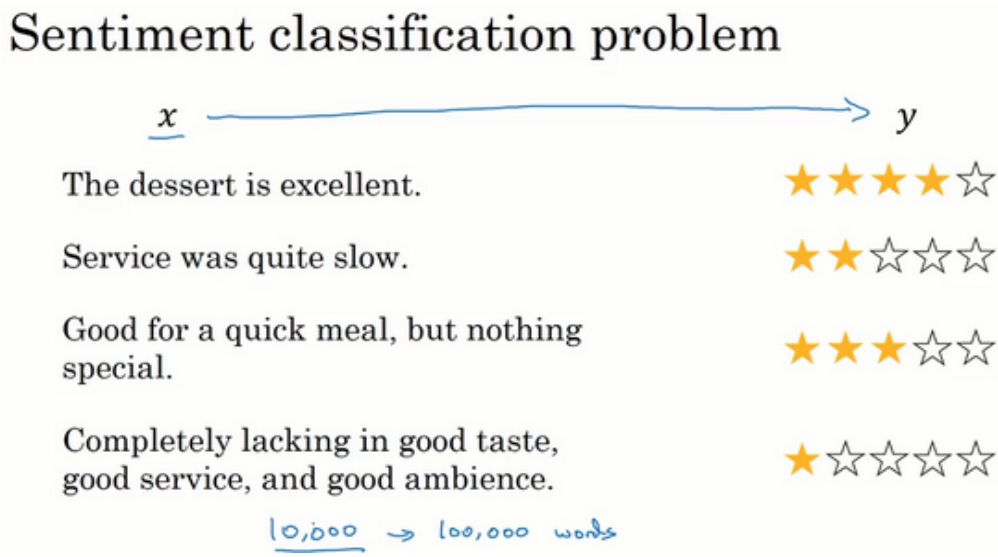
2.9 情感分类（Sentiment Classification）

情感分类任务就是看一段文本，然后分辨这个人是否喜欢他们在讨论的这个东西，这是 **NLP** 中最重要的模块之一，经常用在许多应用中。情感分类一个最大的挑战就是可能标记的训练集没有那么多，但是有了词嵌入，即使只有中等大小的标记的训练集，你也能构建一个不错的情感分类器，让我们看看是怎么做到的。



这是一个情感分类问题的一个例子（上图所示），输入x是一段文本，而输出y是你要预测的相应情感。比如说是一个餐馆评价的星级，

比如有人说，**"The dessert is excellent."**（甜点很棒），并给出了四星的评价；

"Service was quite slow"（服务太慢），两星评价；

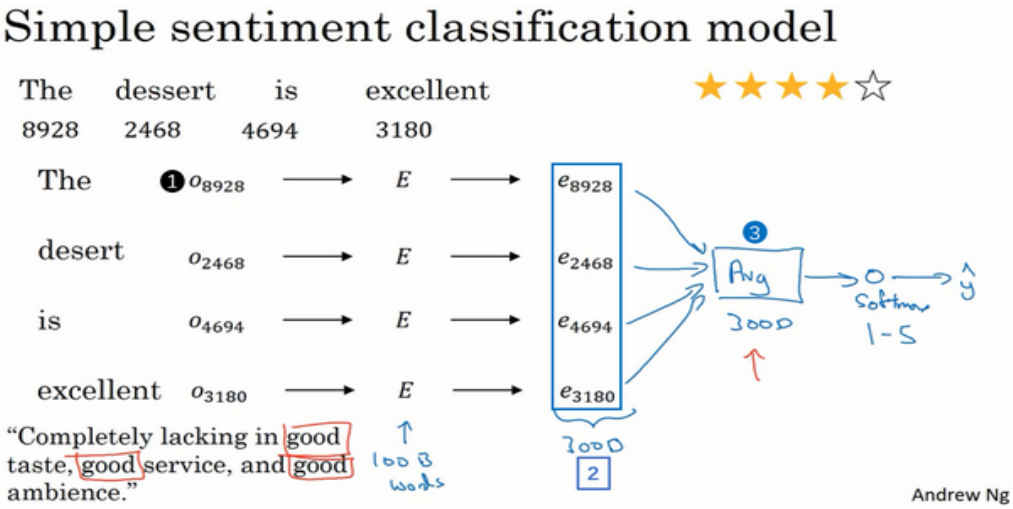
"Good for a quick meal but nothing special"（适合吃快餐但没什么亮点），三星评价；

