

# WikiAutoGen: Towards Multi-Modal Wikipedia-Style Article Generation

Zhongyu Yang<sup>1, 2\*, †</sup>, Jun Chen<sup>3, 1\*</sup>, Dannong Xu<sup>1, 4 †</sup>, Junjie Fei<sup>1</sup>  
Xiaoqian Shen<sup>1</sup>, Liangbing Zhao<sup>1</sup>, Chun-Mei Feng<sup>5</sup>, Mohamed Elhoseiny<sup>1</sup>  
<sup>1</sup>King Abdullah University of Science and Technology,  
<sup>2</sup>Lanzhou University <sup>3</sup>Meta AI <sup>4</sup>The University of Sydney <sup>5</sup>IHPC, A\*STAR  
{zhongyu.yang, jun.chen, junjie.fei, xiaoqian.shen,  
liangbing.zhao, mohamed.elhoseiny}@kaust.edu.sa  
daxu8019@uni.sydney.edu.au, fengcm.ai@gmail.com

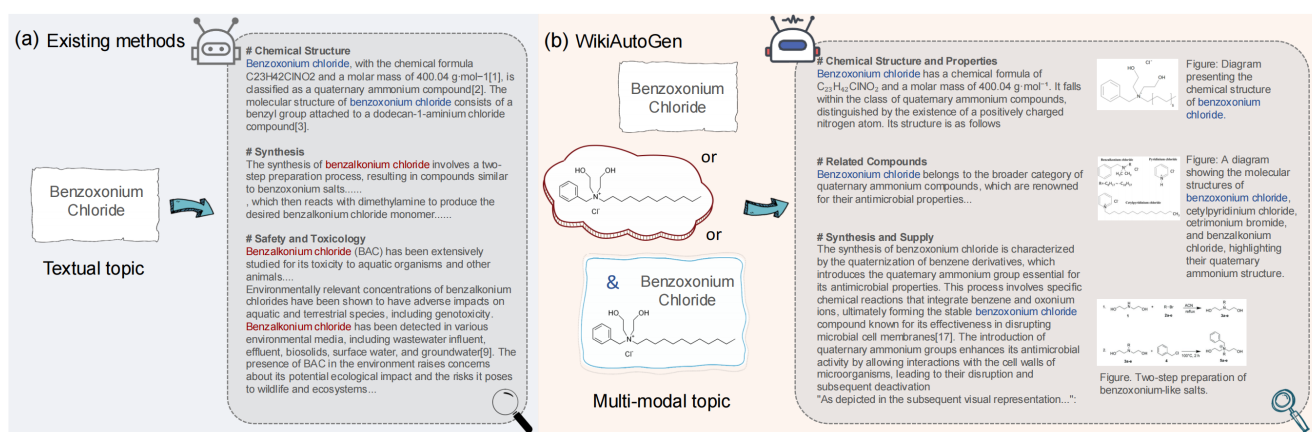


Figure 1. **Comparison of existing text-only article generation methods and our proposed WikiAutoGen.** Existing approaches [14, 28] rely exclusively on textual sources, often producing inconsistent or inaccurate results. For example, in (a), the target topic is ‘Benzoxonium Chloride’, yet the baseline incorrectly generates information about ‘Benzalkonium Chloride’. In contrast, our WikiAutoGen framework integrates both visual and textual modalities to generate coherent multimodal content. Additionally, WikiAutoGen employs a multi-perspective self-reflection mechanism, significantly improving content accuracy and reliability, as illustrated in (b).

## Abstract

Knowledge discovery and collection are intelligence-intensive tasks that traditionally require significant human effort to ensure high-quality outputs. Recent research has explored multi-agent frameworks for automating Wikipedia-style article generation by retrieving and synthesizing information from the internet. However, these methods primarily focus on text-only generation, overlooking the importance of multimodal content in enhancing informativeness and engagement. In this work, we introduce WikiAutoGen, a novel system for automated multi-modal Wikipedia-style article generation. Unlike prior approaches, WikiAutoGen retrieves and integrates relevant images alongside text, enriching both the depth and visual appeal of generated content. To further improve fac-

tual accuracy and comprehensiveness, we propose a multi-perspective self-reflection mechanism, which critically assesses retrieved content from diverse viewpoints to enhance reliability, breadth, and coherence, etc. Additionally, we introduce WikiSeek, a benchmark comprising Wikipedia articles with topics paired with both textual and image-based representations, designed to evaluate multimodal knowledge generation on more challenging topics. Experimental results show that WikiAutoGen outperforms previous methods by 8%-29% on our WikiSeek benchmark, producing more accurate, coherent, and visually enriched Wikipedia-style articles. Our code and examples are available at <https://wikiautogen.github.io/>

## 1. Introduction

Knowledge discovery and content generation are essential for organizing and disseminating information, but they re-

<sup>1</sup>\* Equal contribution

<sup>2</sup>† Work done at KAUST

main time-consuming and intelligence-intensive, requiring substantial human effort to collect, structure, and verify information. With the advent of large-scale AI models like large language models (LLMs) [2, 5, 8, 22, 35], there is growing potential to automate knowledge collection, synthesis, and organization in a more efficient and scalable manner [14, 28]. Such automation not only accelerates knowledge discovery but also enhances accessibility, making information more readily available and up to date.

Recently, several methods, such as Storm [28] and CoStorm [14], have been proposed to automate Wikipedia-style article generation. While they can produce Wikipedia-like content, they still face key limitations: (1) they are limited to text-only generation and lack the ability to incorporate multimodal content such as relevant images; (2) the generated articles often lack breadth, depth, and reliability, reducing their overall informativeness and credibility.

To address these challenges, we introduce WikiAutoGen, a multi-agent framework designed to generate high-quality, multimodal Wikipedia-style articles automatically. Unlike prior works, WikiAutoGen can directly search both textual and visual information and generate multimodal content (see Figure 1), enriching article content with relevant and diverse modalities. Additionally, we propose a novel multi-perspective self-reflection module, which enables the system to self-regulate, refine, and critically evaluate its generated content. This mechanism enhances the reliability, depth, and breadth of the information by encouraging iterative improvement and multi-source validation.

To advance the development and evaluation of multimodal knowledge generation, we introduce **WikiSeek**, a new benchmark designed to tackle challenging topics comprising both textual and visual components. Existing benchmarks primarily focus on text generation or cover only straightforward topics (see Table 1). In contrast, WikiSeek is multimodal and specifically targets more complex subjects with limited coverage on Wikipedia, making it significantly harder for current methods to retrieve and synthesize comprehensive information. This increases the challenge of content generation, pushing models to explore deeper, enhance their retrieval capabilities, and improve their ability to handle underexplored subjects.

