Culturally Yours: A Reading Assistant for Cross-Cultural Content

Saurabh Kumar Pandey[1], Harshit Budhiraja[2], Sougata Saha[1], Monojit Choudhury[1]
[1]MBZUAI, [2]Wadhwani AI
[1]{saurabh.pandey, sougata.saha, monojit.choudhury}@mbzuai.ac.ae
[2]{hbudhi36@gmail.com}

Abstract

Users from diverse cultural backgrounds frequently face challenges in understanding content from various online sources written by people from different cultures. This paper presents Culturally Yours (CY), a first-of-its-kind cultural reading assistant tool designed to identify culture-specific items (CSIs) for users from varying cultural contexts. By leveraging principles of relevance feedback and using culture as a prior, our tool personalizes to the user's preferences based on their interaction with the tool. CY can use any LLM capable of reasoning with the user's cultural background in English-based prompts as the back-end. Using culture as part of the prompt, CY iteratively refines the prompt as the user keeps interacting with the system. In this demo, we use GPT-4o as the back-end. We also conducted a user study across 13 users from 8 different geographies. The results demonstrate CY's effectiveness in enhancing user engagement and personalization alongside comprehension of cross-cultural content. The tool can be accessed by following instructions on Github [1].

1 Introduction

With increasing digitization, people frequently encounter text from diverse sources that they find difficult to understand, often due to a lack of common ground between the writer of the text and the reader. For example, people unfamiliar with the Arabic culture might not understand the meaning of the dishes *"Machboos"* and *"Luqaimat"* from the review text *"The Machboos was perfectly spiced, and the Luqaimat was a real treat"*. Or, someone unfamiliar with the Western culture might not understand that *"golden arches"* refers to *"McDonald's"* in the text *"Let's go to the golden arches for a quick bite"*. Thus, communication can get hampered in cross-cultural contexts due to a lack of appropriate common ground between the interlocutors (Meyer, 2014; Korkut et al., 2018), which, in turn, can adversely impact a user in many scenarios that involve decision-making, such as from *user-reviews* on e-commerce platforms like Amazon and travel platforms like Booking.com.

To address this challenge in the cross-cultural understanding of online text, we have developed a cultural reading assistant, **Culturally Yours (CY)**, which acts as a cultural mediator and identifies exotic concepts from an unknown source culture to the user's culture. CY facilitates cross-cultural communication by globalizing local textual articles and enabling users to understand and engage with content they might otherwise struggle to interpret. Thus, it can help businesses improve user engagement and reach broader audiences across diverse cultural markets worldwide.

CY uses the principles of *relevance feedback* (Rui et al., 1998) from information retrieval, which involves iterative refinement of results using user feedback to improve the system's performance. Not understanding the preferences of new users, famously known as the *cold-start problem*, is a well-established issue in collaborative filtering (Hu et al., 2008) that hinders personalization. Using culture as a prior, CY efficiently ameliorates the cold-start problem and gradually adapts to a user's preference, incorporating multiple iterations of relevance feedback. Defining culture by demographic features, the tool initially highlights and explains certain portions of text that the user might find hard to understand due to the cross-cultural gap in common-ground. Users then provide feedback by deselecting the highlighted spans or highlighting new spans missed by the tool. Over time, with multiple such cycles of relevance feedback spanning texts from diverse domains, CY gradually understands and adapts to the user's preferences. Eventually, CY personalizes to users and helps them acquaint themselves with text from different cul-

[1] https://github.com/skp1999/CULTURALLY_YOURS

Proceedings of the 31st International Conference on Computational Linguistics: System Demonstrations, pages 208–216
January 19–24, 2025.

tures.

