机器学习在文本分类中的应用

张忠敏

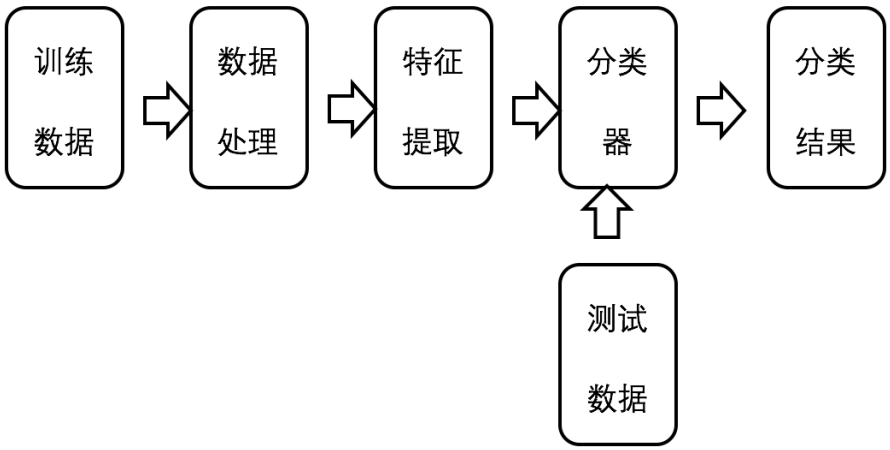
1. 概述

随着信息技术的发展,互联网数据及资源呈现海量特征.为了有效地管理和利用这些海量信息,基于内容的信息检索和数据挖掘逐渐成为备受关注的领域.其中,文本分类(text categorization,简称 TC)技术是信息检索和文本挖掘的重要基础,其主要任务是在预先给定的类别标记(label)集合下,根据文本内容判定它的类别.

文本分类在自然语言处理与理解、信息组织与管理、内容信息过滤等领域都有着广泛的应用.例如文章主题自动分类、邮件自动分类、垃圾邮件识别、用户情感分类等。20 世纪 90 年代，逐渐成熟的基于机器学习的文本分类方法,更注重分类器的模型自动挖掘和生成及动态优化能力,在分类效果和灵活性上都比之前基于知识工程和专家系统的文本分类模式有所突破,成为相关领域研究和应用的经典范例。基于机器学习的文本分类基础技术由文本的表示(representation)、分类方法及效果(effectiveness)评估 3 部分组成。

2. 传统方法

传统文本分类方法遵循机器学习的一般流程，如下图：



2.1 传统特征表示

由训练数据经过预处理后，经过特征工程得到特征表征，进而交给分类器监督训练，得到预测结果。特征工程是这一流程中的工作重点，是把数据转化为样本表征（information）的过程，分类器则是将样本表征（information）转化为结果（knowledge)。在文本分类问题中，经典的特征表示方法有词袋模型（Bags of words）和 tf-idf。

词袋模型，是忽略其词序、语法和句法，将其仅仅看作是一个词的集合，使用词袋模型将文本做特征表征就是一个根据词频(term frequency)表示成 ones-hot 的过程。sklearn 中的 CountVectorizer 是具体的实现。

上述表示方法的问题是停用词对结果的影响非常大，例如，中文中的“的”、

“是”、“在”等最常用的词出现次数非常多，而这些词一般对于分类任务没有任何帮助。这时我们就需要对每个词设置不同的权重，衡量一个词是不是常见词。如果某个词比较少见，但它在这篇文章中多次出现，那么它很可能就反映了这篇文章的特性，也就是说，它是该文章的关键词。tf-idf 与一个词在文档中的出现次数（tf）成正比，与该词在整个语料库中的出现次数（df）成反比，这就是 tf-idf 的基本原理。sklearn 中的 TfidfVectorizer 是具体的实现。

如果一个语料库的词表非常大的情况下，使用这两种特征表示方法，都有非常大的问题：特征稀疏和不含语义。可以使用降维方法解决特征稀疏问题，如 LDA，SVD（LSA）等可以作用于此，这样就得到降维后的稠密表示，并且能起到挖掘隐含主题的作用。对于不含语序可以添加 n-grams 的表示，能在一定程度上捕捉词的语序。sklearn 中的 tfidfvector 可以通过 ngram 参数来增加 n-grams 信息。