Extensive experiments demonstrate that WikiAutoGen significantly outperforms existing methods in generating high-quality textual and visual content. We evaluated text quality across nine key dimensions and image quality across four essential criteria. Experimental results show that WikiAutoGen outperforms prior methods by 8%–29% in textual quality and 11%–14% in image quality, demonstrating consistent gains across the input topics of text-only, image-only, and image-text tasks.

Our contributions can be summarized as follows:

- We introduce WikiSeek, a new multimodal benchmark















DATASETS	Dataset Statistics		
	Type	Retrieval Modality	Difficulty Levels
Surfer100 [18]			easy
FreshWiki [28]			easy
IRP [3]			n/a
WildSeek [14]			n/a
WikiSeek (Ours)	 	 	high

Table 1. **Comparison of WikiSeek with existing benchmarks.** Modalities are indicated by  (text) and  (images). Difficulty levels are categorized based on the average number of characters in corresponding Wikipedia pages: difficult (<500 characters), medium (500–2000 characters), and easy (>2000 characters).

designed for evaluating Wikipedia-style article generation, featuring challenging topics with limited existing coverage, represented through both text and images.

- We propose WikiAutoGen, a multimodal article generation framework that synthesizes comprehensive content by effectively integrating textual and visual inputs. On WikiSeek, WikiAutoGen achieves 8%–29% improvements over the previous best model in textual generation.
- We develop a novel multi-perspective self-reflection module, which iteratively enhances readability, informativeness, and coherence by incorporating feedback from diverse roles, including reader, writer, and editor.

## 2. Related Work

**Automatic Expository Writing.** LLMs have shown strong performance in automatic expository writing, particularly in generating Wikipedia-style articles [4, 17, 21, 33, 37, 41]. Despite their strength in traditional Natural Language Generation (NLG), LLMs still struggle to produce long-form text that is coherent and logically structured [10, 26, 29, 34, 36]. To address this, [32] proposed using domain-specific keywords that are progressively refined into full passages through multi-stage generation. Notably, Shao et al. [14, 28] highlighted the crucial role of pre-writing strategies, identifying them as a key factor in improving article quality. Recently, some automatic writing systems expanded their knowledge boundaries through mindmaps and tree-based methods [3, 19, 29, 40]. However, while these iterative content planning methods effectively leverage widely accessible information for common topics, they remain largely ineffective in less-explored or niche domains due to the scarcity of structured prior knowledge [11, 42]. Meanwhile, existing methods are limited to generating text-based articles and fail to incorporate other modalities, such as visual elements. This inherent constraint in data input inevitably leads to incomplete information and reduced readability, making knowledge acquisition more challenging. Conversely, our WikiAutoGen is the first multimodal writing system that integrates visual content and retrieves knowledge across multiple modalities, allowing visual in-

formation to complement textual content by capturing details that may otherwise be overlooked.

**Self-reflection.** Recent progress in optimizing LLMs through self-reflection mechanisms is significant. The core idea is to enable models to analyze and refine their outputs through self-generated feedback. Existing approaches construct feedback sources from three strategies: (1) The LLM conducts iterative self-evaluation [30]; (2) A separately trained critic module provides specialized feedback [12]; (3) External knowledge from sources like Wikipedia and browsers is integrated [1, 20]. Specifically, REFINER [24] demonstrated that a trained critic module can enhance reasoning without fine-tuning the reasoning module, supporting feedback mechanism optimization. Further, some methods [7, 45] introduced feedback mechanisms based on error-type templates and context-hypothesis mappings. Recent studies [25, 31, 38, 39, 44] focus on LLMs’ in-context learning. They design prompt templates to help models generate feedback from historical outputs or patterns. However, the multi-perspective self-reflection method proposed by [43], which relies on a Navigator-Reasoner heuristic interaction, is limited to closed-loop LLM discussions without external knowledge acquisition, resulting in restricted information richness and verifiability. In contrast, our approach enhances multi-perspective self-reflection with multi-web search, addressing complex topic exploration and retrieval challenges while enabling multi-dimensional control over topic and article quality.

### 3. WikiSeek benchmark

#### 3.1. Task Definition

Given a topic as text ( $T$ ), image ( $I$ ), or a combination of both ( $T, I$ ), the objective is to generate a Wikipedia-style article ( $A$ ) that integrates relevant knowledge and is supported by verifiable references ( $R$ ). This task is particularly important in domains such as investigative journalism, scientific research, and market analysis, where the generation of accurate, well-sourced content is essential.

#### 3.2. WikiSeek Construction

A key challenge in this task is the lack of a suitable benchmark that effectively encompasses multimodal topics. Existing benchmarks, however, remain largely text-centric, thus failing to adequately reflect the complexity of multimodal content generation. To address this limitation, we introduce WikiSeek, a new benchmark specifically designed to evaluate more challenging topics that incorporate both text and images. WikiSeek establishes a robust evaluation framework, enabling a more comprehensive and reliable assessment of multimodal knowledge generation in practice. In the following sections, we detail the construction process of this benchmark.

**Benchmark construction pipeline.** Our WikiSeek benchmark is designed with two key objectives: (1) to evaluate multimodal article generation where both the input and output include text and images; and (2) to target underexplored topics on Wikipedia that present greater challenges for retrieval and synthesis.

We select topics from the WikiWeb2M dataset [6], which comprises approximately 2 million English Wikipedia articles containing both text and images. To identify challenging topics, we focus on articles where the main content has fewer than 500, 300, and 100 characters, categorizing them as hard, very hard, and extremely hard, respectively. Typically, these more challenging or rare topics have minimal coverage on Wikipedia, representing less-documented and lesser-known subjects.

**Topic filtering and quality control.** To curate a high-quality benchmark that includes both challenging and meaningful topics, we implement a rigorous topic filtering process. First, we retain only Wikipedia articles with fewer than 500 characters, ensuring a focus on underexplored and more difficult topics. We then sample hundreds of topics associated with images from the WikiWeb2M dataset [6].

However, some topics, such as “1997 in Japan” and “.bh”, are either overly general or semantically underspecified, making them unsuitable for evaluation. To address this, we manually verify all selected topics and remove those that lack meaningful content or pose evaluation challenges. After this filtering process, we obtain a final set of 300 topics, evenly distributed across three difficulty levels (hard, very hard, and extremely hard), with 100 topics per level. These topics are represented in one of three formats: text-only, image-only, or a combination of both.

