

Model Valhalla 需求调研

姚润茂 方钧同 钟健坤 李科霖

目录

- 1. 项目背景与项目目标2
- 2. 典型用户与用户痛点分析2
 - 2.1 模型提供者2
 - 2.2 数据集提供者2
 - 2.3 网络用户3
- 3. 需求分析3
 - 3.1 调研方法与过程3
 - 3.2 用户类型3
 - 3.3 功能需求3
 - 3.4 设计约束需求 10
 - 3.5 性能需求 10
- 4. 用户故事地图与用户故事卡 11
 - 4.1 用户故事地图与故事卡 11
 - 4.2 功能点优先级与难度分析 11
- 5. 产品原型 13
- 6. 总体架构与技术选型 17
- 7. 开发迭代与分工计划' 18
 - 7.1 开发迭代计划 18
 - 7.2 分工 18

1. 项目背景与项目目标

大语言模型是近年来人工智能领域的热门话题，它们可以在多种任务上表现出惊人的效果，如文本生成、阅读理解、对话、知识问答等。然而，针对大语言模型的测试平台目前仍有诸多缺陷：

- 测评标准与数据集的局限性。目前的测评平台标准不一，且各平台的测试数据集水平不一，造成各平台测试结果难以评估，给用户带来困扰。
- 测评平台的开放程度不同。目前很多测评平台都只能基于固定的测试数据集对指定的模型产生结果，可拓展性差的同时无法支持新生的模型/数据集，不利于生态的开放与进步。

综合考虑到上面所述的问题，我们的项目目标是为各种大语言模型提供一个开放、公平、可靠的评测机会，为人们直观展示模型水平与排行，从而进一步促进模型的发展和创新。

2. 典型用户与用户痛点分析

本产品面向模型提供者、数据集提供者乃至更一般的网络用户提供服务，模型与数据集提供者通过将模型连接至测试平台，或上传数据集的方式与本平台建立连接，进而得到不同数据集产生的结果与对模型的评价，而网络用户也可以通过排行榜了解到大模型的发展现状。典型用户的痛点如下。

2.1 模型提供者

- 缺乏客观的评价标准和数据集，评价标准不公开，导致大语言模型的性能和安全性难以量化和比较，对众多模型测试平台感到“不放心”。
- 缺乏有效的评测工具，大语言模型测试过程复杂，拖慢模型的进一步研发。
- 过于简单的评测结果与反馈，没有进行模型间的水平比较，影响模型的真实水平评估。

2.2 数据集提供者

- 缺乏支持上传数据集的大语言模型测试平台，有力无处使。
- 无法看到提供的数据集在各模型上的直接评测结果，难以进行自身数据集的勘误。

2.3 网络用户

- 类似于模型提供者，不够开放的数据集与测试标准让诸多测试平台可信度不足。
- 没有足够的模型测试规模，现有平台上能看到的都是“老面孔”，无法真正了解到一般大语言模型的情况、差距与进步。

3. 需求分析

3.1 调研方法与过程

1. 竞品分析

对市面上现有的产品（如 Chatbot Arena）进行分析与批判。标准难统一、大模型需要测评的能力边界难以确定，导致没有形成统一的测评标准。

以 Chatbot Arena 为例，其并没有明确的数据集管理功能，也不支持用户自定义的数据集，这可能是由于其面向的用户群体多为具备一定知识的相关行业从业者。

2. 用户访谈

与助教团队沟通得到甲方的第一手需求。数据集方面希望能拓展数据集多样性，支持用户自定义数据集，测评方面则需要支持客观题自动评测，主观题可以采用人工评分、对抗评测、利用 GPT4 评分等方式进行测评。

