网络数据挖掘

垃圾短信识别

XXX 20181801322290**

2018 年 12 月 30 日

目 录

1 介绍 **1**

1.1 背景 1

1.2 与其他任务的异同 1

1.2.1 图像分类 1

1.2.2 推荐系统 2

1.3 模型选择 2

1.4 章节安排 3

2 相关调研 **3**

3 文本处理 **5**

3.1 正则化 5

3.2 分词 5

3.3 词向量 6

4 特征提取 **6**

4.1 TF-IDF 6

4.2 哈希 7

5 模型介绍 **7**

5.1 随机森林 7

5.1.1 方法介绍 7

5.1.2 优缺点 8

5.2 逻辑回归 8

5.2.1 方法介绍 8

5.2.2 优缺点 11

5.3 朴素贝叶斯 11

5.3.1 方法介绍 11

5.3.2 优缺点 12

5.4 支持向量机 12

5.4.1 方法介绍 12

5.4.2 优缺点 13

5.5 卷积神经网络 13

5.5.1 方法介绍 13

5.5.2 优缺点 14

5.6 循环神经网络 15

5.6.1 方法介绍 15
5.6.2 优缺点 16
5.7 模型线性融合 16
5.7.1 方法介绍 16
5.7.2 优缺点 17

6 实验 **17**
6.1 实验设置 17
6.1.1 数据集划分 17
6.1.2 评价指标 17
6.1.3 参数设置 18
6.2 实验结果 19
6.3 Demo 展示 19
6.3.1 前端 20
6.3.2 后端 20
6.3.3 使用说明 20

7 总结及讨论 **21**
7.1 项目总结 21
7.2 模型改进 22

参考文献 **26**

1 介绍

1.1 背景

假定我们现有一封短信，其内容如下：一次价值 *xxx* 元王牌项目；可充值 *xxx* 元店内项目卡一张；可以参与 *V* 动好生活百分百抽奖机会一次！预约电话：*xxxxxxxxxx* 充斥着各种诱人的促销信息，很有可能是一封垃圾短信。现在各个网站与 App 都鼓励人们用手机号进行注册，手机号已经不再是一个非常私人的信息。据《2013 年垃圾短信报告》称，去年中国手机用户收到的垃圾短信总量超过 2000 亿条，给全国手机用户造成超过上亿元的经济损失。而伪基站自去年 7 月出现后逐渐增多，已经成为垃圾短信尤其是诈骗短信孳生的温床。

迄今为止，垃圾短信在国际上并没有一个标准的定义。其基本特征是“不请自来”，而且大部分垃圾短信都带有商业或者其他宣传目的 [1]。要解决垃圾短信问题，必须综合法律、技术等手段。反垃圾短信技术上可以分成两类: “根源阻断” 和 “存在发现”。“根源阻断” 是指通过防止垃圾短信的产生来减少垃圾短信，该方法目前还没得走向实用。目前，主流的反垃圾短信技术是 “存在发现”，即对存在的短信进行识别分类。垃圾短信识别可以通过对短信文本分类来实现。

一直以来，垃圾邮件分类（Spam）都是文本分类问题的一个经典而又重要的任务，本项目的垃圾短信分类，从本质上说和垃圾邮件分类属于同一任务，报告中垃圾短信和垃圾邮件是通用的。

1.2 与其他任务的异同

1.2.1 图像分类

图像分类是一项发展历史悠久的技术，它的基本任务可以概括为，建立一个从图像域 X 到标签域 Y 的映射，从这个抽象的角度看，图像分类问题和文本分类问题异曲同工。通常而言，解决分类问题可以分为两步，首先是对数据域 X 进行特征提取，然后利用提取到的特征进行分类。在传统方法中，这两步是分离进行的，而在深度学习方法则是统一的，通常可以将神经网络的最后一层看作分类器，其他隐层则起到特征提取的作用。

尽管抽象层次上，图像分类问题和文本分类问题是统一的，但在细节上存在诸多差异，主要集中于特征提取部分。首先，图像和文本的组成粒度不同。图像是由大量低层次的像素构成，而文本则由高层次的词语组成。另外，图像中，像素的组合呈现出空间性；而文本中，词语被组合成序列的形式，且在本质上，词语的组合是树形的，组成粒度以及组合方式的不同导致

了图像分类和文本分类特征提取的巨大差异。传统方法中，图像分类常常提取 HOG、SURF、LBP 等特征，而文本则提取如 TF-IDF，或将文本组成词袋的形式，或提取 n-gram 特征。在深度学习方法中，卷积神经网络的应用和发展为图像分类带来了爆炸性的进展，通过在图像上运行卷积神经网络，模型可以层次性地提取到从低层的纹理到高层的目标等等诸多特征。而在文本分类方面，词向量技术的引入使得文本分类可以像图像分类一样，在更低的层次进行特征提取，并且考虑词语组合方式的特殊性，循环神经网络被引入用于挖掘文本的特征。

1.2.2 推荐系统

推荐系统的起源可追溯到近 20 年前，但直到至今推荐系统依然没有一个非常精确的定义，广义上的推荐系统可以理解为是主动向用户推荐物品 (Item) 的系统，所推荐的物品可以是是音乐、书籍、餐厅、新闻条目等等，这依赖于具体的应用领域。

从某些角度，推荐问题可以形式化地定义为一个特征高度稀疏的分类问题 [2]，并利用协同过滤技术求解。而在用户和物品均有大量描述属性的情形下，用户和物品的特征可以由其高度稀疏的编号的独热编码以及其描述属性所构成，并利用混合推荐技术求解，同样地，适用于文本分类的分类器可以在这一场景下使用。除此之外，当用户或物品存在文本形式的特征时，文本分类所用到的一系列进行文本处理的技术会被推荐系统所使用。

近年来，随着深度学习技术在推荐系统中得到广泛地应用，推荐系统中的协同过滤方法愈加类似于自然语言处理领域的词向量技术，如文献 [3] 将物品和用户编号映射到低维的 Embedding 空间，并在其上构建多层感知机，文献 [4] 中，Word2vec 算法被用于用户交互过的物品序列中，用于学习物品的向量化表示。

除了这些相似点，推荐系统与文本分类还存在着更多的差异，如推荐系统所面临的冷启动问题，丰富且跨媒体的特征，如电影推荐中的电影剧照，电影描述，用户档案等，海量数据以及极高维极稀疏特征（上亿维）等等，这些都是文本分类问题所不具备的。

1.3 模型选择

垃圾短信分类可以看作一个文本分类问题，即正常短信与垃圾短信。我们考虑使用常用文本分类技术来解决该任务。因此，各种文本分类方法都可以用于垃圾短信的分类，而基本上大部分机器学习方法都在文本分类领域有所应用，如：朴素贝叶斯分类算法、KNN、SVM、最大熵和神经网络等等。

垃圾短信分类任务的特点是语料多且粗糙杂乱、非垃圾短信要远多于垃圾短信，即数据不平衡。根据以上特点，可以选择稳定性较高以及适用于大量数据不易欠拟合的模型。本项目经过尝试了 6 种算法，包括经典的机器学习算法：易于实现的逻辑回归（Logistic Regression，LR），稳定性很高的朴素贝叶斯 [5]（Naive Bayes，NB），支持向量机 [6]（Support Vector Machine，SVM），适用于大量数据的随机森林 [7]（Random Forest，RF）；和深度学习算法：特征提取能力强大的卷积神经网络 [8]（Convolutional Neural Network，CNN），能够学习长期依赖的循环神经网络 [9]（Recurrent Neural Network，RNN）。

