

负面内容检测

修订记录

日期	修订版本	修订编号	修改章节	修改描述	作者

1. 项目概述——产品补充

1.1 项目背景

随着移动互联网的飞速发展，移动云盘作为用户存储、分享文件的重要平台，其内容的健康与安全日益受到社会关注。在海量图片使用相册的过程中，不可避免地会出现负面内容的图片，这些图片不仅可能违反国家法律法规，还可能对用户造成心理不适，影响平台形象及用户体验。因此开发一套高效、准确的负面内容检测系统，对移动云盘中的图片进行实时监测与过滤，显得尤为重要。

1.2 项目目标与预期成果

项目构建一套集成于算法中心平台的负面内容检测系统，重点针对涉政、消极负面内容（包括但不限于军队、警察、杀人、血腥、尸体、车祸、爆炸火灾、自然灾害、战争废墟、白事、医院、自杀等场景）的图片进行智能识别与过滤，预期成果包括：

阶段一	死亡和丧葬相关物品、凶器或暴力相关的物品、暴力或血腥场景、疾病或痛苦、自然灾害或灾难现场、悲剧或悲伤场景
阶段二	污染和垃圾、动物虐待或死亡、恶心或令人不适的画面、恐怖或惊悚元素
阶段三	象征厄运的符号或物品、负面情感的表情或场景、破损或损坏的物品、腐烂或衰败的景象

1.3 项目范围界定

本项目范围是开发一套先进、高效的负面内容图片检测系统至算法中心平台中。该系统将运用深度学习技术，特别是卷积神经网络（CNN）等先进算法，对云盘的图片进行智能分析，自动识别并过滤掉涉及政治敏感、消极负面内容的图片，包括但不限于军队、警察、暴力事件、血腥场景、自然灾害、战争废墟、医院及自杀等敏感或不适宜公开展示的内容。

2. 需求分析——产品补充

2.1 系统需求规格

系统需求规格如下：一是准确性要求，检测模型需具备高准确率，能够准确识别并分类负面内容的图片；二是实时性要求，系统需具备快速响应能力，对上传的图片进行实时检测与处理；三是可扩展性要求，系统需支持新类型负面内容的添加与识别，以满足未来可能的需求变化；四是安全性要求，系统需确保数据传输与存储的安全性，防止敏感信息泄露；五是系统需具备日志记录、监控报警等辅助功能，以便于问题追踪与故障排查。

2.2 产品需求总概

产品版本	排期计划	需求	实际版本情况
v1.0	2024年8月版本	实现对死亡和丧葬相关物品、凶器或暴力相关的物品、暴力或血腥场景、疾病或痛苦、自然灾害或灾难现场、悲剧或悲伤场景的负面识别检测	

