

# Entity Blackboard: Structured Agent Memory for Long Document Summarization

Ruyi Liu\* Guanyun Zou\* Hongliang Dai Piji Li†

<sup>1</sup>College of Artificial Intelligence,

Nanjing University of Aeronautics and Astronautics, Nanjing, China

<sup>2</sup>The Key Laboratory of Brain-Machine Intelligence Technology,

Ministry of Education, Nanjing, China

{liuliuliu79,zouguanyun,hongldai,pjli}@nuaa.edu.cn

## Abstract

Hierarchical merging is commonly used to summarize long documents by splitting the input into chunks, summarizing each chunk independently, and merging the results into a final summary. However, because each chunk is processed independently, entity state changes across chunks are not tracked, leading to coherence and consistency issues in the final summary. We propose ENTITYBB (ENTITY BLACKBOARD), an agent memory framework that equips chunk-based LLM summarizers with structured entity memory. During sequential chunk processing, the model records and updates entity states through tool calls, while automatic safeguards and model-driven rejection limit noisy writes. Experiments on three long document summarization benchmarks show that ENTITYBB consistently improves over a controlled hierarchical merging baseline across ROUGE and BERTScore, while NarrativeFactScore indicates improved source-grounded factuality. Targeted human and LLM-as-a-Judge evaluations further indicate fewer entity-level issues and higher judged summary quality. Code: [github.com/heimaliu/EntityBB](https://github.com/heimaliu/EntityBB).

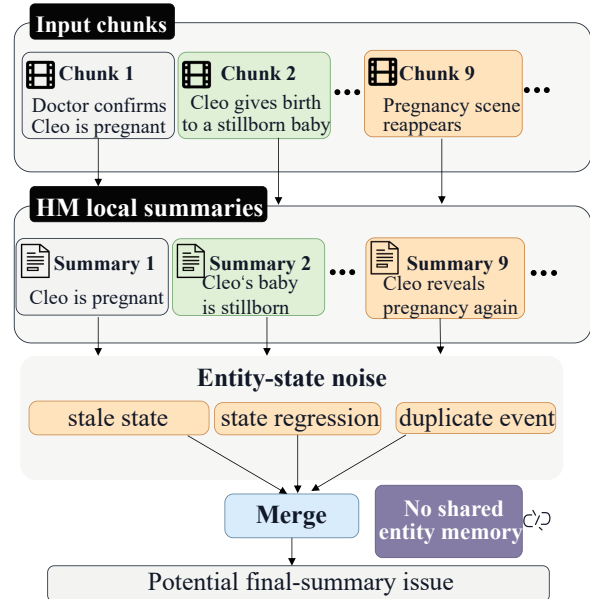


Figure 1: Entity-state noise in baseline hierarchical merging. In the *Roma* (2018) screenplay from MovieSum, different chunks revisit overlapping events involving Cleo’s pregnancy. Because chunks are summarized independently, the final merge can receive stale or repeated evidence about pregnancy and stillbirth, even when each local summary is plausible in isolation.

## 1 Introduction

Long document summarization commonly relies on hierarchical merging: the input is partitioned into chunks, each summarized independently, then merged into a final output (Ou and Lapata, 2025). While this divide-and-conquer strategy handles documents of arbitrary length, it creates a fundamental blind spot: entity states established in one chunk are invisible to the model processing subsequent chunks. This is especially problematic for narrative texts such as movie scripts and novels, where summarization requires tracking event

causality, character behavior, and long-range temporal dependencies (Zhao et al., 2022; Kryściński et al., 2022; Saxena and Keller, 2024a). When chunks are summarized independently, the intermediate summaries can accumulate entity-state noise, including stale states, repeated events, entity confusion, and unsupported entity facts. Such noise is then propagated to the final merge stage, where locally plausible chunk summaries may provide redundant or incompatible evidence for the same entity, degrading final-summary coherence and factual precision. Without explicit entity tracking, for example, a model may describe an event as if it is happening for the first time even after its outcome has already been established in an earlier chunk;

\*Equal contribution.

†Corresponding author.

Figure 1 gives a concrete MovieSum example.

Recent large language models (LLMs) perform well on many summarization tasks (Pu et al., 2023), yet producing coherent summaries of long documents remains challenging: models tend to lose critical information, introduce factual inconsistencies, and fail to maintain coherence across distant sections of the input (Koh et al., 2023). CAHM (Ou and Lapata, 2025) enriches hierarchical merging with source document context to reduce hallucinations, while content selection approaches (Ladhak et al., 2020; Liu et al., 2022; Saxena and Keller, 2024b) attempt to shorten the input by selecting salient sections; however, both strategies risk omitting details that are essential for document-level coherence in the final summary.

Multi-agent systems have recently emerged as a promising direction for long document summarization (Guo et al., 2024). Chang et al. (2024) systematically evaluates book-length summarization strategies, and Zhang et al. (2024) proposes multi-agent collaboration for long-context tasks. Building on these, HM-SR (Jeong et al., 2025) first generates an initial summary through hierarchical merging and then applies factuality judging and iterative refinement, NEXUSSUM (Kim and Kim, 2025) introduces hierarchical agents with iterative compression achieving state-of-the-art results with a 123B-parameter model, and SUMMQ (Wang et al., 2026) employs adversarial quizzer-summarizer agents to improve factuality and coverage. Despite their diversity, strong recent systems often rely on larger models such as GPT-4o or Mistral-Large (123B), or add multiple specialized agent roles beyond standard hierarchical merging, which motivates our controlled comparison using the same Qwen3-4B backbone for HM and ENTITYBB.

Rather than scaling up model size, we ask whether explicit external memory can improve a Qwen3-4B summarization backbone. Recent work on agent memory (Xu et al., 2025) shows that LLMs can store, retrieve, and update structured information through callable tools. Concurrently, modern LLMs have developed strong tool-calling capabilities (Yang et al., 2025). These observations suggest that entity tracking in chunk-based summarization does not have to rely on the model’s implicit memory. It can instead be handled by a structured, tool-accessible module with deterministic safeguards against noisy writes.

We propose ENTITYBB (ENTITY BLACK-

BOARD), an agent memory framework that equips a chunk-based LLM summarizer with structured entity memory exposed as its sole callable tool. During chunk-by-chunk processing, the model actively records and updates entity states through tool calls, building a shared memory that persists across all chunks. The accumulated entity memory is then provided to the final synthesis stage to guide coherent summary generation. To prevent the memory from growing uncontrollably, ENTITYBB combines automatic name matching and content deduplication with model-driven rejection of duplicate or contradictory facts.

We evaluate ENTITYBB on three long document summarization benchmarks: MovieSum (Saxena and Keller, 2024a), MENSA (Saxena and Keller, 2024b), and BookSum (Kryściński et al., 2022). A 4B-parameter model (Qwen3-4B) using ENTITYBB consistently improves controlled hierarchical merging baselines across ROUGE (Lin, 2004) and BERTScore (Zhang et al., 2020) metrics, while remaining competitive with strong large-model systems and fine-tuned summarization models. Ablation studies confirm that entity memory, word limit control, and deterministic safeguards contribute to the gains, while targeted entity-consistency and source-grounded factuality evaluations show fewer entity-level issues and a higher proportion of source-supported atomic facts. Our contributions are as follows:

- We propose ENTITYBB, an agent memory framework that exposes structured entity memory as a callable tool for chunk-based summarization across ordered document chunks during sequential reading.
- We design automatic name matching, content deduplication, and model-driven rejection safeguards that keep the entity memory compact and consistent during chunk processing.
- We show that explicit entity memory substantially improves hierarchical merging under the same Qwen3-4B backbone across three narrative benchmarks in both reference-based metrics and source-grounded factuality.

