**Efficient Estimation of Word Representations in**

**Vector Space（word2vec）**



**摘要**

我们提出了两个新颖的模型架构用来计算大规模数据集中的连续词向量表示。计算得出的结果通过词相似任务进行衡量。通过将这些结果和目前为止表现最好的基于不同类型的神经网络的方法进行对比后发现，该方法的精度得到大幅度提升，并且计算成本要小得多。举例来说，这个方法用了不到一天的时间从含有16亿个词的数字集中训练出了高质量的词向量。不仅如此，用这些词向量进行语法和语义上的词相似度任务均获得了最好的表现。

**1 简介**

当前很多的自然语言处理系统或技术都把词当做原子单元—词表示就是词在字典中的索引，这导致了在词和词之间不能描述相似度的问题。这样做是因为大规模数据集上应用简单模型得到的结果优于在小规模数据集上应用复杂模型的结果，具体表现在：简单性，鲁棒性，可观测性这三方面。一个具体的例子是广为流传的N元模型（N-gram），它用于统计语言模型中。今天，实际上它可能训练任何数据集（万亿级别的词）。

然而，简单模型在很多任务上都有限制。比如说：用于自动语音识别的相关域内数据量是有限的—结果往往由高质量转录的语音数据的规模决定（通常是百万级别的词）。在机器翻译中，各种各样现存的语料仅仅包含不超过十亿级别的词汇量。因此，有些情况下，简单改进基本技术不会导致任何重大进展，我们必须关注更先进的技术。

伴随着近些年机器翻译的进展，在更复杂更大规模的数据集上训练模型成为了可能，而且这些方法比简单的模型效果更好。或许用词的分布式表示是最成功的概念，比如说基于神经网络的语言模型明显地优于N元模型。

**1.1 论文的目标**

该论文的主要目的就是提出了一个学习高质量词向量的技术，这个技术能够应用在十亿级别的数据集，百万级别的词汇量上。据我们所知，之前没有方法能从几百万的词当中成功地学习50维到100维之间的词向量。

我们使用了最近提出的方法并测试其产生的向量的质量，我们不仅仅希望相似的词之间距离更近，而且词有多种相似度。这在曲折语言的上下文中出现过，举例来说，名词可以有不同的尾缀，如果我们在原始向量空间的子空间中搜索一个名词，那我们应该可以找到具有相似尾缀的名词。

令人意外的是，词的表示的相似度比简单的语法规则更复杂。使用词偏移技术，在词向量上执行简单的代数运算，一个典型的例子是：King的词向量减去Man的词向量加上Woman的词向量的结果最接近Queen的词向量。

在本文中，我们试图通过开发新的模型体系结构来最大化这些向量操作的准确性，以保留词之间的线性规律性。我们设计了一个新的综合测试集用于测量语法和语义规则，并且结果显示许多这样的规律可以高准确度地学习到。此外，我们讨论训练时间和精度如何依赖于词向量的维数和训练数据的数量。

**1.2 之前的工作**

把单词表示为连续向量拥有很长的历史；一个用于估计神经网语言模型的非常流行框架模型由Bengio提出，这个模型用一个带有线性投影层和非线性隐藏层的前馈神经网络训练得到词向量和一个统计语言模型。很多人追随这个工作继续深入。

**2 模型框架**

很多不同种类的模型提出后用来估计词的连续表示，包括著名的潜在语义分析 (LSA)和潜在狄利克雷分配(LDA)。在本篇论文中，我们把重点放在通过神经网络学习到的词的分布式表示上，就像之前展示的，它能存储词与词之间的线性规则从而使得性能明显优于LSA，同时，与LDA相比，LDA在大规模数据集上的计算代价很大。

为比较不同的模型框架，首先我们定义模型的计算复杂度为为充分训练模型所需要的参数的数量。接着，我们在最小化计算复杂度的同时最大化模型的准确度。对于接下来提出的所有模型，训练复杂度正比于：

**O=E×T×Q (1)**

其中，E是指训练的迭代次数，T是训练集中词的数量，Q将在之后的模型框架中具体定义。E的一般选择范围是3-50，T可达到十亿量级，所有模型均使用随机梯度下降和反向传播进行训练。

