

# 《人工智能基础》第二次作业实验报告<sup>1</sup>

姓名：苏祎成

学号：23307090051

日期：2025年12月16日

---

## 1 摘要

随着社交媒体数据的爆炸式增长，从非结构化文本中精准挖掘情感倾向已成为自然语言处理（NLP）领域的关键任务。本实验旨在探究基于 Transformer 架构的预训练语言模型在细粒度情感分类任务中的应用效能。我们选用 **DistilBERT** 模型作为核心组件，在包含六类情感（Sadness, Joy, Love, Anger, Fear, Surprise）的 Twitter 数据集上，对比分析了“特征提取（Feature Extraction）”与“全参数微调（Fine-tuning）”两种迁移学习范式的性能差异。

实验结果表明，利用 DistilBERT 提取静态 [CLS] 向量并配合逻辑回归（Logistic Regression）的方案实现了 63.40% 的准确率，验证了预训练特征的线性可分性；而经过端到端微调的模型则展现了压倒性优势，准确率跃升至 **93.55%**，显著提升了模型在复杂语义下的判别能力。此外，通过混淆矩阵分析与对抗样本测试（Adversarial Testing），我们发现尽管微调模型在常规样本上表现卓越，但在面对类别不平衡（如 Surprise 类）及反讽、双重否定等高阶语用逻辑时，仍存在鲁棒性边界。本研究不仅验证了微调策略在特定领域任务中的优越性，也揭示了当前模型在缺乏外部常识推理支持下的认知局限。

**关键词：**情感分析，DistilBERT，迁移学习，微调，鲁棒性分析

---

## 2 引言

在当今的数字化时代，Twitter 等社交媒体平台每天产生数以亿计的用户生成内容（UGC）。这些文本数据中蕴含着公众对事件、产品及社会话题的丰富情感态度，具有极高的商业与社会分析价值。然而，社交媒体文本通常具有短小、非正式、包含大量噪声及复杂语用环境（如俚语、反讽）的特点，这对传统的基于规则或统计（如词袋模型）的情感分析方法提出了巨大挑战。近年来，以 BERT 为代表的预训练 Transformer 模型通过大规模无监督学习捕捉了深层的语言句法与语义特征，彻底改变了 NLP 任务的范式。

本实验聚焦于基于 **DistilBERT** 的情感分类任务。作为 BERT 的轻量化蒸馏版本，DistilBERT 在保留了大部分语言理解能力的同时，大幅降低了计算资源需求，使其更贴近实际应用场景。本实验的主要目标包含三个层面：首先，从底层机制出发，通过对字符级分词与子词级（Subword）分词，理解现代 NLP 模型处理文本输入的基本逻辑；其次，通过实证研究，量化对比“冻结参数特征提取”与“全参数微调”两种策略在 Emotion 数据集上的性能差异，探讨深度学习模型在特定下游任务中的适应性；最后，跳出单纯的指标评估，通过错误分析（Error Analysis）与人为构造的对抗样本，深入探究模型在处理长尾类别及复杂逻辑时的决策边界与潜在缺陷。

通过这一系列实验，我们旨在建立一个从数据预处理、模型构建到深度评估的完整 NLP 实验闭环，并对预训练模型的能力边界形成客观、辩证的认知。

---

<sup>1</sup> 源代码中的模型权重 huggingface 地址：<https://huggingface.co/abraxas417/emotion-bert-distilled>；超参数：（Learning Rate=2e-5, Batch Size=64, epoch=2）

### 3 实验方法

#### 3.1 模型架构与训练范式

本实验选用 **DistilBERT** (distilbert-base-uncased) 作为核心预训练模型。作为 BERT 的轻量化变体, DistilBERT 通过知识蒸馏 (Knowledge Distillation) 技术在保留了 BERT 97% 性能的同时, 减少了 40% 的参数量并提升了 60% 的推理速度, 这使其成为在资源受限环境下进行情感分析任务的理想选择。

为了探究不同迁移学习策略的有效性, 本实验设计了两种对比鲜明的训练范式:

1. **特征提取 (Feature Extraction):** 冻结 DistilBERT 的所有编码器参数, 仅将其作为静态的语义特征提取器。我们将输入序列经由模型前向传播后得到的 [CLS] 隐藏状态向量 (维度 \$1 \times 768\$) 作为下游分类器的输入。在此基础上, 我们分别训练了逻辑回归 (Logistic Regression)、支持向量机 (SVM) 和随机森林 (Random Forest) 三个传统机器学习模型, 以此构建实验的性能基准 (Baseline)。
2. **全参数微调 (Fine-tuning):** 解冻整个预训练模型的参数空间, 并在 DistilBERT 的顶层添加一个全连接分类层 (Classification Head)。通过端到端的反向传播算法, 模型不仅学习了特定任务的分类边界, 同时根据情感数据集的语用特征动态调整了底层的语言表示能力。

#### 3.2 评估指标体系

鉴于本实验所用的 Emotion 数据集存在显著的类别不平衡问题 (例如 "Joy" 类样本量远多于 "Surprise" 类), 单纯依赖准确率 (Accuracy) 可能导致对少数类识别能力的误判。因此, 本实验建立了一套多维度的评估指标体系, 重点关注 **F1-Score** 及其聚合形式。

##### 3.2.1 基础指标定义

对于任意情感类别  $c$ , 我们基于混淆矩阵计算其精确率 (Precision) 和召回率 (Recall)。精确率反映了模型预测的可靠性, 而召回率反映了模型的覆盖能力。F1-Score 作为两者的调和平均数, 能够综合衡量模型在单一类别上的稳健性:

$$\begin{aligned} Precision_c &= (TP_c) / (TP_c + FP_c) \\ Recall_c &= (TP_c) / (TP_c + FN_c) \\ F1_c &= 2 \cdot (Precision_c \cdot Recall_c) / (Precision_c + Recall_c) \end{aligned}$$

##### 3.2.2 全局聚合策略

为了将六个情感类别的性能汇总为单一的评判标准, 并有效应对长尾分布挑战, 我们采用以下两种聚合计算方式:

**宏平均 (Macro Average):** 该指标对所有类别赋予同等权重, 计算各类别 F1-Score 的算术平均值。

$$Macro\ F1 = 1/N \sum_{i=1}^N F1_i$$

宏平均不考虑样本数量的差异, 因此对少数类 (Minority Class) 的性能波动极为敏感。在本实验中, Macro F1 是衡量模型是否具备“无偏见”泛化能力的关键指标, 它能有效揭示模型是否仅仅通过过度拟合主导类别 (如 Joy) 来获取高分。

**加权平均 (Weighted Average):** 该指标根据每个类别的样本占比 (Support) 进行加权求和。

$$Weighted F1 = \sum_{i=1}^N w_i \cdot F1_i, \quad w_i = (Support_i) / (\text{Total Samples})$$