Using Large Language Models (LLMs) (Achiam et al., 2023; Dubey et al., 2024; Bubeck et al., 2023; Nori et al., 2023; Lai et al., 2023) as the underlying model, CY implements a prompt-based algorithm to identify and explain culture-specific items (CSIs) (Newmark, 2003) based on a user's cultural background and preferences. CSIs are cultural items that people from different backgrounds might not understand and be unfamiliar with. Initially, CY identifies the CSIs from any English text solely based on the user's demographics. Over time, as the user interacts with the tool, CY captures their preferences across diverse domains and algorithmically adjusts its prompt to better align the CSI identification and explication with the user's background and preferences. Currently, the tool uses GPT-4o[2] as the backend LLM, but can be replaced by any other LLM suitable for this task. We also experimented with three prompting-based algorithms and conducted a user study over 13 users to determine the best personalization algorithm for the backend. In summary, the main contributions of our work are as follows.

- We introduce CULTURALLY YOURS, a first-of-its-kind reading assistant tool, to help people from different cultural backgrounds understand online text from unknown cultures. Such a tool facilitates cross-cultural communication and promotes globalizing local content.
- We propose and experiment with three strategies for optimizing CY's backend prompt-based algorithm.
- We demonstrate the usefulness and effectiveness of such a tool through a small-scale user study.

2 Culturally Yours

2.1 Overview

Given a user's cultural background, they might be unfamiliar with many concepts mentioned in online text, such as reviews, news articles, blogs, social media posts, etc. CY is a reading assistant that helps users acquaint themselves with such concepts by highlighting spans of text that a user might be unfamiliar with, given their cultural background. As shown in Figure 1, the tool takes a URL as input and identifies CSIs based on the demographic details of the user. The tool also categorizes the identified spans into *Unfamiliar* and *Somewhat Familiar*, depicting different levels of familiarity of the highlighted spans. To personalize CY, users can interact with the tool and make adjustments by (i) Selecting other text spans they don't understand and assigning a level of familiarity. (ii) Modifying the familiarity levels of the currently highlighted spans. (iii) Removing the highlighted spans. The initial back-end prompt is updated based on these interactions, and the updated prompt subsequently identifies spans in new documents. The spans highlighted in a new document show that the tool, starting from the user's culture, has adjusted to their preferences. Users can iteratively use the tool for multiple documents, where each interaction improves the tool's understanding of the user's preferences and facilitates personalization.

2.2 Features

CY incorporates a range of functionalities designed to assist users in identifying CSIs. Below, we outline the key features of the system:

1. **URL Parsing:** The system inputs URLs of documents and efficiently extracts the relevant textual content from the document.
2. **CSI Identification:** The tool highlights CSIs from the extracted text per the user's demographic background and categorizes the CSIs as "unfamiliar" or "somewhat familiar".
3. **User Interaction:** Users can delete highlighted spans, modify the familiarity level of the highlighted span, or select new spans of text with two levels of familiarity. The interaction window enables a customized and interactive experience for the users.
4. **User Feedback:** The system treats the user interaction as *relevance feedback* and adjusts its prompt to align the CSIs according to the user's preference in a new input URL/document.

2.3 Frontend

The front end of CY uses the `Vue.js`[3] framework. The framework manages user sessions, collects the user's socio-demographic information, and highlights CSIs for the user from a given cultural background. The front end mainly consists of the following two pages.

1. **Homepage:** This page allows users to input a URL and provide their demographic informa-