**2.1 前馈神经网络语言模型 (NNLM)**

概率化的前馈神经网络语言模型由Bengio 提出，该模型由输入层，投影层，隐藏层，输出层构成。在输入层，前N个词使用 独热(one-hot)编码,V是指词典中词的数量；接着输入层被投影到到投影层P,该投影层的维度是N×D，共享同一个的投影矩阵，因为在任何给定时间只有N个输入有效，所以投影层的组成是性价比相对较高的操作。

神经网络语言模型架构在投影层和隐藏层之间的计算变得复杂，因为投影层上的值是稠密的。一个通常的选择是N取10,那么投影层可能包含500-2000个数，与此同时，隐藏层通常有500-1000个数，而且隐藏层用来对词典中的所有词计算概率分布的，这导致输出层含有V个数（V是词典中词的个数），所以对于每一个训练样本来说模型的计算复杂度为：

**Q =N×D+N×D×H+H×V (2)**

其中，对计算复杂度影响最大的为H×V这一项。为避免计算该项，Mnih等人提出了一些实际的解决方案，比如用softmax的分层版本，或者通过使用在训练过程中没有归一化的模型来完全避免归一化模型。通过将词典构造成二叉树形式，需要被估计的输出单元的数量可以下降到log2(V)，这样处理的话，对计算复杂度影响最大的就变成了N×D×H这一项。

在我们的模型中，我们使用的分层的softmax形式，词典被构造成哈夫曼树的形式，这延续了我们之前的工作：在神经网络语言模型中词频对于获取类别很有效果。在哈夫曼树上，对于高频词赋予短的二进制编码，这进一步减少了需要被估计的输出单元的数量。虽然平衡二叉树需要评估log2(V)数量的输出单元，但是基于哈夫曼树的分层softmax结构只需要评估大约log2(V的一元困惑度)数量的输出单元。举例来说，当词典含有一百万个词时，使用该结构会使评估速度大约提升两倍，但是对于神经网络语言模型来说这并不是主要的提速，因为计算的瓶颈在于N×D×H这项，之后我们将提出不包括隐藏层的结构，这样模型的计算复杂度主要取决于softmax归一化的效率。

### 2.2 循环神经网络语言模型（RNNLM）

基于循环神经网络的语言模型被用于突破前馈神经网络语言模型的一些局限性，例如前馈神经网络语言模型需要指定上下文的长度，因为理论上比起浅层神经网络，循环神经网络可以有效地表征更复杂的模式，循环神经网络模型没有投影层，只有输入层，隐藏层、输出层。使用时间延迟连接将隐藏层和自身连接的循环矩阵是这类模型的特殊之处，这允许循环模型形成某种短期记忆，因为来自过去的信息可以由隐藏层状态表示，隐藏层状态根据当前输入和上一时间中隐藏层的状态得到更新。循环神经网络在训练每个例子时的复杂度为:

**Q=H×H+H×V (3)**

这里，词的表示D和隐含层状态H拥有相同的维度，这样，项H×V能够通过使用层次softmax降低至H×log2(V)，复杂度主要取决于H×H。

### 2.3 神经网络的并行训练

为了训练庞大的数据集上的模型，我们在称为DistBelief的大规模分布式框架之上实现了几个模型，包括本文提出的前馈神经网络语言模型和本论文提出的新模型。该框架允许我们并行地运行同一模型的多个副本，并且每个副本都通过保留所有参数的中央服务器同步地进行梯度更新。对于这种并行训练，我们使用称为Adagrad的自适应学习速率程序使用小批量且异步的梯度下降。在此框架下，通常使用100个或更多的模型副本，每个副本在数据中心的不同机器上使用多个CPU内核。

## 3 新对数线性模型

在本节中，我们提出了两种新的模型体系结构，用于学习试图最小化计算复杂性的词的分布式表示。由前一节的主要结果可知，大部分复杂性是由模型中的非线性隐藏层引起的。虽然这正是使得神经网络如此有吸引力的地方，但我们决定探索可能无法像神经网络那样精确地表示数据的较简单模型，不过可以对更多数据进行有效地训练。

新架构直接遵循我们早期工作中提出的架构，我们发现神经网络语言模型可以分两步成功训练：首先，使用简单模型学习连续词向量，然后使用N元神经网络语言模型在这些分布式词表示上进行训练。虽然后来大量的工作重点学习词向量，但我们认为我们之前提出的方法是最简单的。注意，相关模型也早已由Hinton等人提出过。