加权平均反映了模型在整体数据流中的平均表现，更能代表模型在实际应用场景（真实分布）下的预期效果。

综上所述，本实验将以 **Accuracy** 作为直观参考，以 **Macro F1-Score** 作为核心判据，结合混淆矩阵分析，对不同模型的实验结果进行全方位的对比与论证。

---

## 4 实验结果与分析

### 4.1 探索性数据分析 EDA

为了深入理解数据的底层结构与分布规律，我们在预处理之前首先进行了探索性数据分析 (EDA)。通过将原始的 `Dataset` 对象转换为 `Pandas DataFrame` 格式，我们得以以表格化的形式直观地检视数据样本，每一条样本由原始文本 (Text) 及其对应的情感标签 (Label) 构成，这种清晰的结构为后续的监督学习提供了标准化的输入-输出对。

在对标签分布进行可视化分析后，我们观察到了一个对模型训练至关重要的现象：数据集存在显著的类别不平衡 (Class Imbalance) 问题。从频率统计柱状图中可以清晰地看出，“Joy”（喜悦）和“Sadness”（悲伤）是出现频率最高的两个主导类别，占据了数据集的绝大部分比例；相比之下，“Love”（爱）和“Surprise”（惊讶）的样本数量则相对稀缺，尤其是“Surprise”类，其样本量远低于主导类别。这种长尾分布特征意味着，如果模型仅仅采取“猜测众数”的策略（即盲目预测出现频率最高的类别），也能获得看似不错的准确率 (Accuracy)，但这显然无法反映模型对少数类别的真实识别能力。

基于上述 EDA 的分析结果，我们确立了后续实验的评估基准与策略。鉴于类别分布的不均匀性，单纯依赖准确率指标可能会产生误导，掩盖模型在少数类（如 Surprise）上表现不佳的事实。因此，在后续的模型评估环节中，我们将重点关注 F1-Score（尤其是 Weighted Avg 和 Macro Avg）以及混淆矩阵，以此来更全面、公正地衡量模型在各个情感类别上的泛化性能，确保模型不仅仅是记住了数据中的频率偏差，而是真正学习到了区分不同情感的语义特征。