### 4. Method

Writing high-quality multimodal Wikipedia-style articles usually requires a coordinated multi-agent system that effectively breaks down the process into distinct stages, including outline generation, web-based material retrieval, and article synthesis. Beyond these stages, maintaining quality necessitates collaboration across roles, ensuring the article is well-structured, accurate, and engaging. To address these challenges, we propose WikiAutoGen framework. This framework facilitates structured collaboration among specialized agents, enabling comprehensive topic exploration, multi-modal content generation, and coherent generation. In the following sections, we provide a detailed breakdown of WikiAutoGen and its core components.

#### 4.1. WikiAutoGen Framework

Our WikiAutoGen framework consists of several key components that work collaboratively to generate high-quality multimodal Wikipedia-style articles, as illustrated in Figure 2. Each module plays a distinct role in the article genera-

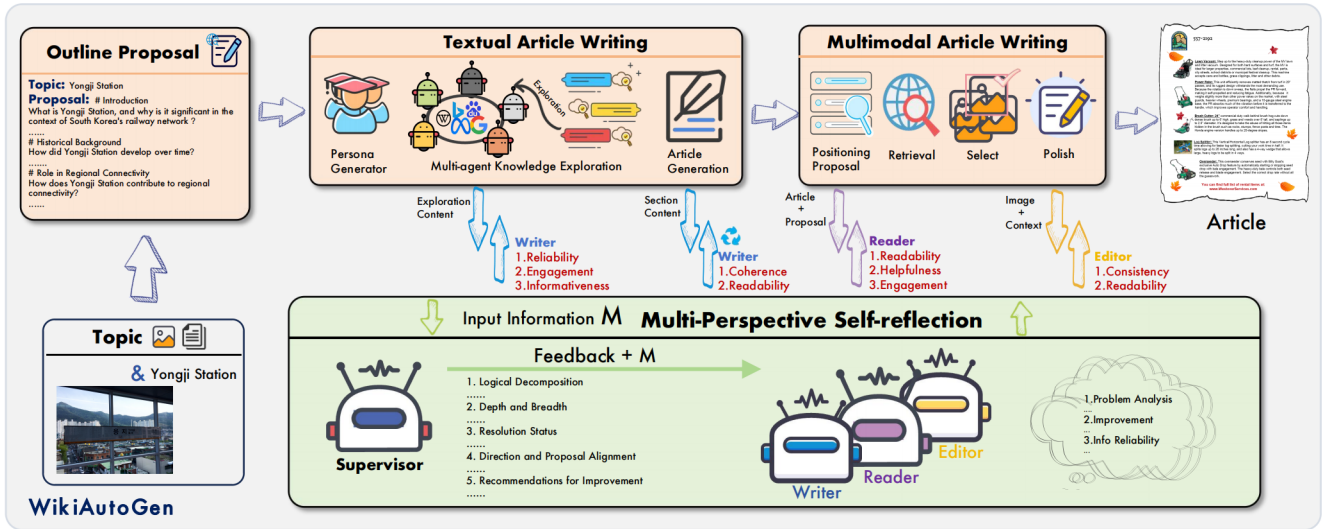


Figure 2. **Overview of WikiAutoGen**, our multimodal framework for Wikipedia-style article generation. The pipeline includes three main stages: (1) an **Outline Proposal** module that structures the article outline based on the multimodal topic input (image and text); (2) a **Textual Article Writing** module involving persona generation, multi-agent collaborative exploration, and article drafting; and (3) a **Multimodal Article Writing** module that incorporates relevant images through positioning proposals, retrieval, selection, and final polishing. The entire generation process is enhanced by a **Multi-Perspective Self-Reflection** module, leveraging supervisory and agent-specific feedback (writer, reader, editor) to iteratively improve article quality in terms of coherence, readability, and engagement, etc.

tion process: 1). **Outline Proposal Module**. This module converts the given text and image topics into structured outline proposals, laying the foundation for content organization. 2). **Textual Article Writing Module**. This stage involves multiple sub-components, including a persona generator, a multi-agent discussion system, and the article generation process, ensuring the content is well-structured and contextually rich. 3). **Multi-Perspective Self-reflection Module**. This component evaluates the generated text from multiple viewpoints, including those of a writer, reader, and editor, providing constructive feedback to refine and enhance the article. 4). **Multimodal Article Writing Module**. This final stage integrates visual content, consisting of image positioning proposals, image retrieval, image selection, and multimodal refinement, ensuring a cohesive and well-balanced article presentation. These components collectively enable WikiAutoGen to produce high-quality and multi-modal Wikipedia-style articles.

## 4.2. Outline Proposal

Given a multimodal topic, the first step is to interpret the input and generate topic-related outlines to guide knowledge exploration and information retrieval. We achieve it by leveraging LLMs [13] and external search tools.

For text-based topics, the LLM analyzes the input, identifies relevant subtopics, and generates a structured outline to facilitate knowledge exploration. For image-based topics, we utilize Google Vision Search to retrieve metadata, including descriptions and contextual information. Named entity recognition (NER) [9] is then applied to extract the top 10 most frequent entities as query keywords. These

keywords, combined with the original topic, are then fed into the LLM to generate a well-structured outline. In the case of image-text topics, we combine insights from both modalities by extracting subtopics from the text and retrieving image metadata. The LLM then refines the topic and synthesizes a cohesive outline, integrating textual and visual information to guide comprehensive knowledge retrieval.

## 4.3. Textual Article Writing

The textual article writing process involves multiple components working together to generate a well-structured and informative article. It consists of a persona generator, multi-agent knowledge exploration, and article generation.

**Persona generator**. Given a draft outline, the LLM generates  $n$  distinct personas relevant to the topic (where  $n$  is a customizable parameter,  $n \geq 1$ ), each acting as an independent agent. The LLM assigns specific objectives to each agent based on their role. These agents are equipped with access to external web search tools and contribute to a more comprehensive and well-supported article.

**Multi-agent knowledge exploration**. The knowledge exploration stage involves a fixed agent, the *asker*, and  $n$  LLM-generated agents who are assigned specific roles. The *asker* iterates through the outline, posing targeted questions, while the other agents search the internet for relevant information. They then share findings, discuss them, and refine their understanding. During the discussion, they also interact with the multi-perspective self-reflection module, which provides feedback and improvement advice from a writer’s perspective on *reliability*, *engagement*, *consistency*, and *informativeness*. This iterative process ensures a well-



rounded knowledge base before article generation.

**Article generation.** After gathering web knowledge, the next step is to summarize the collected content and generate the textual article using an LLM-based writing agent. Once the initial draft is produced, the agent iteratively sends each generated section to the multi-perspective self-reflection module for feedback and refines each paragraph. This module evaluates the article from a writer’s perspective, providing suggestions to enhance coherence and readability. The writing agent incorporates these refinements, progressively improving the text to produce the final article.