2.2 常见传统分类算法

一、Rocchio 算法

Rocchio 算法的基本思路是把一个类别里的样本文档各项取个平均值（例如把所有“体育”类文档中词汇“篮球”出现的次数取个平均值，再把“裁判”取个平均值，依次做下去），就可以得到一个新的向量，形象的称之为“质心”，质心就成了这个类别最具代表性的向量表示。再有新的文档需要判断的时候，比较新文档和质心有多么相像就可以确定新文档属不属于这个类。稍微改进一点的 Rocchio 算法不仅考虑属于这个类别的文档（称为正样本），也考虑不属于这个类别的文档数据（称为负样本），计算出来的质心尽量靠近正样本同时尽量远离负样本。

它的优点是容易实现，计算（训练和分类）特别简单，它通常是用来实现衡量分类系统性能的基准系统，而实用的分类系统很少采用这种算法解决具体的分类问题。

二、朴素贝叶斯（NB）

传统的文本分类方法还有朴素贝叶斯的建模方式，使用贝叶斯公式直接对一句话的分类概率建模。朴素贝叶斯分类器是有监督学习的一种，它是基于贝叶斯定理与特征条件独立假设的分类方法。常见有两种模型，多项式模型(multinomial model)和伯努利模型(Bernoulli model)。对于给定的训练数据集，首先基于特征条件独立假设学习输入输出的联合概率分布；然后基于此模型，对给定的输入 x，利用贝叶斯定理求出后验概率最大的输出 y。

它的优点是发源于古典数学理论，有着坚实的数学基础，以及稳定的分类效率。所需估计的参数很少，对缺失数据不太敏感，算法也比较简单。

缺点有：理论上，NBC 模型与其他分类方法相比具有最小的误差率。但是实际上并非总是如此，这是因为 NBC 模型假设属性之间相互独立，这个假设在实际应用中往往是不成立的（可以考虑用聚类算法先将相关性较大的属性聚类），这给 NBC 模型的正确分类带来了一定影响。在属性个数比较多或者属性之间相关性较大时，NBC 模型的分类效率比不上决策树模型。而在属性相关性较小时，NBC 模型的性能最为良好。需要知道先验概率。分类决策存在错误率。

三、KNN 算法(K-Nearest Neighbour)

k 近邻法假设给定一个训练数据集，其中的实例类别已定。分类时对新的实

例，根据其 k 个最近邻的训练实例的类别，通过多数表决等方式进行预测，因此，k 近邻法不具备显式的学习过程。k 近邻法的三个基本要素是：k 值的选择，距离度量和分类决策规则。

在文本分类中，步骤如下：

（1）文本预处理，向量化，根据特征词的 tf-idf 值计算

（2）当新文本到达后，根据特征词计算新文本的向量

（3）在训练文本中选出与新文本最相近的 k 个文本，相似度用向量夹角的余弦值度量。(注：k 的值目前没有好的办法确定，只有根据实验来调整 k 的值）

（4）在新文本的 k 个相似文本中，依此计算每个类的权重，每个类的权重等于 k 个文本中属于该类的训练样本与测试样本的相似度之和。

（5）比较类的权重，将文本分到权重最大那个类别中。

它的优点：简单、有效。重新训练的代价较低（类别体系的变化和训练集的变化，在 Web 环境和电子商务应用中是很常见的）。计算时间和空间线性于训练集的规模(在一些场合不算太大)。由于 KNN 方法主要靠周围有限的邻近的样本，而不是靠判别类域的方法来确定所属类别的，因此对于类域的交叉或重叠较多的待分样本集来说，KNN 方法较其他方法更为适合。该算法比较适用于样本容量比较大的类域的自动分类，而那些样本容量较小的类域采用这种算法比较容易产生误分。

缺点：KNN 算法是懒散学习方法（lazy learning,基本上不学习），比一些积极学习的算法要快很多。类别评分不是规格化的（不像概率评分）。输出的可解释性不强，例如决策树的可解释性较强。该算法在分类时有个主要的不足是，当样本不平衡时，如一个类的样本容量很大，而其他类样本容量很小时，有可能导致当输入一个新样本时，该样本的 K 个邻居中大容量类的样本占多数。该算法只计算“最近的”邻居样本，某一类的样本数量很大，那么或者这类样本并不接近目标样本，或者这类样本很靠近目标样本。无论怎样，数量并不能影响运行结果。可以采用权值的方法（和该样本距离小的邻居权值大）来改进。计算量较大。目前常用的解决方法是事先对已知样本点进行剪辑，事先去除对分类作用不大的样本。

