Reading.help: Supporting EFL Readers with Proactive and On-Demand Explanation of English Grammar and Semantics

Sunghyo Chung
Kakao
Seongnam, South Korea
shawn.hyo@kakaocorp.com

Hyeon Jeon
Seoul National University
Seoul, South Korea
hj@hcil.snu.ac.kr

Sungbok Shin*
Aarhus University
Aarhus, Denmark
sbshin90@cs.umd.edu

Md Naimul Hoque
University of Iowa
Iowa City, United States
nhoque@umd.edu

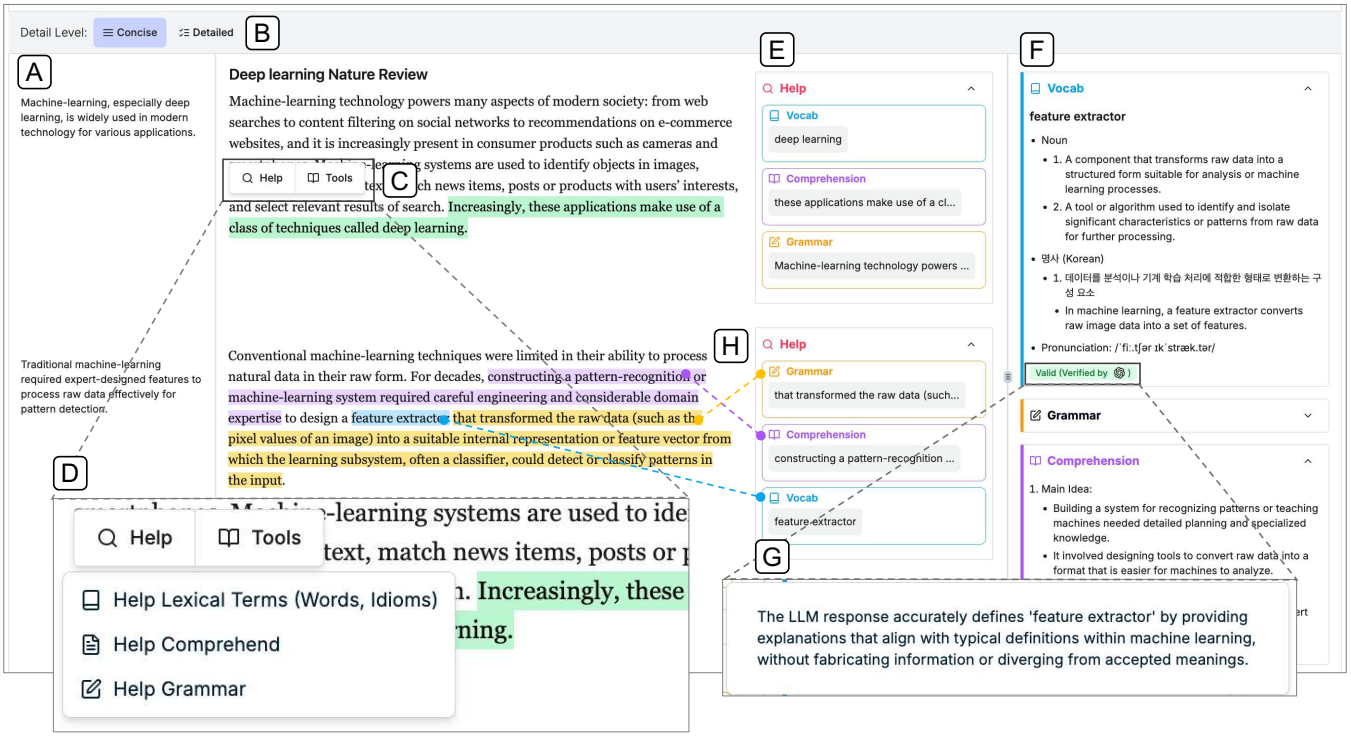

Figure 1: The READING.HELP interface. READING.HELP assists English as a Foreign Language (EFL) readers in understanding English texts by identifying areas that may confuse readers. The system assists EFL readers with (A) content summaries and (B) adjustable level of summaries (concise or detailed). (C) When users select a specific text, they can access the supporting tools. (D) The Tools menu expands to offer three assistance options: Lexical Terms, Comprehension, and Grammar. (E) The system identifies potentially challenging content for EFL readers and provides recommendations for each paragraph. (F) When the vocabulary tool is selected, the explanations appear with definitions and contextual information. (G) All explanations go through a validation process through a second LLM, which validates the reasoning of the first LLM. (H) The suggestions are linked to the text through highlighting.

Abstract

A large portion of texts is written in English, but readers who see English as a Foreign Language (EFL) often struggle to read texts accurately and swiftly. EFL readers seek help from professional teachers and mentors, which is limited and costly. In this paper, we explore how an intelligent reading tool can assist EFL readers. We conducted a case study with EFL readers in South Korea. We at first developed an LLM-based reading tool based on prior literature. We then revised the tool based on the feedback from a study with 15 South Korean EFL readers. The final tool, named READING.HELP, helps EFL readers comprehend complex sentences and paragraphs with on-demand and proactive explanations. We finally evaluated

*Corresponding Author.

the tool with 5 EFL readers and 2 EFL education professionals. Our findings suggest Reading.help could potentially help EFL readers self-learn English when they do not have access to external support.

CCS Concepts

• **Human-centered computing** → **Interactive systems and tools**.

Keywords

Augmented Reading, Human-centered AI, English as Foreign Language, EFL, Automated reading assistant

1 Introduction

English is the most widely used language for communication in academic, professional, and social contexts worldwide. A vast amount of literature, including books and documents, is written in English, with millions of new documents being created at any given moment. As a result, the ability to read English accurately and efficiently remains a crucial skill for individuals globally. Acknowledging this need, many educational institutions in countries where English is considered a foreign language incorporate English-language curricula at various levels, such as in international schools with O and A levels [68]. However, these educational programs are often limited and costly, remaining inaccessible to a significant portion of the population. Even students who receive formal English education face difficulties due to the lack of exposure to advanced vocabulary and cultural references outside the classroom.

Due to the challenges in formal English education, many readers who consider English a foreign language (EFL) seek resources to self-educate themselves. There are now online courses, but they are limited in scope and do not provide on-demand and situational support readers need while reading an article. Various tools are available to fill this gap, including electronic dictionaries and online translators such as the Google Translate[1]. While electronic dictionaries are widely used, they offer limited support in identifying areas where readers may misinterpret or misunderstand the text. Online translators offer a convenient means of understanding English texts. While their strategic use (e.g., searching for short phrases, clauses, etc.) can support both comprehension and learning, providing full-text translations may lead many non-native English learners to translate entire passages, increasing reliance on the tool and reducing motivation to engage with new vocabulary or syntactic structures [1]. These points suggest that EFL readers could benefit from a tool that supports independent interpretation of English texts as well as offer learning opportunities for the readers.

With this objective in mind, we introduce Reading.help, an interface designed to help EFL readers' independent reading of English texts by leveraging natural language processing (NLP) techniques. Reading.help mainly targets EFL readers, who have university degrees, and are willing to independently interpret English texts. Reading.help contains two different modules: 1) an **identification** module for detecting words and sentences that an EFL reader may find difficult; and 2) an **explanation** module for clarifying vocabulary, grammar, and overall context of the text. We developed two specialized and interpretable neural models for detection challenging words and sentences. We then use a Large Language Model (LLM) to provide explanations for the identified words and phrases. Users of the tool can also manually highlight any text and request explanation. To ensure the reliability of these explanations, a secondary Large Language Model (LLM) evaluates and refines the information provided. Finally, the explanation module generates concise summaries of paragraphs within the text, helping readers to grasp the main idea without becoming overwhelmed by details.

As a case study, we base our work in the context of South Korea. We initially conducted a pilot study with 15 South Korean EFL readers. Based on the feedback from the pilot study, we revised the tool and subsequently evaluated Reading.help in a study involving 5 EFL readers and 2 domain experts in English education for EFL readers in South Korea. We assess the effectiveness of our tool and observe how these readers benefit from Reading.help. We find that people use our tool mainly to address text complexity issues and gain comprehension support. Reading.help reports an average recall of 87% when recommending three elements per dimension. Furthermore, while the self-validation performance of LLMs is comparable to that of humans in detecting incorrect meaning issues, LLMs remain limited in identifying other issues, such as missing elements.

2 Background and Related Work

We explain the background of our work in three parts: (1) difficulties for EFL readers, (2) computational approaches to assist academic reading, and (3) work that augments writing.

2.1 Difficulties in EFL Reading

We identified studies to understand effective methods to learn English, in both reading [8, 21, 32, 49] and writing [24, 60, 74]. Hirsch [25] argues that *fluency, the breadth of vocabulary,* and *the domain knowledge about the topic* are the three factors that contribute to reading. Fluency is the capability to connect current text a person is reading with the preceding text. Lack of vocabulary creates latency in connecting the previous and current content. Lack of domain knowledge can make it difficult to understand the context, such as contradiction, and eventually create a mental model that is out of the context. Note that a lack of one factor can lead to difficulty in other factors.

We found additional framing of these challenges in the literature. The first challenge is in *text complexity* [14, 30], where the readers struggle to understand the main gist of the text. Text complexity involves not just difficult vocabulary, but also complex grammatical rules and lengthy sentence structures. All of these factors can lead to misinterpretation or slower reading.Another framing is *reading strategy* [14, 26, 59]: inferring the right context, topic, and takeaways from a text block. Failure to understand the context, or comprehensive breakdown, is common among EFL readers.

Our approach aims to address these general challenges faced by EFL readers. We categorize the challenges into three challenges (vocabulary, grammatical, and comprehension) and addressed them in a unified tool. The goal is to provide on-demand support to EFL readers when they face the issues as well as proactively recommend potential issues to improve the self-learning process.

[1] https://translate.google.com/

2.2 Intelligent Reading Tools

HCI and AI research has a long history of developing intelligent reading tools. The primary focus is to speed up reading [43, 75]. For example, Marvista [10] aids users in reading and comprehending text based on the time they want to spend, using AI-generated questions. CReBot [58] interactively asks section-level critical thinking questions, catering to both experienced researchers and novices. Highlighttext [67] explores effective text highlighting methods, while Living Papers [22] introduces a grammar for augmenting scientific articles. SCIM [16] enables researchers to quickly skim papers, and Vogel et al. [34] show that highlighting a limited number of keywords improves comprehension.

Several works aim to improve the understanding of charts and tables within documents. One approach is adding textual information [18, 66] on visualizations. Kim et al. [40] link text and tables, while Elastic Documents [6] use visualizations to connect them for better summarization. Kori [44] provides an interactive interface linking charts and text, and Statslator [52] visually translates statistical tables for clearer interpretation. Other works focus on improving chart accessibility for readers [11, 12, 27, 41].

Another line of research focuses on using eye movement data to understand and predict confusion [42, 64] and mind wandering [15] through Machine Learning techniques. These studies aim to identify eye movement patterns associated with confusion and use them to predict confusion or mind wandering during reading.

Overall, the above works improve reading experiences for users. However, they do not cater to the needs of EFL readers. The methods also lack focus on the learning aspect, which is critical for EFL readers. Indeed, EFL readers may not only want to read fast but may want to learn about the syntax and semantics. Our work aims to fill that gap.

It is worth noting that researchers have proposed a few tools for learning English. For example, ChatPRCS [71] tailors reading comprehension questions to the student's English proficiency. VocabEncounter [3] uses recent NLP techniques to integrate vocabulary into the user's reading material in near real-time. CriTrainer [73] helps develop critical paper reading skills through text summarization and template-based questions. Our work extends this line of work by developing NLP methods to proactively identify text that EFL readers may find difficult and then using an LLM to explain the grammar and context of the text. Reading.help also provides on-demand explanations for texts identified by readers as difficult. Finally, the tool uses a verification process to improve the reader trust in the explanation provided, improving the learning experience.

2.3 Intelligent Writing Tools

Reading and writing are fundamentally related to each other. Learning to write English could potentially improve reading skills. As such, intelligent writing tools are relevant to our work. Commercial tools such as Grammarly can help writers improve grammatical and semantical aspects of a written text. There is a new wave of intelligent writing tools powered by LLMs [36, 38, 47, 70, 76]. Ito et al. [31] conduct a study validating AI-assisted drafting of English essays by non-native speakers, developing a tool called Langsmith to refine academic essays. Figura11y [65] supports the writing of scientific alt text. There are efforts to provide customized related work suggestions for scientific articles [9, 35, 37, 56], often using algorithms based on users' publication history. The algorithms mainly deployed in these works are based on users' past records. Research search engines like Google Scholar now offer personalized paper recommendations, and there are works [48] that aim to improve recommendation accuracy with LLMs. Finally, several studies also evaluate the impact of LLMs in human-LLM collaborative writing [19, 28, 50].