#### 4.4. Multi-Perspective Self-reflection.

Writing a high-quality Wikipedia-style article requires addressing multiple aspects, including topic consistency, readability, engagement, and informativeness, to provide an optimal reading experience. Therefore, we introduce a multi-perspective self-reflection module that systematically evaluates and refines content across these dimensions. This module takes four distinct viewpoints and assesses the article from seven perspectives.

**Perspectives.** Our multi-perspective self-reflection focuses on the seven key criteria to improve the paper writing quality. They include *reliability*, *engagement*, *informativeness*, *coherence*, *readability*, *consistency* and *helpfulness*. We provide a detailed explanation for them in the Appendix.

**Supervisor viewpoint.** The supervisor assesses whether the generated content fully addresses the questions posed by the asker, evaluates the article’s depth, breadth, and coherence, and reviews the effectiveness of the multi-agent discussion. Additionally, it evaluates whether the generated content aligns with the topic and proposed outlines. Based on these criteria, the supervisor provides an evaluation and passes the feedback to the next role. Depending on the specific needs, the next role can be the *writer*, *reader*, or *editor*.

**Writer viewpoint.** From the writer’s viewpoint, the primary focus is on the multi-agent knowledge exploration and article generation stages. The writer evaluates whether the generated content maintains coherence, ensures engagement, verifies factual accuracy, and upholds logical consistency. Based on these assessments, the writer provides targeted improvement suggestions.

For instance, to enhance coherence, it may be recommended to rearrange sentences or add transitional words and phrases. To improve readability, it might suggest simplifying complex concepts. Finally, the writer responds with a set of targeted and refined suggestions.

**Reader viewpoint.** To create a high-quality multi-modal article, it is essential to effectively integrate textual content with relevant images to enhance reader engagement and readability. To achieve this, our framework first employs an LLM to propose suitable image placements within the article and generate descriptive content specifying the types of

images to include. This initial proposal is then reviewed by the multi-perspective self-reflection module, which evaluates the image positioning and the generated image descriptions from the perspective of readability, engagement, and helpfulness. Based on this assessment, the module provides constructive feedback, ensuring that visual content is effectively integrated to enrich the overall reader experience.

**Editor viewpoint.** After inserting images into the generated article, there may still be discrepancies or inconsistencies between the visual content and the corresponding textual descriptions. To address this, our framework sends the images along with their related text segments to the multi-perspective self-reflection module. This module evaluates the alignment and coherence between the images and their accompanying texts from an editorial viewpoint. It provides targeted suggestions, such as refining the image captions, adjusting image placement, or enhancing textual explanations to better reflect visual content. This final step ensures enhanced relevance, coherence, and readability between visual and textual elements.

#### 4.5. Multi-modal Article Writing

Following textual generation, we incorporate relevant visual content to enhance the article’s readability and expressiveness. This multimodal integration process involves several stages: image positioning proposal, image retrieval, image selection, and finally, an article refinement step that seamlessly integrates the images and text.

**Image positioning proposal.** After generating the complete textual article, an LLM-based agent is employed to propose appropriate placements and corresponding descriptions for relevant images. These initial proposals are then refined through interaction with the multi-perspective self-reflection module, which evaluates their relevance, coherence, and engagement from the reader’s perspective.

**Image retrieval.** We then retrieve relevant images by performing searches based on multiple sources, including general image search engines, Wikipedia, and the websites mentioned in the references. After that, we can obtain a list of relevant image candidates.

**Image selection.** We first use the CLIP model [27] to rank retrieved images based on semantic similarity to the generated captions, selecting the top-3 candidates. Subsequently, we leverage a multi-modal model [13] to further evaluate these candidates and select the most contextually appropriate image for inclusion in the article.

**Article Polishing.** After finalizing image selection, we integrate the chosen images into the article and proceed to a polishing stage. Due to potential discrepancies between textual content and visual figures, we employ a multi-modal model to revise the entire article, enhancing coherence and consistency across modalities. Additionally, this multimodal model interacts with the multi-perspective self-reflection

Methods	Content Quality				Informativeness		Reliability	Engagement		Average
	Alignment	Consistency	Relevance	Repetition	Breadth	Depth	Verifiability	Engagement	Novelty	
Text as Topic										
oRAG [1]	61.35	73.96	76.04	71.11	63.30	52.42	45.47	57.51	45.24	60.71
Storm [28]	72.49	79.13	71.22	69.47	65.62	62.61	52.41	58.80	55.58	65.26
Co-Storm [14]	78.05	84.10	75.11	75.40	68.42	67.70	58.20	61.02	61.61	69.96
OmniThink [42]	70.53	79.67	72.41	69.26	63.55	61.21	48.57	57.39	53.21	63.98
WikiAutoGen (Ours)	81.68	90.87	88.02	83.62	79.64	73.73	70.69	71.14	69.21	78.73
Image as Topic										
oRAG	50.10	72.16	50.92	65.47	42.01	43.26	33.90	40.66	36.91	48.38
Storm	45.80	59.60	45.99	46.38	42.69	39.92	34.23	42.38	35.17	43.57
Co-Storm	47.00	61.40	44.85	47.76	41.98	41.83	35.03	41.99	37.69	44.39
OmniThink	43.61	58.29	43.03	45.67	38.63	42.26	29.18	38.31	33.57	41.39
WikiAutoGen (Ours)	82.57	88.75	87.20	80.22	77.24	74.99	68.41	69.36	68.69	77.49
Image-Text as Topic										
oRAG	60.08	75.16	70.94	72.24	58.38	50.57	42.47	55.01	43.95	58.76
Storm	67.20	75.29	66.64	64.33	61.61	58.26	49.21	56.25	51.27	61.12
Co-Storm	70.15	79.29	67.31	68.89	61.90	61.22	52.44	57.05	55.68	63.77
OmniThink	64.56	75.33	64.63	63.64	57.44	56.38	43.04	54.14	48.86	58.67
WikiAutoGen (Ours)	85.26	90.63	88.44	82.11	79.31	75.20	68.59	68.79	71.06	78.82

Table 2. **Comparison of article generation performance for textual content.** We evaluate content quality, informativeness, reliability, and engagement under three input modalities (Text-only, Image-only, and Image-Text) on our WikiSeek benchmark.

module, obtaining editorial feedback to further refine the integrated content.

