

## 1. 项目背景与目标

飞花令是一种古老的中国传统诗词接龙游戏，参与者需要根据指定的字或词，迅速作答，并且这些答复通常需要符合一定的诗词规则或文化背景。在现代，飞花令不仅是传统文化的传承，也成为了许多文化活动中的娱乐方式。为了丰富飞花令的互动体验，并让用户在玩乐的同时能够更深入地了解中国古典诗词，本项目计划开发一款基于诗词大模型的飞花令问答游戏。

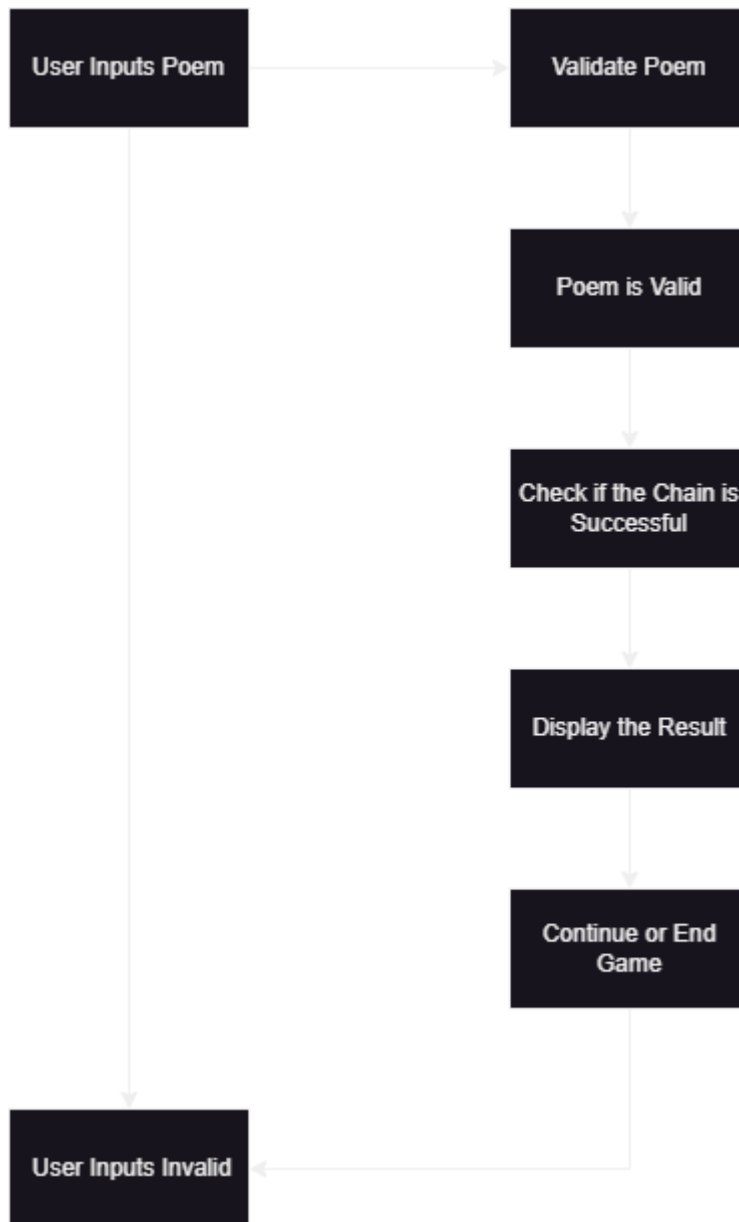
该游戏将结合自然语言处理技术和古诗词大数据，为玩家提供更智能、更趣味性的飞花令体验，借此激发玩家对诗词和文化的兴趣。具体目标如下：

- 提供一个包含诗词知识的大型语言模型，能够生成符合诗词接龙规则的答案。
- 设计飞花令游戏机制，包括游戏规则、题目生成与判断系统等。
- 为玩家提供诗词解析、历史背景等附加功能，帮助玩家更好地理解诗词内容。

