2026 第二届大学生人工智能安全竞赛
——定向式专项命题作品赛参赛指南

竞赛的目的是为培养、选拔、推荐优秀人工智能安全人才创造条件，促进高等学校网络空间安全和人工智能专业课程体系、教学内容和方法的改革，培养学生的创新意识与团队合作精神，普及人工智能安全知识，增强学生人工智能安全意识。

本指南为学生、指导教师和高校如何参与本次人工智能安全竞赛定向式专项命题作品赛提供具体指导。

一、学生参赛及报名

1.报名截止日期内具有正式学籍的全日制在校本科生、专科生均可报名参赛（硕士生和博士生不能参赛）。评审时，如发现参赛队员不符合参赛规定，将取消参赛队伍的参赛或获奖资格。

2.每支参赛队不超过 4 名学生（包括 1 名组长），每支参赛队限指定 1 名指导教师，每名学生限参加 1 支参赛队，各高校参赛队数不限，不可跨校组队。

3.报名步骤：

（1）各高校收到通知后，确定一位老师作为本校唯一联络老师，并填写“高校联络教师登记表”。盖章，扫描后于 2026 年 6 月 10 日前将电子版和盖章纸质版扫描件发至组委会秘书处邮箱 xieyi233@sjtu.edu.cn。

（2）学生收到通知后，联系本校负责老师，进行集体报名。本次竞赛不接收个人报名。

（3）报名时间为 2026 年 6 月 1 日—7 月 1 日。

4.在线提交作品：2026 年 7 月 2 日—8 月 2 日。高校联络教师将本校所有作品上传至 https://ai-contest.sjtu.edu.cn/。

5.参赛名单公布日期：2026 年 8 月 3 日。

6.联络信息：

作品赛联络群：281820424。

高校教师联络群：474997879。

二、参赛作品

（一）赛道说明

1.定向式专项命题作品赛是 2026 第二届大学生人工智能安全竞赛作品赛的重要组成部分，聚焦人工智能安全实战应用方向，围绕具体技术场景设置专业命题，重点考察参赛队伍在指定任务下开展方案设计、系统实现、测试验证、工程交付和文档撰写的综合能力。

2.本赛道共设置 5 道定向式专项命题，覆盖大模型内容风控、越狱测试、Web 攻防、异常行为识别、DNS 隧道检测等技术领域。参赛队须从本指南给出的定向式专项命题中选择 1 道作为参赛题目，并围绕所选命题完成作品设计与实现。

3.本赛道不接受参赛队另行自拟题目。参赛作品应紧扣所选命题的题目背景、题目描述、挑战内容、考察能力、交付物和评价标准要求。参赛队可以在命题范围内进行功能拓展、技术优化和创新设计，但不得偏离所选命题的核心任务和技术方向。

（二）参赛作品要求

1.参赛作品应体现一定的创新性、实用性和工程完整性，能够针对所选定向命题提出清晰的技术方案，并形成可运行、可演示、可验证的作品成果。

2.参赛作品应按照所选定向命题要求完成系统设计、功能实现、测试验证和文档撰写。各赛题对源代码、程序文件、模型文件、数据集、配置文件、说明文档、测试报告等交付材料有具体要求的，以对应赛题“交付物”部分为准。

3.参赛作品应具备明确的应用场景、完整的功能流程、可复现的测试过程和较为规范的代码及文档材料。参赛队应能够在决赛现场对作品的核心功能、运行流程、测试结果和创新特色进行演示和说明。

4.本赛道强调人工智能安全技术的合规研究与防御应用。参赛作品不得用于未授权攻击、恶意利用、违法测试或破坏性活动；涉及安全测试、越狱评估、Web 攻击载荷检测、异常行为识别、DNS 隧道检测等内容的，应限定在合法、授权、可控的实验环境中完成。

5.参赛作品应为参赛队员独立设计、开发完成的原创性作品，严禁抄袭、剽窃、一稿多投等行为。凡发现此类行为，将取消参赛队伍的参赛资格，并追究相关指导教师和高校的责任。

