arXiv:2301.01764v2 [cs.CL] 5 Jan 2023

TSAR-2022 共享任务中的海德堡大学：词汇简化是否只需算力？

DennisAumiller 和 **MichaelGertz**海德堡大学计算机科学研究所 {aumiller, gertz}@informatik.uni-heidelberg.de

摘要

以往词汇简化领域的最先进模型由包含多个组件的复杂流水线构成，而每个组件都需要深厚的技术知识和精细调优的交互，才能充分发挥其潜力。作为替代方案，我们描述了一种基于提示 GPT-3 响应的极其简单的流水线；在训练实例较少的设置下，该方法以明显优势击败了竞争方法。我们提交至 TSAR-2022 共享任务英语赛道的最佳表现系统，由六种不同提示模板组成的“集成”构成，这些模板具有不同层次的上下文。作为一项后续公布的结果，我们进一步详细介绍了一种语言迁移技术，使其能够在英语以外的语言中进行简化。在应用于西班牙语和葡萄牙语子集时，只需对原始提示进行少量修改，我们便取得了最先进的结果。除详细说明实现和设置外，本文其余部分将讨论提示的具体特点及其对未来工作的影响。实验代码可在线获取。[1]

1 引言

近年来，机器学习（ML）研究的进展主要得益于计算预算的增加。Richard Sutton 提出了“苦涩的教训”这一观点：计算能力的提升最终将超越精心设计的解决方案（Sutton，2019年）。近来，在通用架构上不断提升计算能力，也已在自然语言处理（NLP）领域取得了巨大成功（Vaswani 等人，2017；Wei 等人，2022年）。尤其是，超大型语言模型（vLLMs）涌现出的能力，使其能够应对各种任务，而这些任务中只有少量（如果有的话）样本带有标注，并且无需进一步微调任务专用数据。

与现代词汇简化系统中复杂的流程形成鲜明对比（Ferrés等人，2017；Qiang等人，2020；Štajner 等人，2022年），我们提出了一种简单的方法：利用基于视觉大语言模型的少样本提示，并辅以简化方面的基本指令。尽管整个方法总共只使用了四个人工标注的实例，却取得了好得令人惊讶的结果。我们在 TSAR-2022 共享任务（Saggion 等人，2022年）中展示了我们的结果。该任务评估三种语言（英语、西班牙语和葡萄牙语）的词汇简化系统；每种语言均提供了十个已标注实例和约 350 个未标注测试样本。对于英语子集，官方结果将我们的模型评为表现最佳的提交，表明这种方法可能是苦涩的教训的又一例证。虽然初步发现确实令人鼓舞，但我们希望仔细评估测试集中的错误实例，以分析潜在问题，并进一步详述我们手工设计提示的一些经验。我们也注意到，复现（和部署）基于视觉大语言模型的系统存在技术挑战，尤其是合适的模型所需的计算资源超出了传统预算。

2 基于提示的词汇简化

随着 GPT-3 语言模型（Brown 等人，2020）的公开发布，OpenAI 开始推出一系列如今已可使用的视觉大语言模型，用于通用文本生成（Thoppilan 等人，2022年；BigScience，2022年；Zhang 等人，2022年）。在这些模型中，可以观察到一种普遍趋势：模型参数规模不断扩大到某个特定规模以上，同时底层架构设计仍与现有的小型模型保持相近。凭借零样本迁移能力，这类模型在低资源任务中也越来越有吸引力；它们往往能够以自然语言回答问题，取得一定程度令人满意的结果

[1] https://github.com/dennlinger/TSAR-2022-Shared-Task

。人们经常使用特定的模板模式（所谓的提示）来引导模型生成特定的理想输出或答案格式，而无需专门使用带标签的示例进行训练。

利用这一范式转变，我们尝试向 OpenAI 当时可用的最大模型 text-davinici-002 提出不同的提示，该模型共有 176B 个参数。我们首先在零样本设置下使用单个提示模板，为共享任务生成预测；随后，我们通过合并不同提示模板生成的预测，进一步提升这些结果。