1.4 章节安排

第一章为引言部分，介绍了当前垃圾短信泛滥的形势，提出研究垃圾短信分类任务的重要意义，该任务实际上是文本分类任务，介绍了它与其他任务的相同与不同点，选择模型的原因以及主要工作。

第二章介绍了文本分类任务的相关工作。首先介绍了经典的机器学习方法，在分类器设计方面与特征提取方面的研究现状；然后介绍了深度学习方法应用于文本分类工作现状。为本文之后使用的模型打下基础。

第三章介绍了本项目的文本预处理部分，使用正则化，分词，词向量化等方式，将原始语料处理成便于应用机器学习算法的相对规则化的数据。

第四章介绍了特征提取方式，使用了 TF-IDF 特征，哈希特征，用于经典的机器学习算法。

第五章模型部分，介绍了本项目使用的模型理论依据，共有六个模型，并对比了他们各自的优缺点。包括：经典的机器学习算法逻辑回归、朴素贝叶斯、支持向量机、随机森林，与深度学习算法 CNN 和 LSTM。

第六章实验和 Demo 部分，介绍了实验的数据集划分评价指标，各个模型参数设置，实验的结果和分析比较。并介绍了 Demo 的前后端以及使用说明。

2 相关调研

目前对于垃圾邮件识别的研究主要集中在利用邮件的内容来区分垃圾邮件，即利用文本分类技术将垃圾邮件识别问题转化为一个有监督的学习问题 [10]。国外的文本分类研究开始于 20 世纪 50 年代末，早期的文本分类需要人为定义分类规则，效果不尽如人意。随着机器学习的发展，人们将机器学习的技术应用到文本分类中，首先对已经过人工标注的文本进行特征提取，然后利用算法对文本进行自动分类。这一技术为文本分类领域带来了

突破性的进展，也促使了文本分类领域新的研究成果的不断涌现。研究者们主要聚焦于文本分类领域的两个重要方面，分类器的设计和文本特征提取。

在分类器设计方面，研究者们比较了不同的分类器在垃圾邮件识别问题上性能的差异，W.A. Awad 等人 [11] 等人在 SpamAssassin 数据集上比较了 Naive Bayes、SVM、KNN、Rough sets 等方法的性能差异，得出 Naive bayes 和 Rough sets 方法比其他方法在分类正确率上表现更好的结论；Aman Kumar Sharma [12] 在 UCI 数据集上比较了 ID3、CART、ADTree、J48 等方法在垃圾邮件分类正确率上的表现，J48 方法取得了最好结果；Bhagyashri U. Gaikwad [13] 等人将集成学习中的 Random forests 应用于垃圾邮件分类，在 csmining 的 spam-email-datasets 上进行实验，在 TPR 及 FPR 上得分较高；Xavier Carreras 等人 [14] 将 Adaboost 方法应用于垃圾邮件分类，在 PU1 数据集上进行实验，结果表明该方法比 Decision Trees 及 Naive Bayes 方法可以获得更优的 F1-score；Mathew 等人则在 SMS 短信领域进行了研究 [15]，并比较了包括贝叶斯网络、多层感知机、受限玻尔兹曼在内的超过 30 种分类器的性能，得到了多项式朴素贝叶斯分类器效果最好的结论。

在特征提取方面，陶峰等人 [16] 注意到 TFIDF 算法没有考虑到算法并没有考虑到特征词在类间的分布情况的缺陷，对 TFIDF 算法进行了改进；李猛等人 [17] 改进了信息增益算法未分析特征项在类内和类间分散程度的缺陷；王禾清 [18] 则针对互信息特征选择方法缺少词频信息的缺陷，对传统的互信息方法进行了改进，并针对二分类问题，引入了特征贡献比的概念。除此之外，研究者们还比较了不同的特征提取方法在垃圾邮件识别问题上的性能差异。赵晓丹等人 [19] 等人比较了在朴素贝叶斯分类器和 SVM 分类器下，文档频率、信息增益、互信息、卡方统计量和优势率等特征提取方法的效果，实验结果显示优势率、卡方统计量和信息增益最优。

近年来，深度学习技术也开始应用于文本分类领域，并取得了超越传统分类方法的效果。Mikolov 等人 [20] 提出了 word2vec 模型，能够在大规模的未标注语料库上训练词向量，解决了词汇鸿沟问题的同时，还使得大规模语料库的信息能够传递至小规模数据集上。Petters 等人 [21] 则提出了 ELMO，利用了双向的长短期记忆网络来训练语言模型，并利用在大规模的未标注语料库上训练得到的预训练语言模型作为下游任务的特征提取模型，大大提升了文本分类以及其他自然语言处理领域下游任务的性能。在文本分类的模型架构设计方面，Kim [8] 将卷积神经网络应用于文本分类，Zhang 等人 [22] 等人提出了一个字级别的卷积神经网络，能够利用字级别的信息并解决 OOV 问题，在实验上超过了传统方法以及其他基于深度学习的方法。

3 文本处理

本任务使用的原始语料是是非格式化数据，比较杂乱，粗糙，同时中文文本没有间隔符，难以直接进行研究使用。需要进行文本预处理，使用一些正则化方法来将原始语言转换为规则的、易于处理的文本，然后进行分词，同时深度学习需要将文本变为词向量 [23]。同时中文编码不是 utf-8，需要再打开文件的时候以 utf-8 格式打开。

3.1 正则化

原始语料比较粗糙，包含繁体字符、数字、中文标点，英文单词大小写都有。需要进行正则化处理，调用 python 中的 re 模块。

1. 繁简体正则化：使用繁简体字符对照表，将繁体字符转化为简体字。
2. 标点符号正则化：利用建立的标点对照表将中文字符转化为英文字符，同时去掉连续的空格，与连续的点。
3. 英文字符正则化：将大写字母转化为小写。

3.2 分词

中文文本分类时，由于词语间没有分隔，需要分词才能提取文本特征词语。现有的中文分词方法大致可分为基于词典、基于统计和基于自然语言理解三类方法。

目前已经有可用性比较好的分词解决方案，大多使用机器学习算法和词典相结合，一方面能够提高分词准确率，另一方面能够改善领域适应性。本项目的中文分词方案采用结巴分词 Python 版[1]，结巴分词提供三种分词模式：

1. 精确模式，试图将句子最精确地切开，适合文本分析；
2. 全模式，把句子中所有的可以成词的词语都扫描出来，速度非常快，但是不能解决歧义；
3. 搜索引擎模式，在精确模式的基础上，对长词再次切分，提高召回率，适合用于搜索引擎分词。

采用精准模式，可以较好满足本项目需求。

[1] https://github.com/fxsjy/jieba

3.3 词向量

为了使用深度学习来处理文本分类问题，使用词向量，将文本映射到向量，首先尝试了开源的中文词向量[2]，不过 OOV（Out Of Vocabulary，词典中不存在的词）接近一半。又更换了腾讯 AI 实验室今年开源的中文词向量[3]虽然该词向量很大，但是包含了很多短语（将近 10 字），无法有效匹配实际文本，依然存在 3 成的 OOV。因为原始语料足够大，本项目使用 word2vec 训练词向量。

word2vec [24] 是 Google 在 2013 年年中开源的一款将词表征为实数值向量的高效工具，采用的模型有 CBOW （Continuous Bag-Of-Words，即连续的词袋模型) 和 Skip-Gram 两种。word2vec [4] 通过训练，可以把对文本内容的处理简化为 K 维向量空间中的向量运算，而向量空间上的相似度可以用来表示文本语义上的相似度。

