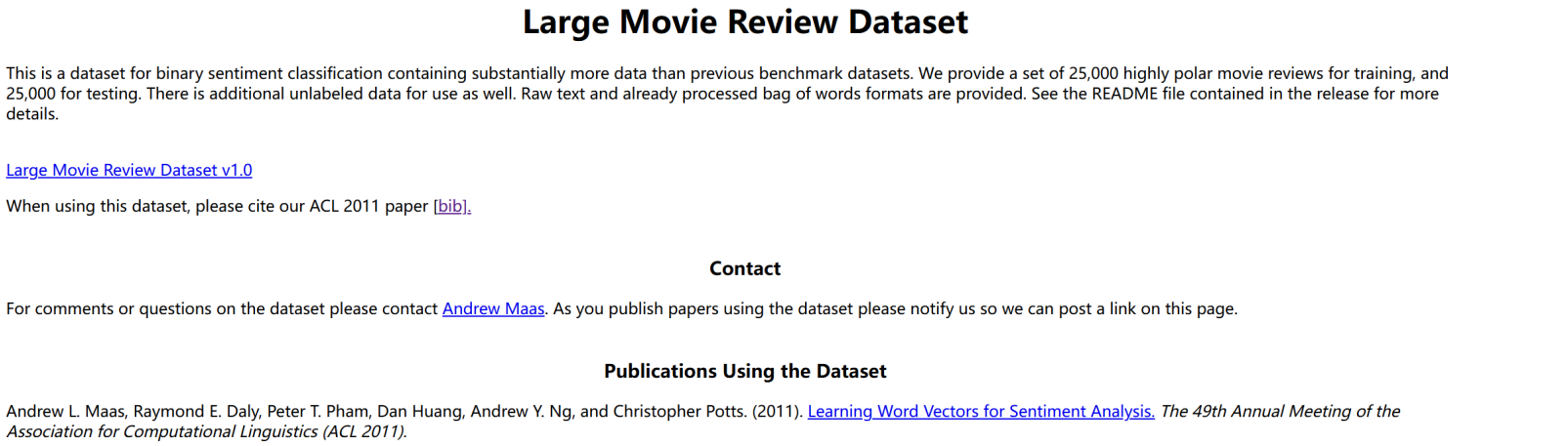
人工智能大作业演示报告

杨熙承 22030531

常州工学院 计信院

2024 年 12 月 16 日

数据集介绍



Large Movie Review Dataset

This is a dataset for binary sentiment classification containing substantially more data than previous benchmark datasets. We provide a set of 25,000 highly polar movie reviews for training, and 25,000 for testing. There is additional unlabeled data for use as well. Raw text and already processed bag of words formats are provided. See the README file contained in the release for more details.

Large Movie Review Dataset v1.0

When using this dataset, please cite our ACL 2011 paper [bib].

Contact

For comments or questions on the dataset please contact Andrew Maas. As you publish papers using the dataset please notify us so we can post a link on this page.

Publications Using the Dataset

Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. (2011). Learning Word Vectors for Sentiment Analysis. *The 49th Annual Meeting of the Association for Computational Linguistics (ACL 2011).*

1. Large Movie Review Dataset (IMDB)
2. 斯坦福官方数据，训练集和测试集各25,000条电影评论
3. data/aclImdb
   - `train/`: 训练集 (pos/neg)
   - `test/`: 测试集 (pos/neg)

特征工程——词频矩阵

原始文本

1. "Dr.Liu always smlies."
2. "Dr.Liu encourages us."
3. "Dr.Liu study harduous."

词汇表

单词	索引	单词	索引
dr	1	smlies	5
liu	4	encourages	2
always	0	us	7
study	6	harduous	3

词频矩阵（Bag of Words）

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & \cdots \\ 0 & 1 & 0 & 0 & 1 & 1 & 0 & 1 & 1 & \cdots \\ 0 & 1 & 1 & 1 & 0 & 0 & 1 & 0 & 0 & \cdots \end{bmatrix}$$

特征工程——Scapy、停用词、N-gram

词形还原 (Spacy)

1. running → run
2. better → good
3. studies → study

停用词 & 最小文档频率

1. 移除常见词：a, an, the...
2. min_df = 6：至少出现6次

N-gram特征

1. 单词组合：1-2个词
2. "very good", "not bad"

核心代码

```
// core code
tfidf_spacy = TfidfVectorizer(
    tokenizer=tokenizer_spacy,
    min_df=6,
    stop_words="english",
    ngram_range=(1, 2))
```

特征工程——TF-IDF

$$TF(t,d) = \frac{\text{词项}t\text{在文档中的次数}}{\text{文档中的总词数}}$$

$$IDF(t,D) = \log_e \frac{\text{文档总数}}{\text{包含词项}t\text{的文档数}}$$

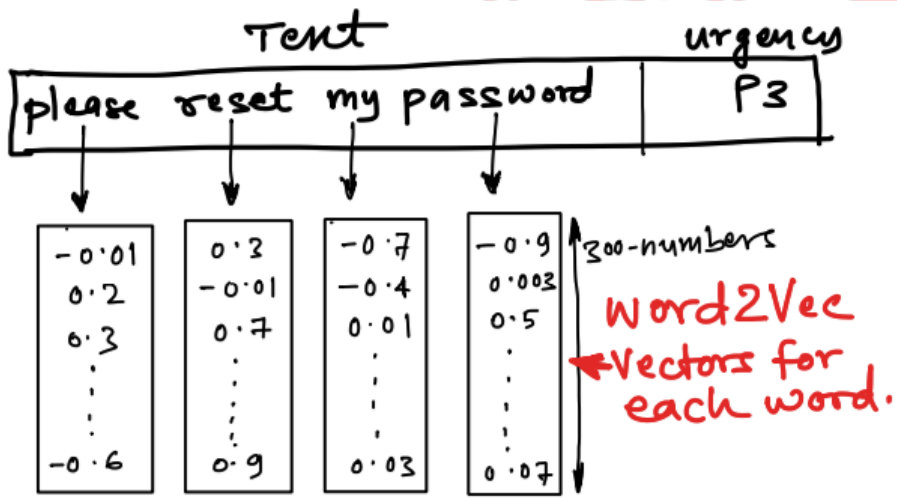
$$TF\text{-}IDF(t,d,D) = TF(t,d) \times IDF(t,D)$$