四、决策树（Decision Trees）

决策树学习通常包括 3 个步骤：特征选择、决策树的生成和决策树的剪枝。其中，特征选择是决定用哪个特征来划分特征空间；决策树的生成对应于模型的局部选择，决策树的剪枝对应于模型的全局选择。

它的优点：

（1）决策树易于解释。它可以毫无压力地处理特征间的交互关系。

（2）对于决策树，数据的准备往往是简单或者是不必要的.其他的技术往往要求先把数据一般化，比如去掉多余的或者空白的属性。

（3）能够同时处理数据型和常规型属性。其他的技术往往要求数据属性的单一。

（4）决策树是一个白盒模型。如果给定一个观察的模型，那么根据所产生的决策树很容易推出相应的逻辑表达式。

（5）易于通过静态测试来对模型进行评测。表示有可能测量该模型的可信度。

（6）在相对短的时间内能够对大型数据源做出可行且效果良好的结果。

（7）可以对有许多属性的数据集构造决策树。

（8）决策树可很好地扩展到大型数据库中，同时它的大小独立于数据库的大小。

它的缺点：

（1）对于那些各类别样本数量不一致的数据，在决策树当中,信息增益的结果偏向于那些具有更多数值的特征（ID3）。

（2）决策树处理缺失数据时的困难。

（3）过度拟合问题的出现。

（4）忽略数据集中属性之间的相关性。

五、Adaboosting 方法

（1）adaboost 是一种有很高精度的分类器。

（2）可以使用各种方法构建子分类器，Adaboost 算法提供的是框架。

（3）当使用简单分类器时，计算出的结果是可以理解的。而且弱分类器构造极其简单。

（4）简单，不用做特征筛选。

（5）不用担心 overfitting。

六、支持向量机（SVM）

优点：

（1）可以解决小样本情况下的机器学习问题。

（2）可以提高泛化性能。

（3）可以解决高维问题。

（4）可以解决非线性问题。

（5）可以避免神经网络结构选择和局部极小点问题。

缺点：

（1）对缺失数据敏感。

（2）对非线性问题没有通用解决方案，必须谨慎选择核函数来处理。

七、人工神经网络

优点：分类的准确度高,并行分布处理能力强,分布存储及学习能力强，对噪声神经有较强的鲁棒性和容错能力，能充分逼近复杂的非线性关系，具备联想记忆的功能等。

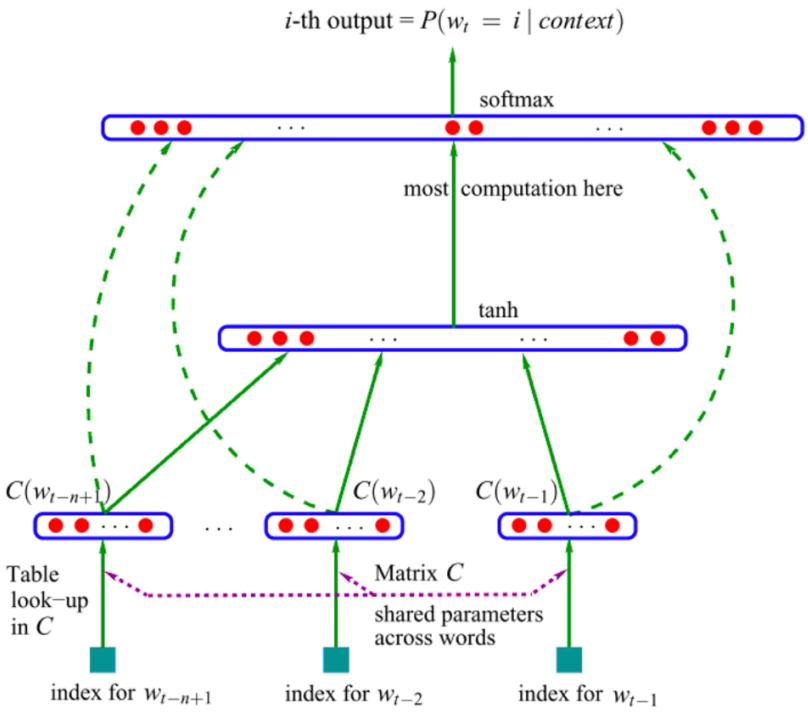
缺点：神经网络需要大量的参数，如网络拓扑结构、权值和阈值的初始值；不能观察之间的学习过程，输出结果难以解释，会影响到结果的可信度和可接受程度；学习时间过长,甚至可能达不到学习的目的。