为了更好的表示文本信息，没有去掉停用词，使用全部文本训练 word embedding，不存在 OOV 问题，而且经过测试，可以用向量空间的相似度来表示词语语义相似度

4 特征提取

4.1 TF-IDF

TF-IDF 是一种加权技术，采用的是统计方法，根据字词在文本中出现的次数和在整个语料中出现的文档频率来计算一个字词在整个语料中的重要程度。它的主要思想是：如果某个词或短语在一篇文章中出现的频率高，并且在其他文章中很少出现，则认为此词或者短语具有很好的类别区分能力，适合用来分类。

在 TF-IDF 中，TF（Term Frequency）表示的是某个关键词在整篇文章中出现的频率。IDF（InversDocument Frequency）表示逆文本频率。指某个关键词在整个语料所有文章中出现的次数，主要用于降低所有文档中一些常见却对文档影响不大的词语的作用。

TF-IDF 十分的简单快捷，能过滤掉一些常见的却无关紧要的词语，同时保留影响整个文本的重要词语。

[2]https://github.com/Embedding/Chinese-Word-Vectors
[3]https://ai.tencent.com/ailab/nlp/embedding.html
[4]https://radimrehurek.com/gensim/models/word2vec.html

4.2 哈希

在大规模的文本处理中，由于特征的维度对应分词词汇表的大小，所以维度可能非常恐怖，因此需要进行降维，最常用的文本降维方法是 Hash Trick。

在 Hash Trick 里，会定义一个特征 Hash 后对应的哈希表的大小，这个哈希表的维度会远远小于总词汇表的特征维度，因此可以看成是降维。具体的方法是，对应任意一个特征，用 Hash 函数找到对应哈希表的位置，然后将该特征对应的词频统计值累加到该哈希表位置。

整体来讲，Hash Trick 可以作为一种降维方法，实现简单，所需计算量小，效果较好。此外，Hash Trick 可以保持原有特征的稀疏性（preserve sparsity）。

5 模型介绍

5.1 随机森林

5.1.1 方法介绍

上世纪八十年代 Breiman 等人 [25] 发明分类树的算法，通过反复二分数据进行分类或回归，计算量大大降低。2001 年 Breiman [7] 把分类树组合成随机森林（RF），即在变量（列）的使用和数据（行）的使用上进行随机化，生成很多分类树，再汇总分类树的结果。随机森林在运算量没有显著提高的前提下提高了预测精度。随机森林对多元公线性不敏感，结果对缺失数据和非平衡的数据比较稳健，可以很好地预测多达几千个解释变量的作用，被誉为当前最好的算法之一 [26]。

随机森林顾名思义，是用随机的方式建立一个森林，森林里面有很多的决策树组成，随机森林的每一棵决策树之间是没有关联的。在得到森林之后，当有一个新的输入样本进入的时候，就让森林中的每一棵决策树分别进行一下判断，看看这个样本应该属于哪一类，然后看看哪一类被选择最多，就预测这个样本为那一类。随机森林可以既可以处理属性为离散值的量，比如 ID3 算法，也可以处理属性为连续值的量，比如 C4.5 算法。另外，随机森林还可以用来进行无监督学习聚类和异常点检测。

决策树（decision tree）是一个树结构（可以是二叉树或非二叉树）。其每个非叶节点表示一个特征属性上的测试，每个分支代表这个特征属性在某个值域上的输出，而每个叶节点存放一个类别。使用决策树进行决策的过程就是从根节点开始，测试待分类项中相应的特征属性，并按照其值选择输出分支，直到到达叶子节点，将叶子节点存放的类别作为决策结果。

基尼系数（GINI）是常用的特征选择评价指标。基尼系数选择的标准就是每个子节点达到最高的纯度，即落在子节点中的所有观察都属于同一个分类，此时基尼系数最小，纯度最高，不确定度最小。

对于一般的决策树，假如总共有 K 类，样本属于第 k 类的概率为：P_k，则该概率分布的基尼指数为：

$$Gini(P) = \sum_{k=1}^{K} P_k(1 - P_k) = 1 - \sum_{k=1}^{K} P_k^2 \tag{1}$$

5.1.2 优缺点

优点： 随机森林不易过拟合，可能比 Bagging 和 Boosting 更快。由于在每次划分时只考虑很少的属性，因此它们在大型数据库上非常有效。有很好的方法来填充缺失值，即便有很大一部分数据缺失，仍能维持很高准确度。给出了变量重要性的内在估计，对于不平衡样本分类，它可以平衡误差。可以计算各实例的亲近度，对于数据挖掘、检测离群点和数据可视化非常有用。

随机森林被证明对大规模数据集和存在大量且有时不相关特征的项（item）来说很有用

缺点： 随机森林在某些噪声较大的分类和回归问题上会过拟合。对于有不同级别的属性的数据，级别划分较多的属性会对随机森林产生更大的影响，所以随机森林在这种数据上产生的属性权值是不可信的。

5.2 逻辑回归

5.2.1 方法介绍

线性模型可以进行回归学习，但我们要做的是垃圾短信分类，是一个分类任务，需要用到广义线性模型. 我们用一个单调可微函数将分类任务的真实标记 y 与线性回归模型的预测值联系起来。

针对垃圾短信分类这样一个二分类任务，其输出标记 $y \in \{0, 1\}$，0 对应正常短信，1 对应垃圾短信。线性回归模型产生的预测值 $z = \mathbf{w}^T\mathbf{x} + \mathbf{b}$ 是实值，于是我们需将实值 z 转换为 0/1 值. 最理想的是” 单位阶跃函数”：

$$y = \begin{cases} 0 & z < 0; \\ 0.5 & z = 0; \\ 1 & z > 0, \end{cases} \tag{2}$$

即若预测值 z 大于零就判为正例小于零则判为反例预测值为临界值零则可任意判别，如下图所示.

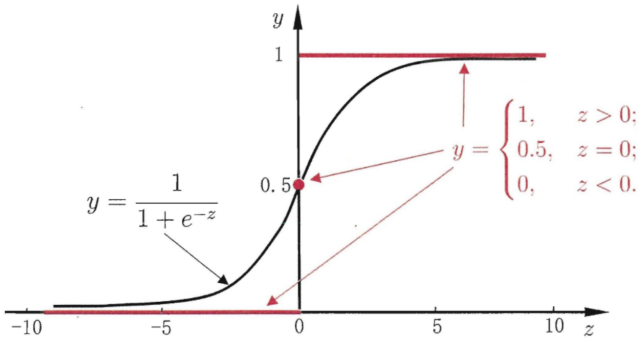

图 1: 单位阶跃函数与 Logistic 函数

但从上图可看出，单位阶跃函数不连续，因此不能直接用作广义线性模型的联系函数. 于是我们希望找到能在一定程度上近似单位阶跃函数的" 替代函数"，并希望它单调可微. 对数几率函数 (logistic function) 正是这样一个常用的替代函数:

$$y = \frac{1}{1 + e^{-z}} \tag{3}$$

从图中可以看出，Logistic 函数是一种 "Sigmoid" 函数，它将 z 值转化为一个接近 0 或 1 的 y 值，并且他的输出值在 $z = 0$ 附近变化很陡，由 $z = w^T x + b$，我们得到

$$y = \frac{1}{1 + e^{-(\mathbf{w}^T \mathbf{x} + b)}}, \tag{4}$$

将其变化为

$$\ln \frac{y}{1 - y} = \mathbf{w}^T \mathbf{x} + b \tag{5}$$

将 y 视为样本 $\mathbf{x}$ 作为正例的可能性，则 $1 - y$ 是反例可能性，两者的比值 $\frac{y}{1-y}$ 称为 "几率"，反应了 $\mathbf{x}$ 作为正例的相对可能性。而对几率取对数就得到 "对数几率":

$$\ln \frac{y}{1 - y} \tag{6}$$

由此可看出，式 (5) 实际上是在用线性回归模型的预测结果去逼近真实标记的对数几率，因此，其对应的模型称为" 对数几率回归"（或称 "逻辑回归"，logistic regression). 虽然它的名字是" 回归"，但实际却是一种分类学习方法. 这种方法有很多优点，例如它是直接对分类可能性进行建模，无需事先假设数据分布，这样就避免了假设分布不准确所带来的问题; 它不是仅预测出" 类别"，而是可得到近似概率预测，这对许多需利用概率辅助决策

