Prompt Disambiguation for Text-to-Image Generation

Woojin Na
Dept. of Electrical and Computer Engineering
Ajou University
Suwon, Korea
jasonna33@ajou.ac.kr

Minjun Kang
Dept. of Artificial Intelligence
Ajou University
Suwon, Korea
baramshu@ajou.ac.kr

Hyung Il Koo
Dept. of Electrical and Computer Engineering
Ajou University
Suwon, Korea
hikoo@ajou.ac.kr (corresponding author)

***Abstract*—Text-to-Image (T2I) diffusion models, such as Stable Diffusion and DALL·E 2, have demonstrated remarkable generative capabilities. However, they often struggle with lexically ambiguous prompts, resulting in outputs that conflate or misinterpret multiple word senses. Prior approaches addressed this issue through prompt design guidelines or interactive workflows based on user feedback. However, these methods require repeated human intervention and lack the ability to automatically detect and resolve lexical ambiguity. To address this limitation, we propose a fully automated pipeline for prompt disambiguation in T2I generation. The pipeline comprises three stages: (1) detecting ambiguous words and selecting appropriate WordNet glosses via a Word Sense Disambiguation (WSD) model; (2) rewriting the prompt into a disambiguated version using a large language model (LLM) informed by the selected glosses; and (3) generating the final image using a T2I diffusion model. We conduct both qualitative and quantitative evaluations to assess the effectiveness of our method. For the quantitative evaluation, we use the V-LAB benchmark, in which human annotators assess whether each generated image aligns with the intended meaning of the prompt. Our results demonstrate that resolving lexical ambiguity prior to image generation significantly improves semantic fidelity and output consistency in T2I models.**

***Index Terms*—Text-to-Image Generation, Lexical Ambiguity, Word Sense Disambiguation, Prompt Engineering, Large Language Model**

I. Introduction

Diffusion-based text-to-image (T2I) generation models such as DALL·E 2 [1], Stable Diffusion [2], and Imagen [3] have recently garnered widespread attention due to their ability to produce diverse and high-quality images from textual prompts. However, crafting prompts that consistently yield the intended visual content remains challenging, particularly when the input prompt contains lexical ambiguities. T2I models often misinterpret such prompts due to their limited understanding of linguistic context, producing outputs biased toward the most frequently observed senses in the training data.

Previous studies have shown that diffusion models often visualize multiple senses of a polysemous word at once—a phenomenon known as *homonym duplication*. For instance, the prompt *"A seal on an envelope"* frequently produces images depicting both a marine animal and a sealing sticker [4], [5], as illustrated on the right side of Fig. 1. Furthermore, the same prompt can yield images that reflect different senses of an ambiguous word across generations, underscoring a lack of semantic consistency. To mitigate this issue, previous work has proposed prompt guidelines or interactive workflows based on user feedback [5], [6], [7]. However, these approaches merely assist users in crafting prompts without resolving the underlying ambiguity, and they rely on repeated human intervention. Schrödinger's Bat [4] empirically demonstrated that polysemous words are often encoded as linear superpositions of their possible meanings in the text embedding space, and illustrated how these embeddings can be edited to reduce ambiguity. However, this method neither detects ambiguous words automatically nor rewrites the prompt, thereby limiting its applicability in fully automated pipelines.

Fig. 1. An example of our pipeline resolving lexical ambiguity, showing our disambiguated result (left) and the baseline SDXL generation (right).

