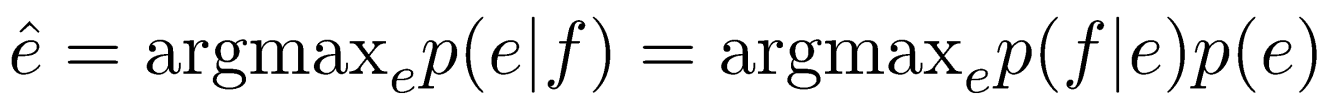
# 机器翻译

## 传统模型

给定源语言f（French）和目标语言E（English），并且现在有关于这俩中语言的平行的语料库，我们的目标是在给定源语言f的情况下，求得使下式成立的e



该公式可以通过贝叶斯定理推得。

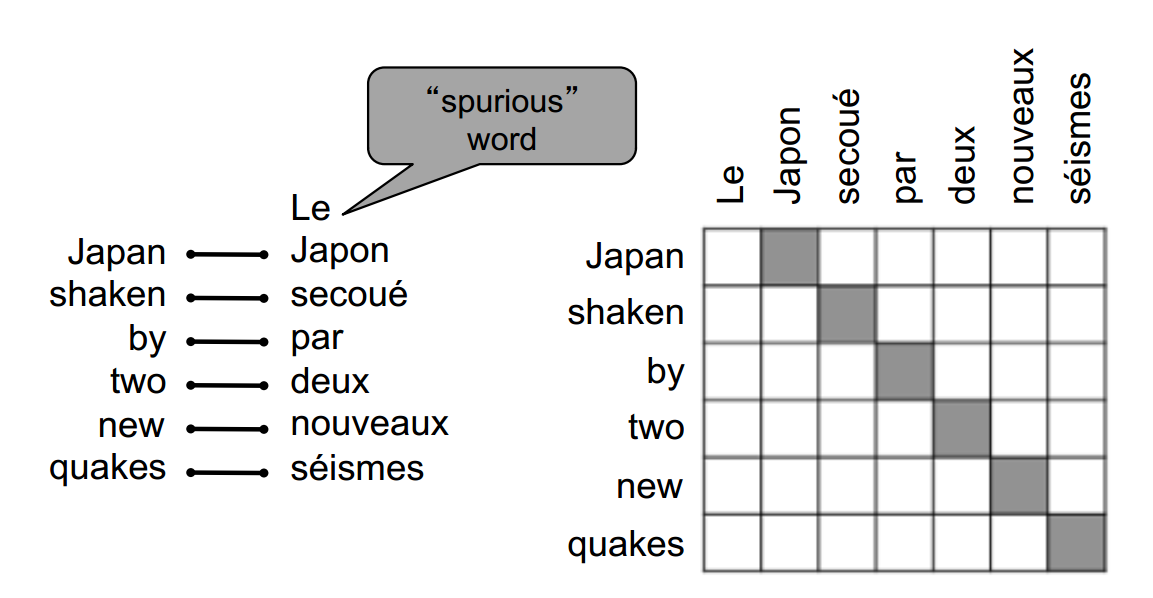
模型中的可以通过平行预料库统计获得，p(e)可以直接在仅包含英语的语料库上统计获得

## 

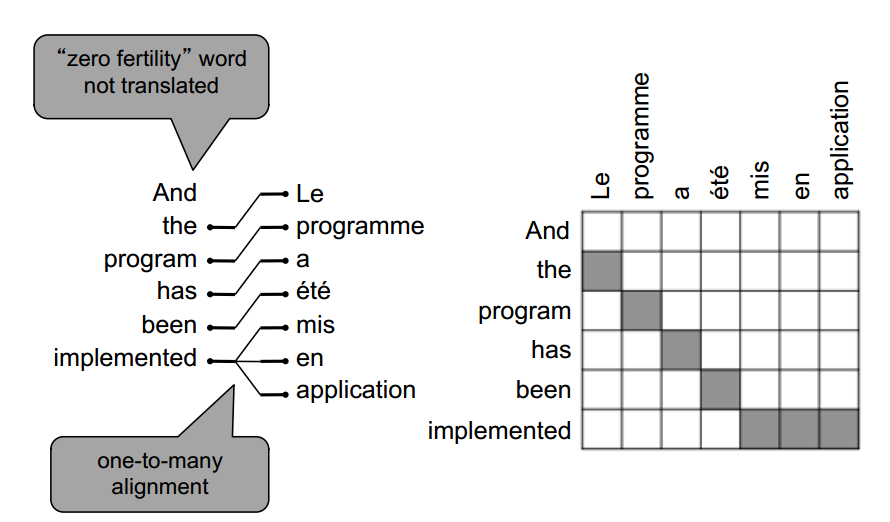
模型已经建立好了，但是现在还有一个问题，源语言中的一个词或者一个短语该翻译成目标语言中的哪个词或者哪个单词，通常来说，这是较为困难的一部分