3. 基于深度学习的文本分类

传统的文本分类需要依赖很多词法、句法相关的 human-extracted feature，自 2012 年深度学习技术快速发展之后，尤其是循环神经网络 RNN、卷积神经网络 CNN 在 NLP 领域逐渐获得广泛应用，使得传统的文本分类任务变得更加容易，准确率也不断提升。

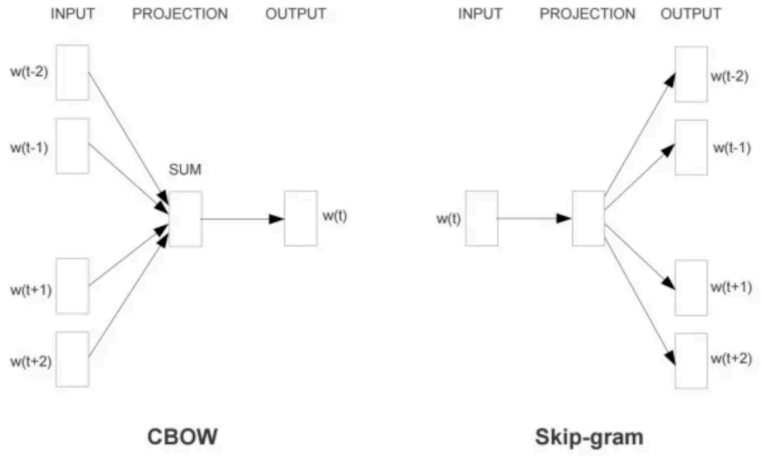
3.1 文本的分布式表示：词向量

NLP 领域最为常用的文本表示有四种：（1）基于 one-hot、tf-idf、textrank 等的 bag-of-words；（2）主题模型：LSA（SVD）、pLSA、LDA；（3）基于词向量的固定表征：word2vec、fastText、glove；（4）基于词向量的动态表征：elmo、GPT、bert。最简单的 one-hot 存在维数灾难和语义鸿沟等问题；通过构建共现矩阵并利用 SVD 求解构建词向量，则计算复杂度高；而早期词向量的研究通常来源于语言模型，比如 NNLM（Nerual Network Language Model）和 RNNLM，其主要目的是语言模型，而词向量只是一个副产物，存在效率不高等问题。



所谓分布式假设，用一句话可以表达：相同的上下文语境的词有相似含义。而由此引申出了 word2vec、fastText，在此类词向量中，虽然其本质仍然是语言模型，但是它的目标并不是语言模型本身，而是词向量，其所做的一系列优化，都是为了更快更好的得到词向量。glove 则是基于全局语料库、并结合上下文语境构建词向量，结合了 LSA 和 word2vec 的优点。上述方法得到的词向量是固定表征的，无法解决一词多义等问题，如“tie”，为此引入基于语言模型的动态表征方法：elmo、GPT、bert。

word2vec 有两种模型：Skip-Gram 和 CBOW，前者在已知 w 的情况下预测 context(w)，后者在已知 context(w)的情况下预测 w。



word2vec 支持两种优化方法：hierarchical softmax 和 negative sampling。hierarchical softmax 使用一棵二叉树表示词汇表中的单词，每个单词都作为二叉树的叶子节点。对于一个大小为 V 的词汇表，其对应的二叉树包含 V-1 非叶子节点。假如每个非叶子节点向左转标记为 1，向右转标记为 0，那么每个单词都具有唯一的从根节点到达该叶子节点的由 {0 1} 组成的代号（实际上为哈夫曼编码，为哈夫曼树，是带权路径长度最短的树，哈夫曼树保证了词频高的单词的路径短，词频相对低的单词的路径长，这种编码方式很大程度减少了计算量）。negative sampling 是一种不同于 hierarchical softmax 的优化策略，相比于 hierarchical softmax，negative sampling 的想法是直接为每个训练实例都提供负例。

