机器学习纳米学位

毕业项目

马远超 2019-07-01

I. 问题定义

项目概述

互联网的本质是为了降低人们获取信息的成本，更便捷的进行沟通和分享。因此，从互联网创建时，就允许世界各地的人们通过互联网进行自由的交流、讨论以及合作。而像国内的贴吧，微博，微信，国外的Twitter，Facebook，Wikipedia等社区平台的建立，形成了这些互动可以发生的基础。为了人们在社区中可以更有序的交流以及促进对话，许多的社区都制定了自己的标准和规则，并防止这些社区被有毒行为劫持或摧毁。然而，随着有毒评论的黑色产业化，利益驱使人们通过各种手段来规避规范和标准，使得通过人为来执行这些规范和标准变得越来越困难。事实上Facebook正在招聘越来越多的版主来筛选可疑的内容[1]。同时，许多新闻网站现在也已经开始禁用评论功能[2]。而这些人工的审核监控机制，是非常低效的做法。

综上，我们需要一种工具来自动化地对用户评论进行监视，分类和标记。此外，不同的网站可能需要监控不同类型的内容。因此需要建立一个能够区分不同类型的言语攻击行为的模型。

我们可以看到在论文[3]中，研究人员对情感分析进行了大量研究。他们的工作重点是情绪分析，这与我们正在研究的领域非常相似。论文中定义了一种使用词袋技术预处理文本的合理方法。他们接着使用SVM和朴素贝叶斯分类器来确定推文的情绪是积极的，中性的还是负面的，并且发现朴素贝叶斯分类器更准确。此外，当他们对推文进行矢量化时，他们通过使用bigrams来提高分类器的准确性。他们的工作可以为我的benchmark model参考。

问题陈述

Toxic Comment Classification Challenge是kaggle上由Jigsaw提出的一个比赛，比赛中提供了带有多标签分类的Wikipedia评论数据，我们通过使用这份数据训练一个**文本多类型分类器**，对任意未知文本进行多标签类型（威胁，色情，侮辱和种族歧视言论等）的分类，并给出文本分别属于每个分类的概率。**这是一个文本多分类问题，并属于有监督学习**。

评价指标

我使用列平均的ROC AUC作为我的评估指标，它是单个类别预测结果ROC AUC的平均值。ROC曲线是在不同分类阈值下使用TPR和FRP绘制的图，而AUC则是ROC曲线下面积，当AUC值越大，当前的分类算法越有可能将正样本排在负样本前面，即能够更好的分类[7]。

ROC空间将假阳性率（FPR）定义为X轴，真阳性率（TPR）定义为Y轴[8]。

- TPR：真阳性率，在所有实际为阳性的样本中，被正确地判断为阳性之比率。

$$TPR = \frac{TP}{(TP + FN)}$$

- FPR：假阳性率在所有实际为阴性的样本中，被错误地判断为阳性之比率。

$$FPR = \frac{FP}{(FP + TN)}$$

将同一模型每个阈值的(FPR, TPR)座标都画在ROC空间里，就成为特定模型的ROC曲线。

AUC为ROC曲线下方的面积（Area under the Curve of ROC），它表示当随机抽取一个阳性样本和一个阴性样本，分类器正确判断阳性样本的值高于阴性样本的概率（假设阈值以上是阳性，以下是阴性）。简单说来说AUC值越大的分类器，正确率越高。

从AUC判断分类器（预测模型）优劣的标准：

- AUC = 1，是完美分类器，采用这个预测模型时，存在至少一个阈值能得出完美预测。绝大多数预测的场合，不存在完美分类器。
- 0.5 < AUC < 1，优于随机猜测。这个分类器（模型）妥善设定阈值的话，能有预测价值。
- AUC = 0.5，跟随机猜测一样（例：丢铜板），模型没有预测价值。
- AUC < 0.5，比随机猜测还差；但只要总是反预测而行，就优于随机猜测。

同时，偏差，方差，精度，召回和F1分数也将用作评估指标，以检查过度拟合和欠拟合。

II. 分析

数据探索

训练数据由Toxic Comment Classification Challenge比赛提供。数据为对恶性行为人工标注的Wikipedia评论数据，每个样本有可能被同时标注为多个类型，当所有类型的标注都为0时，表示该文本不是恶毒评论。标注的类型包括：