Recent T2I generation models primarily rely on diffusion-based methods [1], [2], [3], [8]. These models employ large-scale pretrained text encoders to map input prompts into a shared latent space. However, when semantic ambiguity in a prompt is left unresolved, these models often fail to generate images that faithfully capture the user's intended meaning or maintain semantic consistency across generations. This issue becomes particularly pronounced in prompts containing polysemous terms, abstract concepts (e.g., freedom, peace), or complex compositional structures involving multiple entities and their relationships. While some studies have explored improving compositional fidelity [9] or visualizing abstract prompts [10], these approaches do not address lexical ambi-

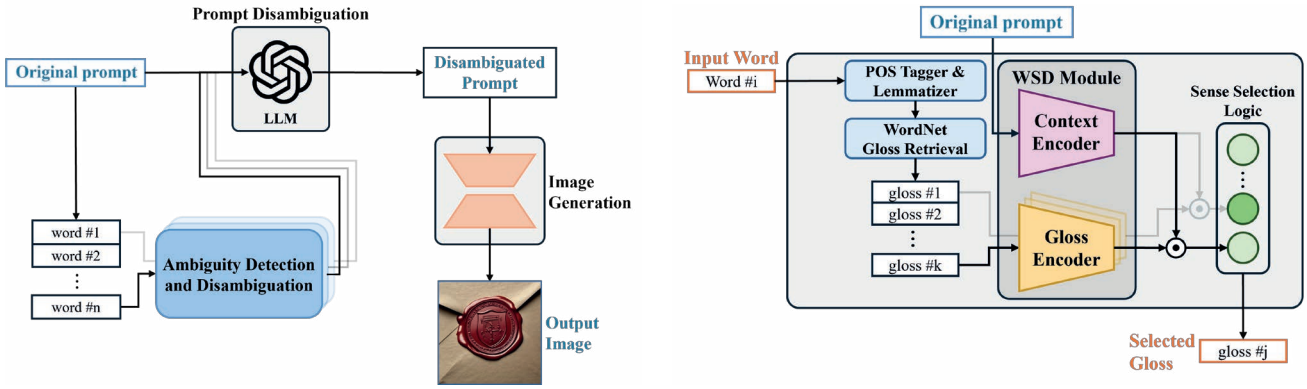

Fig. 2. (a) Overview of the full pipeline, (b) details of the *Ambiguity Detection and Disambiguation* module.

guity at the word level. To the best of our knowledge, no prior work has proposed an automated pipeline that both identifies and resolves lexical ambiguity in prompts.

To address this issue, we draw inspiration from the field of Word Sense Disambiguation (WSD). WSD is the task of identifying the intended meaning of a word in context, and it has been extensively studied using knowledge-based, supervised, and hybrid methods [11]. In particular, bi-encoder-based models [12], [13] typically consist of a context encoder and a gloss encoder. The gloss encoder represents sense definitions (also known as glosses) retrieved from the lexical knowledge base WordNet [14]. These models compute the similarity between the contextual embedding of a target word and the embeddings of its candidate glosses, thereby enabling accurate disambiguation even for rare senses.

In this paper, we propose a fully automated pipeline for prompt disambiguation in T2I generation. Our pipeline consists of:

1) *Ambiguity Detection and Disambiguation Module*, which automatically identifies lexically ambiguous words in the prompt and selects their intended senses based on a WSD model and WordNet glosses;
2) *Prompt Disambiguation Module*, which rewrites the prompt into a disambiguated version using a large language model (LLM); and
3) *Image Generation Module*, which uses the disambiguated prompt to generate the final image with a T2I diffusion model

As shown in Fig. 1, our pipeline successfully resolves lexical ambiguity. It disambiguates the prompt *"A seal on an envelope"* to *"A wax seal (a resinous, plastic-like stamp) on an envelope to securely close it"*, generating a visually unambiguous image (left). In contrast, the baseline Stable Diffusion XL (SDXL) model [8] generates an image depicting both meanings (right).

To evaluate the effectiveness of our method, we conduct experiments on the V-LAB benchmark [5], which includes lexically ambiguous prompts designed to assess semantic interpretation in T2I models. In our experiments, we use SDXL as the T2I model within the *Image Generation Module*. Our approach improves the human interpretation ($\mathrm{INT_H}$) rate over the SDXL baseline, reduces the mixed interpretation ($\mathrm{INT_M}$) rate from 13% to 4% and reduces the non-human interpretation ($\mathrm{INT_N}$) rate from 36% to 15%. It also increases the consistency score from 40% to 60%. These results indicate that our pipeline not only resolves lexical ambiguity but also enhances semantic stability in the generated output. Our contributions are as follows:

1) We propose the fully automated pipeline that systematically identifies and resolves lexical ambiguity in text prompts for T2I models. This distinguishes our work from previous approaches.
2) Through experiments on the V-LAB benchmark, we demonstrate that our pipeline significantly enhances the semantic fidelity and consistency of generated images.