在对齐过程中，会出现以下一些问题：

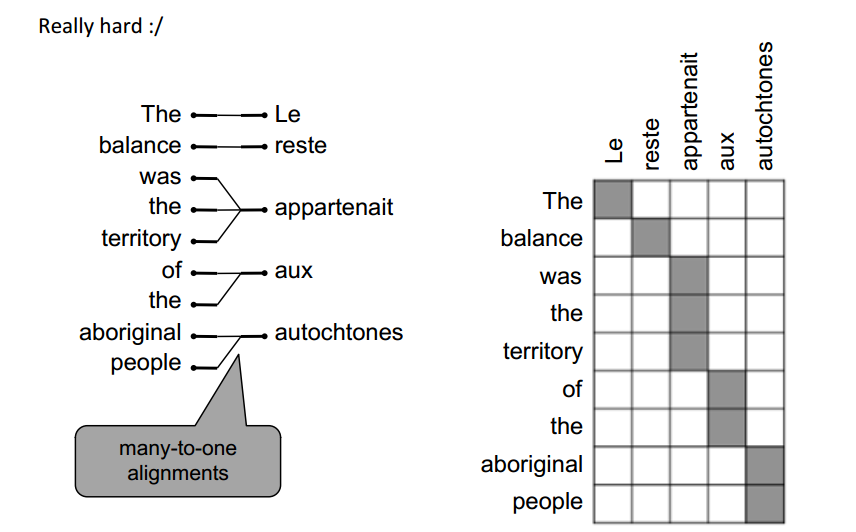
1“虚假”词



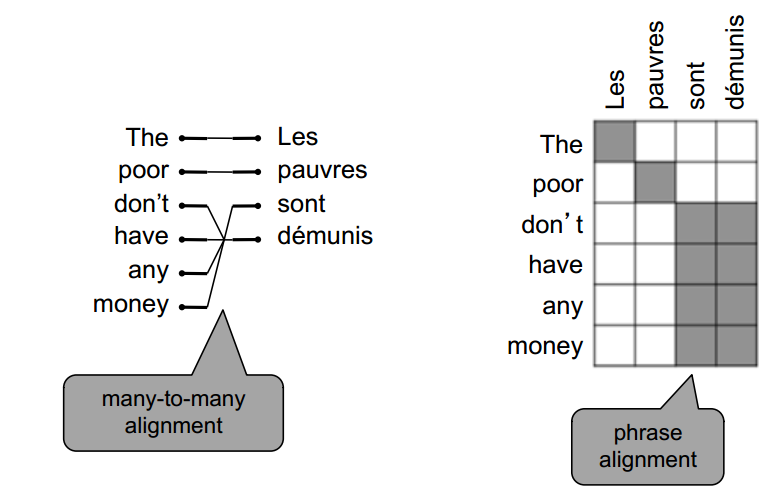
2没有对应的词和一对多的情况

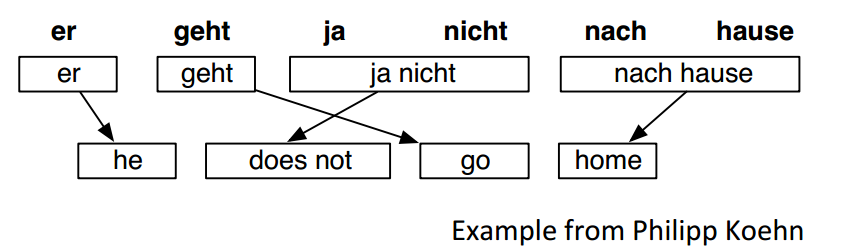


3多对一的情况



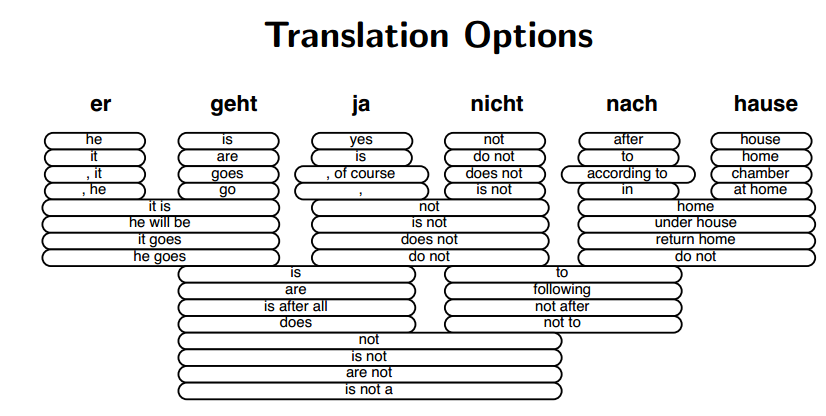
4多对多的情况



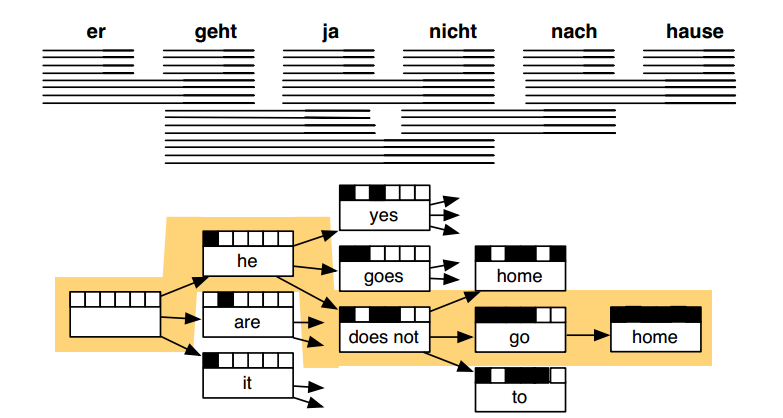
在对齐过程中，我们不仅仅可以考虑使用一个词去对齐，也可以使用一个短语去对齐，考虑完对齐的问题之后，还需要考虑目标语言的排序问题、