的任务很有用; 此外，对率函数是任意阶可导的凸函数，有很好的数学性质，现有的许多数值优化算法都可直接用于求取最优解.

将 y 视为类后验概率估计 $p(y=1|\mathbf{x})$, 则式（5）可以重写为:

$$\ln \frac{p(y=1|\mathbf{x})}{p(y=0|\mathbf{x})} = \mathbf{w}^T\mathbf{x} + b \tag{7}$$

显然有

$$p(y=1|\mathbf{x} = \frac{e^{-(\mathbf{w}^T\mathbf{x}+\mathbf{b})}}{1+e^{-(\mathbf{w}^T\mathbf{x}+b)}} \tag{8}$$

$$p(y=0|\mathbf{x} = \frac{1}{1+e^{-(\mathbf{w}^T\mathbf{x}+b)}} \tag{9}$$

于是我们可以通过极大似然法来估计 $\mathbf{w}$ 和 b, 给定数据集 $\{(\mathbf{x_i}, \mathbf{y_i})\}(i = 1, 2, ..., m)$,Logistic regression 模型最大化 “对数似然”:

$$l(\mathbf{w}, b) = \sum_{i=1}^{m} \ln p(y_i|\mathbf{x}_i; \mathbf{w}, b), \tag{10}$$

即令每个样本属于其真实标记的概率越大越好. 为便于讨论, 令 $\boldsymbol{\beta} = (\mathbf{w}, b), \hat{\mathbf{x}} = (\mathbf{x}; 1)$, 则 $\mathbf{w}^T\mathbf{x} + b$ 可简写为 $\boldsymbol{\beta}^T\hat{\mathbf{x}}$, 再令 $p_1(\hat{\mathbf{x}}, \boldsymbol{\beta}) = p(y=1|\hat{\mathbf{x}}, \boldsymbol{\beta}), p_0(\hat{\mathbf{x}}, \boldsymbol{\beta}) = p(y=0|\hat{\mathbf{x}}, \boldsymbol{\beta}) = 1 - p_1(\hat{\mathbf{x}}, \boldsymbol{\beta})$, 则式（10）中的似然项可重写为

$$p(y_i|\mathbf{x}_i; \mathbf{w}, b) = y_i p_1(\hat{\mathbf{x}}_i; \boldsymbol{\beta}) + (1 - y_i) p_0(\hat{\mathbf{x}}_i; \boldsymbol{\beta}), \tag{11}$$

将式（11）代入（10），并根据式（4）和（5）可知，最大化（(10) 等价于最小化

$$l(\boldsymbol{\beta}) = \sum_{i=1}^{m} (-y_i \boldsymbol{\beta}^T \hat{\mathbf{x}}_i + \ln(1 + e^{\boldsymbol{\beta}^T \hat{\mathbf{x}}_i})) \tag{12}$$

加入正则项 $\frac{1}{C}||\boldsymbol{\beta}||_2$, 上式变为:

$$l(\boldsymbol{\beta}) = \sum_{i=1}^{m} (-y_i \boldsymbol{\beta}^T \hat{\mathbf{x}}_i + \ln(1 + e^{\boldsymbol{\beta}^T \hat{\mathbf{x}}_i})) + \frac{1}{C}||\boldsymbol{\beta}||_2 \tag{13}$$

式（13）是关于 $\boldsymbol{\beta}$ 的高阶可导连续凸函数，根据凸优化理论，经典的数值优化算法梯度下降法、牛顿法、拟牛顿法等都可求得其最优解，于是得到

$$\boldsymbol{\beta}^* = \arg\min_{\boldsymbol{\beta}} l(\boldsymbol{\beta}) \tag{14}$$

5.2.2 优缺点

优点： 1）适合需要得到一个分类概率的场景。2）计算代价不高，容易理解实现。LR 在时间和内存需求上相当高效。它可以应用于分布式数据，并且还有在线算法实现，用较少的资源处理大型数据。3）LR 对于数据中小噪声的鲁棒性很好，并且不会受到轻微的多重共线性的特别影响。

缺点： 1）容易欠拟合，分类精度不高。2）数据特征有缺失或者特征空间很大时表现效果并不好。

5.3 朴素贝叶斯

5.3.1 方法介绍

朴素贝叶斯是一系列以假设特征之间强（朴素）独立下运用贝叶斯定理为基础的简单概率分类器。Sahami 最早提出了把贝叶斯分类算法应用在垃圾邮件过滤 [5]，并且自此朴素贝叶斯一直是垃圾邮件分类的一种基准方法。贝叶斯分类器的思想基础：对于给出的待分类项，求解在此项出现的条件下各个类别出现的概率，哪个最大，就认为此待分类项属于哪个类别。

朴素贝叶斯分类算法 [27] 采用了变量独立假设的最初形式，也是限制最多的一种形式，它假设每个特征变量 X_i 在给定类别变量 C 下都是独立的。

朴素贝叶斯分类正式定义 [28] 如下：

1. 设 $x = \{a_1, a_2, \ldots, a_m\}$ 为一个待分类项，而每个 a 为 x 的一个特征属性
2. 类别集合，本任务只有两个类别 $C = \{y_1, y_2\}$
3. 计算 $P(y_1|x), P(y_2|x)$
4. 如果 $P(y_k|x) = \max\{P(y_1|x), P(y_2|x)\}$

关键就是如何计算第 3 步中的各个条件概率，给定已知分类的待分类集合，即训练样本集；统计得到在各类别下各个特征属性的条件概率估；如果各个特征属性是条件独立的，则有如下推导：

$$P(y_i|x) = \frac{P(x|y_i)P(y_i)}{P(x)} \tag{15}$$

分母对于所有类别为常数，因为我们只要将分子最大化皆可。

朴素贝叶斯的优势在于只需要根据少量的训练数据估计出必要的参数（变量的均值和方差）。由于变量独立假设，只需要估计各个变量的方法，而不需要确定整个协方差矩阵。

在 scikit-learn [29] 中开放了 3 个朴素贝叶斯的分类算法类，分别是 GaussianNB，MultinomialNB 和 BernoulliNB。其中 GaussianNB 就是先验为高斯分布的朴素贝叶斯，在样本特征分布连续时表现较好，MultinomialNB 就是先验为多项式分布的朴素贝叶斯，样本特征的分大部分是多元离散值，而 BernoulliNB 就是先验为伯努利分布的朴素贝叶斯，常用于二元离散值或者很稀疏的多元离散值。本项目使用词袋作为特征，去除了停用词和出现次数小于 3 的词，是离散特征，因而采用先验为多项式分布的 MultinomialNB。

