在车道段0中检测到一辆车，并且会预测在接下来的几个时间段中它将如何行驶，它可能停留在车道段0然后向右转或者可能转向车道段1然后直行，我们已经将预测问题简化为选择问题。



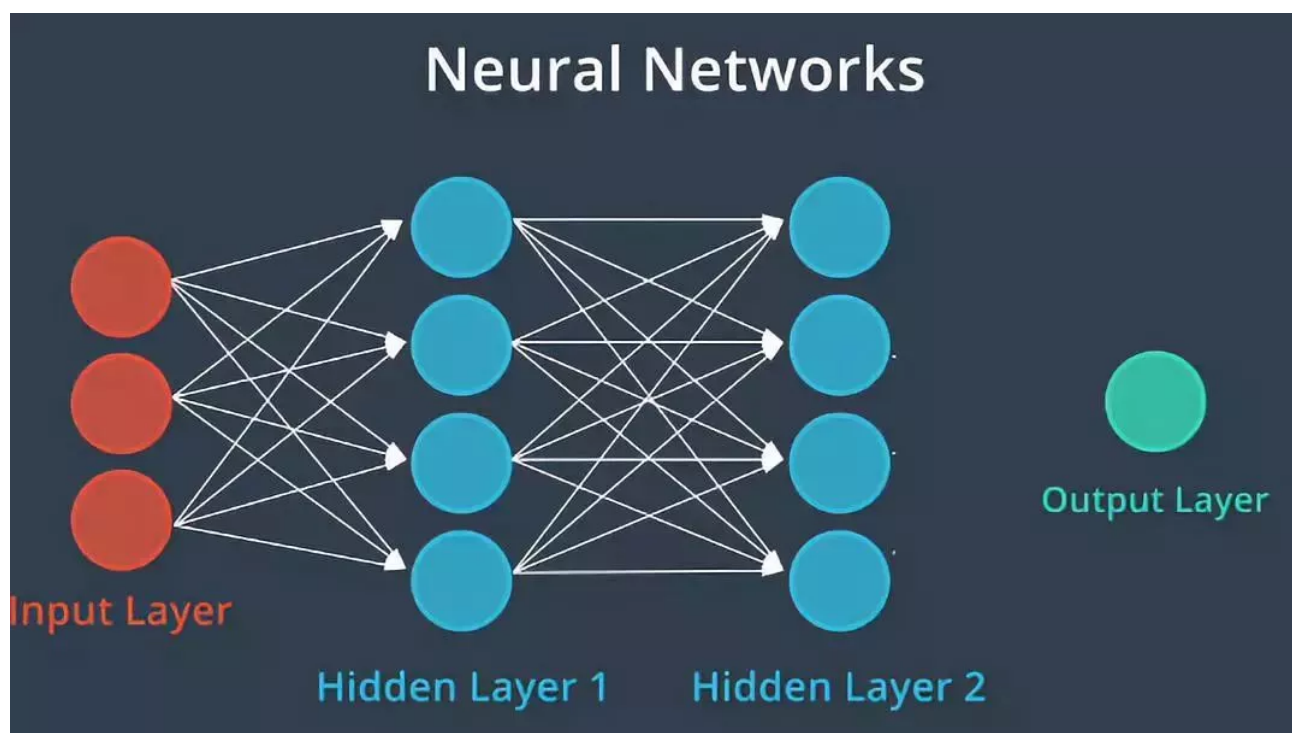
现在所要做的就是选择车辆最有可能采取的车道顺序，可以通过计算每个车道序列的概率来进行选择。将车辆状态和车道段作为输入，该模型用于提供车辆可能采用每个车道序列的概率，希望模型能够学习新的行为因此应该使用观测数据对模型进行经验性训练，在训练中将真实的车辆行为提供给模型，不仅包括车道段和对象的状态，还包括对象最终选择哪条车道序列。随着记录随着时间的增加，模型可以自我迭代更新，精确度不断提升。每个记录将由观察对象跟随的车道段序列和对象的相关状态组成，在每个时间点，对象占用一段并具有特定的状态，整个记录由一系列车道段和对象的相关状态组成。