举个栗子：

在很多步之后，每个源语言中的短语可能在目标语言中对应着多种可能的翻译



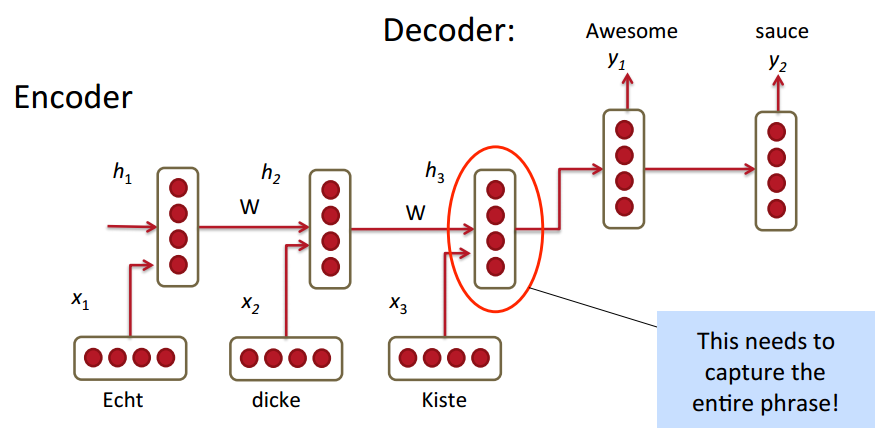
接下要做的就是解码：在众多假设中寻求最合理的那个



已上是传统翻译模型的简介，略过了很多重要的细节，这个模型还需要很多人工的特征，是一个非常复杂的模型，还有很多不同的独立的机器学习问题。

## 基于RNN的机器翻译

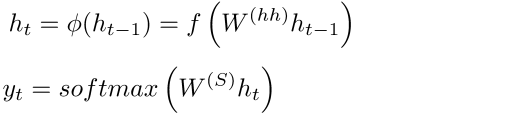
这种的模型的思想是将源语言短语构成的词序列喂食到一个RNN之中，最终，用一个向量来捕捉整个短语的特征，这个过程称之为编码（Encoder），之后再将这个向量喂食到下一层RNN之中，通过第二层RNN来预测目标语言的“下一个词”，这个过程称之为解码（Decoder），一个简单的事例图如下：