[2] https://openai.com/index/hello-gpt-4o/

[3] https://vuejs.org/guide/introduction.html

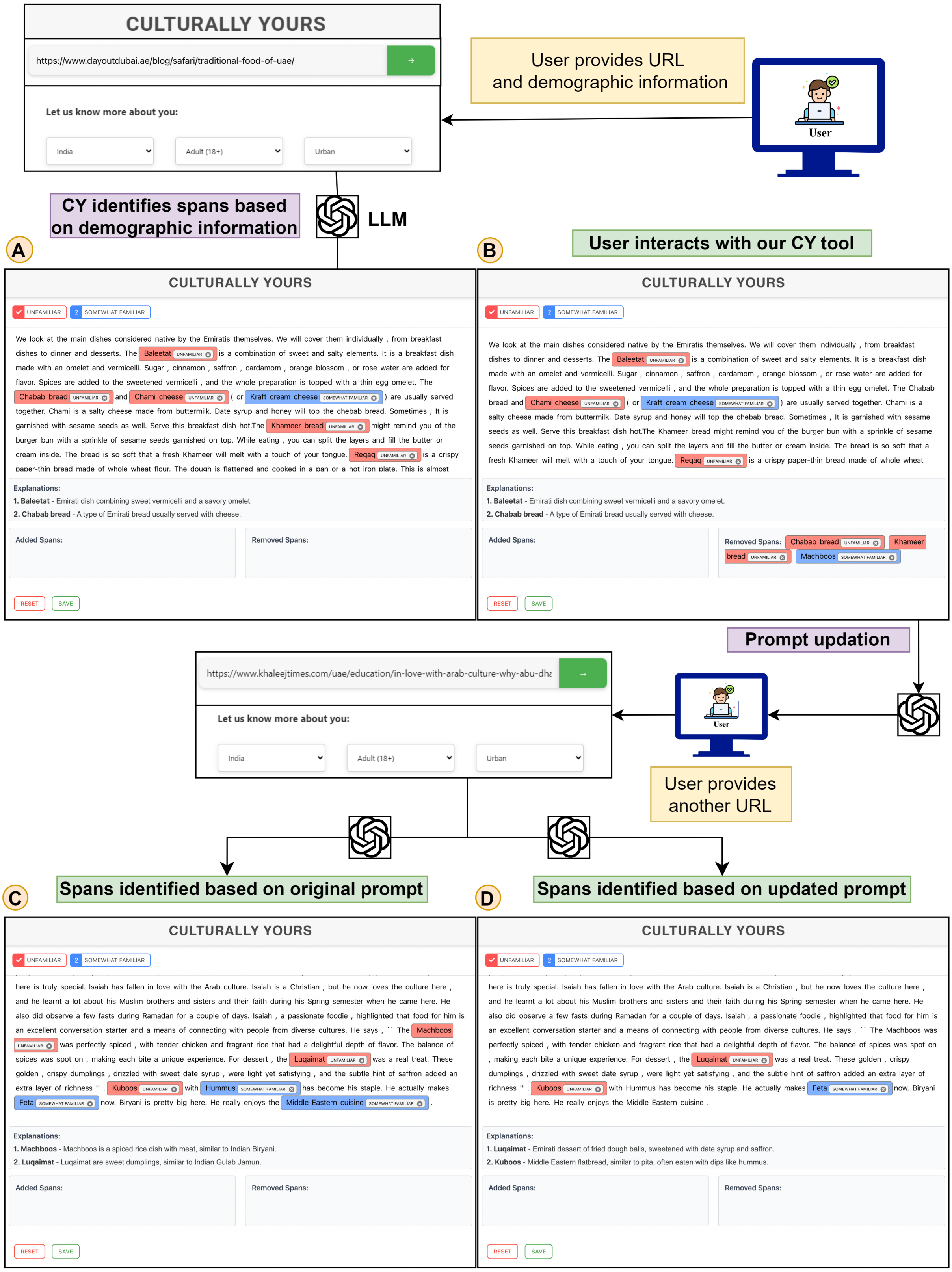

Figure 1: System overview of CULTURALLY YOURS (CY). A user provides a URL and demographic information such as country, age group, and region. (A) CY identifies CSIs based on the user's demographic details. (B) The user interacts with the tool, which updates the user's preferences and the prompt. CSIs are identified in a new text using the updated prompt. (C) Shows the highlighted spans on a new text using the original prompt. (D) Shows the highlighted spans on a new text using the updated prompt.