2.3 功能需求明细

v1.0.版本需求明细

【产品原型】 <https://docs.qq.com/sheet/DUWhwZWFwWFRzdnlm?tab=n6lspq>

【功能建设模块】

- 实现对死亡和丧葬相关物品、凶器或暴力相关的物品、暴力或血腥场景、疾病或痛苦、自然灾害或灾难现场、悲剧或悲伤场景的负面识别检测

【排期计划】

【遗留问题】

3. 技术预研——算法补充

3.1 技术趋势概述

3.2 技术调研/竞品分析摘要

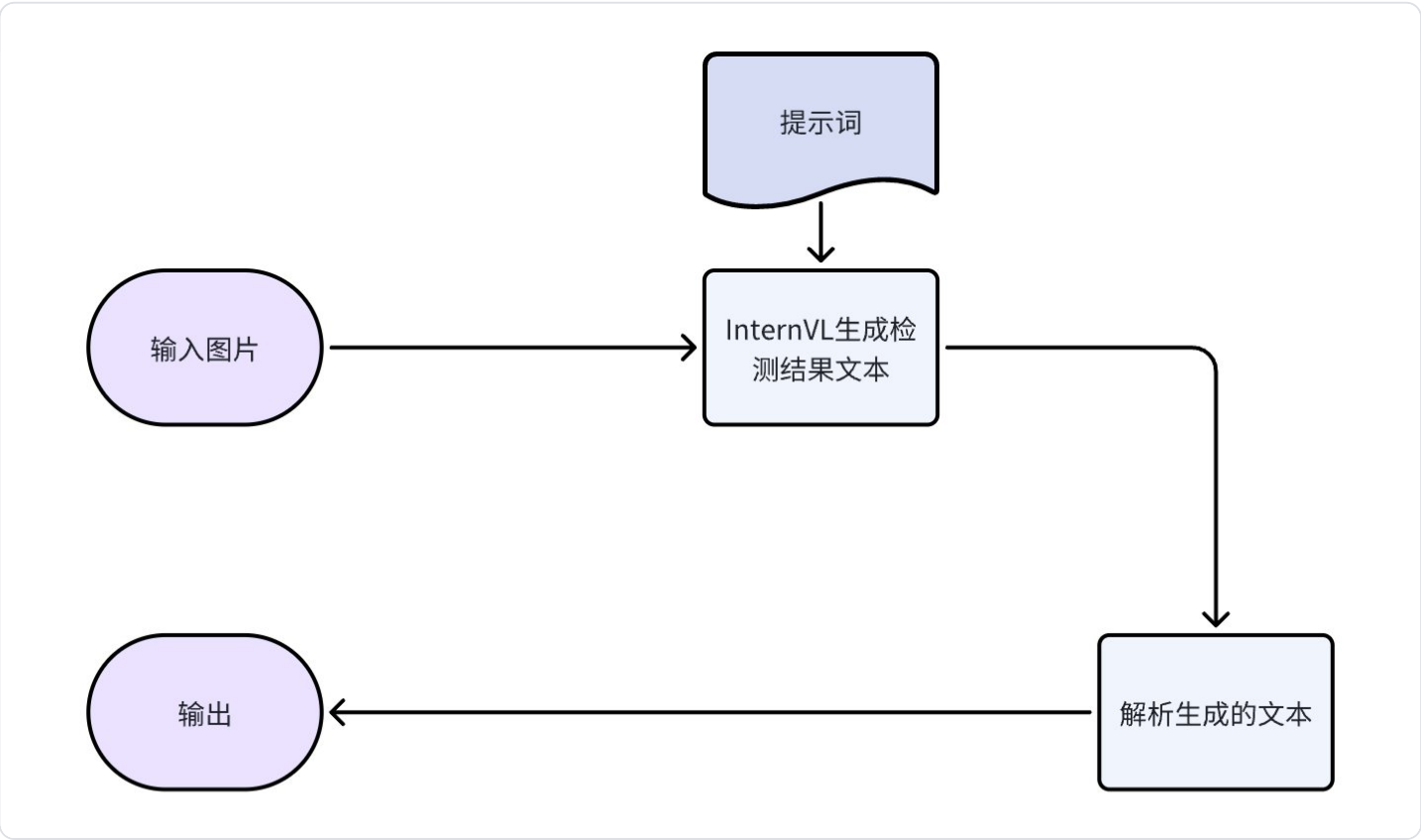
4. 算法服务系统设计——算法补充

需要包含算法架构图，流程图，算法模块中算法模型的选型，接口定义与接口建设，如为大模型提示词，需要明确设计思路。

如果设计训练需要补充训练过程，参数调优

4.1 系统架构设计

4.2 算法流程图



4.3 负面图片检测算法介绍

基于多模态大模型的负面图片检测

多模态大语言模型(MLLM) 将 LLM 的语言能力扩展到多模态领域，能够处理和理解多种模态的信息。MLLM 不仅可以理解文本，还可以分析图像、音频和视频等数据、并将它们与文本信息进行关联和融合，从而实现更全面、更深入的理解。视觉多模态大模型的应用场景包括视觉问答、图文生成等，模型根据图像内容回答文本问题，这需要同时理解图像和文本信息，并进行融合处理。