2.1 Run1：零样本预测

检查所提供的试验数据后，我们注意到，所提供的样例句中，各项简化操作所需的上下文信息截然不同。有些实例只需查找同义词即可解决（例如，“compulsory”和“必须的”），另一些则需要更细致地考察上下文句子（例如，将“disguised”替换为“dressed”）。为避免通过在提示模板中提供样例而使系统预测产生偏差，我们采用一个完全基于单次零样本查询的基线：向模型提供上下文句子并标出复杂词，请模型根据给定上下文，为该复杂词提供十个简化后的同义词。由于没有向模型提供额外知识，这种零样本上下文查询也为该任务设置提供了一个合理的下限。在零样本设置下仅提供少量上下文的另一项优势是计算成本更低，相关内容将在第3.4节进一步讨论。

2.2 预测筛选

模型给出的建议以自由文本预测的形式返回，通常是逗号分隔的列表或枚举形式。要完成共享任务，必须额外将输出预测解析为结构更明确的排序后的预测，而解析方式因所用模型而异。根据我们的经验，无法预期模型会遵循明确的模式，即使模板结构固定，结果似乎也具有非确定性。此外，我们还使用一组简单的过滤器来确保预测质量，详情见附录 C。模型建议按排序顺序纳入考量，未使用预测置信度分数或类似信息来重新排序单提示预测。

2.3 运行 2：集成预测

检查首次运行的结果后，我们注意到，在某些情况下，由于筛选，预测几乎被全部剔除。与此同时，我们此前已经发现，改变提示模板或上下文设置时，系统生成的内容会有很强的变异性。因此，我们采用了多个不同提示模板生成的预测集成，以拓宽可能生成内容的范围，并确保筛选步骤后至少保留一定数量的建议。

2.3.1 提示变体

具体提示详见表3。采用的变体可分为带上下文（提供上下文句子）或不带上下文（仅根据复杂词生成同义词）。此外，不同提示还包含零到两个取自试验数据的示例及其预期输出。这可以视为一种少样本设置，在这种设置下，模型会看到正确答案在此特定任务中的示例。我们还调整了生成温度；温度越高，生成更具创造性（但不一定正确）的预测的可能性越大，从而得到更多样化的候选集。

2.3.2 合并预测

对于六个提示中的每一个 p，我们都要求模型提出十个备选简化表达式S_p，并按照与 Run1 中 singlepromptsystem 完全相同的规则对其进行筛选。为了合并并重新排序建议 s，我们为每个不同的预测 $s \in \bigcup_p S_p$赋予一个组合得分 V：

$$V(s) = \sum_p \max(5.5 - 0.5 \cdot \text{rank}_{S_p}(s), 0), \quad (1)$$

其中，$\text{rank}_{S_p}(s)$ 是建议 s 在提示 p生成的排名中的位置。若$s \notin S_p$，则设 $\text{rank}_{S_p}(s) = \infty$。缩放参数是任意选取的，可根据每个提示预期产生的建议数量进行调整。我们估计，性能提升最大的一部分仅仅来自筛选后提供了足够多的预测。其次，我们发现最靠前的预测槽位表现更加稳定，从而尤其提升了集成方法的 @1 性能。

Run	ACC@1	Acc@k@Top1			MAP@k			Potential@k		
		$k=1$	$k=2$	$k=3$	$k=3$	$k=5$	$k=10$	$k=3$	$k=5$	$k=10$
集成（我们的方法）	**0.8096**	**0.4289**	**0.6112**	**0.6863**	**0.5834**	**0.4491**	**0.2812**	**0.9624**	**0.9812**	**0.9946**
单体（我们的方法）	0.7721	0.4262	0.5335	0.5710	0.5090	0.3653	0.2092	0.8900	0.9302	0.9436
MANTIS-1	0.6568	0.319	0.4504	0.5388	0.473	0.3599	0.2193	0.8766	0.9463	0.9785
UoM&MMU-1	0.6353	0.2895	0.4530	0.5308	0.4244	0.3173	0.1951	0.8739	0.9115	0.9490
LSBert	0.5978	0.3029	0.4450	0.5308	0.4079	0.2957	0.1755	0.8230	0.8766	0.9463
TUNER	0.3404	0.1420	0.1689	0.1823	0.1706	0.1087	0.0546	0.4343	0.4450	0.4450