II. Method

We propose a fully automated prompt disambiguation pipeline to address the image generation failures caused by lexically ambiguous prompts in T2I models. An overview of the proposed pipeline is shown in Fig. 2.

First, the *Ambiguity Detection and Disambiguation Module* detects ambiguous words in the input prompt and identifies their intended meanings using a WSD model. Second, the *Prompt Disambiguation Module* rewrites the prompt using an LLM to produce a disambiguated prompt that reflects the selected WordNet glosses. Finally, the *Image Generation Module* inputs the disambiguated prompt into a T2I diffusion model to generate the final image.

A. Ambiguity Detection and Disambiguation Module

This module identifies lexically ambiguous words in the input prompt and determines their intended meaning. The disambiguation process consists of the following steps. First, we utilize spaCy's Part-of-Speech (POS) tagger and lemmatizer to extract POS and lemma pairs for each word in the prompt [15]. For each selected word w_i in the input prompt $P = \{w_1, w_2, \ldots, w_n\}$, we apply a bi-encoder-based WSD model. This model consists of a context encoder E_C and a gloss

encoder E_G. The context encoder E_C processes the entire prompt to obtain a contextualized representation of the target word w_i, which is denoted as $E_C(P, w_i)$. Simultaneously, for the target word w_i, we utilize its extracted POS and lemma to retrieve its corresponding glosses $G_i = \{g_{i,1}, g_{i,2}, \ldots, g_{i,k}\}$ from WordNet [14]. Each gloss $g_{i,j}$ is then encoded by the Gloss Encoder E_G to obtain its embedding $E_G(g_{i,j})$.

Next, we compute the similarity score $s_{i,j}$ between the contextual representation $E_C(P, w_i)$ and each gloss embedding $E_G(g_{i,j})$ using their inner product:

$$s_{i,j} = \langle E_C(P, w_i), E_G(g_{i,j}) \rangle \tag{1}$$

for $i \in \{1, \cdots, n\}$ and $j \in \{1, \cdots, k\}$

For each target word w_i, a set of similarity scores

$$S_i = \{s_{i,1}, s_{i,2}, \ldots, s_{i,k}\} \tag{2}$$

is obtained. For each word w_i, we sort the similarity scores S_i in descending order and denote the highest and second-highest scores as s_i^* and s_i^{**}, respectively, with their corresponding glosses g_i^* and g_i^{**}. Intuitively, g_i^* represents the most likely sense candidate for w_i, with a similarity score of s_i^*, and g_i^{**} and s_i^{**} denote the second-ranked gloss and its corresponding score.

The determination of word ambiguity and sense is performed according to the two conditions:

1) *Ambiguity Assessment:* The difference between the similarity scores of the top two glosses, s_i^* and s_i^{**}, should be below the ambiguity threshold τ_A, i.e.,

$$s_i^* - s_i^{**} < \tau_A.$$

2) *Confidence Assessment:* The similarity score s_i^* of g_i^* should be above or equal to the confidence threshold τ_C, i.e.,

$$s_i^* \geq \tau_C.$$

The *Ambiguity Assessment* indicates that w_i is lexically ambiguous, while the *Confidence Assessment* determines whether the top-scoring gloss is reliable enough for disambiguation. If either condition fails, the gloss is deemed unreliable and is therefore excluded from use.

B. Prompt Disambiguation Module

This module utilizes the ambiguous words and their corresponding glosses, identified by the *Ambiguity Detection and Disambiguation Module*, to disambiguate the original prompt. Prompt disambiguation is then performed using an LLM.