- toxic
- severe_toxic
- obscene
- threat
- insult
- identity_hate

比赛提供的数据由如下四个文件构成：

- train.csv - 训练集，包括159571条已进行标注的评论数据
- test.csv - 测试集，包括153164条待检测数据

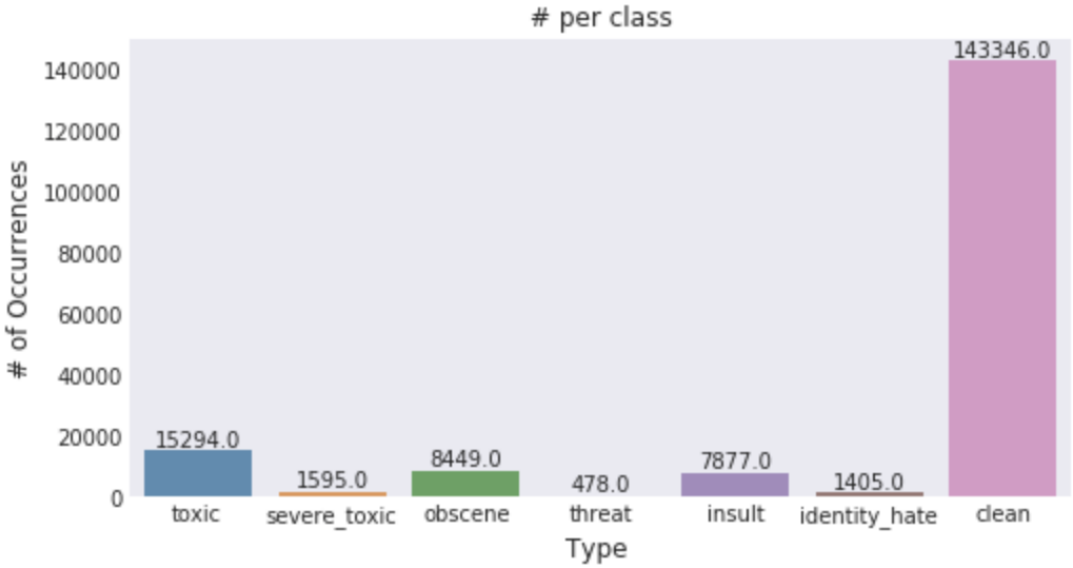
csv文件的数据格式为：

- id
- comment_text
- toxic
- severe_toxic
- obscene
- threat
- insult
- identity_hate

其中，comment_text是模型的输入。toxic，severe_toxic，obscene，threat，insult，identity_hate，如之前所诉为样本的分类标签，样本有可能同时属于多个分类。模型的输出是输入

文本被**分别**判断为每个分类（toxic，insult等）的概率。

同时，在训练集中，评论人工标注类型标签的个数分布如下图[4]，由图可见该数据集是一个非平衡的数据集。



探索性可视化

算法和技术

这个问题可以通过使用一种算法来解决，该算法将评论作为输入并输出一个概率列表，无论它是否有毒和毒性类型。文本分类最常用和简单的基线模型是朴素贝叶斯分类器和支持向量机。

