# GRASP: Graph-Reasoning Aided Survey Planning for High-Fidelity Related Work Generation

**Anonymous ACL submission**

## Abstract

Writing a literature review requires a deep understanding of the relationships among cited papers: how they build on, challenge, or offer alternative perspectives to one another. We present Graph-Reasoning Aided Survey Planning (GRASP), a framework combining LLM planning for related work generation with graph algorithms to extract key relationships among cited papers. Our two-layer graph structure consists of a Graph of Thoughts and an Argument-Counteragument Planning Network, representing the cited papers at different levels of granularity, and we apply topology-aware pruning via a Steiner tree to identify the core inter-paper relationships captured in our graph. Our citation analysis-based evaluation shows that GRASP generates RWS that closely match human-written targets in terms of the discourse roles, intents, and grouping of citations.

## 1 Introduction

Conducting a literature review is an important early step in the process of scientific research. Scientists need a deep understanding of existing work in their field, as well as the relationships among these prior works, in order to identify gaps, limitations, and challenges that their own research can address. The importance of the literature review is such that a dedicated related work section is a key component of scientific articles, and many doctoral programs require a formal literature review as part of their candidacy qualifying exams (Knopf, 2006).

The work of keeping up with an ever-growing body of existing research and drawing connections among published articles requires significant time and cognitive effort from researchers. The number of scientific publications grows exponentially (Bornmann et al., 2021), with the result that most papers about a given topic are relatively recent (Wang and Barabási, 2021). There are more publications, which are growing more specialized, full of jargon, and difficult to read (Plavén-Sigray et al., 2017), creating a need for tools to assist researchers in contextualizing these papers.

The success of Large Language Models in single-document summarization has fueled excitement about their potential to automate complex academic writing. However, proficiency in processing individual, self-contained documents often fails when the challenge shifts to summarizing a collection of interconnected documents, such as synthesizing related papers into a coherent literature review.

Writing a good literature review or related work section (RWS) takes more than simply listing cited paper summaries. It requires a deeper understanding of the intricate web of relationships among papers: how they build on, challenge, or offer alternative perspectives to one another. For current models, the volume of text from multiple papers often exceeds practical context limits, making it difficult to maintain a holistic view and accurately capture inter-paper relationships (Li and Ouyang, 2024; Liu et al., 2025). Yet such relationships are precisely what human readers value in a good RWS (Martin-Boyle et al., 2024; Li and Ouyang, 2025).

In this work, we propose Graph-Reasoning Aided Survey Planning (GRASP), a framework that combines LLM planning methods for related work generation with graph algorithms to extract key relationships among the cited papers. Our main contributions are as follows:

- We propose a two-layer graph structure, consisting of a Graph of Thoughts (Besta et al., 2024) and an Argument-Counteragument Planning Network (Hua et al., 2019), to represent the content of and relationships among cited papers at different levels of granularity.

- We introduce *consensus node* merging and topology-aware pruning to identify core relationships captured in our graph, enabling the

generation of a concise, high-quality RWS explicitly focused on inter-paper relationships.

- We conduct a *citation analysis*-based evaluation that applies citation discourse and intent labeling (Li et al., 2022; Lauscher et al., 2022), as well as citation cluster evaluation, to demonstrate that our generated RWS closely match human-written targets in the relative importance and salient aspects of cited papers.

## 2 Related Work

### 2.1 Relationship-Focused Related Work Generation

Early extractive RWS generation approaches (Hoang and Kan, 2010; Hu and Wan, 2014, inter alia) were not capable of expressing any relationships that the cited papers themselves did not explicitly discuss; early abstractive works (Xing et al., 2020; Luu et al., 2021, inter alia) focused on generating citations of individual cited papers in isolation due to the model length restrictions of the time.

More recently, Chen et al. (2021, 2022) used a graph of cited papers connected through shared keyword nodes to compute relation-aware cited paper encodings. Similarly, Wang et al. (2022) used a scientific information extraction system to build a knowledge graph of cited paper entities (e.g. tasks, metrics, etc.) and generated literature reviews based on lists of salient entities and relations. Like our proposed approach, these works made use of a graph structure to represent cited papers, but the graphs were used indirectly via document encodings or lists of keywords; unlike our approach, they did not use the graph structure to directly guide generation.

Liu et al. (2023) introduced a Causal Intervention Module (CaM) for related work generation that focused on adjusting the generation probabilities of transition words that express cited paper relationships at the beginnings of sentences (e.g. "Furthermore, ..."). However, their assumption that all relevant cited paper relationships are explicitly described using such sentence-initial transition words is unrealistic.

Finally, Li and Ouyang (2025) extracted features for a cited paper by summarizing its edges in a citation network, while Liu et al. (2025) walked a (co-)citation network by iteratively selecting cited paper sections to read and summarize, either continuing to the next section of a given paper or transitioning to a neighboring paper. These works, like most citation network-based approaches, can capture only relationships between papers that cite or are co-cited with each other, and thus suffer from sparsity and cold-start problems.

### 2.2 Planning-Guided Survey Generation

Planning-guided approaches break away from traditional single-shot generation and reframe the process as plan-then-realize. Lai et al. (2024) prompted an LLM to sequentially generate each section of a full survey article sequentially step-by-step, including section abstracts and subsection headers. Wang et al. (2024) built a pipeline of outline generation, subsection drafting, integration, and refinement, while Yan et al. (2025) likewise proposed a pipeline consisting of outline generation, subsection drafting, and refinement via a final editing pass over the concatenated subsections.

None of these works explicitly focused on the relationships among papers, relying solely on supervision from the training targets; while the target surveys should express the relationships among cited papers, they are much longer than RWS, and thus likely to contain disproportionately more specific summary content for individual papers.

## 3 Methodology

Given a citing paper $P$ (excluding the target RWS) and a set of cited papers $R = \{R_1, R_2, \ldots, R_m\}$, our goal is to generate the RWS of $P$ using $R$. Figure 1 shows an overview of our GRASP framework, and Appendix A shows our LLM prompts.

### 3.1 Graph of Thoughts Layer

We combine the semantic reasoning capabilities of Large Language Models (LLMs) with structural guidance via a Graph of Thoughts (GoT; Besta et al., 2024). While LLMs excel at extracting local information, they struggle with macro-level structure and thematic coherence. We address this problem by using traditional graph algorithms to prune the GoT, retaining only the structural backbone and most central content.

