

Effdit: An Assistant for Improving Writing Efficiency

Shuming Shi, Enbo Zhao, Wei Bi, Deng Cai, Leyang Cui, Xinting Huang, Haiyun Jiang, Duyu Tang, Kaiqiang Song, Longyue Wang, Chengyan Huang, Guoping Huang, Yan Wang, Piji Li
ailabnlp@tencent.com

Abstract

Writing assistants are valuable tools that can help writers improve their writing skills. We introduce Effdit (**Efficient and Intelligent Editing**), a digital writing assistant that facilitates users to write higher-quality text more efficiently through the use of Artificial Intelligence (AI) and Natural Language Processing (NLP) technologies. We significantly expand the capacities of a writing assistant by providing functions in three modules: text completion, hint recommendation, and writing refinement. Based on the above efforts, Effdit can efficiently assist users in creating their own text. Effdit has been deployed to several Tencent products and publicly released at <https://effdit.qq.com/>.

1 Introduction

Effective communication through writing is crucial in modern society, as it allows individuals to share their thoughts, opinions, and responses, thereby enabling them to establish meaningful connections with others. However, writing could be a challenging task, regardless of whether you are a beginner or an experienced writer. It requires a combination of creativity, organization, and critical thinking. To address this challenge, several writing assistant tools ([Grammarly](#); [QuillBot](#)) have been developed to help users write faster and more efficiently. These tools use Artificial Intelligence (AI) and Natural Language Processing (NLP) techniques to provide suggestions, corrections, and recommendations to improve writing quality.

In general, the writing process can be divided into three main stages: generating and proposing ideas, composing coherent sentences, and refining the language through editing and polishing ([Flower and Hayes, 1980](#)). Tracking this, we present Effdit (**Efficient and Intelligent Editing**) to provide users with an innovative writing experience, which significantly expands the capacities of a typical writing assistant by providing three categories of functions:

- **Text Completion** module intends to provide high-quality continuations for any given textual prefix for improving the user’s writing efficiency. Effdit offers both short-form and long-form text completion. For short-form completion, Effdit provides two functions including Phrase Completion which takes the prefix as input and suggests suitable follow-up phrases, and a Cloud Input Method Editor (IME) which can pop up instant suggestions during typing. Cloud Chinese Pinyin and English input methods are provided. Meanwhile, Sentence Completion automatically completes the whole sentence based on the prefix, serving as a long-form assistance. The completion can be done by either retrieving from an existing datastore or generating using advanced neural language models. To help users create fun and distinctive content, Effdit also supports stylistic automatic writing for Chinese novel writing. Specifically, four distinct and popular novel styles are covered currently.
- **Hint Recommendation** module can help users to brainstorm when they are struggling to find the inspiring words or are dealing with writer’s block. This module can recommend various hints given the input keywords, such as mono-lingual and multi-lingual retrieval example sentences, generated sentences by infilling the input keywords, representative papers, and a super-power dictionary that can suggest synonyms, antonyms, related words, possible modifiers, and related entities.
- **Writing Refinement** module is designed to help users refine texts they have already typed down by suggesting improvements in grammar, syntax, phrase and vocabulary. Grammar Error Correction (GEC) focuses on identifying and correcting grammar mistakes inside the sentence. Effdit additionally provides grammar-aware explanations and evidence words for corrections in English GEC. Effdit provides three different levels of text polishing: phrase polishing, sentence rewriting (i.e. para-