- 词频高 + 低文档频率 = 高重要性
- 可过滤常见词（the, is, at等）

特征工程——词向量

Word2Vec模型

1. 将词映射到高维向量空间
2. 相似词在空间中距离接近
3. 保持词之间的语义关系



词向量示例——文本

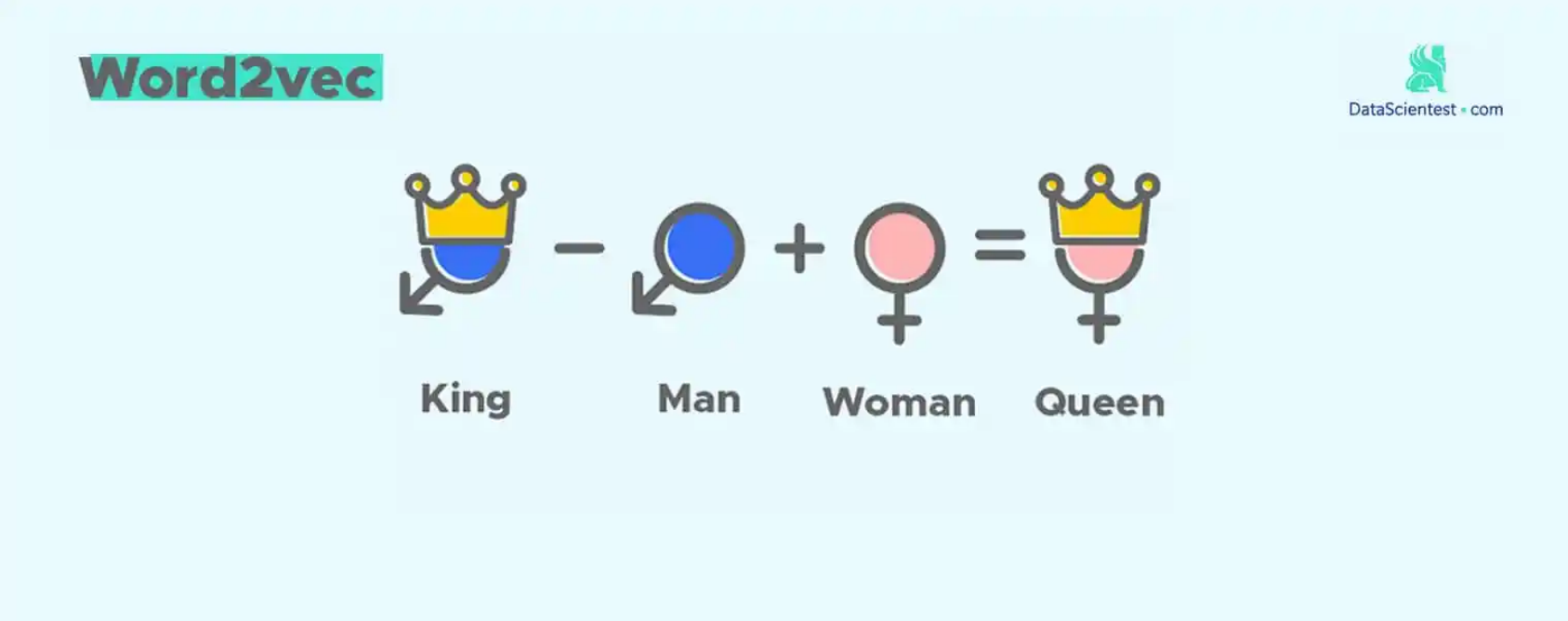
原始文本:

1. "Dr.Liu always smiles, like a Angel."
2. "Dr.Liu always encourages us."
3. "Dr.Liu study harduous."

词序列表示:

$$\text{Text 1:}[1,2,3,4,5,6,7] \rightarrow [0,0,0,1,2,3,4,5,6,7]$$
$$\text{Text 2:}[1,2,3,8,9] \rightarrow [0,0,0,0,0,1,2,3,8,9]$$
$$\text{Text 3:}[1,2,10,11] \rightarrow [0,0,0,0,0,0,1,2,10,11]$$

词向量示例——向量



$$\vec{v_1} = [\,[0.013, -0.011, 0.039, -0.004..... - 0.047, 0.028, 0.025, 0.025]\,]$$

$$\vec{v_2} = [\,[-0.001, 0.014, 0.037, 0.003.....0.001, 0.030, 0.009, 0.020]\,]$$