#### 3.1.1 Topic-Partitioned Graph Construction

A RWS typically covers distinct thematic clusters, so we partition the set of cited papers $\mathcal{R}$ into $k$ topic-specific subsets, $\mathcal{R} = \bigcup_{i \in [1,k]} T_i$, using an LLM-prompting clustering approach. To ensure

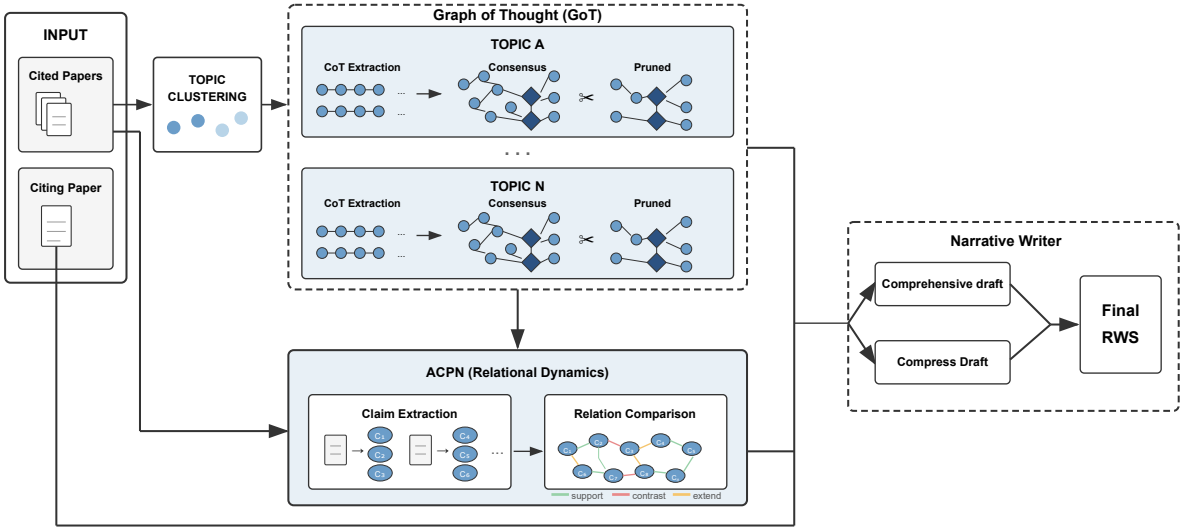


Figure 1: Our proposed framework. The cited papers are partitioned by topic, a Chain of Thoughts is extracted for each paper, and similar thoughts are merged to form a Graph of Thoughts (GoT), which is pruned to remove peripheral thoughts. Claims are extracted from each paper and, along with the paper's GoT nodes, are used to classify the relationships between pairs of cited papers, producing the Argument-Counterargument Planning Network (ACPN). Finally, the Writer module drafts, compresses, and refines the output related work section (RWS).

scalability and coherence, we construct a local reasoning graph $G_i$ for each topic $T_i$ independently.

**Sequential Node Extraction** Within each topic $T_i$, we employ Chain-of-Thought prompting (Wei et al., 2022) to extract the logical workflow of each cited paper $r_j \in T_i$. We prioritize key sections (Abstract, Introduction, and Conclusion) by limiting other sections to a single thought each, ensuring that the extracted thoughts capture core claims rather than peripheral details. These thoughts form the *sequential nodes* of our graph, linked by directed edges to preserve the intra-paper narrative flow.

**Consensus Node Formation** To explicitly capture inter-paper relationships, we use an LLM prompt to assess the semantic similarity between pairs of sequential nodes in $T_i$. When nodes from different papers exhibit significant semantic overlap, they are merged into a *consensus node* that summarizes the original sequential nodes.

Figure 2 shows example sequential and consensus nodes in an abbreviated GoT.

### 3.1.2 Topology-Aware Pruning

The raw reasoning graphs are noisy, containing fine-grained details specific to individual papers that are excessive for a high-level RWS. We prune each topic subgraph $G_i = (V_i, E_i)$ to identify and retain only a semantic backbone of nodes that are salient to multiple cited papers. We formulate this pruning step as a **Steiner Tree** problem, which seeks the minimum-weight subgraph that connects a subset of critical **terminal** nodes.

**Terminal Identification** We identify the set of terminal nodes $S \subset V_i$ that represent the main inter-paper relationships in $G_i$:

- **Consensus nodes:** These nodes aggregate information shared among multiple papers, representing pivotal points of comparison.
- **High-centrality sequential nodes:** Consensus nodes are relatively rare (averaging 1.79 per graph), so we calculate the *betweenness centrality* (Freeman, 1977) for all sequential nodes and retain the top 20% as additional terminals. We hypothesize that high-centrality nodes are topologically close to consensus nodes and serve as connective tissue within the graph, capturing shared methodological contexts common to multiple works.

**Steiner Tree Approximation** Since the Steiner tree problem is NP-hard, we implement a 2-approximation algorithm (Wu and Chao, 2004). First, we construct a metric closure graph, where the edge weight between two terminals $u, v \in S$ is their shortest path distance in the original graph $G_i$. We then compute the minimum spanning tree (MST) of this closure and map the edges of the

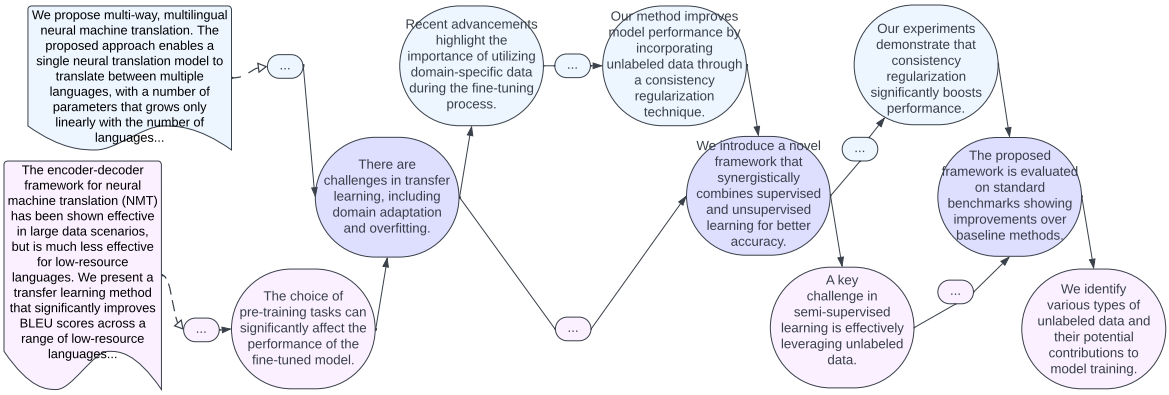


Figure 2: An excerpt from a Graph of Thoughts containing "Multi-Way, Multilingual Neural Machine Translation with a Shared Attention Mechanism" (Firat et al., 2016, shown in blue) and "Transfer Learning for Low-Resource Neural Machine Translation" (Zoph et al., 2016, shown in pink). Shared *consensus nodes* are shown in purple.

MST back to their corresponding paths $G_i$ to form the pruned subgraph $G'_i$.

The Steiner tree rigorously removes dead ends and redundant paths that do not contribute to the connectivity of the cited papers' shared ideas, as captured by the terminals. This pruning step ensures that our Writer module receives a concise, topologically streamlined blueprint that links all main ideas with a minimum of irrelevant, paper-specific details.

### 3.2 Argument-Counterargument Planning Network Layer

The GoT captures fine-grained similarities among cited papers but does not explicitly model the global relationships between papers. Two papers that do not share exact approaches in common may still present supporting evidence for a shared hypothesis, offer contrasting findings, or address difference aspects of the same problem. Recognizing these relationships is essential for generating RWS that accurately characterize the research landscape.