3.2 用户类型

本项目用户分为两类：测试者（即普通用户）与管理员。

3.3 功能需求

3.3.1 用户身份验证

需求：用户在登录以及注册页面可以完成 用户注册、用户登录 两项操作，在个人中心可以完成 查看个人信息、修改个人信息、修改安全信息 三项操作。

管理员可以执行 管理员注册、管理员登录 两项操作。

1. 用户注册

用户点击注册按钮，需要先填写手机号和密码，之后点击“获取验证码”，输入手机上收到的验证码，验证码每 1 分钟可以发送一次。完成验证后勾选左下角的“已阅读并同意《服务协议》和《隐私政策》”，点击“注册”按钮，系统会提示注册成功，并返回到登录页面。考虑到对用户数据和隐私的保护，系统会建议用户使用强密码，并且要求用户填写电子邮箱地址以找回密码。

2. 用户登录

登录应考虑多因素身份验证，提供多种登录方式。用户可以选择输入手机号与密码验证身份，也可以通过进行手机验证码验证，两种测试方式可以通过点击窗口上的相应选项进行切换。无论采取哪种方式，填写表单项后用户点击“登录”，如果信息正确将进入到测试主页。

3. 查看个人信息

用户注册后信息可以随时查看自己的信息，点击“个人中心”会弹出下拉框，点击“个人信息”将跳转到个人信息页，该页面将展示用户名、用户类型等基本信息。此外，还可以看到用户关注的模型、数据集以及用户测试记录。用户通过点击个人页面的部分选项也能直接跳转至相关页面，如页面默认展示部分测试记录，用户可以点击“查看更多”选项跳转至测试记录页面查看完整记录。

4. 修改个人信息

在“用户设置”页面，用户能看到除部分安全信息外的所有各项信息，并可以直接修改。初始部分信息可能不存在，用户可以在各信息标签处通过点击或输入等方式添加或修改头像、用户名，也可以在下方的“基础信息”选项修改国家、个人简介等信息，修改完成后点击保存即可。

5. 修改安全信息

用户选择“安全设置”选项会展示修改安全信息界面，其中展示用户的私密信息如绑定手机号、绑定邮箱等，手机号中部分会以“*”遮挡，保护用户隐私。在该页面修改私密信息如密码时，将需要输入原本密码以验证身份，或是点击“忘记密码”按钮跳转至验证码重置密码界面，正确输入验证码后方可设置新密码。

以下为管理员可以额外执行的操作。

6. 管理员注册

除了普通用户外，还要有少量的管理员用户。在登录页面点击“我是管理员”进入管理员登录界面，点击“注册账号”，除了填写基本信息外，还需要填写由开发人员提供的邀请码，信息正确后，点击“注册”，系统会弹出注册成功并返回管理员登录页面。

7. 管理员登录

管理员登录的过程类似用户登录的过程，首先选择使用账密登录或是手机验证码登录的方式，在页面上的表单中输入正确的管理员账号信息后，点击“登录”即可进入主页。平台会判断该账号为管理员账号，并在浏览过程中解锁管理员权限，能执行各种管理员操作。

3.3.2 数据集维护

需求：用户在数据集页面可以完成 查看数据集列表、查看详细信息、数据集筛选、数据集下载、数据集上传、数据集反馈、数据集举报 七种操作。

管理员可以执行 数据集修改、数据集删除、反馈与举报处理 三种操作。

1. 查看数据集列表

用户可以在页面上直观地看到现有数据集的列表以及各数据集的基本信息，如数据集编号、名称、创建时间等。数据集列表支持按词条排序的功能，通过点击数据集列表上方的词条可以切换按该词条排序/逆序展示数据集列表，下方列表将随之更新。

2. 查看详细信息

用户点击某一数据集的“查看”选项将会展开页面显示该数据集的更详细的信息，该详细信息分为若干栏，包含详细资料栏展示如发布方、应用领域等信息、测试表现栏展示该数据集在平台上进行测试的表现等。用户可以通过点击展开页面上方的标签切换显示内容。

3. 数据集筛选

页面上方有针对数据集列表的筛选框，用户可以在筛选框中按不同条件筛选出想要的数据集，如按编号、名称等关键词条进行检索，或是选取创建时间区间等操作。选择完成后点击“检索”按钮，下方数据集列表也会随之更新。