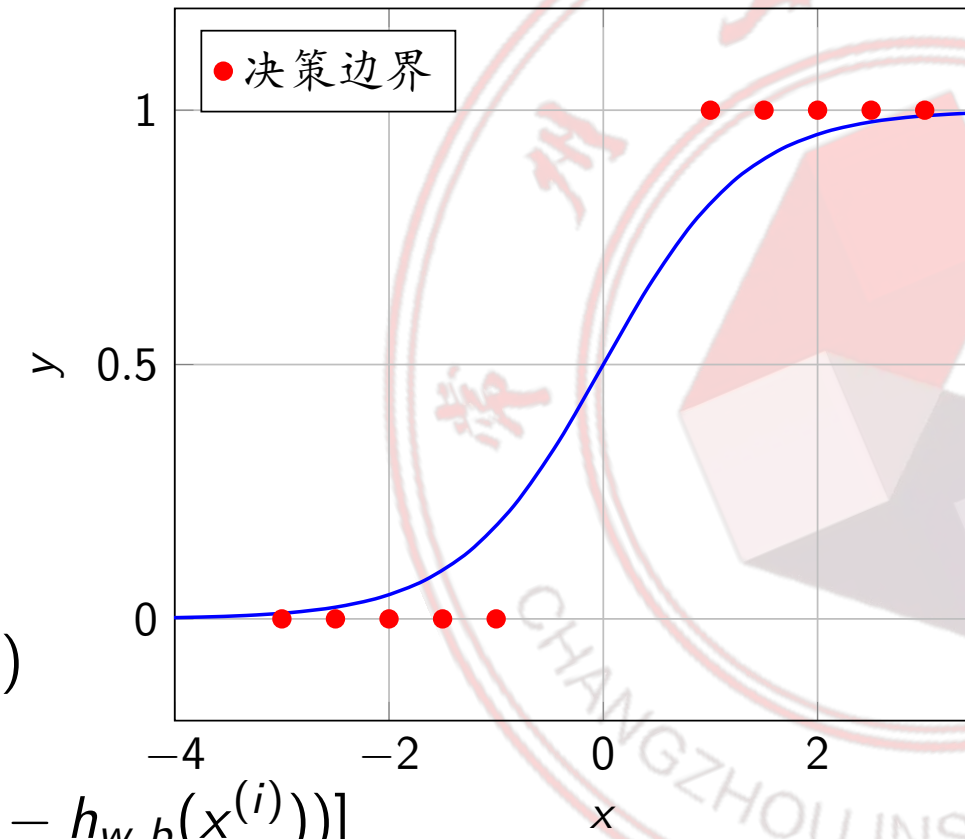
$$\vec{v_3} = [\,[-0.020, -0.008, 0.038, 0.036..... - 0.020, 0.043, -0.012, 0.043]\,]$$

逻辑回归——模型

模型公式

$$P(y=1|x) = \sigma(w^T x + b)$$

$$\sigma(z) = \frac{1}{1+e^{-z}}$$



损失函数

$$J(w,b) = -\frac{1}{m}\sum_{i=1}^{m}[y^{(i)}\log(h_{w,b}(x^{(i)})) + (1-y^{(i)})\log(1-h_{w,b}(x^{(i)}))]$$

逻辑回归——优化

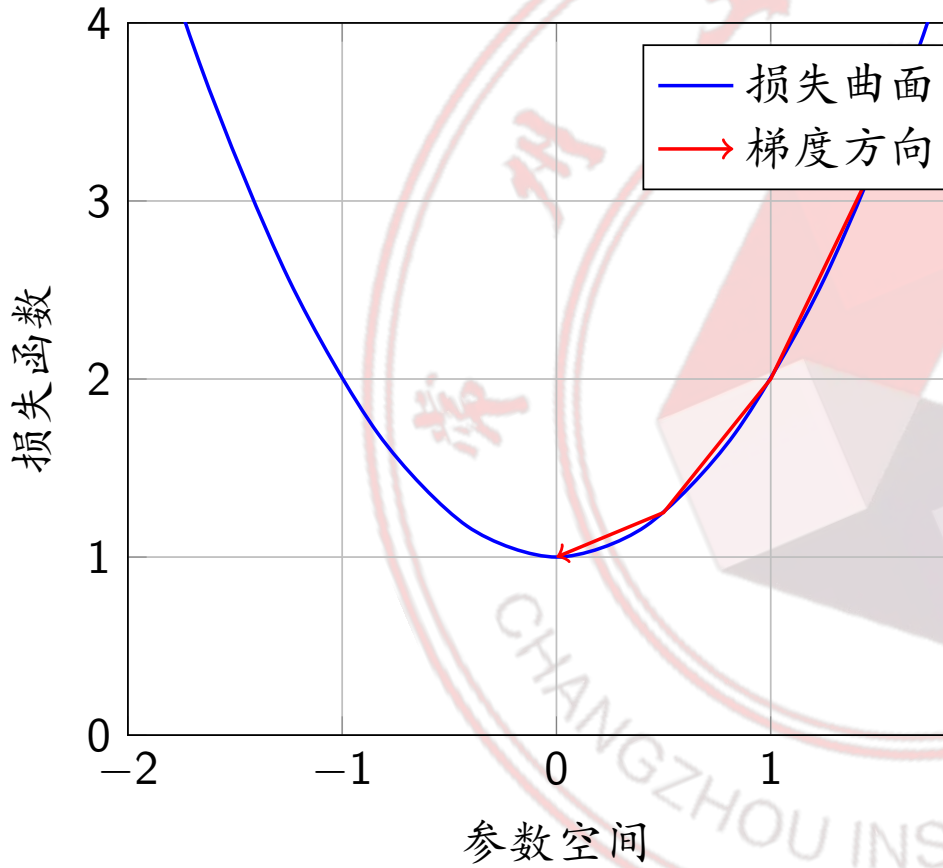
梯度下降

$$w := w - \alpha \frac{\partial}{\partial w} J(w, b)$$

$$b := b - \alpha \frac{\partial}{\partial b} J(w, b)$$

参数说明

- α: 学习率
- $\frac{\partial}{\partial w} J$: 损失对权重的梯度
- $\frac{\partial}{\partial b} J$: 损失对偏置的梯度



逻辑回归——模型训练

模型配置

```
model = LogisticRegression(
    max_iter=1000,
    solver='liblinear'
)
```

1. C: 正则化强度的倒数
2. cv=5: 5折交叉验证
3. liblinear: 大规模文本分类

模型训练

```
model.compile(
    loss='binary_crossentropy',
    optimizer='adam',
    metrics=['accuracy']
)

history = model.fit(
    X_train_padded, y_train,
    epochs=10, batch_size=32,
    validation_split=0.2
)
```