## 5. Experiment

### 5.1. Experiment setup

**Implementation Details.** For the language model (LM) components of WikiAutoGen, we employ zero-shot prompting implemented using the DSPy framework [15] in conjunction with GPT-4o [13], GPT-4o-mini, and GPT-o3-mini [23]. Specifically, we use GPT-o3-mini for the multi-perspective self-reflection module due to its strong reasoning capabilities, GPT-4o for the multimodal knowledge exploration tasks, and GPT-4o-mini for all other remaining tasks. WikiAutoGen retrieves real-time web information via Serper’s API<sup>1</sup>, with each query returning up to 5 web pages. Throughout the experiments, we maintain consistent settings by fixing the LM temperature at 1.0 and the *top-p* value at 0.9. For evaluation, we utilize GPT-4o as the evaluator. To further validate the results, we conduct more evaluations using two distinct evaluators, Gemini2.5-Flash [8] and Prometheus2 [16], as detailed in Appendix B.

**Evaluation metrics.** We evaluate the generated multimodal articles through separate assessments of their textual and visual content.

- **Text quality evaluation.** Following prior evaluation frameworks [14, 28], we utilize GPT-4o as the evaluator to assess generated articles across nine criteria grouped into four main aspects:

- *Content quality*: alignment, relevance, repetition, and consistency;

- *Informativeness*: breadth and depth;
- *Reliability*: verifiability;
- *Engagement*: engagement and novelty.

- **Image quality evaluation.** As there are no existing benchmarks specifically designed for evaluating multimodal Wikipedia-style article generation, we propose an evaluation method inspired by the textual evaluation frameworks [14, 28] to evaluate image quality. Specifically, we assess image quality based on four criteria: *image-text coherence*, *engagement*, *helpfulness*, and *information supplement* (the image’s ability to provide additional useful information beyond the textual context).

**Baselines** We compare WikiAutoGen with four representative LLM-based baseline frameworks for automated expository writing on our WikiSeek benchmark:

- **Outline-driven RAG (oRAG)** generates articles guided by outlines produced by Self-RAG [1].
- **Storm** [28] leverages LLM-driven conversations and outlines from diverse perspectives to generate Wikipedia-style content.
- **Co-STORM** [14] utilizes collaborative discourse among multiple LLM agents to explore and discover unknown information.
- **OmniThink** [42] enhances article quality through iterative expansion and reflection, emulating human slow-thinking to increase knowledge density.

Since these baselines originally lack multimodal capabilities, we equip them with image retrieval functionalities to enable a fair multimodal comparison. Specifically, each baseline can also retrieve images via Google image search, extract relevant metadata, and search for images based on generated textual descriptions.

<sup>1</sup><https://serper.dev/>

Modules			Content Quality				Informativeness		Reliability	Engagement		Average
Multi-agent	Outline proposal	Self-reflection	Alignment	Consistency	Relevance	Repetition	Breadth	Depth	Verifiability	Engagement	Novelty	
✗	✗	✗	50.63	62.87	53.10	51.19	48.58	44.35	43.64	45.54	39.08	48.78
✓	✗	✗	71.81	76.38	72.67	68.25	64.36	69.16	64.20	57.04	55.17	66.56
✗	✓	✗	79.11	86.20	80.93	77.21	72.65	62.13	<b>69.19</b>	64.45	64.11	72.88
✗	✗	✓	75.91	84.08	82.20	75.56	73.24	66.66	65.41	63.03	58.36	71.60
✗	✓	✓	77.68	85.57	84.73	78.49	75.08	68.85	65.59	65.73	66.45	73.80
✓	✓	✓	<b>82.57</b>	<b>88.75</b>	<b>87.20</b>	<b>80.22</b>	<b>77.24</b>	<b>74.99</b>	68.41	<b>69.36</b>	<b>68.69</b>	<b>77.49</b>

Table 3. **Ablation study.** We study the impact of individual modules (Multi-agent knowledge exploration, outline proposal, and self-reflection) on article generation performance for text content. The input modality is image-only.

Method	Coherence	Engagement	Helpfulness	Info. Sup.	Average
<i>Text as Topic</i>					
oRAG	57.36	56.26	63.61	51.90	57.28
Storm	55.20	45.97	51.89	43.97	49.26
Co-Storm	57.62	48.64	54.19	45.07	51.38
OmniThink	58.82	49.36	54.93	47.55	52.67
WikiAutoGen (Ours)	<b>70.12</b>	<b>66.31</b>	<b>74.76</b>	<b>64.78</b>	<b>68.99</b>
<i>Image as Topic</i>					
oRAG	61.21	52.07	58.039	45.96	54.32
Storm	52.59	43.98	49.53	41.84	46.99
Co-Storm	54.32	45.61	51.55	41.56	48.26
OmniThink	59.88	50.62	56.13	47.23	53.47
WikiAutoGen (Ours)	<b>71.90</b>	<b>61.69</b>	<b>77.63</b>	<b>63.88</b>	<b>68.78</b>
<i>Image-Text as Topic</i>					
oRAG	66.38	56.31	62.94	49.78	58.85
Storm	59.67	50.26	55.78	46.78	53.12
Co-Storm	54.28	45.65	51.29	42.88	48.53
OmniThink	61.76	51.40	57.88	49.97	55.25
WikiAutoGen (Ours)	<b>72.24</b>	<b>70.29</b>	<b>72.11</b>	<b>69.29</b>	<b>70.98</b>

Table 4. **Comparison of article generation performance for image content.** We evaluate textual generation for image-text coherence, image engagement, image helpfulness, and information supplement on our WikiSeek benchmark.

## 5.2. Experimental results

**Textual content comparison on WikiSeek.** We demonstrate our results in the Table 2. The results indicate that WikiAutoGen consistently achieves the highest average scores across all evaluation dimensions—content quality, informativeness, reliability, and engagement—highlighting its comprehensive effectiveness in article generation.

For text-only inputs, our WikiAutoGen achieves an average score of 78.73, significantly outperforming the best baseline (Co-Storm, 69.96) by approximately 8.8 points, demonstrating superior coherence, alignment, and informativeness. In image-only scenarios, the improvement is even more pronounced, with WikiAutoGen obtaining 77.49, surpassing the strongest baseline (oRAG, 48.38) by approximately 29.1 points, reflecting its exceptional capability to extract meaningful textual insights from visual content. For combined image-text topics, our model maintains its advantage with an average score of 78.82, showing a clear improvement (+ 15.05 points) over the next-best baseline (Co-Storm, 63.77). Overall, the substantial performance gains demonstrate that WikiAutoGen excels at synthesizing cohesive and engaging content across diverse inputs.