4. 数据集下载

用户可以通过点击“下载”按钮下载指定数据集，也可以通过勾选数据集并点击“批量下载”按钮下载多个数据集。

5. 数据集上传

用户可以通过点击“上传”按钮展开上传数据集界面，填写基本信息、相关论文（可选）、标签、许可后即可生成数据集上传申请，待管理员通过后即可将自己提供的数据集上传至平台。管理员会对上传的数据集的合法性进行检验，如果发现上传的数据集有误会弹出提示。合法的数据集将被存储在平台上，能够参与平台上的模型的能力测试并出现在排行榜上。

6. 数据集反馈

用户点击某一数据集的“反馈”选项将会弹出数据集反馈界面，用户可以从反馈类型标签中点击并选择自己本次反馈的类型，如数据错误、存在更新版本等，并在下方的输入框中写下自己具体的想法，从而对该数据集提出自己的意见，也可以通过“上传文件”按钮附上相关文件。该反馈信息可以被管理员查看，并可以得到管理员的处理。

7. 数据集举报

用户发现某一数据集出现异常或具有很严重的问题时，可以点击“反馈”选项之后选择上方的“举报”选择框进入举报界面，在该界面中选择举报理由以及详细描述，随后举报也将发送至管理员进行处理，且管理员应当优先进行举报的处理。

以下为管理员可以额外执行的操作。

8. 数据集修改

管理员可以针对数据集的属性进行修改，在管理员视角可以选择“修改”选项，之后进入修改页面，可以对数据集的大部分属性进行勘误，如纠正来自数据集提供者的错误等。

9. 数据集删除

管理员发现不合格/错误的数据集后可以删除数据集，在管理员视角选择“删除”选项，会弹窗提示是否确认删除，防止管理员误操作错误删除数据集。如若确实需要删除，选择“确定”即可。

10. 反馈与举报处理

管理员选中某一数据集后可以看到该数据集收到的反馈和举报的情况，其中举报将比反馈以更高优先级显示。管理员可以点击“处理”进入反馈处理界面，可以看到该数据集收到的反馈以及举报信息列表，并可以点击“确认”选项确认某一反馈或举报已经得到处理。之后管理员可以根据获取到的信息进行如上的修改或删除操作。

3.3.3. 排行榜功能

需求：用户在排行榜界面可以完成 查看排行榜表格、查看模型排名、对抗评分支持、模型详细信息、排行榜下载 五种操作。

管理员可以执行 指定单元格修改 一种操作。

1. 查看排行榜表格

页面上的排行榜被组织为表格形式，每一行对应一个模型，大部分的列则对应一个数据集。用户可以直接在排行榜页面的表格中看到各模型在各数据集下测试得到的分数。此外，排行榜表格的列中也存在其余信息项如名称等，用户均可以进行查看。

2. 查看模型排名

排行榜可以选择按多种标准对模型水平进行排名并按序显示，标准可能是简单的按名称排序，按某一数据集上的得分排序，或是按所有数据集上的得分平均值排

序，抑或是选定的几个数据集上的平均值。通过点击排行榜上方的相应词条即可选择按该词条顺序/逆序排列，点击后排行榜将实时更新。

3. 对抗测评支持

平台会记录对抗测评的结果并用于计算模型的 Elo 值，后者表示一个模型的相对水平，分值越高水平越高。Elo 值也是可选的模型排序选项，在上述的排行榜表格中能直接显示。此外，Elo 值也会在对抗测评中用于选择对抗对手与计算新值等。

4. 模型详细信息

用户可以点击某一模型的“查看”选项弹出模型详情界面展示详细信息，该详细信息页面类似数据集详细信息页面，同样分为若干栏，如详细资料栏展示发布方、应用领域等信息、测试表现栏展示该模型在平台上的测试表现等。用户可以通过点击展开页面上方的标签切换显示内容。