5.3.2 优缺点

优点： 朴素贝叶斯分类器稳定性很高效率很高，对缺失数据不太敏感，算法易于实现。算法效率很高，复杂度低，同时存储资源低，模型易于训练，超参数很少。

缺点： 朴素贝叶斯独立性假设过强，本任务提取的特征为词袋，很难保证词袋各个特征完全独立。需要知道先验概率，且先验概率很多时候取决于假设，本任务假设服从多项式分布。

5.4 支持向量机

5.4.1 方法介绍

支持向量机（support vector machine，SVM）是一种基于统计学习理论的模式识别方法，目前被广泛用于模式识别、文本分类以及生物信息学等多个方面。最初是被用于二分类问题，现在被广泛用于高维非线性的分类问题。它通过构造最优超平面来进行分类，它是一种特征空间上的间隔最大的线性分类器，其学习策略是间隔最大化，最终可转化为凸二次规划问题的求解。

对于线性可分得情况，它的目标是求得参数使间隔距离最大，以得到最优超平面的方程，并据此对样本进行预测分类。对于非线性可分的情况，SVM 的处理方法是选择一个核函数，它的原理是用内积函数定义的非线性变换，首先将低维空间中的点映射到高维空间中，使它们在这个高维空间中线性可分，然后再使用线性划分的原理判断分类边界，在这个空间中求 (广义) 最优分类面。此外，核函数能够使得高维空间中的所有运算都可以在低维空间中进行，从而省去了计算映射的过程，也避免了增加运算复杂度。

垃圾邮件过滤任务是线性不可分的情况，使用的 SVM 为线性支持向量机。

线性支持向量机是在线性可分支持向量机基础上推广，它是针对的是

线性不可分的问题。所谓的线性不可分指的是空间上有两类，在空间找不到任何的分割平面能把这两类绝对完全分开，总会有一些落到决策边界内或被分错。这时按照线性可分支持向量求解就找不到合适的超平面来将两类有效的区分开来，针对这类问题可以使用线性支持向量机——在合理的误分类条件下，找到边缘距离较大且误分点到边缘距离较小的超平面，对原本线性可分支持向量机的约束做松弛，以适应放非线性可分数据，这时我们引入松弛变量 ξ 来实现，如下式所示：

$$y_i(w \cdot x_i + b) \geq 1 - \xi_i \tag{16}$$

这里的 ξ_i 就是松弛变量，也就是说对于每一个分错的点，都要付出一个代价，此外，对于任意 ξ_i 都有 $\xi_i \geq 0$。

5.4.2 优缺点

优点： SVM 使用核函数可以向高维空间进行映射，进而可以解决非线性的分类，当核函数已知时，可以简化高维空间问题的求解难度。它的分类思想很简单，就是将样本与决策面的间隔最大化，但却具有较好的分类效果。

缺点： SVM 对大规模数据训练比较困难。且无法直接解决多分类问题，只能用用间接的方法来做，但本任务是二分类的，所以不存在这一问题。

5.5 卷积神经网络

5.5.1 方法介绍

卷积神经网络（CNN）被广泛应用于计算机视觉领域，Kim 等人 [8] 将 CNN 应用于文本分类，并在该领域取得了 state-of-the-art 的效果，从此 CNN 在自然语言处理领域获得了广泛的关注。本次实验中，我们选用了 zhang 等人 [30] 所提出的模型架构用于垃圾短信识别。

在经过文本预处理步骤后，垃圾短信被表示为一连串的词语。首先，我们将其转换为句子矩阵，其中每一行是每个词语的向量化表达，我们采用预训练的 word embedding 来初始化这一向量表示。假设 d 为词向量的维度，且句子长度为 s，则句子矩阵的维度为 $s \times d$。在这个句子矩阵上，我们利用卷积神经网络来提取与垃圾短信相关联的特征用于分类。假设卷积核的权值矩阵为 $\mathbf{w}$，其宽度为 h。由于自然语言的序列性质，我们采用 1D 卷积核，则 $\mathbf{w} \in \mathcal{R}^{h \times d}$。令句子矩阵 $\mathbf{A} \in \mathcal{R}^{s \times d}$，并且用 $\mathbf{A}[i:j]$ 来代表 $\mathbf{A}$ 来代表 $\mathbf{A}$ 从 i 行到 j 行的子矩阵。卷积操作的输出序列 $o \in \mathcal{R}^{s-h+1}$ 由公式(17)计算

获得：

$$o_i = \mathbf{w} \cdot \mathbf{A}[i : i + h - 1] \tag{17}$$

其中 $i = 1...s - h + 1$，我们对每一个 o_i 添加偏置项 $b \in \mathcal{R}$ 以及激活函数 f，则该卷积核产生的特征向量 $c \in \mathcal{R}^{s-h+1}$ 为：

$$c_i = f(o_i + b) \tag{18}$$

在实验中，为了提取更丰富的特征，我们采用了许多不同长度的卷积核。

提取的特征向量的维度依赖于 h 和 s，由于不同句子的长度不用，最终产生的特征向量的维度是不确定的，这将对进一步的分类器的构建造成困难；另一方面，众多卷积核提取的特征向量维度过高，不利于分类器选取显著特征。为了解决这些问题，我们对每个卷积核提取的特征向量进行 1-max 池化 [31]，通过这种操作，我们可以选取出每个卷积核提取出的最显著的特征，再这之后，我们将池化后选出来的显著特征进行拼接，得到了长度为 n 的特征 z，其中 n 为卷积核的个数。并将这一特征送入一个全连接的输出层，进行二分类，如公式(19)所示：

$$\hat{y} = \sigma(W^o z + b) \tag{19}$$

其中 $\hat{y}$ 为标签的估计值，W^o 和 b 为输出层的参数，σ 为 sigmoid 函数，z 为 1-max 池化输出的特征向量。

模型使用交叉熵损失函数，并利用 Adam [32] 进行优化。为了防止过拟合，模型在输出层之前以及词嵌入层之后使用了 Dropout [33]，这种技术按照预设的概率随机地将矩阵的一些值设为零，除此之外我们还使用了 L2 正则化。

5.5.2 优缺点

优点： 卷积神经网络的优势在于对于 n-gram 特征强大的提取能力，通过堆叠多层卷积层，模型可以学习到文本高层次的特征，除此之外，卷积网络可以通过并行计算来提高计算速度。

缺点： 相比于循环神经网络，卷积神经网络更难学习到文本中的长期依赖性。

5.6 循环神经网络

5.6.1 方法介绍

循环神经网络 (RNN) 主要解决序列数据的处理，比如文本、语音、视频等等。这类数据的样本间存在顺序关系，每个样本和它之前的样本存在关联。比如说，在文本中，一个词和它前面的词是有关联的；在气象数据中，一天的气温和前几天的气温是有关联的。不同于传统的前向反馈神经网络 (Feed-forward Neural Networks，FNNs)，RNN 引入了定向循环，能够处理那些输入之间前后关联的问题。因此能够很好地学习到其中包含的信息依赖关系，当前，RNN 已经在自然语言处理 (Natural Language Processing, NLP) 等领域中取得了巨大成功以及广泛应用。

