Anyway，反正不是boosting decision tree就是linear regression (including logistic, SVM)。不会是NN的。   
很多工程师都是从FB跑过去，去帮忙搭架构。

其实他们的问题不需要最优化所谓的match。我打车只在乎价格，不看司机评分。至于carpool 的match，最短路径问题？他们技术上的突破应该是在于利用地图数据定位客户和司机然后找一定范围内的有效司机。。

作者：曾博  
链接：https://www.zhihu.com/question/41043058/answer/89864395

对出租司机而言，单子分两种，好单和差单。如同 答案所说，市区去机场就是典型的好单子，而距离不长但十分拥堵的区域则通常是差单。滴米加入的目的在于调动更多的司机接差单，毕竟每一个打不上车的单子背后都有一个心焦的用户。于是很自然的，好单扣滴米，差单得滴米。

账户上有了滴米之后，滴米在司机实际抢单时会换算成（负）距离。本来判定抢单的方式很简单，哪个司机近就归谁。但在加入滴米机制后，如果一个单子两个司机抢了，A比B稍微远一些，但是通过滴米的折算之后A就可能比B更近，这时候会判定A抢到单子。

另外滴滴为了防止有人game the system，好像有个简单的规定：滴米值在-200到200之间且定期清零。

那么我们来看第二种情况，如果A司机离的近，B司机离的远，系统怎么派？

这就简单了，根据就近分配的原则，我们会把A司机分配给这个订单。嘿嘿~~，假设我们再把问题设置的更加实际一点，当订单发出时，B司机已经在线并空闲，但是A司机还没有出现（没有上线，或者还在送乘客），但再过1s，离得更近的A司机突然出现可被分单了，假设我们使用先到先得的贪心策略，那么B司机就会被分给这个订单，那就违背我们希望就近分单的目标了。所以看上去简单，但实际情况下，算法还需要变的更好一些，这个问题我们把它叫做派单中的时序问题，我们后面再来看怎么解决。