表1：TSAR-2022 共享任务英语测试集上的结果，按*ACC@1* 分数排名。列出的是我们自己的结果（集成 和单体）、表现最佳的两个竞争系统（*MANTIS* 和*UoM&MMU*），以及提供的基线系统（*LSBert* (Qiang等人，2020）和 TUNER（Ferrés等人，2017）。

3 结果与局限性

3.1 英语结果

在官方运行中，我们最初只提交了英语子集的预测；表格中可以看到部分结果1。虽然零样本单提示运行在大多数指标上的结果始终更好，但在候选集较大时并非优于所有系统；例如，Potential@10 低于竞争方法（包括 LSBert 基线）的结果。我们认为这是此前提到的预测筛选问题所致；采用所提出的集成方法后，尤其在更大的k 上，可以看到相应的改进。这里，Potential@10 分数表明，除了三个样本之外，所有样本中都至少有一个可行的预测。

3.2 西班牙语和葡萄牙语结果

鉴于英语子集上的结果出乎意料地好，我们决定将实验扩展到西班牙语和葡萄牙语赛道。将提示迁移到西班牙语或葡萄牙语出乎意料地简单。我们将提示改为：“根据上述上下文，列出十个更容易理解的西班牙语词语，以替换‘complex_word’。”（粗体高亮表示改动）。

如果不进行这种调整，返回的建议通常会是英语，这或许为未来工作中挖掘跨语言简化提供了一个颇具吸引力的机会。明确添加输出语言，可以确保建议符合预期结果。对于葡萄牙语，可以相应地调整提示。

我们的方法在共享任务中也优于所有其他提交的方法；结果对比见附录中的表4 和5。值得注意的是，葡萄牙语的预测效果略好，这与直觉相悖，因为西班牙语通常是一种在多语言语料库中占比很高的语言。我们猜测，与西班牙语中的多词表达相比，葡萄牙语中的同义词措辞更加直白，这可能是造成这一现象的原因。

3.3 错误分析

与序列到序列任务的常见做法一样，设计一种以语言模型为核心的方法时，需要考虑由此产生的特定挑战。我们详细说明了预测中遇到的一些错误，而这些错误不太可能出现在设计更为严格的流程中。表2中列出了具体失败案例。

不稳定的提示 其中一个主要挑战，尤其是在零样本提示设置中，是即使提示模板只有细微变化，结果也会出现不合理的波动。例如，从提示模板中删除明确标注的上下文：、问题：和答案： 后，模型往往会预测出少于要求的十个答案。计算预算方面的实际限制也意味着我们无法保证这些提示能产生尽可能好的结果；鉴于结果存在波动，要全面了解哪种提示“最好”，就应比较多次运行。

上下文缺失 对于上下文较长（或以隐晦方式强加）的实例，模型可能无法正确识别这些提示线索。在表 2中可以看到，尽管原句通过“飞行”一词明确提供了上下文线索，模型却将“碰撞”替换成了某种特定的交通方式，例如“车祸”。

强制使用语言 尽管迁移到西班牙语和葡萄牙语总体上取得了成功，但模型似乎仍无法在所有样本中始终保持使用目标语言。

错误类型	上下文（粗体显示复杂词）	模型预测
上下文缺失	#7-8 尽管有雾，据报道，其他航班 在**碰撞**发生前都已安全着陆。	车祸 ，火车事故 ，...
幻觉	幼虫长至约 120–130 毫米， 并在地下洞室中**化蛹**。	变成蛹，.....
语言	[...] 促成了**法国大革命的衰落**。	衰退、恶化.....

表2：系统预测中观察到的失败类别实例。

对于包含特别罕见的复杂术语的实例，预测有时仍会以英语给出，尽管提示明确要求返回西班牙语/葡萄牙语结果。

幻觉 建议需要进行后过滤，主要是因为回复中会自发出现幻觉。虽然视觉大语言模型中的幻觉较少涉及无效词汇，但我们观察到一些情况：系统选择了不必要的多词建议，而非简单的单词同义表达；或者生成了随机的词形变化（例如在不定式形式前额外加上“to”）。

与语言强制相关的问题类似，这种情况在特别复杂的词汇中更常见；从这个意义上说，系统反而会在最需要简化的实例上失效。不过，我们注意到，有些生成的多词表达实际上更有助于理解，尽管生成结果并未与预期输出完全吻合。