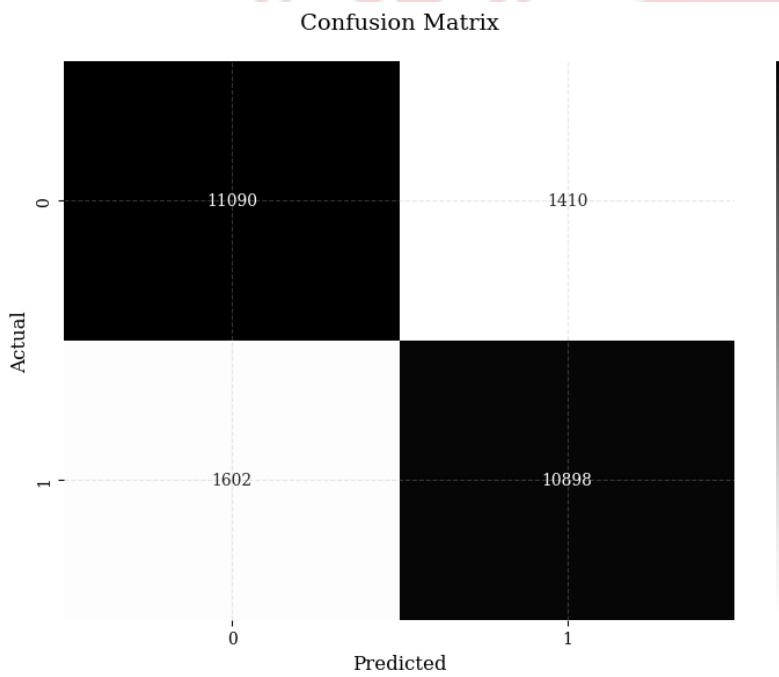
1. 交叉熵损失函数
2. Adam优化器自适应学习
3. 20%验证集划分

逻辑回归——模型评估

分类报告

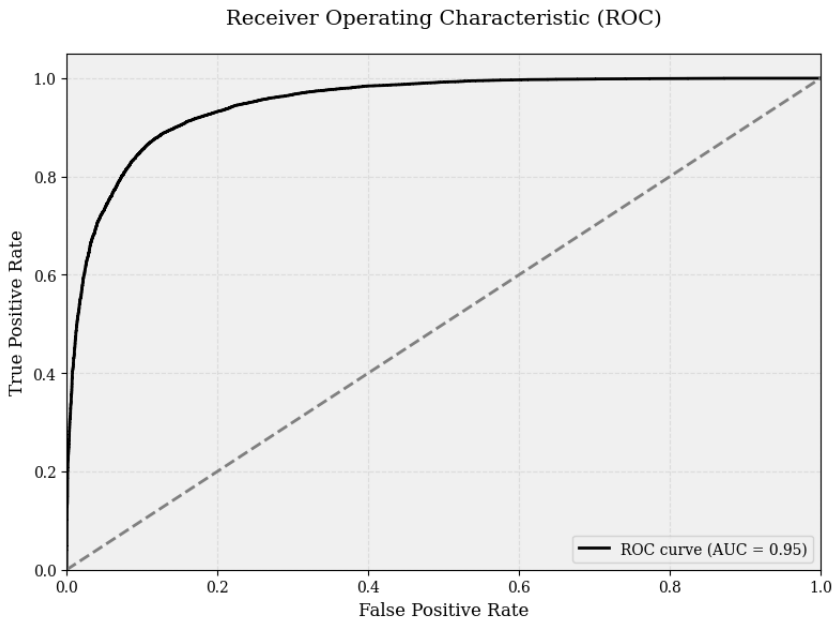
类别	Precision	Recall	F1
0	0.87	0.89	0.88
1	0.89	0.87	0.88
Avg	0.88	0.88	0.88