We use an Argument-Counterargument Planning Network (ACPN; Hua et al., 2019) to construct a paper-level argumentative relation graph over the set of cited papers $R = \{R_1, R_2, ..., R_m\}$: the ACPN produces a directed graph $H$ where each edge $(R_i, R_j)$ is labeled with a relation $r \in \{\text{support}, \text{contrast}, \text{neutral}\}$.

First, we extract the core claims from each cited paper $R_i$ using LLM prompting. Unlike the fine-grained, section-specific summaries in the GoT described in the previous section, these claims capture the overarching theses of each paper. Then, representing each paper by its claims and GoT nodes, we classify the pairwise relationships between papers.

The resulting ACPN graph provides explicit signals for generation. Edges labeled as *contrast* indicate opportunities to discuss methodological differences or conflicting results, while *support* edges identify papers that can be grouped to establish consensus or trace the development of an idea.

### 3.3 RWS Writer

The final component is an LLM-based Writer module that uses citing paper context and the structural insights of the GoT and ACPN to generate a coherent RWS. To preserve graph topological information, we serialize both graph layers into a structured JSON node-link format that can be inserted into our Writer prompt, allowing the LLM to traverse the GoT and ACPN paths. To balance the competing requirements of technical completeness and linguistic conciseness, we implement a three-stage drafting strategy within our prompt design:

1. **Comprehensive Draft:** The model first generates a verbose RWS draft explicitly incorporating every node from the graph to ensure maximal information coverage.

2. **Semantic Compression:** The model then produces a compressed version of RWS, removing redundancy and verbosity from the first draft while optimizing for information density and logical progression.

3. **Final Merging:** Finally, the model performs a reconciliation step that fuses the structural fidelity of the Comprehensive draft with the

brevity of the Compressed version. We explicitly instruct guide the Writer to remove duplicate citations and smooth out transitions, producing a more polished and fluent RWS (see Section 5.7).

We use a post-processing step to standardize the format of citation markers for evaluation and to correct hallucinated author names or years, which we describe in Appendix B.

## 4 Experimental Settings

We use GPT-4o-mini[1] as our base model because of its competitive balance of performance and cost. Appendix A shows all of our prompts.

### 4.1 Dataset

To evaluate our proposed framework, we use the OARelatedWork test set (Docekal et al., 2024), which consists of 1,878 papers from the CORE (Knoth et al., 2023) and S2ORC (Lo et al., 2020) academic writing datasets. Unlike other collections of scientific articles, the papers in OARelatedWork are curated to ensure the availability of their cited papers: for each target RWS, the full text of at least one cited paper from each citation span is included in the dataset. We use a subset of 1,350 papers from the test set; we filter out target RWS that contain fewer than three cited papers in order to focus our evaluation on capturing relationships among cited papers.

Additionally, we clean the target RWS by removing references to cited papers that are not included in OARelatedWork, as only one paper per citation span is guaranteed to be available. Since it is impossible for our approach (or the baselines) to write about papers that are not included in its input, we remove any citation marks whose papers are not available; we further delete any sentence whose citation marks have all been removed in this way[2].

### 4.2 Baselines

We compare our proposed approach with two recent graph-based RWS generation works:

**Li and Ouyang (2025)** (hereafter L&O for brevity) uses a citation network to extract relationship features for each cited paper by summarizing its incoming and outgoing edges, capturing both how other papers cite it and what it says about other papers. We use their provided LLM prompts to generate RWS for evaluation.

**Select, Read, Write (SRW; Liu et al., 2025)** uses a multi-agent framework with a Select module guided by a (co-)citation network, allowing it to transition to an adjacent paper from the one it was previously reading. As with L&O, we use the provided LLM prompts to generate RWS for evaluation.

Appendix F shows an example (cleaned) target RWS from OARelatedWork and the corresponding generated RWS from GRASP and the baselines.

### 4.3 Citation Analysis-Based Metrics

Evaluating the quality of RWS is challenging. On the one hand, traditional summarization metrics like ROUGE and BERTScore, as well as generic LLM-as-judge evaluations, demonstrate poor correlation with human judgments on scientific texts (Chen et al., 2024). On the other hand, human evaluation requires recruiting judges with a strong familiarity with the set of cited papers, which is difficult to achieve for an open-domain dataset like OARelatedWork.

We adopt an automatic, *citation analysis*-based evaluation consisting of four facets.

**Sentence Discourse Roles** Following our L&O baseline, we use the CORWA tagger (Li et al., 2022) to label each sentence in an RWS with a discourse role: *single* or *multi*-paper summary citation, *narrative* citation, *reflection* on the citing paper, or introduction/*transition* sentence. Because there may not be an exact mapping between sentences, and the total number of sentences in the generated and target RWS varies, we calculate the **ratio difference** of each discourse role, evaluating the balance of paper-specific details (single paper summaries) and inter-paper relationships (multi-paper summaries, narrative citations, and reflections) in the generated RWS.

**Citation Importance** The CORWA tagger additionally labels each citation as *dominant* if it is the main focus of its sentence, or *reference* if it is peripheral, roughly corresponding to the level of detail it receives in the RWS. We calculate accuracy and F1 score for each cited paper, treating its dominant/reference status in the target RWS as its ground truth class. This facet evaluates the gener-

[1] https://platform.openai.com/docs/models/gpt-4o-mini

[2] Such sentences are not supposed to exist based on the construction of OARelatedWork, but in practice we found that almost 18% of sentences containing citation marks were missing all of their cited papers.

| Model | ROUGE-1 | | | ROUGE-2 | | | ROUGE-L | | | BERTScore | | | BLEU | | METEOR | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | P | R | F1 | P | R | F1 | P | R | F1 | P | R | F1 | Macro | Micro | Macro | Micro |
| SRW | 0.489 | 0.672 | 0.529 | 0.341 | 0.522 | 0.388 | 0.394 | 0.577 | 0.439 | 0.297 | 0.487 | 0.389 | 31.86 | 37.59 | 0.458 | 0.410 |
| L&O | 0.517 | 0.982 | 0.664 | 0.507 | 0.975 | 0.655 | 0.510 | 0.977 | 0.658 | 0.715 | 0.926 | 0.817 | 51.12 | 54.67 | 0.680 | 0.482 |
| GRASP (unpruned) | 0.631 | 0.977 | 0.751 | 0.622 | 0.973 | 0.745 | 0.626 | 0.975 | 0.748 | 0.731 | 0.934 | 0.829 | 61.36 | 48.75 | 0.768 | 0.488 |
| GRASP (pruned) | 0.653 | 0.978 | **0.771** | 0.644 | 0.974 | **0.766** | 0.648 | 0.975 | **0.769** | 0.738 | 0.932 | **0.832** | **63.42** | **68.09** | **0.778** | **0.509** |

Table 1: Evaluation with traditional text generation metrics. We report average ROUGE, BERTScore, BLEU, and METEOR. Best results are highlighted in **bold**, with the runner-up underlined.

ated RWS on the expressing relative importance of each cited paper.

**Citation Intent** We use the MultiCite tagger (Lauscher et al., 2022) to label the function of each citation as giving *background* information, *motivating* the citing paper, *using* or *extending* a given approach, comparing *similarities* or *differences* among papers, or suggesting *future work*. We again calculate accuracy and F1 score for each cited paper, treating its intent in the target RWS as its ground truth class. This facet evaluates the generated RWS on the capturing most salient aspect of each cited paper.