5. 排行榜下载

用户可以点击“下载”按钮对排行榜进行下载，下载可以将排行榜保存至用户本地，方便随时查看，并能应用已有的样式调整。表格能够以.csv 的文件格式存储数据，从而能在 Excel 中打开。

以下为管理员可以额外执行的操作。

6. 指定单元格修改

管理员可以通过点击“修改单元格”按钮进入修改单元格界面，界面中可以通过点击选中的方式选择并删除部分单元格中的数据，防止测评数据异常等意外情况带来的错误结果正常显示。选择完成后管理员点击“确定”即可执修改除该项数据操作并返回排行榜。

3.3.4 模型测试

需求：用户在模型测试界面可以完成 测试题目选择、模型接口上传、测试流程展示、对抗评测、自动评测、测试结果与反馈、历史记录查看 七种操作。

1. 测试题目选择

平台提供多种类型的测试集，涵盖文本生成、问题回答、语言翻译、情感分析等任务。测试题目拥有不同难度级别，以适应不同用户的需求和技能水平。测试数

数据集的选择界面与数据集列表类似，用户可以查看各数据集的名称、难度、适用场景等信息，通过选择框指定上传的模型将要进行测试的数据集。

2. 模型接口上传

选定测试集后，用户可以上传以提供自己模型的调用接口，在之后的测试流程中平台将会调用该接口进行评测。平台接收到用户上传的接口后会进行尝试调用，若发现接口调用出现异常情况，将弹出提示引导用户确认模型接口正确性并重新上传接口。

3. 测试流程展示

页面将提供清晰的测试流程，在不同的测试流程下提供不同的引导。在自动评测等部分流程中用户只需耐心等待测试完成即可，上述需要用户等待的测试过程中还会提供实时反馈，在页面上显示测试进度和经历时间等信息，用户可以随时观察了解当前情况。

4. 对抗评测

用户在测评时可以选择参与对抗评测，平台随后将会在界面上展示对抗评测界面。用户可以阅读界面上方的对抗评测说明与规则，在下方的对抗选择界面进行操作，如在输入框输入自己的问题或选择随机生成问题，观察下方输出框中两个模型的结果，并通过点击对应按钮的方式做出判断。对抗评测进行时，平台将按照基于 Elo 值的匹配机制选择进行对抗的模型，并基于结果更新 Elo 值。对抗评测和下方自动评测均结束后将跳转至答案评估界面。

5. 自动评测

平台将能对数据集自动进行评测，正常数据集中的题目均为客观题，通过与数据集提供的标准答案比对的方式进行判断正误，过程中平台将记录得分、正确率等相关信息，用于测试结束后的展示。自动评测和上方对抗评测均结束后将跳转至测试结果界面。

6. 测试结果与反馈

测试结束后页面显示测试结果，将模型的测试结果进行分析，生成综合报告，包括得分、准确率、响应时间等指标。页面会提供图表和图形化界面，以便用户更直观地了解模型的性能。用户可以观察报告，并在确认无误后点击“返回”按钮回到测试主页。

7. 历史记录查看

平台会在后台保存用户的测试历史记录，包括测试时间、选择的模型、测试的数据集、得分等信息。用户可以点击“测试记录”按钮进入历史记录页面，随时页面将展示历史记录列表，用户可以方便地查看过往的测试记录，方便复查和比较不同模型的性能。

3.4 设计约束需求

由于模型的上传需要在电脑端完成，故主要考虑提供网页端支持。

3.5 性能需求

1. 任务调度与处理

任务队列：由于大模型生成任务通常耗时较长，可以使用任务队列的方式管理任务，比如使用高吞吐量的消息队列系统或任务队列框架来避免请求长时间阻塞，确保任务可以按序进行处理。