还有比较刁钻的评论，**"Completely lacking in good taste, good service and good ambiance."**（完全没有好的味道，好的服务，好的氛围），给出一星评价。

如果你能训练一个从x到y的映射，基于这样的标记的数据集，那么你就可以用来搜集大家对你运营的餐馆的评价。一些人可能会把你的餐馆信息放到一些社交媒体上，**Twitter**、**Facebook**、**Instagram** 或者其他的社交媒体，如果你有一个情感分类器，那么它就可以看一段文本然后分析出这个人对你的餐馆的评论的情感是正面的还是负面的，这样你就可以一直记录是否存在一些什么问题，或者你的餐馆是在蒸蒸日上还是每况愈下。

情感分类一个最大的挑战就是可能标记的训练集没有那么多。对于情感分类任务来说，训练集大小从 10,000 到 100,000 个单词都很常见，甚至有时会小于 10,000 个单词，采用了

词嵌入能够带来更好的效果，尤其是只有很小的训练集时。



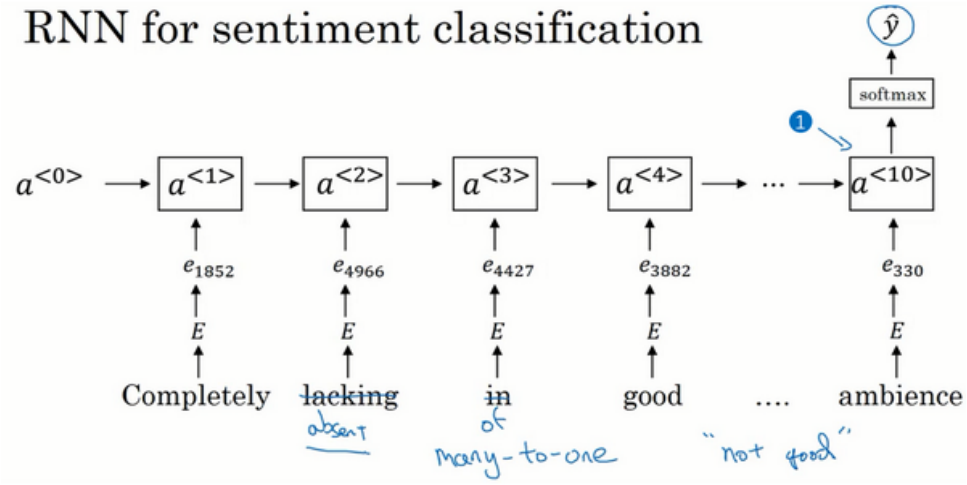
接下来你可以这样做，这节我们会讲几个不同的算法。这是一个简单的情感分类的模型，假设有一个句子"**dessert is excellent**"，然后在词典里找这些词，我们通常用 10,000 个词的词汇表。我们要构建一个分类器能够把它映射成输出四个星，给定这四个词（"**dessert is excellent**"），我们取这些词，找到相应的 **one-hot** 向量，所以这里（上图编号 1 所示）就是o_{8928}，乘以嵌入矩阵E，E可以从一个很大的文本集里学习到，比如它可以从一亿个词或者一百亿个词里学习嵌入，然后用来提取单词 **the** 的嵌入向量e_{8928}，对 **dessert**、**is**、**excellent** 做同样的步骤。