## 2 Related Work

### 2.1 Long Document Summarization

Prior work addresses documents exceeding a model’s context window through architectural scal-

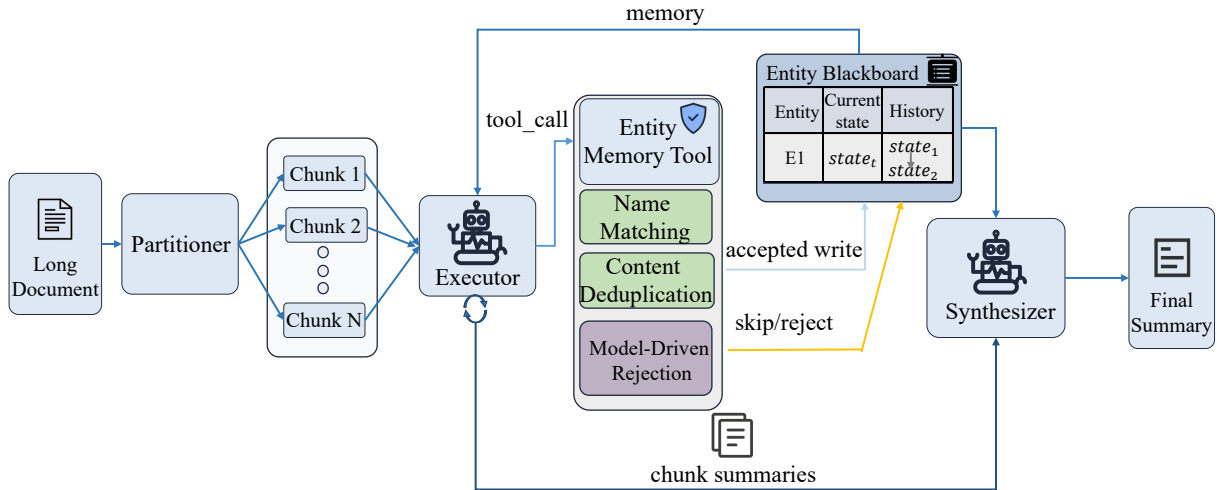


Figure 2: Overview of the ENTITYBB framework. The Partitioner divides a long document into fixed-size chunks. The Executor processes each chunk sequentially and writes entity facts through the Entity Memory Tool. Fuzzy name matching, content deduplication, and model-driven rejection guard updates to the Entity Blackboard. After all chunks are processed, the Synthesizer merges the chunk summaries with the accumulated entity states and a chunk-adaptive word limit.

ing, content selection, and hierarchical processing. Long-context architectures extend effective context via sparse attention, memory-efficient encoding, or long-context finetuning (Zaheer et al., 2020; Beltagy et al., 2020; Guo et al., 2022), yet entity tracking and coherence still degrade over long inputs (Liu et al., 2024). Content selection shortens the input by extracting salient sections (Ladhak et al., 2020; Liu et al., 2022; Saxena and Keller, 2024b), but can discard details needed for document-level entity coherence (Koh et al., 2023). Hierarchical merging chunks, summarizes, and merges the input (Pang et al., 2023); CAHM (Ou and Lapata, 2025) adds source context to reduce hallucinations, but does not explicitly maintain entity states across chunks. ENTITYBB instead processes all chunks with a persistent entity memory, giving the final merge a structured cross-chunk reference that reduces information loss and entity noise during the final synthesis stage.

## 2.2 Multi-Agent Summarization

Multi-agent summarization methods distribute reading, critique, or refinement across agents to improve factuality and coverage (Du et al., 2024; Zhang et al., 2024; Zhao et al., 2024; Fang et al., 2024). Book-length evaluations find that hierarchical merging can produce more coherent summaries than incremental updating, although with a detail trade-off (Chang et al., 2024). Recent systems such as HM-SR (Jeong et al., 2025), NEXUS-SUM (Kim and Kim, 2025), and SUMMQ (Wang

et al., 2026) add judge/refinement, iterative compression, or quizzer-summarizer roles, often with larger backbones. Our work follows a different route: a single Qwen3-4B model gains persistent entity memory through one callable tool, rather than additional agent roles.

## 2.3 Agent Memory

External memory lets LLM agents store and retrieve information beyond the current context. Generative Agents and A-Mem organize experience through memory streams or agentic memory for open-ended interaction (Park et al., 2023; Xu et al., 2025). MemoryBank studies long-term memory for LLM interaction, whereas MemGPT develops virtual-context management for document analysis and multi-session dialogue (Zhong et al., 2024; Packer et al., 2023). Chunk-based summarization has narrower requirements: the system must track entity states across ordered chunks while suppressing redundant, stale, or conflicting writes. ENTITYBB instead maintains structured entity states during sequential chunk processing for final summary synthesis, using one entity-state tool plus safeguards for canonical names, duplicate facts, and model-driven rejection.

# 3 ENTITYBB Framework

## 3.1 Overview

ENTITYBB processes a long document in three stages (see Figure 2). The Partitioner divides the

input into fixed-size chunks based on token count. The Executor then reads each chunk, records and updates entity states through a structured memory tool, and produces a chunk-level summary. After all chunks are processed, the Synthesizer merges the chunk summaries into a single coherent document summary, using the accumulated entity memory to keep entity states aligned across chunks. We describe this entity state management interface in Section 3.3, since it is the only tool available to the model throughout chunk processing.

### 3.2 Document Partitioning

Given a long document, the Partitioner splits it into chunks of  $T$  tokens (default:  $T=10,000$ ) using the model’s tokenizer. Cut positions are aligned to sentence boundaries to avoid splitting mid-sentence. If the remaining text after the last cut is shorter than  $T/3$  tokens, it is merged into the final chunk to avoid producing very short segments. This produces  $K$  chunks  $\{c_1, c_2, \dots, c_K\}$  that together cover the entire document without overlap.

### 3.3 Entity-Aware Chunk Processing

#### Algorithm 1 Entity-aware chunk processing with guarded writes

---

**Require:** chunk  $c_i$ , blackboard  $B$ , maximum tool rounds  $R$

- 1: Render  $B$  and  $c_i$  to the Executor
- 2: **for**  $r = 1$  to  $R$  **do**
- 3:   Obtain  $\text{write}(a, e, f)$ ; force one call when  $r = 1$
- 4:   *Model-driven rejection*
- 5:   **if**  $a = \text{reject}$  **then**
- 6:     Keep  $B$  unchanged
- 7:   **else**
- 8:     *Fuzzy name matching*
- 9:     **if**  $e$  matches an entry by substring or overlap  $\geq 50\%$  **then**
- 10:        $e' \leftarrow$  matched canonical entity
- 11:     **else**
- 12:        $e' \leftarrow e$
- 13:     **end if**
- 14:     *Content deduplication*
- 15:     **if**  $f$  exact-matches after normalization **then**
- 16:       Return skip and keep  $B$  unchanged
- 17:     **else if**  $\text{sim}(f, \text{states}(e')) \geq 85\%$  **then**
- 18:       Return skip and keep  $B$  unchanged
- 19:     **else**
- 20:       Archive previous state and write  $f$  as current
- 21:     **end if**
- 22:   **end if**
- 23: **end for**
- 24: Generate chunk summary  $s_i$  conditioned on  $c_i$  and  $B$
- 25: **return** updated  $B$  and  $s_i$

---

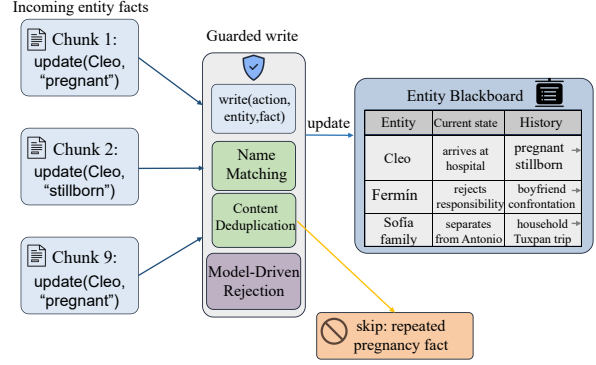


Figure 3: Example of a guarded update to the Entity Blackboard. Incoming entity facts are routed through the write interface and checked by name matching, content deduplication, and model-driven rejection. Accepted updates refresh the current entity state while preserving earlier states in history; repeated facts trigger a no-write skip outcome.

