Problem Chosen	2023	Team Control Number
A	MCM/ICM	2318036
	Summary Sheet	

探索世界:对谜题解决和 Tweet 分享模式的见解

摘要：吸引了数百万人的字谜游戏《世界谜语》现在归《纽约时报》所有。对于该公司的游戏编辑来说，游戏如何破解并在社交媒体上分享是至关重要的信息，因为它可以用来指导未来的谜题设计，并最终最大化玩家总数。本文旨在基于推特上的词属性和结果报告构建定量模型，预测未来玩家的格局。

在对原始数据进行检查和清理后，我们首先定义了 12 个属性指标，衡量其熟悉度(使用频率)、关联度、混淆度和词的组成特征。它们是提前计算出来的，因为下面的模型会频繁地使用这些指标。

对于问题 1，我们建立了一个基于 SIR 模型的动态系统，称为目标-两玩家-损失(T2PL)，以解释世界银行报告的每日波动。玩家还被额外分为两类:普通玩家和忠诚玩家，每一类玩家的流失率都不同。这使得该模型可以更好地模拟不同时间段的不平等流失率。文章还探讨了单词属性与难度模式玩家数量之间的关系，发现某些属性会影响难度模式报告的百分比。

对于问题 2，我们开发了一个 P&S 模型，这是一个使用模拟算法和梯度下降来模拟玩家猜词和分享游戏结果的行为的模型。模拟器的工作原理是使用可观察到的信息消除所有不满意的单词，然后以词频作为权重从剩余的单词列表中随机抽取单词。然而，我们发现模拟结果并不能完美匹配真实分布。因此，我们用 7 个变量重新缩放了这个分布，这些变量代表了玩家在给定不同分数时可能会如何分享他们的分数。通过梯度下降对它们进行优化，可以生成更好的分布预测。利用 P&S 模型，我们预测 2023 年 3 月 1 日 EERIE 一词的分布为(0,0,9%，29%，45%，14%，3%)。

对于问题 3，我们需要根据难度对谜题进行分类。我们使用 3 个聚类 k 均值对所有报告的试验分布进行聚类分析，每个聚类标记为容易、中等和困难。我们拟合一个随机森林模型，使用开始定义的属性指标将单词分为这三类。计算每个指标与难度之间的相关系数，显示这些指标影响谜题难度的方向。对聚类的灵敏度也进行了讨论。基于我们的模型，EERIE 的难度比较大。

对于问题 4，我们进一步探索单词难度的影响。使用线性回归，我们发现单词难度对报告结果的数量有明显的影响:更难的谜题导致更少的报告。难度也与选择难度模式的玩家比例相关。通过这一部分的研究，我们发现这种相关性是由单词难度影响普通模式玩家数量而形成的。

通过揭示单词属性、谜题难度和游戏报告模式之间的相互作用，世界大战运营商可以更深入地了解他们的玩家。基于这一发现，我们还可以提出一些明智的建议。

关键词:世界;动态系统;模拟;k 均值;随机森林

目录

1 介绍

1.1 背景与文献综述

由 Josh Wardle 发明的字谜游戏《世界大战》以其简单和多样的变化吸引了数百万人。除了这款游戏，也许世界大战走红的主要原因是其集成的分享格式，包括表情符号方块，这些表情符号在 Twitter 上广泛传播。2022 年 1 月，世界大战被纽约时报公司收购，并由他们运营至今。《世界大战》的官方网站每天只发布一块拼图，这种稀缺性也被认为是《世界大战》成功的原因之一。

在《世界大战》中，玩家的目标是在六次猜测中破解一个五个字母的单词。每次猜测完成后都会给出反馈:绿色突出显示的字母表示答案在相同的位置有相同的字母。黄色表示这个字母出现在答案中，但在另一个地方。灰色表示答案中没有这个字母。一般来说，普通玩家需要三到五次尝试，但不同的单词之间可能会有很大的差异。除了正常版本之外，还有一个困难模式世界，规定在接下来的猜测中必须保持每个发现的正确单词(黄色或绿色)[11]。

很多研究都集中在寻找解谜的最优策略上[1][4]。然而，玩家的模式似乎也值得探索。作为纽约时报旗下的主要产品，其运营商希望追踪和预测 Twitter 上分享的游戏数量。此外，发布的词也要考虑得很好，因为简单的问题无法挑战有经验的玩家，而像“rebus”或“tapir”这样的罕见词会让大多数粉丝感到沮丧[10]。因此，也期望有一个定量模型来预测根据给定单词的尝试分布。

1.2 问题重述

考虑到背景，在本文中我们需要解决以下问题:

Task 1:结合《世界大战的游戏机制，构建一套反映单词属性的指标，并将其应用于后续模型。

任务 2:建立一个模型，解释报告的结果数量和报告的分数在困难模式下的百分比的趋势，并使用它来预测 2023 年 3 月 1 日报告的结果数量。进一步，分析单词属性对在困难模式下打出的分数百分比的影响。

任务 3:开发一个模型，可以根据单词预测报告结果的分布，并用它来预测 2023 年 3 月 1 日 EERIE 这个单词的分布。此外，说明模型的不确定性和准确性。

任务 4:根据单词的难度对单词进行分类，并解释单词的属性和单词的难度之间的关系。

任务 5:对数据集进行全面分析，并给出一些有趣的结论。

1.3 我们的工作

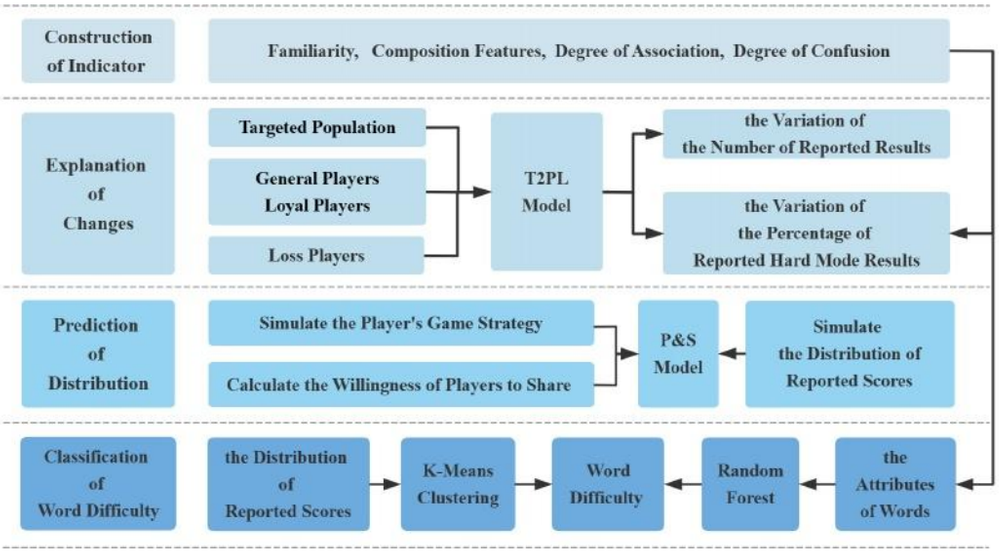

图 1:我们的工作流程图

首先，我们构建了可以衡量单词熟悉度、组成特征、关联度和混淆度的四类指标，并用这些指标来反映单词的属性。

其次，我们在 SIR 模型的基础上开发了 T2PL 模型，这是一个动态模型，可以很好地解释报告结果数量和报告困难模式结果百分比的总体趋势。在此基础上，我们探讨了单词属性对报告的困难模式结果百分比的影响。