如果在很大的训练集上训练E，比如一百亿的单词，这样你就会获得很多知识，甚至从有些不常用的词中获取，然后应用到你的问题上，即使你的标记数据集里没有这些词。我们可以这样构建一个分类器，取这些向量（上图编号 2 所示），比如是 300 维度的向量。然后把它们求和或者求平均，这里我画一个大点的平均值计算单元（上图编号 3 所示），你也可以用求和或者平均。这个单元（上图编号 3 所示）会得到一个 300 维的特征向量，把这个特征向量送进 **softmax** 分类器，然后输出$\hat{y}$。这个 **softmax** 能够输出 5 个可能结果的概率值，从一星到五星，这个就是 5 个可能输出的 **softmax** 结果用来预测y的值。

这里用的平均值运算单元，这个算法适用于任何长短的评论，因为即使你的评论是 100 个词长，你也可以对这一百个词的特征向量求和或者平均它们，然后得到一个表示一个 300 维的特征向量表示，然后把它送进你的 **softmax** 分类器，所以这个平均值运算效果不错。它实际上会把所有单词的意思给平均起来，或者把你的例子中所有单词的意思加起来就可以用了。

这个算法有一个问题就是没考虑词序，尤其是这样一个负面的评价，"**Completely lacking**

in good taste, good service, and good ambiance."，但是 **good** 这个词出现了很多次，有 3 个 **good**，如果你用的算法跟这个一样，忽略词序，仅仅把所有单词的词嵌入加起来或者平均下来，你最后的特征向量会有很多 **good** 的表示，你的分类器很可能认为这是一个好的评论，尽管事实上这是一个差评，只有一星的评价。



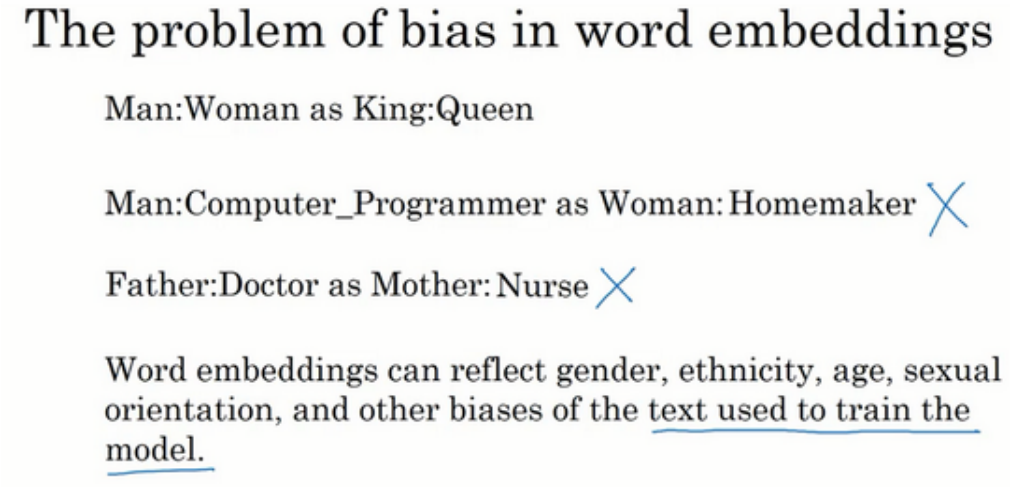
我们有一个更加复杂的模型，不用简单的把所有的词嵌入都加起来，我们用一个 **RNN** 来做情感分类。我们这样做，首先取这条评论，**"Completely lacking in good taste, good service, and good ambiance."**，找出每一个 **one-hot** 向量，这里我跳过去每一个 **one-hot** 向量的表示。用每一个 **one-hot** 向量乘以词嵌入矩阵E，得到词嵌入表达e，然后把它们送进 **RNN** 里。**RNN** 的工作就是在最后一步（上图编号 1 所示）计算一个特征表示，用来预测$\hat{y}$，这是一个多对一的网络结构的例子，我们之前已经见过了。有了这样的算法，考虑词的顺序效果就更好了，它就能意识到**"things are lacking in good taste"**，这是个负面的评价，**“not good”**也是一个负面的评价。而不像原来的算法一样，只是把所有的加在一起得到一个大的向量，根本意识不到**“not good”**和 **“good”**不是一个意思，**"lacking in good taste"**也是如此，等等。

