**# Seq2Seq模型**

**1. 引言**

**1.1 Seq2Seq的由来**

* 在 Seq2Seq 框架提出之前，深度神经网络在图像分类等问题上取得了非常好的效果。在其擅长解决的问题中，**输入和输出通常都可以表示为固定长度的向量**，如果长度稍有变化，会使用**补零**等操作。(如最原始CNN的输入是图像经过flatten后的向量，下图红框部分，往往将数据处理成固定大小的向量作为输入)

图示

描述已自动生成

* 然而许多重要的问题，例如机器翻译、语音识别、自动对话等，表示成序列后，**其长度事先并不知道**。因此如何突破先前深度神经网络的局限，使其可以适应这些场景，成为了13年以来的研究热点，Seq2Seq框架应运而生。
* 当输入是**多个向量**，而且这个**输入向量的数目是会改变的场景**时，Decoder的输出可以有以下三种形式

1. 输出个数与输入向量个数相同，即每一个向量都有对应的一个label或value (如命名实体识别NER、词性标注POS tagging等任务)，也叫Sequence Labeling。

图示

描述已自动生成

1. 只需要输出一个Label或value (比如文本分类、情感分析)。

图示

描述已自动生成

1. 输出个数与输入向量个数不一定相同，机器要自己决定应该要输出多少个Label或value (比如文本翻译、语音识别)，也叫做**Sequence to Sequence(Seq2Seq)**的任务。

文本, 聊天或短信

描述已自动生成

**1.2 Seq2Seq和Encoder-Decoder的关系**

* Seq2Seq 使用的具体方法基本都属于Encoder-Decoder 模型（强调方法）的范畴，Seq2Seq（强调目的）不特指具体方法，满足"输入序列、输出序列"的目的，都可以统称为 Seq2Seq 模型。
* Seq2Seq模型是基于Encoder-Decoder框架设计的，用于解决序列到序列问题的模型。

**2. Seq2Seq模型**

* Sequence-to-Sequence (Seq2Seq) 模型，其输入是一个序列，输出也是一个序列。**其最重要的地方在于输入序列和输出序列的长度是可变的。** 最基础的Seq2Seq模型包含了三个部分，即Encoder、Decoder以及连接两者的中间语义向量，Encoder通过学习输入，将其编码成一个固定大小的向量，继而将传给Decoder，Decoder再通过对状态向量的学习来进行输出。
* 常见结构:

图示

描述已自动生成

图示

描述已自动生成

图示, 示意图

描述已自动生成

* Seq2Seq模型缺点

Seq2Seq模型缺点包括了RNN模块存在的缺点，和基础Encoder-Decoder框架存在的问题

1. 中间语义向量无法完全表达整个输入序列的信息。
2. 中间语义向量对输出所产生的贡献都是一样的，即分配到的权重是相同的。
3. 随着输入信息长度的增加，先前编码好的信息会被后来的信息覆盖，丢失很多信息。