第三，我们使用该算法模拟世界玩家在猜词时的策略，从而模拟结果的初始分布。考虑到玩家的心理特征，我们加入了表示玩家分享分数意愿的参数，并模拟了报告结果的最终分布。

第四，我们根据分数分布对单词进行聚类，并根据难度将单词分为 3 类。将聚类结果作为标签，构建随机森林模型，根据单词的属性对单词的难度进行分类。

最后，基于上述模型的结果，我们进行了进一步的探索，发现了一些有趣的结论。

2 假设和注释

2.1 模型假设

考虑到建模所需的条件，我们做如下假设:

假设 1:世界大战日用户数的总趋势不会发生变化。理由:这是根据观察到的日使用量来预测未来趋势所必需的。

假设 2:大多数玩家使用理性策略。

理由:为潜在玩家建立数学模型，有必要假设他们实际上是在使用策略，不会采取不必要的行动。否则，基于潜在策略来模拟结果就会变得毫无意义。

假设 3:随着时间的推移，玩家的技能没有显著变化。理由:由于《世界大战》是一段时间的游戏，玩家需要改进他们的策略，这可能会影响不同日期的尝试时间分布。然而，经验丰富的玩家放弃世界大战，而新人同时加入，产生了相反的效果。考虑这些可能性就太复杂了。

假设 4:在任务 2 (T2PL 模型)中，玩家和分享结果的人没有区别。

理由:为了方便起见，在任务 2 中对玩家进行建模，尽管实际上使用了 Twitter 报告号码。这是因为在这一步中没有足够的信息来区分这两个类别，从建模玩家切换到建模分享结果的玩家是有意义的。

2.2 符号

Symbol	Definition	Symbol	Definition
N	Number of reported results	P	Number of all players
H	Number of reports in Hard Mode	P_{loy}	Number of loyal players
P_H	Percentage of scores reported that were played in Hard Mode. $P_H = H/N \times 100\%$	P_{gen}	Number of general players
D	All words of length 5 in dictionary data	W_{sj}	Probability of sharing when finishing with j tries
T	Number of Targeted Population	p_{ij}	Probability of solving Wordle #i with j tries

表 1:符号表。

文字属性指标不包括在内，因为下面有详细解释。

3 数据预处理

数据集 Problem_C_Data_Wordle.xlsx 包含 359 天的世界报告信息。每行由日期、当天单词、报告结果的次数、硬模式结果以及每次尝试次数的分布组成。表格中没有缺失值，但仔细检查有几个单词拼写错误。我们通过使用当天的问题编号搜索正确的世界答案来手动纠正这些问题。

Date	Contest number	original word	correct word
2022/12/16	545	rprobe	probe
2022/12/11	540	naïve	naive
2022/11/26	525	clen	clean
2022/10/5	473	marxh	marsh
2022/4/29	314	tash	trash

表 2:所有更正词

我们还检查了每个分布的百分比之和，发现 2022/3/27 问题 281“仙女”的总和为 126，这很可能是一个异常值。我们不知道哪一个百分比是错的，所以我们不能简单地把它缩小。因此，在预测百分比问题中不使用这一行数据。2022/11/30 的第 529 题“study”似乎也是一个异常值。本行报告结果的 Number 为 2569，硬模式下的 Number 为 2405，显然与其他行数据差别很大，所以修正为 25690。

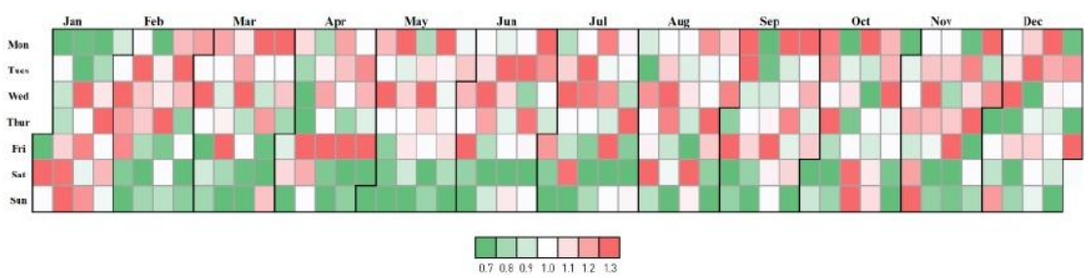

图 2:提交结果的日历图

考虑到提交数量可能存在每周周期性模式，我们创建了一个日历图(图 2)来可视化每周的变化。每一天都参考其所在周的平均值。越超过平均值越红，越低于平均值越绿。可以看出，虽然一周中的每一天都有一定的差异，但并没有普遍的规律。因此，我们假设波动只是基于报告的总趋势和单词的难度。

4 任务 1:单词属性指标

考虑到《世界》的游戏特点，我们构建的指标应该尽可能反映单词的拼写结构以及人们将其联系起来的容易程度。为了实现这一点，我们构建了如下的指标框架。

熟悉度:人们对这个词的熟悉程度以及这个词的使用频率。

构成特征:单词的结构特征，例如是否包含相同的字母内容。

关联度:当玩家猜测单词时，世界将给出更有效信息(即绿色或黄色瓷砖)的可能性。

混淆程度:单词与其他单词之间的相似程度。当相似度太大时，玩家可能需要花费更多的时间来验证在众多相似的单词中哪一个是正确答案。

基于上述框架，我们构建了 12 个指标:

Measured Feature	Symbol	Meanings
Familiarity	F	Frequency of the word
Composition feature	N_c	Number of the most repeated letter in the word
	N_v	Number of vowels in the word
Degree of association	LF_i	Frequency that the ith letter of the word appears in the ith position of all words. $(i = 1, 2, 3, 4, 5)$
	LF_{sum}	Letter Frequency Sum
	N_{FI_2}	Number of frequent 2-itmes
Degree of confusion	N_{D_1}	Number of all the words in D whose Levenshtein distance is 1 from the word
	N_{D_2}	Number of all the words in D whose Levenshtein distance is 2 from the word

表 3:所有选定参数的符号表

以下是对某些符号的详细说明:

重复次数最多的字母(二进制)c的个数:例如，Nc(“apple”)= 2，Nc(“mummy”)= 3。因为玩家通常不会同时尝试两个或三个相同的字母，所以我们预测，二进制数越大，这个c单词被猜出的可能性就越小。

第 i 个字母的字母频率(LFi):例如，苹果的第一个字母是 a，那么 LF1(“苹果”)是D中第一个字母为 a 的单词的比例。LFi越大，i在第 i 个位置出现绿色贴图的概率就越大。

字母频率总和($LFsum$):指标$LFsum$，表示中字母频率的总和这

个词。首先，字母频率(LF)的计算公式为:

$$\text{(initialize)} \quad \forall_{c \in C} \left[LF(c) = 0 \right] \tag{1}$$

其中C为字母表集，D为单词集，$len(w)$表示单词长度。计算完LF 后，对$\mathrm{LF}sum$一个单词中的所有字符求和得到LF:

$$LF_{sum}(\text{"}c_1c_2c_3c_4c_5\text{"}) = \sum_{i=1}^{5} LF(c_i) \tag{4}$$

$\mathrm{LF}sum$越大，玩《世界》时出现黄色贴图的概率就越大。

频繁 2 项数(NFI2):该指标统计频繁 2 项集合中一个单词包含的频繁项数(F$I2$)。通过FP-Growth算法(一种频繁模式挖掘算法)计算得出的，支持度D/10。较大的 N FI2 增加了通过N$FI2$ 接近猜测获得信息的概率，因此假设N F I 2 高时更容易。

