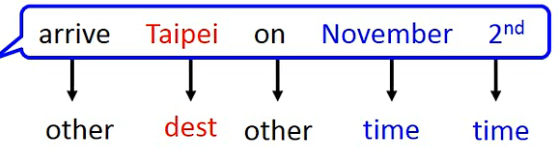
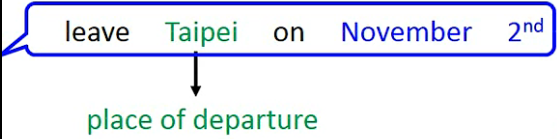
# 机器学习笔记

# Recurrent Neural Network(循环神经网络)

RNN的特点：对输入具有一定的记忆能力，输出与输入顺序有关。

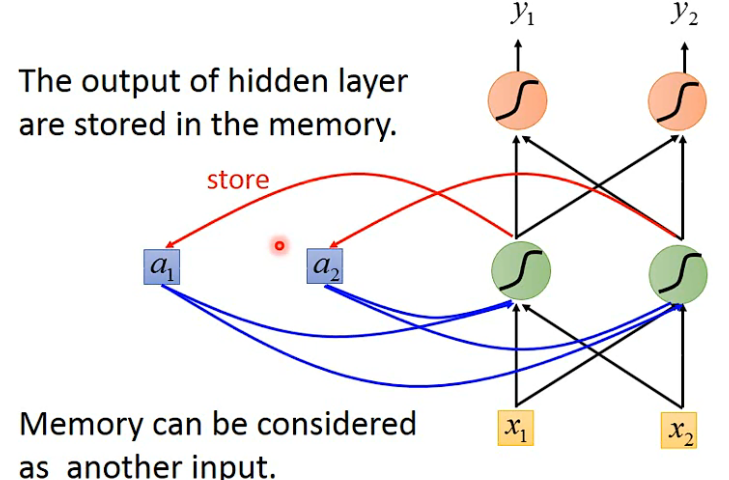
对于之前学过的神经网络来说，输入同样的数据，输出是相同的。

如上图，对于之前学过的神经网络，对于Taipei这个词的输入，它的输出要么都是destination，要么都是place of departure，无法做到根据上下文来判断Taipei这个词汇是属于目的地还是属于出发地。

而使用RNN就可以让我们的神经网络具有一定的记忆力，从而解决这个问题。

RNN的做法如图所示，

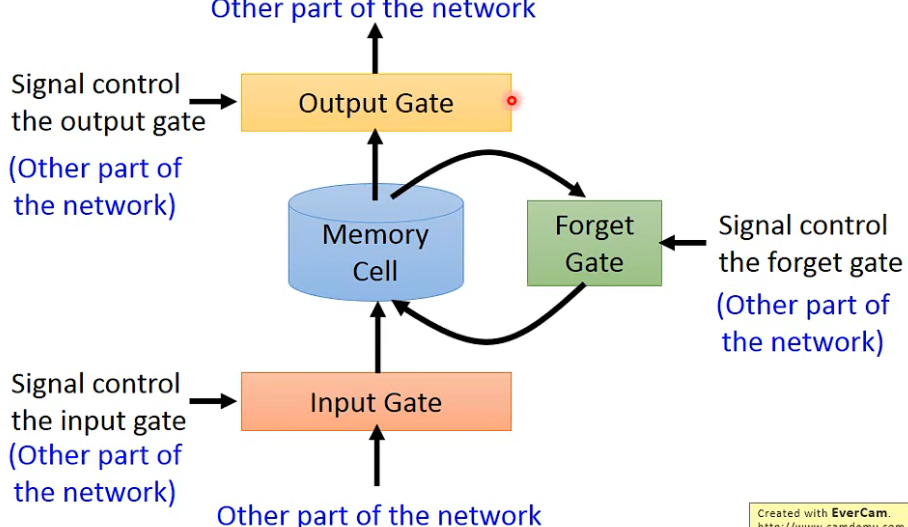


将上一次的输入存到一个用于存储的地方，而下一次输入时，输出不仅仅考虑这次的输入，还同时会考虑被存进记忆区里面的数据。根据这个输出考虑的因素我们可以知道，同样的数据给定不同的输入顺序，RNN得到的输出有可能是不同的。

RNN的结构是由我们自己设计的，我们可以让他是deep的，给他若干个hidden layer。

RNN也可以是双向的，比如一个句子我们默认是从句首读到句尾，但其实我们也可以从句尾读到句首，而这两种方向的RNN是可以同时进行training的。

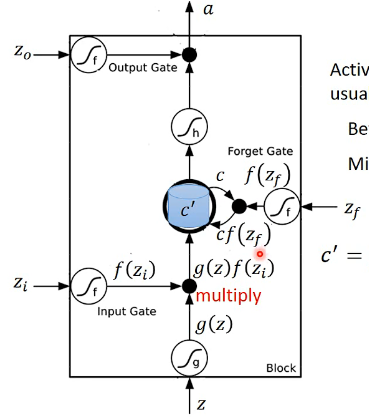
现在RNN最常用的memory称为，Long Short-term Memory(LSTM)，其结构如图所示，



它拥有三个阀门，分别用来控制输入输出以及是否遗忘记忆区里面的信息，而这三个阀门又有其对应的控制标志，用于控制阀门的开关。

每个阀门的函数一般用σ()函数，对于输入z，经过σ()函数转化得到f(z)，用于表示每个阀门的打开程度。

如图所示，输入通过函数g()转化，与函数f()相乘，然后将其存入记忆区，而遗忘阀门的函数f()的值中，0表示阀门完全不打开，记忆内容被遗忘，1表示阀门完全打开，记忆内容保留。