并行处理：支持多任务并行处理，以提高测试效率。使用多线程、多进程或分布式计算等技术，确保系统能够同时处理多个测试任务。

2. 数据处理与传输

分页技术：由于文本数据集通常较大，采用前后端分页策略。后端分页可以降低网络传输延迟，将数据按需发送到前端。前端分页可以减少页面渲染开销，确保用户在浏览数据时有良好的体验。

3. 数据更新与排行榜

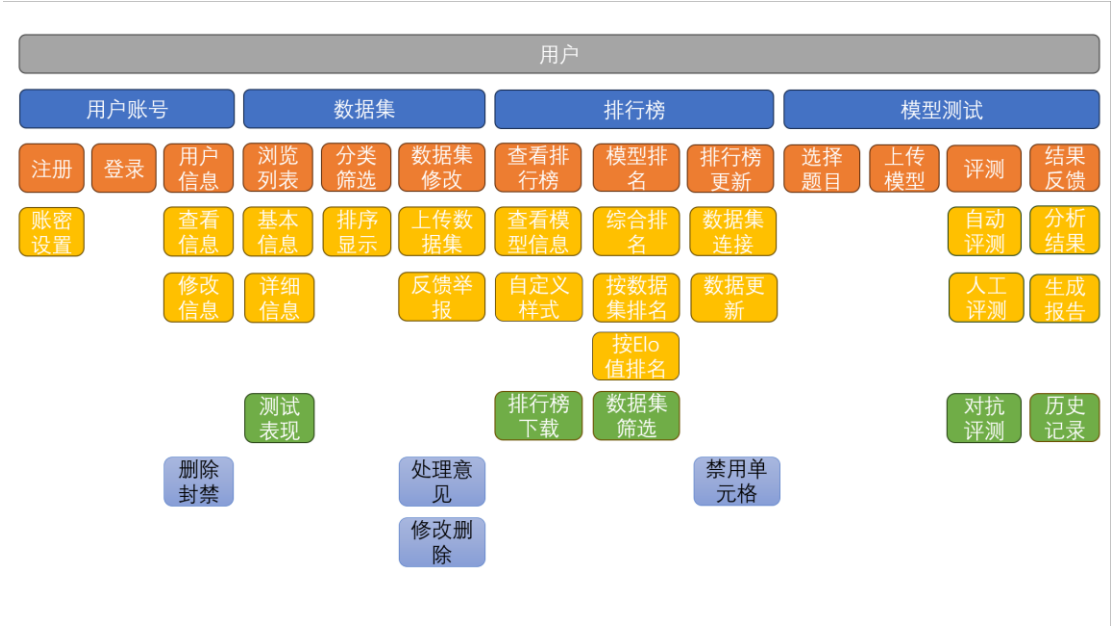
局部更新：当数据集新增条目或个别条目更新时，只对新增或修改的数据进行处理和测试。避免对整个数据集进行全量测试，以降低系统开销。

排行榜更新：定期更新排行榜结果。在有新的测试结果时，及时更新排行榜信息，确保用户获取到最新的模型性能排名。

4. 用户故事地图与用户故事卡

4.1 用户故事地图与故事卡

以各大类功能为分割标准，项目分为四个部分：个人信息、数据集、排行榜与模型测试。故事地图中，蓝色表示功能大类，橙色为主要功能点，其下每列展示子功能点，绿色为可选功能点，浅蓝色为管理员特权功能。



4.2 功能点优先级与难度分析

以 1-4 为等级划分给出各功能点的优先级与难度情况，1 为最低，4 为最高。

功能点	详细说明	优先级	难度
注册	用户注册账号	4	1
登录	用户登录，验证身份	4	1
查看用户信息	用户可以查看自己的信息	4	1
数据集查看	用户可以查看现有数据集列表	4	1
详细信息查看	用户可以点击查看指定数据集详细信息	4	2