This line of research motivated our research. While there are numerous intelligent writing tools (and reading tools), how NLP and LLMs can assist EFL readers remains largely unknown. Our work shows that carefully designed NLP heuristics, along with the LLM can capture a wide range of problems a reader might have and provide proactive and on-demand explanation.

3 Design Requirements

We adopted an iterative design approach to develop the proposed tool for EFL readers. Before developing our final tool, we created an initial version, Reading.help-Lite, to evaluate the strengths and limitations of our approach. Below, we describe our preliminary design requirements, grounded in prior literature.

- **DR1** **Proactive and On-Demand Guidance.** The tool should guide EFL readers on parts that are potentially difficult to parse or understand. Such proactive guidance could help EFL readers focus on textual complexity that they might have overlooked [14, 30] or proceed with the wrong interpretation [14, 26, 59]. It is also important that such guidance is provided on-demand, upon the user's request.
- **DR2** **Explain vocabulary, grammar, and semantics.** Our tool should provide explanations for improving text comprehension. The tool should effectively explain unknown vocabulary and complex grammatical structure. The tool should also explain the core topic and main takeaways of a text segment.
- **DR3** **Drill-down to the Details.** To deliver knowledge and information in a limited amount of computer space, feedback should be arranged hierarchically [62], where the tool initially provides a high-level summary and the user can drill down to refer to more details [63].
- **DR4** **User-centered adaptive support.** EFL readers may have different levels of profiency in English. The readers' formal education and social exposure to English can also vary. Thus, it is important to provide feedback that is tailored to varying levels of English proficiency.
- **DR5** **Visual Emphasis.** Visual emphasis techniques [29] such as word highlights can make the tool effective for two reasons: (1) the emphasis could perceptually guide users to look for salient parts; and (2) it could help readers spend less time in looking for a specific keyword of interest in the text [67].

4 Reading.help-lite

Here, we describe the implementation of Reading.help-Lite, our initial attempt to address DR1-DR5. Figure 2 shows the interface of Reading.help-Lite. We chose an LLM as the main analytical pipeline for this version because of the ease of access and strong

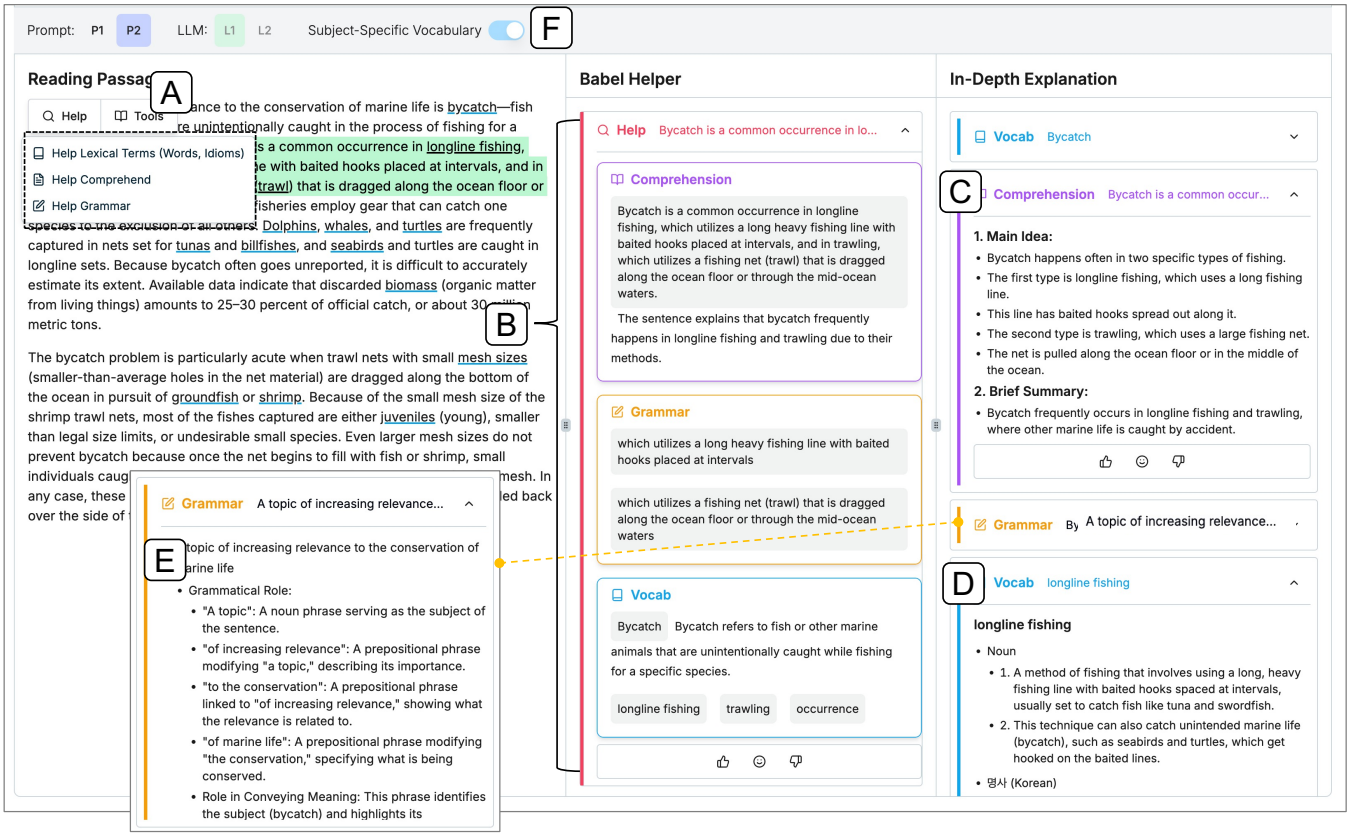

Figure 2: READING.HELP-lite interface. (A) A control panel that appears upon selection of a text. The user can choose to get help, or manually look for what they want by pressing the button 'tools.' (B) On selecting 'help', READING.HELP-Lite displays potential challenging issues that the user might be interested in. A user can select any of the issues to see it in a detailed view (C, D, E). By toggling the 'subject-specific vocabulary,' (F) the user can view subject-specific keywords underlined in the text.

analytical capability. We anticipated that the LLM-powered prototype will allow us to conduct preliminary study with real users and help us design the final version of the tool.

4.1 Interface Overview

4.1.1 Workflow. READING.HELP-lite identifies and suggests potential sources of difficulty by analyzing a piece of text uploaded by a user (DR1). The text is analyzed along three dimensions: vocabulary (denoted as 'lexical terms'), grammar, and comprehension (DR2). The *vocabulary* component identifies unfamiliar keywords or phrases (e.g., idioms and short clauses). The *comprehension* component situates the text within the broader context of the passage. Finally, the *grammar* component addresses grammatical issues. The three types of issues (i.e., dimensions) are presented in three different types of colored boxes (DR5) (Figure 2B).

4.1.2 Detailed explanations to users. The *vocabulary* module provides the literal definition of a keyword or idiom. For words with multiple interpretations (e.g., 'take' and 'get'), we offer only the context-relevant definition. A corresponding Korean translation is also supplied for verification (see Figure 2D). In the *comprehension* module (see Figure 2C), we provide the main idea, and the intention of the selected text in relation to the flow of the entire passage. The main idea is provided as a bullet point to facilitate the understanding. We also provide several examples of sentences that convey the same meaning, but using different wordings and structure. This is to help users better understand the current context by comparing it with other possible contexts. Finally, the *grammar* module (see Figure 2E) examines sentence structure by segmenting text into phrases. For each phrase, the system offers keyword-level explanations that delineate the function of each component. The intent behind this is to help users quickly locate issues within a sentence while understanding how individual keywords form the overall phrase.

4.1.3 Additional Features. READING.HELP-Lite has several other interactive features. First, we highlight subject-specific keywords. The user can toggle the button next to the 'subject-specific vocabulary' label in Figure 2, and the tool underlines subject-specific keywords in blue. Hovering over an underlined keyword will show a brief definition of the keyword in a tooltip. Second, if a user encounters text segments not recommended by the tool but still remains unclear, then they can use the 'tools' button in the panel to directly select and analyze the segment by choosing a dimension

(vocabulary, grammar, or comprehension). The explanations provided to the user differ bt the proficiency level of the user (DR4) (discussed next).

4.2 Prompt Engineering

We use OpenAI's GPT-4o model for providing explanations to users. We iteratively tested various prompting methods to search for the optimal prompt in our setting. Effective prompt engineering is challenging as the responses from LLMs are not predictable at times, due to issues such as hallucinaton, verbosity and so on. We deploy seven guidelines from prior work and ChatGPT instructions in designing effective and fast prompts:

- Assign a role to GPT — to ensure responses are aligned with a specific expertise (e.g., expert English instructor)
- guide LLMs to respond in a specific response template — to avoid verbose responses.
- structure responses clearly and concisely — to avoid verbosity and maintain precision.
- use exact keywords — to avoid misunderstandings when prompt chaining.
- add global variables (e.g., user proficiency for supporting DR4) in the system prompt — to manage a user's specifications equally across various tasks.
- add local variables (e,g., tasks) in the user prompt – to use the same template equally for multiple users.

We explain three main types of prompt template used in the tool. The first template is for providing various explanations (e.g., grammar, components, lexical terms) used in Reading.help. The second template is for computing recommendations about what the user may not know about the user-selected text.

4.2.1 Template for providing explanations. Let us define the template for providing explanations as Tem_{ex}. Let $[Role]$ be the string that contains the prompt that specifies a role. We provide the role of an English instructor (e.g., "*You are an expert English instructor tasked with providing a comprehensive analysis of sentences.*"). Let $[Prof_u]$ be the information on the user's English proficiency skills (e.g., '*the user's English proficiency level is 'intermediate.*') . Note that $[Role_I]$ and $[Prof_u]$ are added in the system prompt. Let $[Text]$ be the string that contains the text selected by the user. $[Text]$ can be a part of the passage, such as '*Because bycatch often goes unreported, it is difficult to accurately estimate its extent.*' Let $[All-text]$ be the string that contains entire passage provided by the user, and $[Inst_T]$ be the string that contains of all task instructions needed to conduct the task. For example, for providing definitions about a vocabulary, the prompt is *'Provide a comprehensive definition for the given vocabulary word.'* Finally, let $[Form]$ be the string that contains the exact desirable answer format (e.g., '*Provide the response in the follwing manner: Banana 1. A fruit ... 2. ...* '). To sum up, the template for Tem_{ex} is:

$$Tem_{ex} = [Role_{ex}]+[Prof_u]+[Text]+[All-text]+[Inst_{ex}]+[Form_{ex}].$$

4.2.2 Template for recommendations (Help). Let us define the template Tem_R for providing recommendations in help components (comprehension, grammar, and lexical terms) for the user. We skip the definitions of $[Role_I]$, $[Prof_u]$, $[All-text]$ and $[Text]$, as they are defined in a similar manner to those in the template above. Let $[Inst_R]$ be the string that contains of all task instructions needed to provide recommendations (e.g., '*Please structure your response according to the guidelines provided in the system prompt.*'). Similar to above, $[Form_R]$ is the string that contains the desired exact answer format, Hence, to sum up, the tamplate for Tem_R is:

$$Tem_R = [Role_I]+[Prof_u]+[Text]+[All-text]+[Inst_R]+[Form_R].$$

Further details about the prompting and its relevant code can be found in here [2].

4.3 Implementation details

The front end Reading.help is implemented as a full-stack web application using Next.js, a React-based framework for both front-end and back-end development. In the backend system, the data management and routing are handled entirely within Next.js. The tool is hosted on Amazon Web Services (AWS) during the user study.

5 Pilot Study

We conducted a pilot study by using Reading.help-lite as a probe. The goal was to evaluate the effectiveness of Reading.help-Lite and identify limitations for improving the tool. The study was approved by the National Institutional Review Board (IRB) of South Korea. We first describe the study procedure, and then provide the results, and takeaways from the study.