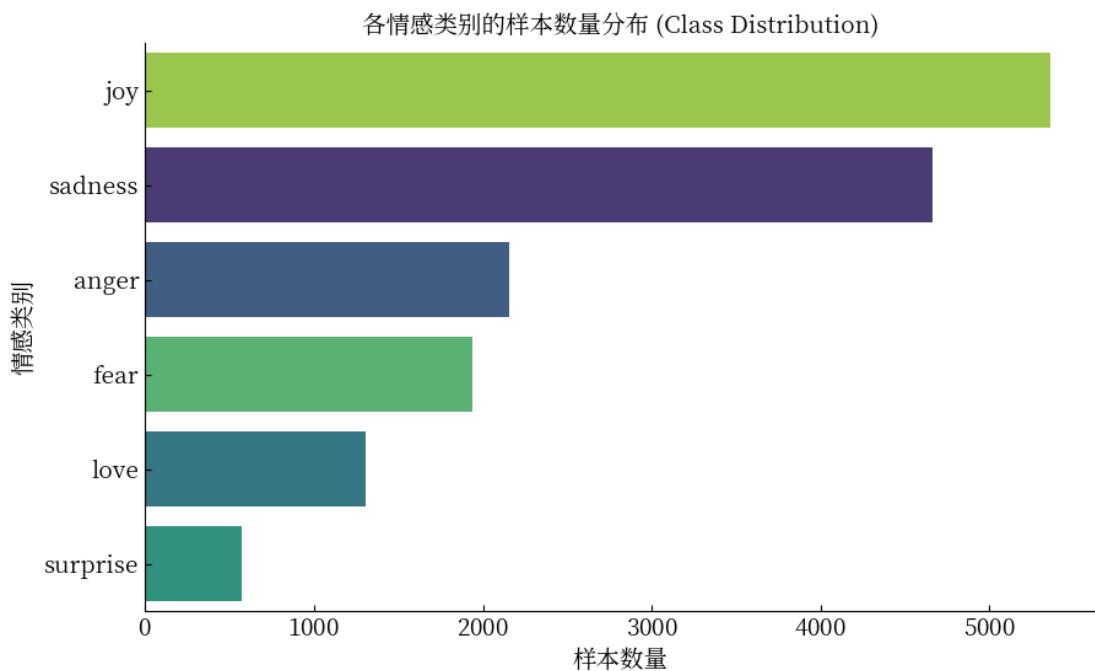


图 1 emotion 数据集均衡性检查

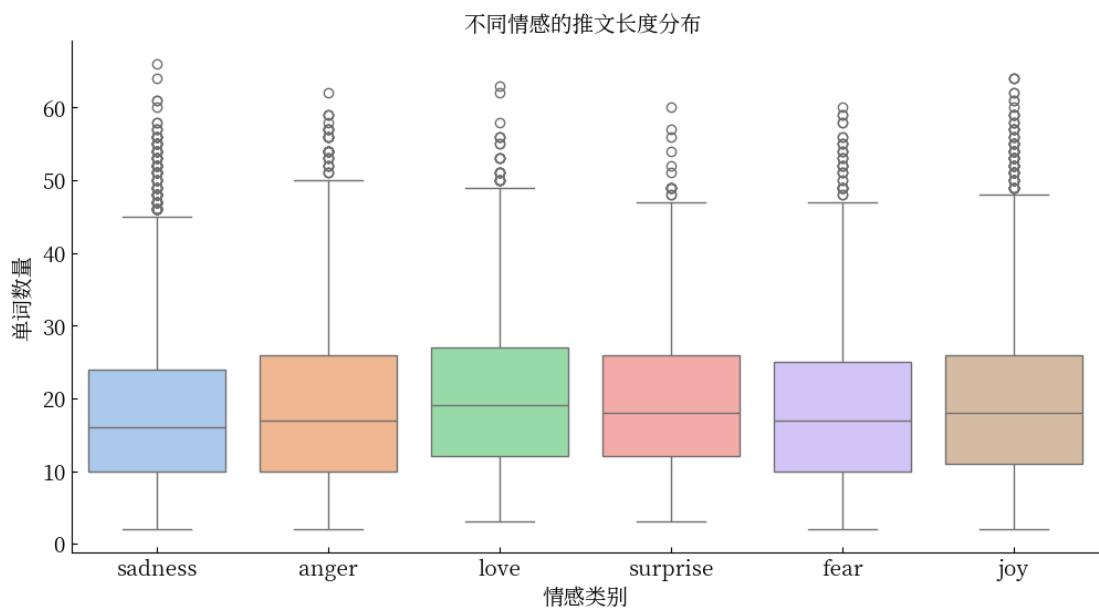


图 2 不同情感标签推文长度箱线图

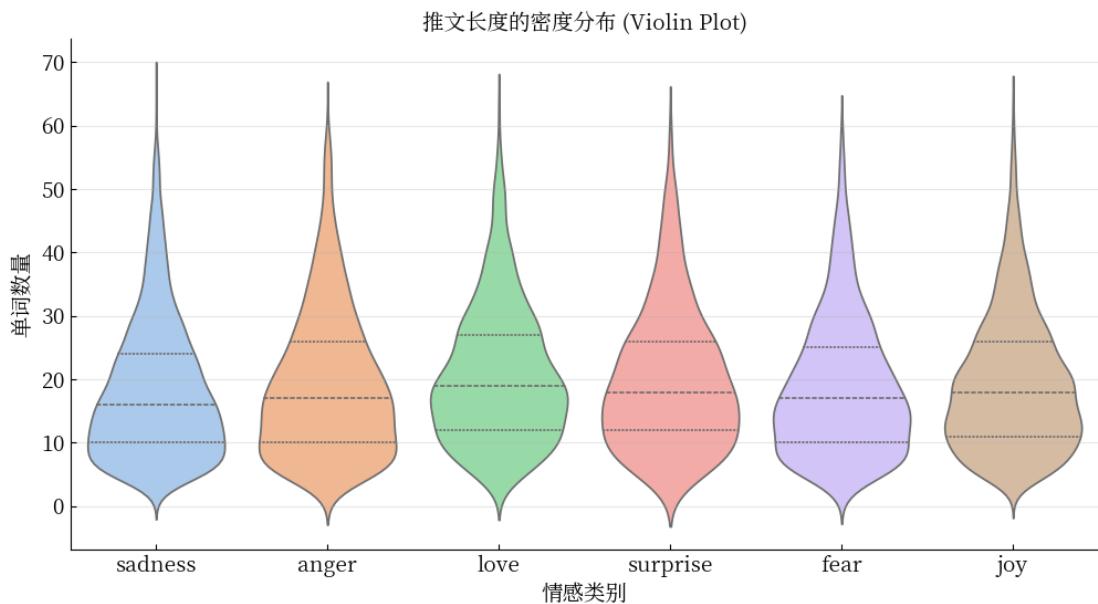


图 3 不同情感标签推文的密度分布

#### 4.2 Task1 & 2: 字符级(character-level)与字词级(subword-level)分词化(tokenization): 特征提取机制(extraction mechanism)

本实验的第一阶段旨在解决自然语言处理 (NLP) 中的核心问题：如何将非结构化的文本数据转化为计算机可计算的数值形式。我们首先通过构建字符级索引 (Character-level Tokenization) 建立了对“词表映射”这一基础概念的直观理解。通过遍历训练语料并去重排序，我们构建了一个将字符映射为唯一整数 ID 的 `token2idx` 字典。这一过程揭示了模型处理文本的最底层逻辑——即通过查表法 (Look-up) 将人类语言符号转换为机器可读的离散数字序列。然而，字符级表示虽然简单，却存在序列过长且难以捕捉词汇层级语义信息的缺陷，因此在实际的深度学习模型中，我们需要更高级的分词策略。

为了克服字符级表示的局限性，实验引入了 DistilBERT 预训练模型专用的分词器 (Tokenizer)。该分词器采用子词 (Subword) 切分算法 (如 WordPiece)，有效地在词表大小与序列长度之间取得了平衡，并解决了未登录词 (OOV) 的问题。通过调用 `dataset.map` 方法，我们对整个 Twitter 情感数据集进行了批量化处理，将原始文本转换为模型所需的 `input_ids` 和 `attention_mask`。这一步不仅实现了文本的数字化，更通过 Padding (填充) 和 Truncation (截断) 操作，确保了输入数据在张量维度上的一致性，为后续的批量计算奠定了基础。

在完成分词后，实验的核心任务转向了语义特征的提取。不同于传统的词袋模型 (Bag-of-Words)，我们利用 DistilBERT 模型的编码器 (Encoder) 作为特征提取器。通过将预处理后的 Token IDs 输入模型，我们获取了输入序列在模型最后一层隐藏层 (Last Hidden State) 的输出。具体而言，我们提取了序列第一个标记 [CLS] (Classification Token) 对应的 768 维向量。在 Transformer 架构中，[CLS] 标记被设计为汇聚整个输入序列的上下文信息，其对应的向量不仅仅是数字的堆叠，而是蕴含了深层语义和情感倾向的高维稠密表征 (Dense Representation)。

最终，通过这一系列预处理与前向传播操作，我们将原始的文本数据集成功转换为了一个高维的特征矩阵 (Feature Matrix)。在这个矩阵中，每一条推文不再是由字母组成的字符串，而是一个固定长度的数学向量。这标志着我们完成了从“符号空间”到“向量空间”的跨越，生成的 `hidden_state` 数据集已具备了输入各类机器学习分类器 (如逻辑

回归)的条件,为后续的情感分类任务提供了富含语义信息的数值基础。

---

#### 4.3 Task3: 经典机器学习模型与微调后 BERT 分类效果比对

在完成基于预训练模型的特征提取后,本实验进入核心分类任务阶段。为了量化评估深度学习微调策略的有效性,我们首先构建了一组基于 Scikit-Learn 的传统机器学习分类器作为基准(Baseline)。利用前一阶段提取的静态[CLS]向量作为输入特征,我们分别训练了逻辑回归(Logistic Regression)、支持向量机(SVM)以及随机森林(Random Forest)模型。实验结果显示,在传统模型中,逻辑回归表现最佳,达到了 63.40% 的准确率,其加权 F1 分数为 0.6220。这表明 DistilBERT 提取的高维特征在一定程度上是线性可分的,简单模型反而能比更复杂的模型更好地捕捉数据规律。相比之下,SVM(58.30%)和随机森林(52.05%)的表现则相对逊色,尤其是在“Love”和“Surprise”这类样本较少的类别上,随机森林甚至出现了 F1 分数为 0 的极端情况(Recall 仅为 0.0056),这反映了在不更新底层特征提取器的情况下,传统非线性模型在高维特征空间中容易陷入过拟合或难以收敛的困境。

随后,我们将实验策略从“冻结参数的特征提取”升级为“全参数微调(Fine-tuning)”。我们解冻了 DistilBERT 的所有权重,并在特定超参数配置下(Learning Rate = 3e-5, Batch Size = 64, Epochs = 2, Weight Decay = 0.01)对模型进行了端到端的训练。微调后的模型性能出现了质的飞跃,整体准确率飙升至 93.55%,相比最佳基准模型(逻辑回归)提升了惊人的 30.15%。从详细的分类报告来看,微调后的 BERT 在各个情感类别上均实现了极其均衡且卓越的表现,其中“Sadness”和“Joy”类别的 F1 分数分别高达 0.9646 和 0.9531,即便是样本稀缺且最具挑战性的“Surprise”类别,其 F1 分数也达到了 0.8354,远超传统模型的几乎不可用状态。