3.4 计算限制

在实践中运行 vLLM，即便仅用于推理，也并非易事，并且需要远超许多公共机构硬件预算的计算资源。对于可公开获取检查点的最大规模模型[2],，总共需要约**325GB GPU显存**，前提是采用 `bfloat16` 或类似精度进行高效存储。常见的替代方案是通过（通常收费的）API 获取预测，本研究采用的正是这种方式。尤其是集成模型，每个样本都要向 API 单独发出六次请求，这会进一步推高单次预测的净成本。为了说明总费用，我们为整个测试集中的 373 个英语样本计算预测，总共花费略高于 7 美元，折合每个样本约 1000 个 token，按当前 OpenAI定价方案约为 0.02 美元。[3] 西班牙语子集的处理以及针对语言的提示词开发，总费用约为 10 美元，主要是因为样本上下文更长。葡萄牙语处理费用约为 6.50 美元。单体提示词方案更便宜，费用约为总成本的 1/6；即便如此，持续部署的模型仍需配备足够的预算。除了费用问题，此类大规模部署还需考虑环境影响（Lacoste等人，2019年）。

4 结论与未来工作

利用视觉大语言模型（vLLMs）的提示式响应似乎是词汇简化的一条有前景的方向；尤其是在预先识别复杂词汇的受限设置中，即便标注训练数据极其有限，模型仍表现出色。尽管这种方法也为多语和跨语言方法提供了有前景的方向，并有望在其他语言中取得最先进的结果，但每个样本实例所需的计算量却高得令人望而却步。因此，若能将其应用于资源受限系统，将预测能力置于略有不同的视角下考察，会是一个值得探索的方向。最后，我们也再次注意到神经语言模型的不稳定性：即使输入相似，不同样本的质量也可能大相径庭，甚至会出现性能彻底崩溃的情况。

未来的工作中，我们正在考虑从视觉大语言模型中生成静态资源的方法（Schickand Schütze，2021），这或许只需一次性投入数据集，随后即可将其用作成本更低的系统的训练数据。探索用于自动搜索合适提示模板的提示调优方法，也将大幅加快特定领域应用的开发过程（Lester et al.，2021）。

[2]撰写本文时，这将是 176B Bloom 模型（BigScience，2022年），其参数数量与 OpenAI 的 davinci-002 模型相近。

[3]https://openai.com/api/pricing/，最后访问日期：2022-10-01

参考文献

BigScience 研讨会。2022年。 BLOOM（修订版 4ab0472）。

TomBrown, Benjamin Mann, Nick Ryder, MelanieSubbiah, Jared DKaplan, Prafulla Dhariwal,Arvind Neelakantan, PranavShyam, Girish Sastry,AmandaAskell, Sandhini Agarwal, Ariel Herbert-Voss, GretchenKrueger, TomHenighan, RewonChild, Aditya Ramesh, DanielZiegler, Jeffrey Wu,Clemens Winter, ChrisHesse, Mark Chen, EricSigler, Mateusz Litwin, ScottGray, Benjamin Chess,Jack Clark, ChristopherBerner, Sam McCandlish,AlecRadford, Ilya Sutskever, and Dario Amodei.2020. 语言模型是少样本学习器。载于神经信息处理系统进展，第 33 卷，第 1877–1901 页。Curran Associates 公司。

DanielFerrés, Horacio Saggion, and XavierGómezGuinovart. 2017. 面向主要伊比利亚-罗曼语族语言的自适应词汇简化架构。载于首届构建语言学可泛化 *NLP* 系统研讨会论文集，第 40–47 页，丹麦哥本哈根。计算语言学协会。

Alexandre Lacoste、Alexandra Luccioni、Victor Schmidt 和 Thomas Dandres。2019年。量化机器学习的碳排放。*CoRR*，abs/1910.09700。

Brian Lester、Rami Al-Rfou 和 Noah Constant。2021。参数高效提示调优的规模效应。载于 *2021*年自然语言处理实证方法会议论文集，第 3045–3059 页，线上会议及多米尼加共和国蓬塔卡纳。计算语言学协会。