目前支持中文的开源可商用的多模态大模型主要有QwenVL、InternVL、miniCPM-V、GLM-4V等

InternVL Family: A Pioneering Open-Source Alternative to GPT-4V. 接近GPT-4V表现的可商用开源多模态对话模型。于7月4日发布InternVL2系列。

在线demo: <https://huggingface.co/spaces/OpenGVLab/InternVL>

Qwen-VL 是阿里云研发的大规模视觉语言模型（Large Vision Language Model, LVLM）。Qwen-VL 可以以图像、文本、检测框作为输入，并以文本和检测框作为输出。

https://github.com/QwenLM/Qwen-VL/blob/master/README_CN.md

提示词：

```
1  # 任务\n你是一个图片审核人员，需要判断用户提供的图片是否包含负面信息\n# 条件\n- 判断图片是否包含死亡人物。 \n- 判断图片是否与葬礼有关。 \n- 判断图片是否包含枪支、刀具。 \n- 判断图片是否与暴力、打架相关。 \n- 判断图片是否包含血腥场景。 \n- 判断图片是否与疾病、痛苦相关。 \n- 判断图片是否为自然灾害、废墟场景。 \n如果符合上述条件，则以json形式返回\n{\"result\": True}, 如果没有则以json形式返回{\"result\": False}
```

4.4 进展-20241113-负面检测算法

1. 当前算法思路

当前负面图片检测算法利用8B大模型的底层认知能力结合简单的prompt工程技术进行二分类。这种方法的核心在于设计有效的prompt，使其能够引导模型判断图片是否包含负面元素。然而，直接让模型区分正面与负面图片存在一定的挑战，因此我们采用了一种间接的方法：通过多模态模型对图片中的元素进行检测，以识别可能归类为负面的元素。

2. 目前已测试的负面情绪词汇

测试图片共335张。词汇列表：车祸，警察，血腥，自然灾害，自杀，聚餐，白事，战争废墟，医院，军队，其他等。

负面优化1-共194张。负面情绪：鬼怪动画等。

负面优化2-共427张：负面情绪的表情或场景，被遗弃的宠物，破碎的镜子，破损的家具，污染的水源/垃圾堆/废弃物等，断裂的物品，战争，在医院接受治疗的图片，自然灾害或灾难现场，象征厄运的符号或者物品等。

以上测试内容具体参见-自测报告: <https://www.kdocs.cn/l/cqJx0Btas7xg>

测试：测试图片110张。词汇列表：白事、暴力&血腥&刀具、悲伤/悲剧、车祸、医院&疾病、自杀、战争废墟&军队&枪支、自然灾害、自杀、正常图片。**详细测试报告&测试用例，详见6.4 测试用例与报**

3. 未来可能优化方向

- 1) **建立负面图片向量库，提高输出稳定性：**类似于图配文模块，我们可以构建一个负面图片的向量库。当接收到新的图片时，系统首先会在负面库中进行检索，寻找相似的图片。如果检索到的图片与待检测图片的相似度超过预设的阈值，系统可以直接返回负面判断结果，从而避免了对大模型的推理过程。
- 2) **提升响应速度：**通过上述向量库检索机制，可以在某些情况下显著提升系统的响应速度。这是因为向量库检索通常比复杂的模型推理过程要快得多，尤其是在处理大量图片时，这种速度优势尤为明显。
- 3) **动态阈值调整：**是否可以考虑引入动态阈值调整机制。根据图片的复杂度和历史检测结果，系统可以自动调整相似度阈值，以实现更精细的控制。
- 4) **模型微调：**在保持模型基本结构不变的情况下，考虑对模型进行微调，以更好地适应负面图片检测的任务。这可能包括调整模型的某些层的权重，以强化模型对负面元素的识别能力。