这一显著的性能差异揭示了迁移学习中两种范式的核心区别:传统机器学习方法受限于预训练模型提供的“通用语义特征”,这些特征虽然丰富但并非针对特定情感任务优化;而通过微调,DistilBERT 能够根据特定的情感分类任务动态调整其注意力机制和权重分布,将通用的语言理解能力转化为领域特定(domain-specific)的判别能力。最终的数据证明,在算力允许的情况下,端到端的微调策略能够充分释放预训练 Transformer 模型的潜力,是解决复杂自然语言分类任务的首选方案。

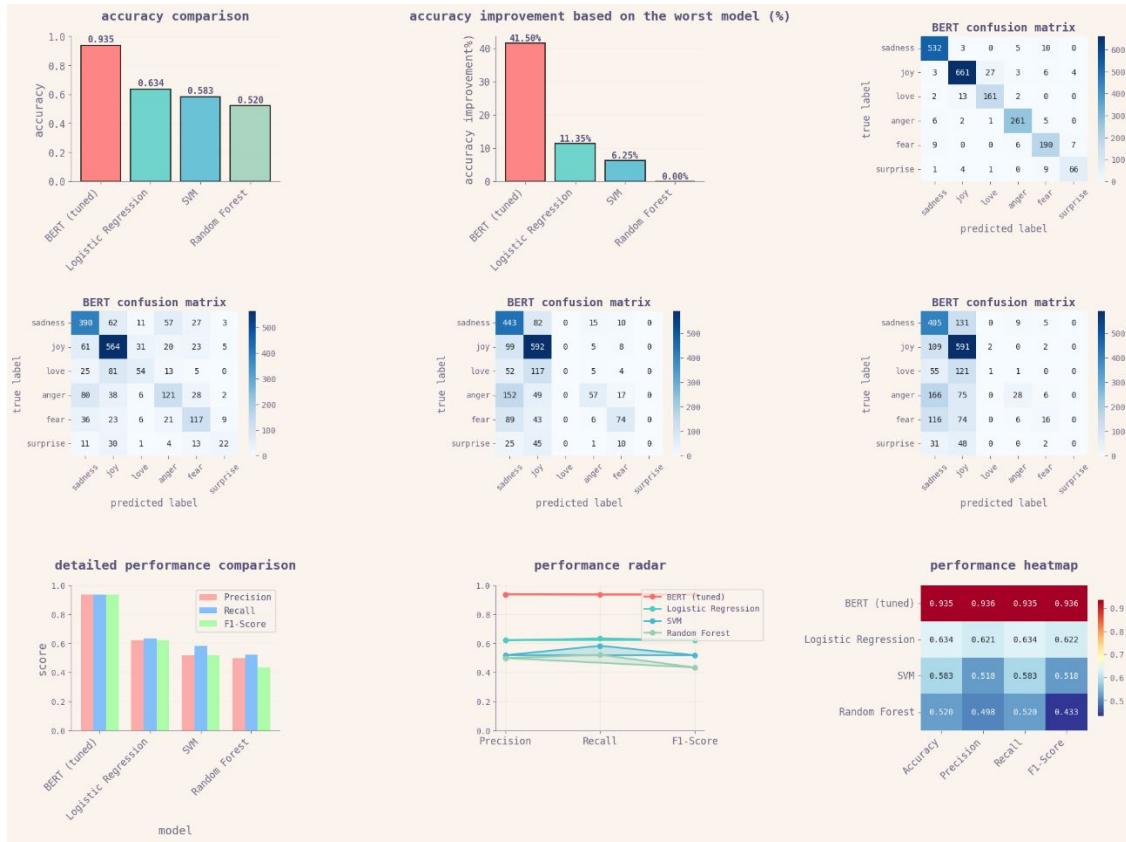


图 4 四种模型性能可视化综合比对<sup>2</sup>

表格 1 四种模型性能综合对比

SVM:		BERT (tuned):								
		precision	recall	f1-score	support	precision	recall	f1-score	support	
sadness	0.5151	0.8055	0.6284	550		0.9620	0.9673	0.9646	550	
joy	0.6379	0.8409	0.7255	704		0.9678	0.9389	0.9531	704	
love	0.0000	0.0000	0.0000	178		0.8474	0.9045	0.8750	178	
anger	0.6404	0.2073	0.3132	275		0.9422	0.9491	0.9457	275	
fear	0.6016	0.3491	0.4418	212		0.8636	0.8962	0.8796	212	
surprise	0.0000	0.0000	0.0000	81		0.8571	0.8148	0.8354	81	
accuracy			0.5830	2000		accuracy		0.9355	2000	
macro avg	0.3992	0.3671	0.3515	2000		macro avg	0.9067	0.9118	0.9089	2000
weighted avg	0.5180	0.5830	0.5181	2000		weighted avg	0.9365	0.9355	0.9358	2000
Random Forest:										
		precision	recall	f1-score	support	precision	recall	f1-score	support	
sadness	0.4592	0.7364	0.5656	550		0.6468	0.7091	0.6765	550	
joy	0.5683	0.8395	0.6778	704		0.7068	0.8011	0.7510	704	
love	0.3333	0.0056	0.0110	178		0.4954	0.3034	0.3763	178	
anger	0.6364	0.1018	0.1755	275		0.5127	0.4400	0.4736	275	
fear	0.5161	0.0755	0.1317	212		0.5493	0.5519	0.5506	212	
surprise	0.0000	0.0000	0.0000	81		0.5366	0.2716	0.3607	81	
accuracy			0.5205	2000		accuracy		0.6340	2000	
macro avg	0.4189	0.2931	0.2603	2000		macro avg	0.5746	0.5128	0.5314	2000
weighted avg	0.4982	0.5205	0.4332	2000		weighted avg	0.6212	0.6340	0.6220	2000
Logistic Regression:										
		precision	recall	f1-score	support	precision	recall	f1-score	support	
sadness	0.6468	0.7091	0.6765	550		0.6468	0.7091	0.6765	550	
joy	0.7068	0.8011	0.7510	704		0.7068	0.8011	0.7510	704	
love	0.4954	0.3034	0.3763	178		0.4954	0.3034	0.3763	178	
anger	0.5127	0.4400	0.4736	275		0.5127	0.4400	0.4736	275	
fear	0.5493	0.5519	0.5506	212		0.5493	0.5519	0.5506	212	
surprise	0.5366	0.2716	0.3607	81		0.5366	0.2716	0.3607	81	
accuracy			0.5205	2000		accuracy		0.6340	2000	
macro avg	0.5746	0.5128	0.5314	2000		macro avg	0.5746	0.5128	0.5314	2000
weighted avg	0.6212	0.6340	0.6220	2000		weighted avg	0.6212	0.6340	0.6220	2000

<sup>2</sup> 注意后三幅混淆矩阵的标题均错误显示为 BERT confusion matrix，实际上是因为我在尝试运用 f 格式字符串编辑 {model\_name} 时出现疏忽。三幅图表实际上依次分属于逻辑回归、支持向量机与随机森林。

