Glove：Global Vectors for Word Representation（全局向量词表示）

语义向量空间模型用实值向量表示每个词。这些向量可作为各种应用的特征，如信息检索（information retrieval），文档分类(document classification)，问题回答(question answering)，命名实体识别(named entity recongnition)和句法分析(parsing)等问题上。

学习词向量的两个主要模型家族是：

1）全局矩阵分解方法 ：潜在语义分析 (LSA)

优点 ：可以有效地利用统计信息，训练快（数据不太大）

缺点 ：不方便处理新词和文档(需要重新计算SVD)，与其他深度模型训练方法不兼容，共现矩阵非常稀疏，矩阵维度很高（与词表大小相同），在只能捕捉到词的相似性，无法捕捉到其他模式，在词类比任务上做得相对较差，是一种次优的向量空间结构。

2）局部上下文窗口方法 :skip-gram,CBOW

优点 ：在类比任务上做得比较好，不仅可以很好的捕获语义相似性，也能捕获一定的句法和语义相关性

缺点 ：在单独的本地语境中训练，**没有使用统计数据** ，**计算量巨大** ，比较耗时。

Glove 结合了上述两个主要模型家族的优点，首先该方法明确了句法和语义规则在词向量中出现所需的模型属性，使用一个新的全局对数二次回归模型，在词类比任务上的表现为75%，显著优于skip-gram等模型。另外该方法通过训练仅仅在字 - 词共生矩阵中的非零元素，而不是在整个稀疏矩阵上或者在大的语料库中的个体上下文窗口上进行训练，充分利用了统计信息，而且计算量也大为减少。

下面介绍Glove的思路

（主要 来自Jeffrey Pennington, Richard Socher等《GloVe: Global Vectors for Word Representation》）

首先定义一些符号：

词共现矩阵用X表示，其中的X_{ij}列出单词j出现在单词i的上下文中的次数。

$X_i = \sum_k X_{ik}$：是任何单词在单词i的上下文中出现的次数

$P_{ij} = P(j|i) = X_{ij}/X_i$

是单词j出现在单词i的上下文中的概率。

为了说明词向量学习的出发点，glove的paper中首先举了一个例子，说明**词向量学习的适宜的出发点应该是同现概率的比率，而不是概率本身**，如下图所示，solid 与ice有比较大的相似性，与stream不太相关，因此，有：

$p(k|ice)/p(k|steam)$比较大，而与steam相似，与ice不太相关的gas，则$p(k|ice)/p(k|steam)$比较小，而对于与ice与steam都每太大关系的fashion，比例接近于1.

$p(k|ice)/p(k|steam)$取决于三个单词i,j,k。首先定义最一般的模型形式：

$F(w_i, w_j, \hat{w}_k) = \frac{P_{ik}}{P_{jk}}$（1）

在这个等式中，右边是从语料库中提取的，F可能依赖于一些尚未指定的参数。F的可能性数量很大，但是分析一些需求，我们可以选择一个特别的选择。

下面根据一些需求，推导出F的具体形式：

1、我们希望F在字向量空间中编码呈现比率Pik / Pjk的信息 。由于向量空间本质上是线性结构，所以最自然的方法是使用向量差，我们**将我们的考虑限制在那些仅依赖于两个目标词的差异的函数F上**，修改方程，得到：

$$F(w_i - w_j, \hat{w}_k) = \frac{P_{ik}}{P_{jk}}, \text{(2)}$$

2、方程（2）坐标是矢量，而右边是标量 ,为了统一，将F内部函数修改为点积（可以为复杂函数，但我们希望捕获简单的线性结构）。

$$F((w_i - w_j)^T . \hat{w}_k) = \frac{P_{ik}}{P_{jk}} \text{ (3)}$$

3、单词和上下文单词之间没有区别 ，我们可以自由地交换这两个角色，要做到这一点，我们不仅要交换w与$\tilde{w}$，而且要交换两个X 和X^T。我们的**最终模型应该在重新标记后保持不变** ，但方程（3）不是。但是，对称性可以分两步恢复。首先，要求同态：