Jipeng Qiang、Yun Li、Zhu Yi、Yunhao Yuan 和 Xin-dong Wu。2020。利用预训练编码器进行词汇简化。第三十四届 *AAAI* 人工智能大会，第 8649–8656 页。

Horacio Saggion、Sanja Štajner、Daniel Ferrés、Kim Cheng Sheang、Matthew Shardlow、Kai North 和 Marcos Zampieri。2022。TSAR-2022 多语言词汇简化共享任务研究结果。载于与 *EMNLP 2022* 联合举办的 *TSAR* 研讨会论文集。

Timo Schick 和 Hinrich Schütze。2021年。使用预训练语言模型生成数据集。载于*2021*年自然语言处理经验方法会议论文集，第6943–6951页，线上及多米尼加共和国蓬塔卡纳。计算语言学协会。

Richard Sutton。2019年。苦涩的教训。未完成的想法（博客），13:12。

Romal Thoppilan、Daniel De Freitas、Jamie Hall、Noam Shazeer、Apoorv Kulshreshtha、Heng-Tze Cheng、AliciaJin、Taylor Bos、Leslie Baker、Yu Du, YaGuang Li、Hongrae Lee、Huaixiu Steven Zheng、Amin Ghafouri、Marcelo Menegali、Yanping Huang、Maxim Krikun、Dmitry Lepikhin、James Qin、Dehao Chen、YuanzhongXu、ZhifengChen、Adam Roberts、Maarten Bosma、Yanqi Zhou、Chung-Ching Chang、Igor Krivokon、Will Rusch、Marc Pickett、Kath- leen S. Meier-Hellstern、Meredith Ringel Morris、Tulsee Doshi、Renelito Delos Santos、Toju Duke、Johnny Soraker、Ben Zevenbergen、Vinodkumar Prabhakaran、Mark Diaz、Ben Hutchinson、Kristen Olson、Alejandra Molina、Erin Hoffman-John、Josh Lee、Lora Aroyo、Ravi Rajakumar、Alena Butryna、Matthew Lamm、Viktoriya Kuzmina、Joe Fenton、Aaron Cohen、Rachel Bernstein、Ray Kurzweil、Blaise Aguera-Arcas、Claire Cui、Marian Croak、Ed H. Chi 和 Quoc Le。2022年。LaMDA：用于对话应用的语言模型。*CoRR*，abs/2201.08239。

Ashish Vaswani、Noam Shazeer、Niki Parmar、Jakob Uszkoreit、Llion Jones、Aidan N. Gomez、Lukasz Kaiser 和 Illia Polosukhin。2017。Attention I s All You Need。载于神经信息处理系统进展第*30*卷：*2017*年神经信息处理系统年度会议，*2017*年*12*月*4*日至*9*日，美国加利福尼亚州长滩，第5998–6008页。

Sanja Štajner、Daniel Ferrés、Matthew Shardlow、Kai North、Marcos Zampieri 和 Horacio Saggion。2022年。英语、葡萄牙语和西班牙语词汇简化基准。人工智能前沿，5。

Jason Wei、Yi Tay、Rishi Bommasani、Colin Raf- fel、Barret Zoph、Sebastian Borgeaud、Dani Yo- gata ma、Maarten Bosma、Denny Zhou、Donald Met- zler、EdH. Chi、TatsunoriHashimoto、Oriol Vinyals、Percy Liang、Jeff Dean 和 William Fedus。2022年。大语言模型的涌现能力。 *CoRR*，abs/2206.07682。

Susan Zhang、Stephen Roller、Naman Goyal、Mikel Artetxe、Moya Chen、Shuohui Chen、Christopher Dewan、Mona T. Diab、Xian Li、Xi Victoria Lin、Todor Mihaylov、Myle Ott、Sam Shleifer、Kurt Shus- ter、Daniel Simig、Punit Singh Koura、Anjali Srid- har、Tianlu Wang 和 Luke Zettlemoyer。2022年。OPT：开放式预训练变换器语言模型。*CoRR*，abs/2205.01068。