#### 4.4 Task4: 超参数敏感性分析与模型优化策略

表格 2 超参数调整实验设置

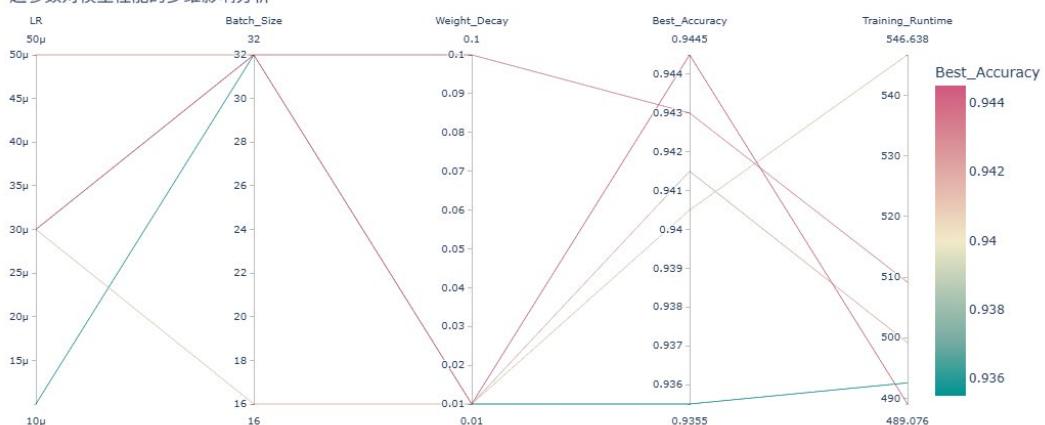
实验组别	实验 ID	学习率 (LR)	Batch Size	Weight Decay	Epochs
A. 基准组	Baseline	3e-5	32	0.01	4
B. 学习率组	Low_LR	1e-5	32	0.01	4
	High_LR	5e-5	32	0.01	4
C. 批次组	Small_BS	3e-5	16	0.01	4

在验证了经过参数微调之后的 BERT 相对于传统机器学习方法的巨大优势后，下一阶段实验旨在探究模型性能对关键超参数的敏感性，以确定最佳的训练配置。我们对训练轮数变量 Epochs 进行控制，设计了四组对比实验（Baseline, High\_WD, High\_LR, Small\_BS, Low\_LR），重点考察学习率（Learning Rate）、批次大小（Batch Size）和权重衰减（Weight Decay）对验证集准确率及损失函数的边际效应。

表格 3 准确度与训练时长对比

	Experiment	LR	Batch_Size	Weight_Decay	Best_Val_Loss	Best_Accuracy	Training_Runtime
0	Baseline	0.00003	32	0.01	0.137867	0.9445	489.0757
4	High_WD	0.00003	32	0.10	0.136634	0.9430	509.1513
2	High_LR	0.00005	32	0.01	0.142960	0.9415	499.1923
3	Small_BS	0.00003	16	0.01	0.176577	0.9405	546.6383
1	Low_LR	0.00001	32	0.01	0.164264	0.9355	492.5596

超参数对模型性能的多维影响分析



## 1. 学习率的主导作用与相关性分析

实验数据表明，学习率是影响模型性能的最关键因子。通过相关性热力图分析发现，学习率与最佳准确率之间存在显著的正相关关系（Pearson 相关性系数 = 0.62），远高于批次大小（0.082）和权重衰减（0.33）。

低学习率的局限性：Low\_LR 组 ( $1e-5$ ) 的准确率仅为 93.55%，显著低于其他组别。这表明过低的学习率导致模型收敛速度过慢，未能在既定的 4 个 Epoch 内逃离局部最优解，陷入了欠拟合状态。

最优区间的确认：基准组 ( $3e-5$ ) 取得了全场最高的准确率 94.45%，而 High\_LR 组 ( $5e-5$ ) 虽然表现尚可 (94.15%)，但相比基准略有下降且 Loss 略高 (0.1429)，暗示过大的步长可能导致模型在极值点附近产生震荡。因此， $3e-5$  被确认为该数据集下的“黄金学习率”。

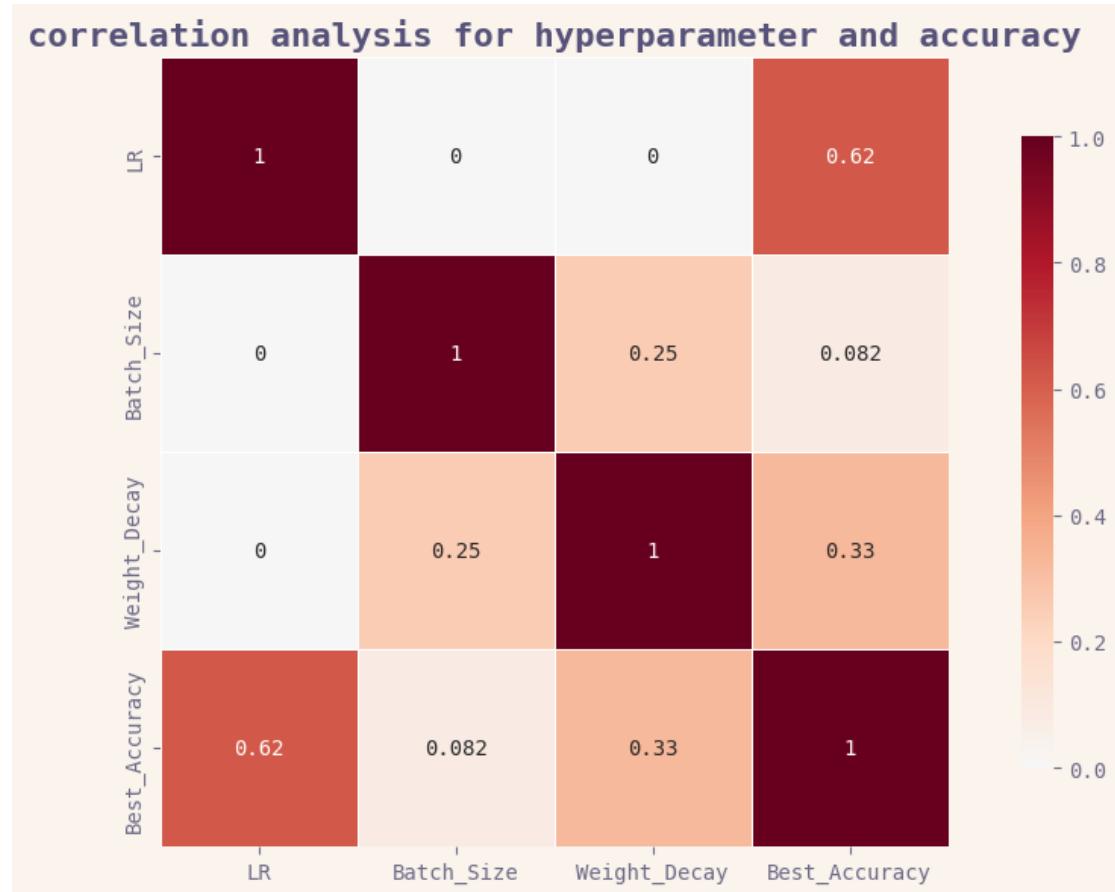


图 5 超参数与准确率的相关性系数矩阵

## 2. 正则化与泛化能力的权衡

High\_WD 组 (Weight Decay = 0.1) 提供了一个极具价值的观察视角。尽管其准确率 (94.30%) 略低于 Baseline (94.45%)，但它取得了全场最低的验证集损失 (0.1366)。这一结果揭示了正则化的深层作用：通过施加更强的 L2 惩罚，模型在牺牲极微小 (0.15%) 训练集拟合精度的情况下，换取了更稳健的特征表示和更低的过拟合风险。在实际部署场景中，若优先考虑模型的鲁棒性与校准度 (Calibration)，High\_WD 的配置可能优于准确率最高的 Baseline。