1/2 Levenshtein 距离单词数(n$D1$, $ND2$):Levenshtein 距离是两个字符串之间从一个到另一个所需的最小编辑操作数，因此所有 Levenshtein 距离等于 1 或 2 的单词都是与答案相似的单词。当$ND1$ 和 $ND2$ 是 高时，玩家可能需要花费更多的时间来验证众多相似的单词中哪一个是正确的答案，并且更有可能猜不出一个单词。

5 任务 2:预测每日报告和困难模式百分比

5.1 问题分析

很明显，《世界大战》在去年迅速走红。人们发送这些标志性的世界大战结果推文，吸引更多的人尝试并分享这款游戏。2 月初，分享推文呈指数级上升，3 月后逐渐下降。因此，SIR(疑似感染-康复)模型可以很好地捕捉到这种趋势，该模型最初旨在解释传染病[9]。建立类似的动态模型来解释游戏用户数量波动也在许多研究中被观察到，并被证明是成功的[5]。

5.2 模型的建立

与 SIR 模型一样，我们可以假设三组人:目标人群、当前玩家和流失玩家。目标人群具有成为玩家的潜力，由于他们推文的广告效应，这种转化率与当前玩家总数成正比。因此，目标人群的减少可以写成目标人群、当前玩家和一个常数的随机数的倍数。对于当前玩家来说，除了新加入的玩家，也有玩家变得厌倦而放弃游戏。这群人只与当前玩家的数量和常数β时延有关。有了这些条件，一个动态系统可以公式化如下:

$$\begin{cases} \dfrac{\partial T}{\partial t} = -\alpha \cdot T \cdot P \\ \dfrac{\partial P}{\partial t} = \alpha \cdot T \cdot P - \beta \cdot P \\ \dfrac{\partial L}{\partial t} = \beta \cdot P \end{cases} \tag{5}$$

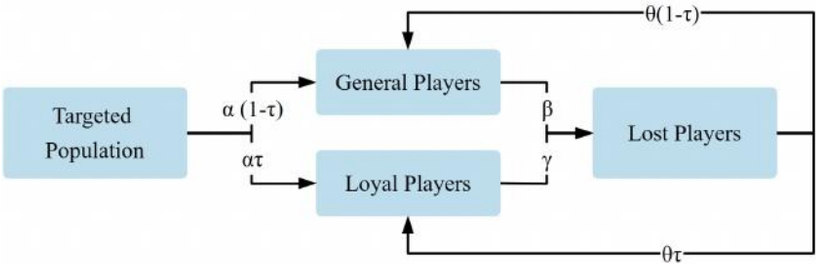

图 3:两型球员动态系统

然而，这一模式既不能满足 2 月份见顶后的急剧下降，也不能满足下半年的小幅下降。毕竟，虽然许多人可能会逐渐放弃这款游戏，但其他人可能会保持对《世界大战》的喜爱。建模所有玩家都有相同的玩家离开率βmight 效果并不好。因此，我们对模型进行了改进，将所有玩家分为对世界大战感到厌倦的倾向较低并不断发送世界大战 tweet 的忠诚玩家和跟随趋势并随趋势离开的普通玩家，称为 T2PL(Target-2-players-Lost)模型。每两个类别都有一个独特的衰减率$\beta$and <s:1>。两种类型的球员的推文都会有相似的效果，所以在计算潜在人口减少时使用两类的总和。玩家在每个类别中所占的百分比也需要确定。我们设置变量τas 为新玩家成为忠实玩家的比率，设置变量$p$as 为两种类型的初始分裂。如果有些玩家在离开一段时间后想要再次尝试《世界大战》，那么就需要使用一个额外的变量$\theta$is 来描述失而复得率，这意味着必须提前对失而复得玩家的初始 0 值Lalso 进行建模。现在，最终的公式应该是:

$$\begin{cases} \dfrac{\partial T}{\partial t} = -\alpha T \cdot (P_{loy} + P_{gen}) \\ \dfrac{\partial P_{loy}}{\partial t} = \tau \cdot \left[\alpha T \cdot (P_{loy} + P_{gen}) + \theta L\right] - \beta \cdot P_{loy} \\ \dfrac{\partial P_{gen}}{\partial t} = (1-\tau) \cdot \left[\alpha T \cdot (P_{loy} + P_{gen}) + \theta L\right] - \gamma \cdot P_{gen} \\ \dfrac{\partial L}{\partial t} = \beta \cdot P_{loy} + \gamma \cdot P_{gen} - \theta L \end{cases} \tag{6}$$

初始值： $T_0,\ P_{loy_0} = P_0 \cdot p,\ P_{gen_0} = P_0 \cdot (1-p),\ L_0$

5.3 求解模型

通过估计一组初始值(如:α,β,γ,T0,L0,τ,θ,p)和观测值$P0$，可以利用正演欧拉法对动态系统进行数值求解。我们的目标是将动态系统拟合到结果曲线上。目标函数是 1 与每个样本点估计值与实际值之比之间的均方误差。因为地面真值曲线上最大值和最小值之间的范围太大，所以重新缩放。

$$\min_{\alpha,\beta,\gamma,T_0,\tau,p} \sum_{t=1}^{N} \left(1 - \frac{\hat{P}_i(\alpha,\ \beta,\ \gamma,\ T_0,\ \tau,\ p)}{P_i}\right)^2 \tag{7}$$

在没有显式求解函数的情况下，我们的团队使用自动梯度包(Pytorch)来优化参数。每个参数通过梯度下降来更新。只要初始值大致是正确的尺度，它就会找到产生良好拟合曲线的参数的局部最小值。

5.4 解与结果

经过 5000 epoch 的训练，学习率为 0.01，我们找到了每个变量的最优值。

	α	β	γ	θ	T_0	L_0	τ	p
initial value	5×10^{-7}	0.02	2×10^{-3}	2×10^{-4}	310000	30000	0.2	0.2
optimal value	3.03×10^{-7}	0.0228	3.7×10^{-3}	2.0×10^{-4}	431193	29918	0.0992	0.1683

表 4:各变量的初始值和最优值

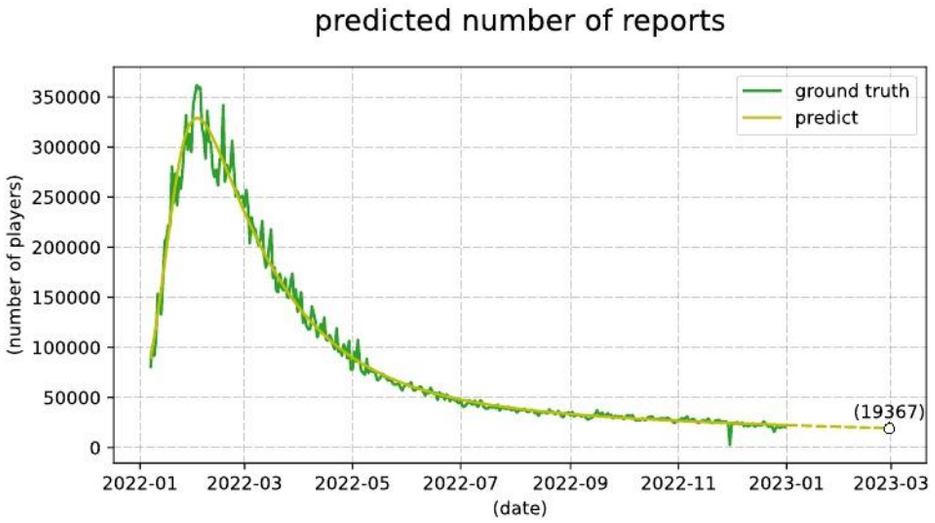

图 4:预测的报告数。预测天数用虚线绘制。

绘制曲线如图 4 所示。该模型很好地反映了 2022 年世界大战结果的趋势。到 2023.3.1 年，预计将有 19367 个用户发送世界大战。

结果：由于模型本身不能提供可能的预测范围，因此我们通过考虑 MAPE(平均绝对百分比误差)来进行估计，该误差通常用作预测精度的度量。它的定义为:

$$\text{MAPE} = \frac{100\%}{n}\sum_{t=1}^{n}\left|\frac{\hat{P}_t - P_t}{P_t}\right| \tag{8}$$