**Image content comparison on WikiSeek.** Table 4 compares the performance of image content across different article generation methods. Our WikiAutoGen consistently achieves the highest scores across all image evaluation metrics for all three input modalities.

Specifically, our method significantly improves upon baseline methods, outperforming the next-best method by

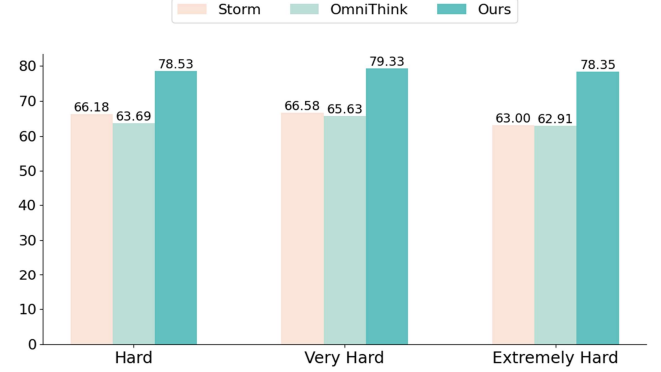


Figure 3. **Ablation study across different data difficulty levels.** We compare our WikiAutoGen with Storm and OmniThink on three difficulty categories: hard (300–500 characters), very hard (100–300 characters), and extremely hard (<100 characters).

approximately 11.34 points on image-text coherence (72.24 vs. Storm’s 59.67) under the image-text topics. Similarly, for image-only topics, our approach excels particularly in helpfulness (77.63) and coherence (71.90), demonstrating WikiAutoGen’s superior capability in selecting images that meaningfully complement textual content and provide additional useful information. Overall, these results highlight WikiAutoGen’s effectiveness in integrating images to enhance coherence, engagement, and informativeness, substantially advancing the overall quality and readability of multimodal articles.

## 5.3. Ablation studies

**Ablation on different components of WikiAutoGen.** We conduct an ablation study to analyze the individual contributions of three core components to textual article generation from image-only topics. Specifically, we examine the impact of components, including multi-agent knowledge exploration, outline proposal, and multi-perspective self-reflection. Results are shown in Table 3. Specifically, without the outline proposal, the system simply retrieves a single image and extracts its description from metadata. Without multi-agent knowledge exploration, only a single static agent responds to the asker.

Incorporating the multi-agent module improves performance from 48.78 (baseline without modules) to 66.56 (+17.78 points), highlighting its effectiveness in collaborative knowledge exploration. Using the outline proposal alone further increases performance to 72.88 (+24.10 points), underscoring its importance for content structur-

ing. The self-reflection module individually achieves 71.60 (+22.82 points), indicating its strength in refining coherence and consistency. Combining all three modules results in the highest performance (77.49), validating their complementary roles in generating coherent, informative, and engaging articles from image-only inputs.

**Ablation on different difficulty levels.** In our WikiSeek benchmark, topics are grouped into three difficulty levels based on article length (character count): hard (300–500 characters), very hard (100–300 characters), and extremely hard (fewer than 100 characters), with 100 examples per level. We evaluate text-only inputs and present the average textual evaluation results in Figure 3. The results indicate that our WikiAutoGen consistently outperforms all baseline methods across all three difficulty levels. Notably, as the topic difficulty increases (from hard to extremely hard), the performance gap between WikiAutoGen and Storm widens from 12.35 points to 15.35 points, with similar trends observed against other baselines. These results demonstrate the robustness and stability of WikiAutoGen in effectively handling highly challenging and underexplored topics.

#### 5.4. Compared with Commercial Deep Research

Method	Text as Topic	Image as Topic	Image-Text as Topic	Average	Compute Time
OpenAI	92.75	91.95	94.50	93.06	~ 30 min
Google	81.91	—	—	—	~ 12 min
Grok	88.35	81.70	86.13	85.06	~ 10 min
WikiAutoGen	88.58	89.55	88.89	89.01	~ 8 min

Table 5. Comparison of commercial models’ article generation performance with Prometheus2 [16].

We evaluate the performance of WikiAutoGen against leading commercial models on the WikiSeek benchmark, spanning three input modalities. As shown in Table 5, WikiAutoGen achieves an impressive average score of 89.01, closely approaching OpenAI (93.06), while significantly outperforming Grok (85.06) and Google (81.91). WikiAutoGen delivers consistently strong results across text (88.58), image (89.55), and image-text (88.89) inputs, demonstrating robust cross-modal capabilities. In terms of efficiency, WikiAutoGen generates articles in just 8 minutes, making it over  $3.75\times$  faster than OpenAI (30 minutes), the fastest among the commercial baselines. This substantial speed advantage underscores its practical scalability and suitability for real-world, time-sensitive applications.

#### 5.5. Human evaluation

To evaluate the quality of the generated Wikipedia-style articles, we conduct a human evaluation study via Amazon Mechanical Turk (AMT)<sup>2</sup>. We randomly sample 100 text-only topics from the WikiSeek benchmark dataset and perform pairwise comparisons between articles generated by our method (WikiAutoGen), Storm, and OmniThink. Each

<sup>2</sup><https://www.mturk.com/>

topic is evaluated by three independent participants in randomized order. Participants first answer the question: “Do you think adding images would improve comprehension of the topic?” As shown in Figure 4 (left), 97.7% of participants agree that images improve topic comprehension.

Additionally, participants answer multiple-choice questions, including: (1) Which article is the easiest to understand? (2) Which article is the most engaging in terms of narrative, examples, or overall presentation? (3) Which article provides the most comprehensive background information and in-depth analysis? (4) Which article is your overall favorite? The question order is randomized to mitigate potential evaluation bias. As illustrated in Figure 4 (right), participants consistently prefer articles generated by WikiAutoGen over those by Storm and OmniThink across all evaluation criteria.

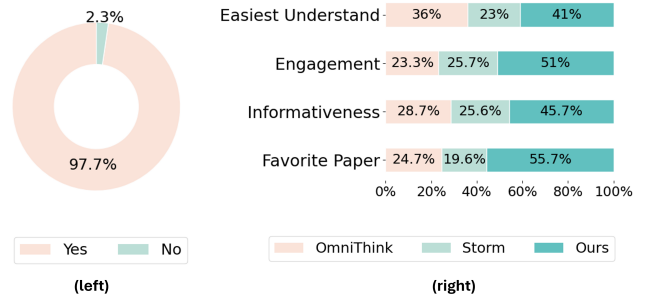


Figure 4. **Human evaluation study.** We randomly sample text-only topics and conduct comparative evaluations between WikiAutoGen, Storm, and OmniThink. **Left:** Participants respond to the question, “Do you think adding images would improve comprehension of the topic?”. **Right:** Participants answer multiple-choice questions evaluating readability, engagement, informativeness, and overall preference.