The Executor processes each chunk  $c_i$  in two phases: entity-memory writing followed by chunk-summary generation. Algorithm 1 summarizes the guarded write policy.

**Entity Memory Tool** The Entity Blackboard exposes  $\text{write}(\text{action}, \text{entity}, \text{fact})$  for incremental state maintenance, analogous in design to TaoType’s predictor for incremental search input (Yu et al., 2026).  $\text{action}$  is add, update, or reject;  $\text{entity}$  is a stable identifier; and  $\text{fact}$  is a state description.  $\text{add}$  creates an entity;  $\text{update}$  records state changes; and  $\text{reject}$  declines duplicate, conflicting, or uninformative facts. Deterministic checks return skip for duplicates. Each entity stores a *current state* and *history*; updates archive the prior state. Figure 3 illustrates this update path with a concrete *Roma* example.

**Guarded Write Policy** ENTITYBB combines deterministic safeguards with model-driven rejections. Algorithm 1 gives the exact guarded-write flow and thresholds: (a) **fuzzy name matching** canonicalizes aliases and redirects matched add calls to update; (b) **content deduplication** returns skip for facts already covered by current or historical states; and (c) **model-driven rejection** uses reject when the current memory and chunk indicate that a candidate fact is already recorded, stale, conflicting, or unsupported. Together, these checks make the blackboard accept genuinely new state changes rather than another noisy summary of the chunk. This is important because the Synthesizer later treats the memory as trusted evidence; re-

peated or stale writes can otherwise amplify cross-chunk conflicts during the final merge. The first round forces one tool call to ensure memory engagement, while later rounds may continue tool use or stop; Appendix A analyzes the maximum number of rounds.

**Chunk Summary Generation** After tool calling, the model generates a detailed summary of the current chunk without an explicit intermediate word limit, conditioned on both the chunk text and the updated entity memory. The chunk summary is therefore written with access to previously established entity states.

### 3.4 Blackboard-Guided Synthesis

After all  $K$  chunks have been processed, the Synthesizer receives the  $K$  chunk summaries  $\{s_1, s_2, \dots, s_K\}$  and the complete entity memory. The final summary is produced under a maximum word limit rather than a required output size. We set the word limit as  $L_{\max} = K \times W$ , where  $W$  is the per-chunk word limit multiplier. This chunk-adaptive word limit scales summary capacity with document length without using reference summaries or an oracle length signal. We set  $W=150$  in our experiments and provide the sensitivity analysis for this setting in Appendix A.

The entity memory is formatted as a structured knowledge section and included in the synthesis prompt alongside the chunk summaries. The Synthesizer can then reduce entity-state noise across chunk summaries by referencing the accumulated entity states and the full document context. The final merge therefore receives a stable cross-chunk reference rather than relying only on local summaries from individual chunks.

## 4 Experimental Setup

### 4.1 Datasets

We evaluate controlled runs on three narrative summarization benchmarks: **MovieSum** (Saxena and Keller, 2024a), which contains screenplays with recurring characters and scene-level state changes, using 100 examples; **MENSA** (Saxena and Keller, 2024b), a movie scene saliency dataset built from scripts and plot summaries, using its full 50-example test set; and **BookSum** (Kryściński et al., 2022), using 50 evaluation examples from literary book chapters with long-range character developments and extended narrative dependencies.

### 4.2 Baselines

We compare against methods from three categories, following the grouping in Table 1.

**Zero-Shot** Mistral-Large (123B) and GPT-4o are long-context zero-shot baselines reported by Kim and Kim (2025), while GPT-5, DeepSeek-R1, and Qwen3-32B are prompting baselines reported by Wang et al. (2026). These baselines do not use task-specific training.

**Supervised Fine-Tuning** CachED (Saxena et al., 2025) uses gradient caching to train BART-Large on documents exceeding 500K tokens without truncation, while SLED (Ivgi et al., 2023) applies sliding-window encoding and decoder fusion to aggregate evidence across segmented long inputs.

**Multi-Agent** HM-SR (Jeong et al., 2025), also referred to as Agent-as-Judge, uses hierarchical merging to produce an initial summary, then applies factuality judging and iterative refinement, while SummQ (Wang et al., 2026) employs adversarial quizzer-summarizer agents.

For controlled comparison, we also implement a plain hierarchical merging (HM) baseline using the same Qwen3-4B backbone, chunking strategy, and chunk-summary settings. This baseline does not use the Entity Blackboard, the chunk-adaptive final-summary word limit, or any judge/refinement stage from HM-SR.

### 4.3 Implementation Details

We use Qwen3-4B (Yang et al., 2025) as the backbone model, served via vLLM on a single RTX 3090 GPU. Documents are partitioned into chunks of 10,000 tokens using the Qwen3 tokenizer, with cut positions aligned to sentence boundaries. All variants generate chunk summaries without an explicit intermediate word limit. For blackboard-enabled variants, the Executor makes up to 6 rounds of entity memory tool calls per chunk (the first round is forced). When word-limit control is enabled, the Synthesizer is instructed to keep the final summary below  $K \times 150$  words, where  $K$  is the number of chunks. We use temperature 0.3 for all generation steps throughout all experiments.

### 4.4 Evaluation Metrics

We use BERTScore-F1 (Zhang et al., 2020), computed with DeBERTa-XLarge-MNLI, as the primary metric because it captures semantic similarity beyond lexical overlap. We also report ROUGE-1,

| Method (Model)                | MovieSum         |              |              |              | MENSA            |              |              |              | BookSum          |              |              |              |
|-------------------------------|------------------|--------------|--------------|--------------|------------------|--------------|--------------|--------------|------------------|--------------|--------------|--------------|
|                               | BS <sub>F1</sub> | R-1          | R-2          | R-L          | BS <sub>F1</sub> | R-1          | R-2          | R-L          | BS <sub>F1</sub> | R-1          | R-2          | R-L          |
| <b>Zero-Shot</b>              |                  |              |              |              |                  |              |              |              |                  |              |              |              |
| Mistral-Large                 | 55.50            | 39.22        | 10.53        | <b>22.55</b> | 54.80            | 37.43        | 10.52        | 21.52        | 46.42            | 19.63        | 2.99         | 12.00        |
| GPT-4o                        | —                | —            | —            | —            | 52.80            | 26.60        | 6.70         | <b>22.50</b> | 47.24            | 20.30        | 3.50         | 17.68        |
| GPT-5                         | —                | —            | —            | —            | 60.44            | 37.38        | 9.14         | 17.11        | 58.55            | 23.98        | 5.69         | 12.38        |
| DeepSeek-R1                   | —                | —            | —            | —            | 56.82            | 27.63        | 7.66         | 14.82        | 55.85            | 18.86        | 4.69         | 9.71         |
| Qwen3-32B                     | —                | —            | —            | —            | 55.76            | 23.49        | 5.58         | 12.77        | 56.51            | 20.19        | 4.68         | 10.68        |
| <b>Supervised Fine-Tuning</b> |                  |              |              |              |                  |              |              |              |                  |              |              |              |
| CachED (BART-Large)           | —                | —            | —            | —            | —                | —            | —            | —            | 54.40            | 42.80        | 10.50        | 18.80        |
| SLED (BART-Large)             | —                | —            | —            | —            | —                | —            | —            | —            | 52.40            | 38.90        | 7.50         | 15.80        |
| <b>Multi-Agent</b>            |                  |              |              |              |                  |              |              |              |                  |              |              |              |
| HM-SR (GPT-4o-mini)           | 59.32            | 31.31        | 8.81         | 18.62        | 60.22            | 34.26        | 9.74         | 13.46        | —                | —            | —            | —            |
| SUMMQ (GPT-4o)                | —                | —            | —            | —            | 62.76            | 41.58        | <b>10.96</b> | 18.24        | <b>61.49</b>     | <b>44.62</b> | <b>11.14</b> | <b>20.38</b> |
| HM (Qwen3-4B)                 | 56.07            | 35.93        | 8.18         | 16.51        | 55.58            | 34.14        | 8.17         | 15.91        | 53.43            | 27.57        | 5.75         | 13.77        |
| ENTITYBB (Qwen3-4B)           | <b>61.98</b>     | <b>45.54</b> | <b>11.03</b> | 18.06        | <b>62.94</b>     | <b>45.36</b> | 10.67        | 18.11        | 61.03            | 42.11        | 10.96        | 18.97        |

Table 1: Main results on three long document summarization benchmarks. Parentheses indicate the backbone model when it is not already part of the method name. BS<sub>F1</sub> denotes BERTScore-F1; R-1, R-2, and R-L are ROUGE F1 scores. Baseline results are sourced from previous studies (Kim and Kim, 2025; Wang et al., 2026; Saxena et al., 2025; Jeong et al., 2025). HM and ENTITYBB are controlled runs with our Qwen3-4B implementation; bold marks the best reported score in each metric column.