To facilitate this process, we construct an LLM message sequence composed of a system message and a user message. The system message instructs the LLM to rewrite an ambiguous prompt into a clearer version, preserving the original meaning while explicitly clarifying each ambiguous word. The user message provides the original prompt along with a list of ambiguous words and their corresponding glosses. Based on these instructions and inputs, the LLM generates a revised prompt in which the lexical ambiguity present in the original has been resolved.

TABLE I
QUANTITATIVE RESULTS FOR LEXICAL AMBIGUITY PROMPTS

Model	$\text{INT}_\text{H}\uparrow$	$\text{INT}_\text{N}\downarrow$	$\text{INT}_\text{M}\downarrow$	Consistency$\uparrow$
OURS	**81%**	**15%**	**4%**	**60%**
SDXL	51%	36%	13%	40%

C. Image Generation Module

The final module receives the disambiguated prompt from the *Prompt Disambiguation Module* and passes it to a T2I diffusion model to generate the output image. As a result, the generated image is semantically faithful to the intended meaning.

III. EXPERIMENTS AND RESULTS

A. Experimental Setup

To evaluate our pipeline, we compare it against the SDXL [8] baseline. In our implementation, we use the Z-reweighting bi-encoder model [13] for WSD in the *Ambiguity Detection and Disambiguation Module*, GPT-4.1 nano accessed via the OpenAI API [16] as the LLM in the *Prompt Disambiguation Module*, and SDXL as the T2I diffusion model in the *Image Generation Module*.

B. Quantitative Results

For quantitative evaluation, we used the 10 lexical ambiguity prompts from the Visual Linguistic Ambiguity Benchmark (V-LAB) dataset [5]. For each prompt, we generated 10 images using both our method and SDXL, resulting in 100 images per model. Each generated image was manually annotated by the authors according to the interpretation categories defined in the V-LAB dataset: INT_H (human interpretation), INT_N (non-human interpretation) and INT_M (mixed interpretation) based on its semantic alignment with the human interpretation of the prompt.

As shown in Table I, the proposed method achieves a high INT_H rate of 81%, compared to 51% for SDXL. In addition, the proportions of INT_N and INT_M are reduced from 36% to 15% and from 13% to 4%, respectively. These results suggest that our method effectively resolves lexical ambiguity in prompts.

Furthermore, we compute a consistency score, defined as the percentage of prompts for which all 10 generated images yield the same interpretation. Using this metric, our method achieves a consistency score of 60%. This reflects the effectiveness of our prompt disambiguation pipeline in improving semantic stability.

Fig. 3 visualizes the distribution of interpretation categories for each prompt. Ten images were generated per prompt using both our method (left) and SDXL (right). Compared to SDXL, our method shows more consistent alignment with human interpretation across prompts, highlighting the effectiveness of the proposed prompt disambiguation pipeline.

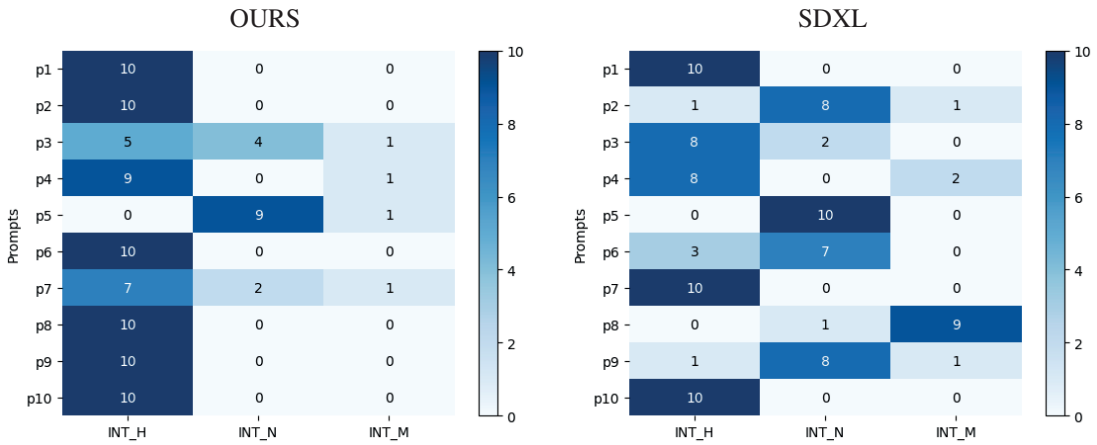

Fig. 3. Heatmaps showing the per-prompt distribution of interpretation categories for our method (left) and SDXL (right).