给定观测日期的 MAPE = 0.083，我们可以假设结果将落在±MAPE 范围内。因此，估计的区域给出为:

$$\text{Confidence Interval}(t) = \left[(1-\text{MAPE})\cdot\hat{P}_t, (1+\text{MAPE})\cdot\hat{P}_t\right] \tag{9}$$

因此，报告数量的预测区间为[17760,20976]。

此外，该模型还可以进一步深入了解《world》玩家的结构。我们绘制了 2022 年每个类别的变化。从图 5 可以看出，一般玩家在 2 月份快速增加，然后在结束时下降到非常低的水平。而忠实玩家在整个过程中变化不大。该模型预测，2022 年 12 月 31 日的普通玩家比例只有 19%，2023 年 3 月 1 日的普通玩家比例为 21%，这表明在不久的将来，大多数玩家将是忠诚的，而且可能会有更多的熟练玩家。

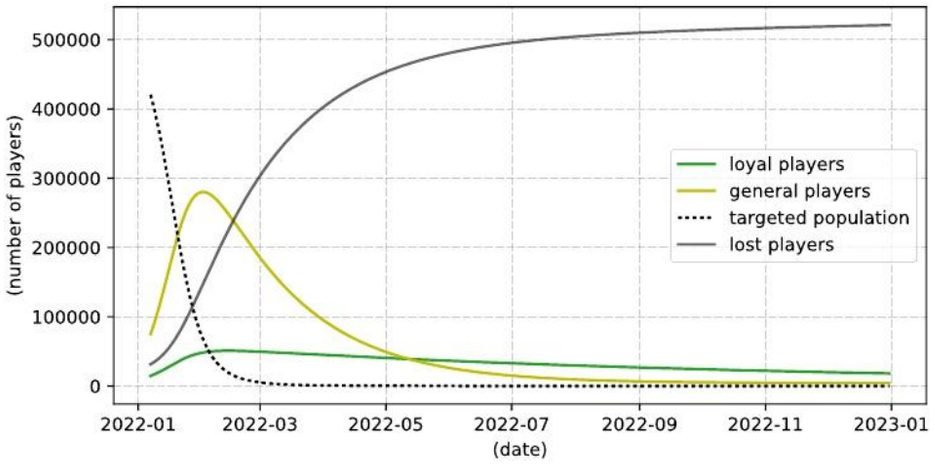

图 5:两种类型的玩家动态系统

5.5 困难模式百分比估计

与报告结果数量的趋势不同，报告在困难模式中玩的分数百分比从 2%缓慢上升到 10%，然后保持稳定。这种整体趋势或许可以用上面提到的玩家结构的变化来解释，其中忠诚的玩家

更有可能选择困难模式，因为他们更喜欢这款游戏。如下图所示，忠实玩家在整体玩家基础中的百分比趋势与在困难模式中体验的分数百分比趋势基本相同。

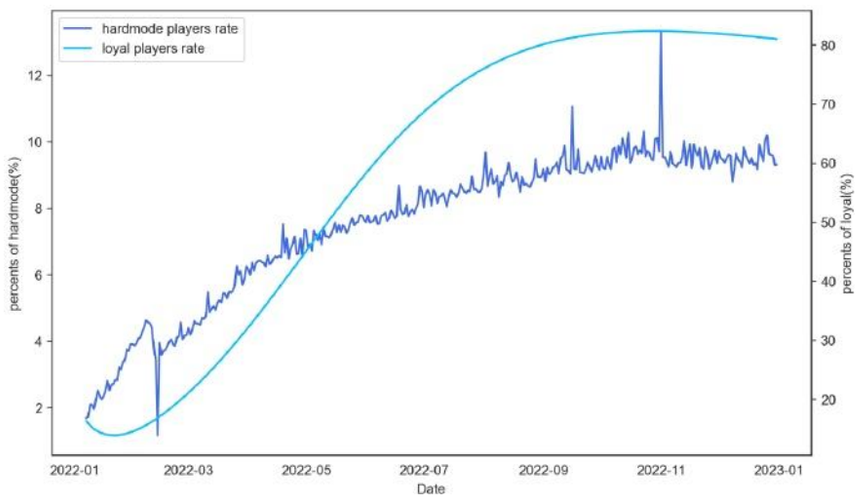

图 6:高难度模式报告百分比与忠实玩家百分比曲线，重新缩放。

此外，在困难模式中报告得分的百分比总是围绕整体趋势出现小幅波动，这些波动可能与当天单词的属性有关。为了探讨这种关系，我们以困难模式报告的百分比为因变量，以反映词属性的所有指标为自变量，建立了多元线性回归模型。为了避免整体趋势对波动探索的影响，选择前一天报告的分数百分比作为控制变量。

回归结果如下(此处仅给出回归系数显著的变量结果):

P_H on the previous day	F	N_c	LF_4	LF_{sum}	N_{FI_2}
0.9608^{***}	$-6.342 \times 10^{-12*}$	$2.776 \times 10^{-3***}$	0.0291^{*}	-0.0423^{***}	$6.777 \times 10^{-4*}$
(0.0111)	(2.814×10^{-12})	(5.959 × 10−4)	(0.0113)	(0.0110)	(2.761×10^{-4})

表 5:回归结果，给出系数及相应的标准误差。请注意 : *, **, *** 代表$p< 0.05$,$p< 0.01$,$p< 0.001$。

结果表明，F和$LFsum$有 对在困难模式下玩的分数报告百分比有显著的负面影响，而Nc，LF 4 和$NFI2$ 对它有显著的积极影响。并且通过后面的分析可以发现，这些指标都是决定单词难度的重要因素。

选择两个最突出的属性，并绘制在困难模式中报告的分数百分比的分类框图。结果表明，在困难模式下，当单词包含相同的字母时，报告的分数百分比较高，而当单词包含较高频率的字母时，报告的分数百分比较低。