排行榜查看	用户可以查看排行榜	4	2
排行榜排序	用户可以选择排行榜排序标准	4	2
测试流程	用户可以选择数据集、上传模型并展开测试	4	4
排行榜维护	实时更新排行榜情况	3	1
管理用户信息	管理员可以查看其他普通用户的信息	3	2
修改用户信息	用户可以对密码、头像等信息修改	3	2
数据集筛选	数据集针对不同标准进行筛选后显示	3	2
测试流程导引	测试过程中提示用户随时操作	3	3
数据集上传	用户可以上传数据集	3	3
实时测试反馈	测试过程中将展示测试进度等实时数据	3	3
榜单样式修改	用户自定义排行榜显示的样式	2	2
指定单元禁用	管理员可以禁用排行榜中的部分单元	2	2
测试历史记录	用户可以看到进行过的历史测试的信息	2	2
测试报告生成	测试完毕后向用户展示测试报告	2	2
数据集反馈	用户可以对数据集提出意见并反馈	2	3
处理用户反馈	管理员可以处理用户对数据集的反馈	2	4
排行榜下载	用户可以下载排行榜	1	2
数据集筛选	用户可以选择部分数据集作为排行标准	1	3
人工测试	用户可以参与人工评测环节，如对抗评测等	1	4

5. 产品原型

原型设计链接为 <https://modao.cc/app/3pX6Xolus2xse8s3RD7ZIM>，下面对各主要界面进行简要介绍。

1. 主页



用户进入平台网页后展示主页，主页显示简要的网站介绍与其余页面的跳转栏，用户应当首先点击右上角的登录按钮完成登录或注册功能。

2. 注册/登录页面

此处展示用户注册页面。新用户初次注册账号，通过手机验证码的方式绑定后即可完成验证，老用户也可直接点击立即登录跳转至登录页面。登录页面与之类似，用户完成登录操作后返回主页，此时即可进行各种操作。

3. 个人信息页面





李好

普通用户

我关注的模型

查看更多

GPT-4

OpenAI

当前排在第一位

文心一言

百度

当前排在第二位

GPT-3.5-turbo

OpenAI

当前排在第五位

ChatGLM-130B

清华大学&智谱AI

当前排在第八位

我关注的数据集

查看更多

TriviaQA

文本问答

大规模数据集

Quoref

文本问答

较大规模数据集

GraphQuestions

事实型问答

中规模数据集

QALD-9

多语言问答

超大规模数据集

我的测试记录

查看更多

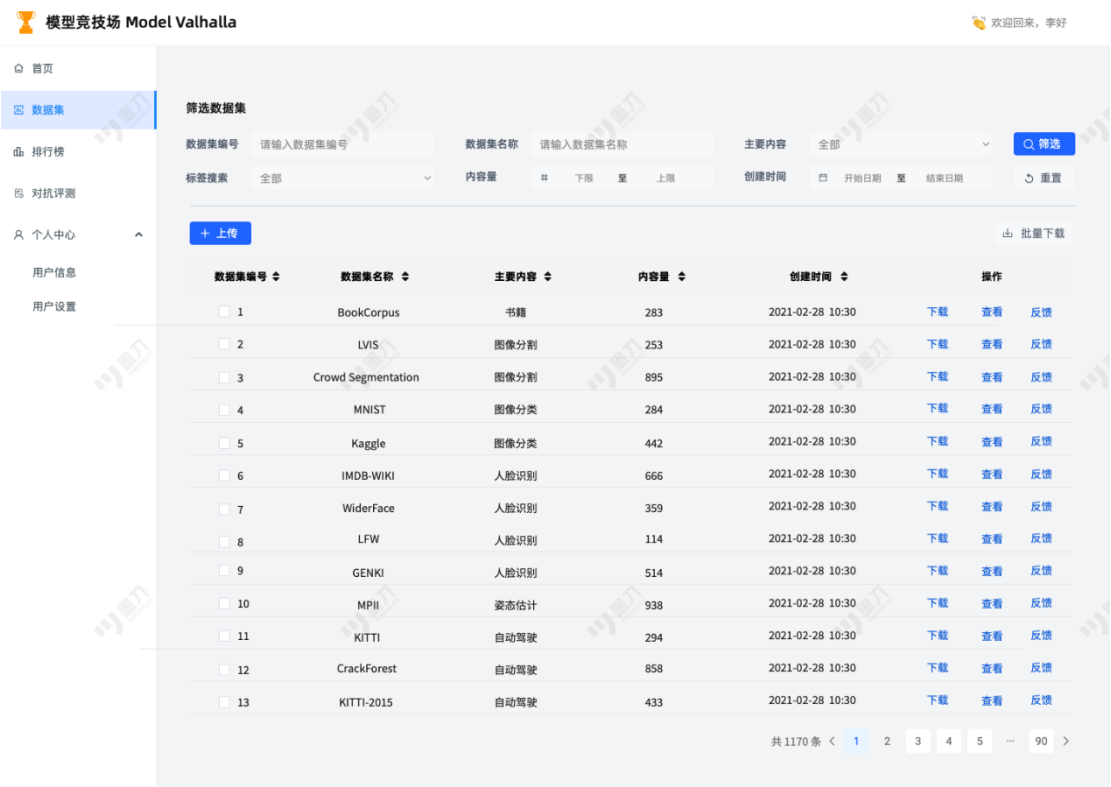
模型: GPT-4 ; 任务: 机器翻译; 数据集: WMT 2014 English-French
2023年10月23日