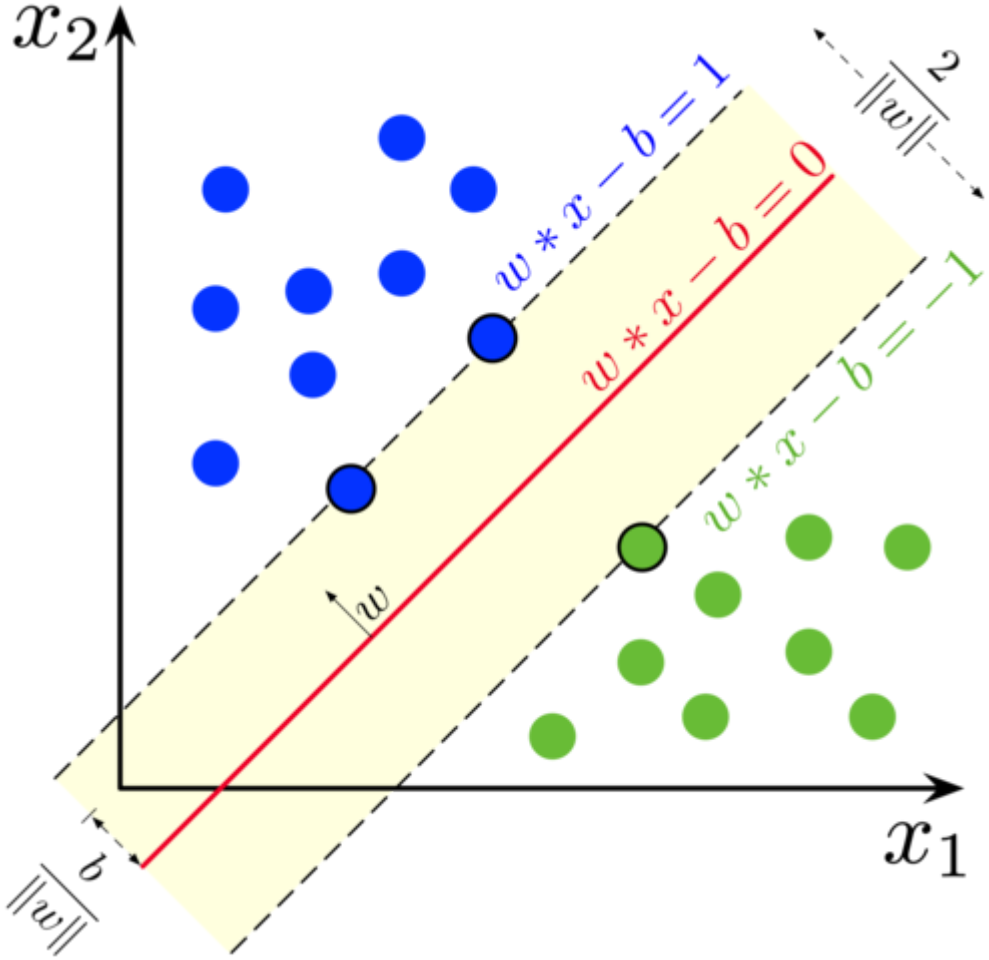
朴素贝叶斯分类器是一个基于贝叶斯规则的线性分类器：

$$P(w_j|X_i) = \frac{P(X_i|w_j) \cdot P(w_j)}{P(X_i)}$$

后验概率$P(w_j|X_i)$可以被描述为输入属于类的概率，给定输入的特征和特征的条件概率（如果它们属于特定类）。朴素贝叶斯分类器还假设每个特征独立于其他特征。

它们通过找出单词x属于某个类的概率来识别给定一组单词$(w_1..w_n)$的类$P(w_j|X_i)$的概率w对于数据集$P(X_i|w_j)$中的每个单词，并将其乘以类的概率$P(w_j)$并将其除以单词出现的概率$P(X_i)$然后最大化这种后验概率。

支持向量机是一种监督学习模型，可用于分类和回归。 SVM将每个输入样本映射为空间中的点，并构造超平面或超平面集，以便每个类的样本除以明确的间隙。然后将新样本映射到空间中并基于平面之间的间隙预测属于特定类。



结合NB和SVM分类器并且工作得相当好的算法是Naive Bayes - 支持向量机（NB SVM）[10]，它将用作基线模型。 为了解决这个问题，我们也可以使用神经网络。神经网络是一层互连的节点，其中包含类似于人类大脑神经元的激活功能。神经网络通过输入层接收输入作为数字，并将输入传递给处理它的隐藏层。隐藏层可以是多个层。每层中的每个节点都有一个权重，节点将输入与权重相乘以得到输出。然后输出层提供单个数字，该数字基于激活函数提供概率或类。通过为每个节点分配随机权重来训练网络，并且网络自动调整权重以使预测接近实际输出。

在我们的例子中，网络的输入是转换成数字形式的文本，其中每个单词输入到节点，输出层包含6个节点，表示6个输出类的概率。 然而，这些模型和其他反馈模型（如CNN）的问题是它们不跟踪顺序数据，即它们不跟踪句子中单词的上下文，并且它们长时间表现不佳文本和他们倾向于过度训练数据。 为了克服这个问题，使用了递归神经网络。在递归神经网络中，每个节点具有反馈回路，该回路在给定时间步长处获取先前输入。这允许RNN展示时间序列的时间动态行为。 RNN具有短期存储器，允许其基于节点的权重和先前的输入决策来确定当前输入。如下所示，反馈环路使用前n个输入序列的输入处理输入。

长期短期内存网络[4]是一种回归神经网络算法，是一种专为自然语言处理而设计的算法，经证明可以很好地运行，并且可以作为解决方案的基础。 LSTM网络克服了传统RNN的缺点，它们无法从长时间运行的顺序数据中获取上下文。 LSTM网络记住长时间的重要信息。 LSTM单元如下所示。每个单元有3个门：输入门，输出门和忘记门。门由S形和矢量运算表示。输入门决定需要在单元状态中存储哪些新信息。遗忘门如果不重要则删除旧信息，输出门使输入影响当前时间步的输出。Everygateisasmmoidlayert输出在0和1之间的数字，其中0表示不通过，1表示让一切通过。