如果你训练一个这样的算法，最后会得到一个很合适的情感分类的算法。由于你的词嵌入是在一个更大的数据集里训练的，这样效果会更好，更好的泛化一些没有见过的新的单词。比如其他人可能会说，**"Completely absent of good taste, good service, and good ambiance."**，即使 **absent** 这个词不在标记的训练集里，如果是在一亿或者一百亿单词集里训练词嵌入，它仍然可以正确判断，并且泛化的很好，甚至这些词是在训练集中用于训练词嵌入的，但是可以不在专门用来做情感分类问题的标记的训练集中。

以上就是情感分类的问题，我希望你能大体了解。一旦你学习到或者从网上下载词嵌入，你就可以很快构建一个很有效的 **NLP** 系统。

2.10 词嵌入除偏（Debiasing Word Embeddings）

现在机器学习和人工智能算法正渐渐地被信任用以辅助或是制定极其重要的决策，因此我们想尽可能地确保它们不受非预期形式偏见影响，比如说性别歧视、种族歧视等等。本节视频中我会向你展示词嵌入中一些有关减少或是消除这些形式的偏见的办法。

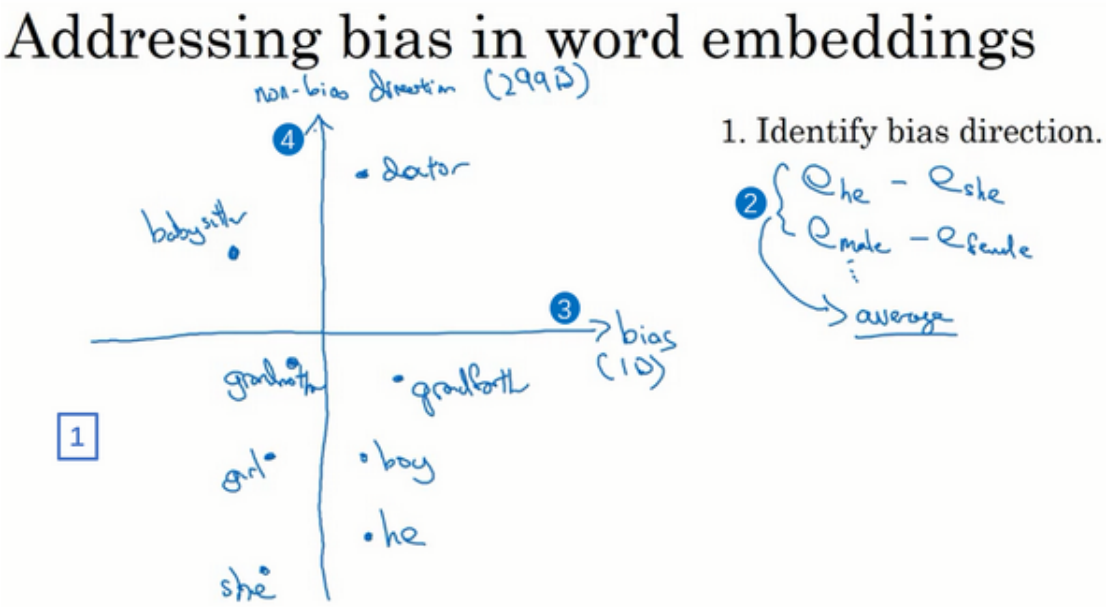


本节视频中当我使用术语 **bias** 时，我不是指 **bias** 本身这个词，或是偏见这种感觉，而是指性别、种族、性取向方面的偏见，那是不同的偏见，同时这也通常用于机器学习的学术讨论中。不过我们讨论的大部分内容是词嵌入是怎样学习类比像 **Man：Woman**，就像 **King：Queen**，不过如果你这样问，如果 **Man** 对应 **Computer Programmer**，那么 **Woman** 会对应什么呢？所以这篇论文（上图编号 1 所示:**Bolukbasi T, Chang K W, Zou J, et al. Man is to Computer Programmer as Woman is to Homemaker? Debiasing Word Embeddings[J]. 2016.**）的作者 **Tolga Bolukbasi**、**Kai-Wei Chang**、**James Zou**、**Venkatesh Saligrama** 和 **Adam Kalai** 发现了一个十分可怕的结果，就是说一个已经完成学习的词嵌入可能会输出 **Man：Computer Programmer**，同时输出 **Woman：Homemaker**，那个结果看起来是错的，并且它执行了一个十分不良的性别歧视。