6.凡已公开发布并已获得商业价值的产品不得参赛；凡有知识产权纠纷的作品不得参赛；与企业合作即将对外发布的产品不

得参赛；凡在其他公开竞赛，不含校内比赛，中获奖的作品不得参赛。

7.本次竞赛不支持论文参赛。参赛队应提交可运行、可演示、可验证的作品成果，不能仅以论文、方案设想或概念设计作为参赛作品。

（三）定向式专项命题

参赛队应在以下定向式专项命题中选择一个题目完成作品设计与实现。各赛题的具体题目背景、题目描述、挑战内容、考察能力、交付物和评价标准详见文末“定向式专项命题详情”。

序号	定向式专项命题
题目 1	面向对话场景的大模型输入/输出违规内容过滤系统
题目 2	针对混淆逃逸的 Web 攻击载荷动态检测与对抗方案
题目 3	基于 Transformer 架构的异常行为智能检测
题目 4	大模型越狱提示词自动化生成与安全测试工具
题目 5	基于流量特征的 DNS 隐蔽隧道检测

三、初赛

1.2026 第二届大学生人工智能安全竞赛分初赛和决赛。凡取得参赛资格的参赛队均自动进入初赛。

2.初赛作品提交截止时间为 2026 年 8 月 2 日。各参赛队应在 2026 年 7 月 2 日—8 月 2 日期间完成参赛作品并网上提交，以参加初赛。

3. 各参赛队应在截止时间前提交所选定向命题对应的参赛材料。提交材料包括作品报告、作品原创性声明，以及所选赛题“交付物”部分要求的源代码、可运行程序、模型文件、数据集、配置文件等相关材料。具体提交内容以本指南中各赛题“交付物”部分的要求为准。

4. 作品报告和作品原创性声明应使用竞赛官网发布的统一模板填写。其中，作品报告应结合所选定向命题，完整说明作品的系统设计方案、功能实现、测试方案、测试结果、创新特色及其他必要内容；作品原创性声明须由所有参赛队员手写签名，并按要求加盖学校教务处或教务部公章。

5. 提交方式：各参赛队将参赛作品及相关材料交由高校联络教师，由高校联络教师统一通过竞赛官网提交。

6. 本次竞赛的组委会将在全国范围内组织专家对参赛队伍提交的定向命题作品进行网络评审，初赛评审时间为 2026 年 8 月 5 日—8 月 15 日。依据网络评审结果，由专家组评审并最终确定进入决赛名单。进入决赛的参赛队伍由专家组根据参赛队伍总数及参赛作品质量确定。

7. 评审方式：评审专家审阅参赛作品材料，结合所选定向式专项命题的题目要求、交付物要求和评价标准，对参赛作品进行综合评审，并给出评审意见。每一件作品将至少由 2 至 3 位专家进行评审。

8. 专家评审以各定向式专项命题中对应的“评价标准”为主要依据，同时综合考察参赛作品与所选命题的契合度、作品完成

度、系统可运行性、测试结果有效性、工程实现质量、创新性、应用价值以及文档规范性等。

四、决赛

1.组委会将在2026年8月16日公示进入作品赛决赛的名单。

2.在获得决赛资格后，各参赛队伍可以继续对参赛作品进行完善和修改。

3.线下决赛会评安排为2026年8月20日报到，8月21—22日决赛。获得决赛资格的参赛队伍应在规定时间内参加决赛。决赛分为作品演示和答辩两个环节。

4.作品演示

参赛队自行携带作品、文档及设备，到决赛地点进行演示。决赛时，承办方提供因特网接入环境。各参赛队伍须自带测试设备，如对作品的演示环境有特殊要求，请提前与组委会秘书处协商(注意：本次竞赛演示时，原则上不能以视频方式演示；只能现场操作演示。如有特殊演示要求，需要提前申请并获得组委会秘书处同意)。