#### 6. Conclusion

In this paper, we introduced WikiAutoGen, a comprehensive multi-agent framework designed for automated multimodal Wikipedia-style article generation. WikiAutoGen integrates both visual and textual content, significantly enhancing the depth, informativeness, and engagement of generated articles. To address key limitations in prior work, we proposed a novel multi-perspective self-reflection mechanism, which systematically improves article coherence, reliability, and overall quality. Furthermore, we presented WikiSeek, a challenging multimodal benchmark specifically crafted to evaluate the performance of models in generating content for less-explored topics. Experimental results demonstrated that WikiAutoGen substantially outperforms existing state-of-the-art baselines across both textual and visual evaluation metrics. Particularly notable were the improvements in content quality, informativeness, and reader engagement, validating the effectiveness of integrating multimodal inputs and iterative self-reflection.



## References

- [1] Akari Asai, Zeqiu Wu, Yizhong Wang, Avirup Sil, and Hananeh Hajishirzi. Self-rag: Learning to retrieve, generate, and critique through self-reflection. *ArXiv*, abs/2310.11511, 2023. 3, 6
- [2] Jinze Bai et al. Qwen technical report. *arXiv preprint arXiv:2309.16609*, 2023. Accessed: 2025-03-04. 2
- [3] Nishant Balepur, Jie Huang, and Kevin Chen-Chuan Chang. Expository text generation: Imitate, retrieve, paraphrase. *ArXiv*, abs/2305.03276, 2023. 2
- [4] Maciej Besta, Florim Memedi, Zhenyu Zhang, Robert Gerstenberger, Guangyuan Piao, Nils Blach, Piotr Nyczyk, Marcin Copik, Grzegorz Kwaśniewski, Jurgen Muller, Lukas Gianinazzi, Aleš Kubček, Hubert Niewiadomski, Aidan O’Mahony, Onur Mutlu, and Torsten Hoeffler. Demystifying chains, trees, and graphs of thoughts. In *ArXiv*, 2024. 2
- [5] DeepSeek-AI: Xiao Bi et al. Deepseek llm: Scaling open-source language models with longtermism. *arXiv preprint arXiv:2401.02954*, 2024. Accessed: 2025-03-04. 2
- [6] Andrea Burns, Krishna Srinivasan, Joshua Ainslie, Geoff Brown, Bryan A. Plummer, Kate Saenko, Jianmo Ni, and Mandy Guo. Wikiweb2m: A page-level multimodal wikipedia dataset. *ArXiv*, abs/2305.05432, 2023. 3
- [7] Steven Coyne. Template-guided grammatical error feedback comment generation. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics: Student Research Workshop*, pages 94–104, Dubrovnik, Croatia, 2023. Association for Computational Linguistics. 3
- [8] Google DeepMind. Gemini: Multimodal language model. <https://deepmind.google/technologies/gemini/>, 2023. Accessed: 2025-03-04. 2, 6
- [9] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805, 2018. 4
- [10] Zican Dong, Tianyi Tang, Junyi Li, Wayne Xin Zhao, and Ji-Rong Wen. BAMBOO: A comprehensive benchmark for evaluating long text modeling capacities of large language models. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 2086–2099, Torino, Italia, 2024. ELRA and ICCL. 2
- [11] Fan Gao, Hang Jiang, Rui Yang, Qingcheng Zeng, Jinghui Lu, Moritz Blum, Dairui Liu, Tianwei She, Yuang Jiang, and Irene Li. Evaluating large language models on wikipedia-style survey generation. In *Annual Meeting of the Association for Computational Linguistics*, 2023. 2
- [12] Zhibin Gou, Zhihong Shao, Yeyun Gong, Yelong Shen, Yujia Yang, Nan Duan, and Weizhu Chen. Critic: Large language models can self-correct with tool-interactive critiquing. *ArXiv*, abs/2305.11738, 2023. 3
- [13] Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, et al. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*, 2024. 4, 5, 6
- [14] Yucheng Jiang, Yijia Shao, Dekun Ma, Sina J. Semnani, and Monica S. Lam. Into the unknown unknowns: Engaged human learning through participation in language model agent conversations. In *Conference on Empirical Methods in Natural Language Processing*, 2024. 1, 2, 6
- [15] Omar Khattab, Arnav Singhvi, Paridhi Maheshwari, Zhiyuan Zhang, Keshav Santhanam, Sri Vardhamanan, Saiful Haq, Ashutosh Sharma, Thomas T. Joshi, Hanna Moazam, Heather Miller, Matei Zaharia, and Christopher Potts. DSPy: Compiling declarative language model calls into self-improving pipelines. In *Proceedings of the Twelfth International Conference on Learning Representations (ICLR)*, 2024. 6
- [16] Seungone Kim, Juyoung Suk, Shayne Longpre, Bill Yuchen Lin, Jamin Shin, Sean Welleck, Graham Neubig, Moontae Lee, Kyungjae Lee, and Minjoon Seo. Prometheus 2: An open source language model specialized in evaluating other language models. *arXiv preprint arXiv:2405.01535*, 2024. 6, 8
- [17] Ishita Kumar, Snigdha Viswanathan, Sushrita Yerra, Alireza Salemi, Ryan A Rossi, Franck Dernoncourt, Hanieh Deilamsalehy, Xiang Chen, Ruiyi Zhang, Shubham Agarwal, et al. Longlamp: A benchmark for personalized long-form text generation. *arXiv preprint arXiv:2407.11016*, 2024. 2
- [18] Irene Li, Alexander R. Fabbri, Rina Kawamura, Yixin Liu, Xiangru Tang, Jaesung Tae, Chang Shen, Sally Ma, Tomoe Mizutani, and Dragomir R. Radev. Surfer100: Generating surveys from web resources, wikipedia-style. In *International Conference on Language Resources and Evaluation*, 2021. 2
- [19] Yi Liang, You Wu, Honglei Zhuang, Li Chen, Jiaming Shen, Yiling Jia, Zhen Qin, Sumit K. Sanghai, Xuanhui Wang, Carl Yang, and Michael Bendersky. Integrating planning into single-turn long-form text generation. *ArXiv*, abs/2410.06203, 2024. 2
- [20] Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegrefe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, Sean Welleck, Bodhisattwa Prasad Majumder, Shashank Gupta, Amir Yazdanbakhsh, and Peter Clark. Self-refine: Iterative refinement with self-feedback. *ArXiv*, abs/2303.17651, 2023. 3
- [21] Sewon Min, Kalpesh Krishna, Xinxin Lyu, Mike Lewis, Wentao Yih, Pang Koh, Mohit Iyyer, Luke Zettlemoyer, and Hananeh Hajishirzi. FActScore: Fine-grained atomic evaluation of factual precision in long form text generation. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 12076–12100, Singapore, 2023. Association for Computational Linguistics. 2
- [22] OpenAI. Chatgpt: Optimizing language models for dialogue. <https://openai.com/blog/chatgpt>, 2022. Accessed: 2025-03-04. 2
- [23] OpenAI. Openai o3-mini, 2025. 6
- [24] Debjit Paul, Mete Ismayilzada, Maxime Peyrard, Beatriz Borges, Antoine Bosselut, Robert West, and Boi Faltings. Refiner: Reasoning feedback on intermediate representations. In *Conference of the European Chapter of the Association for Computational Linguistics*, 2023. 3