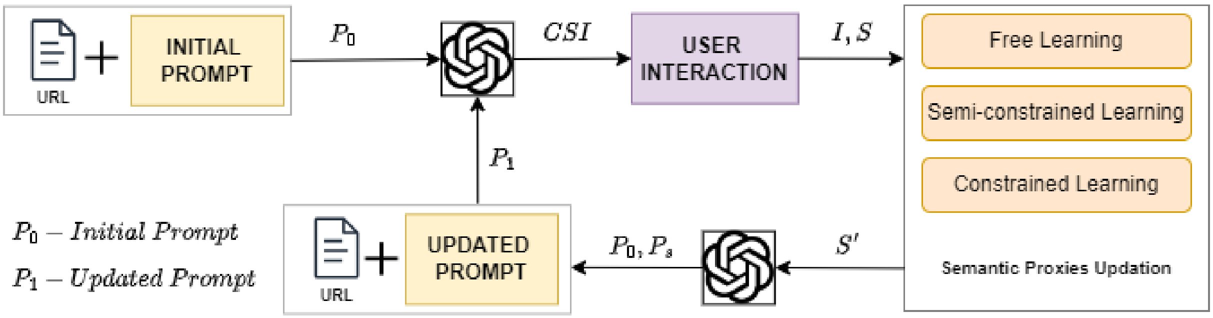

Figure 2: Overview of the prompt refinement in the backend. I represents the list of selected and deselected spans of text. S represents the initial semantic proxies. S' represents the updated semantic proxies based on I.

tion: country, age group, and region.

2. **Interaction page:** Given a URL and demographic information, this page displays the relevant text with CSIs highlighted with different levels of familiarity. This page also allows users to interact with our tool by modifying the familiarity level of the highlighted spans, deleting highlighted spans, and adding spans unfamiliar to the user that were unidentified by the tool. This interaction helps the tool learn user preferences and adjust accordingly.

2.4 Backend

The backend consists of a REST-based web server hosted on an Azure Virtual Machine with 16 GB of RAM. This setup enables scalable and efficient interactions with the APIs of various closed-source and open-source LLMs, supporting the execution of experimental workflows. The backend uses inputs from the user to interact with the LLMs, which generate responses and return the processed output in JSON format. The backend system performs the following tasks:

1. Parses the textual content from the user-provided URL.
2. Given the parsed URL text and the user's demographic information, the backend identifies CSIs for the user. It also categorizes the CSIs into two levels of familiarity - somewhat familiar and unfamiliar.
3. Explains the highlighted CSIs by simplifying them as per the user. It tries to relate them to concepts from the user's culture.
4. Utilizes the interactions to reformulate the prompts and improve alignment with the user's cultural background and preferences. The overview of the prompt refinement in the backend is shown in Figure 2. The prompts used for updating semantic proxies are shown in Figure 6.

2.5 Prompting Strategies for Personalization

We implement three prompting-based learning strategies for personalizing the tool to a user's preferences. (i) **Free learning:** The user's selected and deselected text spans are used directly in the backend LLM's prompt as preferences without explicitly interpreting their meaning. We implement a chat-based system to interact with the LLM. We append the spans to the LLM's prompt history, and the model personalizes to the user's preferences without explicitly interpreting the meaning of highlighting or deselecting a span in terms of preferences. (ii) **Constrained learning:** We introduce four semantic proxies, *political awareness*, *food cuisine*, *education level*, and *literature preference*, to denote user preferences. Semantic proxies refer to deeper representations of a culture and help bridge the gap between various cultural understandings (Thompson et al., 2020; Adilazuarda et al., 2024). We use the user's interaction to update the semantic proxies and reformulate the chat-based prompts to align the LLM with the user's preferences. (iii) **Semi-constrained learning:** This strategy mixes free and constrained learning, where we update only two semantic proxies based on the user's interaction and append the selected or deselected spans for the other two proxies, much like the free learning strategy.

3 User Study

We perform a user study using CY with three different prompting-based settings to evaluate the effectiveness of the CY tool and determine the best-performing setting for personalization to the user's culture.