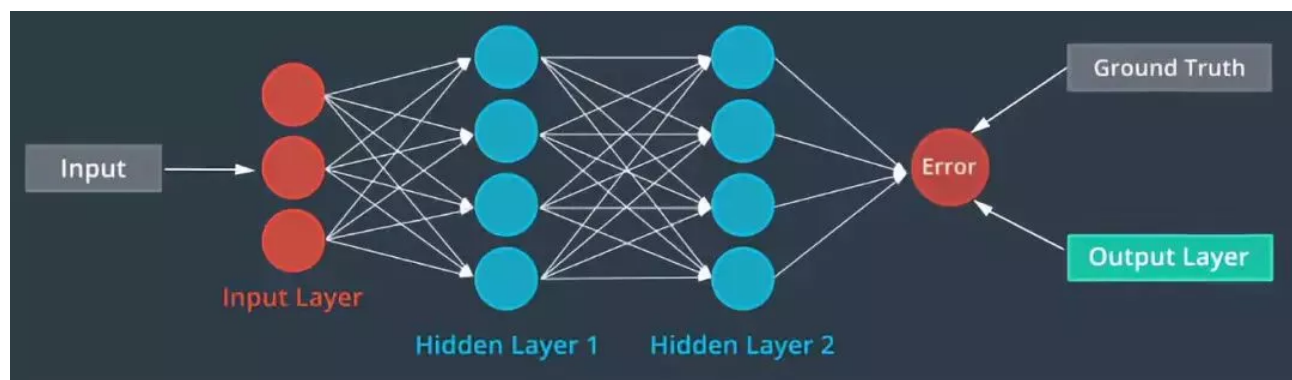
6 | 递归神经网络

递归神经网络RNN是一种利用时间序列数据特征的一种预测方法。

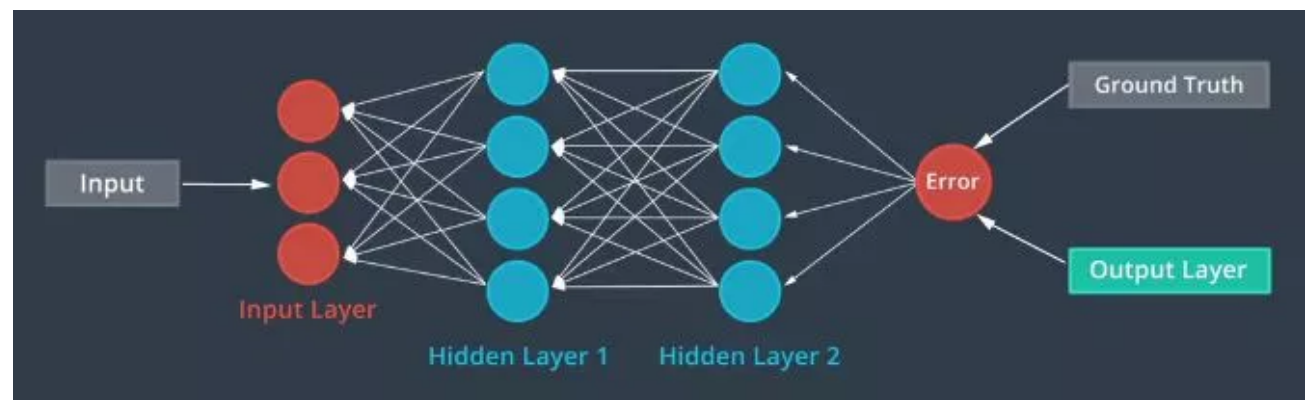
神经网络是可训练的多层模型，从输入提取高级特征并使用这些特征来计算得到输出。神经网络有许多结构，一个基本的神经网络首先得到输入，然后将数据通过隐藏层，然后经过处理得到输出，这种结果有时被称作多层感知网络MLP。