5.1 Study Details

Participants. We recruited 15 South Korean participants for this study through the word of mouth approach. All participants had a bachelor degree but reported difficulties reading English (i.e., considered themselves as EFL readers). Among them 3 participants were English educators. Two were high-school teachers whereas the other is a consultant in an educational startup. Among the participants, 11 were men, while 4 were women. The gender distribution is imbalanced, partly because it was difficult to find adults who would identify as EFL readers as it could be a socially uncomfortable situation for many people.

Task and Materials. We define a unit task (UT) as a task in which, over the course of 5 minutes, a participant would read a passage (~200-250 words), highlight any part of the text that they find difficult, and read explanation provided by Reading.help-Lite. After each session, participants verbally informed the session administrator about the issues they faced and how the tool helped address the issues. The passages are derived from TOEFL free reading test samples from ETS [3] [4]. We decided to use passages from TOEFL, as it is a popular standardized test for university-level students.

Study Process. Each session consists of an introduction (~10 minutes), main study (~35 minutes), and a post-study interview (~15 minutes). Following consent and the collection of demographic information and relevant background information, participants were guided to Reading.help-Lite, where the study administrator (the first-author of this paper) demonstrated its features and explained

[2] https://osf.io/tf2w8/?view_only=f715f4c297334f55a95410f2c03da596
[3] https://www.ets.org/pdfs/toefl/toefl-ibt-free-practice-test.pdf
[4] https://www.ets.org/pdfs/toefl/toefl-ibt-reading-practice-sets.pdf

the required tasks. The main study started with a test to determine the level of English proficiency of the participants. For this test, we asked participants to read a passage for 5 minutes, and answer three questions pertaining to the passage. When a participant answered at most one question incorrectly, we classified them as 'proficient'. When they answered more than one question incorrectly, they are categorized as 'not proficient.' Based on this categorization, we prompted the LLM to provide easy-to-understand explanations for 'not proficient' participants. After that, participants completed four unit tasks.

During the post-interview study, we inquired about the effectiveness of the tool and the components and their general satisfaction with the tool. We also asked about limitations and failures of the tool. Each session lasted around 60 minutes. We provided a compensation of around KRW 15,000.- (around US$12.00) for the compensation. This rate is based on the hourly minimum rate of KRW 9,860/hr.

5.2 Results

Overall, the feedback from the participants were positive. They saw the value of such a tool for personal growth and improvement. We discuss the major findings below.

Usage patterns. We noticed that participants sought the most help with vocabulary, followed by grammatical structure. On average, participants sought 4.2 explanations for vocabulary, 2.1 for comprehension, and 0.9 for grammar, per each passage within a document (see Figure 3A). We find that participants used the 'help' button 1.6 times to identify all issues for a selected text. In contrast, participants sought specific explanation (that is either comprehension, grammar, or lexical terms) for a selected text 1.2 times (Figure 3B). While most participants found the feedback accessible, some users with lower TOEFL proficiency (4/15) reported that the responses were overly verbose and difficult to comprehend.

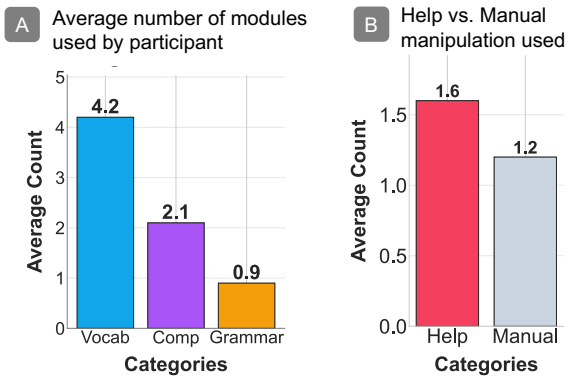

Figure 3: Reading.help-Lite usage during pilot study. (A) shows the average number of modules (vocabulary, comprehension, and grammar) used by participants while conducting a unit task. (B) compares the number of times 'help' and manual manipulation, or 'tools' are accessed while conducting a unit task.

Post-study interview. To begin with, All participants reported that LLM recommendations were beneficial, though they did not expect them to be completely error-free. Second, participants expressed a strong preference for the comprehension module (9/15) and the vocabulary module (8/15) among the deployed components. The comprehension module was utilized to decode lengthy, complex sentences. As one participant explained, *"The comprehension component explains difficult parts by showing the sentence's intention and considering the passage's overall context. This helps people spot their misinterpretations."*

5.3 Lessons Learned

We identified several challenges that must be addressed to make the tool more effective. Following limitations informed the revised tool in § 6.

- **Lack of trust and validation.** There is a consensus among participants that the responses of LLMs are insightful, but it is difficult to trust the responses. Participants were not confident in accepting the responses of the LLMs in some cases. Several participants queried about how the system is determining words that might be challenging.
- **Lack of adequate support for comprehension.** While Reading.help-Lite provided comprehension support for a selected text, we did not provide comprehension support for the whole document. The current selection based approach assumes that users will recognize what they do not know. However, we have observed that users may be uncertain about where to look for additional guidance. This issue is also echoed in the literature [59] that lack of effective English reading strategies also impedes EFL reading.
- **Verbosity of the grammar explanation.** the grammar component was widely regarded as the least useful, with 12/15 participants reporting minimal benefit. Participants were overwhelmed by its verbosity. Several participants suggested that this component could be really useful if the explanation is brief and concise.
- **Adapting to varying proficiency levels.** In our study, participants with lower TOEFL scores in three-question test received simplified responses. Reactions varied: some found the suggestions acceptable, while others struggled to comprehend them, indicating that the responses could be made more accessible for EFL readers in general.
- **Usability issues.** We also found some usability issues regarding Reading.help-Lite, such as the confusion with the different panels in the tool and lack of explanations for the colors used in the interface.

6 Reading.help: Updated Version

We develop Reading.help by addressing the issues we found in Reading.help-Lite (see §5.3). In this section, we describe the revised system with the key modifications.

6.1 Interpretable CEFR Prediction Models

One of the main concerns from the pilot study was that Reading.help-lite was not transparent. Participants were hesitant to

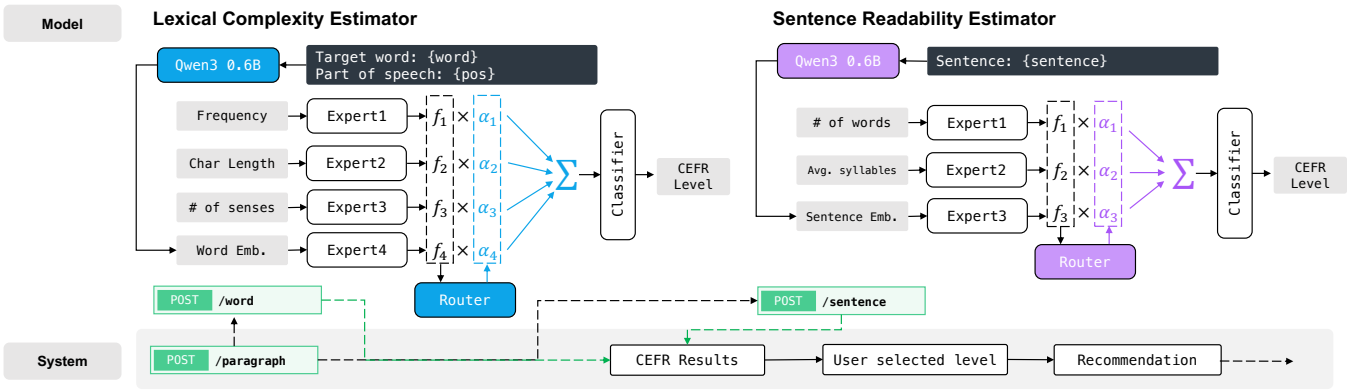

Figure 4: Overview of our interpretable CEFR prediction models. Left: Lexical Complexity Estimator (word-level). Right: Sentence Readability Estimator (sentence-level). Each feature is processed by a small expert network f_i; a data-dependent router predicts weights α that gate the experts and form a weighted sum, which a classifier maps to a CEFR level (A1-C2). Word features: frequency, character length, number of senses, and a contextual embedding from a LoRA-adapted Qwen3 0.6B [72]. Sentence features: number of words, average syllables per word, and a contextual embedding from the same model. The gray callouts show the prompts used to obtain contextual embeddings. The learned α weights provide instance-specific, explanations of which features drove each prediction. The word- and sentence-level CEFR results are combined, then filtered to include only items at or above the user-selected CEFR level, and presented as the final recommendations.

accept the recommendation without a clear decision-making process. Thus, we decided to develop an interpretable model for recommending challenging words and sentences. Another concern was that participants could not weigh the severity of a recommendation (that is, how complex the word or sentence is) without referring to a baseline. Thus, we decided that the recommendation should align with a standard practice. We adopted the Common European Framework of Reference (CEFR) [54] for Languages, a standardized system of six levels (A1, A2, B1, B2, C1, C2) to measure language proficiency. This standard is typically used to prepare learning materials that match the six levels. A1 and A2 indicate basic users, B1 and B2 indicate independent users, and C1 and C2 proficient users. Prior research shows that CEFR is an effective measure to calculate lexical complexity. We propose two interpretable neural models for CEFR level prediction: a Lexical Complexity Estimator and a Sentence Readability Estimator.

6.1.1 Lexical Complexity Estimator. Suppose a document D contain S sentences and W words. Denote the i-th sentence by s_i and the j-th word by w_j. The Lexical Complexity Estimator is defined as the following function f:

$$f : X \rightarrow \{0, 1\}^C$$

where X is the input feature space and C denotes the number of CEFR classes. Our Lexical Complexity Estimator is designed to classify individual words into CEFR levels based on four linguistically meaningful features x_i inspired by prior work [2, 17, 39]: 1) frequency or rarity of the word in English ($freq(w_j)$); 2) length or character count of the word ($|w_j|$); 3) number of senses or polysemy ($poly(w_j)$); and 4) the word embedding of w_j. To compute the word embedding, we use the Qwen3-0.6B [72] model. We provide the word and its part of speech to Qwen3 model in a prompt and take the mean of the final hidden states as the embedding. The model architecture consists of four main components (Figure 4A). First, **feature transformation networks** process each of the four features through dedicated transformation networks, acting as specialized experts:

$$\begin{aligned} f_i &= \text{Expert}_i(x_i) \\ &= \text{LayerNorm}(\text{GELU}(\text{Linear}(\text{LayerNorm}(\text{Linear}(x_i))))) \end{aligned}$$

Here, each expert network transforms a scalar feature into a 512-dimensional representation via two linear layers with LayerNorm [5] and GELU [23] activations.

Second, **dynamic alpha prediction** uses a data-dependent gating mechanism. Instead of fixed feature weights, we use an alpha predictor network that determines the importance of each feature based on the current input:

$$\alpha = \text{softmax}(\text{AlphaPredictor}(\text{concat}(f_1, f_2, f_3, f_4)))$$

Third, **feature fusion** combines the weighted features through element-wise multiplication and summation:

$$h = \sum_{i=1}^{4} (\alpha_i \times f_i)$$

Finally, the fused representation is passed through a final classifier network to produce CEFR class logits.

$$z = \text{Classifier}(h), \quad z \in \mathbb{R}^K \tag{1}$$

$$\hat{y} = argmax_{k \in \{1,\ldots,C\}} z_k \tag{2}$$

The final **classification layer** is a small feed-forward network with two linear layers. It first reduces the feature dimension by half, then normalizes the activations with LayerNorm, and applies the GELU activation function, similar to feature transformation networks. Finally, another linear layer maps the features to the number of output classes.

Table 1: CEFR prediction results (%). Best per column (within each category) in bold.

		Levels F1 (%)						Overall (%)		
Category	Model	A1	A2	B1	B2	C1	C2	Avg. F1	Acc.	Grouped Acc.
Word CEFR	GPT-4o	68.5	40.4	51.8	59.3	41.2	49.5	51.8	52.9	62.3
	Ours	**81.2**	**57.9**	**53.0**	**68.2**	**63.3**	**67.5**	**65.2**	**64.8**	**71.1**
Sentence CEFR	GPT-4o	50.0	59.5	62.0	71.1	65.8	**49.0**	59.6	64.0	69.9
	Ours	**70.6**	**86.3**	**90.0**	**87.6**	**86.8**	28.6	**75.0**	**87.4**	**89.7**

6.1.2 Sentence Readability Estimator. The estimator follows the same functional form for a sentence s_i:

$$g : Y \to \{0, 1\}^C$$

where Y is the sentence-level feature space. This estimator follows the same design principles but is based on three features inspired by prior work [4, 53]: 1) number of words ($|s_i|$); 2) average syllables per word ($syl(s_i)$); and 3) the sentence embedding of s_i. The sentence embedding likewise uses the Qwen3 model. The architecture uses three expert networks and an alpha predictor, with the same fusion and classification approach.