3.1 Document Samples

We select articles from two domains - political news and food reviews. We consider three online articles each from news related to US elections and the traditional food of UAE. Food reviews contain descriptions of local and global food spanning various cultures. The news articles pertain to global political news that is widely recognized and understood. This choice of articles allows us to analyze how cultural familiarity influences user interactions through localized and universally known content. We limit the number of articles for each domain to three to ensure a focused user study while allowing us to gather meaningful insights across a diverse range of demographics.

3.2 Method

For a domain, the user enters their demography (country, region, and age group) and a URL. Estimating the user's culture by their demography, we prompt GPT-4o to identify CSIs in the text extracted from the URL, using culture as a prior. CY identifies CSIs and categorizes them into different familiarity levels. The user interacts with the tool by deselecting the highlighted CSIs they are already familiar with and selecting new CSIs from the text they are unfamiliar with. Once the user is satisfied with the interaction, they can save their preferences. The user interacts with the tool subsequently with two more URLs and saves their preferences every time. A user repeats this study for both domains (food and politics) under the three learning strategies (free, constrained, and semi-constrained). Lastly, we collect feedback from the user on the following aspects of CY.

- **CSI Identification:** How effective is the tool at identifying CSIs?
- **CSI Explanation:** How accurate is the tool at explaining CSIs?
- **Personalization:** How good is the tool at personalization?

We perform this study across 13 users from 8 diverse demographics of India, Indonesia, China, Mexico, Sri Lanka, Egypt, Uzbekistan, and Kazakhstan.

3.3 Evaluation

We define a metric, Average Interaction Rate (AIR), to measure the effectiveness of CY. The AIR is computed for each domain (S) and overall, and

Strategy	Domain		
	Food ↓	Politics ↓	Overall ↓
Free	0.57	**0.60**	0.58
Semi-constrained	0.53	0.64	0.59
Constrained	**0.52**	0.62	**0.57**

Table 1: Average Interaction Rate for different strategies across 13 users for Food, Politics, and Overall

defined as follows.

$$\text{AIR}(S) = \frac{1}{|U| \times |D|} \sum_{u=1}^{U} \sum_{d=1}^{D} \frac{I(u,d)}{HS(u,d)}$$

U is the set of all the users and D is the set of all documents for a domain S. $I(u,d)$ is the total number of interactions for a user u on a document d. $HS(u,d)$ represents the total number of highlighted spans after interaction from a user u on a document d. The fraction represents the percentage of interaction by a user u on a document d.

Lesser selection and deselection by the user yields a lower AIR score and indicates that the tool appropriately highlighted CSIs according to the user's culture and preferences, demonstrating better personalization to the user. A higher AIR score suggests otherwise.

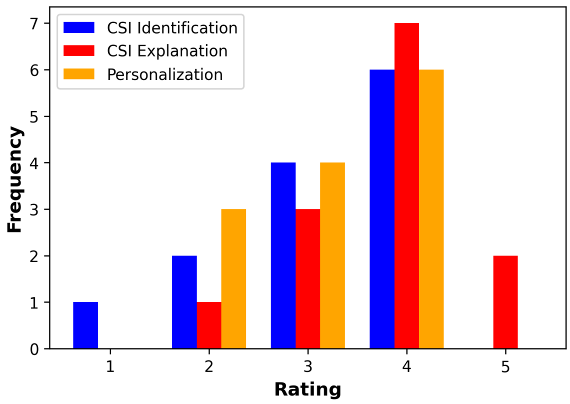

Figure 3: Plot of user ratings on a Likert scale.

3.4 Findings

From Table 1, we observe that **Free Learning** yields a lower average interaction value on Politics (0.60), whereas **Constrained learning** attains the best result for Food domain (0.52) and overall (0.57). We hypothesize that the nature of a domain impacts the performance of a learning strategy, where each domain might implement distinct strategies. We leave testing this hypothesis across multiple domains and more users as future work.