## 3. 批次大小与计算效率

针对 Small\_BS 组 (Batch Size = 16) 的实验显示，减小批次大小并未带来预期的性能提升。该组准确率为 94.05%，且验证集 Loss 显著升高至 0.1765，表现出较大的训练波动性。更重要的是，从计算效率角度看，Small\_BS 组的训练时长达到 546.6 秒，相比基准(489.1 秒)增加了约 12% 的时间成本。这说明在该规模的数据集上，缩小 Batch Size 引入的梯度噪声并未有效帮助模型优化，反而降低了并行计算效率。

## 小结

综上所述，本次超参数调优实验证实了 DistilBERT 微调过程对学习率高度敏感，而对批次大小相对鲁棒。综合考虑准确率、泛化能力与训练效率，可以确定基准配置 (LR=3e-5, BS=32, WD=0.01) 为最优解，该配置实现了 94.45% 的分类准确率与 0.1379 的低损失值，达到了性能与效率的最佳平衡。

---

### 4.5 Task5：BERT 错误分析与模型局限性探究

在最初的模型微调阶段（即源代码中的模型参数），我们采用了较为保守且稳定的超参数配置 (Learning Rate=2e-5, Batch Size=64)，这使得 DistilBERT 在 2 个 Epoch 内实现了良好的收敛，在验证集的 2000 个样本中仅产生了 142 个错误分类，整体错误率控制在 7.10%。在本部分，我们将重访考察这一模型配置。为了深入理解模型的决策边界与局限性，我们结合 EDA 阶段发现的类别不平衡现象，通过混淆矩阵 (Confusion Matrix) 与损失排序 (Loss Sorting) 技术，对这部分“失败样本”进行了多维度的可视化分析。