**Citation Co-Occurrence** Our SRW baseline (Liu et al., 2025) evaluates relationships among cited papers via a graph where edges between papers represent citation within the same sentence. We argue that this dependence on intra-sentence co-occurrence is too strict; Liu et al. find only two to three edges per RWS on average. Further, their metrics of edge count, node degree, and clustering coefficient consider only the *frequency* of citation co-occurrence, but not *which* papers are cited together.

We instead evaluate citation co-occurrence at the paragraph level, with edges between papers cited within the same paragraph. We use **edge-connected Jaccard similarity** to measure grouping fidelity by comparing each cited paper's neighbors in a generated RWS paragraph to those in its target RWS paragraph. Additionally, we evaluate the relative *ordering* of citations within a generated RWS (e.g. how foundational works should be discussed before papers that extend their approaches) by enumerating its citation marks in order of occurrence and calculating **Kendall's** $\tau$ with the target RWS ordering.

## 5 Results and Discussion

### 5.1 Text Generation Metrics

Table 1 evaluates the performance of our proposed GRASP framework using traditional text generation metrics: ROUGE-1, -2, and -L; BERTScore; BLEU; and METEOR. GRASP outperforms both baselines across all metrics.

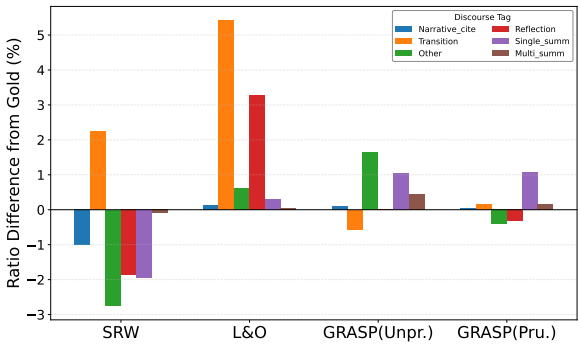


Figure 3: Ratio difference of discourse roles by method; our proposed approaches match the target discourse distribution more closely (shorter bars) than the baselines.

In terms of lexical overlap-based metrics, our approach captures significantly more relevant $n$-grams and maintains better structural coherence than the baselines. While L&O achieves high ROUGE recall, their corresponding precision is notably lower, suggesting their approach may be over-generating content to maximize coverage at the expense of conciseness (this interpretation is further supported by Appendix Table 5). In contrast, GRASP maintains comparable recall while delivering substantially higher precision, resulting in a higher-quality RWS. Beyond lexical overlap, GRASP also demonstrates superior semantic alignment with the target RWS, as evidenced by its lead in BERTScore.

### 5.2 Sentence Discourse Roles

Fig 3 compares the distribution of sentence discourse roles[3] in generated RWS. Both baselines exhibit significant deviations from the target distribution. L&O disproportionately over-generates *introduction/transition* and *reflection* sentences (shown in orange and red, respectively), suggesting a ten-

[3]The *other* role (shown in green in Figure 3) is a catch-all category for unexpected sentence formats, usually subsection or paragraph headers.

| Model | Background | | | Differences | | | Extends | | | Future Work | | | Motivation | | | Similarities | | | Uses | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | P | R | F1 | P | R | F1 | P | R | F1 | P | R | F1 | P | R | F1 | P | R | F1 | P | R | F1 |
| SRW | 0.163 | 0.518 | 0.248 | 0.313 | 0.464 | 0.374 | 0.125 | 0.700 | 0.212 | 0.000 | 0.000 | 0.000 | 0.091 | 0.581 | 0.158 | 0.027 | 0.211 | 0.048 | 0.489 | 0.319 | 0.387 |
| L&O | 0.670 | 0.965 | 0.791 | 0.590 | 0.960 | 0.731 | 0.340 | 1.000 | 0.508 | 0.429 | 1.000 | 0.600 | 0.274 | 0.983 | 0.429 | 0.492 | 0.955 | 0.650 | 0.827 | 0.950 | 0.884 |
| GRASP (unpruned) | 0.700 | 0.965 | 0.811 | 0.880 | 0.960 | **0.918** | 0.940 | 1.000 | **0.971** | 0.750 | 1.000 | 0.857 | 0.635 | 0.983 | 0.772 | 0.977 | 0.955 | **0.966** | 0.985 | 0.950 | 0.967 |
| GRASP (pruned) | 0.728 | 0.964 | **0.829** | 0.844 | 0.960 | 0.898 | 0.862 | 0.980 | 0.917 | 1.000 | 1.000 | **1.000** | 0.684 | 0.983 | **0.806** | 0.955 | 0.955 | 0.955 | 0.991 | 0.950 | **0.970** |

Table 2: Citation intent fidelity. We report average Precision, Recall, and F1-score, measured against the intent of each cited paper in the target RWS. Best results are highlighted in **bold**, with the runner-up underlined.

| Model | Dominant | | | Reference | | |
|---|---|---|---|---|---|---|
| | P | R | F1 | P | R | F1 |
| SRW | 0.263 | 0.263 | 0.263 | 0.389 | 0.523 | 0.446 |
| L&O | 0.710 | 0.928 | 0.804 | 0.944 | 0.973 | **0.958** |
| GRASP (unpruned) | 0.675 | 0.915 | 0.777 | 0.918 | 0.983 | 0.949 |
| GRASP (pruned) | 0.916 | 0.928 | **0.922** | 0.917 | 0.983 | 0.949 |

Table 3: Citation importance fidelity. We report average Precision, Recall, and F1-score, measured against the *Dominant*/*Reference* status of each cited paper in the target RWS.

| Model | Edge Jaccard | Kendall's $\tau$ |
|---|---|---|
| SRW | 0.569 | 0.628 |
| L&O | 0.791 | **0.887** |
| GRASP (unpruned) | 0.835 | 0.885 |
| GRASP (pruned) | **0.847** | 0.886 |

Table 4: Evaluation of citation co-occurrence. We report edge-connected Jaccard similarity (citation grouping) and Kendall's Tau (citation ordering).

dency make generalized statements and focus on the citing paper, rather than providing enough information about the cited papers (see also Appendix Table 5). SRW likewise over-generates *introduction/transition* while under-generating other sentence types. In contrast, our Steiner-pruned GRASP achieves a distribution that aligns closely with the target, capturing a natural balance of summarization, transition, and synthesis.

### 5.3 Citation Importance

Table 3 shows the citation importance fidelity of a cited paper in a generated RWS to its ground truth *dominant*/*reference* status in the target RWS. Pruned GRASP achieves the strongest performance on *Dominant*-type citations, while only slightly underperforming L&O on *Reference*-type citations. Our approach successfully identifies the most important cited papers that are salient to the narrative backbone of the RWS and thus deserve the longer, more detailed *Dominant* citations.

Appendix Table 5 further shows that, in terms of total length, SRW is extremely concise and over-generates the shorter, more generalized *Reference*-type citations. L&O and unpruned GRASP generate over-length RWS containing disproportionately many *Dominant*-type citations, indicating a tendency to generate individual paper descriptions, rather than focusing on inter-paper relationships. Pruned GRASP reduces verbosity by filtering topologically peripheral GoT nodes, thereby reducing the sentence count and producing the *dominant*/*reference* ratio closest to that of the target RWS.