如果算法输出的是 **Man：Computer Programmer**，同时 **Woman：Computer Programmer** 这样子会更合理。同时他们也发现如果 **Father：Doctor**，那么 **Mother** 应该对应什么呢？一个十分不幸的结果是，有些完成学习的词嵌入会输出 **Mother：Nurse**。

因此根据训练模型所使用的文本，词嵌入能够反映出性别、种族、年龄、性取向等其他方面的偏见，一件我尤其热衷的事是，这些偏见都和社会经济状态相关，我认为每个人不论你出身富裕还是贫穷，亦或是二者之间，我认为每个人都应当拥有好的机会，同时因为机器学习算法正用来制定十分重要的决策，它也影响着世间万物，从大学录取到人们找工作的途径，到贷款申请，不论你的的贷款申请是否会被批准，再到刑事司法系统，甚至是判决标准，

学习算法都在作出非常重要的决策，所以我认为我们尽量修改学习算法来尽可能减少或是理想化消除这些非预期类型的偏见是十分重要的。

至于词嵌入，它们能够轻易学会用来训练模型的文本中的偏见内容，所以算法获取到的偏见内容就可以反映出人们写作中的偏见。在漫长的世纪里，我认为人类已经在减少这些类型的偏见上取得了进展，幸运的是对于人工智能来说，实际上我认为有更好的办法来实现更快地减少 AI 领域中相比与人类社会中的偏见。虽然我认为我们仍未实现人工智能，仍然有许多研究许多难题需要完成来减少学习算法中这些类型的偏见。

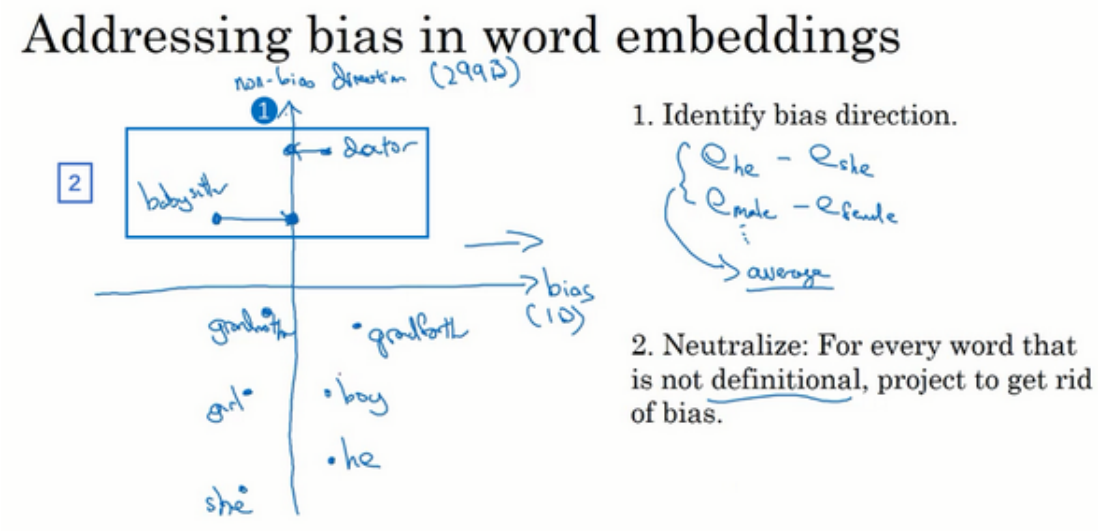


本节视频里我想要做的是与你们分享一个例子，它是一篇论文的一套办法，就是下面引用的这篇由 **Bolukbasi** 和其他人共同撰写的论文，它是研究减少词嵌入中偏见问题的。就是这些，假设说我们已经完成一个词嵌入的学习，那么 **babysitter** 就是在这里，**doctor** 在这里，**grandmother** 在这里，**grandfather** 在这里，也许 **girl** 嵌入在这里，**boy** 嵌入在这里，也许 **she** 嵌在这里，**he** 在这里（上图编号 1 所示的区域内），所以首先我们要做的事就是辨别出我们想要减少或想要消除的特定偏见的趋势。