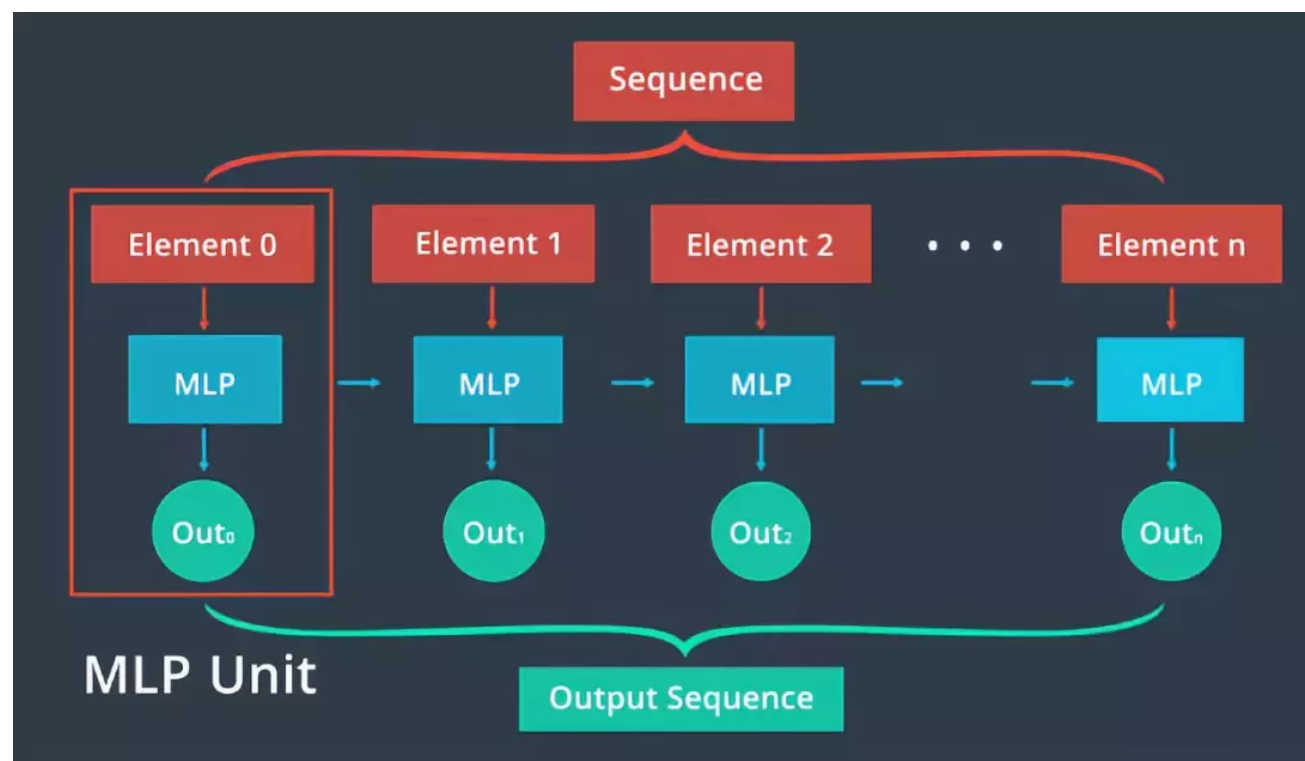
在训练的过程中会有很多训练数据输入模型，每一个数据都由原始的数据和对应的标签组成，例如输入数据是一张图片，而标签就是一个包含汽车的符号或者是其他符号。神经网络从数据中学习的方式叫做后向传播，首先神经网络得到输入并产生输出，然后计算机比较输出与真值之间的误差。