- 准确率: 88%
- 样本数: 25,000
- 正负样本均衡

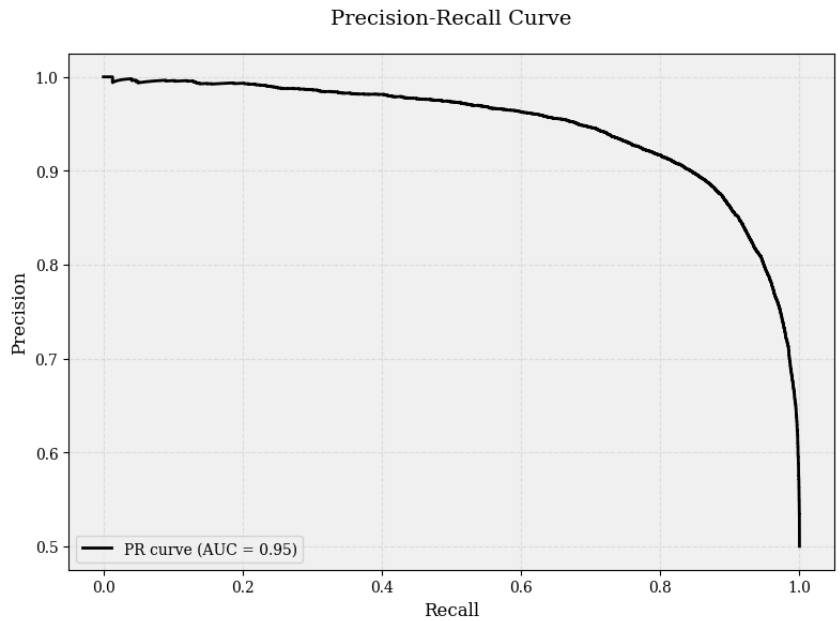


逻辑回归——性能曲线

ROC曲线



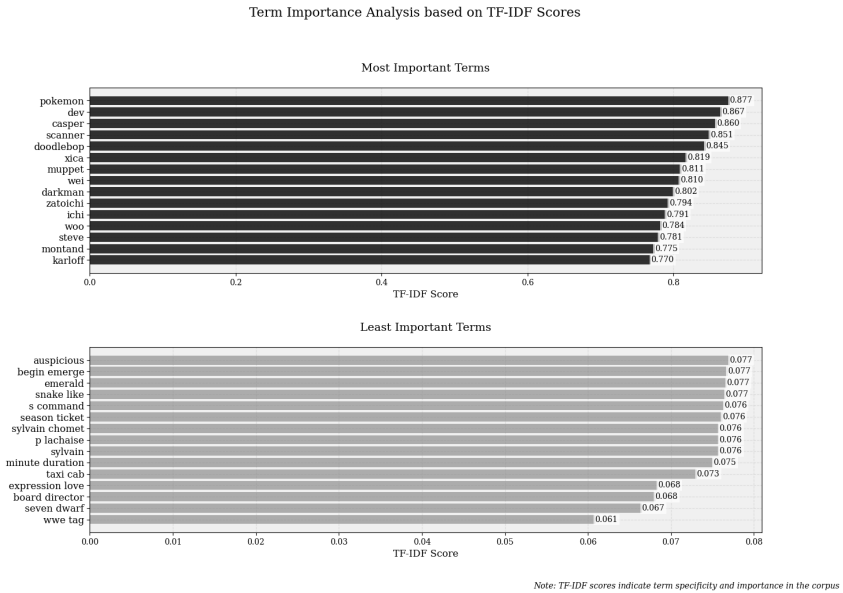
Precision-Recall曲线



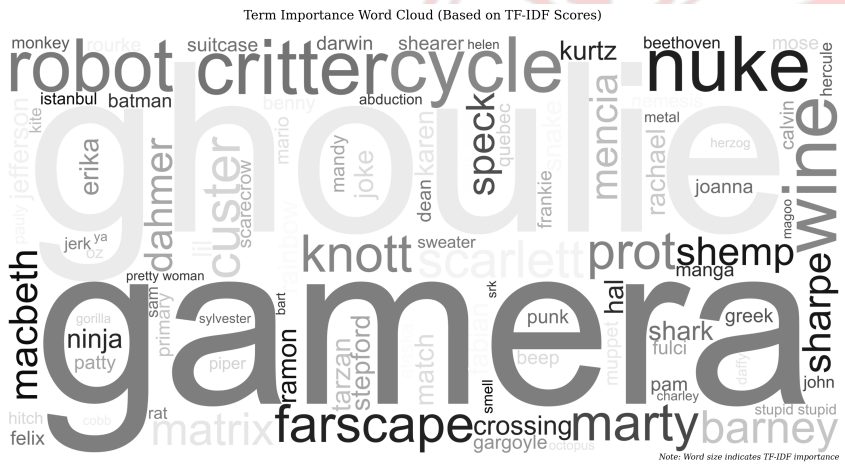
- ROC-AUC: 衡量模型区分正负样本的能力
- PR曲线: 在类别不平衡时更有参考价值

特征分析——TF-IDF得分

特征重要性排序



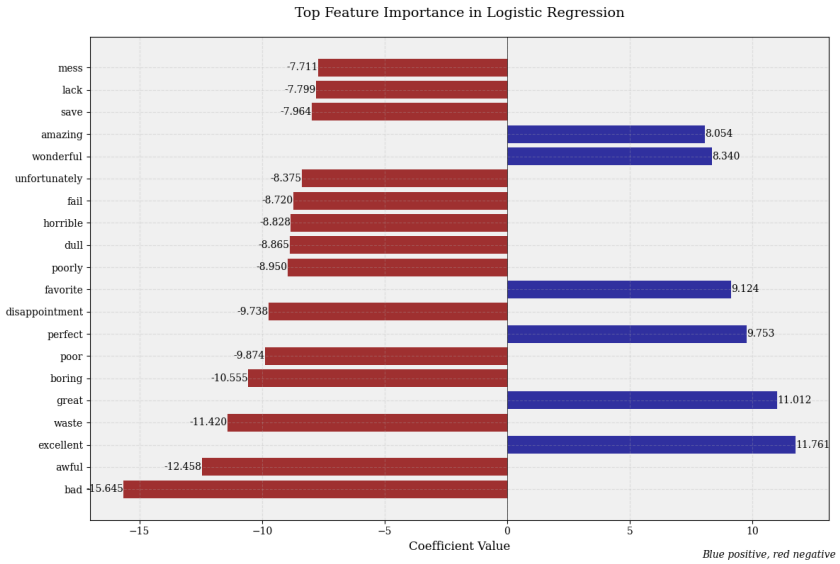
TF-IDF词云图



1. 横向条形图显示各词的TF-IDF得分
2. 词云直观展示高频重要词汇

特征分析——模型系数

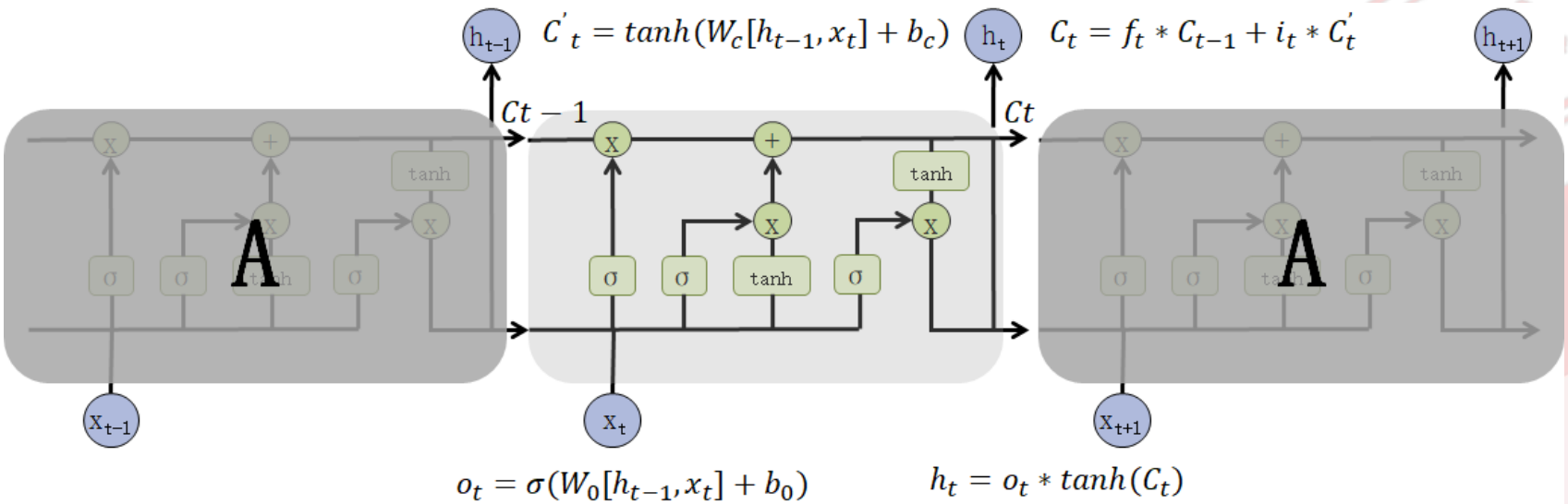
系数条形图



评论词云图

1. 横向条形图显示各词的系数
2. 词云直观展示高频重要词汇

LSTM



门控机制

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f)$$
$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i)$$
$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o)$$