### 3.1 连续词袋模型

首先提出的架构类似于前馈神经网络语言模型，其中非线性隐藏层被移除并且投影层由所有词共享（不仅仅是投影矩阵），因此所有词都被投影到相同的位置（它们的向量被平均）。 我们称这种架构为词袋模型，因为词的顺序不影响投影。

此外，我们还使用前面的文字，通过用某个词前面四个词和后面四个词作为输入建立一个对数线性分类器，我们在接下来将要提到的任务里到达了最优的表现，其中训练标准是正确分类当前的词(中间的词)，那么训练的复杂度为：

**Q= N×D+D×log2(V) (4)**

我们把这个模型称为CBOW，不同于标准的词袋模型，它使用上下文的连续的分布表示。这个模型框架如图1所示，注意输入层与投影层之间的权重矩阵由所有位置的词共享，和神经网络语言模型类似。

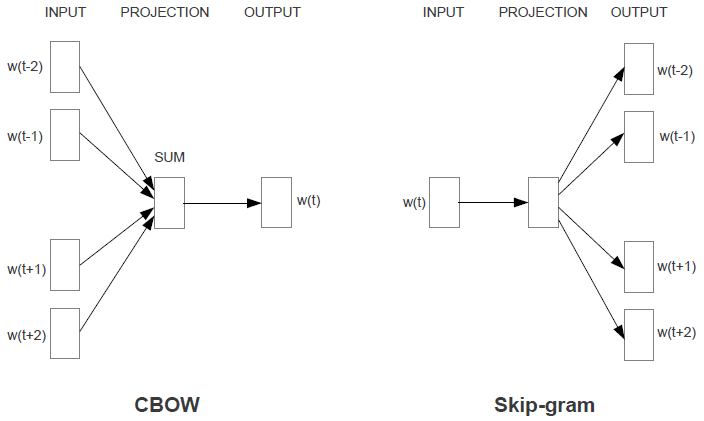
### 3.2 连续的Skip-gram模型

第二个框架与CBOW相似，但是不再基于上下文预测当前词了，该模型试图优化一个基于同一个句子中的其他词的词分类器；更精确的说，我们使用当前词作为带有连续的投影层的对数线性分类器的输入，然后预测当前词之前和之后一定范围内的词。我们发现增加范围能够提升词向量的质量，但同时也增加了计算复杂度。因为距离当前词远的词通常不如距离它近的词更相关，所以在我们的训练集中通过采样更少的那些距离当前词远的词从而赋给它们更小的权重。

这个框架的训练复杂度为:

**Q=C×（D+D×log2(V)) (5)**

这里，C是词与词之间的最大距离，所以如果我们取C=5，对于每个训练中的词，我们将随机地从1到C中选择一个数字R，然后使用当前词的前R个词和后R个词作为当前词的正确标签。这需要我们对R×2个单词做分类，分类器的输入为当前词，当前词的前R个词和后R个词作为输出，在下面的实验中，我们取C = 10。

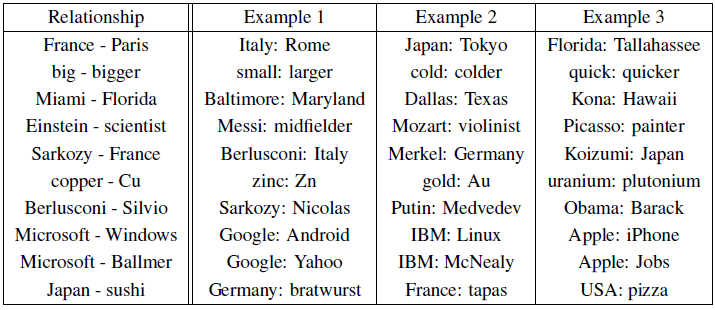


图一：新模型架构，CBOW架构根据上下文预测当前词，Skip-gram架构根据当前词预测上下文

## 4 学习到的关系示例

表8显示了遵循各种关系的词。我们遵循上面提到的方法：通过减去两个单词向量来定义关系，并将结果添加到另一个单词中。例如，巴黎 - 法国 + 意大利 = 罗马。可以看出，虽然显然还有很大的进一步改进空间，但准确性还是相当不错的（请注意，使用我们的准确度度量假设完全匹配，表8中的结果只会得到约60％的分数）。我们相信在更大维数的更大数据集上训练的单词向量将表现得更好，并且可以开发新的创新应用程序。另一种提高准确性的方法是提供多个关系的例子。通过使用十个例子而不是一个来形成关系向量（我们将各个向量平均到一起），我们发现，在语义语法测试中，我们的最佳模型的准确性提高了约10％。