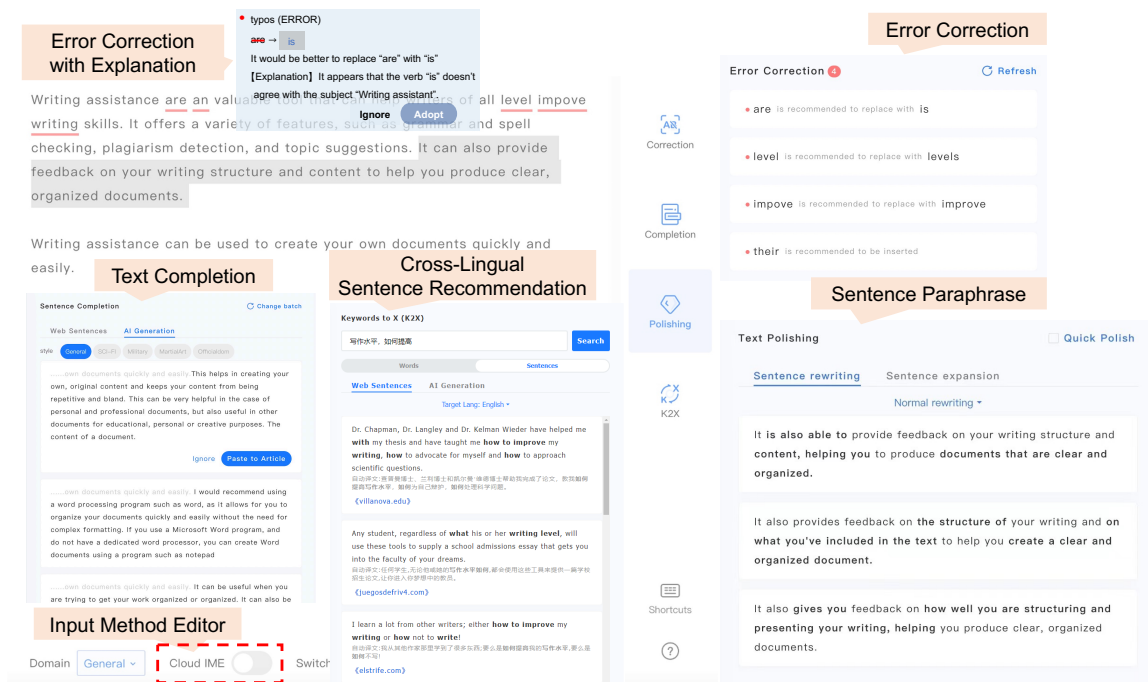


Figure 1: Screenshot of the Effdit online demo. To save our space, we put the results of Text Completion (Sentence Generation) and Hint Recommendation (Cross-Lingual Sentence Recommendation) on the bottom of the left part, which are originally on the right pane.

phrasing) and sentence expansion. By selecting some words, phrase-level polishing automatically recommends words that are more suitable for the context, making the overall sentence more accurate and vivid. Sentence rewriting automatically modifies a sentence while retaining its semantics, thus enhancing sentence diversity. Sentence expansion generates longer sentences with consistent but richer semantics for a target sentence by adding some modifiers.

The first version of Effdit was released in April, 2022. Tencent AI lab constantly maintains and updates Effdit, aiming to provide users with a better writing experience. The current version supports both Chinese and English. Effdit is now providing support for many products and users both inside and outside Tencent. Detailed documents and tutorials are available at <https://effdit.qq.com/en>.

2 Overview

Figure 1 illustrates the process of users using the online demo of Effdit. The left part is a plain-text editor for users to add and edit text, whereas the right pane is for triggering most core functions of Effdit and displaying corresponding results. In addition, some UI elements are at the bottom for

changing domains and setting up the cloud IME. Table 1 presents a comparison between Effdit and several well-designed writing assistants. As can be seen, previous writing assistants typically provide the function of error checking (to detect and correct spelling and grammatical errors) and limited text-rewriting functionality. In Effdit, we significantly expand the capacities of a writing assistant by providing functions in three perspectives. More technical details can be found at Shi et al. (2022).

2.1 Architecture

Figure 2 shows three modules supported by Effdit: Text Completion module takes the incomplete text as input, and help users compose coherent sentences; Hint Recommendation module suggests relevant words/phrases/sentences given the input words; Writing Refinement module polishes the existing passage to improve writing quality. These modules are introduced in the following subsections.

3 Text Completion Module

Effdit offers both short-form and long-form assistance for text completion, which can help users enhance the efficiency and effectiveness of their writing. As an advanced feature, Effdit can provide completion suggestions in specific writing styles.