TABLE II
ILLUSTRATIVE EXAMPLES OF IMAGE GENERATIONS FROM OUR METHOD AND SDXL ON LEXICALLY AMBIGUOUS PROMPTS.

Ambiguous Word	Original Prompt	Disambiguated Prompt	OURS	SDXL	Comparison of Outputs
light	The man carried the light bag	A man carried a **small, lightweight rectangular bag designed for easy carrying**			SDXL presents mixed interpretations of "light" (weight and illumination), while our method clearly generates the intended lightweight meaning.
bow	A bow displayed in the market	A bow **(a curved weapon for shooting arrows)** is displayed for sale in the marketplace			SDXL misinterprets "bow" as a ribbon, while our method generates it as a weapon.
glasses	Glasses on the table	**A pair of drinking glasses (containers for holding liquids)** are placed on the table			SDXL presents mixed interpretations of "glasses" (eyewear and drinking instrument), while our method clearly generates the intended drinking glasses.
bat	A boy holds a black bat	A boy holds a dark-colored (black) **baseball bat**			SDXL presents mixed interpretations of "bat" (animal and baseball equipment), while our method clearly generates the intended baseball bat.

Fig. 4. Failure case comparison between our method (left) and SDXL (right).

C. Qualitative Results

We qualitatively evaluate our method using four representative prompts containing lexical ambiguity, selected from prior studies on prompt ambiguity in T2I diffusion models [4], [5]. For each ambiguous word, we compare the image generation results produced by our method and SDXL side by side.

Table II presents the original and disambiguated prompts, the corresponding output images from our method and SDXL, and a side-by-side comparison for each ambiguous word. The results indicate that SDXL often produces images with mixed or incorrect interpretations of ambiguous words. In contrast, the proposed method generates visually unambiguous images that clearly reflect the intended meaning. For example, for the word *bow*, SDXL incorrectly visualizes a ribbon. In contrast, our method successfully disambiguates the prompt and generates a weapon. Similar improvements are observed in other cases as well, further supporting the effectiveness of our disambiguation pipeline.

D. Failure Case

Despite the overall improvement in resolving lexical ambiguity, some failure cases were observed. These failures primarily stem from misclassifications by the *Ambiguity Detection and Disambiguation Module*, which subsequently affect the performance of the entire prompt disambiguation pipeline. Fig. 4 presents one such failure case.

As shown in Fig. 4, in the original prompt "A man stuck in a jam", the word *jam* is more plausibly interpreted as referring to traffic congestion. However, the WSD model used in this study incorrectly predicted its sense as 'preserve of crushed fruit.' This erroneous sense was then propagated to the *Prompt Disambiguation Module*, resulting in the generation of an incorrect disambiguated prompt: "A man trapped in a thick layer of fruit preserve (jam) made from crushed fruit." Consequently, our method generates an image of 'A man stuck in fruit jam', which does not reflect the user's intended meaning.

IV. Conclusion

We proposed a fully automated prompt disambiguation pipeline that integrates (i) an *Ambiguity Detection and Disambiguation Module*, (ii) an LLM-based *Prompt Disambiguation Module*, and (iii) an *Image Generation Module*. When applied to Stable Diffusion XL, the pipeline increased the $\mathrm{INT_H}$ rate from 51% to 81%, reduced the $\mathrm{INT_M}$ and $\mathrm{INT_N}$ rates from 13% to 4% and from 36% to 15%, respectively. The consistency score has also increased from 40% to 60%.

