关于句子embedding的一些工作简介（一）

原创 M 没啥深度 2018-07-19

对于NLP方面的工作来讲，毫无疑问词嵌入(word embedding)是最重要的基石。然而人们往往最关心的是如何利用词嵌入表达好一个句子或者一段话，也就是如何找到sentence embedding, 或者paragraph embedding。

过去的一些解决方案

Bag of Words

最直观的想法是忽略句子里词语的顺序，从而把它看做bag of words 。
比如简单的对所有词语取平均。好处是计算速度较快，但是缺点是它忽略了词序，在一些对于词语顺序比较敏感的任务中，比如情感分析(sentiment analysis), 效果不佳。

DAN

其实DAN(Deep Averaging Networks)应该属于Bag of Words类的算法。因为比较特殊，单独列出来。 它是在对所有词语取平均后，在上面加上几层神经网络。特殊的地方在于它在sentiment analysis中表现也不错，这在BOW类方法中比较罕见。

RNN

很显然利用RNN(GRU或者LSTM)是一种不错的解决方案，它完全克服了BOW中忽略语序的缺点。但是它往往和supervised task结合在一起，缺乏可扩展性或者说迁移性(transferrable)，在某个task中可以取得不错的成绩，但是遇到其他的问题就得重新进行训练。LSTM往往开销比较大，而且不适合GPU的并行处理。

Paragraph2vec

这种算法直接把句子或者段落作为一个向量放入词语的context中，利用word2vec中的CBOW方法进行训练。优点是属于unsupervised learning，可以用在任意task中，缺点是inference时间长。

Skip-thought vectors

首先利用LSTM对句子进行编码，然后利用编码的结果去预测周围的句子。形式上有点像word2vec的skip-gram算法。它也是非监督学习，可迁移性好。
从机器学习的类型上看，

方法	类型
BOW	无监督
DAN	监督
RNN	监督
Paragraph2vec	无监督
Skip-thought	无监督

新的一些进展

SIF

文章声称找到一种简单且有效（simple but tough-to-beat basline）的基线模型。（算法被称为SIF：smooth inverse frequency)

方法的确很简单：每个句子先表达为所含词语embedding的加权平均；然后把句子放在一起找出最大主轴；最后从每个句子中移除掉这个最大主轴即可。

文章同时论证了采取subsampling的word2vec方法中的gradient也是周围词的加权平均，和SIF的权重非常相似。

实验结果上文章区分了textual similarity tasks和supervised tasks。前者的目的是预测两个句子之间的相似性（***Pearson's r***），后者包括SICK的entailment和SST（sentiment analysis)。 结果总结如下：

任务	**效果最好的模型** **(SIF vs DAN, LSTM, skip-thought)**
similarity	SIF（明显好）
entailment	SIF（稍好）
sentiment	LSTM

文章总结sentiment中表现不好的原因有二：一是SIF直接利用了word embedding的加权平均，而很多word embedding都有"antonym problem"（例如难以区分good和bad）二是SIF中downweigh了很多诸如"not"的词语，而"not"在sentiment analysis中是非常关键的词语。

InferSent

2017年非常重要的一篇文章，在很多NLP的任务中都取得了state-of-arts的成果。

论文基于这样的观察：在Computer Vision领域，人们可以通过在专门的数据集（ImageNet）上训练模型，然后把模型应用在各种其他的任务, 也就是人们常说的transfer learning。 那么在NLP领域，能否遵循同样的线路，训练出universal的sentence representations ?

文章成功的找到了NLP领域的ImageNet — SNLI (Stanford Natural Language Inference dataset), 并且试验了不同的深度学习模型，最终确定bi-LSTM max pooled 为最佳模型。

这篇文章很有意思，影响力也非常大，成为后续很多文章比较的对象。第二篇系列文章会专门介绍Infersent。

Concatenated p-mean Word Embeddings

这篇文章找到了一种简单而非常有效的句子编码方式（concatenated p-mean word embeddings)，基本思想简单讲就是把"求平均"这个操作generalized，在

embedding的每个维度上求平均值对应p=1, 求最大值对应p取值正无穷，求最小值对应p取值负无穷。作者对不同的word embedding（word2vec, glove， etc）求得*power means*以后简单的把这些vectors串起来，输入给logistic regression模型。效果上在monolingual领域超过了SIF, 只略输于InferSent，在cross lingual方面优于其他算法。 第三篇系列文章会专门介绍这篇论文。

Quick Thoughts

这篇文章和skip-thought很像，区别在于前者在编码之后有个解码过程，最终生成对周围句子的预测；后者在编码之后（同时也对其他句子编码）直接输出给一个classifier,去判断哪个句子是临近的句子，哪些不是。毫无疑问，算法上效率高很多。第四篇系列文章会专门介绍quick thoughts。

Universal Sentence Encoder

这篇文章基于InferSent， 也是想找到一个universal encoder。不同之处在于文章把InferSent的bi-lstm换成了DAN（或者Transformer)，而使用DAN这样“简单”的encoder的效果竟然相当好（尤其是时间和内存消耗和其他算法比小很多。）但是老实讲我越来越觉得今年来Google有点标题党之嫌。（Universal Sentence Encoder这名字起得太大，和去年的Attention is all you need一样霸气）

结语

总结一下，新的算法和过去的算法有如下关系:

新方法	监督	基于的旧算法	贡献
SIF	无	BOW	一个简单而有效的baseline算法
Infer-Sent	有	NA	找到了NLP领域的ImageNet – SNLI， 并给出了一个state-of-art 算法
P-mean	无	BOW	比SIF更简单且有效的一个算法且适用于cross-lingual
Quick-thought	有	Skip-thought	multi-channel版本的QT成为新的state-of-art
Universal-sentence-encoder	有	Infer-Sent	更加简单的encoder