6.1.3 Interpretability. A key advantage of our approach is interpretability through dynamic gating. The alpha weights α provide instance-specific explanations of which features contributed to each prediction. This enables feature-level interpretability (each weight corresponds to a linguistically meaningful feature) and instance-specific explanations (the gating is data-dependent).

6.1.4 Model Training. Both models are trained end-to-end using the PyTorch library for CEFR classification. The Qwen3-based embedding model is trained with a LoRA adaptation. In multi-label scenarios where a word or sentence may belong to multiple CEFR levels simultaneously, we treat each CEFR level as an independent binary classification task and use multi-label cross-entropy loss (`BCEWithLogitsLoss`). The training configuration is as follows: batch size 64, learning rate 2e-4, 10 epochs using the AdamW optimizer [51] on a single NVIDIA A100 GPU.

6.1.5 Datasets. For the sentence dataset, we merged the SCoRE and WikiAuto datasets provided with CEFR-SP [4]. We also used CEFR-SP's predefined split to construct the train and test sets. For the word dataset, following previous work [17], we used CEFR-J [69] and the Octanove Vocabulary Profile [17], and additionally incorporated the Oxford 5000 [55] dataset. In CEFR-J, the word list is based on major English textbooks used in China, Korea, and Taiwan. To reduce label ambiguity, we exclude cases where the same (word, POS) appears in two or more sources but differs by ≥ 2 CEFR levels. We then split the data into training and test sets with a 90%/10% ratio.

6.1.6 Model Evaluation. Table 1 reports F1 by level, overall accuracy, and grouped accuracy (A*=A1,A2, B1, B2, C*=C1,C2). We compare interpretable neural models with GPT-4o on CEFR classification tasks at both the word and sentence levels. The dataset may have up to two levels (either because there are multiple annotators or due to the process of merging datasets), and both levels were regarded as equally reliable. Therefore, during testing, a prediction was counted as correct if it matched either of the annotated labels. The prompts provided to GPT-4o were as follows.

Word–level prediction

```
Task: Predict the CEFR level (A1, A2, B1, B2, C1, C2) of the
target word.
Target word: {word}
Part of speech: {pos}
CEFR Level:
```

Sentence–level prediction

```
Task: Predict the CEFR level (A1, A2, B1, B2, C1, C2) of the
given sentence.
Sentence: {sentence}
CEFR Level:
```

For word-level CEFR prediction, our Lexical Complexity Estimator significantly outperforms GPT-4o on all metrics. The overall average F1 of 65.2% represents a substantial improvement over GPT-4o's 51.8%. Likewise, our model achieves higher overall accuracy (64.8% vs. 52.9%) and grouped accuracy (71.1% vs. 62.3%). These results suggest that our feature-aware architecture captures frequency, length, and sense (polysemy) based lexical difficulty more reliably than a general purpose LLM. This aligns with prior work showing that, in complex word identification, ChatGPT performs poorly on calibrated scoring while task-specific models achieve stronger performance [39].

For sentence-level CEFR prediction, our Sentence Readability Estimator achieves higher F1 scores than GPT-4o on five of the six CEFR levels. GPT-4o is superior only at C2, yet our model still obtains a markedly higher average F1 (75.0% vs. 59.6%). Our model also outperforms on overall accuracy (87.4% vs. 64.0%) and grouped accuracy (89.7% vs. 69.9%). These findings are consistent with prior research showing that fine-tuned models outperform few-shot, prompt-based LLMs in readability assessment [53].

6.1.7 Inference efficiency. To compare runtime speed, we measured the average inference time per example on the test set with batch size 1. Our models were executed on a single NVIDIA A100 GPU, and GPT-4o latencies were measured as end-to-end API round-trip times from the same client machine. The proposed models are substantially faster than GPT-4o on both tasks: for word-level prediction, 0.05 s vs. 0.611 s ($\approx$ 12$\times$ speedup), and for sentence-level prediction, 0.052 s vs. 0.592 s ($\approx$11$\times$ speedup). This advantage stems from the compact, feature-aware architecture, and in practical deployments latency can further decrease with batch processing.

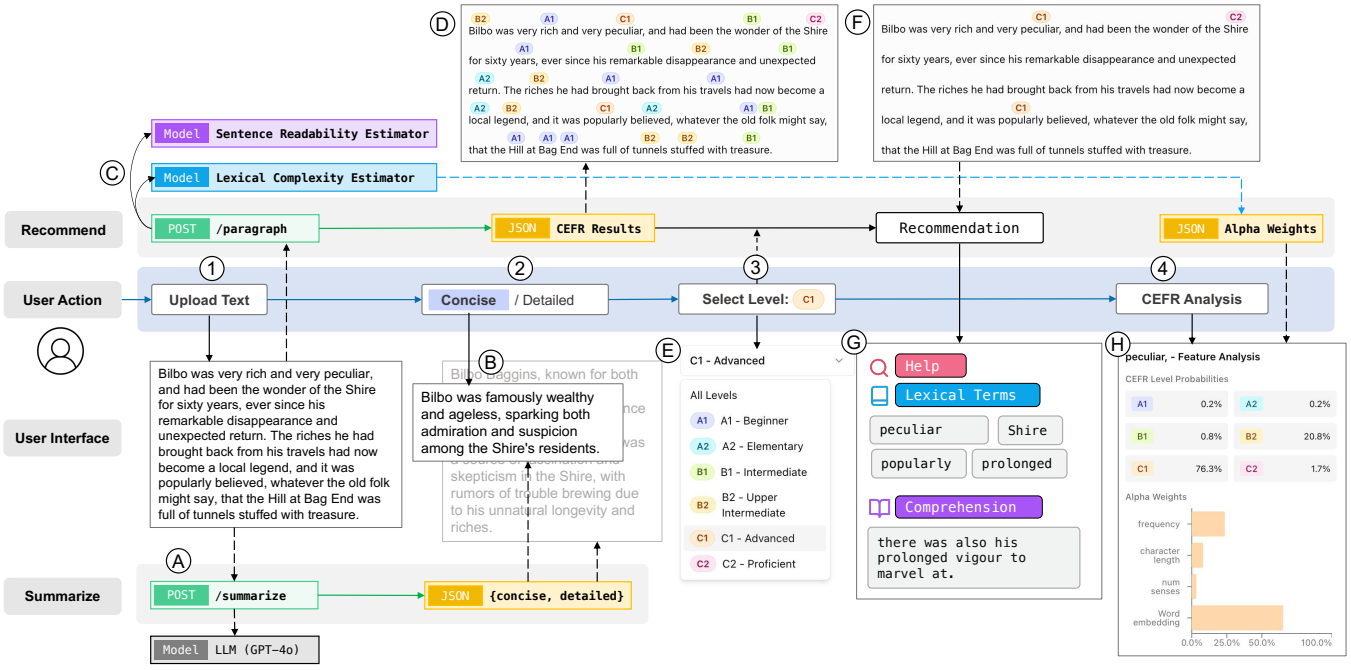

Figure 5: User Actions and System Pipeline for Summarization and Proactive Recommendations. Users first upload a document (①); then choose a summary style, *Concise* or *Detailed* (②); set a target CEFR level (e.g., B2 or C1) (③); and run the CEFR analysis (④). In the system, the backend endpoint `POST /summarize` generates paragraph-aligned summaries (A), which are displayed in a fixed left sidebar at the selected detail level (B). Interpretable CEFR predictors, the Sentence Readability Estimator (SRE) and the Lexical Complexity Estimator (LCE), perform batched inference and serve results (C). The passage view is annotated with CEFR difficulty labels at the token and sentence levels (D), and proactive recommendations surface words and sentences that exceed the user's threshold (E, F). A help panel lists recommended items that are likely to be difficult for the user (G). When the user requests a CEFR decision, an interpretable feature analysis presents level probabilities and compact feature-contribution bars (H).

In summary, the proposed interpretable neural models consistently surpass GPT-4o across both tasks. These results confirm that our specialized, interpretable models are advantageous for CEFR-level classification and are well-suited for deployment in our system.

6.2 Automated Summary

To address DR2 (improving text comprehension), the updated system automatically produces short, paragraph-aligned summaries as soon as a document is uploaded (Figure 5A). These summaries foreground topic sentences and key claims so that EFL readers can quickly grasp the main idea before reading the paragraph in detail. Each summary is anchored to its source paragraph in the main text, minimizing context switching and preserving local coherence during skimming and review. A summary toggle offers two settings (Figure 5②): *concise* shows a short cue focused on the topic sentence, and *detailed* provides a detailed information. This lets readers choose how much help they want while keeping each summary anchored to its paragraph, reducing context switching and supporting the reading strategies.

6.3 Validation

While the CEFR model will improve interpretability of the identification task, we are still relying on LLMs to generate the explanation for grammar because of its generational capabilities. To improve the reliability of the explanation, we added a validation mechanism along with our interpretable models.

After the primary LLM generates an explanation (lexical term, comprehension note, or grammar analysis), a second LLM (i.e., validator) checks whether the response (i) is grounded in the selected span or paragraph, (ii) matches the requested assistance type, and (iii) is linguistically correct and non-contradictory. The validator returns a binary decision (VALID/INVALID) and a brief rationale. Validated items are visually indicated in the UI (Figure 6D). This dual-LLM scheme does not eliminate all errors but makes trust cues explicit to users with modest added latency. See Table 5 for results.

6.4 UI and Usability Improvements

To enhance usability, we refined the interface with several improvements. We made two changes to make the interface adaptable to different proficiency levels. First, readers can set a target difficulty level from a dropdown with different *recommended CEFR levels* (Figure 5E). Second, we introduced a toggle button that allows users

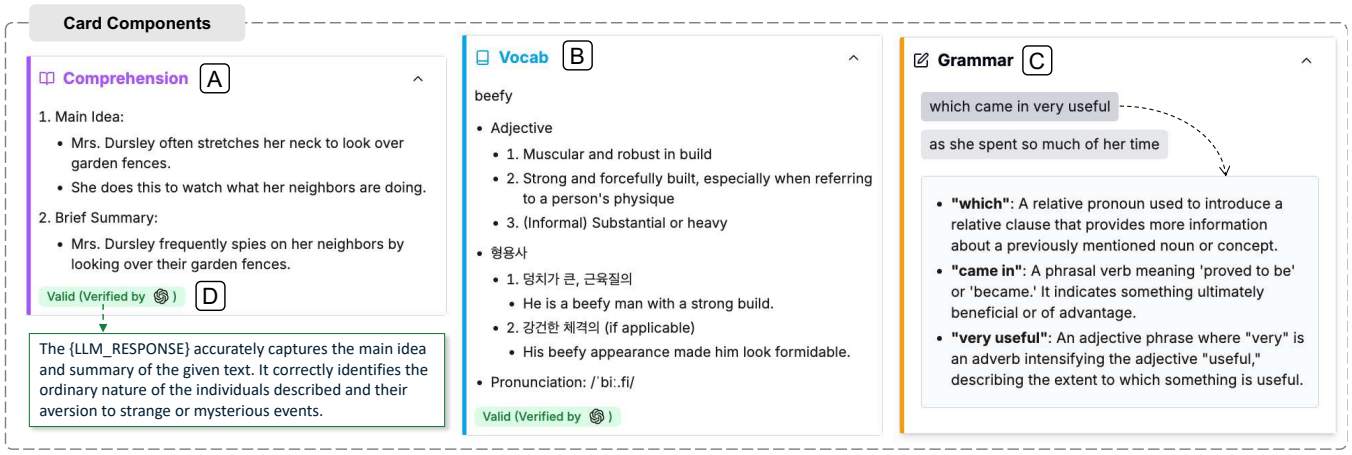

Figure 6: Detailed explanation of components to help EFL academic reading. We provide three components to help EFL researchers in academic reading. They are: (A) vocabulary (lexical terms), (B) comprehension, and (C) grammar. In (A), we present the definition of the selected keyword, or the idiom, usage, as well as its meaning in Korean. In (B), we help users understand the text by analyzing the main idea of the user-selected text with respect to the entire context. In condition (C), we provide grammatical explanations about the user-selected text at the phrase level.

to control the level of details in the summaries (either detailed or concise).