Acknowledgment

This research was supported by the MSIT(Ministry of Science and ICT), Korea, under the ITRC(Information Technology Research Center) support program(IITP-2025-2020-0-01461) supervised by the IITP(Institute for Information & Communications Technology Planning & Evaluation)

References

[1] A. Ramesh, P. Dhariwal, A. Nichol, C. Chu, and M. Chen, "Hierarchical text-conditional image generation with clip latents," *arXiv preprint arXiv:2204.06125*, vol. 1, no. 2, p. 3, 2022.

[2] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, "High-resolution image synthesis with latent diffusion models," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2022, pp. 10 684–10 695.

[3] C. Saharia, W. Chan, S. Saxena, L. Li, J. Whang, E. L. Denton, K. Ghasemipour, R. Gontijo Lopes, B. Karagol Ayan, T. Salimans *et al.*, "Photorealistic text-to-image diffusion models with deep language understanding," *Advances in neural information processing systems*, vol. 35, pp. 36 479–36 494, 2022.

[4] J. C. White and R. Cotterell, "Schr\"{o} dinger's bat: Diffusion models sometimes generate polysemous words in superposition," *arXiv preprint arXiv:2211.13095*, 2022.

[5] W. Elsharif, M. Alzubaidi, J. She, and M. Agus, "Visualizing ambiguity: Analyzing linguistic ambiguity resolution in text-to-image models," *Computers*, vol. 14, no. 1, p. 19, 2025.

[6] V. Liu and L. B. Chilton, "Design guidelines for prompt engineering text-to-image generative models," in *Proceedings of the 2022 CHI conference on human factors in computing systems*, 2022, pp. 1–23.

[7] Y. He, J. Wang, K. Li, Y. Wang, L. Sun, J. Yin, M. Zhang, and X. Wang, "Enhancing intent understanding for ambiguous prompt: A human-machine co-adaption strategy," *Available at SSRN 5119629*, 2024.

[8] D. Podell, Z. English, K. Lacey, A. Blattmann, T. Dockhorn, J. Müller, J. Penna, and R. Rombach, "Sdxl: Improving latent diffusion models for high-resolution image synthesis," *arXiv preprint arXiv:2307.01952*, 2023.

[9] W. Feng, W. Zhu, T.-j. Fu, V. Jampani, A. Akula, X. He, S. Basu, X. E. Wang, and W. Y. Wang, "Layoutgpt: Compositional visual planning and generation with large language models," *Advances in Neural Information Processing Systems*, vol. 36, pp. 18 225–18 250, 2023.

[10] Z. Fan, X. Li, K. Nag, C. Fang, T. Biswas, J. Xu, and K. Achan, "Prompt optimizer of text-to-image diffusion models for abstract concept understanding," in *Companion Proceedings of the ACM Web Conference 2024*, 2024, pp. 1530–1537.

[11] M. Bevilacqua, T. Pasini, A. Raganato, and R. Navigli, "Recent trends in word sense disambiguation: A survey," in *International joint conference on artificial intelligence*. International Joint Conference on Artificial Intelligence, Inc, 2021, pp. 4330–4338.

[12] T. Blevins and L. Zettlemoyer, "Moving down the long tail of word sense disambiguation with gloss informed bi-encoders," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 2020, pp. 1006–1017.

[13] Y. Su, H. Zhang, Y. Song, and T. Zhang, "Rare and zero-shot word sense disambiguation using z-reweighting," in *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2022, pp. 4713–4723.

[14] C. Fellbaum, *WordNet: An electronic lexical database*. MIT press, 1998.

[15] M. Honnibal and I. Montani, "spaCy 2: Natural language understanding with Bloom embeddings, convolutional neural networks and incremental parsing," 2017, to appear.

[16] OpenAI. (2025) Introducing GPT-4.1 in the API. Accessed: Jul. 21, 2025. [Online]. Available: https://openai.com/index/gpt-4-1/