状态更新

$$\tilde{C}_t = \tanh(W_c \cdot [h_{t-1}, x_t] + b_c)$$
$$C_t = f_t \odot C_{t-1} + i_t \odot \tilde{C}_t$$
$$h_t = o_t \odot \tanh(C_t)$$

LSTM——模型训练

模型构建

```
model = Sequential()
model.add(Embedding(
    input_dim=vocab_size,
    output_dim=128,
    weights=[embedding_matrix],
    trainable=False
))
model.add(Bidirectional(LSTM(64)))
model.add(Dropout(0.5))
model.add(Dense(1, activation='sigmoid'))
```

1. 词嵌入层固定权重
2. 双向LSTM捕捉上下文
3. Dropout防止过拟合

模型训练

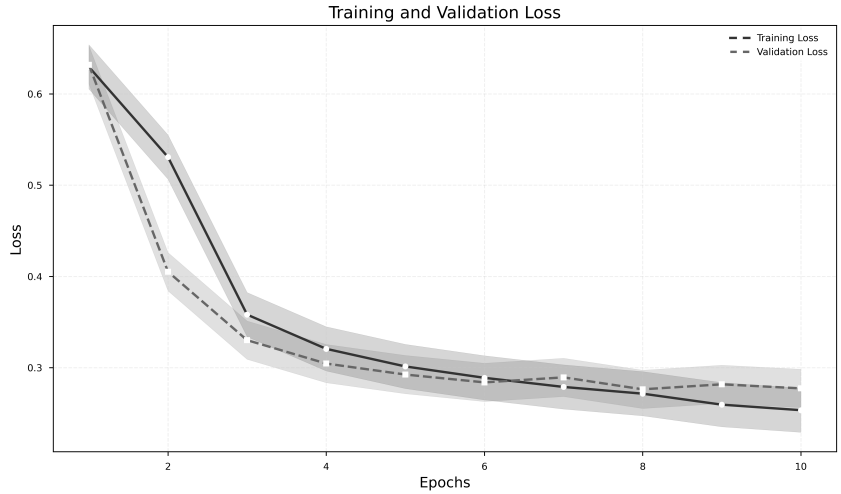
```
model.compile(
    loss='binary_crossentropy',
    optimizer='adam',
    metrics=['accuracy']
)

history = model.fit(
    X_train_padded, y_train,
    epochs=10, batch_size=32,
    validation_split=0.2
)
```

1. 交叉熵损失函数
2. Adam优化器自适应学习
3. 20%验证集划分

LSTM——训练结果

Epoch	Loss	Accuracy	Val Acc
1	0.631	0.651	0.688
2	0.484	0.782	0.822
3	0.352	0.853	0.863
...	...	...	...
8	0.269	0.890	0.884
9	0.264	0.892	0.862
10	0.254	0.897	0.882



1. 最终测试准确率: 88.24%
2. 训练时间: 约35s/epoch
3. 最终准确率基于测试集

LR-LSTM模型融合——权重学习

Algorithm 1 Learn Ensemble Weights

Input: validation dataset D_{val}
Output: optimal weights w_{lr}, w_{lstm}
Load pre-trained models M_{lr}, M_{lstm}
$best\_acc \leftarrow 0$
for $w \leftarrow 0$ to 1.0 step 0.05 **do**
    $acc \leftarrow accuracy(Y_{true}, (w \cdot M_{lr} + (1 - w) \cdot M_{lstm}).predict\_proba(D_{val}) \geq 0.5)$
    if $acc > best\_acc$ **then**
        $best\_acc \leftarrow acc$
        $w_{lr} \leftarrow w$
    end if
end for
$w_{lstm} \leftarrow 1 - w_{lr}$

return w_{lr}, w_{lstm}

权重学习策略:

- 使用验证集(review_polarity)评估不同权重组合
- 网格搜索最优权重配比
- 优化目标：最大化准确率
- 权重和为1，确保概率有效

LR-LSTM模型融合——预测

Algorithm 2 Ensemble Prediction

Input: text T, weights w_{lr}, w_{lstm}
Output: predicted sentiment and topic
$p_{lr} \leftarrow$ LR.predict(T)
$p_{lstm} \leftarrow$ LSTM.predict(T)
$sentiment \leftarrow w_{lr} \cdot p_{lr} + w_{lstm} \cdot p_{lstm}$
if $sentiment > 0.5$ **then**
    return positive
else
    return negative
end if

模型融合优势:

- 结合两个模型的优势
- LR模型：特征解释性强
- LSTM模型：捕捉序列特征
- 加权平均：平滑预测结果
- 准确率提升到90%

LDA——潜在狄利克雷分配

文档生成过程:

1. 按照先验概率$p(d_i)$选择一篇文档d_i
2. 从Dirichlet分布α中取样生成文档d_i的主题分布θ_i
3. 从主题的多项式分布θ_i中取样生成文档d_i第j个词的主题$z_{i,j}$
4. 从Dirichlet分布β中取样生成主题$z_{i,j}$对应的词语分布$\phi_{z_{i,j}}$
5. 从词语的多项式分布$\phi_{z_{i,j}}$中采样最终生成词语$\omega_{i,j}$

模型实现:

```
lda = LatentDirichletAllocation(
    n_components=15,
    max_iter=5,
    learning_method='online',
    learning_offset=50.,
    random_state=0
)
lda.fit(X)
```

参数说明:

- n_components: 主题数量
- max_iter: 最大迭代次数
- learning_method: 在线学习
- learning_offset: 学习率参数

LDA——主题分布

15个主题类别:

观后感 thought, thing, lot...