在经过简单的文本预处理后，对其中的标点和繁体字符进行了相应的转换。在实验中发现，相比于使用 Jieba 对短信内容进行分词，直接使用一层 RNN 网络来学习词语级别信息能够更好地对短信内容特征进行抽取。因此，我们直接对文本进行字符级别的处理，首先对训练集中短信文本进行单字拆分并对其中字符进行统计，获取其中频度最高的前 k 个构建词汇表 V。接着使用随机初始化训练得到的 word embedding 对表示词汇表中词汇 e，设其词向量维度为 d。在对短信文本进行填充对齐后，固定其长度为 l，由此将短信内容表示为 $l\times$ 的矩阵 M。接着，我们将句子矩阵 M 输入至两层 RNN 网络中，对应字到词、词到句的特征抽取。网络中使用 LSTM 核，若前一时刻状态为 C_{t-1}，f_t 为遗忘 $t-1$ 时刻内容的概率，则可由公式(20)得到当前状态 C_t

$$C_t = C_{t-1} * f_t + i_t * \hat{C}_t \tag{20}$$

其中 i_t 和 $\hat{C}_t$ 分别为使用 Sigmod 和 tanh 作为激活函数得到的当前时刻信息。

同时，当前时刻的输出 h_t 可由 $t-1$ 时刻输出 h_{t-1} 和当前时刻输入 e_t 得到，其计算方法如公式(21)：

$$h_t = Sigmod(W[h_{t-1}, e_t] + b) * tanh(C_t) \tag{21}$$

之后，我们将 RNN 网络得到的短信信息表达输入到 ReLU 激活函数中，并通过 Softmax 函数实现对短信文本的二分类。

模型也使用交叉熵作为损失函数，并且在每层 RNN 网络和 RelU 激活函数之后都使用 Dropout 来防止模型过拟合。

5.6.2 优缺点

优点： 循环神经网络最大的优点在于能够学习到句子中存在的长期依赖性，而 LSTM 模型中遗忘门结构能够去除其中的无效信息，由此能够对一些较长且隐蔽的短信文本进行正确分类。

缺点： RNN 训练时无法并行操作，导致训练速度相比于 CNN 更慢，同时其对带宽内存等资源消耗较大，系统调参方面有着较大的限制。

5.7 模型线性融合

5.7.1 方法介绍

首先我们通过一个例子证明集成能取得好的效果，如下图所示，平面上分布着一些待分类的点。如果要求只能用一条水平的线或者垂直的线进行分类，那不论怎么选取直线，都达不到最佳的分类效果。但是，如果可以使用集体智慧，比如一条水平线和两条垂直线组合而成的图中折线形式，就可以将所有的点完全分开，得到了最优化的预测模型。

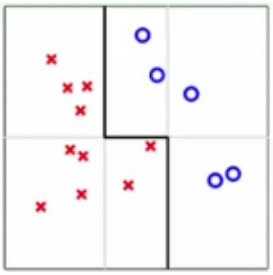

图 2: 集成示意图

这个例子表明，通过将不同的假设以一定方式结合起来，得到了比单一假设更好的预测模型。这就是集成的优势所在，它拓展了模型的复杂度，提高了预测模型的能力，起到了特征转换的效果。

本文采取的线性融合的方法，对于每一个分类器 g_t，给出它的权重 α_t，线性融合模型如下所示：

$$G(x) = sign\left(\sum_{t=1}^{T} \alpha_t g_t(x)\right) \ , \ \alpha_t \geq 0 \tag{22}$$

对于 α_t 的选择，本文采用 Logistic regression 模型，将不同模型在训练集上的预测结果作为 LR 集成模型的输入，拟合训练集的真实的 Lable，学习得到不同模型的参数，然后在验证集做实验，得到验证集上的评价指标；对于所需要集成的模型，本文选择在验证集上表现最好的三个模型 CNN、

LSTM、Random Forests 作为分类器。本文在实验过程中发现加入其他模型如 SVM、Naive Bayes 等在验证集上表现较差的模型，LR 集成模型学得的其他模型的权重为负值或者与这三者相比权重较小，加入表现差的模型后，整体的集成结果也变差。采用表现最好的 CNN、LSTM、Random Forests 作为分类器，通过学习三者的权重，集成模型在验证集上的表现 (垃圾短信作正例时的 F1-score 与非垃圾短信作正例时的 F1-score) 优于所有单个分类器模型。

5.7.2 优缺点

优点： 1）集成 models 有助于防止欠拟合（underfitting)。它把所有比较弱的分类器 g(t) 结合起来，利用集体智慧来获得比较好的模型 G 。集成就相当于是特征转换，来获得复杂的学习模型。2）集成 models 有助于防止过拟合（overfitting)。它把所有分类器 g(t) 进行组合，容易得到一个比较中庸的模型，类似于 SVM 的最大间隔一样的效果，从而避免一些极端情况包括过拟合的发生。从这个角度来说，集成起到了正则化的效果。由于集成具有这两个方面的优点，所以在实际应用中集成 models 都有很好的表现。

缺点： 1）增加了一次训练过程；2）需要所有其他单模型的分类结果。

6 实验

6.1 实验设置

6.1.1 数据集划分

本次实验中，我们采用五折交叉验证来划分数据集，其中 80% 的数据作为训练集，10% 的数据作为验证集进行超参数调整，剩余 10% 的数据作为测试集。所有结果均为五折测试集结果的平均值。

6.1.2 评价指标

由于垃圾短信数据集中，垃圾短信类别和非垃圾短信类别的样本数目不平衡，我们采用了精确率（Precision)，召回率（Recall）和 F1 来评估各

种方法的分类效果。

$$
\begin{aligned}
Precision &= \frac{TP}{TP+FP} \\
Recall &= \frac{TP}{TP+FN} \\
F1 &= 2 \cdot \frac{Precision \times Recall}{Precision + Recall}
\end{aligned} \tag{23}
$$

其中给定类别，TP 是预测为该类别且真实为该类别的样本数目，FP 是预测为该类别且真实不是该类别的样本数目，FN 为预测为不是该类别且真实为该类别的样本数目。

6.1.3 参数设置

随机森林（RF） 我们首先从句子中抽取 3000 维的 TF-IDF 特征，并将其输入到随机森林模型中进行训练。其中，随机森林模型采用随机且有放回地抽样，树的数量为 100，使用基尼系数作为特征选择的评价标准，并使用 Out-of-Bag(OOB) 来估计泛化误差。

逻辑斯蒂回归（LR） 采用拟牛顿方法中的 lbfgs 方法，正则项系数 C 设置为 1。

朴素贝叶斯（NB） 实现时使用的是 sklearn 的 naive_bayes 库的 MultinomialNB，并设置平滑项参数 α 为 1。

支持向量机（SVM） 考虑到训练速度的问题，在具体实现时使用的是 sklearn 的 liblinear 库中的 LinearSVC：对错误分类的惩罚的松弛变量为 0.6，停止训练的误差值大小 1e-4，损失函数选择 hinge loss。

卷积神经网络（CNN） 经过超参数搜索后，我们在数据集上训练了 word2vec，并将之用于 embedding layer 的初始值。除此之外，我们还随机初始化了 300 维的 word embedding，并将其与预训练的 word embedding 进行连接。word2vec 我们使用了 gensim[5]的实现，并设置为 gensim 库的默认参数。我们设置 dropout 为 0.1，L2 正则化比例为 1e-5, 卷积核个数为 150，使用大小为 2,3,4,5 的卷积核，学习率设置为 2e-5，并过滤出现次数小于 3 的词语。