## 2. 功能需求

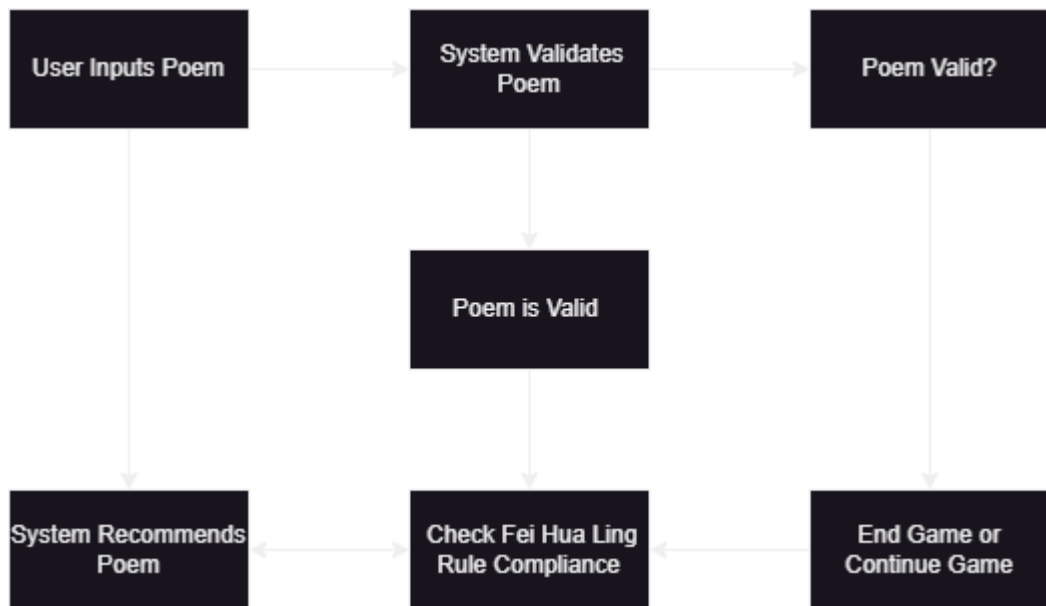
### 2.1 诗词生成与判断

- 输入解析：**根据玩家给出的字或词，系统能即时解析并生成符合飞花令规则的诗句。
- 诗词匹配：**确保系统生成的诗句符合古诗词接龙的规则，如押韵、字词搭配等。
- 诗词库：**使用经过筛选和分类的古诗词库。
- 自动判断：**根据玩家的输入，自动判断其是否符合游戏规则并给出反馈。若输入无效（不符合字词搭配、格式错误等），系统应提示错误。

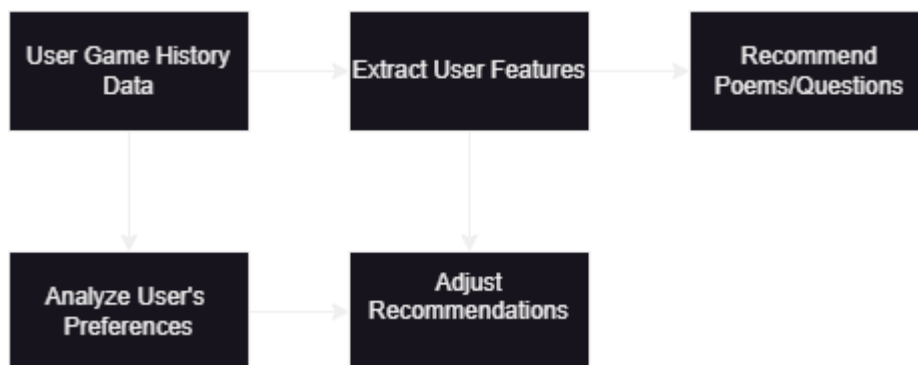


## 2.2 游戏模式

- **飞花令模式：**玩家可以与系统对战，系统自动生成题目并判断玩家答案的正确性。



- **诗词解析：**为玩家提供每首诗词的详细解析，包括诗词背景、作者简介、历史文化背景等。
- **互动学习：**通过游戏化的方式，玩家可以在答题过程中逐渐加深对诗词的理解和记忆。



## 2.3 用户交互与反馈

- **即时反馈：**每次玩家提交答案后，系统应给出正确性判断及反馈，同时给出是否通过的提示。

## 2.4 诗词大模型集成

- **诗词生成模型：**集成经过数据训练的诗词大模型，支持从古典诗词库中自动生成符合规则的诗句。
- **自然语言理解与生成：**模型能够理解玩家的输入，并生成合理的诗句响应，符合飞花令的要求。

- **动态更新与优化：**定期更新诗词库和模型，加入新的诗词、名句等内容，不断提升生成诗句的质量和多样性。

## 3. 技术需求

### 3.1 系统架构

- **前端：**采用现代 Web 或移动应用框架，如 React、Vue、Flutter 等。
- **后端：**使用 Python 语言实现，利用 Flask/Django 或 Spring Boot 等框架搭建。
- **诗词大模型：**可以使用星火、Chat-GPT 或其他大语言模型的 API，结合中文自然语言处理技术进行诗词生成。

### 3.2 模型需求

- **训练数据：**大量的诗词文本数据，涵盖各个历史时期的诗词，如唐诗、宋词、元曲等。数据应进行标准化和清洗，去除无关内容。
- **诗词规则：**模型应能够学习并遵守诗词的基本格式和韵律要求，能够生成符合押韵、对仗、平仄等传统诗词规则的句子。
- **自然语言处理能力：**模型需要具备较强的中文自然语言理解能力，能够分析和生成符合语境和历史背景的诗句。

## 4. 用户需求

- **易于上手：**游戏应设计为简单易懂的玩法，适合不同年龄层和文化背景的用户，尤其是对传统文化感兴趣的用户。
- **互动性强：**通过后续增加更多玩法，增加软件的互动性和趣味性。
- **学习与娱乐结合：**游戏不仅仅是娱乐工具，也希望能够通过诗词学习提升自己的文学素养。

## 5. 非功能需求

### 5.1 性能需求

- **响应时间：**系统应保证快速响应，用户输入后，系统在 1-3 秒内提供反馈。
- **可扩展性：**系统架构应支持快速扩展，方便后续可以实现更多功能。
- **高可用性：**系统应具备高可用性，防止因故障导致游戏中断等问题。

## 5.2 安全需求

- **数据保护：**确保用户数据的安全性，符合相关法律法规（如《个人信息保护法》）的要求。

## 5.3 用户体验需求

- **界面友好：**UI 设计应简洁、直观、易于操作，适配不同设备的屏幕尺寸。
- **流畅体验：**游戏的加载和运行应保持流畅，减少卡顿和延迟。

# 6. 发展与未来规划

- **丰富的诗词内容：**随着时间推移，计划逐步丰富诗词库，加入更多的古诗词、现代诗歌等，满足不同文化背景玩家的需求。
- **AI 学习与优化：**随着玩家的使用，系统应不断学习，优化诗词生成算法，提高生成质量和多样性。