音乐舞蹈 music, musical, songs...

喜剧 comedy, funny, fun...

表演 role, performance, cast...

社会议题 sex, women, american...

恐怖 horror, effects, gore...

【观后感】	【音乐舞蹈】	【喜剧】	【表演】	【社会议题】
topic 0	topic 1	topic 2	topic 3	topic 4
didn	music	comedy	role	sex
thought	musical	funny	john	women
thing	songs	fun	performance	tom
lot	song	zombie	cast	american
though	dance	horror	play	black
doesn	dancing	house	actor	woman
want	singing	dr	played	around
going	kelly	comedies	mr	another
10	number	humor	james	christmas
things	numbers	doctor	british	white
re	allen	afraid	plays	house
real	band	hilarious	new	our
actually	stage	laughs	young	doesn
few	voice	grant	jack	three
every	tarzan	werewolf	ben	own

【恐怖惊悚】	【负面评价】	【经典电影】	【政治历史】	【动作警匪】
topic 5	topic 6	topic 7	topic 8	topic 9
horror	worst	cartoon	war	action
effects	awful	de	political	police
gore	script	sucks	us	fight

LDA——文档-主题矩阵

矩阵结构:

- 行: 每一行代表一条文本
- 列: 每一列代表一个主题
- 值: 文本属于该主题的概率

矩阵特点:

- 概率和为1
- 反映文本的主题分布
- 可用于文本分类和聚类

	观后感	音乐舞蹈	...
T1	p11	p12	...
T2	p21	p22	...
T3	p31	p32	...
...	...	...	...

后端接口一：情感分析

接口说明:

- 路由: /api/sentiment
- 方法: POST
- 功能: 融合LR和LSTM的情感分析

特点:

- 模型融合: LR权重0.6, LSTM权重0.4
- 文本预处理: 清洗、分词、序列化
- 返回详细概率分布

返回示例:

```
{
    "sentiment": "positive",
    "probability": 0.85,
    "model_details": {
        "logistic_regression_prob": 0.82,
        "lstm_prob": 0.89
    }
}
```

后端接口二：主题分类

接口说明:

- 路由: /api/topic
- 方法: POST
- 功能: LDA主题分布分析

特点:

- 15个预定义主题
- 返回完整主题分布
- 主题映射转换

返回示例:

```
{
    "topic": "music and dance",
    "topic_id": 1,
    "confidence": 0.75,
    "topic_distribution": {
        "0": 0.1,
        "1": 0.75,
        "2": 0.15
    }
}
```

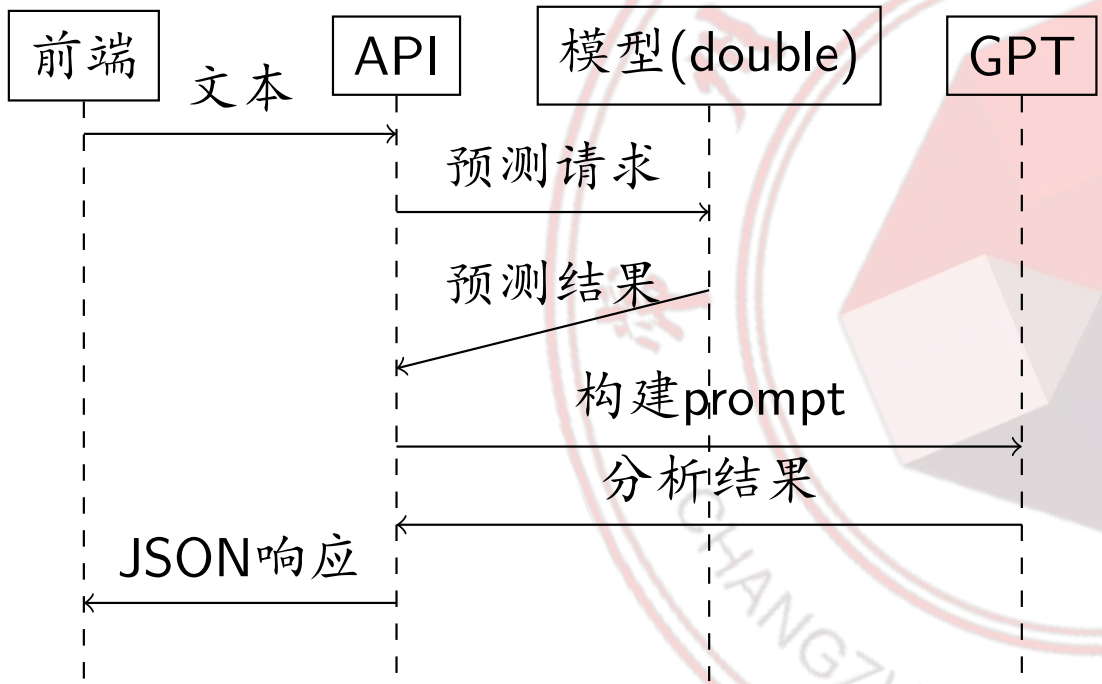
后端接口三：智能分析集成

接口说明:

- 路由: /api/analyze
- 方法: POST
- 功能: 模型预测 + GPT分析

创新设计:

1. 模型预测结果(情感/主题)作为GPT输入
2. 定制化prompt引导分析
3. 结构化输出集成



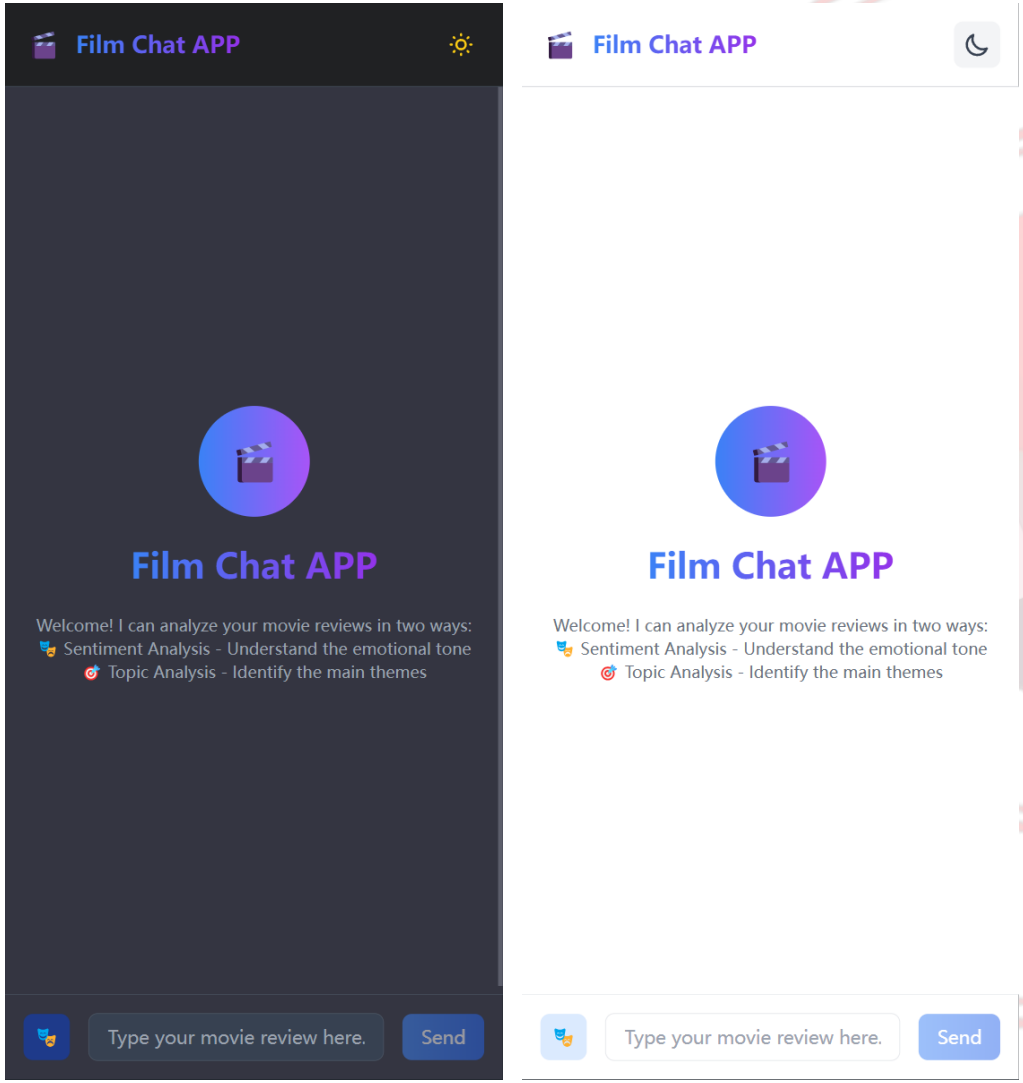
前端应用

技术栈:

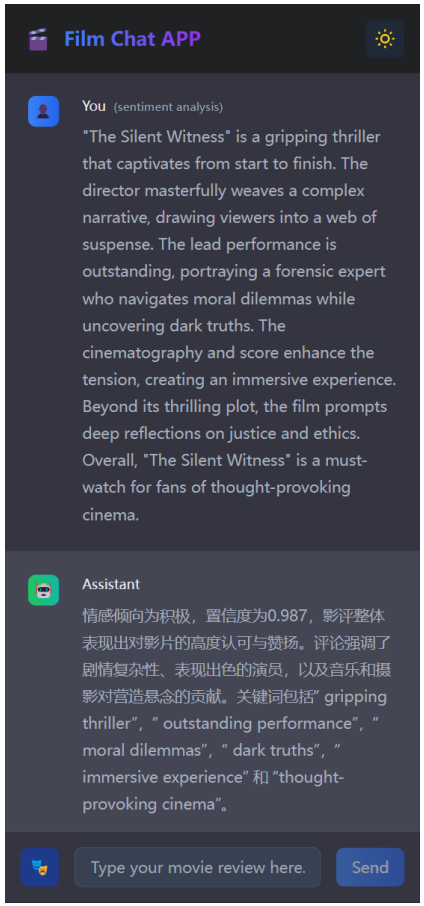
- React + Vite
- TailwindCSS
- Modern ES6+

主要组件:

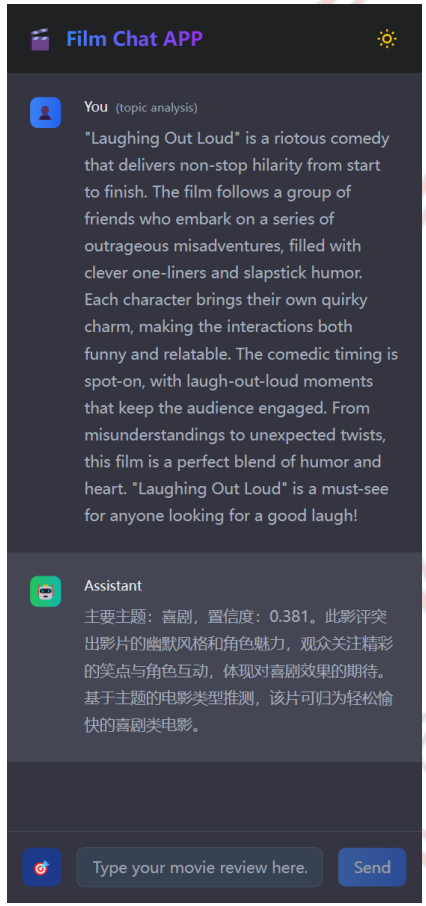
- ChatArea: 对话展示
- InputArea: 用户输入
- ChatMessage: 消息气泡
- Header: 导航与功能



App应用



展示积极/消极概率分布



显示主题分布情况

Thanks