ROUGE-2, and ROUGE-L F1 (Lin, 2004) as complementary lexical metrics. For blackboard quality, we audit average entity entries, duplicate facts, and duplicate entity records, where the last metric counts cases where aliases for the same entity are stored separately. For targeted entity consistency, we count four entity-level issue types in final summaries: contradiction, duplicate event, entity confusion, and unsupported entity fact. For qualitative evaluation, we employ LLM-as-a-Judge scoring across four dimensions: comprehensiveness, coherence, detail coverage, and overall quality for each compared system. We further use Narrative-FactScore (Jeong et al., 2025) to evaluate whether atomic facts in a generated summary are supported by the source document.

## 5 Results and Analysis

Table 1 presents the main results across all three benchmarks, with controlled HM and ENTITYBB runs reported alongside cited systems.

**MovieSum** On this dataset, ENTITYBB achieves 61.98 BERTScore-F1, outperforming the cited HM-SR result (59.32) and the prompt-based Mistral-Large (55.50). Compared to the controlled plain HM baseline on the same Qwen3-4B model, ENTITYBB gains +5.91 BERTScore-F1 and +9.61 ROUGE-1. This dataset contains screenplay-style inputs with recurring characters and scene-level state changes, so the gain is consistent with the intended role of the blackboard.

**MENSA** On movie scene saliency summarization, ENTITYBB reaches 62.94 BERTScore-F1, outperforming all prompt-based baselines, the cited HM-SR result, and the multi-agent SummQ system. The +7.36 BERTScore-F1 gain over the controlled plain HM baseline suggests that the framework helps track salient events and character states across distant scenes. The improvement is also reflected in ROUGE-1 (+11.22), indicating that the final summaries preserve more reference-relevant event and entity mentions.

**BookSum** On this dataset, ENTITYBB achieves 61.03 BERTScore-F1, surpassing the fine-tuned CachED (54.40) by +6.63 points and trailing SummQ (61.49) by 0.46 points. The +7.60 BERTScore-F1 gain over controlled plain HM is the largest among the three benchmarks, suggesting that the blackboard is not merely exploiting screenplay structure. Book chapters contain fewer explicit scene boundaries than scripts, so this result provides a useful check on whether entity memory transfers beyond movie-style inputs.

**Summary** Across all three benchmarks, ENTITYBB consistently outperforms the controlled plain HM baseline by +5.9 to +7.6 BERTScore-F1 under the same Qwen3-4B backbone. The average improvement of +6.96 BERTScore-F1 suggests that the full framework, including structured entity memory, word-limit control, and deterministic safeguards, benefits chunk-based summarization across reference styles. This comparison evaluates

the full framework against plain hierarchical merging, while the ablations below separate the effects of entity memory and the final word limit.

### 5.1 Ablation Study

We first isolate the effects of entity memory and the final-summary word limit through a two-factor ablation on MovieSum (Table 2). This design separates the blackboard from output-length control: the no-BB variant keeps the final word limit, while the no-word-limit variant keeps the blackboard but removes the final synthesis constraint.

| Variant         | BB  | WL  | BS <sub>F1</sub> | R-1          | R-2          | R-L          |
|-----------------|-----|-----|------------------|--------------|--------------|--------------|
| ENTITYBB (Full) | yes | yes | <b>61.98</b>     | <b>45.54</b> | <b>11.03</b> | 18.06        |
| w/o BB          | no  | yes | 58.31            | 45.10        | 10.16        | <b>18.33</b> |
| w/o Word Limit  | yes | no  | 59.26            | 36.28        | 8.26         | 17.28        |
| HM              | no  | no  | 56.07            | 35.93        | 8.18         | 16.51        |

Table 2: Ablation study on MovieSum. BB indicates the Entity Blackboard; WL indicates the chunk-adaptive final-summary word limit.

Adding the Entity Blackboard improves BERTScore-F1 both with the word limit (+3.67) and without it (+3.19), showing that the gain is not merely an artifact of output-length control. The word limit also improves BERTScore-F1 without the blackboard (+2.24) and with the blackboard (+2.72). The full system performs best on the primary metric and on ROUGE-1/2, indicating that entity memory and chunk-adaptive word limit control provide complementary benefits. This pattern clarifies the role of each component: the plain HM baseline uses the same chunk summaries, while ENTITYBB adds a persistent entity state and an adaptive final-summary word limit before synthesis. To test whether a blackboard without safeguards is sufficient, we compare ENTITYBB with NaiveBB, a write-only blackboard variant that removes the guarded update policy (Table 3). NaiveBB can still store cross-chunk facts, but it cannot canonicalize aliases, skip repeated facts, or reject stale updates before they enter memory.

| Variant  | BS <sub>F1</sub> | R-1          | R-2          | R-L          |
|----------|------------------|--------------|--------------|--------------|
| NaiveBB  | 59.30            | 42.81        | 10.11        | 17.42        |
| ENTITYBB | <b>61.98</b>     | <b>45.54</b> | <b>11.03</b> | <b>18.06</b> |

Table 3: Naive blackboard ablation on MovieSum. NaiveBB keeps the external write interface but removes fuzzy name matching, content deduplication, and model-driven rejections.

Table 3 shows that NaiveBB improves over plain HM but still trails the full system by 2.68 BERTScore-F1 points. This gap suggests that the blackboard interface is useful only when the memory itself is kept clean enough for the final Synthesizer to rely on it. Figure 4 helps explain why NaiveBB still lags behind ENTITYBB: without fuzzy name matching, content deduplication, or model-driven rejection, the memory grows to 36.1 entities per sample, with 54.3 duplicate facts and 9.5 duplicate entity records on average. The full ENTITYBB reduces duplicate facts to 4.2 and eliminates duplicate entity records in this audit, suggesting that guarded writes are needed for external memory to remain compact and reliable enough for the final summary synthesis stage.

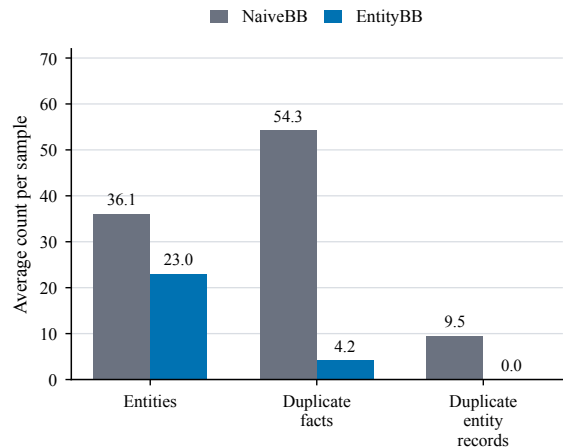


Figure 4: Memory audit for NaiveBB and ENTITYBB on MovieSum. Duplicate entity records count cases where aliases for the same entity are stored as separate blackboard entries rather than a single canonical record.