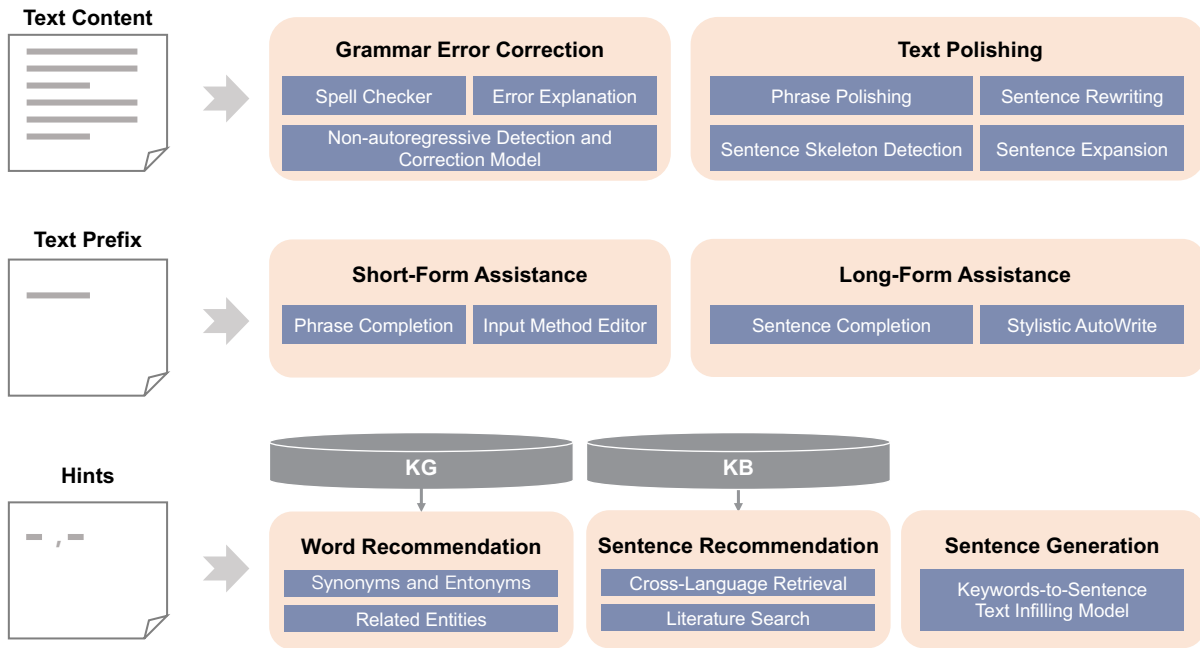


Figure 2: Overview of all modules and the main backbone models in Effdit. We categorize them based on the semantic completeness of the model inputs: Text Content refers to all text that users have written down, which could include multiple sentences, paragraphs, and articles; Text Prefix is a short piece of text with incomplete or partial semantics that should be continued; Keywords are one or a few words that users want to explore further information as hints to inspire their writing.



Figure 3: Phrase completion examples (Left: English; Right: Chinese).

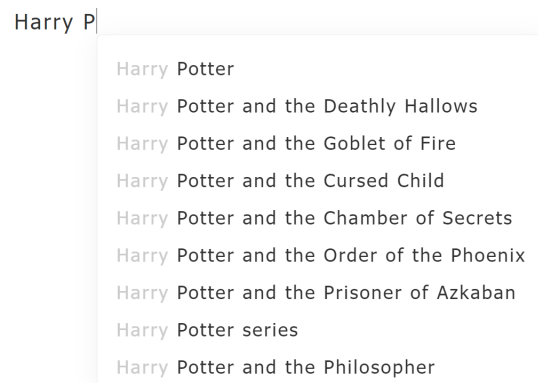


Figure 4: An example of Cloud Input Methods.

Currently, four unique styles are available, including science fiction (SCI-FI), military novel (Military), martial art fantasy (MartialArt), and urban story (Urban), allowing users to create engaging and stylistic content that is both fun and distinctive.