循环神经网络（RNN） 由于从字符层面对短信文本进行分词，得到词汇表大小仅为 5000，所以我们将词向量维度设置为 64，以提高训练速度，同时减小资源消耗。另外，在进行参数调整优化后，我们将句子长度设置为 300，学习率设置 0.001，dropout 比例设置为 0.1，隐藏层神经元个数设置为 128。

[5]https://radimrehurek.com/gensim/index.html

表 1: 各方法对垃圾短信识别效果的比较

模型	垃圾短信			非垃圾短信		
	Precision	Recall	F1	Precision	Recall	F1
CNN	**0.9976**	0.9895	0.9935	0.9988	**0.9997**	**0.9993**
RNN	0.9954	0.9668	0.9809	0.9963	0.9995	0.9979
NB	0.8665	0.9807	0.9201	0.9978	0.9831	0.9904
RF	0.9957	0.9828	0.9892	0.9981	0.9995	0.9988
LR	0.9739	0.9608	0.9673	0.9956	0.9971	0.9964
SVM	0.9959	0.9839	0.9899	0.9982	0.9995	0.9989
Ensemble	0.9971	**0.9905**	**0.9938**	**0.9989**	**0.9997**	**0.9993**

集成模型（Ensemble） 训练 CNN、LSTM、RF 模型的权重时用 logistic regression 方法, 采用拟牛顿方法中的 lbfgs 方法, 正则项系数 C 设置为 0.05。

6.2 实验结果

各方法在垃圾短信识别数据集上的结果如表1所示，由表1的结果可以看到：一方面，绝大多数模型在该任务上都达到了近乎完美的分类效果，即使是简单的逻辑斯蒂回归模型，在垃圾短信类别和非垃圾短信类别的 F1 值也分别超过了 0.967 和 0.996，这说明垃圾短信分类任务是一种较为简单的文本分类任务；另一方面，垃圾短信类别的分类效果均落后于非垃圾短信类别，这主要是由于数据集的不平衡引起的。尽管试验中采用了一定程度的措施来缓解数据不平衡问题，但依然无法完全避免这一现象的发生。

在单模型中，CNN 取得了最好的效果，这证明了深度学习方法在垃圾短信分类任务的强大能力，也应证了 [8, 30] 等的研究结果。深度学习方法中，CNN 的分类效果超越了基于字符的 RNN，这可能表明在垃圾短信识别领域，相比于捕获语言的长期依赖性，模型提取 n-gram 特征的能力更为重要，也从侧面表明垃圾短信识别任务并不需要对文本的深层次理解。经典模型当中，SVM 和 RF 的效果几乎相当，SVM 略胜一筹。

尽管只采用了简单的线性集成方法，集成模型的效果依然超过了所有的单模型。这一结果是令人鼓舞的，这表明在该数据集上，模型的效果依然有改进的潜力。

6.3 Demo 展示

Demo 系统采用前后端分离的方式编写。前端负责接收用户的输入并传送给后端，后端经过解析处理将结果返回给前端，最后前端将结果展示给用

户。前后端之间通过 JSON 进行数据交换，这种解耦合的方式让整个系统具有极大的灵活性。

6.3.1 前端

前端采用 Vue[6]框架编写。Vue 是基于 JavaScript 编写的一套渐进式框架。Vue 由数据驱动，当数据改变，界面也随之改变。相对于其他框架，Vue 更加简洁易写，能较快地搭建出原型，进行实验分析。

6.3.2 后端

后端采用 Flask[7]框架编写。Flask 是一个使用 Python 编写的轻量级 Web 应用框架。其 WSGI 工具箱采用 Werkzeug，模板引擎则使用 Jinja2。Flask 使用 BSD 授权。Flask 也被称为 “microframework”，因为它使用简单的核心，用 extension 增加其他功能。Flask 没有默认使用的数据库、窗体验证工具。

6.3.3 使用说明

Demo 系统如图3所示, 系统的网址为 http://spam.zengyutao.me:5000。由于系统部署在实验室服务器中，所以系统暂时只能在计算所网数实验室内网内访问。同时因为深度学习模型占显存比较大，暂时是关闭的[8]整个系统由三部分组成。

第一部分 Demo 系统介绍区域。该系统的名字叫 “WHAT ARE WE CALLING”，同时这也是我们小组的名字。

第二部分用户输入区域，主要元素为输入文本框及按钮。用户可在该区域输入要识别的文本，并按回车键或者点击文本框后的按钮，就能将信息发送给后端处理。

第三部分结果展示区域。该区域主要由两部分组成，分别是用户输入展示和文本识别展示。用户输入展示部分显示用户即时的输入，即用户所要识别的文本。文本识别展示部分显示每一个模型对用户输入所判定的结果，该结果用表格来表示。其中，主要包含三个元素，分别是模型的名字，判定为垃圾短信的概率，判定为非垃圾短信的概率。

[6] https://vuejs.org
[7] http://flask.pocoo.org
[8] 如果需要检查可以联系陶舒畅同学把它打开

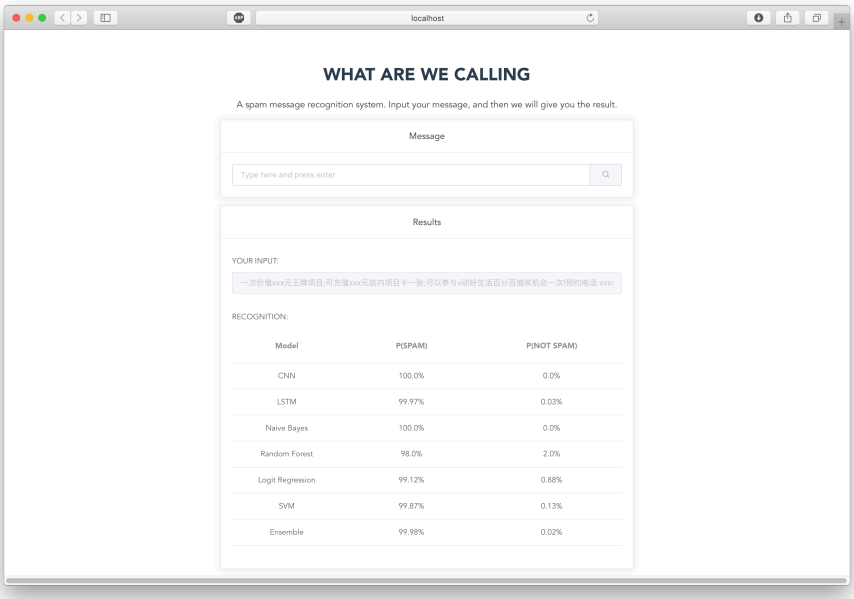

图 3: Demo 系统界面

7 总结及讨论

7.1 项目总结

近年来，垃圾短信不断泛滥，给人们生活带来了很大困扰，高效的识别垃圾短信已经成为日常生活的重要技术，垃圾短信的识别可以对短信文本分类来实现。我们对比了文本分类问题和图像分类以及推荐系统的异同点，并根据文本分类的特点选取了 6 个模型进行实现。

垃圾短信识别是一个经典的文本分类任务，传统的机器学习方法在分类器方面和特征提取方面在该任务上已经有了较好的效果，深度学习方法应用于该任务，更取得了不小进展。

本项目首先进行了文本的预处理使用正则化，分词，词向量化等方式，将原始语料处理成便于应用机器学习算法的相对规则化的数据。然后提取了 TF-IDF 特征，哈希特征用于经典的机器学习算法。