The updated interface is more proactive in providing guidance. Uploaded documents are automatically processed to produce short, paragraph-aligned summaries displayed in a fixed left sidebar (Figure 5B). In addition, the system recommends potentially difficult words and sentences (Figure 5G). The recommended words and sentences are highlighted with three colors—yellow for grammar, blue for vocabulary, and purple for comprehension. Interacting with a highlighted item reveals its CEFR label, and hovering over it presents a compact attribution chart that indicates which features contributed the most to the prediction of difficulty level (Figure 5H).

To minimize the verbosity of the grammar module, we prompted the LLM to make the explanation brief. Finally, validation by the second LLM is visually indicated in the UI: green icons mark responses confirmed by the second LLM, and hovering over the icon reveals the rationale behind the decision (Figure 6D).

Original features from Reading.help-Lite are preserved: users may still highlight spans of interest and obtain detailed explanations on demand (Figure 2D).

6.5 System Architecture and Implementation

The system comprises two components: a web-based interface for reading assistance (Reading.help) and a backend that consists of CEFR model and LLMs. Here, we describe the system architecture (Figure 7) and analysis pipeline.

Document preprocessing. When a document is uploaded, the backend runs the Lexical Complexity Estimator (LCE) for words and the Sentence Readability Estimator (SRE) for sentences to estimate CEFR levels across the document.

CEFR Predictors and serving. The backend of Reading.help integrates interpretable models with a validation mechanism to deliver reliable assistance at scale. Both interpretable models are implemented in PyTorch and exposed to the interface via a FastAPI [5] server. To reduce latency, the pipeline executes *batched* inference over tokens and sentences rather than per-item calls.

Processing workflow. The pipeline has five stages: (1) segment uploaded documents into paragraphs; (2) split into sentences and tokens; (3) run batched LCE and SRE inference to obtain CEFR estimates; (4) reassemble predictions into the original text structure; and (5) filter predictions according to the user's target CEFR level (or history-inferred default), ensuring that only items above the chosen threshold are surfaced in the interface.

Response generation and validation. When the primary LLM generates a candidate explanation for a selected span, a secondary validator LLM evaluates it using a structured prompt containing {context_paragraph, selected_span, component_type, explanation}. The validator returns a JSON object {`valid`, `rationale`}, which the decision logic parses to annotate the item. The interface renders this annotation by attaching an indicator and, when available, surfacing the rationale (see Figure 7). Although not perfect, this layered approach aligns with established validation techniques in AI systems [20].

7 Evaluation

Our evaluation focuses on three aspects: (1) understanding how Reading.help helps EFL readers, (2) the effectiveness of recommendations based on CEFR models, and (3) the robustness of LLM validation. For this, we again conducted a user study with 5 participants, which is described in Table 2. After detailing the study procedure, we report our results from the study.

[5] https://fastapi.tiangolo.com/

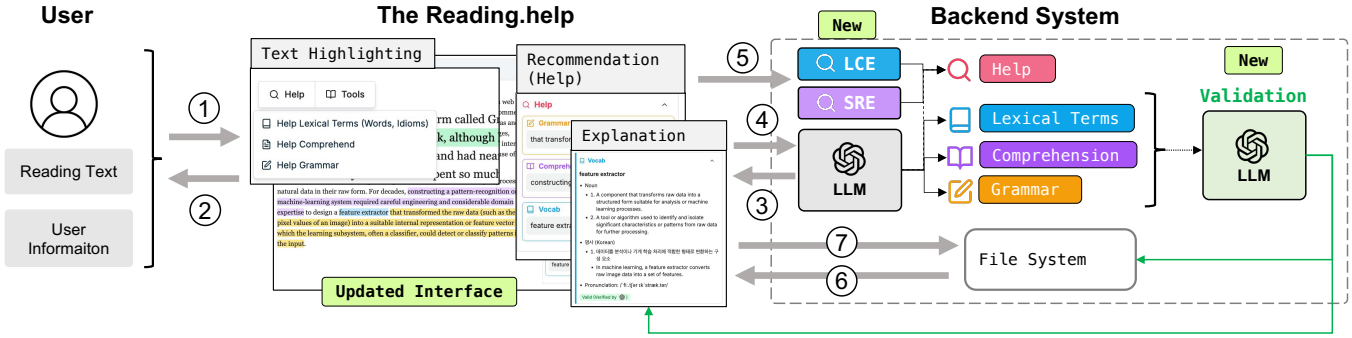

Figure 7: System overview. The mechanism of our tool is composed of an interface (the READING.HELP) and a back-end system that provides and stores responses. ① The user delivers their demographic information, as well as the part of text that they do not know to the back-end system via the READING.HELP. Then, READING.HELP receives the input and delivers it to the LLMs ④. Afterwards, LLMs provide the responses to the READING.HELP ③, and they are delivered to the user ②. ⑤ When a document is uploaded, the backend also runs the Lexical Complexity Estimator (LCE) for words and the Sentence Readability Estimator (SRE) for sentences to estimate CEFR levels across the document. When a new input is uploaded and new response comes, then the information existing in the interface is saved to the file system in the back-end ⑦, and can be called back by the interface upon the user's request ⑥. Note that the new validation mechanism is added in READING.HELP. The validation takes place prior to sending the data to the READING.HELP interface from the backend server.

7.1 Study on EFL readers

We conducted a user study with five EFL readers, all at or above the undergraduate level, who received their entire education, including English instruction, in South Korea. All participants reported Korean as their most proficient language for both speaking and writing, and none majored in English-related fields. The study consisted of two phases, with the main session lasting approximately 50 minutes, followed by a 10-minute post-interview.

Phase 1. In Phase 1, participants were provided with five separate passages as plain text, without any reading support tools. Each passage consisted of four paragraphs and ranged from 340 to 370 words. Passages were selected from the of two novels, two *Economist* articles, and one deep learning research paper to evaluate our tool's applicability across various text types. Participants were asked to read all the passages for 30 minutes and highlight any sections they found difficult.

Phase 2. In Phase 2, participants were introduced to the web-based READING.HELP tool. After a brief five-minute tutorial, participants revisited the five passages using READING.HELP. They were asked to interpret text segments they had previously marked as unclear and use READING.HELP to resolve the comprehension issue. Each participant then detailed the issue, assessed whether the tool provided effective support, and explained how it addressed the problem. The session ended after 20 minutes. Then, we conducted a brief post-interview to gather additional insights on their experiences and READING.HELP's effectiveness. Their usages of modules in READING.HELP are summarized in Figure 8.

7.2 Results

We report our findings in three parts. First, we describe general usage patterns of participants. Next we evaluate the effectiveness of the recommendation of READING.HELP. Finally, we assess the validity of the LLM.

Table 2: User study demographics. A total of 5 participants with backgrounds in medicine, computer science, and engineering took part in the study. All participants were above the age of 25 and held at least a bachelor's degree in their respective fields. They were all Korean nationals who regularly engage with English academic texts as part of their professional or academic work. We present the participants' gender, age, education (Edu.), and job title.

ID	Gender	Age	Edu.	Job Title
P1	Male	30	M.D.	Medical Doctor
P2	Female	29	M.D.	Medical Doctor
P3	Female	27	M.S.	Ph.D. student in CS
P4	Female	26	M.S.	Ph.D. student in CS
P5	Male	29	B.S.	Software engineer

Usage patterns. Overall, 5 participants highlighted a total of 101 challenging parts of the text across five texts. Figure 8 shows an overview of which modules participants used in READING.HELP. We find that participants use vocabulary models the modules the most, followed by comprehension, and grammar modules. Participants' comments indicate that the comprehension module aided their understanding of uncertain sentences. It explained not only key terms but also provided background context for the entire sentence. The grammar component clarified ambiguous interpretations, and two participants noted that the paragraph summary helped them read and follow the text more swiftly.

Assessment of CEFR model recommendations. We evaluate interpretable CEFR prediction models on their ability to recommend text that readers might not know. Table 3 reports recall, precision,

Table 3: Precision, recall, and F1 scores of keywords recommended by interpretable CEFR prediction models, with respect to the words and sentences that the participants in our study found it difficult. The analysis of the table is described in § 7.2.

	Recall (%)						Precision (%)						F1 Score (%)					
	R1	R2	R3	R4	R5	Avg.	R1	R2	R3	R4	R5	Avg.	R1	R2	R3	R4	R5	Avg.
B2 ≥	**75.00**	**81.25**	**87.50**	**94.44**	**100**	**87.64**	18.37	19.40	19.44	20.24	**14.89**	18.47	29.51	**31.33**	31.82	33.33	**25.93**	30.38
C1 ≥	38.46	27.27	50.00	81.25	42.86	47.97	**35.71**	**31.58**	**45.00**	**65.00**	11.54	**37.77**	**37.04**	29.27	**47.37**	**72.22**	18.18	**40.82**

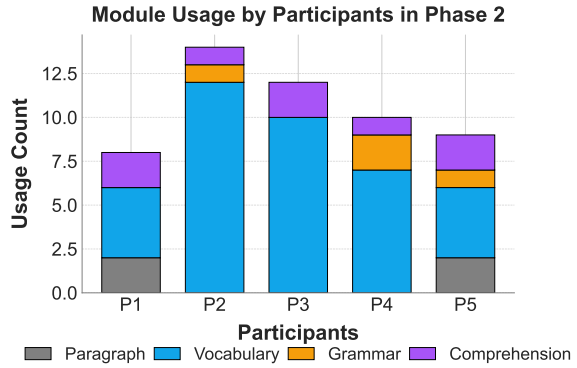

Figure 8: Usage of modules in READING.HELP during the experiment by participants (P1-P5). We find that participants who used READING.HELP used the tool mostly for searching for vocabulary.

and F1 across five readings (R1–R5) under two difficulty thresholds, ≥**B2** and ≥**C1**. Because participants could freely highlight text without specifying whether they were unsure about word meaning or overall context, we employed token-based Jaccard similarity to measure semantic overlap between each recommendation and the union of user highlights. For each AI recommendation, we tokenized text by word boundaries and computed the Jaccard coefficient (intersection over union of word sets) against all user-highlighted content. A match was determined when the similarity exceeded $\theta = 0.3$.

At the **B2** ≥ threshold, the model achieves very high recall (75.00–100%, Avg. 87.64%) with low precision (14.89–20.24%, Avg. 18.47%), yielding an average F1 of 30.38%. At the stricter **C1** ≥ threshold, precision improves (11.54–65.00%, Avg. 37.77%) while recall is lower and more variable (27.27–81.25%, Avg. 47.97%), resulting in a higher average F1 of 40.82% with a peak of 72.22% on R4. These results indicate a clear balance between coverage and specificity. The B2 ≥ threshold favors coverage by recovering most difficult items that participants highlighted. The C1 ≥ threshold favors specificity by reducing false positives and improving overall balance as reflected in the higher F1.

Qualitative assessment of CEFR model recommendations. To better understand the strengths and limitations of our AI-based recommendation system, we conducted a pattern analysis using data from three reading passages: Reading 1 (R1), Reading 3 (R3), and Reading 5 (R5). We compared the language features that EFL readers highlighted as confusing with those that the system automatically recommended.

Table 4 shows the examples of keywords/phrases that the CEFR models predicted to be what the EFL readers may not know. We identify three patterns about CEFR models based on the analysis of keywords that are detected and missed. This comparative analysis revealed three dominant types of learner confusion: rare or invented terms (C1), idiomatic and figurative expressions (C2), and polysemous common words (C3).

First, under CEFR-based recommendation (>B2), the system retrieves many context-related terms but still misses a few, depending on text type. It correctly surfaced items such as *Grunnings* and *tawny* (R1), and from R3: *deterrence*, *spectacle*, *calculus*, *arsenal*, and *flashpoint*; R5 yielded *motifs* and *detection*. In contrast, nonce or short forms like *unDursleyish* (R1) and *nukes* (R3) were "missed." Given that *unDursleyish* lacks a stable lexical entry, which makes CEFR estimation uncertain, we heuristically assign it B2; both terms therefore fall at or below our >B2 gate and are not recommended by design.

Second, for idioms and figurative language, the system captured expressions such as *didn't hold with such nonsense* (R1), *toyed with*, *The idea has gone from fringe to mainstream*, and *may be an explosive political spectacle* (R3). The phrase *talked down* (R3) was not captured, but it is B1 and thus intentionally filtered out under the current threshold.