Phrase completion Figure 3 shows an example. It is noteworthy that Effdit considers both the prefix and suffix for phrase completion, setting it apart from most other writing assistants that only take the prefix into account. As illustrated in the left part of the figure, when the caret is after the first letter “B”, the top completion results contain both the prefix “B” and the suffix “Spears”. Phrase comple-

tion can improve writing efficiency and reduce the chance of typographical errors (Lee et al., 2021). For example, if a user types “Los A” and triggers phrase completion, the topmost suggestion is “Los Angeles”, which can be directly selected by the user. In this way, potential spelling mistakes such as “Los Angelas” can be avoided.

Cloud input method editors (Cloud IMEs) Effdit provides cloud IMEs for both Chinese and English to improve the input efficiency of words and phrases. Instant suggestions pop up when the user is typing in our text editor. Figure 4 depicts an example where the names of some novels in the

Function	QuillBot	Grammarly	DeepL	Pitaya	Phrasebank	Effdit
Phrase Completion	-	-	-	-	-	✓
Retrieval-based Completion	-	-	-	-	-	✓
Generation-based Completion	-	-	-	-	-	✓
Mono-lingual Retrieval	-	-	-	✓	-	✓
Cross-lingual Retrieval	-	-	-	-	-	✓
keyword-to-sentence	-	-	-	✓	-	✓
dictionary	-	-	-	✓	✓	✓
Grammar Error Correction	✓	✓	✓	✓	-	✓
Phrase Polish	✓	✓	✓	✓	-	✓
Sentence Rewrite	✓	-	-	-	-	✓
Sentence Expansion	-	-	-	✓	-	✓
Supported language	En	En	En	En&Zh	En	En&Zh

Table 1: Comparison between Effdit and other writing assistants.

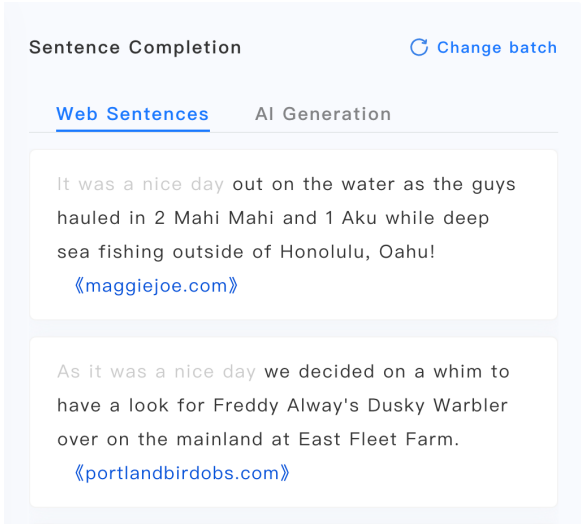


Figure 5: Examples of sentence-level completion using Web Retrieval. The input prefix is “It was a nice day”.

Harry Potter series are shown as suggestions.

Sentence completion Effdit also provides sentence completion suggestions (Zweig et al., 2012). Figure 5 and 6 demonstrate the results of two completion methods: web retrieval and AI generation. Similar to phrase completion, Effdit supports both Chinese and English. The web retrieval method utilizes a large collection of high-quality sentences to search for the most similar sentences to the input prefix. On the other hand, the AI generation method leverages large language models to generate the most probable continuations. To alleviate the serious degeneration problem, i.e., the generated texts from the language model tend to be dull and contain undesirable repetitions at different lev-

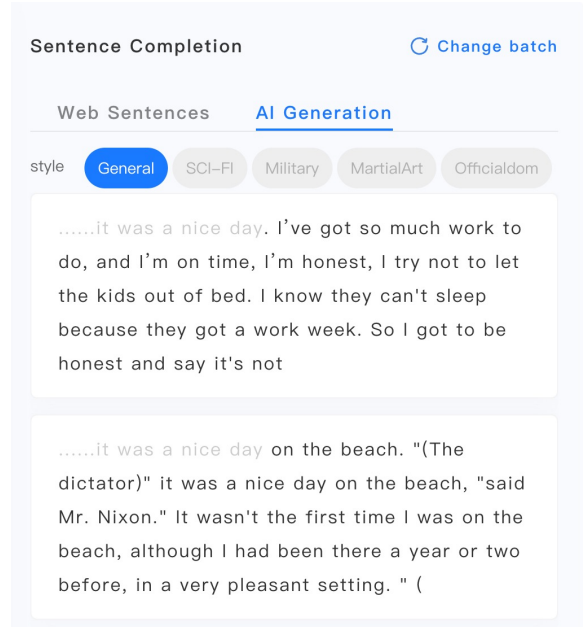


Figure 6: Examples of sentence-level completion using AI generations. The input prefix is “It was a nice day”.