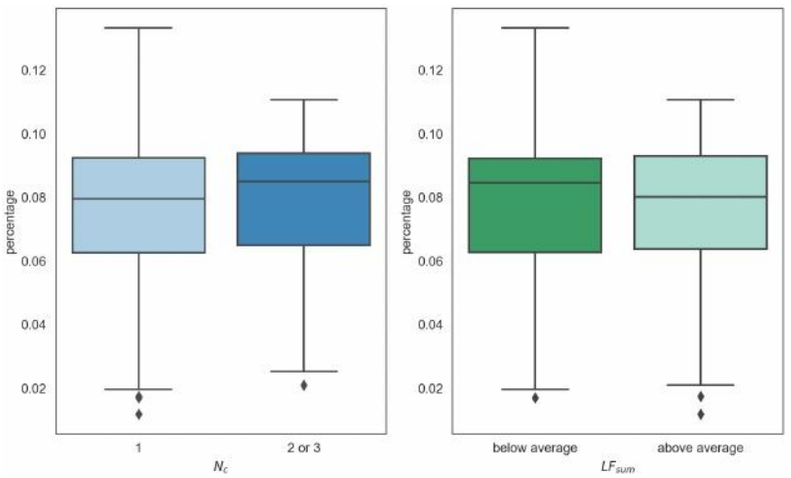

图 7:箱形图结果

6 任务 3:预测报告分布

6.1 问题分析

《世界大战》是一款互动游戏。当玩家填写一个单词时，世界大战给出反馈，玩家根据世界大战的反馈选择一个新单词。玩家策略的每一步都与前一步填写的单词有关，这为游戏继续进行提供了很多可能性，以至于几乎不可能计算出玩家在指定次数内猜出单词的概率。为了预测未来日期(1,2,3,4,5,6,X)的相关百分比，我们设计了一个模拟算法来模拟玩家玩世界大战的策略，并通过对每个单词进行 5000 次模拟来获得模拟分布。

6.2 模型的建立

模拟算法的主要思想是:玩家从字典中随机选择一个单词进行猜测，得到世界的游戏状态。对于每个字母的状态:

1.如果是绿色，则只选择与字母对应位置的单词进行后续猜测;

2.如果是黄色，则只有该字母出现在其他位置的单词才会被选中进行后续猜测;

3.如果它是灰色的，并且不在黄色字母中，则随后的猜测将不会选择包含该字母的单词。

重复上述过程，直到玩家成功或猜出 6 次以上。注:这里我们假设玩家知道字典 D 中的所有单词。

Algorithm 1 Simulate(W_S)

(**Note:** Evaluate-Guess(W_S,W_G) return a game state of wordle. For example: W_S ="tools", W_G="brook", it return "NNYMN". "Y" means the letter is in the answer and in the correct position. "M" means the letter is in the answer but in a different position. "N" means the letter is not in the answer.)

Input: Solution word W_S

Output: The list of game states L_G

$D_W \leftarrow D$;
$L_G \leftarrow \phi$;
for i from 1 to 6 **do**
    random choose a guess word W_G from D_W;
    $S_i \leftarrow$ Evaluate-Guess(W_S, W_G);
    $L_G \leftarrow L_G \bigcup \{S_i\}$;
    if S_i ="YYYYY" **then** break;
    end if
    for j from 1 to 5 **do**
        if S_i^j ="Y" **then**
            remove from D_W the words W satisfying $W^j \neq W_G^j$;
        else if S_i^j ="M" **then**
            filter out the word W satisfying $W^k = W_G^j, k \neq j$ from D_w;
        else if S_i^j ="N" & $W_G^j \notin \{W_G^k | S_i^k = \text{"M"}, k = 0,1,2,3,4\}$ **then**
            remove from D_W the words W satisfying $W^j = W_G^j$;
        end if
    end for
end for
return L_G;

我们将模拟分布和真实分布对应点上的概率相减，取绝对值作为误差。误差值为 5.36%，并不令人满意。通过对比模拟分布的均值与真实分布的均值，我们发现 79.6%的模拟分布的均值大于真实分布的均值，即模拟结果表明玩家需要更多的猜测尝试。我们认为，这是因为尝试次数较少的猜对的玩家更有可能在 Twitter 上分享他们的结果，而多次尝试后猜对的玩家更有可能不会在 Twitter 上分享结果。

根据以上考虑，我们提出 P&S(玩过和分享)模型。玩家在玩完《world dle》后(无论是成功还是失败)进行分享的概率定义为Wsi，表示在进行 i 次尝试成功后的分享意愿($Wsi \in [0,1]$)。我们根据Wsi对模拟分布进行了修正，使用梯度下降法将误差最小化。修正后的公式如下:

$$\hat{p_{ij}} = \frac{p_{ij} W_{sj}}{\sum_{k=1}^{5} p_{ik} W_{sk}} \times 100\% \tag{10}$$

pij是第 i 个单词尝试 j 次后成功的玩家百分比，pij是第 i 个单词尝试 j 次后成功的玩家百分比。

梯度下降法得到的最优参数如下:

W_{s1}	W_{s2}	W_{s3}	W_{s4}	W_{s5}	W_{s6}	W_{s7}
1.000	0.993	0.940	0.755	0.698	0.696	0.301

表 6:尺度参数的最优值

我们用上述参数对模拟分布进行了修正，并与真实分布进行了再次比较，发现修正后的分布能够更好地拟合数据集的分布。误差从 5.36%降低到 1.50%，只有 45.6%的模拟分布的均值大于真实分布，这是一个比较理想的结果。我们将这个模型命名为 P&S 模型，其中 P 代表玩游戏，S 代表分享结果。

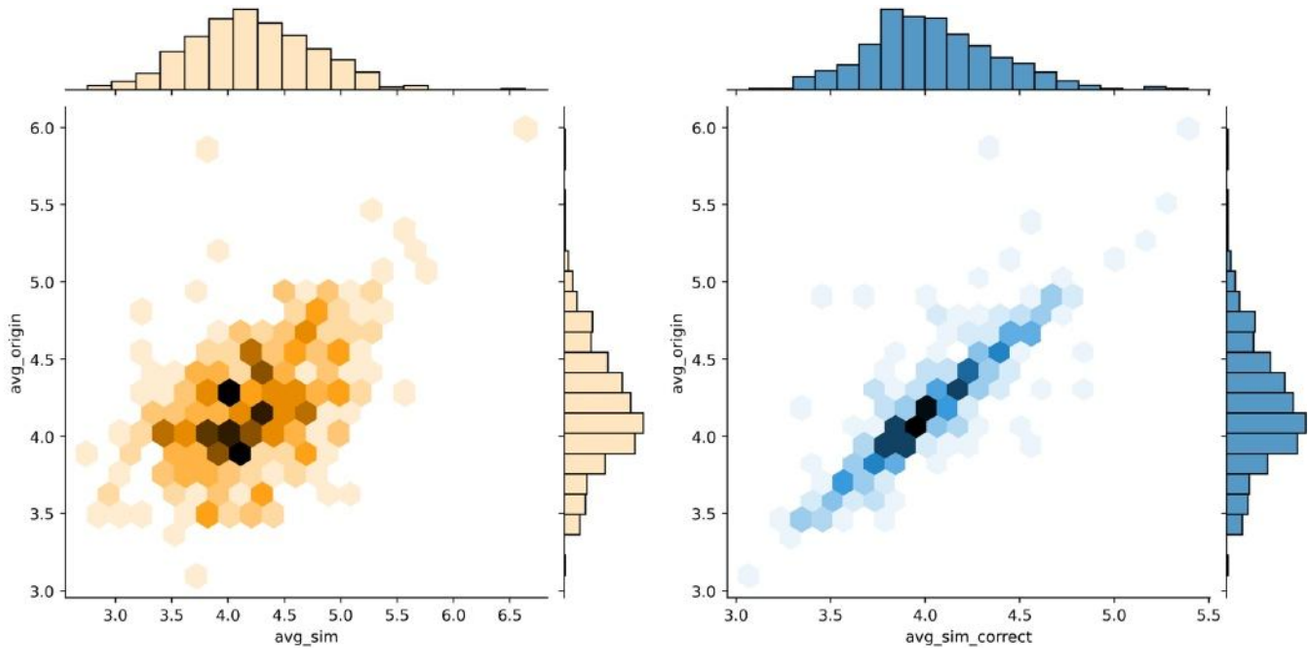

图 8:可视化模拟分布与真实分布的对比，校正前后。

6.3 预测信心和不确定性

基于我们的模型，2023年3月1日怪异的词分布情况为(0,0,9%，29%，45%，14%，3%)。由于误差值为 1.50%，我们推测真实分布的每个点的概率可能会落在模拟分布的正负 1.50%范围内。