5.答辩

答辩时间为15分钟（10分钟讲解，5分钟提问），包括PPT陈述、演示、测试与专家提问，专家会现场检查源代码。

6.评审专家对每个竞赛作品实行分项打分，集体讨论，综合评定，最终确定参赛作品的获奖等级。

7.决赛时，由竞赛承办方统一提供附近宾馆、饭店等相关信

息，食宿及相关费用由参赛学校自理。

五、获奖

1.本届竞赛设一等奖、二等奖和三等奖。获奖比例由竞赛专家委员会开会决定。

2.竞赛颁发统一的获奖证书，所有获奖队伍及名单将以多种方式公布，并报送相关高校，作为高校评定奖学金、推荐研究生等的参考。

3.获奖队伍将获邀参加 2026 年 8 月 23 日进行的“2026 第二届大学生人工智能安全竞赛颁奖”。

六、指导教师

1.指导教师必须是参赛队伍所在高校在职教师。

2.指导教师可以指导学生理解定向命题要求、开展题目选择和设计方案论证，但具体的软件编程、系统调试、作品文档撰写必须由参赛学生独立完成。

3.指导教师负责把握所指导学生参赛作品的原创性，并确保其不具攻击性，以及不与国家法律、法规相违背。

七、参赛高校

1.各高校在收到参赛通知后，指定 1 位教师作为联络人（联络人须为高校领队），负责本校竞赛相关事宜，并在竞赛网站上下载“高校联络教师登记表”，将该教师信息填写完后，发至组委会秘书处邮箱：xieyi233@sjtu.edu.cn（含电子版和盖鲜章纸质版的扫描件）。

2.各高校负责本校范围内的竞赛组织、选拔等工作，并对本校范围内参赛队伍及指导教师的真实性负责。

3.本次竞赛将对在竞赛组织工作中表现出色和作出贡献的高校给予奖励。

4.各高校应从培养和选拔创新人才的角度出发，对获奖学生在奖学金评定等方面予以优先考虑。

5.禁止参赛高校弄虚作假。对违反国家有关法律、法规以及大赛章程的行为，组委会将取消相关奖项，并依照有关规定进行处罚。

定向式专项命题详情

题目 1：面向对话场景的大模型输入/输出违规内容过滤系统

【题目背景】

大语言模型在聊天、问答、客服等场景广泛使用，用户输入或模型输出常会出现低俗、暴力、广告、谣言等违规内容，带来使用风险。工业界入门级防护方案以关键词匹配、语义规则、轻量文本分类为主，技术难度适中、落地性强。本题目聚焦基础内容安全防护，基于开源大模型接口，搭建一套轻量化内容过滤系统，实现输入拦截、输出修正、日志记录等基础防护能力。

【题目描述】

对接开源轻量大模型（如 Qwen-Tiny、ChatGLM 精简版等）API，实现用户输入检测、模型输出校验、违规内容过滤、基础统计四大基础功能，要求如下：

1. 违规词库与规则构建：自行整理常见违规词汇、短句、正则规则，分为色情、暴力、广告、敏感话术四类；支持手动添加、删除、导入违规词。

2. 双层过滤逻辑：

第一层：关键词+正则匹配，快速拦截明显违规内容；

第二层：调用开源预训练文本分类模型，对疑似内容做二次语义判断。

3. 分级处理逻辑：

高违规内容：直接拦截，拒绝向大模型转发，并提示用户。

低风险疑似内容：自动脱敏替换违规词汇，再送入大模型。

正常内容：正常请求大模型并返回结果。

4. 模型输出二次校验：对大模型返回的回答再次做内容检测，若存在违规内容，屏蔽并替换为标准合规话术。

5. 基础辅助功能：记录每条请求的检测结果、处理方式、时间；简单统计每日违规请求数量、违规类型占比。

【挑战内容】

1. 合理设计词库与正则规则，减少正常文本的误拦截。
2. 串联规则检测与预训练模型检测，保证整体运行效率。
3. 实现违规词汇智能替换，尽量不破坏原有语句语义。
4. 完成前后全链路打通，保证系统稳定运行。

【考察能力】

1. 基础编程、模块化代码设计能力。
2. 正则表达式、文本处理、字符串操作能力。
3. 开源模型/API 调用、第三方库使用能力。
4. 基础逻辑设计、流程调度与简单数据统计能力。
5. 基础软件项目开发、测试与排错能力。