### 5.4 Citation Intent

Table 2 shows the citation intent fidelity of a cited paper in a generated RWS to its ground truth intent in the target. GRASP demonstrates strong performance in capturing relationships in the development of cited papers, especially *Differences*, *Extends*, and *Similarities*. This result shows that the explicit inter-paper argumentative relation modeling in our ACPN module successfully guides the Writer module to accurately describe how each paper builds upon or diverges from prior methodologies.

GRASP also captures the broader research context. We achieve a perfect F1 score on the relatively rare *Future Work* category and outperform the baselines on *Motivation*, suggesting that our consensus-based pruning strategy effectively emphasizes high-level research objectives and speculative directions discussed in the cited papers. In contrast, the baselines achieve their highest scores in the *Background* and *Uses* categories, focusing more on generalized problem statements and descriptions of specific approaches.

### 5.5 Citation Co-Occurrence

Table 4 evaluates the grouping and ordering of citations in generated RWS, as compared to the target. GRASP significantly outperforms the baselines in edge-connected Jaccard similarity, measuring the the co-occurrence of cited papers at the paragraph level. This grouping fidelity is due to the explicit topic partitioning step upstream of our Graph of Thoughts construction. By partitioning the set of

cited papers into topic clusters prior to subgraph generation, we enforce an overall graph topology where topically related papers form densely connected subgraphs. This structural prior acts as a blueprint for the Writer, guiding it to generate paragraphs that discuss groups of closely related papers, thereby minimizing the scattering of related papers noted by Li and Ouyang (2025).

GRASP also achieves a high Kendall's $\tau$, measuring the correlation of the generated citation ordering with that of the target RWS; our scores are comparable to L&O, which used a simple — and apparently very strong — chronological ordering heuristic. This result shows that GRASP successfully balances local citation grouping with global ordering, producing an RWS that is both information-rich and coherent.

### 5.6 Effect of GoT Pruning

A key contribution of our work is the use of Steiner Tree-based pruning to filter irrelevant nodes from the cited paper GoT. Our experimental results validate this design choice: pruned GRASP outperforms the unpruned variant across all text generation metrics (Table 1), discourse roles (Figure 3), and citation importance/co-occurrence metrics (Tables 3-5). As discussed in Section 5.3, the citation importance evaluation highlights the success of our pruning approach at removing peripheral nodes, effectively reducing noise and verbosity without discarding critical information and allowing the Writer module to focus on the semantic backbone of the cited paper set.

### 5.7 Ablation Studies

We evaluate the contribution of each of our two graph layers — the GoT capturing topical grouping and cited paper content and the ACPN capturing argumentative relations between papers — as well as the effect of our Writer module's drafting stages. Due to space limitations, the quantitative results are found in Appendices D and E, and we briefly summarize them here.

**GoT versus ACPN** As shown in Appendix D, the GoT-only variant maintains strong recall performance on text generation metrics and citation intent fidelity, but suffers from lowered precision, while the ACPN-only variant performs significantly worse in both precision and recall. These findings reflect the design of our two graph layers. The GoT encodes the cited paper content, so the GoT-only model is able to generate most of the target RWS content. The ACPN provides higher-level guidance on inter-paper argumentative relationships, aiding the Writer module in focusing on the most salient aspects of the cited papers, but the ACPN alone does not capture enough cited paper content to generate a good RWS. In terms of sentence discourse roles, both ablated variants over-generate single cited paper summaries, suggesting that both graph layers are needed to focus the Writer on inter-paper relationships.

**Compressed versus Final Drafts** As shown in Appendix E, the Compressed and Final drafts produced by our Writer module achieve very close performance across metrics. The Final draft trades a slight dip in lexical metrics for improved discourse role and *dominant/reference* ratios, citation intent fidelity, and citation grouping/ordering. Qualitatively, we find that the Final draft is more expressive and fluent; the Compressed draft relies on ambiguous or unexpected word choice in order to reduce the overall word count, often falling into noticeably AI-like phrasing (see Appendix Figures 5 and 6 to compare example Compressed and Final drafts).

## 6 Conclusion

We have proposed GRASP, a structure-aware framework for automated RWS generation. By combining a cited paper Graph-of-Thoughts for topical clustering and detailed relationship extraction with an Argument-Counterargument Planning Network for high-level inter-paper relationships and transition planning, our approach effectively models both cited paper content and narrative flow. Our Steiner Tree pruning step enhances the conciseness and relevance of generated RWS by explicitly focusing on consensus and high-centrality nodes. To validate our framework, we introduced several citation analysis-based metrics, including citation intent fidelity and Edge Jaccard for citation grouping. Our experiments demonstrate that GRASP significantly outperforms existing baselines, producing RWS that closely follow the structure and citation distribution of human-written targets.

## Limitations

Our current framework achieves strong performance but still has several limitations. First, we assume the set of cited papers' full texts are given

as input. While this assumption is used in most existing related work generation approaches, it may not reflect a realistic use case, as a user would need to manually collect and pre-process the cited paper PDFs. Second, constructing the GoT and ACPN for a large number of cited papers is expensive and time-consuming; since the GoT is an input to the ACPN construction module, these two graphs cannot be built concurrently. Third, we partition the cited papers into disjoint topic clusters, which may be overly restrictive. A cited paper can be relevant to more than one topic, and restricting its discussion to only one topic could weaken the overall informativeness of the generated RWS. Finally, our framework is not intended to replace human RWS writing, but only as an assistive tool. We do not check for plagiarism, and while we do check for hallucinated citation marks (i.e. non-existent author/year combinations), LLM-generated RWS can still contain factually incorrect statements.

## References


Maciej Besta, Nils Blach, Ales Kubicek, Robert Gerstenberger, Michal Podstawski, Lukas Gianinazzi, Joanna Gajda, Tomasz Lehmann, Hubert Niewiadomski, Piotr Nyczyk, and 1 others. 2024. Graph of thoughts: Solving elaborate problems with large language models. In *Proceedings of the AAAI conference on artificial intelligence*, volume 38, pages 17682–17690.

Lutz Bornmann, Robin Haunschild, and Rüdiger Mutz. 2021. Growth rates of modern science: a latent piecewise growth curve approach to model publication numbers from established and new literature databases. *Humanities and Social Sciences Communications*, 8(1):1–15.

Xiuying Chen, Hind Alamro, Mingzhe Li, Shen Gao, Rui Yan, Xin Gao, and Xiangliang Zhang. 2022. Target-aware abstractive related work generation with contrastive learning. In *Proceedings of the 45th international ACM SIGIR conference on research and development in information retrieval*, pages 373–383.

Xiuying Chen, Hind Alamro, Mingzhe Li, Shen Gao, Xiangliang Zhang, Dongyan Zhao, and Rui Yan. 2021. Capturing relations between scientific papers: An abstractive model for related work section generation. Association for Computational Linguistics (ACL).