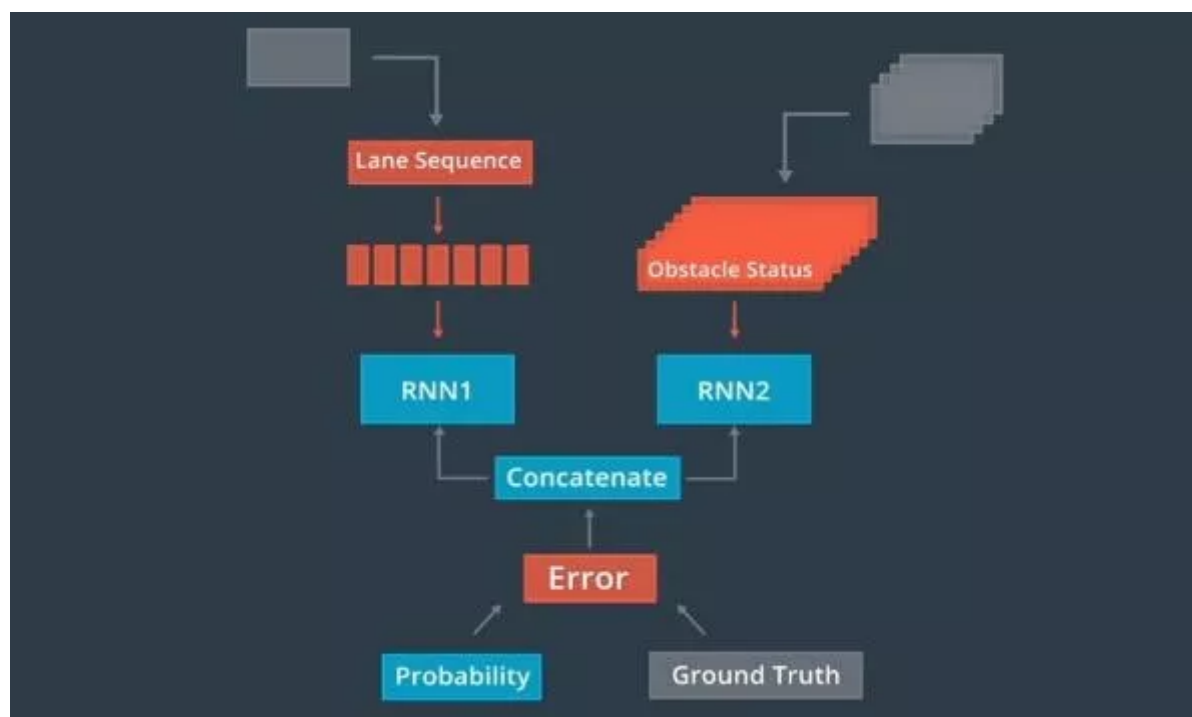
接着这种误差通过后向传回到整个网络，中间的隐藏层根据观察到的这种差别调整其中的中间值或者叫权重，这样可以在未来提高神经网络的准确率。



建立像这样的多重结构的递归神经网络称之为 **MLP单元**，从数据序列中提取出高级特征，每个MLP单元将序列的一个元素作为输入并预测序列的下一个元素作为输出。为了对元素之间的顺序关系建立模型，在每个单元之间建立一个额外的连接，这意味着每个单元根据原始输入和前一个单元的输进行预测，这 RNN的基本结构。



Apollo使用RNN建立一个模型来预测车辆的目标车道，为车道序列提供一个RNN模型，为相关对象状态提供另一个RNN模型。连接这两个RNN的输出并将它们馈送到另一个神经网络，该神经网络会估计每个车道序列的概率，具有最高概率的车道序列是我们预测目标车辆将遵循的序列。为了训练这个网络，使用现有的记录，每条记录都包含一个车道序列、相关的对象状态、一个标签，用于指示对象是否遵循此特定车道序列。在训练中，比较网络输出和真值标记并使用反向传播来训练网络。



8 | 轨迹生成

轨迹生成是预测的最后一步，一旦我们预测到物体的车道序列，就可以预测物体的轨迹。在任何两点A和B之间，物体的行进轨迹有无限的可能。



如何预测最有可能的轨迹？可以先通过设置约束条件来去除大部分候选轨迹，首先假定汽车将与目标车道的中心对齐，继续去除车辆无法实际执行的轨迹。通过考虑车辆当前的速度和加速度从剩余的轨迹中进行选择，实际上我们并没有实际列出所有可能的轨迹并逐一去除它们。

相反只是在数学理论上来应用这一想法，注意车辆在两点的位置和方位，这两个姿势表示运动模型的初始状态和最终状态，可以使用这两个条件来拟合一个多项式模型，在大多数情况下这种多项式足以进行预测。