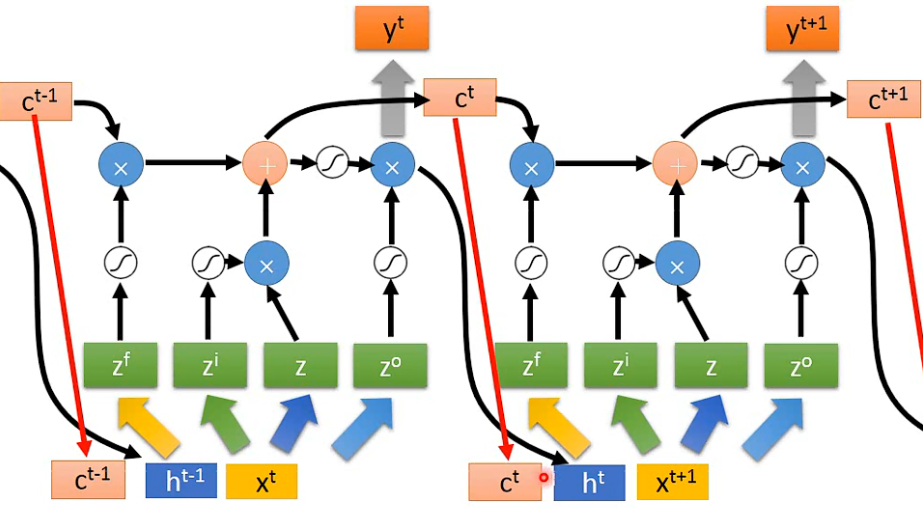


新的记忆内容就是输入内容和记忆内容之和，表示为

输出阀门和输入阀门类似。

LSTM和神经网络的关系，我们可以看成对于原来神经网络的每个神经，我们都将其换成LSTM cell。

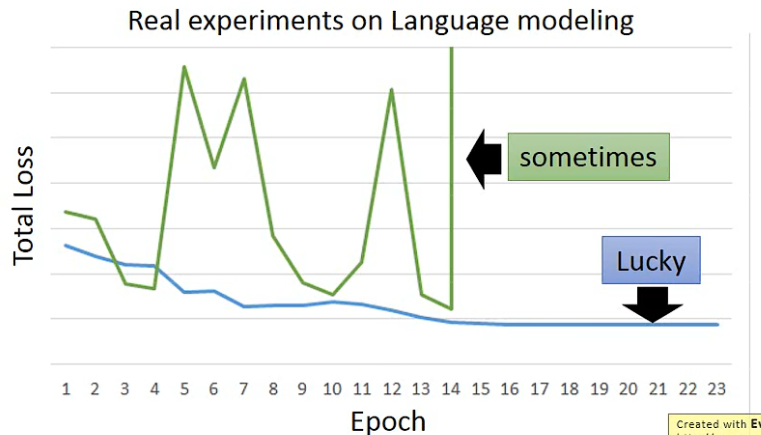
因为LSTM四个输入对应一个输出，所以LSTM需要的参数量是同等神经数量的别的神经网络的4倍。



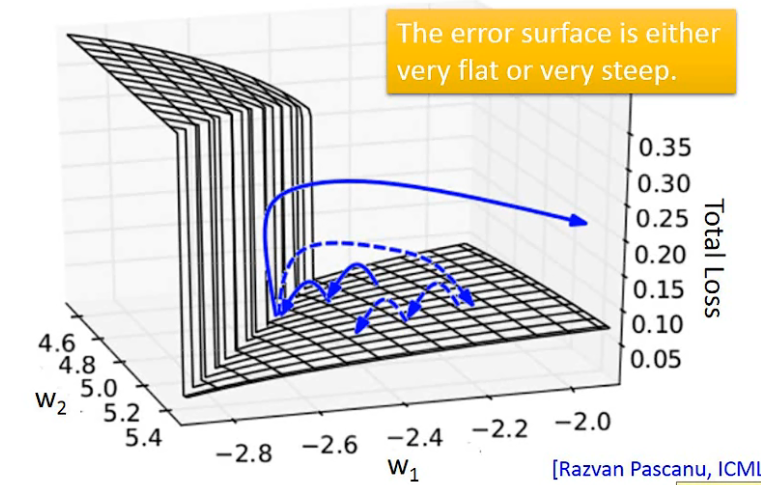
所有的LSTM cell是可以共同运算的，如上图所示，其实每组输入不仅考虑当前实际输入，还考虑上一次的输出以及存在记忆区里面的值。

# RNN的一些问题

在做机器学习的时候，我们通常希望随着学习次数的增加，函数的total loss不断地减少，但是在做RNN的时候，我们很可能会遇到下图所示的问题。



产生这种问题的原因是error surface有可能很平坦也有可能很陡峭，如下图所示，



在“悬崖”边缘的点时，我们做gradient descent得到的值就会很大，就会导致我们走的距离特别大，跑出最优值所在的地方。

而LSTM就可以解决梯度消失的问题，在LSTM里面我们让其只有在memory cell里面的值清除的时候才能将新的值存进去。

# RNN的应用场景

RNN可以应用的地方有很多，可以用于自然语言处理、声音识别、语言翻译等等场景上。