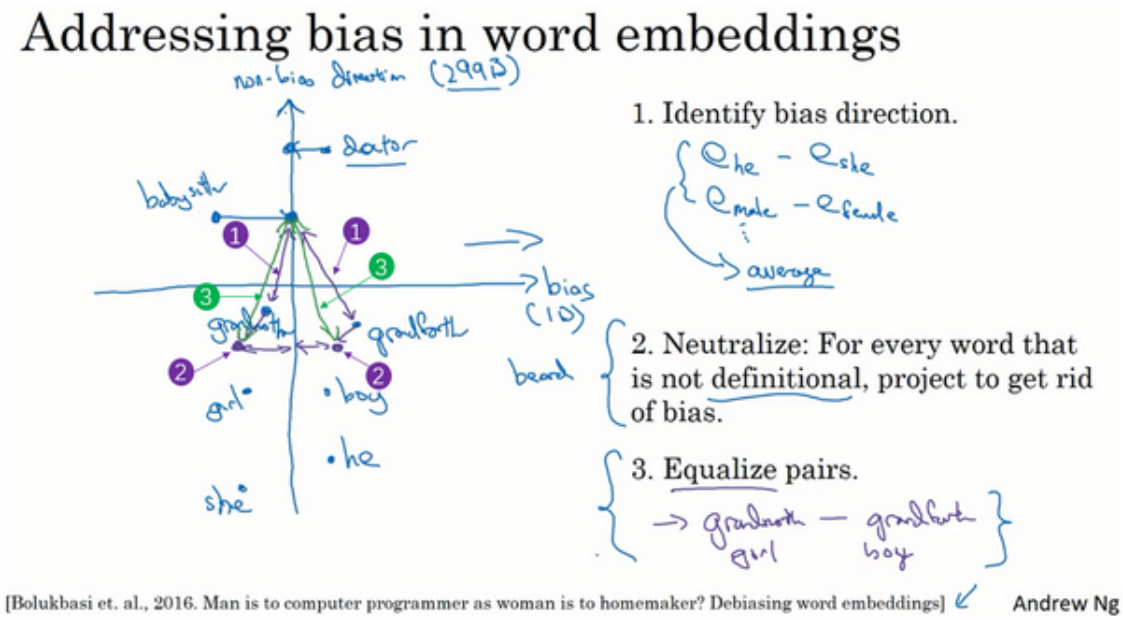
为了便于说明，我会集中讨论性别歧视，不过这些想法对于所有我在上个幻灯片里提及的其他类型的偏见都是通用的。这个例子中，你会怎样辨别出与这个偏见相似的趋势呢？主要有以下三个步骤：

一、对于性别歧视这种情况来说，我们能做的是$e_{\text{he}} - e_{\text{she}}$，因为它们的性别不同，然后将$e_{\text{male}} - e_{\text{female}}$，然后将这些值取平均（上图编号 2 所示），将这些差简单地求平均。这个趋势（上图编号 3 所示）看起来就是性别趋势或说是偏见趋势，然后这个趋势（上图编号 4 所示）与我们想要尝试处理的特定偏见并不相关，因此这就是个无偏见趋势。在这种情况下，

偏见趋势可以将它看做 **1D** 子空间，所以这个无偏见趋势就会是 **299D** 的子空间。我已经略微简化了，原文章中的描述这个偏见趋势可以比 1 维更高，同时相比于取平均值，如同我在这里描述的这样，实际上它会用一个更加复杂的算法叫做 **SVU**，也就是奇异值分解，如果你对主成分分析（**Principle Component Analysis**）很熟悉的话，奇异值分解这个算法的一些方法和主成分分析 **(PCA)**其实很类似。