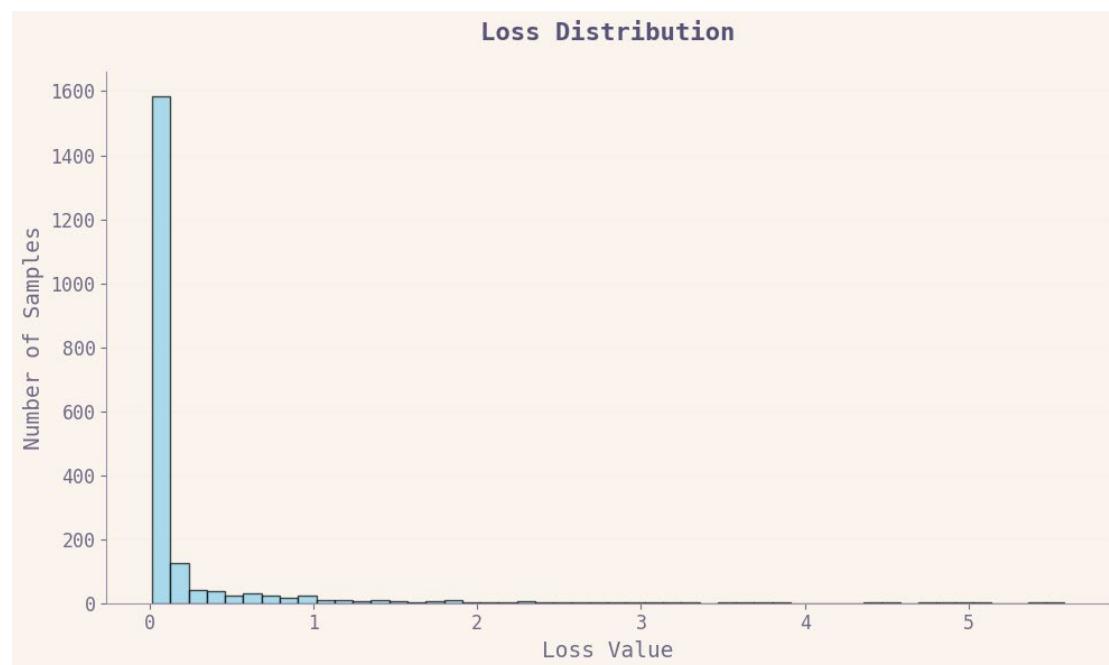


图 6 样本损失函数贡献分布

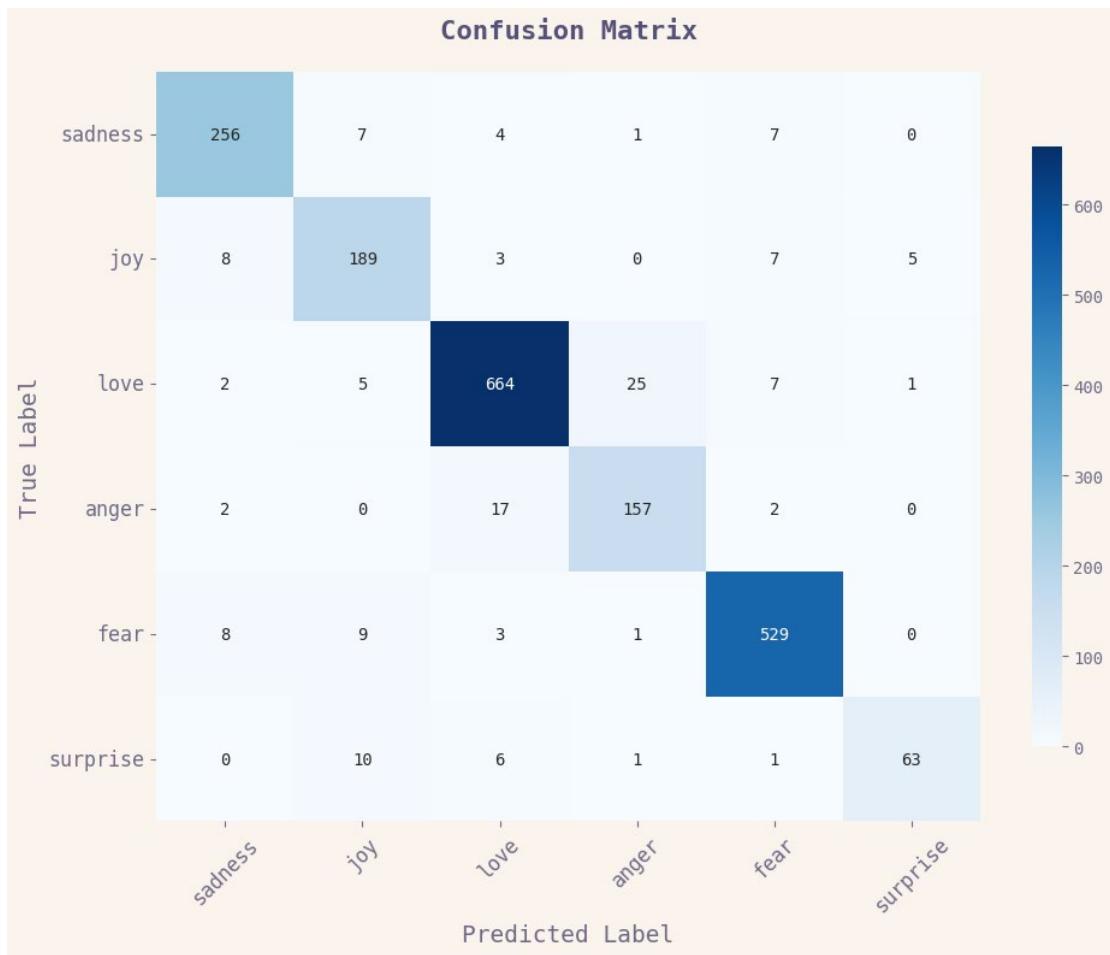


图 7 模型混淆矩阵

首先，观察混淆矩阵的热力图分布，可以清晰地发现数据集的**类别不平衡（Class Imbalance）**对模型预测偏好产生了显著影响。尽管整体准确率较高，但错误主要集中在语义相近或样本稀缺的类别之间。最典型的现象发生在“Love”（爱）与“Joy”（喜悦）之间，以及“Surprise”（惊讶）与“Fear”（恐惧）之间。由于“Love”在语义上往往被包含于广义的“Joy”之中，且“Joy”作为训练集中的主导类别（Majority Class），模型在面对表达含蓄的“爱意”推文时，倾向于将其归类为出现频率更高的“Joy”。同样，“Surprise”作为样本量最少的长尾类别，模型难以充分学习其特征，往往将其误判为情感效价相似的“Fear”或“Joy”。这种系统性的偏差表明，虽然预训练模型具有强大的上下文理解能力，但在面对极度不平衡的数据分布时，仍会表现出向大类偏移的“多数类偏见”。



图 8 诸标签错误率

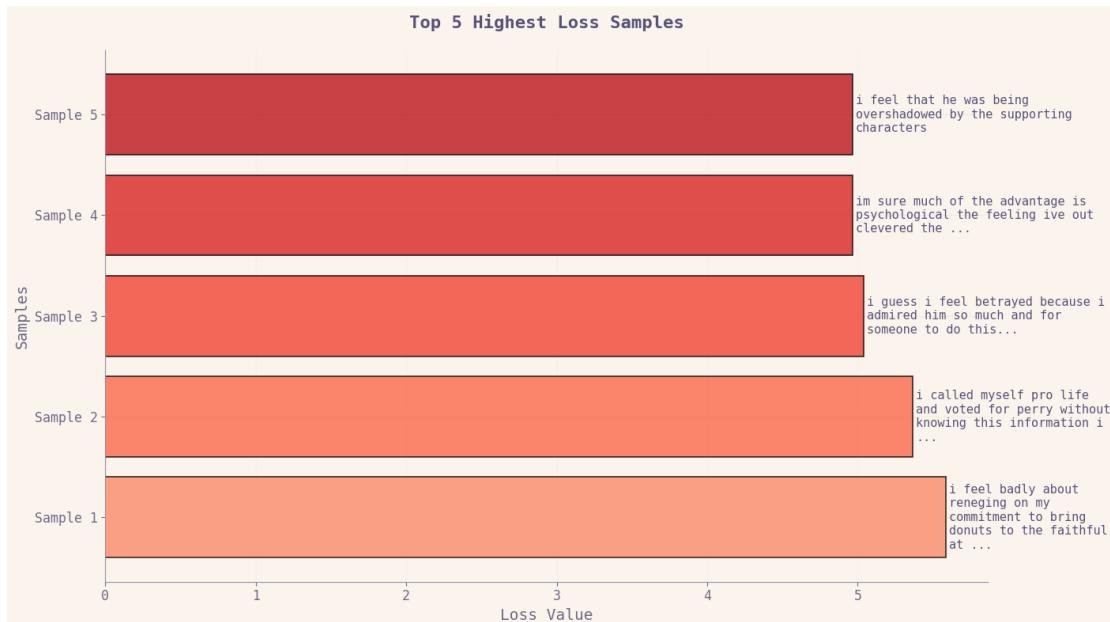


图 9 高损失贡献样本

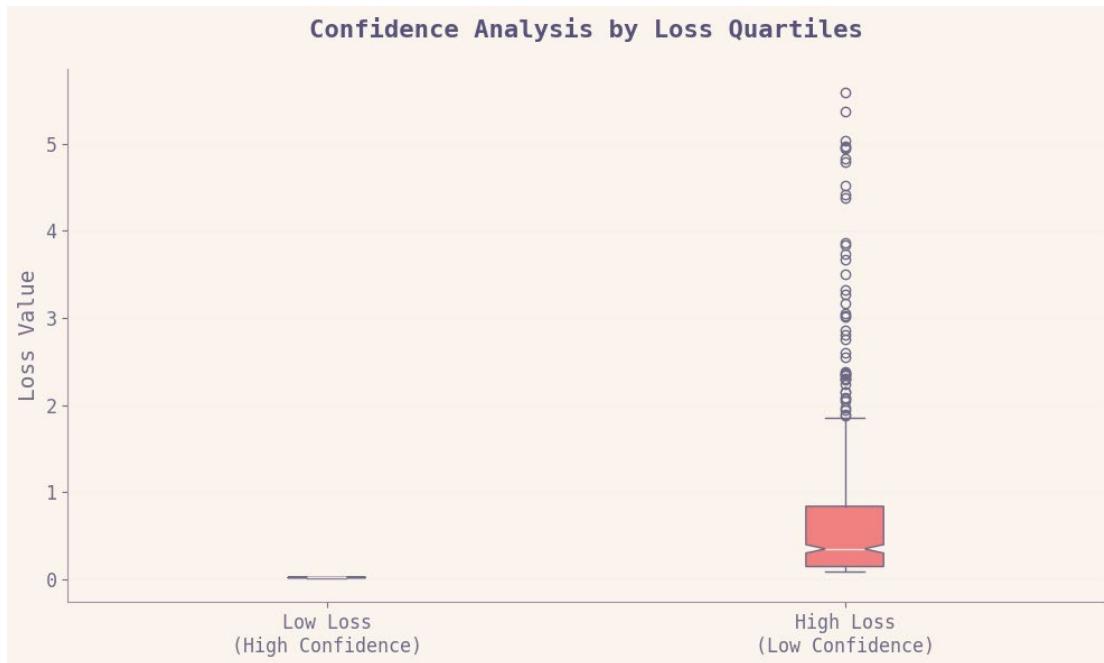


图 10 置信度-损失贡献对比箱线图

进一步地，我们通过计算每个样本的交叉熵损失（Cross-Entropy Loss）并按降序排列，筛选出了模型“最困惑”或“最确信但错误”的样本。对 Top-Loss 样本的定性文本分析揭示了导致分类错误的深层原因，主要可归纳为标注噪声（Label Noise）与语义多义性（Semantic Ambiguity）。

