【笔记】搜索引擎中的Query改写Keyword技术研究

rzhangpku 关注
0.69 2018.04.24 00:41:08 字数 1,339 阅读 2,484

技术文章链接：https://mp.weixin.qq.com/s/aW5NaF6-SqJXpkO687XdAw
发表于微信公众号：360搜索实验室
以下是对360搜索实验室发表的这篇技术文章的阅读笔记。

背景

搜索引擎对关键词形式的query返回结果好，而对一般自然语言形式的query返回结果差。需要将一般自然语言形式的query转化成关键词形式的query，其实是从一句话中提取关键词。

关键词提取

抽取方式

1. 分词
2. 计算词语的重要程度，计算方式有基于tf-idf和基于TextRank的
3. 按照词语的重要程度排序，挑选top n个词语作为关键词

基于tf-idf

tf：词频
idf：词的区分能力
tf*idf：词的重要性*，tf*idf高，则选为关键词。按照tf*idf排序，挑选top n个词语作为关键词。

基于TextRank

借鉴PageRank
在k长度窗口中词的相邻关系来得到PageRank的链接指向关系。所以如果一个词V_i在这个k长度窗口中只出现一次，则只有一个词（w的前一个词）指向V_i，它也只指向一个词（V_i的后一个词）。但是如果V_i在这个k长度窗口出现多次，或者V_i在其他的k长度窗口也出现了，则会有多个词指向V_i，V_i也会指向多个词。
迭代公式如下：

$$WS(V_i) = (1-d) + d * \sum_{V_j \in In(V_i)} \frac{w_{ji}}{\sum_{V_k \in Out(V_j)} w_{jk}} WS(V_j)$$

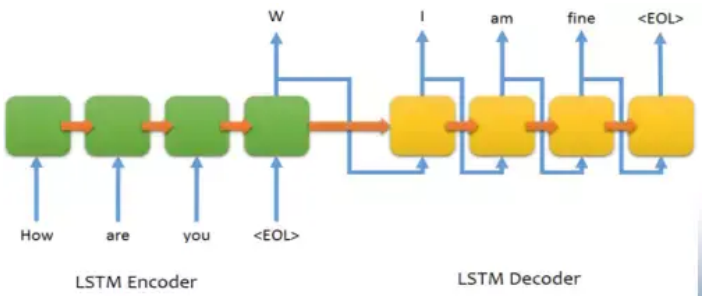
迭代公式.png

WS(V_j)表示词V_j的重要性；d是阻尼系数，决定TextRank算法一次能影响多少；In(V_i)是指向该词V_i的集合；Out(V_j)该词V_j指向的词的集合；w_ji表示词V_j指向词V_i的链接的权重。公式的计算结果得到WS(V_i)词V_i的重要性

生成方式

1. 理解用户原始query
2. 生成与用户原始query意思一致的关键词
   利用深度学习文本生成技术来进行关键词抽取

seq2seq结构



seq2seq结构.png

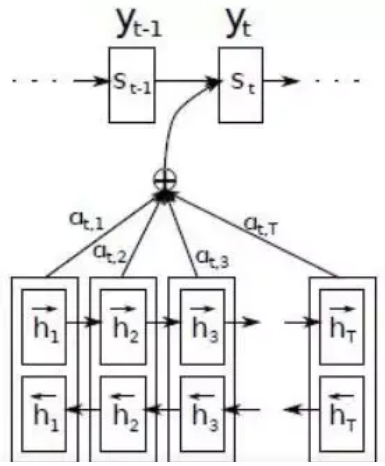
seq2seq结构包括：
（1）encoder：将可变长度的序列的信息（以<EOL>作为输入序列的结束标志）存放在一个固定长度的向量里
（2）decoder：将encoder得到的固定长度向量的信息解码成可变长度的序列（以<EOL>作为输出序列的结束标志）
nmt机器翻译模型就是采用了这种seq2seq结构。

seq2seq结构的局限性

编码和解码之间的联系只有一个固定长度的向量。encoder要将整个序列的信息都压缩到一个固定长度的向量，很有可能整个序列的信息无法都压缩到这个向量里，而且即使是LSTM做为encoder，仍然无法记住很久之前的信息，这个固定长度的向量只能保留少部分先输入的信息。