3.2 DL 在文本分类领域相关的 6 篇论文

（1）Convolutional Neural Networks for Sentence Classification

主要内容：基于预先训练好的 word embedding，采用卷积神经网络（ CNN ）训练了一个 word-level 的句子分类器，并进行了一些列的实验来验证分类效果。实验证明，一个简单的 CNN 模型，只需要调整少量超参数和 word embedding，在多个标准数据集上都取得了很好的效果。根据特定的任务对 word embedding 进一步 fine-tuning，可以进一步提高分类效果。此外，还提出了一些对模型结构的进行简单修改的建议，以允许模型同时使用 task-specific embedding 和预先训练好的 static embedding。

（2）Character-level Convolutional Networks for Text Classification

主要内容：本文主要研究字符级（character-level）卷积网络（ ConvNets ）在文本分类中的应用。构建了几个大规模数据集，以证明字符级的卷积网络可以获得更好的分类结果。并与传统模型(如 bag-of-words、n - gram 及 TFIDF 变体)和深度学习模型(如基于单词的 ConvNets 和递归神经网络)进行了比较。

（3）Very Deep Convolutional Networks for Text Classification

主要内容：本文是首次将非常深度卷积网络应用于文本处理。NLP 领域使用最多的 DL 模型有递归神经网络，特别是 LSTMs 和卷积神经网络。但与计算机视觉领域的深层卷积网络（Google InceptionNet，ResNet）相比，NLP 常用的深度学习模型深度还是比较浅。本文提出了一种新的，character-level 的文本处理架构（ VDCNN ），只使用小的卷积和池化操作。实验证明，模型的性能随着深度

的增加而增加：最后达到 29 个卷积层，并在多个文本分类任务上的取得了最优的成绩。

（4）Text Classification Improved by Integrating Bidirectional LSTM with Two-dimensional Max Pooling

主要内容：递归神经网络(RNN)是自然语言处理(NLP)任务中最常用的网络结构之一，因为它的递归结构非常适合处理不同长度的文本。RNN 可以基于 word 的 embedding，把整个句文本抽取成一个矩阵。这个矩阵包括两个维度:时间步长维度和特征向量维度。现有的大多数模型通常只在时间步长维度上通过一维(1D)max-pooling 操作或基于注意力的操作来把整个整个句子转换成一个固定长度的向量。但这就存在一个问题：特征向量维上的特征向量之间并不是相互独立的，简单地在时间步长维度上单独应用 1D 的 max-pooling 可能破坏特征表示的结构。相反，在二维上应用二维(2D)pooling 操作可以获得更多对序列建模任更有意义的特征。为了整合矩阵的两个维度上的特征，本文提出使用 2D max-pooling 操作来获得文本的固定长度表示。本文还利用二维卷积对矩阵中更有意义的信息进行了采样。对情感分析、问题分类、主观性分类和新闻组分类 6 个文本分类任务进行了实验。与现有模型相比，所提出的模型在 6 个任务中的 4 个任务上取得了最优的结果。

（5）Attention-Based Bidirectional Long Short-Term Memory Networks for Relation Classification

主要内容：关系分类是自然语言处理领域的一项重要语义处理任务。但存在两个问题：1、即使是最先进的系统仍然需要依赖一些 lexical resources(如 WordNet)或 NLP 系统(如依赖与句法分析和命名实体识别)来获得高级特征。2、重要信息会出现在句子的任何位置。针对这些问题，本文提出了基于注意力机制的双向长短期记忆网络(Att-BLSTM)来捕捉句子中最重要的语义信息。在 SemEval - 2010 关系分类任务上的进行试验，结果证明该方法优于现有的大多数方法。

（6）Recurrent Convolutional Neural Networks for Text Classification

主要内容：文本分类是许多 NLP 应用中的基础任务。传统的文本分类器往往依赖于许多人为设计的特征，如字典、知识库和特殊的树结构。与传统的文本分类方法相比，本文将卷积神经网络和循环神经网络相结合，提出了一种无需人为 feature 的递归卷积神经网络。在模型结构中中，采用一种递归结构来尽可能地捕获上下文信息，学习 word 的表示，这与传统的基于窗口的神经网络相比，引入更少的噪声。还采用了一个 max - pooling 层，自动判断哪些词在文本分类中起着关键作用，以捕获文本中的关键信息。在四个常用数据集上进行了实验，实验结果表明，在多个数据集上，特别是在文档级数据集上，该方法的性能优于现有的方法。