We also collect user feedback on a Likert Scale (1-5) for three different aspects of our tool, namely CSI Identification, CSI Explanation, and Personalization, as described in Section 3.2 and Section 6. From Figure 3, we observe high satisfaction among users for the explanation of CSIs. We also observe positive feedback on CSI Identification and Personalization, with 10 out of 13 users providing ratings of 3 and 4. The absence of ratings of 5 for CSI identification and personalization suggests that while users are generally satisfied, there is still room for further enhancement of the tool's features to reach higher satisfaction levels.

4 Related work

Copilots: The rapid advancement of AI-based copilots has significantly influenced software development and writing assistance. One of the most notable examples is GitHub Copilot, which assists developers by providing code suggestions in real-time. Finnie-Ansley et al. (2022) demonstrates that while copilots enhance workflow efficiency, human oversight is essential for accuracy. Dakhel et al. (2023) also shows that copilot works for almost all fundamental algorithmic problems. However, some solutions are buggy and non-reproducible. An empirical study was carried out by Nguyen and Nadi (2022), which shows some shortcomings of copilots, such as generating complex code that is reducible and code that relies on undefined helper methods.

Writing assistants: Paetzold and Specia (2016) proposed the task of Complex Word Identification (CWI) to learn which words are challenging for non-native English speakers. Recent methods (North et al., 2023) also show that the complexity of words within a given text various for different readers. With the recent advancements of AI technologies, research suggests that digital writing tools can positively impact the quality of English writing (Nobles and Paganucci, 2015). AI-powered writing tools have emerged to support users in their English writing processes (Barrot, 2022; Coenen et al., 2021) and enhance users' writing skills while facilitating their learning (Pokrivcakova, 2019; Nazari et al., 2021). Most writing tools focus on the revision and editing stage (Winans, 2021). Zhao (2023) introduced Wordtune, an AI-powered technology that helps users during the writing process by understanding what they wish to say and helping them formulate their ideas into sentences by offering rephrasing options.

5 Conclusion

We introduce Culturally Yours (CY), a unique cultural reading assistant designed to bridge the gap in cross-cultural understanding of online texts. By leveraging user feedback and relevance feedback techniques, CY captures user preferences and algorithmically adapts its prompt to suit the user's background and preferences. The tool's ability to overcome the cultural cold-start problem and improve personalization based on user interaction under three different experimental settings prove the usefulness of the tool. With its focus on cultural adaptation, CY enhances cross-cultural content understanding and opens up avenues for improving user engagement across global platforms. Thus helping globalize locally written content.

Acknowledgements

This work is supported by the Microsoft Accelerate Foundation Models Research (AFMR) Grant. We thank all the participants involved in the internal ideation workshop of the tool. We also thank all the users involved in the tool's internal pilot study.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Muhammad Farid Adilazuarda, Sagnik Mukherjee, Pradhyumna Lavania, Siddhant Singh, Ashutosh Dwivedi, Alham Fikri Aji, Jacki O'Neill, Ashutosh Modi, and Monojit Choudhury. 2024. Towards measuring and modeling" culture" in llms: A survey. *arXiv preprint arXiv:2403.15412*.

Jessie Saraza Barrot. 2022. Integrating technology into esl/efl writing through grammarly. *Relc Journal*, 53(3):764–768.

Sébastien Bubeck, Varun Chandrasekaran, Ronen Eldan, Johannes Gehrke, Eric Horvitz, Ece Kamar, Peter Lee, Yin Tat Lee, Yuanzhi Li, Scott Lundberg, et al. 2023. Sparks of artificial general intelligence: Early experiments with gpt-4. *arXiv preprint arXiv:2303.12712*.

Andy Coenen, Luke Davis, Daphne Ippolito, Emily Reif, and Ann Yuan. 2021. Wordcraft: A human-ai collaborative editor for story writing. *arXiv preprint arXiv:2107.07430*.