Third, the system struggled with polysemous, high-frequency words whose meanings shift in context. It identified *dull* (R1) and, in R3, *fringe* and *reassure*, but it missed *bear* (B2) in "They didn't think they could bear it if anyone found out about the Potters." Because B2 items sit at or below our recommendation cutoff (>B2), this miss does not affect recommendations; however, it highlights that lower-CEFR words can still be confusing when used in marked senses or collocations.

Overall, CEFR gating usefully reduces oversuggestion and focuses attention on clearly challenging (C1–C2) items, especially domain-specific and idiomatic expressions. A remaining limitation is sensitivity to contextually tricky but lower-CEFR words; augmenting CEFR-based filtering with sense-aware cues could better flag these cases.

Assessment of LLM Validity. To identify READING.HELP's response quality, three English experts were recruited to evaluate a total of 300 examples. Responses deemed error-free were labeled "Valid," and those containing errors "Invalid,"; experts recorded the error types focusing on hallucinations and incorrect explanations and provided detailed justifications for each judgment. After this human-led evaluation, a self-validation process was conducted using an LLM, and we compared the results.

As shown in Table 6, the human experts' assessments revealed an overall valid rate of 88% for vocabulary, 93% for comprehension,

Table 4: Confusion pattern types and examples of Reading.help recommendations that matched (or unmatched) actual EFL learner confusions.

Confusion Type	Explanation	Type	Detected	Missed
C1: Contextual terms	Uncommon, technical, or domain-specific terms	R1	*Grunnings* / *tawny*	*unDursleyish*
		R3	*deterrence* / *spectacle* / *calculus* / *arsenal* / *flashpoint*	*nukes*
		R5	*motifs* / *detection*	-
C2: Idioms & Figurative	Metaphorical or non-literal expressions	R1	*didn't hold with such nonsense*	—
		R3	*toyed with* / *The idea has gone from fringe to mainstream* / *may be an explosive political spectacle*	*talked down*
C3: Polysemous words	Words with multiple context-sensitive meanings	R1	*dull*	*bear*
		R3	*fringe* / *reassure*	—

and 90% for grammar. Vocabulary emerged as the least accurate component, with 12% of the responses flagged for errors. Incorrect translation of vocabulary accounted for 8 cases. Upon rechecking these eight cases, six involved semantically equivalent Korean paraphrases, whereas two were judged completely incorrect. Hence, while there exist some minor errors, we want to highlight that true semantic errors were rare. Comprehension errors (7% invalid) mainly involved reasoning that extended beyond the scope of the given sentences, while grammar errors (10% invalid) frequently related to incorrect parts of speech.

LLM's self validation vs. human validation. The LLM's self-validation rated the responses as 93% Valid for vocabulary, 97% Valid for comprehension, and 96% valid for Grammar. According to the LLM, vocabulary mistakes (7% invalid) were often due to fabricated or irrelevant contexts and mistranslations, while comprehension issues (3% invalid) stemmed from fabricated details beyond the original text. Grammar errors (4% invalid) centered on incorrect verb tenses. Except for omissions of Korean translations in Vocabulary, the LLM's error diagnoses largely aligned with those of the human experts.

In addition to validity rates, Table 5 presents accuracy metrics for vocabulary, comprehension, and grammar tasks by self-validation. All metrics (Precision, Recall, F1 Score) were calculated using 'valid' as the positive class, with human expert evaluations serving as the ground truth.

Comprehension shows the highest agreement with human judgments. Vocabulary is similar to grammar and slightly higher. Overall, these findings indicate that LLMs and humans exhibit similar validation behavior.

7.3 Expert feedback

After the experiment, we asked two domain experts, one high-school English teacher, and another expert who is a certified English teacher, but working at a startup on online English education company. We first asked them about the effectiveness of the tool, and then about the ideal role of a reading assistant for self-studying purposes for EFL readers. Below we describe their opinions.

Effectiveness of Reading.help. To begin with, we identify four benefits of using our tool. A primary advantage is its capacity to mitigate text complexity issues and enhance comprehension. Consequently, it helps the user to grasp not only the details but also the overall structure of the text. One of the experts had never encountered such experience while reading, and expressed the willingness to test reading the entire fiction using Reading.help. The second benefit is its ability to provide recommendations regarding topics that the user may find challenging. It is acknowledged that from a learner's perspective, it is not always easy to know what one does not know, and such recommendations help address this issue. Third, a significant advantage is the freedom to make unlimited requests at any time without pressure. Since Reading.help is a machine and not a human, not only does it respond to requests synchronously, but the user is also free to ask as many questions as desired without interference. Fourth, it provides information tailored to the user's English proficiency level. Both experts argue that they expect such a tool to offer adaptive feedback, which is a key attribute of an effective human tutor.

Table 5: Accuracy metrics for Vocabulary, Comprehension, and Grammar tasks by Self-Validation. All metrics (precision, recall, F1) were calculated using 'valid' as the positive class, with human expert evaluations serving as the ground truth.

	Precision	Recall	F1 Score	Accuracy
Vocabulary	91.4	96.59	93.92	89
Comprehension	93.81	97.85	95.79	92
Grammar	90.63	96.67	93.55	88

Table 6: Comparison of valid and invalid response rates (in %) across Vocabulary, Comprehension, and Grammar, as evaluated by both human experts and the LLM.

	Vocab		Comprehension		Grammar	
	Valid	Invalid	Valid	Invalid	Valid	Invalid
Human Expert	88%	12%	93%	7%	90%	10%
LLM (GPT-4o)	93%	7%	97%	3%	96%	4%

Improvement areas needed on Reading.help. The experts also identified three issues that must be addressed to improve service for EFL readers. First, they noted that the tool is effective only for individuals who are highly motivated to self-study English, rather than for those lacking sufficient motivation. To attract less motivated users, the tool should be made more accessible. For example, by offering additional recommendations and visual highlights. Second, the tool is not fully tailored to individual users. It could provide a more detailed analysis of each learner's condition, offer an assessment of their English proficiency, and deliver explanations that are better customized to their needs. Although the current interface incorporates some of these features, both experts did not consider them sufficient. Finally, because the content is generated using LLMs, teachers acknowledge that while it can produce interesting material, verifying its accuracy remains challenging. In this regard, teachers expressed the need for a tool that they can use without having to worry about constant validation.

8 Discussions

We describe the discussions based on the observation of the results. To begin with, we interpret the results (§ 8.1). Then, we discuss, in detail, the lessons learned from the study (§ 8.2, § 8.3, and § 8.4.)

8.1 Implication of Results

Below, we present the implications from two parts: (1) how the tool is used, and the CEFR model's recommendation and validation performances of LLMs.

From the results, we find out that while reading a 400-word script, they most of the time look for vocabulary, on 5, to 6 times on average, and once or twice in comprehension. We also find that occasionally, they used grammar and looked for paragraph summaries. This represents that in general, for EFL readers to fluently understand the text as well as the context (see § 2.1), it requires a lot of effort. This showcases the utility of providing proactive guidance for EFL readers.

While both experts and participants acknowledged the effectiveness of the LLMs, there is still some issues on the validation of the contents. For this, we assess our system from two ways (1) how well their recommendations adhere to those of humans, and (2) how factual the responses are. Our recommendation provides From our results, we observed approximately 37.77% precision when recommending a few items (see Table 3). However, by increasing the number of recommendations, the tool can capture a larger portion of the information that EFL readers may not know. This is supported by the high recall rate—up to 87.64%—observed in the B2 $\geq$ configuration in Table 3. In summary, a recommendation strategy similar to the B2 $\geq$ configuration can facilitate the reading process by suggesting many keywords that the user may not know.

8.2 Designing Adaptiveness for EFL Learning

A new level of adaptiveness, as mentioned by the experts, can deliver more precise and efficient support for EFL readers. A developed version of Reading.help may enlighten the users with faster, proactive guidance on unfamiliar word meanings, grammar, and comprehension. Both the language of guidance (e.g., level of explanation, text organization) and the type of feedback can be personalized based on analysis of a learner's accumulated skills. Such enhancements can accelerate reading and build confidence.

However, we should also be aware of two potential dangers: extra mental load and over-reliance. Excessive explanation can overwhelm the users, and replace the productive effort that builds skill. This is suggested by how participants took advantage of detailed grammatical information in Reading.help (see § 7.2—participants were overwhelmed with the amount of text, and later on did not again refer to the part). We therefore recommend brief and lightweight by default. If providing details is necessary, then first hide detailed information, then show details only if the user chooses to.

Over-reliance is the second concern. The ultimate goal of reading tools like Reading.help is in increasing the user's English skills, so that reliance on the tool decreases as proficiency grows. Consistent with the "no pain, no gain" principle in learning [13], guidance should not be provided in a way that requires least amount of effort from the users. Instead, systems should adopt policies that deliberately preserve productive struggle to some extent. For example, withholding hints for terms that have been encountered repeatedly or gradually fading support. Success for tools like Reading.help should be measured not only by faster reading but also by growing independence over time.

8.3 Engagement Strategies for EFL Readers

Domain experts emphasized that higher engagement can sustain attention and improve learning outcomes for EFL readers. One way to increase engagement is to actively attract the user's interest. In data visualization, for example, researchers have studied visual embellishment [7, 57]—adding pictures or icons to charts to make them stand out. These additions can draw attention and encourage exploration, but prior work shows they can also reduce how clearly the data is understood. This trade-off has led some to call overly decorative elements *chartjunk*. The same risk applies to reading assistance tools: methods that focus mainly on catching the eye may not improve understanding or skill development. We therefore treat interest-driven cues as tools to be used sparingly and with clear task alignment. For example, tools themselves can support motivation in the form of timely, lightweight texts, or progress bars [33]. The key is to channel attention toward the learning objective rather than to compete for it.

An alternative is to raise engagement by removing distraction. Rather than competing for attention, we envision an immersive [45, 46, 61], purpose-built environment for deep reading, paired with a human-like virtual English tutor. Within this controlled space, the tutor offers just-in-time, ad-hoc clarification, and examples tailored to the reader's current difficulty, without interruptions that could potentially disturb comprehension. The system helps by making the reader's focus remains on the text by minimizing opportunities for attention shifts, and at the same time providing visible and accessible support. We identify various challenges along this strategy, such as how to calibrate guidance and pace interventions, and validation.

To that end, we propose a deep assessment of these two strategies-interest attraction versus distraction-free immersion, for future development of foreign language education using computing devices.

8.4 Augmenting, Not Replacing: The Role of LLMs in English Reading Education

With the power of generative models, we can envision an English reading assistant that has absorbed a vast range of texts, can detect the specific difficulties a learner faces, and can deliver content at the learner's level of proficiency. Such a tool could also generate and grade questions to check comprehension, raising the question: What would the ideal English reading assistant look like, and could it fully replace human teachers?

Despite these possibilities, there remain essential roles that only humans can fulfill in teaching English reading. Teachers are accountable for learner progress in a way that no generative models can be. They provide motivation, encouragement, and adaptive care—especially for beginners or those with low self-efficacy—when learners tire or lose momentum. Moreover, language learning is not only about mastering reading skills but also about building collaboration and social interaction, dimensions that require authentic human presence.

In this light, we envision an LLM-powered English reading assistant that augments rather than replaces teachers. For any text a learner brings, the assistant could offer adaptive support and remain available anytime and anywhere, even when instructors are not present. At the same time, when teachers are available, the tool could enhance their pedagogical decision-making and amplify their ability to guide students. Looking ahead, it is crucial to design such tools to advance English learning while preserving and strengthening the agency of both teachers and learners—an important direction for future work.

9 Conclusion

In this work, we developed a tool, called Reading.help to help EFL readers' reading processes. In doing so, we utilized LLMs and interpretable neural networks. From our initial design, Reading.help-Lite, we conducted a pilot study, and based on it we developed a tool that effectively helps EFL readers, the Reading.help. From our experiment, we also found that CEFR models are capable of effectively recommending parts that EFL readers do not know. Moreover, LLMs possess abilities, though limited, to verify if the contents are incorrect.

Acknowledgments

We thank the anonymous reviewers for their precious feedback. We also thank Jinyoung Lee and Minhye Park for their precious feedback regarding the work.

References

[1] Afiliani Afiliani, Inggrit O Tanasale, and Helena M Rijoly. 2024. The Use of Google Translate in the Translation Class at English Education Study Program Pattimura University. *Journal of English Language Teaching and Linguistics* 9, 1 (2024), 95–109.

[2] Desislava Aleksandrova and Vincent Pouliot. 2023. Cefr-based contextual lexical complexity classifier in english and french. In *Proceedings of the 18th Workshop on Innovative Use of NLP for Building Educational Applications (BEA 2023)*. 518–527.