由于模拟的随机性，P&S 模型的预测结果并不总是相同的，这将导致预测结果存在一定的不确定性。

7 任务 4:单词难度分类

在上一节中，我们建立了一个模型来分析每个单词的可能分布。然而，从这个模型中寻求洞察力的谜题设计师会发现很难使用这个信息。预计将创建一个描述谜题难度的综合标签，并通过每个 5 个字母单词的可解释属性产生对这些类别的分类。因此，我们对数据集中的分数分布进行了聚类分析，希望能识别出 3 种难度类型:简单、中等和困难。然后，我们枚举每个单词中所有可能的属性，并使用这些属性将分类算法运行到三个难度级别。通过这种方式，可以发现最重要的分类属性，然后我们可以对每个类别与单词特征的关系有一个深思熟虑的理解。

7.1 聚类分析

对于聚类算法，我们比较了两种常用的方法:k-means[3]和高斯混合模型(Gaussian Mixture Model, GMM)[7]。

k -means 尝试将数据划分为 k 个类别，生成 k 个类别质心CK。对于每个点，其最近的质心被期望为其所属类别的质心。通过初始化几个随机点作为质心，使用以下公式迭代更新:

$$x_n \in C_k^{(t+1)}, \text{ if } k = \arg\min_k d(x_n, c_k^{(t)}) \tag{11}$$

$$c_k^{(t+1)} = \frac{1}{|C_k^{(t+1)}|} \sum_{x_n \in C_k^{(t+1)}} x_n \tag{12}$$

GMM 考虑从 k 个正态分布 Nk(μk, ΣK)中抽样的所有观测数据，并试图找到参数k的最大似然对μk, ΣK。它可以使用期望最大化(EM)算法来求解。在 e 步中，计算每个样本点的责任(聚类的后验分布)。

$$p^{(t)}(k|x_n) = \frac{p(x_n; \mu_k^{(t)}, \Sigma_k^{(t)})}{\sum_{i=1}^{K} p(x_n; \mu_i^{(t)}, \Sigma_i^{(t)})} \tag{13}$$

然后在 m 步中，责任固定，模型参数更新为:

$$\mu_k^{(t+1)} = \frac{\sum_{n=1}^{N} p^{(t)}(k|x_n) x_n}{\sum_{n=1}^{N} p^{(t)}(k|x_n)} \qquad \Sigma_k^{(t+1)} = \frac{\sum_{n=1}^{N} p^{(t)}(k|x_n)(x_n - \mu_k^{(t+1)})(x_n - \mu_k^{(t+1)})^T}{\sum_{n=1}^{N} p^{(t)}(k|x_n)} \tag{14}$$

收敛后，每个样本点都属于责任最大的分布。

$$x_n \in C_k, \text{ if } k = \arg\max_k p(k|x_n) \tag{15}$$

为了决定更好的算法并选择合适的聚类数量，我们使用廓形系数[8]和 卡林斯基-哈拉巴斯指数[6]对结果进行比较。

系数是通过对所有样本点的剪影值求平均值获得的，它描述了样本点与其他样本点相比，与其聚类的距离有多近。剪影值越大，聚类就越容易区分。其数学公式为:

$$SC = \frac{1}{N} \sum_{i=1}^{N} s(x_n), \qquad s(x_n) = \frac{b(x_n) - a(x_n)}{\max(a(x_n), b(x_n)} \tag{16}$$

$$\text{Where}: a(x_n) = \frac{1}{C_i - 1} \sum_{x_m \in C_i, m \neq n} d(x_m, x_n) \qquad \text{(same cluster closeness)}$$

$$b(x_n) = \min_{j \neq i} \frac{1}{C_j} \sum_{x_m \in C_j} d(x_m, x_n) \qquad \text{(different cluster closeness)}$$

卡林斯基-哈拉巴斯指数是聚类中心方差与每个聚类方差之间的比值。如果每个聚类的中心分散开来，并且样本点聚集在每个聚类质心周围，那么卡林斯基-哈拉巴斯指数预计会很高。

$$\text{C-H Index} = \left[\frac{\sum_{k=1}^{K} |C_k| \cdot ||c_k - \mu||^2}{K - 1}\right] \Big/ \left[\frac{\sum_{k=1}^{K} \sum_{x \in C_k} ||x - c_k||^2}{N - K}\right] \qquad (c_k \text{ is the centriod of } C_k) \tag{17}$$

这两个指标的得分如表所示:

	k-means (3)	GNN(3)	k-means(2)	GNN(2)
silhouette coefficient	0.369	0.306	0.425	0.366
Calinski-Harabasz index	311.9	203.4	329.6	122.3

表 7:不同类型聚类算法的得分

最后，我们决定使用 k-means 创建 3 个聚类。虽然 2 类 k-means 似乎有更高的评价分数，但详细分类更有利于得出更具体的结论，3 类 k-means 的性能也离它不远。为了更直观地观察聚类效果，我们使用 t-SNE 算法将数据降维到 2 维。图 9 将所有三个聚类可视化，根据每个类质心的预期尝试次数分别标记为“容易”、“中等”和“困难”。因此，我们成功地找到了一种方法，将每个单词根据它们被报告的解决方式分配到三个难度中。

7.2 难度分类

将所有的词特征量化为 12 个数值并赋值为难度等级后，就可以建立基于词属性的监督分类模型来预测难度等级。我们在这个任务中使用随机森林分类[2]。随机森林已被证明可以在大范围的数据上提供稳定的分类结果。该算法的动机是通过对训练集和特征空间进行 bagging 来减少方差。B 决策树是在一个采样上训练的

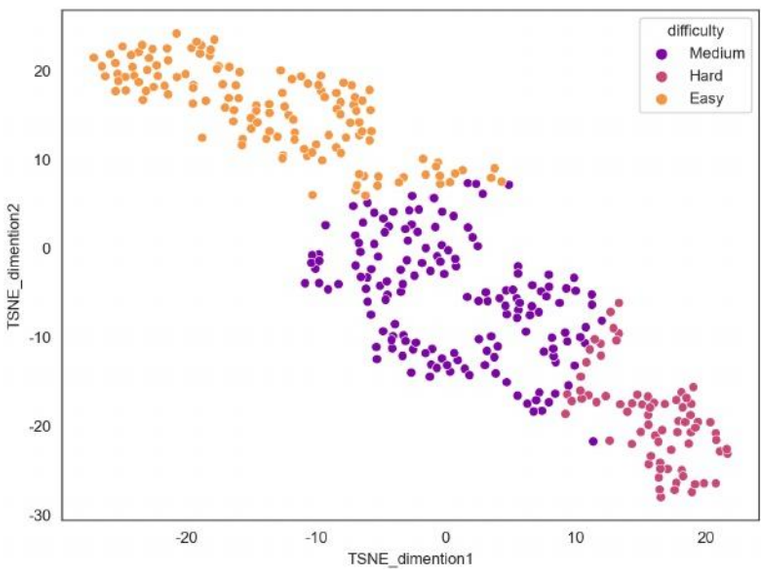

图 9:2 维投影的可视化 3 个集群

所有具有特定特征的数据集被屏蔽，并通过汇总每个决策树的所有B来产生分类结果。

$$\text{(train)} \qquad f_b = \arg\min_{f_b} L(f_b(X_b), Y_b) \qquad (b = 1, \ldots, B) \tag{18}$$

$$\text{(predict)} \qquad \hat{f} = \frac{1}{B}\sum_{b=1}^{B} f_b(X) \tag{19}$$

所有记录被分成 80%的训练集和 20%的测试集。对模型进行训练后，测试数据上的分类度量如表 8 所示。看起来，每个类别之间的分类得分总体上是均匀的，准确率(0.68)很高，这意味着模型可以成功地掌握单词属性与其难度之间的关系。

	precision	recall	f1-score	support number
easy	0.63	0.70	0.67	27
medium	0.77	0.68	0.72	34
hard	0.58	0.64	0.61	11
accuracy			0.68	72
weighted average	0.69	0.69	0.68	72

