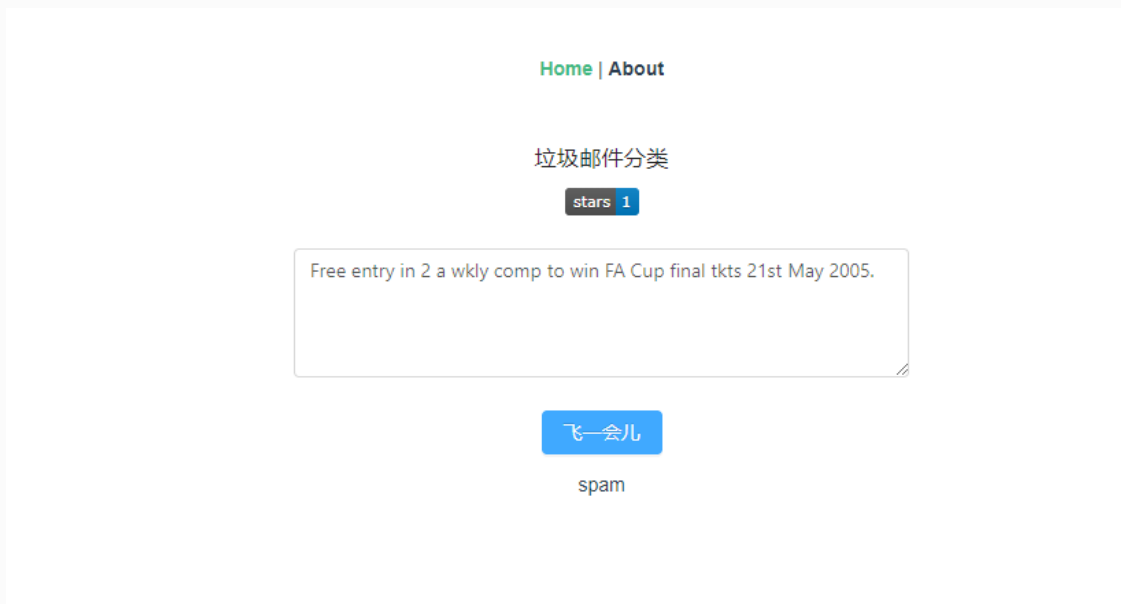


H1 mini-sms-classify

基于支持向量机的垃圾邮件分类，使用SVM+flask+vue



数据集 SMS Spam Collection Data Set 来源于 [UCI](#)。样例被分为非垃圾邮件（86.6%）和垃圾邮件（13.4%），数据格式如下：

```
1 ham Go until jurong point, crazy.. Available only in bugis
n great world la e buffet...
2 ham Ok lar... Joking wif u oni...
3 spam Free entry in 2 a wkly comp to win FA Cup final
tkts 21st May 2005.
4 ham U dun say so early hor... U c already then say...
```

H2 代码结构

```
1 - client: 前端实现
2 - server: 后端实现
3     - models: svm model
4         - sms_classify.py: 垃圾邮件分类实现
5         - SMSSpamCollection: 数据集
6     - app.py 系统实现
7 - svm.py: 支持向量机算法实现
8 - svm_test.py: 算法test
```

H2 性能评估

综合比较了垃圾邮件分类任务在支持向量机、朴素贝叶斯、最近邻、决策树算法下的性能，评估指标包括accuracy、precision、recall、f1-score等。

从accuracy来看，支持向量机的accuracy为98%，是所有测试算法中最高的，可以看出垃圾邮件分类任务适合使用支持向量机来做。

各算法表现具体如下表：

- 支持向量机：

1		precision	recall	f1-score	support
2					
3	0	0.98	1.00	0.99	482
4	1	1.00	0.86	0.92	76
5					
6	accuracy			0.98	558
7	macro avg	0.99	0.93	0.96	558
8	weighted avg	0.98	0.98	0.98	558

支持向量机的accuracy有 98.029%。

- 贝叶斯算法：

1		precision	recall	f1-score	support
2					
3	0	0.94	1.00	0.97	482
4	1	1.00	0.62	0.76	76
5					
6	accuracy			0.95	558
7	macro avg	0.97	0.81	0.87	558
8	weighted avg	0.95	0.95	0.94	558
9					

贝叶斯算法的accuracy只有 94.803%。

- 最近邻算法：

1		precision	recall	f1-score	support
2					
3	0	0.97	0.99	0.98	482
4	1	0.93	0.83	0.88	76
5					
6	accuracy			0.97	558
7	macro avg	0.95	0.91	0.93	558
8	weighted avg	0.97	0.97	0.97	558
9					

最近邻算法的accuracy为 96.774%。

- 决策树算法：

1	precision	recall	f1-score	support	
2					
3	0	0.97	0.98	0.98	482
4	1	0.88	0.79	0.83	76
5					
6	accuracy			0.96	558
7	macro avg	0.92	0.89	0.90	558
8	weighted avg	0.96	0.96	0.96	558

决策树算法的accuracy为 95.699%。

H2 如何运行

首先安装必要的包

```
1 # 创建虚拟环境
2 python -m venv env
3 # 激活虚拟环境
4 source env/bin/activate
5 # 安装依赖包
6 pip install -r requirements.txt
```

H3 运行SVM算法实现

```
1 # 确保安装 matplotlib 和 numpy
2 python3 svm_test.py
```

H3 运行垃圾邮件分类

```
1 ~ cd server/models/
2 ~ python3 sms_classify.py
```

H3 运行垃圾邮件分类系统

H4 server端

```
1 # 确保安装必要的包
2 # 启动flask
3 python app.py
4
```

H4 client端

```
1 # 确保安装node & npm
2 npm install
3 npm run server
```