### 5.2 Case Study

We analyze recurring pregnancy-related events for Cleo in *Roma* (2018) in Table 4. The HM baseline produces locally plausible chunk summaries, but its merge input contains stale or repeated mentions of pregnancy, labor, and stillbirth, which can reopen an already resolved event. ENTITYBB instead keeps an ordered entity history and skips duplicate writes, allowing the Synthesizer to treat the stillbirth as a resolved state before final synthesis.

### 5.3 Human Evaluation of Entity Consistency

To directly evaluate entity-state noise, two NLP master’s students conduct a blind human entity-consistency audit on MENSA. They compare anonymized summaries with the source document

| Contrast Point        | HM: Noisy Merge Signal                                                                                                                                                     | ENTITYBB: Guarded Synthesis Signal                                                                                                                                                                      |
|-----------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Resolved entity state | <b>Outdated state.</b> Local summaries separately state that Cleo is pregnant, her water breaks during the Corpus Christi massacre, and the baby is delivered stillborn.   | <b>State history.</b> The Cleo memory keeps the event sequence as a state history: pregnant → labor → baby stillborn.                                                                                   |
| Duplicate handling    | <b>Redundant signal.</b> Later English and Spanish chunks revisit pregnancy, hospital, and labor scenes, so the final merge sees similar local facts again.                | <b>Duplicate writes skipped.</b> Repeated pregnancy/birth writes are skipped: Ch. 8 skips 27 of 28 attempts, and the Spanish repeat in Ch. 9 is also skipped.                                           |
| Update selectivity    | <b>New and old facts mixed.</b> Late hospital summaries add context, but also restate the already resolved birth outcome.                                                  | <b>New facts only.</b> Only new hospital context is accepted; covered pregnancy and birth facts are skipped before final synthesis.                                                                     |
| Synthesis consequence | <b>Resolved event repeated.</b> Cleo suspects she is pregnant and later delivers a stillborn baby, but subsequent local summaries again return to her pregnancy and labor. | <b>Event resolved.</b> After Fermín abandons Cleo, her pregnancy is confirmed; the Corpus Christi massacre triggers her labor, and the stillbirth becomes the grief she carries into the Tuxpan rescue. |

Table 4: Case study on *Roma* (2018) from MovieSum. The colored phrases mark the contrast between noisy HM merge signals and the guarded ENTITYBB synthesis signal for Cleo’s pregnancy-related events.

and assign each issue to one of four categories: contradiction, duplicate event, entity confusion, or unsupported entity fact. Contradictions are incompatible entity states; duplicate events repeat an event as new; entity confusion assigns states or actions to the wrong entity; unsupported facts lack document evidence. The full annotation protocol, including category definitions and annotation instructions, is given in Appendix F for reproducibility.

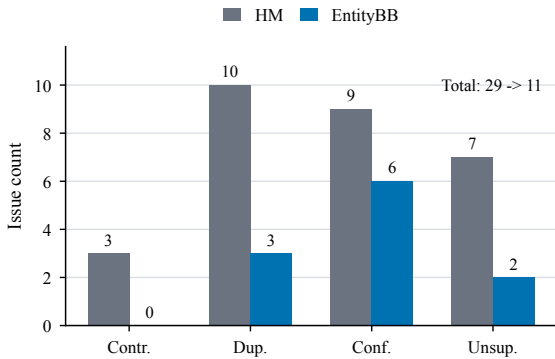


Figure 5: Human entity-consistency evaluation on MENSA. The bars report manually verified entity-level issue counts in final summaries for HM and ENTITYBB; lower is better. Contr./Dup./Conf./Unsup. denote contradiction, duplicate event, entity confusion, and unsupported entity fact.

Figure 5 shows that ENTITYBB reduces total entity-level issues from 29 to 11 on MENSA, with the largest drop in duplicate events (10 to 3). It also removes observed contradictions and reduces unsupported facts, indicating a more reliable syn-

thesis signal rather than greater verbosity.

#### 5.4 LLM-as-a-Judge Evaluation

To complement automatic metrics, DeepSeek-V4 scores anonymized MENSA summaries on four 1–5 dimensions: comprehensiveness, coherence, detail coverage, and overall quality (Figure 6). Each API call contains one summary with no system name; the prompt instructs the judge not to reward verbosity by itself. ENTITYBB improves all dimensions, especially detail coverage (+1.04) and comprehensiveness (+0.92), indicating stronger judged content coverage and summary quality rather than rewards for longer or more verbose outputs.

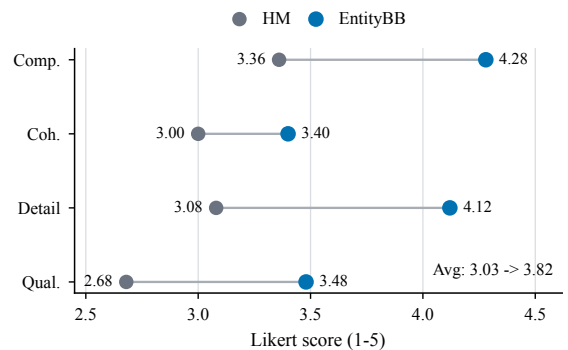


Figure 6: LLM-as-a-Judge evaluation on MENSA. Scores use a 1–5 scale over four quality dimensions.

#### 5.5 Source-Grounded Factuality

Following NarrativeFactScore (NFS) (Jeong et al., 2025), each fixed generated summary is decomposed into atomic facts, and DeepSeek-V4-Pro

determines whether each fact is supported by the source document. For a summary with  $N_{\text{all}}$  atomic facts, of which  $N_{\text{supported}}$  are judged source-supported, we compute

$$\text{NFS} = 100 \times \frac{N_{\text{supported}}}{N_{\text{all}}}. \quad (1)$$

We use NFS only for evaluation; its judgments are not fed back to revise either system’s summaries.

| Method   | MovieSum     | MENSA        | BookSum      | Average      |
|----------|--------------|--------------|--------------|--------------|
| HM       | 89.31        | 88.70        | 85.92        | 87.98        |
| ENTITYBB | <b>96.75</b> | <b>96.28</b> | <b>94.23</b> | <b>95.75</b> |

Table 5: NarrativeFactScore on the three datasets, using DeepSeek-V4-Pro for atomic-fact factuality judgments. Higher is better.

ENTITYBB improves source-grounded factuality on all three datasets. Its average pairwise gain over the controlled HM baseline is 7.78 NFS points, supporting the conclusion that explicit entity-state memory improves factual consistency beyond ROUGE and BERTScore overlap.

## 6 Conclusion

We introduced ENTITYBB, an agent memory framework that equips chunk-based long-document summarization with guarded entity-state memory. By maintaining entity states across chunks, ENTITYBB provides the final Synthesizer with a structured cross-chunk reference alongside the local summaries. Across MovieSum, MENSA, and BookSum, ENTITYBB consistently improves the controlled HM baseline in ROUGE, BERTScore, and NarrativeFactScore. Ablations and memory audits show that guarded writes help keep the memory compact, while human entity-consistency and LLM-as-a-Judge evaluations show fewer entity-level issues and higher judged summary quality.

## Limitations

Our evaluation is limited to English narrative datasets; generalization to multilingual or code-switched documents and to domains such as scientific reports or instruction manuals remains untested. Entity-state extraction depends on the capability of the backbone model, and the current design does not adapt entity granularity to different document types. The current benchmarks also lack an extreme entity-density stress test. The NaiveBB audit and threshold analyses show that

memory fragmentation and redundant growth degrade summary quality within the range observed on MovieSum, but they do not test documents containing very large numbers of distinct entities. The blackboard also stores updates in document order without explicit event timestamps, temporal relations, or occurrence identifiers. Future work should evaluate diverse domains and languages, systematically vary the number of distinct entities, and explore adaptive entity granularity and explicit temporal metadata.

## Acknowledgments

This research is supported by the National Natural Science Foundation of China (No.62476127), the Natural Science Foundation of Jiangsu Province (No.BK20242039), the Basic Research Program of the Bureau of Science and Technology (ILF24001), the Fundamental Research Funds for the Central Universities (No.NJ2023032), the Scientific Research Starting Foundation of Nanjing University of Aeronautics and Astronautics (No.YQR21022), and the High Performance Computing Platform of Nanjing University of Aeronautics and Astronautics.