二、中和步骤，所以对于那些定义不确切的词可以将其处理一下，避免偏见。有些词本质上就和性别有关，像 **grandmother**、**grandfather**、**girl**、**boy**、**she**、**he**，他们的定义中本就含有性别的内容，不过也有一些词像 **doctor** 和 **babysitter** 我们想使之在性别方面是中立的。同时在更通常的情况下，你可能会希望像 **doctor** 或 **babysitter** 这些词成为种族中立的，或是性取向中立的等等，不过这里我们仍然只用性别来举例说明。对于那些定义不明确的词，它的基本意思是不像 **grandmother** 和 **grandfather** 这种定义里有着十分合理的性别含义的，因为从定义上来说 **grandmothers** 是女性，**grandfather** 是男性。所以对于像 **doctor** 和 **babysitter** 这种单词我们就可以将它们在这个轴（上图编号 1 所示）上进行处理，来减少或是消除他们的性别歧视趋势的成分，也就是说减少他们在这个水平方向上的距离（上图编号 2 方框内所示的投影），所以这就是第二个中和步。



三、均衡步，意思是说你可能会有这样的词对，**grandmother** 和 **grandfather**，或者是 **girl** 和 **boy**，对于这些词嵌入，你只希望性别是其区别。那为什么要那样呢？在这个例子中，**babysitter** 和 **grandmother** 之间的距离或者说是相似度实际上是小于 **babysitter** 和 **grandfather** 之间的（上图编号 1 所示），因此这可能会加重不良状态，或者可能是非预期的偏见，也就是说 **grandmothers** 相比于 **grandfathers** 最终更有可能输出 **babysitting**。所以在最后的均衡步中，我们想要确保的是像 **grandmother** 和 **grandfather** 这样的词都能够有一致的相似度，或者说是相等的距离，和 **babysitter** 或是 **doctor** 这样性别中立的词一样。这其中会有一些线性代数的步骤，但它主要做的就是将 **grandmother** 和 **grandfather** 移至与中间轴线等距的一对点上（上图编号 2 所示），现在性别歧视的影响也就是这两个词与 **babysitter** 的距离就完全相同了（上图编号 3 所示）。所以总体来说，会有许多对像 **grandmother-grandfather**，**boy-girl**，**sorority-fraternity**，**girlhood-boyhood**，**sister-brother**，**niece-nephew**，**daughter-son** 这样的词对，你可能想要通过均衡步来解决他们。

最后一个细节是你怎样才能够决定哪个词是中立的呢？对于这个例子来说 **doctor** 看起来像是一个应该对其中立的单词来使之性别不确定或是种族不确定。相反地，**grandmother** 和 **grandfather** 就不应是性别不确定的词。也会有一些像是 **beard** 词，一个统计学上的事实是男性相比于比女性更有可能拥有胡子，因此也许 **beard** 应该比 **female** 更靠近 **male** 一些。

因此论文作者做的就是训练一个分类器来尝试解决哪些词是有明确定义的，哪些词是性别确定的，哪些词不是。结果表明英语里大部分词在性别方面上是没有明确定义的，意思就是说性别并是其定义的一部分，只有一小部分词像是 **grandmother-grandfather**，**girl-boy**，**sorority-fraternity** 等等，不是性别中立的。因此一个线性分类器能够告诉你哪些词能够通过