表 8:分类度量。

除了总准确率外，还计算了每个类别的精度、召回率和 f1-score，并按权重平均。

我们的模型还提供了每个特征如何影响谜题难度的见解。图 10 显示了每个特征与单词难度的相关性。这个因子是通过给每个难度级别赋值(容易=0, 中等=1, 困难=2)并计算每个特征的协方差矩阵得到的。

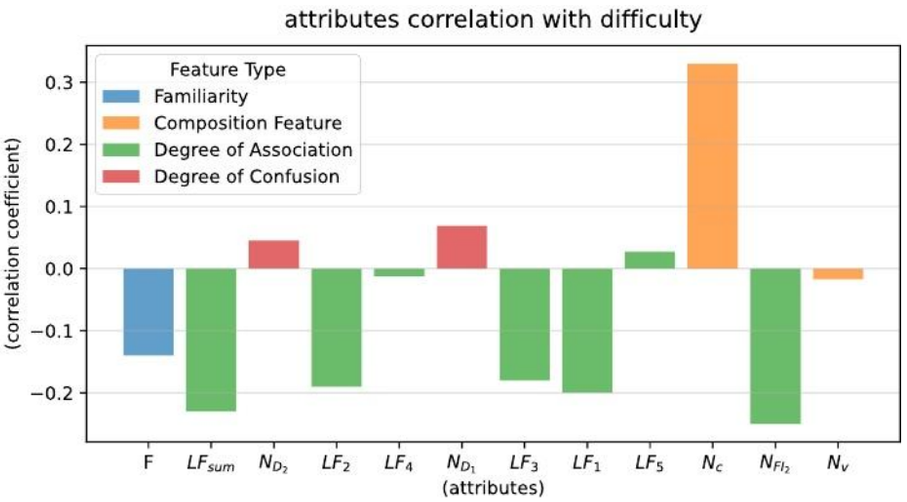

图 10:每个特征与难度的相关性，按其在模型中的重要性排序

这个模式非常明显:如果一个词使用频率更高(熟悉度)，或者可以很容易地与另一个词联系起来，那么它就会与难度呈负相关，即更容易猜测。此外，如果它经常与另一个单词混淆，它就会变得更加困难。组合特征的影响也可以解释:尝试多个相同字母的单词相对来说没有那么本能，人类玩家更喜欢先破解元音来解决 Wordle，这使得有多个元音的单词更容易。因此，相同字母的数量与难度成正相关，元音的数量与难度成负相关。

根据我们的模型，“EERIE”被归类为难答。该类别在测试集上的精度为 0.58。

7.3 灵敏度分析

敏感性分析确定了在给定的一组假设下，自变量的变化如何影响特定的因变量结果，通过这种方式，我们可以检验结果的稳健性。为了测试我们的模型的结果是否对输入参数的变化敏感，我们需要进行敏感性分析。在实际的数据计算中，由于四舍五入的原因，数据中的百分比之和可能不会达到 100%。为了分析我们的聚类模型在原始数据不准确时的稳定性，我们使用敏感性分析来评估模型。为了模拟舍入造成的精度损失，我们在数据中加入了一个随机扰动i项$\varepsilon$to。我们假设随机扰动在原始数据的 5%范围内随机波动。我们将加入随机扰动后的数据定义为:

$$p'_{ij} = p_{ij} \times \varepsilon_i \qquad \varepsilon_i \sim U(0.95, 1.05) \tag{20}$$

然后，我们使用新数据进行聚类。通过比较有随机干扰和没有随机干扰的聚类结果，我们发现有 3.34%的聚类结果发生了变化，这表明聚类结果没有受到数据误差的严重影响，我们的模型是相对稳定的。

	Easy	Medium	Hard
without stochastic disturbance	133	153	73
with stochastic disturbance	132	159	68

表 9:有无干扰的聚类结果

8 任务 5:其他特征

8.1 报告结果数量的波动

与在硬模式下所报告的分数所占百分比的变化类似，报告结果的数量总是围绕整体趋势上下波动。由于人们总是更愿意分享自己更好的分数，我们推测这种波动可能与单词的难度有关。为了探究两者之间的关系，我们将第 4 部分中 T2PL 模型拟合的结果从报告结果的实际数量中减去作为因变量，并将第 6 部分中聚类得出的难度水平作为自变量，建立线性回归模型。

得到的结果如下:

Easy	Medium	Hard	R^2
6589.5***	1087.5	−2649.3***	0.09
(1143.7)	(1371.6)	(781.3)	

表 10:单词难度对报告结果数量的影响

注:因为难度等级是一个分类变量，所以在回归中为它设置了哑变量。

回归结果表明，词的难易程度可以在一定程度上解释报道结果数的变化。具体来说，当单词为简单时，报告结果的数量将比拟合结果高 6589.5，当单词为中等难度时，报告结果的数量将比拟合结果高 1087.5，而当单词为困难时，报告结果的数量将比拟合结果少 2649.3。我们修正了 T2PL 模型预测的区间。因为 EERIE 这个词比较硬，所以我们同时在原始区间的端点值上减去 2649，得到修正后的预测区间[15111,18327]。

8.2 单词难度对困难模式报告百分比的影响

结合第 5 部分和第 7 部分的发现，我们可以发现影响在困难模式下打出的分数百分比的单词属性恰好是影响单词难度的重要属性，因此我们有理由相信在困难模式下打出的分数百分比与单词难度之间存在某种关系。与 7.1 类似，我们通过创建一个线性回归方程来探索这种关系(在前一天添加HP作为控制变量，之后的每个回归都是一样的)。

得到的结果如下:

Easy	Hard
−0.0021938***	0.0021471***
(0.0005296)	(0.0006389)

表 11:单词难度对报告的困难模式结果百分比的影响

注:本节将中等难度的虚拟变量替换为常数项，因此中位数的系数不再报告。

令人惊讶的是，在困难模式下玩的分数百分比在单词困难时上升，而在单词简单时下降。为了调查这种现象的原因，我们运行了另外两个回归，其中报告的在正常模式(N-H)下玩的分数和报告的在困难模式下玩的分数作为因变量。

结果如下所示:

	Easy	Hard
$N-H$	6374^{***}	-5125^{***}
	(1430)	(1718)
H	166.8	229.0
	(90.27)	(158.5)