5. 接口设计——算法补充

输入图片,返回事物标签列表

5.1.1 请求类型

post

5.1.2 请求Url

{BaseURL}+/{厂商标识}/+yun/ai/image/sentiment

5.1.3 请求参数

参数名称	必填	类型	描述信息
requestId	M	String	请求ID
sendType	M	Int	传送类型：1——url传送2——base64传送3——文件信息4——共享存储的文件路径
fileUrl	O	String	图像下载地址
base64	O	String	图片Base64编码的内容,需要需要去掉编码头部。
fileInfo	O	FileInfo	文件信息
localPath	O	String	共享存储文件路径

5.1.4 响应参数

参数名称	必填	类型	描述信息
requestId	M	String	请求ID
thingList	O	<ThingLabel>	物体标签检测结果

6. 测试计划——测试补充

6.1 测试目标

- 1、评估各种负面场景下的检测精度和可靠性，从而有效过滤和标记负面内容，提升系统的安全性和用户体验。
- 2、接口的规范性、功能性、时延

6.2 测试指标

一、精确率（Precision）

定义：在所有被模型预测为正类的样本中，真正为正类的样本所占的比例

计算公式：Precision = TP / (TP + FP)，其中TP是真正例，FP是假正例

二、召回率（Recall）

定义：在所有真实为正类的样本中，被模型正确预测为正类的样本所占的比例

计算公式：Recall = TP / (TP + FN)，其中FN是假负例

三、F1分数（F1-Score）

定义：准确率和召回率的调和平均数，用于综合衡量模型的性能

计算公式：F1-Score = 2 * (Precision * Recall) / (Precision + Recall)

四、准确率（Accuracy）

定义：预测正确的结果占总样本的百分比

计算公式：Accuracy = TP + TN / TP + TN + FP + FN

五、接口时延

6.3 测试策略与方法

测试标准

功能测试：（每个场景的图片数量需分布平均）

- 1、不同负面场景图片是否能够通过检测，返回正确的错误码。

2、正常场景下的图片是否能够正常通过检测

性能测试：

不同图片大小的接口时延（平均值）、推理耗时（平均值）

6.4 测试用例与报告

图片覆盖的场景包括：

阶段一：

白事、暴力&血腥&刀具、悲伤/悲剧、车祸、医院&疾病、自杀、战争废墟&军队&枪支、自然灾害、自杀、正常图片

阶段二：

待补充

算法功能性测试报告

2024-11-14




负面检测V1.1.0算法效果测试报告.docx

21.07MB




用例110张图片



负面情绪图片.rar

20.17MB



接口耗时(秒)	推理耗时(秒)	图片名称
1	0.64	白事 (1).jpeg
4.37	0.71	白事 (2).jpeg
0.98	0.54	白事 (3).jpeg
1.03	0.63	白事 (4).jpeg
3.79	0.67	白事 (5).jpeg
2.6	0.98	白事 (6).jpeg
1.05	0.62	白事 (7).jpeg

 【1113-二次回归】负面检测数据.xlsx

7. 监控日志——算法补充

7.1 日志结构与内容说明

时间|ip地址|主机名|服务名|进程|线程|日志级别|日志内容|请求信息|

日志内容：主要包括接口返回信息

7.2 具体日志示例

7.2.1 单元日志内容示例

7.2.4 完整日志内容示例

```
2024-09-11 16:17:06.070|192.168.145.205|yun-ai-image-captioning-76b6d6d4-zhxr2|ai service; image_captioning|p13t:3986472721128|INFO|API|success; Result={'code': '0000', 'message': '成功', 'data': {'requestId': '07f5ed3b61cc4e76ad794f6b47eb5383', 'fileId': 'PpUeJa7oA9d8f1747MFLdfrJ8a30C6n', 'thingList': [], 'success': True}; costtime=4.9|requestId:07f5ed3b61cc4e76ad794f6b47eb5383|fileId:PpUeJa7oA9d8f1747MFLdfrJ8a30C6n|sendType:4|localPath:/kcs_data/yun-ai-metadata-test/yun-ai-node-task/2024/09/11/719/e6b2dfb83a3473bb301375266917aa2_downloadFile.ME
```

8. bug跟进与修复——产品补充

算法版本	意图识别BUG列表	修复时间