Arghavan Moradi Dakhel, Vahid Majdinasab, Amin Nikanjam, Foutse Khomh, Michel C Desmarais, and Zhen Ming Jack Jiang. 2023. Github copilot ai pair programmer: Asset or liability? *Journal of Systems and Software*, 203:111734.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

James Finnie-Ansley, Paul Denny, Brett A Becker, Andrew Luxton-Reilly, and James Prather. 2022. The robots are coming: Exploring the implications of openai codex on introductory programming. In *Proceedings of the 24th Australasian Computing Education Conference*, pages 10–19.

Yifan Hu, Yehuda Koren, and Chris Volinsky. 2008. Collaborative filtering for implicit feedback datasets. In *2008 Eighth IEEE international conference on data mining*, pages 263–272. Ieee.

Perihan Korkut, Mustafa Dolmacı, and Burcu Karaca. 2018. A study on communication breakdowns: Sources of misunderstanding in a cross-cultural setting. *Eurasian Journal of Educational Research*, 18(78):139–158.

Jinqi Lai, Wensheng Gan, Jiayang Wu, Zhenlian Qi, and Philip S Yu. 2023. Large language models in law: A survey. *arXiv preprint arXiv:2312.03718*.

Erin Meyer. 2014. *The culture map: Breaking through the invisible boundaries of global business*. Public Affairs.

Nabi Nazari, Muhammad Salman Shabbir, and Roy Setiawan. 2021. Application of artificial intelligence powered digital writing assistant in higher education: randomized controlled trial. *Heliyon*, 7(5).

Peter Newmark. 2003. A textbook of translation.

Nhan Nguyen and Sarah Nadi. 2022. An empirical evaluation of github copilot's code suggestions. In *Proceedings of the 19th International Conference on Mining Software Repositories*, pages 1–5.

Susanne Nobles and Laura Paganucci. 2015. Do digital writing tools deliver? student perceptions of writing quality using digital tools and online writing environments. *Computers and Composition*, 38:16–31.

Harsha Nori, Nicholas King, Scott Mayer McKinney, Dean Carignan, and Eric Horvitz. 2023. Capabilities of gpt-4 on medical challenge problems. *arXiv preprint arXiv:2303.13375*.

Kai North, Marcos Zampieri, and Matthew Shardlow. 2023. Lexical complexity prediction: An overview. *ACM Computing Surveys*, 55(9):1–42.

Gustavo Paetzold and Lucia Specia. 2016. Semeval 2016 task 11: Complex word identification. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval-2016)*, pages 560–569.

Silvia Pokrivcakova. 2019. Preparing teachers for the application of ai-powered technologies in foreign language education. *Journal of Language and Cultural Education*, 7(3):135–153.

Yong Rui, T.S. Huang, M. Ortega, and S. Mehrotra. 1998. Relevance feedback: a power tool for interactive content-based image retrieval. *IEEE Transactions on Circuits and Systems for Video Technology*, 8(5):644–655.

Bill Thompson, Seán G Roberts, and Gary Lupyan. 2020. Cultural influences on word meanings revealed through large-scale semantic alignment. *Nature Human Behaviour*, 4(10):1029–1038.

Michael D Winans. 2021. Grammarly's tone detector: Helping students write pragmatically appropriate texts. *Relc Journal*, 52(2):348–352.

Xin Zhao. 2023. Leveraging artificial intelligence (ai) technology for english writing: Introducing wordtune as a digital writing assistant for efl writers. *RELC Journal*, 54(3):890–894.

6 Appendix

User Study Feedback Tables 2, 3, 4 represent the descriptions of various rating levels used for CSI identification, CSI explanation, and Personalization.