表 12:单词难度对运行结果的影响:$H$and$H$

可以发现，在正常模式下，当单词是困难的时候，报告的分数数量明显减少，相反，当单词是容易的时候，报告的分数数量明显增加。而单词难度对困难模式下的得分没有显著影响。因此，我们可以分析这一奇怪现象的原因:选择困难模式的玩家可能有更高的猜词水平，更享受游戏，所以即使单词难度增加，他们也能保持更稳定的猜对正确单词的水平。更进一步说，即使猜不出单词，他们也渴望分享自己的游戏结果。但在正常模式下，当玩家猜不出单词时，他们可能不会分享游戏结果。因此，当单词比较困难时，在困难模式下玩的分数百分比会增加。

9 长处与短处

9.1 优势

1. 评价指标的选择科学易懂。我们考虑了《世界大战的游戏特点，构建了四类指标，可以全面反映单词的拼写结构和人们对它的联想程度。

2. 该模型的构建充分考虑了现实情况，具有良好的可解释性。考虑到玩家对《世界大战》的忠诚度，对 T2PL 模型进行了改进，在 SIR 模型的基础上，将球员分为两类:普通球员和忠诚球员。结果证明，这种改进使模型能够很好地模拟和解释报告结果数量的变化趋势。P&S 模型在模拟参与者猜词策略的基础上，设定了不同尝试次数的参与者的分享概率，很好地模拟了人的心理特征。

3. 考虑很全面。在探索报告结果的数量和报告分数在硬模式下玩的百分比的趋势时，我们不仅分析了它们的整体趋势，还分析了它们围绕整体趋势的波动。进一步，我们解释了波动形成的原因。基于这一分析的结果，我们修正了第一个问题的预测区间。

4. 这些模型相互联系，相互印证。我们构建的反映单词属性的指标被应用到第 5 部分和第 7 部分的分析中。根据难度对 P&S 模型预测的 EERIE 分布进行聚类，结果与第 7 部分分类器的结果一致。

5. 避免了过拟合问题。考虑到数据量不是很大，我们没有使用包含大量参数的模型来避免过拟合问题。

9.2 缺点

1. 模型和现实还是有差距的。不同的人会用不同的策略猜词，不同的人有不同的词汇量，但我们在 P&S 模型中没有充分考虑。

2. 分类模型的准确率不够高。第 7 部分的分类器准确率只有 68%，并不是特别好，可能是因为我们在构建指标时省略了单词的一些特征。

9.3 进一步的讨论

9.3.1 模型改进

考虑到不同的玩家有不同的策略和词汇，我们可以用不同的算法来模拟不同玩家的策略，用不同的词汇来模拟不同玩家的词汇，最后用优化算法计算出每种类型的玩家在整体玩家中所占的比例。

9.3.2 模型扩展

我们构建的 T2PL 模型可以很好地模拟世界的受欢迎程度，解释玩家结构的变化。通过修改一些参数，这个模型可以应用到其他游戏中。

References

[1] Benton J Anderson and Jesse G Meyer. Finding the optimal human strategy for wordle using max imum correct letter probabilities and reinforcement learning. arXiv preprint arXiv:2202.00557, 2022.

[2] Leo Breiman. Random forests. Machine learning, 45:5–32, 2001.

[3] John A Hartigan and Manchek A Wong. Algorithm as 136: A k-means clustering algorithm. Journal of the royal statistical society. series c (applied statistics), 28(1):100–108, 1979.

[4] Peter G Jensen, Kim G Larsen, and Marius Mikuˇcionis. Playing wordle with uppaal stratego. In A Journey from Process Algebra via Timed Automata to Model Learning: Essays Dedicated to Frits Vaandrager on the Occasion of His 60th Birthday, pages 283–305. Springer, 2022.

[5] Xian Jiang, Jing Hua, and Yimin Li. Dynamic research on game communication. Mathematic in Practice and Theory, (277-283), 2016.

[6] Ujjwal Maulik and Sanghamitra Bandyopadhyay. Performance evaluation of some clustering algorithms and validity indices. IEEE Transactions on pattern analysis and machine intelligence, 24(12):1650–1654, 2002.

[7] Douglas A Reynolds et al. Gaussian mixture models. Encyclopedia of biometrics, 741(659-663), 2009.

[8] Peter J. Rousseeuw. Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. Journal of Computational and Applied Mathematics, 20:53–65, 1987.

[9] David Smith, Lang Moore, et al. The sir model for spread of disease-the differential equation model. Convergence, 2004.

[10] Daniel Victor. Wordle is a love story. The New York Times, 2022.

[11] Wikipedia contributors. Wordle Wikipedia, the free encyclopedia. https://en.wikipedia. org/w/index.php?title=Wordle&oldid=1139735175, 2023.

信

亲爱的先生/女士:

大家好，我们是一群世界狂热者。我们每天都会在你的应用中发现新的《世界》谜题。这款游戏确实给我们的日常生活带来了很多乐趣，我们希望做些事情来改进这款游戏。因此，我们正在为2022年所有推特世界报告建立一个数学模型。我们希望我们的分析可以帮助你的团队改进世界。

因为《世界》的规则是固定的，所以在游戏中最具挑战性的部分可能就是选择一个合适的词。但我们怎么知道一个词是否足够合适呢?是常见好还是罕见好?更根本的是，谜题的难度应该有多大?要回答这些问题，我们需要考虑四个重要因素之间的复杂关系:单词特征、猜测次数分布、谜题难度和玩家数量。我们可以建立模型来探索它们之间的关系。

首先，我们从每个单词中提取12个指标，测量其熟悉度(使用频率)、关联程度、混淆程度和单词组成特征，包括不同单词如何影响猜测的所有维度。

接下来，我们用一个名为“目标-二-玩家-丢失”(T2PL)的动态系统对总报告的总体趋势进行建模，将玩家分为一般和忠诚两类。该模型可以预测未来的报告数量(例如，3月1日的报告预计将落在[17760,20976]之间)。它还可以解释玩家结构，表明大多数《世界大战》玩家现在都是忠实玩家。

然后我们尝试用仿真算法来预测解的分布。为了改善结果，考虑到人们在获得不同分数时分享结果的可能性，重新调整分布，因此称为P&S (play and Shared)模型。根据P&S模型，对于任何单词(例如“eerie”)，其分布预计为(0,0,9%，29%，45%，14%，3%)。

需要定义每个谜题的难度。我们对报告分布运行k-means，并将其分为3种类型，分别标记为简单、中等和困难。然后我们将单词属性分为三个难度级别。我们知道每个单词的难度(上面使用的“eerie”的例子当然被标记为困难)，我们从相关分析中知道指标是如何影响单词难度的。

最后一块拼图是单词难度和报告数量之间的关系。使用线性回归，我们发现更简单的世界大战鼓励更多的分享推文。我们还发现单词属性会影响困难模式玩家的比例。虽然这有点违反直觉，但我们最终解释说，普通模式玩家不太可能分享困难的单词，而困难模式玩家则不会。

对于你们的团队，我们想知道我们的模型是否可以用来预测玩家在每个谜题发布之前的反应，并估计未来玩家的数量。此外，一个困难的单词可能不是一个合适的选择，因为它可能会阻止中级玩家分享他们的结果。我们推荐那些只有一两个指标使其变得困难的单词，所以它不属于困难的类别。如果所选择的使单词变得困难的指标每天都在变化(例如，某天单词频率较低，另一天元音较少)，这甚至可能给日常的《世界大战》玩家带来新鲜感。我们很乐意与你的团队进一步讨论这个游戏。我们期待着在未来能够玩到更棒的世界大战谜题。