Xiuying Chen, Tairan Wang, Qingqing Zhu, Taicheng Guo, Shen Gao, Zhiyong Lu, Xin Gao, and Xiangliang Zhang. 2024. Rethinking scientific summarization evaluation: grounding explainable metrics on facet-aware benchmark. *arXiv preprint arXiv:2402.14359*.

Martin Docekal, Martin Fajcik, and Pavel Smrz. 2024. Oarelatedwork: A large-scale dataset of related work sections with full-texts from open access sources. *Preprint*, arXiv:2405.01930.

Orhan Firat, Kyunghyun Cho, and Yoshua Bengio. 2016. Multi-way, multilingual neural machine translation with a shared attention mechanism. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 866–875, San Diego, California. Association for Computational Linguistics.

Linton C Freeman. 1977. A set of measures of centrality based on betweenness. *Sociometry*, pages 35–41.

Cong Duy Vu Hoang and Min-Yen Kan. 2010. Towards automated related work summarization. In *Coling 2010: Posters*, pages 427–435, Beijing, China. Coling 2010 Organizing Committee.

Yue Hu and Xiaojun Wan. 2014. Automatic generation of related work sections in scientific papers: An optimization approach. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1624–1633, Doha, Qatar. Association for Computational Linguistics.

Xinyu Hua, Zhe Hu, and Lu Wang. 2019. Argument generation with retrieval, planning, and realization. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2661–2672, Florence, Italy. Association for Computational Linguistics.

Jeffrey W Knopf. 2006. Doing a literature review. *PS: Political Science & Politics*, 39(1):127–132.

Petr Knoth, Drahomira Herrmannova, Matteo Cancellieri, Lucas Anastasiou, Nancy Pontika, Samuel Pearce, Bikash Gyawali, and David Pride. 2023. Core: A global aggregation service for open access papers. *Scientific Data*, 10.

Yuxuan Lai, Yupeng Wu, Yidan Wang, Wenpeng Hu, and Chen Zheng. 2024. Instruct large language models to generate scientific literature survey step by step. In *CCF International Conference on Natural Language Processing and Chinese Computing*, pages 484–496. Springer.

Anne Lauscher, Brandon Ko, Bailey Kuehl, Sophie Johnson, Arman Cohan, David Jurgens, and Kyle Lo. 2022. MultiCite: Modeling realistic citations requires moving beyond the single-sentence single-label setting. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1875–1889, Seattle, United States. Association for Computational Linguistics.

Xiangci Li, Biswadip Mandal, and Jessica Ouyang. 2022. CORWA: A citation-oriented related work annotation dataset. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human*

*Language Technologies*, pages 5426–5440, Seattle, United States. Association for Computational Linguistics.

Xiangci Li and Jessica Ouyang. 2024. Related work and citation text generation: A survey. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 13846–13864, Miami, Florida, USA. Association for Computational Linguistics.

Xiangci Li and Jessica Ouyang. 2025. Explaining relationships among research papers. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 1080–1105, Abu Dhabi, UAE. Association for Computational Linguistics.

Jiachang Liu, Qi Zhang, Chongyang Shi, Usman Naseem, Shoujin Wang, Liang Hu, and Ivor Tsang. 2023. Causal intervention for abstractive related work generation. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 2148–2159, Singapore. Association for Computational Linguistics.

Xiaochuan Liu, Ruihua Song, Xiting Wang, and Xu Chen. 2025. Select, read, and write: A multi-agent framework of full-text-based related work generation. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 7009–7028, Vienna, Austria. Association for Computational Linguistics.

Kyle Lo, Lucy Lu Wang, Mark Neumann, Rodney Kinney, and Daniel Weld. 2020. S2ORC: The semantic scholar open research corpus. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4969–4983, Online. Association for Computational Linguistics.

Kelvin Luu, Xinyi Wu, Rik Koncel-Kedziorski, Kyle Lo, Isabel Cachola, and Noah A. Smith. 2021. Explaining relationships between scientific documents. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 2130–2144, Online. Association for Computational Linguistics.

Anna Martin-Boyle, Aahan Tyagi, Marti A Hearst, and Dongyeop Kang. 2024. Shallow synthesis of knowledge in gpt-generated texts: A case study in automatic related work composition. *arXiv preprint arXiv:2402.12255*.

Pontus Plavén-Sigray, Granville James Matheson, Björn Christian Schiffler, and William Hedley Thompson. 2017. The readability of scientific texts is decreasing over time. *Elife*, 6:e27725.

Devendra Sachan and Graham Neubig. 2018. Parameter sharing methods for multilingual self-attentional translation models. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 261–271, Brussels, Belgium. Association for Computational Linguistics.

Dashun Wang and Albert-László Barabási. 2021. *The science of science*. Cambridge University Press.

Pancheng Wang, Shasha Li, Kunyuan Pang, Liangliang He, Dong Li, Jintao Tang, and Ting Wang. 2022. Multi-document scientific summarization from a knowledge graph-centric view. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 6222–6233, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

Yidong Wang, Qi Guo, Wenjin Yao, Hongbo Zhang, Xin Zhang, Zhen Wu, Meishan Zhang, Xinyu Dai, Qingsong Wen, Wei Ye, and 1 others. 2024. Autosurvey: Large language models can automatically write surveys. *Advances in neural information processing systems*, 37:115119–115145.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, and 1 others. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837.

Bang Ye Wu and Kun-Mao Chao. 2004. *Spanning trees and optimization problems*. Chapman and Hall/CRC.

Xinyu Xing, Xiaosheng Fan, and Xiaojun Wan. 2020. Automatic generation of citation texts in scholarly papers: A pilot study. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6181–6190, Online. Association for Computational Linguistics.

Xiangchao Yan, Shiyang Feng, Jiakang Yuan, Renqiu Xia, Bin Wang, Lei Bai, and Bo Zhang. 2025. SURVEYFORGE : On the outline heuristics, memory-driven generation, and multi-dimensional evaluation for automated survey writing. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 12444–12465, Vienna, Austria. Association for Computational Linguistics.

Barret Zoph, Deniz Yuret, Jonathan May, and Kevin Knight. 2016. Transfer learning for low-resource neural machine translation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1568–1575, Austin, Texas. Association for Computational Linguistics.

# A LLM Prompts

## A.1 Prompt for Graph of Thoughts Construction

**System Prompt: Graph-of-Thought Construction**

You are an expert scientific assistant.
### **Task** You are constructing a Graph-of-Thought (GoT) for a given topic. Each

paper provides a chain-of-thought (CoT) – an ordered list of steps. Build a Directed Acyclic Graph that follows these rules:

1. There are two types of nodes: original and consensus.
   - 1.1) Each Original node is one step in each CoT.
   - 1.2) Each Consensus node is an aggregation of two or more similar original nodes.
2. There are two types of edges: sequential and consensus.
   - 2.1) A sequential edge is an edge connecting two original nodes.
   - 2.2) A consensus edge is an edge connecting an original node to a consensus node or a consensus node to another consensus node.
3. You should name all nodes starting from n0. The order of nodes in each CoT is determined.
4. The content of a consensus node should be a summary of the content of the original nodes that comprise it. **IMPORTANT: The consensus node text MUST start with metadata from ALL contributing papers, formatted as: |||paper_id:X; author:A; year:Y||| |||paper_id:Z; author:B; year:W||| followed by the summary.** This preserves which papers contributed to the consensus.
5. Once several original nodes are aggregated into a consensus node, the edges connecting the original nodes will be converted to connect the consensus node, regardless of whether they are outgoing or incoming edges, and the aggregated original nodes will be removed from the graph. The 'paper_id' of the edge will not change; it stays the same as it is linked to the original nodes.
6. The original nodes that make up the consensus nodes must come from different CoTs and cannot be different nodes under the same CoT.
7. In each CoT, the first node will contain some metadata of the paper between "|||" markers. The metadata contains the paperid, author, and year. It helps you to understand where each node comes from because the graph is an acyclic directed graph and you can trace back to every node's predecessor to figure it out. Just keep it there and ignore it when doing content analysis.
8. **IMPORTANT: Actively look for opportunities to create consensus nodes!** You should be AGGRESSIVE in merging similar steps across papers. Consider the following as candidates for consensus:
   - Steps that discuss similar methodology or technique (even if worded differently)
   - Steps that address a similar problem or challenge
   - Steps that describe similar experimental setups or datasets
   - Steps that present similar conclusions or findings
   - Steps that reference similar background concepts

   Do NOT require exact wording match - semantic similarity is sufficient. If two steps from different papers are talking about roughly the same idea, merge them into a consensus node. You should lower your threshold to merge nodes when there is not even one in each topic. If still can't find one, then you move on.
9. In each original node there might be some section names indicating where does this node's idea come from, when merging, you can do whatever you want to keep or drop or change some form of the information to finally better maintain necessary information.
10. Only return JSON following the given schema. No prose, no markdown.
11. **Creating consensus nodes is a KEY objective.** A good GoT should have

at least a few consensus nodes that capture shared ideas across papers. If you find no similar steps at all, double-check - papers on the same topic usually share some common ground (e.g., problem definition, evaluation metrics, baseline methods).

You can refer to the following example to check the graph structure format:

**Example (two papers):**
Paper A: [A1, A2, A3] → nodes A_step1, A_step2, A_step3
Paper B: [B1, B2, B3] → nodes B_step1, B_step2, B_step3
Suppose A_step2 ≈ B_step3. Create n_con1 and replace A_step2, B_step3 with n_con1.
Edges should be:
A_s1 → n_con1 → A_s3 and B_s1 → B_step2 → n_con1
(The above example's name of node is not what really in the graph, you should follow the requirement to name the nodes.)

**### Output format**

```
{
  "topic": "<topic string>",
  "nodes": [
    {"id": "n0", "text": "step
    content"},
    {"id": "n1", "text": "..."}
  ],
  "edges": [
    {"source": "n0", "target": "n1
    ", "paper_id": "P001"},
    {"source": "n1", "target": "n2
    ", "paper_id": "P001"}
  ]
}
```

## A.2 Prompts for ACPN Construction

### A.2.1 Claim Extraction Prompt

**System Prompt: Claim Extraction**

You are a helpful research assistant designed to output JSON.
**### Task** Your task is to carefully read the provided text from a scientific paper and extract the main claims or hypotheses made by the authors. The paper is given in the format of `{section name: section texts}`; you should think of the paper as a whole, not section by section.
A **claim** is a statement of finding, discovery, or a core argument the paper is making. Extract between **3 to 7** of the most important claims.
**### Constraints**

- Paper text begins with metadata between "|||" markers - **ignore this metadata during analysis**.
- The paper sections are prioritized by importance (abstract, introduction, conclusion, results, methods) to ensure key claims are captured.
- Some papers may have `[...TRUNCATED...]` markers indicating omitted content to fit context limits.
- Your output MUST be a valid JSON object that strictly adheres to the schema provided.
- Do not add any other explanatory text, introductions, or markdown formatting like "'`json`.

**### Output format**

```
{
  "claims": [
    "Claim 1 text...",
    "Claim 2 text..."
  ]
}
```

### A.2.2 Relation Prediction Prompt

**System Prompt: Relation Prediction**

You are an expert reasoning engine. Your task is to decide the **direction-agnostic relation** between two scientific papers.
**### Possible Labels**

- **support** – the two papers make compatible or mutually reinforcing claims.
- **contrast** – one paper challenges, contradicts, or weakens the other's claims.
- **neutral** – neither clear support nor clear contrast.

The optional field "additional_context" contains up to 1000 characters of Graph-of-Thought steps for each paper; treat it as high-level reasoning evidence. There might be metadata between "|||" markers - ignore this metadata during analysis.
Return **only** the strict JSON object shown in the output format. Do not add markdown, comments, or extra keys.
**### Output format**

```
{
  "relation": "support"
}
```

### A.3 Prompt for RWS Generation

**System Prompt: Related Work Generator**

You are an academic writing assistant. Your task is to write a well-structured Related Work section for the given TOPIC.
You have access to three kinds of information:

- **Graph-of-Thought (GoT)** — a reasoning graph that organizes cited works into several topic-specific subgraphs. Identify the distinct topics in GoT. For each topic, determine which reference papers belong to it. Use GoT's logical structure to order them (precursor → complementary → contrasting).
- **Argument-Counter-argument Planning Network (ACPN)** — a directed graph connecting all references, where edges labeled *support*, *contrast*, or *neutral* indicate argument relations across papers. Integrate this information to form comparative statements.
- **Source text excerpts** (abstract, introduction, conclusion, etc.) from the current paper. Use them to infer what the current work does and to highlight its novel contribution in contrast to prior work.

**### Citation style rules (CRITICAL - you MUST follow these exactly):**

The `'citations information'` field contains ALL valid references. Each reference has:

- `'k'`: numeric citation key (integer like 1, 2, 3...)
- `'authors'`: list of author names (e.g., ['Trevor Cohn', 'Chris Dyer', ...])
- `'year'`: publication year (e.g., 2016)
- `'title'`: paper title

Determine citation style by checking the source text:

- If **NUMERIC style** → use [k] format, e.g., [1], [2, 3], [1-4]. The 'k' values come from `'citations information'`.
- If **AUTHOR-YEAR style** → use (FirstAuthorSurname et al., year) or (FirstAuthorSurname, year) for single-author papers. Extract the FIRST author's SURNAME from the `'authors'` list in `'citations information'`.
  *Example:* if authors=['Trevor Cohn', 'Chris Dyer'] and year=2016, cite as (Cohn et al., 2016).
- For multiple citations: [1, 2] or (Cohn et al., 2016; Tang et al., 2016).

**CRITICAL: You MUST use ONLY the author names and years that appear in `'citations information'`. If you cannot find a reference in `'citations information'`, do not cite it.**

**### Paragraph structure rules:**

- Let the GoT topology guide your paragraph structure naturally.
- If GoT contains multiple DISCONNECTED subgraphs (separate topic clusters), write one paragraph per subgraph.
- If GoT is a single CONNECTED component, write as a cohesive single paragraph or use natural topic transitions.

- Do NOT artificially split or merge paragraphs. Let the logical structure of the content determine breaks.