Rating	Description
5	Highlighted all the spans correctly
4	Highlighted most of the spans correctly, missed a few spans
3	Highlighted some of the spans correctly, missed a few spans
2	Higlighted some of the spans incorrectly, missed a lot of spans
1	Higlighted most of the spans incorrectly, missed a few spans

Table 2: Rating description for Identification of Spans

Rating	Description
5	Personalizes perfectly, identifies all spans correctly
4	Personalizes reasonably well, identifies most of the spans correctly
3	Personalizes to a certain extent, some spans were identified incorrectly
2	Does not personalize properly, most of the spans were identified incorrectly
1	Does not personalize at all, all the spans were identified incorrectly

Table 3: Rating description for feedback on personalization

Rating	Description
5	Gave perfect explanations, helped me learn new things
4	Gave reasonably good explanations, helped me understand the text better
3	Some more explanations were needed, some cases it did not help
2	Explanations seem factually correct but did not help me understand the article better
1	Explanations were factually incorrect and confused me a lot

Table 4: Rating description for Explanation of CSIs

Initial Prompt

AI Rules
- Output response in JSON format only
- Do not output any extra text
- Do not wrap the JSON codes in JSON or Python markers
- JSON keys and values in double-quotes

You are a cultural mediator who understands all cultures across the world. As a mediator, your job is to identify and translate culturally exotic concepts from texts from an unknown source culture to my culture. I am a well-educated [age_group] person who grew up in [region] [country], which defines my culture. I came across a piece of text.

Task 1: Identify all culture-specific items (CSIs) from the text that I might find hard to understand due to my cultural background. CSIs are textual spans denoting concepts and items uncommon and not prevalent in my culture, making them difficult to understand.

Task2: For each CSI, identify its familiarity from one of the following three levels: 1. Familiar: Most people from my culture know and relate to the concept as intended. 2. Somewhat familiar: Only some people from my culture know and relate to the concept as intended. 3. Unfamiliar: Most people from my culture do not know or relate to the concept.

Task 3: Within 50 words, detail your reason for highlighting the span as CSI in Task 1 by correlating it with my background.

Task 4: Explain each CSI span within 20 words to make it more understandable to your client. Provide facts, examples, equivalences, analogies, etc, if needed.

Format your response as a valid Python dictionary formatted as: "spans": [List of Python dictionaries where each dictionary item is formatted as: "CSI": <task 1: copy the CSI span from text>, "familiarity": <task 2: familiarity level name>, "reason": <task 3: reason within 50 words>, "explanation": <task 4: explain the span within 20 words>]. Respond with "spans": "None" if you think I will not find anything difficult to understand.

Text: [review_text]

Update Prompt (Free Learning)

AI Rules
- Output response in JSON format only
- Do not output any extra text
- Do not wrap the JSON codes in JSON or Python markers
- JSON keys and values in double-quotes

On further understanding, I observe the following things.

I am familiar with spans of text like [[spans of text]]. I am somewhat familiar with spans of text like [[spans of text]]. I am unfamiliar with spans of text like [[spans of text]].

Update Prompt (Semi-constrained Learning)

AI Rules
- Output response in JSON format only
- Do not output any extra text
- Do not wrap the JSON codes in JSON or Python markers
- JSON keys and values in double-quotes

On further understanding, I observe the following things.

I am familiar with spans of text like [[spans of text]]. I am somewhat familiar with spans of text like [[spans of text]]. I am unfamiliar with spans of text like [[spans of text]].

Based on familiarity with these spans, update my background cultural information. Return them as a valid Python dictionary. {"political-awareness":<yes/no>, "food-cuisine":<japanese/mexican/american/emirati>}

Update Prompt (Constrained Learning)

AI Rules
- Output response in JSON format only
- Do not output any extra text
- Do not wrap the JSON codes in JSON or Python markers
- JSON keys and values in double-quotes

On further understanding, I observe the following things.

I am familiar with spans of text like [[spans of text]]. I am somewhat familiar with spans of text like [[spans of text]]. I am unfamiliar with spans of text like [[spans of text]].

Based on familiarity with these spans, update my background cultural information. Return them as a valid Python dictionary. {"political-awareness":<yes/no>, "education-level":<primary/secondary>, "food-cuisine":<japanese/mexican/american/emirati>, "literature-preference":<bengali/english/hindi>}