在部分高损失样本中，我们认为，模型的预测实际上在语义上是合理的，反而是原始数据集的 Ground Truth 标签存在争议。例如，某些表达强烈失落感的推文被人工标注为“Anger”，而模型将其预测为“Sadness”，这种不一致性反映了情感本身的主观性和模糊性。此外，部分推文包含复杂的修辞（如反讽）或混合情感（如“悲喜交加”），DistilBERT 虽然能够捕捉到局部的关键词，但在缺乏更长上下文或外部常识知识库辅助的情况下，难以准确解析这种高阶的语用逻辑。

综上所述，DistilBERT 在处理标准情感表达时表现出了卓越的性能，其 92.9% 的准确率证明了微调策略的成功。然而，剩余 7.10% 的错误并非随机分布，而是集中暴露了单一模态模型在处理细粒度情感区分（如 Love vs Joy）以及长尾小样本学习（Surprise）时的短板。未来的改进方向应着重于数据增强（对小样本类别进行过采样）或引入加权损失函数（Weighted Loss），以矫正模型对主导类别的过度依赖，从而提升其在复杂语境下的鲁棒性。

#### 4.6 Task6：对抗样本测试与模型鲁棒性边界分析

在实验的最终阶段，为了探究 DistilBERT 模型在面对复杂、非典型语言表达时的鲁棒性边界，我们人为构建了一组包含反语、逻辑转折、双重否定及俚语的对抗样本（Adversarial Examples）。这组样本的设计逻辑旨在打破模型对“特定词汇对应特定情感”的简单依赖，测试其是否真正具备了结合语境（Context）与语用学（Pragmatics）进行深层推理的能力。例如，我们引入了“I absolutely love waiting in line...”（反讽）来测试模型能否识别语义反转；利用“...but I fell asleep”（转折）来考察模型对长距离依赖和句子重心的判断；以及使用“This movie is sick!”（俚语）来检测模型对多义词在特定文化语境下的适应性。

测试结果显示，尽管模型在标准测试集中表现优异，但在面对这些人造陷阱时却暴露出了显著的认知短板，主要表现为词汇表面特征的过度依赖（Over-reliance on Lexical Cues）。

具体而言，在反讽句（样本 1）中，模型被“love”和“Best day”这两个强烈的正向情感词误导，完全忽略了“waiting in line in the rain”这一消极语境的常识性暗示，错误地将其预测为“Joy”。这表明模型更多地是在进行加权词袋式的统计，而非真正理解反讽修辞中的“言行相悖”。同样的机制也导致了转折句（样本 2）的误判，尽管后半句“fell asleep”清晰地表达了无聊，但模型似乎被前半句的“promising”和“great”所占据，未能正确处理“but”所带来的极性翻转，显示出注意力机制在处理冲突信息时的局限性。

尤其是对于反语（样本 3），DistilBERT 给出了高达 **98%** 的“Joy”置信度。这表明模型内部的注意力机制（Attention Mechanism）被句子中显性的强情感词 love 和 Best day 完全捕获，分配了极高的权重。尽管 rain 和 waiting 在语义上暗示了负面情境，但它们在情感词典中往往属于中性或弱负向词。模型未能识别出正向词汇与负向语境之间的**语义不一致性**（Semantic Incongruity），即反讽的核心特征。这种对局部关键词的过度拟合，导致模型在语用学层面完全失效，表现为典型的“词袋模型”行为。

### 3. 双重否定

句子: I can't say that I didn't dislike the ending.

预测: anger (置信度: 0.928)

详细得分:

anger:	0.9277
love:	0.0594
sadness:	0.0090
joy:	0.0017
fear:	0.0012
surprise:	0.0010

### 4. 倔语

句子: This movie is sick! I screamed!

预测: anger (置信度: 0.929)

详细得分:

anger:	0.9289
fear:	0.0656
sadness:	0.0026
surprise:	0.0020
joy:	0.0007
love:	0.0002

图 11 双重否定与倔语具体预测得分

此外，模型在处理领域特定语言与逻辑推理时也显现出“缺乏世界知识（World Knowledge）”的特征。在倔语测试（样本 4）中，模型将“sick”（在流行语中意为“酷/精彩”）按其字面意思理解为“生病/恶心”，从而错误预测为“Anger”；在比较级陷阱（样本 5）中，模型未能理解“better than a root canal”（比根管治疗好）实际上是一种极低标准的评价，反而因捕捉到“better”一词而误判为“Joy”。综上所述，这一系列对抗性测试揭示了当前微调后的 DistilBERT 模型的一个核心本质：它依然是一个基于概率统计的高级模式识别器，而非具备逻辑推理能力的理解者，更远远达不到 LeCun 近来强调的基于“因果性”的世界模型（world model）。它能够敏锐地捕捉情感关键词，但在面对需要常识推理、文化背景支持或复杂逻辑运算的语言现象时，仍显得脆弱且容易被愚弄。这为未来的改进指明了方向——即需要引入对抗训练（Adversarial Training）或结合知识图谱，以增强模型在非规范语境下的泛化能力。

```
--- Testing Custom/Adversarial Examples ---
```

1. 反语/讽刺

句子: I absolutely love waiting in line for three hours in the rain. Best day ever.  
预测: joy

2. 转折句

句子: The plot was promising and the actors were great, but I fell asleep in the first 20 minutes.  
预测: joy

3. 双重否定

句子: I can't say that I didn't dislike the ending.  
预测: anger

4. 哀语

句子: This movie is sick! I screamed!  
预测: anger

5. 比较级陷阱

句子: It was better than getting a root canal, I guess.  
预测: joy

图 12 对抗样本预测结果

## 反思与总结

相比于上次计算机视觉项目，在处理本次作业时，我已经能更加从容地适应 torch 语法和深度学习模型建构与评估的通用思路与指标。例如，我更为科学合理地在 vs code 中配置了 colab 的免费 GPU 资源，将全部模型训练都在本地实现；超参数调优的实验设置更为有效，模型性能分析更为融贯，可视化绘图也更加成熟。但最重要的是，我真正意义上尝试了解自己每天都在使用的大语言模型的底层逻辑，体会到词嵌入方法的发展已经能够把握语言深层的指称与意义，这让我大为震撼。我希望自己能继续强化自己应用大语言模型的经验，在社会科学研究领域做出探索。