表8 使用来自表4的最佳单词向量的单词对关系的示例



也可以应用矢量运算来解决不同的任务。例如，通过计算单词列表的平均向量并找到最远的单词向量，我们已经观察到了选择不在列表单词的良好准确性。在某些人类智力测试中，这是一种流行的问题。显然，通过使用这些技术仍然有很多东西等待我们发现。

## 5 结果

为了比较不同词向量版本的质量，之前的论文通常会使用表格展示样本词和与它最相似的词，在直观上体会词向量的质量。虽然展示法国这个词向量类似于意大利这个词向量(或者其它国家的词向量)很简单，但是将词向量应用于更复杂的相似度任务中更具挑战性。我们知道有很多种词与词之间的相似度类型，比如说，在一定程度上，“大的”的词向量类似于“更大的”词向量，这种类似关系与“小的”的词向量类似于“更小的”词向量一致，另一种类型的相似关系可以是“大的”与“最大的”和“小的”与“最小的”。进一步，我们对拥有相同关系的两对词提出一个问题，比如我们可以问：按照“最大的”类似于“大的”这种关系，“小的”类似于那个词？

令人惊讶的是，这种问题可以通过词向量之间简单的代数运算得到答案。为了找到“小的”类似于哪个词，我们仅仅通过计算向量X=“最大的”词向量-“大的”词向量+“小的”词向量，然后再词向量空间搜索一个与该向量余弦距离最近的词向量作为X，即问题的答案(在搜索过程中我们抛弃了问题中出现的词)。当词向量被很好的训练后，通过这种代数运算找到问题的正确答案(“最小的”)是可能的。

最后，我们发现，当我们在大规模数据集上训练高维词向量时，得到的向量可以用来回答单词之间非常微妙的语义关系，例如城市和它所属的国家，像法国对于巴黎来说就像德国对于柏林的关系。具有这种语义关系的词向量可以用来改进许多现有的自然语言处理应用，例如机器翻译，信息检索和问答系统，没准可以应用于将来的某项应用中。

## 6 结论

在本论文中，通过进行一系列语法和语义的语言任务，我们研究了由各种模型训练得到的词向量的质量。我们观察到，比起流行的神经网络模型（前馈和循环），使用非常简单的模型架构训练高质量的词向量是可能做到的。由于计算复杂度的大幅度降低，从大规模数据集上训练非常精确的高维词向量是可能的。使用DistBelief分布式框架，可能能够在万亿级别的语料（基本上是无限大的词汇量）上训练CBOW和Skip-gram模型。 这比类似模型的最佳结果大上好几个数量级。

SemEval-2012任务2是一个有趣的任务，其中使用单词向量进行任务最近被证明明显地优于先前的方法。将公开可用的基于循环神经网络的词向量与其他技术一起使用在Spearman’s rank correlation任务上获得了超过50%的提升。基于神经网络的词向量在之前已经应用于许多其他自然语言处理任务中，例如情感分析和释义检测。可以预料，这些应用可以从本论文提出的模型架构中受益。

我们正在进行的工作表明，词向量可以成功地应用于知识库中事实的自动扩展，并且还可以用于验证已有事实的正确性。词向量用于机器翻译的实验结果看起来也很有前景。将来，将我们的技术与潜在关系分析等技术进行比较也会很有趣的。我们相信，我们的综合测试集将有助于研究社区改进估计词向量的现有技术。我们也期望高质量的单词向量将成为未来自然语言处理应用的重要组成部分。

## 7 接下来的工作

在完成该论文的最初版本之后，我们开源了单机多线程C ++代码，用于计算词向量，同时使用CBOW模型和Skip-gram模型。训练速度明显高于论文之前记录的速度，举例来说，对于典型的超参数选择，模型每小时大约能够训练十亿级别的词向量。我们还开源了140多万个表示命名实体的向量，这是在超过1000亿词的语料上训练得到的。我们的一些后续工作将在即将发布的NIPS 2013论文中发表。