模型: 文心一言 ; 任务: 文本分类; 数据集: NatCat
2023年10月22日

模型: GPT-3.5-turbo ; 任务: 上下文问答; 数据集: QuAC
2023年10月21日

用户点击“个人中心”后弹出下拉框，点击“用户信息”，将跳转到用户信息面，如图所示，可以看到用户的姓名、头像和身份。此外，页面也展示了用户关注的模型、数据集和测试的记录等基本信息。点击“用户设置”，跳转到用户信息编辑页，里面可以看到用户名、账号、手机号码、电子邮箱以及注册时间

4. 数据集页面



数据集页面如上所示，包括了筛选框、数据集列表和操作功能区域。

5. 排行榜页面

排行榜页面如上所示，会展示模型在各项排序标准下的排名情况。排行榜和数据集均提供详情展示，具体展示见下一页面。

6. 详情页面

此处以模型详情为例，展示详情将弹出展示框，框中包含详细资料、数据集表现等若干分栏，用户可以根据需要点击并分别查看。

欢迎来到模型排行榜！

这里记录了各个模型的详细信息、数据集综合表现以及基于elo计算的对抗得分
想知道谁最强大？这里会给出最公正的答案



综合排名	模型名称	平均消息长度	数据集综合得分	elo对抗得分	详情	许可
1	GPT-4	86.4	8.99	1181	查看	Proprietary
2	GPT-4	86.4	8.99	1181	查看	Proprietary
3	GPT-4	86.4	8.99	1181	查看	Proprietary
4	GPT-4	86.4	8.99	1181	查看	Proprietary
5	GPT-4	86.4	8.99	1181	查看	Proprietary
6	GPT-4	86.4	8.99	1181	查看	Proprietary
7	GPT-4	86.4	8.99	1181	查看	Proprietary
8	GPT-4	86.4	8.99	1181	查看	Proprietary
9	GPT-4	86.4	8.99	1181	查看	Proprietary
10	GPT-4	86.4	8.99	1181	查看	Proprietary
11	GPT-4	86.4	8.99	1181	查看	Proprietary
12	GPT-4	86.4	8.99	1181	查看	Proprietary
13	GPT-4	86.4	8.99	1181	查看	Proprietary
14	GPT-4	86.4	8.99	1181	查看	Proprietary