## References

- Iz Beltagy, Matthew E. Peters, and Arman Cohan. 2020. [Longformer: The Long-Document Transformer](#). *arXiv preprint arXiv:2004.05150*.
- Yapei Chang, Kyle Lo, Tanya Goyal, and Mohit Iyyer. 2024. [Booookscore: A systematic exploration of book-length summarization in the era of LLMs](#). In *The Twelfth International Conference on Learning Representations (ICLR)*.
- Yilun Du, Shuang Li, Antonio Torralba, Joshua B. Tenenbaum, and Igor Mordatch. 2024. [Improving factuality and reasoning in language models through multiagent debate](#). In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 11733–11763. PMLR.
- Jiangnan Fang, Cheng-Tse Liu, Jieun Kim, Yash Bhedaru, Ethan Liu, Nikhil Singh, Nedim Lipka, Puneet Mathur, Nesreen K. Ahmed, Franck Dernoncourt, Ryan A. Rossi, and Hanieh Deilamsalehy. 2024. [Multi-LLM text summarization](#). *arXiv preprint arXiv:2412.15487*.
- Mandy Guo, Joshua Ainslie, David Uthus, Santiago Ontanon, Jianmo Ni, Yun-Hsuan Sung, and Yinfei Yang. 2022. [LongT5: Efficient text-to-text transformer for long sequences](#). In *Findings of the Association for*

- Computational Linguistics: NAACL 2022*, pages 724–736.
- Taicheng Guo, Xiuying Chen, Yaqi Wang, Ruidi Chang, Shichao Pei, Nitesh V. Chawla, Olaf Wiest, and Xi-angliang Zhang. 2024. [Large language model based multi-agents: A survey of progress and challenges](#). In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence (IJCAI)*, pages 8048–8057.
- Maor Ivgi, Uri Shaham, and Jonathan Berant. 2023. [Efficient long-text understanding with short-text models](#). *Transactions of the Association for Computational Linguistics*, 11:284–299.
- Yeonseok Jeong, Minsoo Kim, Seung-won Hwang, and Byung-Hak Kim. 2025. [Agent-as-judge for factual summarization of long narratives](#). In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 23591–23608, Suzhou, China. Association for Computational Linguistics.
- Hyuntak Kim and Byung-Hak Kim. 2025. [Nexus-Sum: Hierarchical LLM agents for long-form narrative summarization](#). In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 10120–10157, Vienna, Austria. Association for Computational Linguistics.
- Huan Yee Koh, Jiaxin Ju, Ming Liu, and Shirui Pan. 2023. [An empirical survey on long document summarization: Datasets, models, and metrics](#). *ACM Computing Surveys*, 55(8):1–35.
- Wojciech Kryściński, Nazneen Rajani, Divyansh Agarwal, Caiming Xiong, and Dragomir Radev. 2022. [BOOKSUM: A collection of datasets for long-form narrative summarization](#). In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 6536–6558, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.
- Faisal Ladhak, Bryan Li, Yaser Al-Onaizan, and Kathleen McKeown. 2020. [Exploring content selection in summarization of novel chapters](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5043–5054.
- Chin-Yew Lin. 2004. [ROUGE: A package for automatic evaluation of summaries](#). In *Text Summarization Branches Out*, pages 74–81.
- Dongqi Liu, Xudong Hong, Pin-Jie Lin, Ernie Chang, and Vera Demberg. 2022. [Two-stage movie script summarization: An efficient method for low-resource long document summarization](#). In *Proceedings of The Workshop on Automatic Summarization for Creative Writing*, pages 57–66.
- Nelson F. Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. 2024. [Lost in the middle: How language models use long contexts](#). *Transactions of the Association for Computational Linguistics*, 12:157–173.
- Litu Ou and Mirella Lapata. 2025. [Context-aware hierarchical merging for long document summarization](#). In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 5534–5561, Vienna, Austria. Association for Computational Linguistics.
- Charles Packer, Sarah Wooders, Kevin Lin, Vivian Fang, Shishir G. Patil, Ion Stoica, and Joseph E. Gonzalez. 2023. [MemGPT: Towards LLMs as operating systems](#). *arXiv preprint arXiv:2310.08560*.
- Bo Pang, Erik Nijkamp, Wojciech Kryscinski, Silvio Savarese, Yingbo Zhou, and Caiming Xiong. 2023. [Long document summarization with top-down and bottom-up inference](#). In *Findings of the Association for Computational Linguistics: EACL 2023*, pages 1267–1284.
- Joon Sung Park, Joseph C. O’Brien, Carrie J. Cai, Meredith Ringel Morris, Percy Liang, and Michael S. Bernstein. 2023. [Generative agents: Interactive simulators of human behavior](#). In *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology (UIST)*.
- Xiao Pu, Mingqi Gao, and Xiaojun Wan. 2023. [Summarization is \(almost\) dead](#). *arXiv preprint arXiv:2309.09558*.
- Rohit Saxena and Frank Keller. 2024a. [MovieSum: An abstractive summarization dataset for movie screenplays](#). In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 4043–4050, Bangkok, Thailand. Association for Computational Linguistics.
- Rohit Saxena and Frank Keller. 2024b. [Select and summarize: Scene saliency for movie script summarization](#). In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 3439–3455, Mexico City, Mexico. Association for Computational Linguistics.
- Rohit Saxena, Hao Tang, and Frank Keller. 2025. [End-to-end long document summarization using gradient caching](#). *Transactions of the Association for Computational Linguistics*, 13:1271–1297.
- Weixuan Wang, Minghao Wu, Barry Haddow, and Alexandra Birch. 2026. [Learning to summarize by learning to quiz: Adversarial agentic collaboration for long document summarization](#). In *The Fourteenth International Conference on Learning Representations (ICLR)*.
- Wujiang Xu, Zujie Liang, Kai Mei, Hang Gao, Juntao Tan, and Yongfeng Zhang. 2025. [A-MEM: Agentic memory for LLM agents](#). In *Advances in Neural Information Processing Systems*, volume 38.
- An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, Chujie Zheng, Dayiheng Liu, Fan Zhou, Fei Huang, Feng Hu, Hao Ge, Haoran Wei, Huan Lin, Jialong Tang, and 41 others. 2025. [Qwen3 technical report](#). *arXiv preprint arXiv:2505.09388*.

Yipeng Yu, Yichen Yuan, Chengxiao Feng, and Xu Liu. 2026. [TaoType: Predicting fine-grained typing intent for faster search](#). In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 6: Industry Track)*, pages 184–192, San Diego, California, USA. Association for Computational Linguistics.

Manzil Zaheer, Guru Guruganesh, Avinava Dubey, Joshua Ainslie, Chris Alberti, Santiago Ontanon, Philip Pham, Anirudh Ravula, Qifan Wang, Li Yang, and Amr Ahmed. 2020. [Big Bird: Transformers for longer sequences](#). In *Advances in Neural Information Processing Systems*, volume 33, pages 17283–17297.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. [BERTScore: Evaluating text generation with BERT](#). In *8th International Conference on Learning Representations (ICLR)*.

Yusen Zhang, Ruoxi Sun, Yanfei Chen, Tomas Pfister, Rui Zhang, and Sercan O. Arik. 2024. [Chain of agents: Large language models collaborating on long-context tasks](#). In *Advances in Neural Information Processing Systems*.

Chao Zhao, Faeze Brahman, Kaiqiang Song, Wenlin Yao, Dian Yu, and Snigdha Chaturvedi. 2022. [NarraSum: A large-scale dataset for abstractive narrative summarization](#). In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 182–197, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Jun Zhao, Can Zu, Hao Xu, Yi Lu, Wei He, Yiwen Ding, Tao Gui, Qi Zhang, and Xuanjing Huang. 2024. [Longagent: Scaling language models to 128k context through multi-agent collaboration](#). *arXiv preprint arXiv:2402.11550*.