[3] Riku Arakawa, Hiromu Yakura, and Sosuke Kobayashi. 2022. VocabEncounter: NMT-powered Vocabulary Learning by Presenting Computer-Generated Usages of Foreign Words into Users' Daily Lives. In *Proceedings of the ACM Conference on Human Factors in Computing Systems (CHI)*. Association for Computing Machinery, New York, NY, USA, Article 6, 21 pages. https://doi.org/10.1145/3491102.3501839

[4] Yuki Arase, Satoru Uchida, and Tomoyuki Kajiwara. 2022. CEFR-based sentence difficulty annotation and assessment. *arXiv preprint arXiv:2210.11766* (2022).

[5] Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E. Hinton. 2016. Layer Normalization. arXiv:1607.06450 [stat.ML] https://arxiv.org/abs/1607.06450

[6] Sriram Karthik Badam, Zhicheng Liu, and Niklas Elmqvist. 2019. Elastic Documents: Coupling Text and Tables through Contextual Visualizations for Enhanced Document Reading. *IEEE Transactions on Visualization and Computer Graphics* 25, 1 (Jan 2019), 661–671. https://doi.org/10.1109/TVCG.2018.2865119

[7] Scott Bateman, Regan L. Mandryk, Carl Gutwin, Aaron Genest, David McDine, and Christopher Brooks. 2010. Useful junk? the effects of visual embellishment on comprehension and memorability of charts. In *Proceedings of the ACM Conference on Human Factors in Computing Systems*. Association for Computing Machinery, New York, NY, USA, 2573–2582. https://doi.org/10.1145/1753326.1753716

[8] Thomas H. Carr. 1981. Building theories of reading ability: On the relation between individual differences in cognitive skills and reading comprehension. *Cognition* 9, 1 (1981), 73–114. https://doi.org/10.1016/0010-0277(81)90015-9

[9] Joseph Chee Chang, Amy X. Zhang, Jonathan Bragg, Andrew Head, Kyle Lo, Doug Downey, and Daniel S. Weld. 2023. CiteSee: Augmenting Citations in Scientific Papers with Persistent and Personalized Historical Context. In *Proceedings of the ACM Conference on Human Factors in Computing Systems (CHI)*. Association for Computing Machinery, New York, NY, USA, Article 737, 15 pages. https://doi.org/10.1145/3544548.3580847

[10] Xiang "Anthony" Chen, Chien-Sheng Wu, Lidiya Murakhovs'ka, Philippe Laban, Tong Niu, Wenhao Liu, and Caiming Xiong. 2023. Marvista: Exploring the Design of a Human-AI Collaborative News Reading Tool. *ACM Trans. Comput.-Hum. Interact.* 30, 6, Article 92 (sep 2023), 27 pages. https://doi.org/10.1145/3609331

[11] Jinho Choi, Sanghun Jung, Deok Gun Park, Jaegul Choo, and Niklas Elmqvist. 2019. Visualizing for the Non-Visual: Enabling the Visually Impaired to Use Visualization. *Computer Graphics Forum* 38, 3 (2019), 249–260. https://doi.org/10.1111/cgf.13686

[12] Pramod Chundury, Yasmin Reyazuddin, J. Bern Jordan, Jonathan Lazar, and Niklas Elmqvist. 2024. TactualPlot: Spatializing Data as Sound Using Sensory Substitution for Touchscreen Accessibility. *IEEE Transactions on Visualization and Computer Graphics* 30, 1 (Jan 2024), 836–846. https://doi.org/10.1109/TVCG.2023.3326937

[13] Anique BH de Bruin, Felicitas Biwer, Luotong Hui, Erdem Onan, Louise David, and Wisnu Wiradhany. 2023. Worth the effort: The start and stick to desirable difficulties (S2D2) framework. *Educational Psychology Review* 35, 2 (2023), 41.

[14] Krismalita Sekar Diasti, Cecilia Titiek Murniati, and Heny Hartono. 2023. The Implementation of KWL Strategy in EFL Students' Reading Comprehension. *Journal of English Teaching* 9, 2 (2023), 176–185.

[15] Sidney D'Mello, Kristopher Kopp, Robert Earl Bixler, and Nigel Bosch. 2016. Attending to Attention: Detecting and Combating Mind Wandering during Computerized Reading. In *Proceedings of the Extended Abstracts on Human Factors in Computing Systems (CHI)*. Association for Computing Machinery, New York, NY, USA, 1661–1669. https://doi.org/10.1145/2851581.2892329

[16] Raymond Fok, Hita Kambhamettu, Luca Soldaini, Jonathan Bragg, Kyle Lo, Marti Hearst, Andrew Head, and Daniel S Weld. 2023. Scim: Intelligent Skimming Support for Scientific Papers. In *Proceedings of the International Conference on Intelligent User Interfaces (IUI)*. Association for Computing Machinery, New York, NY, USA, 476–490. https://doi.org/10.1145/3581641.3584034

[17] Yoshinari Fujinuma and Masato Hagiwara. 2021. Semi-supervised joint estimation of word and document readability. *arXiv preprint arXiv:2104.13103* (2021).

[18] Aimen Gaba, Vidya Setlur, Arjun Srinivasan, Jane Hoffswell, and Cindy Xiong. 2023. Comparison Conundrum and the Chamber of Visualizations: An Exploration of How Language Influences Visual Design. *IEEE Transactions on Visualization and Computer Graphics* 29, 1 (Jan 2023), 1211–1221. https://doi.org/10.1109/TVCG.2022.3209456

[19] Katy Ilonka Gero, Tao Long, and Lydia B Chilton. 2023. Social Dynamics of AI Support in Creative Writing. In *Proceedings of the ACM Conference on Human Factors in Computing Systems (CHI)*. Association for Computing Machinery, New York, NY, USA, Article 245, 15 pages. https://doi.org/10.1145/3544548.3580782

[20] Zhibin Gou, Zhihong Shao, Yeyun Gong, Yelong Shen, Yujiu Yang, Nan Duan, and Weizhu Chen. 2024. CRITIC: Large Language Models Can Self-Correct with Tool-Interactive Critiquing. arXiv:2305.11738 [cs.CL] https://arxiv.org/abs/2305.11738

[21] Arthur C. Graesser, Nicholas L. Hoffman, and Leslie F. Clark. 1980. Structural components of reading time. *Journal of Verbal Learning and Verbal Behavior* 19, 2 (1980), 135–151. https://doi.org/10.1016/S0022-5371(80)90132-2

[22] Jeffrey Heer, Matthew Conlen, Vishal Devireddy, Tu Nguyen, and Joshua Horowitz. 2023. Living Papers: A Language Toolkit for Augmented Scholarly Communication. In *Proceedings of the ACM Symposium on User Interface Software and Technology (UIST)*. Association for Computing Machinery, New York, NY, USA, Article 42, 13 pages. https://doi.org/10.1145/3586183.3606791

[23] Dan Hendrycks and Kevin Gimpel. 2023. Gaussian Error Linear Units (GELUs). arXiv:1606.08415 [cs.LG] https://arxiv.org/abs/1606.08415

[24] Eli Hinkel. 2003. *Teaching academic ESL writing: Practical techniques in vocabulary and grammar.* Routledge.

[25] Eric Donald Hirsch. 2003. Reading comprehension requires knowledge of words and the world. *American educator* 27, 1 (2003), 10–13.

[26] Kay Hong-Nam and Larkin Page. 2014. Investigating metacognitive awareness and reading strategy use of EFL Korean university students. *Reading Psychology* 35, 3 (2014), 195–220.

[27] Md Naimul Hoque, Md Ehtesham-Ul-Haque, Niklas Elmqvist, and Syed Masum Billah. 2023. Accessible Data Representation with Natural Sound. In *Proceedings of the ACM CHI Conference on Human Factors in Computing Systems (CHI)*. Association for Computing Machinery, New York, NY, USA, Article 826, 19 pages. https://doi.org/10.1145/3544548.3581087

[28] Md Naimul Hoque, Tasfia Mashiat, Bhavya Ghai, Cecilia D. Shelton, Fanny Chevalier, Kari Kraus, and Niklas Elmqvist. 2024. The HaLLMark Effect: Supporting Provenance and Transparent Use of Large Language Models in Writing with Interactive Visualization. In *Proceedings of the ACM Conference on Human Factors in Computing Systems (CHI)*. Association for Computing Machinery, New York, NY, USA, Article 1045, 15 pages. https://doi.org/10.1145/3613904.3641895

[29] Md Naimul Hoque, Sungbok Shin, and Niklas Elmqvist. 2024. Visualization for human-centered AI tools. *arXiv preprint arXiv:2404.02147* (2024).

[30] Tsui-Chun Hu, Yao-Ting Sung, Hsing-Huang Liang, Tsung-Jen Chang, and Yeh-Tai Chou. 2022. Relative roles of grammar knowledge and vocabulary in the reading comprehension of EFL elementary-school learners: Direct, mediating, and form/meaning-distinct effects. *Frontiers in psychology* 13 (2022), 827007.

[31] Takumi Ito, Naomi Yamashita, Tatsuki Kuribayashi, Masatoshi Hidaka, Jun Suzuki, Ge Gao, Jack Jamieson, and Kentaro Inui. 2023. Use of an AI-powered Rewriting Support Software in Context with Other Tools: A Study of Non-Native English Speakers. In *Proceedings of the ACM Symposium on User Interface Software and Technology (UIST)*. Association for Computing Machinery, New York, NY, USA, Article 45, 13 pages. https://doi.org/10.1145/3586183.3606810

[32] Yuko Iwai. 2008. The perceptions of Japanese students toward academic English reading: implications for effective ESL reading strategies. *Multicultural Education* 15, 4 (2008), 45–50.

[33] Daeun Jeong, Sungbok Shin, and Jongwook Jeong. 2025. Conversation Progress Guide: UI System for Enhancing Self-Efficacy in Conversational AI. In *Proceedings of the ACM Conference on Human Factors in Computing Systems*. Association for Computing Machinery, New York, NY, USA, Article 180, 11 pages. https://doi.org/10.1145/3706598.3714222

[34] Nikhita Joshi and Daniel Vogel. 2024. Constrained Highlighting in a Document Reader can Improve Reading Comprehension. In *Proceedings of the ACM Conference on Human Factors in Computing Systems (CHI)*. Association for Computing Machinery, New York, NY, USA, Article 893, 10 pages. https://doi.org/10.1145/3613904.3642314

[35] Hyeonsu Kang, Joseph Chee Chang, Yongsung Kim, and Aniket Kittur. 2022. Threddy: An Interactive System for Personalized Thread-based Exploration and Organization of Scientific Literature. In *Proceedings of the ACM Symposium on User Interface Software and Technology (UIST)*. Association for Computing Machinery, New York, NY, USA, Article 94, 15 pages. https://doi.org/10.1145/3526113.3545660

[36] Hyeonsu B Kang, Rafal Kocielnik, Andrew Head, Jiangjiang Yang, Matt Latzke, Aniket Kittur, Daniel S Weld, Doug Downey, and Jonathan Bragg. 2022. From Who You Know to What You Read: Augmenting Scientific Recommendations with Implicit Social Networks. In *Proceedings of the ACM Conference on Human Factors in Computing Systems (CHI)*. Association for Computing Machinery, New York, NY, USA, Article 302, 23 pages. https://doi.org/10.1145/3491102.3517470

[37] Hyeonsu B Kang, Nouran Soliman, Matt Latzke, Joseph Chee Chang, and Jonathan Bragg. 2023. ComLittee: Literature Discovery with Personal Elected Author Committees. In *Proceedings of the ACM Conference on Human Factors in Computing Systems (CHI)*. Association for Computing Machinery, New York, NY, USA, Article 738, 20 pages. https://doi.org/10.1145/3544548.3581371

[38] Jakob Karolus, Sebastian S. Feger, Albrecht Schmidt, and Paweł W. Woźniak. 2023. Your Text Is Hard to Read: Facilitating Readability Awareness to Support Writing Proficiency in Text Production. In *Proceedings of the ACM Designing Interactive Systems Conference (DIS)*. Association for Computing Machinery, New York, NY, USA, 147–160. https://doi.org/10.1145/3563657.3596052

[39] Abdelhak Kelious, Matthieu Constant, and Christophe Coeur. 2024. Complex word identification: A comparative study between ChatGPT and a dedicated model for this task. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*. 3645–3653.