A 提示模板

表3 列出了本次提交中使用的确切提示模板。值得注意的是，带上下文的零样本提示包含了两次，但生成温度不同；这样做可以提高保留优质候选项的可能性。对于少样本提示，我们从相应语言先前发布的试验集中抽取样本。对于标注者提供的不同建议少于 10 条的情况，我们根据自己的判断手动扩充示例列表，使结果恰好达到十条。对于提供的建议更多的情况，我们只选取出现频率最高的十条。这样做的原因是，在初步测试中，GPT-3 否则往往会返回数量不一致的建议。西班牙语和葡萄牙语运行所用的确切提示可在我们的代码库中找到。

B 超参数

我们的实验使用 OpenAI Python 软件包[4] 0.23.0版本。生成文本时使用函数 openai.Completion.create()，其中大多数超参数在所有提示中保持不变。下面明确列出与各自默认值不同的超参数。

1. model="text-davinci-002"，这是目前可用于文本补全的最新、最大的模型。

2. max_tokens=256，以确保为生成的输出留出充足空间。实际上，大多数补全文本都远未达到此上限。

3. frequency_penalty=0.5，以及 presence_penalty=0.3，两者共同对已出现的令牌及令牌重复进行惩罚。这些值远低于最大值（取值范围为 -2 至 2），因为单个子词令牌确实可能在多个（有效的）预测中反复出现。更详细的计算方法可参阅 OpenAI 的文档。[5]

除重复惩罚之外，对生成结果影响最大的参数选择是采样温度。总体而言，我们采取比默认值（温度=1.0）更为审慎的方法，但会在集成提示中调整温度，以确保整体结果集更加多样化。我们在表3中列出了所使用的温度。 带上下文的零样本在集成中使用了两次：一次采用更保守的温度，另一次采用更“创造性”（更高）的温度。对于单独的提示运行，我们使用保守的带上下文的零样本变体。

C 后过滤操作

鉴于语言模型预测结果的不确定性，我们采用一系列后过滤步骤来确保输出质量。这包括删除换行符/空格/标点符号（\n :;.?!）、转换为小写、移除不定式形式（在某些情况下，我们观察到预测结果的形式是“to deploy”，而非简单的“deploy”），以及移除恒等预测（例如，预测结果与原始复杂词相同）并对建议去重。此外，我们注意到，在某些情况下，生成的同义词更像是“描述”，而非真正的同义表达（例如，将“people that are crazy”作为“maniacs”的建议）。鉴于所提供数据的性质，我们移除了过长的多词表达（英语中，任何超过两个单词的建议；西班牙语和葡萄牙语中，任何单个表达超过三个单词的建议）。

[4] https://github.com/openai/openai-python [5] https://beta.openai.com/docs/api-reference/ 参数详情

提示类型	模板
带上下文的零样本 温度： 0.3（保守）， 0.8（创造性）	上下文：{context_sentence}\n 问题：根据以上上下文，列出十个替代词，用于“{complex_word}”的更易懂替代词。\n 答案：
单样本 /w 上下文 温度：0.5	上下文：一名当地目击者称，另一组袭击者伪装在布卡罩袍中——这种从头到脚的长袍由保守的阿富汗女性穿着——随后试图强行闯入院落。\n 问题：根据上述上下文，列出十个更容易理解的替代词来替换“伪装”。\n 答案：\n1. 隐藏的\n2. 穿着\n3. 隐藏的\n4. 伪装的\n5. 改变了的\n6. 覆盖的\n7. 戴面具的\n8. 无法辨认的\n9. 转变的\n10. 冒充的\n\n 上下文：{context_sentence}\n 问题：根据上述上下文，列出十个更容易理解的“{complex_word}”的替代词。\n 答案：
带上下文的双样本提示 温度：0.5	上下文：这促使军方部署了其最大的军舰，BRP Gregorio del Pilar，该舰最近从美国购得。\n 问题：根据上述上下文，列出十个替代词来替换更容易理解的“deploy”。\n 答案：\n1. 发送\n2. 发布\n3. 使用\n4. 安置\n5. 发出\n6. 雇用\n7. 延伸\n8. 发起\n9. 释放\n10. 组织\n\n 上下文：叔利亚每日死亡人数已减少，因为观察者人数有所增加，但很少有专家认为联合国计划能够全面成功。\n 问题：根据上述上下文，列出十个替代词更容易理解的“观察者”。\n 答案：\n1. 观看者\n2. 观众\n3. 听众\n4. 观看者\n5. 目击者\n6. 顾客\n7. 追随者\n8. 侦探\n9. 记者\n10. 旁观者\n\n 上下文：{context_sentence}\n 问题：根据上述上下文，列出十个更容易理解的“{complex_word}”替代词。\n 答案：
无上下文零样本 温度：0.7	请为以下单词提供十个简化同义词： {complex_word}
无上下文单样本 问题： 温度：0.6	为“强制的”找出十个更简单的词。\n 答案：\n1. 必须的\n2. 必需的\n3. 必不可少的\n4. 被迫的\n5. 重要的\n6. 必要的\n7. 义务性的\n8. 不可避免的\n9. 有约束力的\n10. 规定的\n\n 问题：为“{complex_word}”找出十个更简单的词。\n 答案：