【交付物】

1. 完整可运行源码，代码注释清晰，模块划分明确。
2. 违规词库文件、正则规则配置文件。
3. 系统使用说明文档。
4. 项目实训报告（含功能设计、流程说明、测试结果、问

题总结）。

5. 功能测试用例与测试截图。

【评价标准】

1. 功能完整性（40%）：完成词库管理、双层检测、分级过滤、输出校验、日志统计全部功能，缺失一项酌情扣分。

2. 过滤效果（25%）：明显违规内容拦截率≥90%，正常文本误判率控制在 5%以内。

3. 代码质量（20%）：代码结构清晰、命名规范、注释完整，可直接运行，无明显 Bug。

4. 文档与测试（15%）：文档描述清楚，测试用例齐全，结果记录完整。

题目 2：针对混淆逃逸的 Web 攻击载荷动态检测与对抗方案

【题目背景】

传统的 Web 应用防火墙（WAF）和基于规则的攻击检测引擎严重依赖已知的攻击特征库。攻击者通过多种混淆技术（如编码混淆、SQL/命令语句变形、恶意脚本模糊处理等）对攻击载荷（如 SQL 注入、XSS、命令执行字符串）进行变换，旨在绕过静态特征匹配。尽管基于机器学习的检测模型具备一定的泛化能力，但其在面临高强度、动态变化的混淆对抗时，鲁棒性仍面临挑战。本课题要求参赛者设计一种能有效应对混淆逃逸的 Web 攻击动态检测方案，提升检测系统在对抗环境下的稳定性和准确性。

【题目描述】

设计并实现一套检测与对抗系统：

1.载荷净化与特征提取：设计预处理模块,对输入的HTTP 请求参数（URL, Body, Headers）进行深度解析，尝试对常见的混淆手段（如多层编码、HTML 实体化、字符串分割与拼接等）进行归一化还原，并从中提取更有判别力的静态与动态特征。

2.对抗性检测模型构建：构建一个分类模型（可使用传统机器学习或深度学习），该模型应针对混淆攻击的特点进行优化。

3.混淆对抗模拟与评估：实现一个简单的混淆载荷生成器，能够对已知攻击样本进行自动化变种（如随机插入无效字符、变换编码、使用等价函数/语句），用于测试和增强所构建检测模型的鲁棒性。

性能与效果约束：

1.对单次HTTP 请求的检测耗时≤10 毫秒。

2.在包含基础混淆的测试集上，模型应保持高检出率。

【挑战内容】

高鲁棒性检测：专注于提升模型对未知、复杂混淆手法的泛化检测能力。

【考察能力】

1.Web 安全与攻防知识：深入理解常见Web 攻击原理及其混淆绕过技术。

2.AI 模型鲁棒性设计：掌握对抗样本、数据增强、特征工

程等提升模型泛化与鲁棒性的方法。

3.系统化工程思维：从攻击模拟、数据预处理、模型训练到性能评估的全流程实现能力。

【交付物】

1.系统完整源代码。

2.详细设计报告。

3.测试与验证报告。

4.可运行的演示程序。

【评价标准】

1.检测效果（40%）：在高强度混淆测试集上的漏报率和误报率的综合表现。

2.方案创新性与完整性（30%）：在载荷净化、特征工程、模型鲁棒性设计或对抗模拟方面的创新性；系统各模块设计的完整性与合理性。

3.性能与实用性（20%）：单请求检测耗时是否满足约束；系统是否易于集成到实际检测流程中。

4.文档与代码质量（10%）：报告逻辑清晰，实验充分；代码结构良好。

题目 3：基于 Transformer 架构的异常行为智能检测

【题目背景】

用户行为模式复杂多变，传统检测手段在应对行为模式漂移

和未知异常场景时，误报率高、适应性差，且告警缺乏解释性，安全运维人员难以高效决策。Transformer 架构具备对长序列行为数据进行上下文建模的能力，适合捕捉行为数据中的复杂依赖关系与微妙偏差。本赛题允许参赛者自由选择 Transformer 变体（从轻量编码器到大规模预训练模型），探索其在异常行为检测中的多样化方案。