Wanjun Zhong, Lianghong Guo, Qiqi Gao, He Ye, and Yanlin Wang. 2024. [MemoryBank: Enhancing large language models with long-term memory](#). *Proceedings of the AAAI Conference on Artificial Intelligence*, 38(17):19724–19731.

## A Hyperparameter and Threshold Sensitivity

We report one-factor sensitivity analyses on MovieSum for three implementation hyperparameters: chunk size, final-summary word limit multiplier, and the maximum number of entity-memory tool-call rounds. We additionally vary the entity name-matching and fact-deduplication thresholds used by the guarded write policy. To keep the appendix compact, we report BERTScore-F1 and R-Avg, the arithmetic mean of ROUGE-1/2/L F1 scores.

| Chunk | Word Limit     | Rounds | BS <sub>F1</sub> | R-Avg        |
|-------|----------------|--------|------------------|--------------|
| 5K    | $K \times 200$ | 6      | <b>61.80</b>     | 23.79        |
| 8K    | $K \times 200$ | 6      | 61.38            | <b>24.33</b> |
| 10K   | $K \times 200$ | 6      | 61.75            | 24.29        |
| 15K   | $K \times 200$ | 6      | 59.06            | 23.59        |

Table 6: Chunk size sweep. R-Avg is the arithmetic mean of ROUGE-1/2/L F1 scores.

| Chunk | Word Limit     | Rounds | BS <sub>F1</sub> | R-Avg        |
|-------|----------------|--------|------------------|--------------|
| 10K   | $K \times 100$ | 6      | 59.79            | 23.43        |
| 10K   | $K \times 150$ | 6      | <b>61.98</b>     | <b>24.88</b> |
| 10K   | $K \times 200$ | 6      | 61.16            | 24.55        |
| 10K   | $K \times 300$ | 6      | 61.58            | 24.04        |

Table 7: Word limit multiplier sweep. R-Avg is the arithmetic mean of ROUGE-1/2/L F1 scores.

| Chunk | Word Limit     | Rounds | BS <sub>F1</sub> | R-Avg        |
|-------|----------------|--------|------------------|--------------|
| 10K   | $K \times 150$ | 1      | 60.43            | 23.69        |
| 10K   | $K \times 150$ | 3      | 60.87            | 24.13        |
| 10K   | $K \times 150$ | 6      | <b>61.98</b>     | <b>24.88</b> |
| 10K   | $K \times 150$ | 10     | 61.30            | 24.32        |

Table 8: Maximum tool-call round sweep. R-Avg is the arithmetic mean of ROUGE-1/2/L F1 scores.

### A.1 Guarded-Write Thresholds

Table 9 varies one guarded-write threshold at a time on MovieSum. The default  $\tau_{\text{name}}=0.50$  and  $\tau_{\text{dedup}}=0.85$  setting gives the best observed compactness–coverage trade-off. Lowering the name threshold merges entity names more aggressively, whereas raising it fragments aliases into more records; the latter increases duplicate facts from 5.0 to 10.4. For fact deduplication, a lower threshold removes more candidate facts, while a higher threshold admits more redundancy, increasing stored facts from 44.4 to 59.2 and duplicate facts from 5.0 to 11.8. Both departures from the default reduce BERTScore-F1 and R-Avg.

| Varied                | Value       | BS <sub>F1</sub> | R-Avg        | Entities | Facts | Dup. facts |
|-----------------------|-------------|------------------|--------------|----------|-------|------------|
| $\tau_{\text{name}}$  | 0.40        | 61.10            | 24.17        | 16.2     | 41.0  | 3.8        |
|                       | <b>0.50</b> | <b>61.98</b>     | <b>24.88</b> | 23.0     | 44.4  | 5.0        |
|                       | 0.60        | 61.16            | 24.33        | 27.6     | 53.8  | 10.4       |
| $\tau_{\text{dedup}}$ | 0.75        | 61.22            | 24.14        | 22.6     | 33.4  | 4.4        |
|                       | <b>0.85</b> | <b>61.98</b>     | <b>24.88</b> | 23.0     | 44.4  | 5.0        |
|                       | 0.95        | 61.05            | 24.32        | 22.8     | 59.2  | 11.8       |

Table 9: Threshold sensitivity on MovieSum. When  $\tau_{\text{name}}$  is varied,  $\tau_{\text{dedup}}$  is fixed at 0.85; when  $\tau_{\text{dedup}}$  is varied,  $\tau_{\text{name}}$  is fixed at 0.50. Bold marks the default setting.

## B Backbone Generalization

We evaluate HM and ENTITYBB with three additional backbones on MovieSum. Within each controlled pair, the chunking, chunk-summary prompts, and final synthesis procedure are held fixed; the pair differs in whether the entity memory is used.

| Backbone    | BS <sub>F1</sub> |              | R-Avg        |              |
|-------------|------------------|--------------|--------------|--------------|
|             | HM               | EntityBB     | HM           | EntityBB     |
| Qwen3-8B    | 57.46            | 61.53        | 21.39        | 24.71        |
| Qwen3-32B   | 57.54            | 62.86        | 21.99        | 24.84        |
| DeepSeek-R1 | <b>58.58</b>     | <b>63.17</b> | <b>22.78</b> | <b>25.76</b> |
| Average     | 57.86            | 62.52        | 22.05        | 25.11        |

Table 10: Backbone generalization on MovieSum. R-Avg is the arithmetic mean of ROUGE-1/2/L F1 scores.

ENTITYBB improves both metrics for every additional backbone. Across these models, the average gains are 4.66 BERTScore-F1 and 3.06 R-Avg points, indicating that the improvement is not specific to Qwen3-4B.

## C Time Complexity Analysis

Let  $N$  be the document length in tokens,  $K$  the number of chunks,  $C$  the maximum chunk size,  $R$  the maximum number of entity-memory tool-call rounds per chunk,  $E$  the number of blackboard entities, and  $F$  the number of stored facts. Since  $K \approx N/C$ , the dominant computational cost comes from LLM calls over chunk-sized contexts rather than from deterministic blackboard operations.

| Component       | LLM Calls | Auxiliary Cost                |
|-----------------|-----------|-------------------------------|
| Partitioning    | 0         | $O(N)$                        |
| Entity writing  | $O(KR)$   | $O(E + F)$ per accepted write |
| Chunk summary   | $O(K)$    | —                             |
| Final synthesis | $O(1)$    | $O(E)$ rendering              |

Table 11: Asymptotic cost of ENTITYBB for one document.

Plain HM uses  $O(K)$  chunk-summary calls plus one final merge call. ENTITYBB adds at most  $O(KR)$  tool-call rounds before chunk summarization. In our default setting,  $R$  is a small constant, so the number of LLM calls remains linear in the number of chunks. The deterministic safeguards are lightweight: fuzzy name matching scans existing entity names, and content deduplication compares a proposed fact against the current and historical

facts for the matched entity. Thus the added non-LLM computation is small relative to generation, while the blackboard provides the final Synthesizer with compact entity states instead of another pass over the full source document.

The blackboard size also grows sublinearly in practice because duplicate writes are skipped and aliases are redirected to existing entries. This matters for long narratives: without safeguards, repeated scene-local mentions can accumulate as separate facts and make the final synthesis prompt noisy. With guarded writes, the Synthesizer receives one current state per entity plus a short history, so the memory section remains much smaller than the concatenated source chunks. The design therefore trades a bounded number of additional tool-call rounds for a more compact and structured final context.