- [25] Alexandre Piché, Aristides Milios, Dzmitry Bahdanau, and Chris Pal. Llms can learn self-restraint through iterative self-reflection. *ArXiv*, abs/2405.13022, 2024. 3
- [26] Haoran Que, Feiyu Duan, Liqun He, Yutao Mou, Wangchunshu Zhou, Jiaheng Liu, Wenge Rong, Zekun Moore Wang, Jian Yang, Ge Zhang, et al. Hellobench: Evaluating long text generation capabilities of large language models. *arXiv preprint arXiv:2409.16191*, 2024. 2
- [27] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *Proceedings of the 38th International Conference on Machine Learning (ICML)*, pages 8748–8763. PMLR, 2021. 5
- [28] Yijia Shao, Yucheng Jiang, Theodore A. Kanell, Peter Xu, Omar Khattab, and Monica S. Lam. Assisting in writing wikipedia-like articles from scratch with large language models. In *North American Chapter of the Association for Computational Linguistics*, 2024. 1, 2, 6
- [29] Zejiang Shen, Tal August, Pao Siangliulue, Kyle Lo, Jonathan Bragg, Jeff Hammerbacher, Doug Downey, and Joseph Chee Chang. Beyond summarization: Designing ai support for real-world expository writing tasks. *ArXiv*, abs/2304.02623, 2023. 2
- [30] Noah Shinn, Federico Cassano, Beck Labash, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. Reflexion: language agents with verbal reinforcement learning. In *Neural Information Processing Systems*, 2023. 3
- [31] Zhiqing Sun, Yikang Shen, Qinhong Zhou, Hongxin Zhang, Zhenfang Chen, David D. Cox, Yiming Yang, and Chuang Gan. Principle-driven self-alignment of language models from scratch with minimal human supervision. *ArXiv*, abs/2305.03047, 2023. 3
- [32] Bowen Tan, Zichao Yang, Maruan Al-Shedivat, Eric Xing, and Zhiting Hu. Progressive generation of long text with pre-trained language models. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4313–4324, Online, 2021. Association for Computational Linguistics. 2
- [33] Haochen Tan, Zhijiang Guo, Zhan Shi, Lu Xu, Zhili Liu, Yunlong Feng, Xiaoguang Li, Yasheng Wang, Lifeng Shang, Qun Liu, et al. Proxyqa: An alternative framework for evaluating long-form text generation with large language models. *arXiv preprint arXiv:2401.15042*, 2024. 2
- [34] Chen Tang, Hongbo Zhang, Tyler Loakman, Chenghua Lin, and Frank Guerin. Enhancing dialogue generation via dynamic graph knowledge aggregation. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4604–4616, Toronto, Canada, 2023. Association for Computational Linguistics. 2
- [35] Hugo Touvron et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024. 2
- [36] Qiyao Wang, Shiwen Ni, Huaren Liu, Shule Lu, Guhong Chen, Xi Feng, Chi Wei, Qiang Qu, Hamid Alinejad-Rokny, Yuan Lin, and Min Yang. Autopatent: A multi-agent framework for automatic patent generation. *ArXiv*, abs/2412.09796, 2024. 2
- [37] Yidong Wang, Qi Guo, Wenjin Yao, Hongbo Zhang, Xin Zhang, Zhen Wu, Meishan Zhang, Xinyu Dai, Min Zhang, Qingsong Wen, Wei Ye, Shikun Zhang, and Yue Zhang. Autosurvey: Large language models can automatically write surveys. *ArXiv*, abs/2406.10252, 2024. 2
- [38] Yifei Wang, Yuyang Wu, Zeming Wei, Stefanie Jegelka, and Yisen Wang. A theoretical understanding of self-correction through in-context alignment. *ArXiv*, abs/2405.18634, 2024. 3
- [39] Yutong Wang, Jiali Zeng, Xuebo Liu, Fandong Meng, Jie Zhou, and Min Zhang. Taste: Teaching large language models to translate through self-reflection. In *Annual Meeting of the Association for Computational Linguistics*, 2024. 3
- [40] Yilin Wen, Zifeng Wang, and Jimeng Sun. Mindmap: Knowledge graph prompting sparks graph of thoughts in large language models. In *Annual Meeting of the Association for Computational Linguistics*, 2023. 2
- [41] Yuhao Wu, Ming Shan Hee, Zhiqing Hu, and Roy Ka-Wei Lee. Spinning the golden thread: Benchmarking long-form generation in language models. *arXiv preprint arXiv:2409.02076*, 2024. 2
- [42] Zekun Xi, Wenbiao Yin, Jizhan Fang, Jialong Wu, Runnan Fang, Ningyu Zhang, Jiang Yong, Pengjun Xie, Fei Huang, and Huajun Chen. Omnithink: Expanding knowledge boundaries in machine writing through thinking. *ArXiv*, abs/2501.09751, 2025. 2, 6
- [43] Hanqi Yan, Qinglin Zhu, Xinyu Wang, Lin Gui, and Yulan He. Mirror: A multiple-perspective self-reflection method for knowledge-rich reasoning. *ArXiv*, abs/2402.14963, 2024. 3
- [44] Tianjun Zhang, Aman Madaan, Luyu Gao, Steven Zheng, Swaroop Mishra, Yiming Yang, Niket Tandon, and Uri Alon. In-context principle learning from mistakes. *ArXiv*, abs/2402.05403, 2024. 3
- [45] Yangqiaoyu Zhou, Haokun Liu, Tejes Srivastava, Hongyuan Mei, and Chenhao Tan. Hypothesis generation with large language models. *ArXiv*, abs/2404.04326, 2024. 3