els (e.g., token-, phrase-, and sentence-level), we apply the contrastive search (Su et al., 2022) and FSD decoding (Yang et al., 2023) methods. The key intuitions behind are: (i) At each decoding step, the output should be selected from the set of most probable candidates predicted by the model to better maintain the semantic coherence. (ii) The patterns that occur in the input prefix and/or the previously generated sequence should be penalized to avoid degeneration. Effdit integrates both retrieval and generation methods by presenting multiple candidates from each of them, enabling users to choose the best option according to their preferences.

Stylistic AutoWrite To assist users in crafting engaging, creative, and versatile stories, Effdit expands its sentence completion function to include stylistic autowrite (Dathathri et al., 2020). With four novel writing styles to choose from (SCI-FI, Military, MartialArt, and Urban), Effdit generates multiple possible continuations that adhere to the selected writing style and the input prefix. This functionality allows users to explore multiple stylistic possibilities and inspires their creativity. Unlike other text completion modules, stylistic autowrite only supports Chinese writing currently, and we are trying to upgrade our models for English and more languages, which will be released soon.

4 Hint Recommendation Module

Effdit can take one or more keywords as input and recommend a list of related words, sentences and documents. Especially, each recommended output sentence/document either contains the input keywords, or is semantically related to them. These modules are built for the scenario that users only have some basic concepts in their minds but do not have enough information to organize sentences to express their ideas.

Word Recommendation Effdit provides a super-power dictionary for users to first explore a wide range of related words including synonyms, antonyms, similar words, possible modifiers, and related entities, etc. As shown in Figure 7, given a keyword “linguistic”, Effdit produces both monolingual and bi-lingual dictionaries.

Keyword2sentence (K2S) This module takes several keywords as input and returns a list of sentences. Effdit supports the retrieval-based method and the generation-based method. For the retrieval-based method, top sentences containing the input keywords are retrieved from a corpus of high-quality sentences collected beforehand from the Web. Generation-based K2S is formulated as a text-infilling problem (Zhu et al., 2019; Donahue et al., 2020), which is to generate missing spans of text. Our system considers a general text-infilling setting, where the incomplete sentence can contain an arbitrary number of blanks to be filled in, and each blank can involve an unknown number of tokens. Illustrated examples are provided in Figure 8. For both types of results, Effdit displays multiple candidates, from which the user can select the most appropriate one for reference.