$$F((w_i - w_j)^T \hat{w}_k) = \frac{F(w_i^T \tilde{w}_k)}{F(w_j^T \tilde{w}_k)}, \text{ (4)}$$

$$F(w_i^T \tilde{w}_k) = P_{ik} = \frac{X_{ik}}{X_i}, \text{(5)}$$

方程的解F为**指数函数** ，因此：

$$w_i^T \tilde{w}_k = log(P_{ik}) = log(X_{ik}) - log(X_i), \text{ (6)}$$

接下来，我们注意到方程（6）如果不是右边的对数（Xi），就会表现出对称性。然而，log^{X_i}**与k无关，所以它可以被吸收到wi的偏差bi中** 。最后，为w_k增加一个额外的偏差$\tilde{b}_k$恢复对称性.

$$w_i^T \tilde{w}_k + b_i + \tilde{b}_k = log(X_{ik})。 \text{ (7)}$$

方程（7）对公式（1）进行了极大的简化，但是由于**对数每当它的参数为零时就会变差，因此它实际上是不稳定的** 。对这个问题的一个解决办法是在对数中包含一个加法移位log（Xik）→log（1 + Xik），它保持了X的稀疏性，同时避免了异常。

这种模式的一个**主要缺点是，它平等地衡量所有的共同事件**，即使是那些很少发生或从未发生过的事件。这些很少发生的事情可能是噪音，相比于出现次数频率更高的，携带着非常少的信息。而且有75-95的零条目（这和词汇表的大小和语料库有关）

我们提出了一个**新的加权最小二乘回归模型** 来解决这些问题。方程（7）作为最小二乘问题，并将权重函数f（Xij）引入到成本函数中，给出了模型。

$$J = \sum_{i,j=1}^{V} f(ij)(w_i^T \tilde{w}_j + b_i + \tilde{b}j + logX_{ij})^2 \text{ (8)}$$

其中V是词汇的大小,权重函数应该服从以下属性：

1、f(0) = 0

2、f(x)递增，因此那些发生次数很少的不会给予过多的权重。

3、如果x非常大（过频次），f（x）应该相对较小，以避免其频率过高。

当然，大量的函数满足这些性质，但是我们发现的一类函数可以参数化为，

$$f(x) = \begin{cases} (x/x_{max})^{\alpha} if(x < x_{max}) \\ (1) otherwise \end{cases} \quad \text{(9)}$$

模型的性能取决于截断值，我们在所有实验中将x_{max} 设为 100。我们发现α= 3/4比α= 1的线性版本上改进不少。尽管我们只提供选择3/4值的经验动机，但有趣的是，发现一个简单的分数功率缩放可以给出最好的性能（限制过高频词的影响）。

文章还分析了**与其他模型的关系**

所有无监督的学习单词向量的方法最终都是基于语料库的出现统计数据，所以模型之间应该具有可比性

skip-gram或ivLBL方法的起点是:模型Q_{ij}表示单词j出现在单词i的上下文中的概率。我们假设Q_{ij}是一个softmax模型，$Q_{ij}=\frac{exp(w_i^T\tilde{w}_j)}{\sum_{k=1}^{V}exp(w_i^T\tilde{w}_k)}$ (10)

模型的目的是尝试最大化扫描整个语料库的上下文窗口的对数概率，因此全局目标函数可以写成：

$J=-\sum_{i\in corpus,j\in context(i)}logQ_{ij}$ (11)

和式中的每个项的softmax的正规化因子是昂贵的。为了有效地进行训练，skip-gram和ivLBL模型引入了Q_{ij}的近似值。公式（11）可以更有效地评估，如果我们首先将那些具有相同的i和j值的项组合在一起。

$J=-\sum_{i=1}^{V}\sum_{j=1}^{V}X_{ij}logQ_{ij}$ (12)

这样，我们可以使用类似项的数目由共生矩阵X给出的统计信息。

根据我们之前做过的标记：

$X_i=\sum_k X_{ik}$

$P_{ij}=P(j|i)=X_{ij}/X_i$

我们可以将J写为：

$J=-\sum_{i=1}^{V}X_i\sum_{j=1}^{V}P_{ij}logQ_{ij}=\sum_{i=1}^{V}X_iH(P_i,Q_i)$ (13)