【题目描述】

给定一个月的主机行为日志数据集（包含进程、网络、文件、注册表等事件，以正常活动为主，混杂少量各类异常行为），参赛者需设计并实现一套以 Transformer 架构为核心的异常行为检测系统，完成以下任务：

1. **异常行为检测：**利用 Transformer 模型对行为序列进行建模，从数据中自动发现偏离常规的异常行为模式，输出异常告警；

2. **可解释性输出：**每个告警需附带自然语言解释或关键特征标注，说明哪些行为片段或特征触发了异常判定，并提供原始日志作为证据。

【挑战内容】

可选挑战方向（选一即可）

方向 1-高精度检测：在误报率≤20% 的前提下，最大化异常行为检出覆盖率。

方向 2-轻量化部署：在保持检出率≥80% 的条件下，通过

模型压缩、量化或知识蒸馏等手段，使系统可在无GPU 设备上稳定运行，并最小化资源占用。

【考察能力】

1. Transformer 建模能力：自注意力机制在行为序列上的应用、序列编码方案设计、模型规模与性能的权衡（允许使用小型 Transformer 或大规模预训练模型）；

2. 异常行为分析能力：异常行为定义、非均衡数据处理、误报控制、未知威胁泛化；

3. 可解释性设计：将 Transformer 的注意力权重或隐层特征转化为可读的解释信息；

4. 系统工程化：模型轻量化与部署、模块化可扩展设计。

【交付物】

1. 系统完整源代码、依赖说明及一键部署脚本；

2. 设计说明书（含 Transformer 选型与理由、数据预处理、模型结构、可解释性实现）；

3. 检测分析报告（告警列表、可解释性示例、性能指标截图）。

【评价标准】

1、检测有效性（50%）：异常行为检出率、误报率；需提供日志证据佐证。

2、模型应用有效性（20%）：Transformer 模型选型与任务匹配度、推理效率、输出可靠性。

3、性能指标（15%）：批量分析耗时、在线延迟、资源占用是否满足约束。

4、可解释性与创新性（15%）：告警解释的准确性与可读性、技术创新点、文档质量。

及格线参考：

1.对数据集中标注的异常行为检出率≥70%;

2.误报率≤20%;

3.系统可稳定完成全量日志分析。

题目 4：大模型越狱提示词自动化生成与安全测试工具

【题目背景】

评估大模型安全性需要大量越狱测试用例。人工构造越狱提示词效率低、覆盖有限。通过模板化策略和自动化变异技术，可以快速生成多样化的越狱提示词，用于测试模型安全边界。本项目技术难度适中，聚焦文本变异和批量测试，落地性强。

【题目描述】

设计并实现一个越狱提示词自动化生成与安全测试工具：

1. **越狱模板库：**整理至少 5 种越狱策略模板（角色扮演、约束绕过、上下文混淆、分步诱导、翻译伪装等）；

2. **自动化变异引擎：**基于模板进行同义词替换、句式重组、多语言混合、符号插入等变异，批量生成多样化越狱提示词；

3. **批量测试模块：**对接开源大模型 API（如 Qwen、ChatGLM

等），自动发送生成的提示词并收集模型响应；

4. 结果评估与报告：根据响应内容判断是否越狱成功，统计各类策略成功率，生成测试报告。

建议输入输出：

输入为模板文件和变异配置；

输出包含 generated_prompts、response_texts、jailbreak_success、success_rate_by_strategy 等字段。