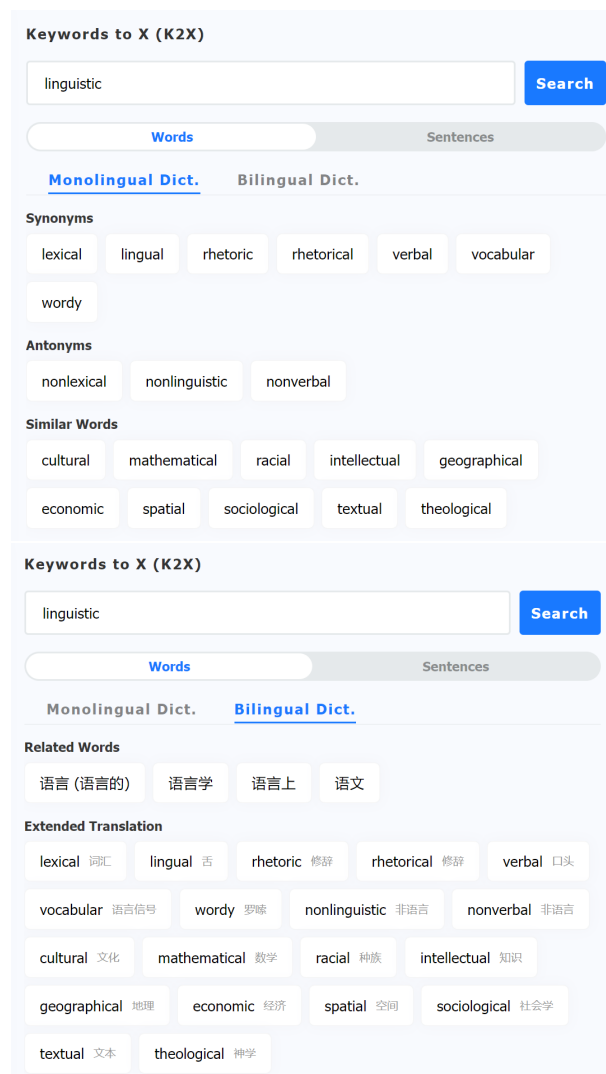


Figure 7: Examples of word recommendations provided by the super-power dictionary (Upper: Mono-lingual; Lower: Bilingual).

Sentence/Document Recommendation We use information retrieval methods (Baeza-Yates et al., 1999; Manning, 2008) to index a large corpus of sentences and documents. Note that Effdit supports cross-lingual retrieval, where the output is English sentences and the input can be a mixture of Chinese and English keywords. Effdit also support document-level search, especially paper recommendation. Unlike other paper search services such as (Scholar), the paper search of Effdit focuses more on recalling papers that are semantically related to the input keywords other than those containing the input keywords in their titles or contents. In particular, when the input keyword is a research topic, important papers on this topic will be returned even if some keywords are missing in some results.

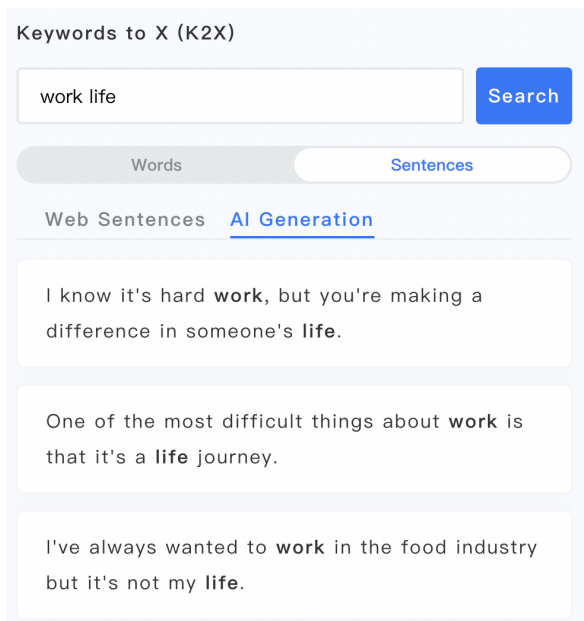


Figure 8: Examples of keywords2sentence.

5 Writing Refinement Module

Grammar Error Correction The grammar error correction (GEC) capability is a pivotal attribute for writers, particularly for novices and those utilizing a non-native language (Dahlmeier and Ng, 2012; Wang et al., 2020). The GEC module of Effdit can automatically detect and correct grammatical, syntactical, and spelling errors in written texts. As shown in the upper right part of Figure 1, Effdit offers users various types of suggestions, such as substitution, deletion, and insertion, during the process of error correction. We have a tail-to-tail model (Li and Shi, 2021) to handle substitution errors and a tailored pretrained model (Zhou et al., 2022) to handle deletion and insertion errors. Moreover, Effdit advances its GEC capabilities by providing grammar-aware explanations and evidence words. Specifically, as shown in the upper left part of Figure 1, the explanation describes why a correction was made to a particular error (e.g. the grammatical rule that was violated), and evidence words provide a clear indication of such error (Fei et al., 2023). With explainable GEC functions, Effdit can increase users' trust and acceptance of the corrections made by the algorithm, and help them to improve their writing skills via detailed and specific feedback.