其中$H(P_i,Q_i)$**是分布**P_i和 Q_i**的交叉熵**，我们以类比习惯来定义X_i。作为交叉熵误差的加权和，这个目标与公式的加权最小二乘目标有一些形式上的相似性。事实上，可以直接优化方程（13），而不用skip-gram模型和ivLBL模型中使用的在线训练方法。可以把这个目标作为一个“全局skip-gram”模型加以考虑，但还有一些更有趣的东西。另一方面，公式（13）表现出许多不好的特性，在将其作为学习单词向量的模型之前，应该加以解决。

首先，交叉熵误差只是概率分布之间的许多可能的距离度量之一，具有**长尾效应**（对于许多不重要的项，也要计算（因为它们的和比较大））的分布模型不好的特性，对于不太可能发生的事件给予了太多的权重。而且，对于有界的措施，**要求对模型分布Q进行适当的归一化处理**。因此需要方程（10）中整个词汇的总和，**存在计算瓶颈**，因此希望考虑不需要Q的这种性质的不同距离度量。一个自然的**选择是最小二乘目标**，并且**丢弃Q和P中的归一化因子**。

$\hat{J}=\sum_{i,j}X_i(\hat{P}_{i,j}-\hat{Q}_{ij})^2$ (14)

其中$\hat{P}_{ij}=X_{ij}$且 $\hat{Q}_{ij}=exp(w_i^T\tilde{w}_j)$是非归一化的分布，在这个阶段还会出现另外一个问题，即X_{ij}**往往非常大，这可能使优化复杂化**。有效的补救办法是**最小化P和Q的对数的平方误差**,即：

$\hat{J}=\sum_{i,j}X_i(log\tilde{P}_{ij}-log\tilde{Q}_{ij})^2=\sum_{i,j}X_i(w_i^T\tilde{w}_j-logX_{ij})^2$ (15)

最后，我们观察到，尽管加权因子X_i是通过skip-gram和ivLBL模型固有的在线训练方法预先确定的，但不保证是最优的。事实上，Mikolov等人（2013a）观察到通过对数据进行过滤可以提高性能，从而降低频繁词的加权因子的有效值。考虑到这一点，我们引入了一个更加一般的权重函数，我们可以自由地依赖上下文单词。结果就为：

$$J = -\sum_{i=1}^{V} X_i \sum_{j=1}^{V} f(X_{ij})(w_i^T \tilde{w}_j - logX_{ij})^2 \quad (16)$$

这相当于我们之前导出的公式（8）的成本函数。

实验

4.1评估方法

我们对Mikolov等人（2013a）的**词类比任务**（Lua et al 2013）描述的各种**单词相似性**任务和NER的**共享基准数据集**（Tjong Kim Sang and De Meulder，2003）

词类别

单词比喻任务包括如下问题，'a is to b as c is to __?'数据集包含19,544个这样的问题，分为语义子集和句法子集。语义问题通常是关于人或地点的类比，比如"雅典对希腊来说，相当于柏林对？"。为了正确回答这个问题，模型应该唯一地标识缺少的术语，只有一个确切的对应关系被计算为正确的匹配。我们根据余弦的相似性找出表示wd最接近wb - wa + wc的单词d来回答"a是b是c是？

词相似性

虽然类比任务是我们关注的向量空间子结构的主要关注点，但我们也在表3中的各种单词相似度任务上评估了我们的模型。其中包括WordSim-353（Finkelstein等，2001），MC（Miller和Charles，1991），RG（Rubenstein和Goodenough，1965），SCWS（Huang等，2012）和RW（Luong等，2013）。