## D Output Length Calibration

The synthesis word limit in ENTITYBB is a maximum output length rather than a target length. Table 12 reports average input lengths, average output lengths for HM and ENTITYBB, and average reference-summary lengths on each benchmark. Plain HM produces much shorter summaries, averaging fewer than 300 words across datasets. With the chunk-adaptive word limit, ENTITYBB produces outputs closer to the dataset-specific reference scale, especially on MENSA, while allowing longer outputs for MovieSum and shorter outputs for BookSum. Together with the ablation in Table 2, this suggests that the chunk-adaptive word limit provides useful verbosity control without collapsing all datasets to a fixed output length across benchmarks.

| Dataset  | Input  | HM  | EntityBB | Ref |
|----------|--------|-----|----------|-----|
| MovieSum | 26,052 | 284 | 761      | 615 |
| MENSA    | 24,508 | 269 | 654      | 670 |
| BookSum  | 18,154 | 292 | 690      | 855 |

Table 12: Average input and output lengths in words. HM, ENTITYBB, and Ref columns report summary lengths.

The length pattern is consistent with the behavior observed in the main results. Plain HM often compresses aggressively during the final merge, which can remove secondary characters, intermediate events, and causal links that are useful for reference matching. ENTITYBB does not force

a fixed target length; instead, the word limit provides an upper bound that scales with the number of chunks. As a result, outputs can remain relatively concise for BookSum while still expanding enough on MovieSum and MENSA to preserve scene-level details. These patterns are consistent with the word limit acting as a control mechanism rather than the sole source of the performance gains.

## E Prompt Templates

This appendix lists the main prompt templates used by ENTITYBB and by our diagnostic evaluations, following the compact prompt-table format used in prior agentic summarization work. Placeholders in braces are filled at runtime.

| Role          | Prompt Template                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
|---------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| System Prompt | <p>You are a fact extraction agent. Your ONLY job is to record genuinely NEW entity information.</p> <p><b>CURRENT ENTITY BLACKBOARD:</b> This is ALL information already recorded from previous chunks: {blackboard_text}.</p> <p><b>When to use reject:</b> use action='reject' if the fact is already in the blackboard, means the same thing as an existing fact, contradicts a resolved state, or the chunk has no new entity information.</p> <p><b>When to use add/update:</b> use add only for a person/topic not yet in the blackboard; use update only for a genuinely new state change to an existing entity.</p> <p><b>Rules:</b> use a stable entity name; write one concise fact; never re-record, rephrase, or re-summarize facts already in the blackboard; never write one character's event under another character's name.</p> |
| User Prompt   | <p>Text Segment: {chunk}</p> <p>Extract NEW entity facts from this text segment. Before each tool call, check the blackboard above. If the fact is already recorded or the chunk has no new entity information, use action='reject'; only genuinely new facts should use add or update.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |

Table 13: Entity-memory writing prompt used by the Executor before chunk summarization.

| Field       | Tool Schema                                                                                                                                                                                                                                                                                                                                              |
|-------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Signature   | update_blackboard(entity, fact, action) where action is one of add, update, or reject.                                                                                                                                                                                                                                                                   |
| Description | Record or update a fact about an entity in the entity memory. add registers a brand-new entity that does not exist in the blackboard. update records a genuinely new state change for an existing entity after checking current state and history. reject is used when the fact is duplicate, contradictory, or the chunk has no new entity information. |
| Arguments   | entity: a stable entity identifier.<br>fact: a single concise factual statement.<br>action: add, update, or reject.                                                                                                                                                                                                                                      |

Table 14: Entity-memory tool schema exposed to the model.

| Role          | Prompt Template                                                                                                                                                                                                                                                                                                                                                                                                               |
|---------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| System Prompt | <p>You are an expert document summarizer. Produce a detailed, informative summary of the current text segment. Capture all important facts, events, character actions, and plot developments. Include specific names, locations, numbers, and dialogue where relevant. Preserve the sequence of events and causal relationships. It is better to include too much detail than too little. Do not use Markdown formatting.</p> |
| User Prompt   | <p>Text Segment: {chunk}</p> <p>Entity blackboard (facts collected so far): {entity_blackboard}</p> <p>Produce an informative summary of this text segment. Capture all key information from this text segment. Do not include meta-commentary. Write a single coherent paragraph.</p>                                                                                                                                        |

Table 15: Chunk summary prompt. The default setting uses no explicit intermediate word limit.

| Role          | Prompt Template                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
|---------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| System Prompt | <p>You are provided with: (1) an ENTITY KNOWLEDGE BASE containing verified facts from the document and (2) summaries of different parts of the document. Your task is to merge the summaries into one final summary.</p> <p><b>Rules:</b> use the entity knowledge base to ensure factual accuracy and consistency; eliminate redundancy; maintain chronological or logical flow; do not use Markdown formatting; keep the summary fewer than {target_words} words; preserve specific names, places, numbers, and key details; keep as much specific detail as possible within the word limit.</p> |
| User Prompt   | <p>Maximum length: fewer than {target_words} words.</p> <p>[ENTITY KNOWLEDGE BASE – verified facts extracted from the document]<br/>{blackboard_section}</p> <p>Summaries to merge: {summaries}</p>                                                                                                                                                                                                                                                                                                                                                                                                |

Table 16: Blackboard-guided final synthesis prompt used by ENTITYBB.

| Role          | Prompt Template                                                                                                                                                                                                                                                                                                                                                                                                             |
|---------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| System Prompt | <p>You are provided with the summaries of different parts of a document. Your task is to merge these summaries into one summary that comprehensively captures the entire document.</p> <p><b>Rules:</b> eliminate redundancy across parts; maintain chronological or logical flow; do not use Markdown formatting; provide a coherent and concise paragraph; preserve specific names, places, numbers, and key details.</p> |
| User Prompt   | Summaries to merge: {summaries}                                                                                                                                                                                                                                                                                                                                                                                             |

Table 17: Blackboard-free merge prompt used by HM and no-BB variants when word-limit control is disabled.

| Role          | Prompt Template                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
|---------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| System Prompt | <p>You are an expert evaluator for narrative summarization quality. You will be given a SUMMARY of a narrative work and a REFERENCE SUMMARY for context. Evaluate the SUMMARY on its own quality, using the reference only to understand what the narrative is about. Do not penalize the summary for including details not present in the reference. A good summary should be detailed and comprehensive, as long as the added details are relevant and not fabricated. Do not reward verbosity by itself; a longer summary is better only when it adds correct and important information. Rate the SUMMARY on a 5-point Likert scale for four dimensions: <i>Comprehensiveness</i>, <i>Narrative Coherence</i>, <i>Character &amp; Detail Coverage</i>, and <i>Overall Quality</i>. Prefer information-rich summaries over brief generic ones when the added information is faithful and useful. Respond only with a JSON object: {"comprehensiveness": N, "coherence": N, "detail_coverage": N, "overall_quality": N}, where each N is an integer from 1 to 5.</p> |
| User Prompt   | <p>REFERENCE (for context only): {reference}</p> <p>SUMMARY TO EVALUATE: {prediction}</p> <p>Evaluate the SUMMARY. Remember: do not reward length alone; reward relevant, faithful, and specific plot points and character details.</p>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |

Table 18: LLM-as-a-Judge prompt used for the MENSA quality evaluation.

## F Human Entity-Consistency Annotation Protocol

For the human entity-consistency audit, both the HM and ENTITYBB outputs for each of the 50 examples in the full MENSA test set are anonymized before annotation, yielding 100 summaries whose system identities are hidden from the annotators. Two NLP master’s students compare each final summary with the source document and mark entity-level issues that affect factual consistency or event tracking. Each issue is assigned to exactly one category. *Contradiction* covers incompatible states for the same entity. *Duplicate event* covers cases where a previously completed event is presented again as new. *Entity*

*confusion* covers assigning an action, relation, or state to the wrong entity. *Unsupported entity fact* covers entity-specific claims that cannot be supported by the document. If the same underlying error is repeated within one summary, it is counted once; if the same summary contains distinct errors involving different entities or events, they are counted separately. Before adjudication, the two annotators achieved Cohen's  $\kappa=0.67$  across all binary category-presence decisions. The calculation covers 50 examples  $\times$  2 system outputs  $\times$  4 issue categories, yielding 400 binary decisions from their independent labels. The annotators then reconcile disagreements through discussion to produce the final issue counts. We report raw issue counts rather than asking annotators to choose the better system.