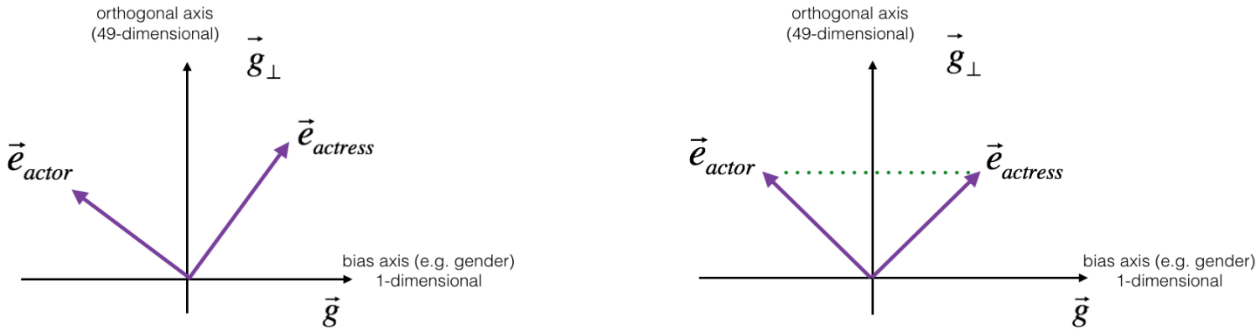
中和步来预测这个偏见趋势，或将其与这个本质是 **299D** 的子空间进行处理。

最后，你需要平衡的词对的数实际上是很小的，至少对于性别歧视这个例子来说，用手都能够数出来你需要平衡的大部分词对。完整的算法会比我在这里展示的更复杂一些，你可以去看一下这篇论文了解详细内容，你也可以通过编程作业来练习一下这些想法。

参考资料：针对性别特定词汇的均衡算法

如何对两个单词除偏，比如：**"actress"**（**"女演员"**）和**"actor"**（**"演员"**）。 均衡算法适用于您可能希望仅通过性别属性不同的单词对。 举一个具体的例子，假设**"actress"**（**"女演员"**）比**"actor"**（**"演员"**）更接近"保姆"。 通过将中和应用于**"babysit"**（**"保姆"**），我们可以减少与保姆相关的性别刻板印象。 但是这仍然不能保证**"actress"**（**"女演员"**）和**"actor"**（**"演员"**）与**"babysit"**（**"保姆"**）等距。 均衡算法可以解决这个问题。

均衡背后的关键思想是确保一对特定的单词与 49 维$g_{\perp}$距离相等 。均衡步骤还可以确保两个均衡步骤现在与$e_{receptionist}^{debiased}$ 距离相同，或者用其他方法进行均衡。下图演示了均衡算法的工作原理：



before equalizing,
"actress" and "actor" differ in many ways beyond the direction of $\vec{g}$

after equalizing,
"actress" and "actor" differ only in the direction of $\vec{g}$, and further are equal in distance from $\vec{g}_{\perp}$

公式的推导有点复杂(参考论文：**Bolukbasi T, Chang K W, Zou J, et al. Man is to Computer Programmer as Woman is to Homemaker? Debiasing Word Embeddings[J]. 2016.**)

主要步骤如下:

$$\mu = \frac{e_{w1} + e_{w2}}{2}$$

$$\mu_B = \frac{\mu * \text{bias}_a\text{xis}}{||\text{bias}_a\text{xis}||_2} + ||\text{bias}_a\text{xis}||_2 * \text{bias}_a\text{xis}$$

$$\mu_{\perp} = \mu - \mu_B$$

$$e_{w1B} = \sqrt{|1 - \| \mu_{\perp} \|_2^2|} * \frac{(e_{w1} - \mu_{\perp}) - \mu_B}{|(e_{w1} - \mu_{\perp}) - \mu_B)|}$$

$$e_{w2B} = \sqrt{|1 - \| \mu_{\perp} \|_2^2|} * \frac{(e_{w2} - \mu_{\perp}) - \mu_B}{|(e_{w2} - \mu_{\perp}) - \mu_B)|}$$

$$e_1 = e_{w1B} + \mu_{\perp} \quad e_2 = e_{w2B} + \mu_{\perp}$$

总结一下，减少或者是消除学习算法中的偏见问题是个十分重要的问题，因为这些算法会用来辅助制定越来越多的社会中的重要决策，在本节视频中分享了一套如何尝试处理偏见问题的办法，不过这仍是一个许多学者正在进行主要研究的领域。