GPT-4

关注

综合排名

1

详细资料

数据集表现

对抗记录

讨论

基本信息

GPT-4是由OpenAI开发的大规模语言模型，可以在各种主题和任务上生成自然语言文本。它是GPT-3.5的后继者，也是ChatGPT的动力来源。

GPT-4代表着 Generative Pre-trained Transformer 4。它是一个深度神经网络，使用一种叫做 self-attention 的技术，从大量数据中学习并基于给定输入生成文本。

GPT-4有1.8万亿个参数，比ChatGPT使用的模型多10倍。它还可以访问一个更大更多样化的1PB数据集，包括网络文本、书籍、新闻文章、社交媒体帖子、代码片段等等。

GPT-4使用了 self-attention 层的 Transformer 架构，可以学习输入文本的长距离依赖和上下文信息。它还采用了稀疏注意力、可逆层和激活检查点等技术，以减少内存消耗和计算成本。

GPT-4使用自监督学习进行训练，这意味着它在没有任何人工标签或反馈的情况下从自己生成的文本中学习。它使用一种称为掩码语言建模 (MLM) 的目标函数，随机屏蔽输入文本中的某些标记，并要求模型根据周围标记来预测它们。

GPT-4可以执行各种自然语言任务，如文本摘要、问答、文本生成、情感分析、机器翻译等。它还可以使用自然语言提示或说明在单个文本中处理多个任务。

引入论文

[GPT-4 Technical Report](#)

OpenAI

2023年3月15日

许可

Proprietary

官方主页

openai.com/gpt-4

7. 对抗评测页面



此处以对抗评测页面作为测评页面的展示示例，页面包含评测规则指引与展示区域，让用户可以轻松地完成测评操作。

6. 总体架构与技术选型

我们的产品以网页为载体，采用前后端分离的总体架构。

前端采用 Vue.js 框架开发，通过组件化思想，将页面拆分为独立组件，方便维护。通过 Vue 的数据绑定机制和指令系统，我们可以简化 DOM 操作，提高团队的开发效率。同时，Vue 强大的状态管理工具和丰富的插件工具支持也有利于我们进行快速、高效的前端开发。

后端采用 Django REST Framework 框架开发。Django REST Framework 是一个强大的 Django 扩展，用于构建灵活、高效的 Web API。使用 DRF 进行开发，我们可以定义数据模型和序列化器，将复杂的数据结构转换为 JSON 格式，或反之；也可以通过 DRF 提供的多种视图类，包括基于类的视图和基于函数的视图处理 API 请求，控制数据的输入和输出。另外，DRF 的路由系统将 API 端

点映射到相应的视图类上，并且还提供了丰富的身份验证和权限控制选项，确保 API 的安全性。在处理请求时，我们可以利用 DRF 的请求解析器和响应渲染器处理不同的数据格式，包括 JSON、XML 等。在数据处理方面，我们可以使用 DRF 提供的过滤、排序和分页功能。这些优点使得我们可以构建出功能强大、易于维护的 Web API。后端部署在服务器上，使用 docker 部署，数据库采用 MySQL。

7. 开发迭代与分工计划

7.1 开发迭代计划

轮次	周次	任务
第一轮	第 6-7 周	搭建环境(前端 Vue.js，后端 docker+DRF+MySQL)，创建前后端通信连接，完成用户注册登录功能
第二轮	第 8 周	完成用户账号的注册/登录，个人信息的查看/更改，密码的找回/重置等功能，发布 版本 1
第三轮	第 9-10 周	完成排行榜的排序、筛选、模型查看等功能
第四轮	第 11 周	完成数据集的检索、筛选、查看、上传、下载等功能，发布 版本 2
第五轮	第 12-13 周	完成测试部分模型筛选、任务选择、主客观评测等功能，发布 版本 3
收尾	第 14 周	整合全部功能，设计使用教程、完善文档，发布 最终版本

7.2 分工

原型设计：姚润茂，方钧同

前端：姚润茂，方钧同

后端：钟健坤，李科霖

项目需求沟通负责人：姚润茂

会议记录：方钧同

组织组内会议、统筹进度：李科霖，钟健坤