【挑战内容】

1. 自动化生成的越狱提示词中，至少 30%能在测试模型上触发非预期响应；

2. 变异后的提示词保持语义通顺，避免生成无意义乱码；

3. 控制 API 调用成本，批量测试时支持并发和速率限制；

4. 建立合理的越狱成功判定标准（如关键词匹配+响应语义分析）。

【考察能力】

1. Prompt 工程：理解越狱攻击原理、模板设计、策略分类；

2. 文本变异技术：同义词替换、句式变换、多语言处理；

3. API 调用与并发：Python requests、异步调用、速率控制；

4. 数据分析：成功率统计、策略效果对比、结果可视化。

【交付物】

1. 完整可运行源码，模块划分清晰；

2. 越狱模板库文件（至少 5 种策略，每种策略含 3 个以上基础模板）；

3. 测试报告（含各类策略成功率、典型案例、失败案例分析）；

4. 系统使用说明文档(含配置方式、测试流程、结果解读)；

5. 生成的越狱提示词样本集（不少于 100 条）。

【评价标准】

1. 生成有效性（35%）：越狱提示词的成功率、策略覆盖度；

2. 变异质量（20%）：生成文本的通顺度、多样性、与模板的差异性；

3. 工程实现（20%）：代码质量、并发效率、错误处理、文档完整性；

4. 结果分析（15%）：测试报告的深度、策略对比分析、可视化呈现；

5. 创新性（10%）：是否有独特的变异策略或判定方法。

题目 5：基于流量特征的 DNS 隐蔽隧道检测

【题目背景】

DNS 隐蔽隧道是常见的数据外泄和 C2 通信手段，攻击者利用 DNS 协议将数据封装在 DNS 查询和响应中传输，绕过传统防火墙检测。DNS 隧道在流量特征上与正常 DNS 存在明显差异（如请求频率异常、域名熵高、Payload 过大等），适合基于机器学

习的检测方案。本项目聚焦单一协议检测，范围明确，数据易获取。

【题目描述】

设计并实现一个基于流量特征的 DNS 隐蔽隧道检测原型系统：

1. DNS 流量解析：从 PCAP 文件或 DNS 日志中提取查询记录，解析域名、查询类型、响应码、Payload 长度等基础信息；

2. 特征提取：构建不少于 10 维检测特征，包括请求频率（单位时间查询数）、域名熵值、子域名长度、TXT/NULL 记录比例、请求-响应大小比、CNAME 链长度、知名 DNS 服务器占比等；

3. 检测模型：使用机器学习分类器（如 Random Forest、XGBoost、孤立森林）或轻量神经网络进行二分类（正常 DNS vs 隐蔽隧道）；

4. 告警输出：对检测到的可疑隧道输出告警，包含风险等级、关键特征值和建议处置动作。

建议输入输出：

输入为 PCAP 文件或 CSV 格式 DNS 日志；

输出包含 is_tunnel、confidence、risk_level、key_features、domain_list 等字段。

【挑战内容】

1. 在测试集上检测率≥85%，误报率≤10%；

2. 能够识别至少 2 种常见 DNS 隧道工具特征（如 iodine、dns2tcp、dnscat2）；

3. 单文件/日志分析耗时≤5 秒（支持 10000 条 DNS 记录）；

4. 对正常高频率 DNS 场景（如 CDN、动态 DNS）具有较好的区分能力。

【考察能力】

1. DNS 协议基础：查询类型、响应格式、常见记录类型；

2. 流量分析：PCAP 解析（Scapy/Tshark）、日志处理、统计特征计算；

3. 机器学习：分类模型训练、类别不平衡处理、模型评估；

4. Python 工程：数据批处理、模块化设计、结果可视化。

【交付物】

1. 完整可运行源码及依赖安装说明；

2. 预训练模型文件和训练脚本；

3. 测试数据集（含正常 DNS 流量和至少 2 种 DNS 隧道工具产生的流量）；

4. 特征说明文档（含特征列表、计算公式、检测原理）；

5. 测试报告（含检测率、误报率、各工具识别效果、处理耗时、资源占用）。

【评价标准】

1. 检测效果（40%）：检测率、误报率、F1 值、各工具识别效果；

2. 特征设计（20%）：特征的合理性、覆盖度、对检测效果的贡献；

3. 模型效率（15%）：处理速度、内存占用、是否满足准实时要求；

4. 可解释性（15%）：是否提供关键特征值、误判案例说明；

5. 工程完整度（10%）：代码质量、文档、可复现性、易用性。