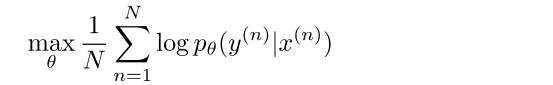
编码：



解码：

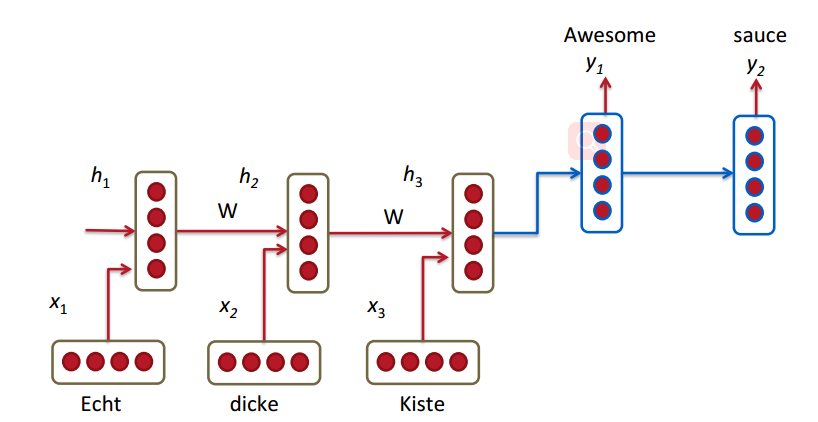


损失函数：



但是，由于简单的RNN网络存在梯度消失和梯度爆炸的问题（见lec7），RNN并不能捕捉到很早之前的单词，因此提出了以下的一些改进方式：

1. 为编码层和解码层分别训练不同的权重矩阵

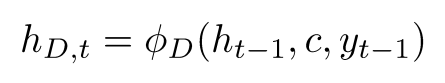


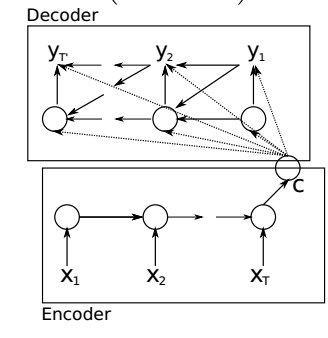
1. 在解码层中，每个神经元的输入来自三个部分

前一个省神经元的预测输出

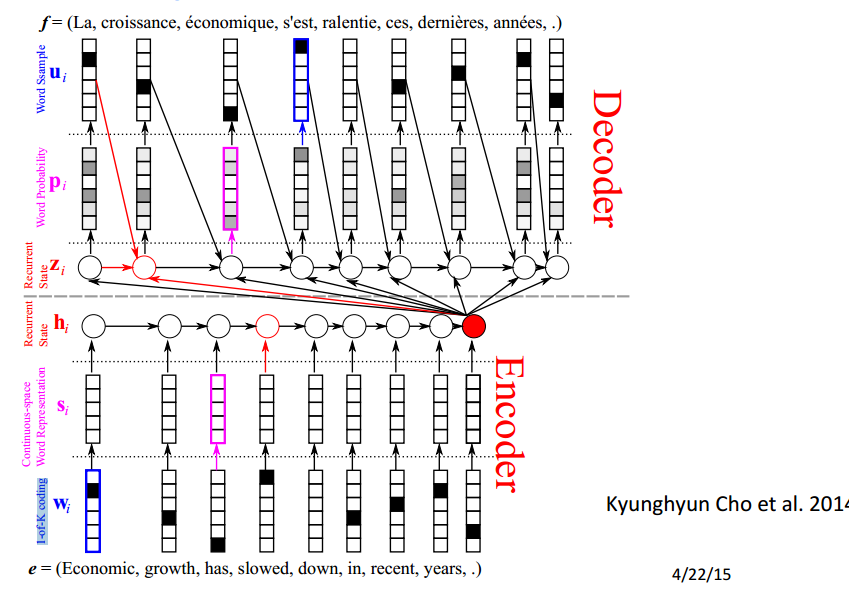
编码层中最后一个神经元的状态

前一个神经元的状态

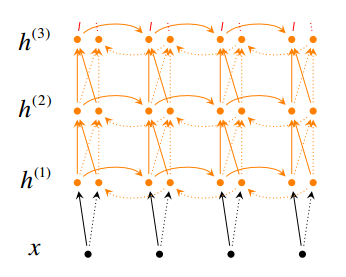




一个复杂一点图（原理一样）