Text Polishing Text polishing is a process of refining and improving written text by enhancing its clarity, readability, and overall quality (Bhagat and

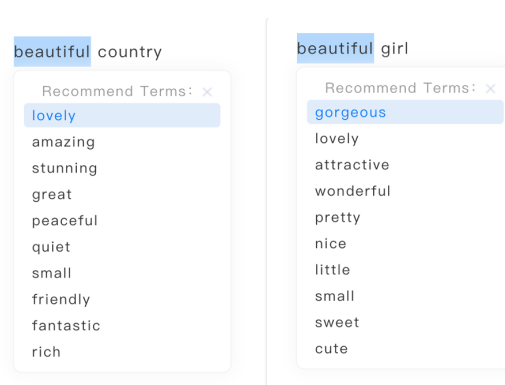


Figure 9: Examples of phrasal paraphrasing.

Hovy, 2013; Gupta et al., 2018). Effdit provides two functions, phrasal and sentential paraphrasing, to achieve the goal of text polishing. Phrasal paraphrasing involves rephrasing individual words or phrases within a sentence to create more impactful or clearer language. As shown in Figure 9, by phrasal paraphrasing, writers can avoid repetitive language and create a more interesting and varied writing style. Sentence paraphrasing, on the other hand, involves rewriting an entire sentence to convey the same meaning in a different way. The lower right quadrant of Figure 1 illustrates an instance in which a range of diverse paraphrased options are presented. Based on options suggested by Effdit, writers can then choose alternative phrasing that is more natural and intuitive. In this way, it would be able to eliminate awkward or convoluted sentence structures that make writing difficult to follow. Apart from normal rewriting, we also provide the conversion between classical↔modern Chinese language.¹ Different from modern language, classical language is often used in poetry, prose, and other forms of traditional literature. The modern→classical conversion can help writers seamlessly integrate classical elements into their writing, making their works more artistic and expressive. On the other hand, classical→modern conversion can accessibly help users better understand classical literature, especially in the field of education.

Text Expansion Effdit provides the function of text expansion by adding some elegant modifiers to an input sentence to make a longer one with rich information. In general, the new sentence will keep the core meaning with the input sentence. For ex-

¹https://en.wikipedia.org/wiki/Chinese_classics.

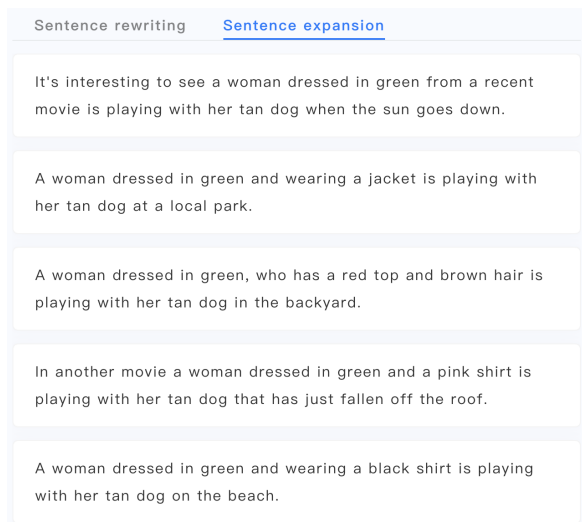


Figure 10: Examples of text expansion.

ample, given the text “A woman dressed in green is playing with her tan dog”, the expanded sentences are presented in Figure 10. Text expansion module was implemented by two complementary ideas. The first is to resort to syntactic parsing to extract the skeleton of a long sentence to build the training pairs, i.e., <skeleton, sentence>. Then the skeleton is used to generate the long sentence with a seq2seq model (Lewis et al., 2019). The second idea is to identify some places (mainly around a noun or verb) explicitly and then use a text in-filing model (Raffel et al., 2020) to predict some words or phrases that can be added to each place.

6 Conclusion

We introduced Effdit, a writing assistant that facilitates users to write high quality text efficiently using AI technologies. Our system supports two languages, Chinese and English, and has three categories of functions: text completion, hint recommendation, and writing refinement. With these modules, Effdit significantly expands the capacities of a typical writing assistant. In the future, we plan to keep improving the quality of each module to make the system more helpful and easy-to-use.