本项目共实现了六个模型，其中包括经典的机器学习方法逻辑回归、朴素贝叶斯、支持向量机、随机森林，两个深度学习算法卷积神经网络和循环神经网络，并将模型进行集成，对比了各自的优缺点，并使用所给数据集进行了实验以及参数的调整。在单模型中，CNN 取得了最好的效果，这证明了深度学习方法在垃圾短信分类任务的强大能力，而集成模型的效果超过

了所有的单模型。同时我们实现了 Demo，部署在了计算所网数实验室服务器上。

7.2 模型改进

- 实验中，对分类错误样本的分析后发现，存在较多分词错误导致的错误分类，在未来可以通过整理垃圾短信词表，或本地训练分词模型等方式来改进分词的精确度。
- 由于很多垃圾短信用词和行文并不规范，可能存在难以避免的分词错误。为尽量减轻这一影响，对于卷积神经网络，可以引入字符级别的 embedding 表示。
- 由于设备性能的限制，随机森林中树的数量只设置了 100 棵，TF-IDF 特征只选取 3000 维。如果在计算资源充足的条件下，增大森林规模和特征多样性，可能会给模型带来一定的提升。
- 对于部分传统模型，如逻辑斯蒂回归，朴素贝叶斯等，可以通过卡方检验、互信息等方式过滤掉无用的词项特征，提高准确度和效率，对于其他传统模型而言，这种方式也可以提高它们的计算效率。
- 虽然在该数据集上，大多数模型都达到了近乎完美的分类效果，不需要引入更多的关于主题方面的特征，但在其他数据集上的情形未知，可以通过 LDA 等主题模型来学习文本的主题分布，作为附加特征辅助分类。

参考文献

[1] 王斌 and 潘文锋, "基于内容的垃圾邮件过滤技术综述," 中文信息学报, vol. 19, no. 5, pp. 3–12, 2005.

[2] J. L. Herlocker, J. A. Konstan, A. Borchers, and J. Riedl, "An algorithmic framework for performing collaborative filtering," in *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*. ACM, 1999, pp. 230–237.

[3] X. He, L. Liao, H. Zhang, L. Nie, X. Hu, and T.-S. Chua, "Neural collaborative filtering," in *Proceedings of the 26th International Conference on World Wide Web*. International World Wide Web Conferences Steering Committee, 2017, pp. 173–182.

[4] O. Barkan and N. Koenigstein, "Item2vec: neural item embedding for collaborative filtering," in *Machine Learning for Signal Processing (MLSP), 2016 IEEE 26th International Workshop on*. IEEE, 2016, pp. 1–6.

[5] M. Sahami, S. Dumais, D. Heckerman, and E. Horvitz, "A bayesian approach to filtering junk e-mail," in *Learning for Text Categorization: Papers from the 1998 workshop*, vol. 62. Madison, Wisconsin, 1998, pp. 98–105.

[6] S. Tong and D. Koller, "Support vector machine active learning with applications to text classification," *Journal of machine learning research*, vol. 2, no. Nov, pp. 45–66, 2001.

[7] L. Breiman, "Random forests," *Machine learning*, vol. 45, no. 1, pp. 5–32, 2001.

[8] Y. Kim, "Convolutional neural networks for sentence classification," *arXiv preprint arXiv:1408.5882*, 2014.

[9] M. Sundermeyer, R. Schlüter, and H. Ney, "Lstm neural networks for language modeling," in *Thirteenth annual conference of the international speech communication association*, 2012.

[10] 赖文辉，乔宇鹏, "基于词向量和卷积神经网络的垃圾短信识别方法," 计算机应用, pp. 0–0, 2018.

[11] E. W.A, Awad.S.M, "Machine learning methods for spam e-mail classification," *International Journal of Computer Science and Information Technology*, vol. 3, no. 1, pp. 173–184, 2011.

[12] A. K. Sharma and S. Sahni, "A comparative study of classification algorithms for spam email data analysis," *International Journal on Computer Science and Engineering*, vol. 3, no. 5, pp. 1890–1895, 2011.

[13] B. U. Gaikwad and P. P. Halkarnikar, "Spam e-mail detection by random forests algorithm," *International Journal of Advanced Computer Engineering and Communication Technology*, vol. 4, no. 2, pp. 2278–5140, 2013.

[14] X. Carreras and L. Marquez, "Boosting trees for anti-spam email filtering," *arXiv preprint cs/0109015*, 2001.

[15] K. Mathew and B. Issac, "Intelligent spam classification for mobile text message," in *Computer Science and Network Technology (ICCSNT), 2011 International Conference on*, vol. 1. IEEE, 2011, pp. 101–105.

[16] 陶峰, 汤鲲, 程光, "基于改进 tfidf 算法的邮件分类技术," 计算机技术与发展, vol. 28, no. 8, pp. 27–31, 2018.

[17] 李猛 and 刘元宁, "一种基于信息增益的新垃圾邮件特征选择算法," 吉林大学学报: 理学版, vol. 55, no. 2, pp. 379–382, 2017.

[18] 王禾清, "改进的互信息特征选择方法在垃圾邮件检测中的应用," 电脑知识与技术, no. 5X, pp. 163–166, 2017.

[19] 赵晓丹, 徐燕, "垃圾邮件分类技术对比研究," 信息网络安全, no. 2, pp. 0–0, 2014.

[20] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Advances in neural information processing systems*, 2013, pp. 3111–3119.

[21] M. E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer, "Deep contextualized word representations," *arXiv preprint arXiv:1802.05365*, 2018.

[22] X. Zhang, J. Zhao, and Y. LeCun, "Character-level convolutional networks for text classification," in *Advances in neural information processing systems*, 2015, pp. 649–657.

[23] 周钦强, 孙炳达, and 王义, "文本自动分类系统文本预处理方法的研究," Ph.D. dissertation, 2005.

[24] Y. Goldberg and O. Levy, "word2vec explained: deriving mikolov et al.'s negative-sampling word-embedding method," *arXiv preprint arXiv:1402.3722*, 2014.

[25] B. Leo, J. H. Friedman, R. A. Olshen, and C. J. Stone, "Classification and regression trees," *Wadsworth International Group*, 1984.

[26] L. R. Iverson, A. M. Prasad, S. N. Matthews, and M. Peters, "Estimating potential habitat for 134 eastern us tree species under six climate scenarios," *Forest Ecology and Management*, vol. 254, no. 3, pp. 390–406, 2008.

[27] P. Langley, W. Iba, K. Thompson *et al.*, "An analysis of bayesian classifiers," in *Aaai*, vol. 90, 1992, pp. 223–228.

[28] 张铭锋, 李云春, and 李巍, "垃圾邮件过滤的贝叶斯方法综述," Ph.D. dissertation, 2005.

[29] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg *et al.*, "Scikit-learn: Machine learning in python," *Journal of machine learning research*, vol. 12, no. Oct, pp. 2825–2830, 2011.

[30] Y. Zhang and B. Wallace, "A sensitivity analysis of (and practitioners' guide to) convolutional neural networks for sentence classification," *arXiv preprint arXiv:1510.03820*, 2015.

[31] Y.-L. Boureau, J. Ponce, and Y. LeCun, "A theoretical analysis of feature pooling in visual recognition," in *Proceedings of the 27th international conference on machine learning (ICML-10)*, 2010, pp. 111–118.

[32] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[33] G. E. Hinton, N. Srivastava, A. Krizhevsky, I. Sutskever, and R. R. Salakhutdinov, "Improving neural networks by preventing co-adaptation of feature detectors," *arXiv preprint arXiv:1207.0580*, 2012.