表 3：用于查询 OpenAI 模型的英文提示模板及其对应的生成温度。只有明确写出的“\n”符号表示换行；为便于阅读而插入了可见的换行。单提示（运行 1）和集成运行（运行 2）中都只使用了最上方采用保守温度的提示模板。所有其他提示仅纳入集成提交。

Run	ACC@1	Acc@k@Top1			MAP@k			Potential@k		
		$k=1$	$k=2$	$k=3$	$k=3$	$k=5$	$k=10$	$k=3$	$k=5$	$k=10$
集成（我们的方法）	**0.6521**	**0.3505**	**0.5108**	**0.5788**	**0.4281**	**0.3239**	**0.1967**	**0.8206**	**0.8885**	**0.9402**
单体（我们的方法）	0.5706	0.3070	0.3967	0.4510	0.3526	0.2449	0.1376	0.6902	0.7146	0.7445
PresiUniv-1	0.3695	0.2038	0.2771	0.3288	0.2145	0.1499	0.0832	0.5842	0.6467	0.7255
UoM&MMU-3	0.3668	0.1603	0.2282	0.269	0.2128	0.1506	0.0899	0.5326	0.6005	0.6929
LSBert	0.2880	0.0951	0.1440	0.1820	0.1868	0.1346	0.0795	0.4945	0.6114	0.7472
TUNER	0.1195	0.0625	0.0788	0.0842	0.0575	0.0356	0.0184	0.144	0.1467	0.1494

表4：TSAR-2022 共享任务西班牙语测试集上的结果，按*ACC@1* 分数排序。列出我们自己的结果（集成 和 单体）、表现最好的两个参赛系统（*PresiUniv* 和*UoM&MMU*），以及提供的基线系统（*LSBert* (Qiang等人，2020）和 TUNER（Ferrés等人，2017）。

Run	ACC@1	Acc@k@Top1			MAP@k			Potential@k		
		$k=1$	$k=2$	$k=3$	$k=3$	$k=5$	$k=10$	$k=3$	$k=5$	$k=10$
集成（我们的方法）	**0.7700**	**0.4358**	**0.5347**	**0.6229**	**0.5014**	**0.3620**	**0.2167**	**0.9171**	**0.9491**	**0.9786**
单体（我们的方法）	0.6363	0.3716	0.4625	0.5160	0.4105	0.2889	0.1615	0.7860	0.8181	0.8422
GMU-WLV-1	0.4812	0.2540	0.3716	0.3957	0.2816	0.1966	0.1153	0.6871	0.7566	0.8395
Cental-1	0.3689	0.1737	0.2433	0.2673	0.1983	0.1344	0.0766	0.524	0.5641	0.6096
LSBert	0.3262	0.1577	0.2326	0.286	0.1904	0.1313	0.0775	0.4946	0.5802	0.6737
TUNER	0.2219	0.1336	0.1604	0.1604	0.1005	0.0623	0.0311	0.2673	0.2673	0.2673

表 5：TSAR-2022 共享任务葡萄牙语测试集上的结果，按*ACC@1* 得分排名。列出的是我们自己的结果（集成 和 单体）、表现最佳的两个参赛系统（*GMU-WLV* 和*Cental*），以及提供的基线系统（*LSBert* (Qiang等人，2020）和 TUNER（Ferrés等人，2017）。