References

- Ricardo Baeza-Yates, Berthier Ribeiro-Neto, et al. 1999. *Modern information retrieval*, volume 463. ACM press New York.
- Rahul Bhagat and Eduard Hovy. 2013. What is a paraphrase? *Computational Linguistics*, 39(3):463–472.

Daniel Dahlmeier and Hwee Tou Ng. 2012. Better evaluation for grammatical error correction. In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 568–572.

Sumanth Dathathri, Andrea Madotto, Janice Lan, Jane Hung, Eric Frank, Piero Molino, Jason Yosinski, and Rosanne Liu. 2020. Plug and play language models: A simple approach to controlled text generation. In *8th International Conference on Learning Representations*,.

DeepL. <https://www.deepl.com/write>.

Chris Donahue, Mina Lee, and Percy Liang. 2020. Enabling language models to fill in the blanks. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2492–2501.

Yuejiao Fei, Leyang Cui, Sen Yang, Wai Lam, Zhenzhong Lan, and Shuming Shi. 2023. Enhancing grammatical error correction systems with explanations. Toronto, Canada. Proceedings of the 61th Annual Meeting of the Association for Computational Linguistics.

Linda S. Flower and J. R. Hayes. 1980. The cognition of discovery: Defining a rhetorical problem. *College Composition and Communication*, 31:21–32.

Grammarly. <https://www.grammarly.com/>.

Ankush Gupta, Arvind Agarwal, Prawaan Singh, and Piyush Rai. 2018. A deep generative framework for paraphrase generation. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32.

Dong-Ho Lee, Zhiqiang Hu, and Roy Ka-Wei Lee. 2021. Improving text auto-completion with next phrase prediction. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 4434–4438, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. 2019. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *arXiv preprint arXiv:1910.13461*.

Piji Li and Shuming Shi. 2021. Tail-to-tail non-autoregressive sequence prediction for chinese grammatical error correction. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing*, pages 4973–4984.

Christopher D Manning. 2008. *Introduction to information retrieval*. Syngress Publishing,.

Academic Phrasebank. <https://www.ref-n-write.com/academic-phrasebank/>.

Pitaya. <https://www.mypitaya.com/en>.

QuillBot. <https://quillbot.com/>.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. [Exploring the limits of transfer learning with a unified text-to-text transformer](#). *Journal of Machine Learning Research*, 21(140):1–67.

Google Scholar. <https://scholar.google.com/>.

Shuming Shi, Enbo Zhao, Duyu Tang, Yan Wang, Piji Li, Wei Bi, Haiyun Jiang, Guoping Huang, Leyang Cui, Xinting Huang, et al. 2022. Effidit: Your ai writing assistant. *arXiv preprint arXiv:2208.01815*.

Yixuan Su, Tian Lan, Yan Wang, Dani Yogatama, Lingpeng Kong, and Nigel Collier. 2022. A contrastive framework for neural text generation. In *Advances in Neural Information Processing Systems*.

Yu Wang, Yuelin Wang, Jie Liu, and Zhuo Liu. 2020. A comprehensive survey of grammar error correction. *arXiv preprint arXiv:2005.06600*.

Haoran Yang, Deng Cai, Huayang Li, Wei Bi, Wai Lam, and Shuming Shi. 2023. A frustratingly simple decoding method for neural text generation. *arXiv preprint arXiv:2305.12675*.

Cong Zhou, Yong Dai, Duyu Tang, Enbo Zhao, Zhangyin Feng, Li Kuang, and Shuming Shi. 2022. Pretraining chinese bert for detecting word insertion and deletion errors. *arXiv preprint arXiv:2204.12052*.

Wanrong Zhu, Zhiting Hu, and Eric Xing. 2019. Text infilling. *arXiv preprint arXiv:1901.00158*.

Geoffrey Zweig, John C Platt, Christopher Meek, Christopher JC Burges, Ainur Yessenalina, and Qiang Liu. 2012. Computational approaches to sentence completion. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics*, pages 601–610.