[40] Dae Hyun Kim, Enamul Hoque, Juho Kim, and Maneesh Agrawala. 2018. Facilitating Document Reading by Linking Text and Tables. In *Proceedings of the ACM Symposium on User Interface Software and Technology (UIST)*. Association for Computing Machinery, New York, NY, USA, 423–434. https://doi.org/10.1145/3242587.3242617

[41] Gyeongri Kim, Jiho Kim, and Yea-Seul Kim. 2023. "Explain What a Treemap is": Exploratory Investigation of Strategies for Explaining Unfamiliar Chart to Blind and Low Vision Users. In *Proceedings of the ACM Conference on Human Factors in Computing Systems (CHI)*. Association for Computing Machinery, New York, NY, USA, Article 805, 13 pages. https://doi.org/10.1145/3544548.3581139

[42] Sébastien Lallé, Cristina Conati, and Giuseppe Carenini. 2016. Predicting Confusion in Information Visualization from Eye Tracking and Interaction Data.. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*. 2529–2535.

[43] Michelle S. Lam, Janice Teoh, James A. Landay, Jeffrey Heer, and Michael S. Bernstein. 2024. Concept Induction: Analyzing Unstructured Text with High-Level Concepts Using LLooM. In *Proceedings of the ACM Conference on Human Factors in Computing Systems (CHI)*. Association for Computing Machinery, New York, NY, USA, Article 766, 28 pages. https://doi.org/10.1145/3613904.3642830

[44] Shahid Latif, Zheng Zhou, Yoon Kim, Fabian Beck, and Nam Wook Kim. 2022. Kori: Interactive Synthesis of Text and Charts in Data Documents. *IEEE Transactions on Visualization and Computer Graphics* 28, 1 (Jan 2022), 184–194. https://doi.org/10.1109/TVCG.2021.3114802

[45] Geonsun Lee. 2024. Designing Interfaces for Enhanced Productivity and Collaboration in Virtual Workspaces. In *Adjunct Proceedings of the IEEE International Symposium on Mixed and Augmented Reality*. 666–667. https://doi.org/10.1109/ISMAR-Adjunct64951.2024.00202

[46] Geonsun Lee, Yue Yang, Jennifer Healey, and Dinesh Manocha. 2025. Since U Been Gone: Augmenting Context-Aware Transcriptions for Re-Engaging in Immersive VR Meetings. In *Proceedings of the ACM Conference on Human Factors in Computing Systems*. Association for Computing Machinery, New York, NY, USA, Article 785, 20 pages. https://doi.org/10.1145/3706598.3714078

[47] Mina Lee, Percy Liang, and Qian Yang. 2022. CoAuthor: Designing a Human-AI Collaborative Writing Dataset for Exploring Language Model Capabilities. In *Proceedings of the ACM Conference on Human Factors in Computing Systems (CHI)*. Association for Computing Machinery, New York, NY, USA, Article 388, 19 pages. https://doi.org/10.1145/3491102.3502030

[48] Yoonjoo Lee, Hyeonsu B Kang, Matt Latzke, Juho Kim, Jonathan Bragg, Joseph Chee Chang, and Pao Siangliulue. 2024. PaperWeaver: Enriching Topical Paper Alerts by Contextualizing Recommended Papers with User-collected Papers. In *Proceedings of the ACM Conference on Human Factors in Computing Systems (CHI)*. Association for Computing Machinery, New York, NY, USA, Article 19, 19 pages. https://doi.org/10.1145/3613904.3642196

[49] Shuyun Li and Hugh Munby. 1996. Metacognitive strategies in second language academic reading: A qualitative investigation. *English for specific purposes* 15, 3 (1996), 199–216.

[50] Zhuoyan Li, Chen Liang, Jing Peng, and Ming Yin. 2024. The Value, Benefits, and Concerns of Generative AI-Powered Assistance in Writing. In *Proceedings of the ACM Conference on Human Factors in Computing Systems (CHI)*. Association for Computing Machinery, New York, NY, USA, Article 1048, 25 pages. https://doi.org/10.1145/3613904.3642625

[51] Ilya Loshchilov and Frank Hutter. 2017. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101* (2017).

[52] Damien Masson, Sylvain Malacria, Géry Casiez, and Daniel Vogel. 2023. Statslator: Interactive Translation of NHST and Estimation Statistics Reporting Styles in Scientific Documents. In *Proceedings of the ACM Symposium on User Interface Software and Technology (UIST)*. Association for Computing Machinery, New York, NY, USA, Article 91, 14 pages. https://doi.org/10.1145/3586183.3606762

[53] Tarek Naous, Michael J Ryan, Anton Lavrouk, Mohit Chandra, and Wei Xu. 2024. Readme++: Benchmarking multilingual language models for multi-domain readability assessment. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing. Conference on Empirical Methods in Natural Language Processing*, Vol. 2024. 12230.

[54] Council of Europe. Council for Cultural Co-operation. Education Committee. Modern Languages Division. 2001. *Common European framework of reference for languages: Learning, teaching, assessment*. Cambridge University Press.

[55] Oxford University Press. [n. d.]. The Oxford 5000™ by CEFR Level. https://www.oxfordlearnersdictionaries.com/external/pdf/wordlists/oxford-3000-5000/The_Oxford_5000_by_CEFR_level.pdf. Oxford Learner's Dictionaries. Accessed: 2025-09-08.

[56] Srishti Palani, Aakanksha Naik, Doug Downey, Amy X. Zhang, Jonathan Bragg, and Joseph Chee Chang. 2023. Relatedly: Scaffolding Literature Reviews with Existing Related Work Sections. In *Proceedings of the ACM Conference on Human Factors in Computing Systems (CHI)*. Association for Computing Machinery, New York, NY, USA, Article 742, 20 pages. https://doi.org/10.1145/3544548.3580841

[57] Paul Parsons and Prakash Shukla. 2020. Data Visualization Practitioners' Perspectives on Chartjunk. In *Short Proceedings of the IEEE Visualization Conference*. 211–215. https://doi.org/10.1109/VIS47514.2020.00049

[58] Zhenhui Peng, Yuzhi Liu, Hanqi Zhou, Zuyu Xu, and Xiaojuan Ma. 2022. CReBot: Exploring interactive question prompts for critical paper reading. *International Journal of Human-Computer Studies* 167 (2022), 102898. https://doi.org/10.1016/j.ijhcs.2022.102898

[59] Agustina Ramadhianti and Sugianti Somba. 2023. Reading comprehension difficulties in Indonesian EFL students. *Journal of English Language Teaching and Literature (JELTL)* 6, 1 (2023), 1–11.

[60] Vai Ramanathan and Dwight Atkinson. 1999. Individualism, academic writing, and ESL writers. *Journal of second language writing* 8, 1 (1999), 45–75.

[61] Sungbok Shin, Andrea Batch, Peter William Scott Butcher, Panagiotis D. Ritsos, and Niklas Elmqvist. 2024. The Reality of the Situation: A Survey of Situated Analytics. *IEEE Transactions on Visualization and Computer Graphics* 30, 8 (2024), 5147–5164. https://doi.org/10.1109/TVCG.2023.3285546

[62] Sungbok Shin, Sanghyun Hong, and Niklas Elmqvist. 2025. Visualizationary: Automating Design Feedback for Visualization Designers Using LLMs. *IEEE Transactions on Visualization and Computer Graphics* (2025), 1–17. https://doi.org/10.1109/TVCG.2025.3579700

[63] Sungbok Shin, Inyoup Na, and Niklas Elmqvist. 2025. Drillboards: Adaptive Visualization Dashboards for Dynamic Personalization of Visualization Experiences. *IEEE Transactions on Visualization and Computer Graphics* (2025), 1–14. https://doi.org/10.1109/TVCG.2025.3542606

[64] Shane D. Sims and Cristina Conati. 2020. A Neural Architecture for Detecting User Confusion in Eye-tracking Data. In *Proceedings of the International Conference on Multimodal Interaction*. Association for Computing Machinery, New York, NY, USA, 15–23. https://doi.org/10.1145/3382507.3418828

[65] Nikhil Singh, Lucy Lu Wang, and Jonathan Bragg. 2024. FigurA11y: AI Assistance for Writing Scientific Alt Text. In *Proceedings of the International Conference on Intelligent User Interfaces (IUI)*. Association for Computing Machinery, New York, NY, USA, 886–906. https://doi.org/10.1145/3640543.3645212

[66] Chase Stokes, Vidya Setlur, Bridget Cogley, Arvind Satyanarayan, and Marti A. Hearst. 2023. Striking a Balance: Reader Takeaways and Preferences when Integrating Text and Charts. *IEEE Transactions on Visualization and Computer Graphics* 29, 1 (Jan 2023), 1233–1243. https://doi.org/10.1109/TVCG.2022.3209383

[67] Hendrik Strobelt, Daniela Oelke, Bum Chul Kwon, Tobias Schreck, and Hanspeter Pfister. 2016. Guidelines for Effective Usage of Text Highlighting Techniques. *IEEE Transactions on Visualization and Computer Graphics* 22, 1 (2016), 489–498. https://doi.org/10.1109/TVCG.2015.2467759

[68] Emily Tse and Ujjaini Sahasrabudhe. [n. d.]. O's and A's and other Letters of the Alphabet: Making Sense of British-based Secondary and Pre-University Level Qualitifications. accessed on: April 2025.

[69] Satoru Uchida and Masashi Negishi. 2018. Assigning CEFR-J levels to English texts based on textual features. In *Proceedings of Asia Pacific Corpus Linguistics Conference*, Vol. 4. 463–467.

[70] Thiemo Wambsganss, Tobias Kueng, Matthias Soellner, and Jan Marco Leimeister. 2021. ArgueTutor: An Adaptive Dialog-Based Learning System for Argumentation Skills. In *Proceedings of the ACM Conference on Human Factors in Computing Systems (CHI)*. Association for Computing Machinery, New York, NY, USA, Article 683, 13 pages. https://doi.org/10.1145/3411764.3445781

[71] Xizhe Wang, Yihua Zhong, Changqin Huang, and Xiaodi Huang. 2024. ChatPRCS: A Personalized Support System for English Reading Comprehension Based on ChatGPT. *IEEE Transactions on Learning Technologies* 17 (2024), 1762–1776. https://doi.org/10.1109/TLT.2024.3405747

[72] An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jing Zhou, Jingren Zhou, Junyang Lin, Kai Dang, Keqin Bao, Kexin Yang, Le Yu, Lianghao Deng, Mei Li, Mingfeng Xue, Mingze Li, Pei Zhang, Peng Wang, Qin Zhu, Rui Men, Ruize Gao, Shixuan Liu, Shuang Luo, Tianhao Li, Tianyi Tang, Wenbiao Yin, Xingzhang Ren, Xinyu Wang, Xinyu Zhang, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yinger Zhang, Yu Wan, Yuqiong Liu, Zekun Wang, Zeyu Cui, Zhenru Zhang, Zhipeng Zhou, and Zihan Qiu. 2025. Qwen3 Technical Report. *arXiv preprint arXiv:2505.09388* (2025).

[73] Kangyu Yuan, Hehai Lin, Shilei Cao, Zhenhui Peng, Qingyu Guo, and Xiaojuan Ma. 2023. CriTrainer: An Adaptive Training Tool for Critical Paper Reading. In *Proceedings of the ACM Symposium on User Interface Software and Technology (UIST)*. Association for Computing Machinery, New York, NY, USA, Article 44, 17 pages. https://doi.org/10.1145/3586183.3606816

[74] Vivian Zamel. 1982. Writing: The process of discovering meaning. *TESOL quarterly* 16, 2 (1982), 195–209.

[75] Xiaoyu Zhang, Jianping Li, Po-Wei Chi, Senthil Chandrasegaran, and Kwan-Liu Ma. 2023. ConceptEVA: Concept-Based Interactive Exploration and Customization of Document Summaries. In *Proceedings of the ACM Conference on Human Factors in Computing Systems (CHI)*. Association for Computing Machinery, New York, NY, USA, Article 204, 16 pages. https://doi.org/10.1145/3544548.3581260

[76] Dennis Zyska, Nils Dycke, Jan Buchmann, Ilia Kuznetsov, and Iryna Gurevych. 2023. CARE: Collaborative AI-Assisted Reading Environment. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (Volume 3: System Demonstrations)*. Association for Computational Linguistics, Toronto, Canada, 291–303. https://doi.org/10.18653/v1/2023.acl-demo.28