LSTM单元有3个输入：输入样本x在时间步长t（x t），存储器的内存 先前的LSTM单元格（Ct -1），从前一个LSTM单元格输出（ht -1）， LSTM细胞完成的步骤： 1.前一个单元格的输出和当前输入通过 忘记图层摆脱旧信息。 2.下一层节点决定需要在单元状态中存储哪些新信息。这是通过识别要更新的值并创建新值的向量来完成的。 3.接下来，需要将旧单元状态更新为新单元状态。

这是通过将单元状态与步骤1（忘记向量）和步骤2（新值）的输出相乘来完成的。 4.然后，将输入和新存储器相乘以得到输出。

使用称为双向LSTM的LSTM形式，其具有两个网络，一个用于正常文本序列，一个用于反向序列文本。此外，在LSTM层之后使用卷积神经网络层，其从网络中提取局部特征。这个组合的灵感来自于这篇文章，它成功地使用了这个模型来提高性能。 在将数据馈送到网络之前，需要对数据进行标记化。需要将文本数据转换为数字形式，其中每个单词由数字表示。然后，该数值数据用于将每个注释转换为具有表示实际单词的索引的向量表示（单热编码）。 然而，由于每个唯一字将占据向量中的空间，因此该向量表示是稀疏且低效的。为了克服这一点，将使用将每个单词表示为向量的单词嵌入来预处理数据，并且该表示基于单词的用法，具有相似含义的单词将具有相似的表示。

单词嵌入 要将文本数据提供给神经网络，我们需要将它们标记为表示文本中每个单词的数字。常见形式，即单热编码，因为它创建了非常稀疏的矢量模型，因此非常无效。 为了克服这个问题，创建了向量空间模型，它表示向量空间中的单词，其中相似的单词彼此嵌入。 VSM取决于分布假设，该假设指出出现在相同上下文中的单词具有相似含义。 遵循这一原则的两种方法是： 1.基于频率的嵌入 - 查找数据集中每个单词的频率，为每个单词创建密集向量。 2.基于预测的嵌入。 - 使用神经网络，根据给定词汇表中每个单词的上下文创建单词嵌入。 我正在进行基于预测的嵌入，因为它更加密集和高效，并且有预先训练好的字嵌入，例如Google和stanford的GloVe [8]和Facebook的FastText [5] [6] [7]。它们都包含单词嵌入，用于训练超过一大类通用英语词汇的数百万个单词。 具体来说，我选择了FastText作为预训练的单词嵌入，因为它的表现平均比GloVe更好（尽管不是很大）。 Fasttext基于word2vec，其中每个单词被分成子单词是n-gram，最终的嵌入向量将是所有n-gram的向量之和。 使用的FastText嵌入模型在Common Crawl数据集上进行训练，该数据集包含6000亿个令牌并包含200万个单词向量。并使用基于这两篇论文的skip-gram模型创建。 完成嵌入模型后，定义模型并将数据拟合到模型中，并使用测试数据测试模型。

总结一下： 1.准备和预处理数据。在此步骤中，删除数字和 标点符号并用词干替换单词。 2.使用单词嵌入创建数据的向量表示 3.定义模型并定义模型的各个层。模特我 define接收嵌入数据作为第一层，然后是LSTM层 和CNN层。输出层具有sigmoid功能。 4.使用AUROC作为度量标准来训练和测试模型。

基准模型

SVM是最常用的文本分类算法之一，可用作基准模型。基于SVM和朴素贝叶斯算法的SVMNB[6]，它提供了比传统SVM更好的性能，是在kaggle比赛中的推荐的benchmark，我将使用SVMNB作为我的benchmark model。

Reference

1. http://fortune.com/2018/03/22/human-moderators-facebook-youtube-twitter/
2. https://www.theguardian.com/science/brain-flapping/2014/sep/12/comment-sections-toxic-moderation
3. http://crowdsourcing-class.org/assignments/downloads/pak-paroubek.pdf
4. https://github.com/udacity/cn-machine-learning/blob/master/toxic-comment-classification/pics/hist.png
5. https://www.researchgate.net/profile/Sepp_Hochreiter/publication/13853244_Long_Short-term_Memory/links/5700e75608aea6b7746a0624/Long-Short-term-Memory.pdf
6. https://nlp.stanford.edu/pubs/sidaw12_simple_sentiment.pdf
7. http://alexkong.net/2013/06/introduction-to-auc-and-roc/

8. https://zh.wikipedia.org/wiki/ROC%E6%9B%B2%E7%BA%BF
9. https://github.com/Kirupakaran/Toxic-comments-classification/blob/master/proposal.pdf
10. Baselines and Bigrams