命名实体识别

NER的CoNLL-2003英文基准数据集是路透社新闻稿文章的集合，由四种实体类型注明：人员，地点，组织和杂项。我们对CoNLL-03训练数据进行了三个数据集的测试：1）ConLL-03测试数据，2）ACE阶段2（2001-02）和ACE-2003数据，以及3）MUC7正式运行测试集。我们采用BIO2认证标准，以及（Wang和Manning，2013）所述的所有预处理步骤。我们使用斯坦福NER模型的标准分布（Finkel等，2005）的一套全面的离散特征。CoNLL-2003训练数据集共产生了437,905个离散特征。另外，五字上下文的每个词的50维向量被添加并用作连续特征。以这些特征作为输入，我们用与（Wang和Manning，2013）的CRFjoin模型完全相同的设置来训练条件随机场（CRF）。

结果

我们在表2中的单词类比任务中给出了结果。GloVe模型比其他方法表现得更好，通常具有更小的矢量大小和更小的语料库。我们使用word2vec工具的结果比以前发表的结果要好一些。这是由于许多因素造成的，包括我们选择使用负面抽样（通常比分层softmax更有效），负样本的数量和语料库的选择。

我们证明，模型可以很容易地在一个大型的420亿token语料库上进行训练，并有相当的性能提升。我们注意到，增加语料库的大小并不能保证改善其他模型的结果，这可以从这个更大的语料库上SVD-L模型的性能下降来看出来。这个基本的SVD模型不能很好地扩展到大的范围的事实为我们模型中提出的加权方案的必要性提供了进一步的证据。

表3显示了五个不同的单词相似性数据集的结果。通过首先对整个词汇表中的每个特征进行归一化，然后计算余弦相似度，从词向量获得相似度分数。我们计算这个分数与人类判断之间的斯皮尔曼等级相关系数。CBOW *表示word2vec网站上提供的经过100B字新闻数据的单词和短语向量的训练。当使用不到一半大小的语料库时，GloVe胜过它。

表4显示了基于CRF模型的NER任务的结果。L-BFGS训练在迭代25次之后在开发集上没有改进时做了终止。所有的配置都与Wang和Manning（2013）使用的配置相同。标注为“离散”的模型是使用斯坦福NER模型的标准分布附带的全面离散特征组成的方法，但是没有任何词汇特征。除了之前讨论的HPCA和SVD模型之外，我们还比较了Huang等人（2012）（HSMN）和Collobert和Weston（2008）（CW）的模型。我们使用word2vec工具来训练CBOW模型。除了CoNLL测试集以外，GloVe模型在所有评估指标上都优于其他所有方法，而HPCA方法在这个方法上略胜一筹。我们得出结论GloVe矢量在下游的（机器翻译等高级任务）NLP任务中是有用的，如（Turian等人，2010）中神经矢量首次显示的那样。

结论

对于相同的语料库，词汇量，窗口大小和培训时间，Glove始终优于word2vec。速度更快，效果更好，速度也更快 ，关于分配词表示是否最好从基于计数的方法或从基于预测的方法中学习的问题已经引起了相当大的关注。目前，基于预测的模型获得了大量的支持，例如Baroni等（2014）认为，这些模型在一系列任务中表现更好。在这项工作中，我们认为这两类方法在有趣的基础上并没有显着的不同，因为它们都探索了语料库的同构统计，但基于计数的方法捕捉全球统计数据的效率可能是有利的。我们构建了一个利用计数数据主要特点的模型，同时捕获最近基于log-bilinear预测的方法（如word2vec）中常见的有意义的线性子结构。结果GloVe是一个新的全局对数双线性回归模型，用于无监督学习的单词表征，在词类比，单词相似性和命名实体识别任务方面优于其他模型。

word2vec的评价标准

内部标准

所以内在评价通常是在某个特定或中间的子任务上，比如内积相似性等。

好处：很容易计算

得到一个数字，很直观

可以进行融合和调整

但是不能说系统性能确实提升（目标可能不对。可能并不是人们关心的）。

外部标准

通过对外部实际应用的效果提升来体现（比如将embeding于机器翻译），耗时较长，需要将不同方法计算出的vector进行同样的操作，看看那个提升了性能，而且，每次只能有一个变量（多变量分不清）

相关课程：斯坦福 CS224n 《Natural Language Processing with Deep Learning》 Advanced Word Vector Representation

相关文章：Jeffrey Pennington, Richard Socher等《GloVe: Global Vectors for Word Representation》