1. 使用含有多层的深度RNN
2. 使用bidirectional encoder



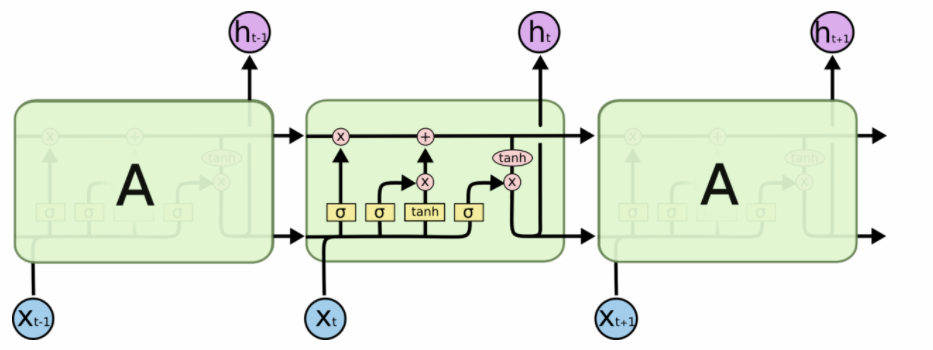
1. 对于简单的优化问题，将源短语反序输入，使用CBA→XY而不是ABC→XY，这是因为通过反序输入，网络能够捕获到短语中靠前的词
2. 主要的提升方法：使用更好的神经元

更复杂的神经元

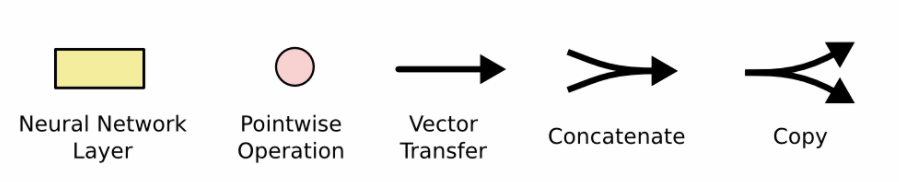
使用GRU，这种单元可以让模型学习何时遗忘从而将记忆保持很久、允许误差根据输入的不同而不同。

使用LSTM

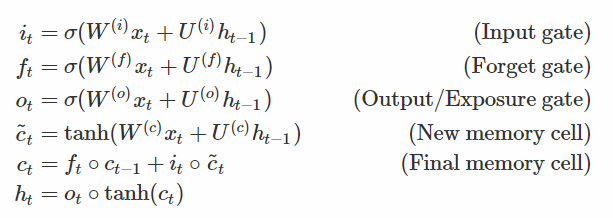
可视化的LSTM网络图：



其中，图中各个图标的含义如下：



由于之前已经介绍过LSTM模型，所以现在直接给出相关公式



GRU模型，它将忘记门和输入门合成了一个单一的 更新门。同样还混合了细胞状态和隐藏状态，和其他一些改动。最终的模型比标准的 LSTM 模型要简单