seq2seq结构的改进：attention based seq2seq

attention其实是个矩阵，表示输出时需要重点关注输入的哪些部分。
attention based seq2seq结构如下：



推荐阅读
浅谈智能搜索和对话式OS
阅读 11,109
机器之心翻译-GNMT开源教程
阅读 4,568
论文笔记：Attention is all you need
阅读 47,149
Summarization概述
阅读 1,565
中国新四大发明之共享单车去"世界旅行"，在美国、德国遭遇"水土不...
阅读 174

attention矩阵计算公式为：

$$\alpha_{ij} = \frac{exp(e_{ij})}{\sum_{k=1}^{T_x} exp(e_{ik})}$$
$$e_{ij} = a(s_{i-1}, h_j)$$

attention矩阵计算公式.png

e_ij评估了第j个输入h_j与第i个输出s_{i-1}的match程度；a_ij相关于是e_ij的归一化结果，其实就是e_ij的softmax值，得到0-1的值a_ij，a_ij仍然是衡量第j个输入h_j与第i个输出s_{i-1}的match程度。

attention based seq2seq运用到机器翻译任务时，attention也被称为对齐模型，比如"今天天气真好"翻译完"今天"之后， 注意力就会在"天气"上，考虑应该将"天气"翻译成什么词。相当于将当前翻译的词与新生成的词进行对齐。

attention based seq2seq广泛应用于机器翻译，文本摘要和智能问答等任务，但对于文本摘要，关键词提取等任务，其decoder部分仍然有很大的提升空间。

attention based seq2seq问题

问题一：

OOV（Out of Vocabulary）问题

decoder产生的词只能是来自 训练数据分词得到的词汇表。测试时，测试集target句子中的词有可能没有在训练数据中出现，则decoder无法生成这些词；测试集source句子中的词有可能没有在训练数据中出现，则encoder不认识这些词，无法对这些词进行编码，会直接将其认为是unknown "<unk>"，而直接输出到target预测结果中，并保持在source中的位置。

问题一的解决方法：pointer network

利用attention矩阵的softmax分布作为pointer指针，指针指向的输入中的词作为输出，实际上是平衡了"抽取"（指针直接指向重要的词）和"生成"两种方式的优点。

问题二：

decoder过度依赖其输入，也就是先前的总结词，会导致一个词的出现触发无尽的重复。

问题二的解决方法：采用coverage机制

对前期注意力覆盖的词进行惩罚，防止用过的词再被使用。

综合两种解决方法：Pointer-Generator Network

推荐阅读

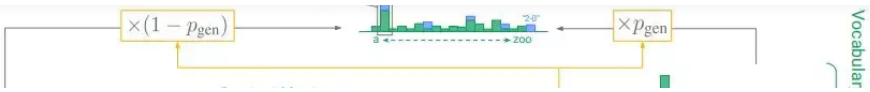
浅谈智能搜索和对话式OS
阅读 11,109

机器之心翻译-GNMT开源教程
阅读 4,568

论文笔记：Attention is all you need
阅读 47,149

Summarization概述
阅读 1,565

中国新四大发明之共享单车去"世界旅行"，在美国、德国遭遇"水土不...
阅读 174



Pointer-Generator Network.png

将输入中词的概率和词表中词的概率做一个加权和

$$P(w) = p_{\text{gen}} P_{\text{vocab}}(w) + (1 - p_{\text{gen}}) \sum_{i:w_i=w} a_i^t$$

image.png

当词语不在词表中，则利用输入中的attention分值进行选取。能有效解决OOV问题。
对于重复问题，维护一个coverage向量，记录之前所有attention和

$$c^t = \sum_{t'=0}^{t-1} a^{t'}$$

image.png

在loss里加入这一项，这样可以对之前考虑过的词进行惩罚，防止重复。

结果对比

生成方式的训练数据：qt对，query和点击的URL的title。将title作为输入，query作为输出。

QUERY	抽取式	生成式
清谈鸡汤怎么做?	清炖，鸡汤，怎么	清炖，鸡汤，做法
怎么可以去青春痘	青春痘，怎么，可以	怎么，去，青春痘
微信的作用是什么	微信，作用，什么	微信，作用

结果对比.png

结论

生成方式要好于抽取方式