**### Writing requirements:**

- Write in formal academic tone and follow the structure conventions found in the source paper.
- Integrate GoT and ACPN insights coherently rather than list them.
- Highlight how the current paper differs from and improves upon prior work.
- Avoid redundancy and ensure the text reads naturally.
- Output length should be proportional to the graph content: write approximately 1-2 sentences per unique thought node in GoT. Do not pad or elaborate beyond what the graph provides. If the graph contains fewer nodes, produce correspondingly shorter output.

**### Three-step generation plan:**

1. **Step 1:** Generate a comprehensive related work section using all available information.
2. **Step 2:** Generate a more concise version that summarizes and compresses the first.
3. **Step 3:** Compare the two versions and synthesize a final version that is brief yet complete, ensuring that no important content is lost.
    - You should balance the sentence distribution to make it close to human-written text.
    - Not only summarize prior work, but highlight the current paper's differences and transit smoothly between topics.
    - In the final version, remove duplicated citation markers and keep only the first occurrence, as long as doing so does not reduce readability or create ambiguity about which paper is being referenced.

In each step, respect its description in the provided schema as a behavioral instruction.

## B Citation Post-Processing

Ensuring bibliographic accuracy is paramount. Our approach decouples citation *placement* from citation *formatting*. During generation, our Writer LLM is instructed to strictly adhere to the citation style found in the source text context. If the source utilizes numeric markers (e.g. "[1]" ), the model generates numeric citations corresponding to the provided cited paper indices; if the source employs an author-year format, the model attempts to generate citations in that style directly. This adaptability minimizes generation friction by allowing the model to mimic the citation placement used in the rest of the target citing paper.

For each of these two citation formats, we apply a deterministic post-processing step:

**Numeric-to-Text Conversion** If the generated text uses the numeric format, we map each numeric index back to the set of cited papers and substitute the numeric marker with the corresponding author-year string (e.g., "(Smith et al., 2025)"). We also apply this post-processing step to the target RWS and all comparison systems' outputs to standardize the citation markers for evaluation.

**Citation Verification** If the generated text uses the author-year format, we perform a validation check against the set of cited papers. We verify that the author-year pair exists in the input bibliography, flagging and correcting any hallucinations where the model may have invented non-existent dates or authors.

## C Generation and Citation Importance Statistics

Table 5 shows that, in terms of total RWS length, the SRW baseline is extremely concise, generating only 87% of the target sentence count. In contrast, both GRASP and especially L&O are more verbose. The excessive length and disproportionately high number of Dominant citation spans generated by both L&O and unpruned GRASP indicate a tendency to over-generate descriptions of individual

| Metric | Target | SRW | L&O | GRASP (unpr.) | GRASP (pru.) |
|---|---|---|---|---|---|
| # Sentences | 24,753 | 21,480 | 40,636 | 39,788 | 36,237 |
| # Dominant | 255 | 159 | 503 | 880 | 423 |
| # Sent. w/ Dom. | 254 | 1058 | 503 | 849 | 423 |
| # Reference | 594 | 459 | 884 | 940 | 796 |
| # Sent. w/ Ref. | 479 | 1356 | 742 | 757 | 643 |
| # Ref / # Dom | 2.33 | 2.87 | 1.76 | 1.07 | 1.88 |

Table 5: Total RWS sentence and citation count statistics on the OARelatedWork test set.

cited papers, rather than focusing on inter-paper relationships. Pruned GRASP effectively reduces verbosity by filtering topologically peripheral GoT nodes, thereby reducing the sentence count and increasing the ratio of *Reference* to *Dominant* citations to 1.88, the closest to the target ratio.

By comparing the number of *Dominant*/*Reference* citations to the number of sentences containing such citations, we can see that the target, L&O, and GRASP all follow the observations of Li et al. (2022): most *Dominant* citations occupy an entire sentence, while most *Reference* citations cover less than a single sentence. Interestingly, we also see that SRW contains a very high ratio of citation sentences to citations; despite generating fewer total citations than the target, each citation spans multiple sentences, suggesting that cited papers are discussed with an unnecessary amounts of detail.

## D Ablation Study Results

These tables compare the performance of the pruned GRASP full model against ablated variants (the *GoT only* variant is also pruned).

In Table 6, we see that the GoT-only variant maintains comparable performance on ROUGE and BERTScore recall but suffers from lowered precision, while the ACPN-only variant has significantly lower precision and recall. The GoT encodes the cited paper content, so the GoT-only model is able to generate most of the target RWS content. In contrast, the ACPN provides higher-level guidance on inter-paper argumentative relationships, guiding the full model to focus on the most salient aspects of the cited papers, but the ACPN alone does not capture enough cited paper content to generate a good RWS.

In Table 7, we see that both variants over-generate single cited paper summaries, suggesting that both graph layers are needed to focus the Writer module on inter-paper relationships.

In Table 8, we see that the ACPN-only model performs slightly better on *Reference*-type citations, while GoT-only performs slightly better on *Dominant* citations, again showing that the GoT provides most of the cited paper content, while the ACPN provides higher-level inter-paper relationships.

In Table 9, we see that the GoT-only variant performs comparably to the full model (or even slightly better, in the cast of *Motivation* citations), while the ACPN-only variant is much worse. Again we conclude that, since the ACPN captures argumentative relationships between cited papers, but very little paper content, the quality of individual citations generated by the ACPN-only variant is relatively poor.

In Table 10, we see that both variants perform similarly in terms of citation grouping and ordering, with GoT-only pulling slightly ahead. Both graphs contain grouping and ordering information via edges between cited papers/thoughts and complement each other to produce the strong performance of the full model.

| Metric | Full Model | ACPN only | GoT only |
|---|---|---|---|
| ROUGE-1 (P) | **0.653** | 0.441 | 0.465 |
| ROUGE-1 (R) | 0.978 | 0.724 | **0.983** |
| ROUGE-1 (F1) | **0.771** | 0.515 | 0.619 |
| ROUGE-2 (P) | 0.644 | 0.298 | 0.456 |
| ROUGE-2 (R) | 0.974 | 0.549 | **0.975** |
| ROUGE-2 (F1) | **0.766** | 0.365 | 0.610 |
| ROUGE-L (P) | **0.648** | 0.338 | 0.459 |
| ROUGE-L (R) | 0.975 | 0.600 | **0.977** |
| ROUGE-L (F1) | **0.769** | 0.407 | 0.613 |
| BERTScore (P) | **0.738** | 0.356 | 0.702 |
| BERTScore (R) | **0.932** | 0.495 | 0.919 |
| BERTScore (F1) | **0.832** | 0.423 | 0.806 |
| BLEU (Macro) | **63.42** | 29.89 | 46.61 |
| BLEU (Micro) | **68.09** | 33.91 | 50.21 |
| METEOR (Macro) | **0.778** | 0.468 | 0.670 |
| METEOR (Micro) | **0.509** | 0.427 | 0.475 |

Table 6: Ablation study results using traditional text generation metrics.

| Metric | Full Model | ACPN only | GoT only |
|---|---|---|---|
| Single_summ | **0.011** | 0.064 | 0.050 |
| Multi_summ | **0.002** | 0.003 | **0.002** |
| Reflection | **-0.003** | 0.013 | 0.023 |
| Narrative_cite | **0.000** | -0.013 | **0.000** |
| Transition | **0.002** | -0.014 | 0.020 |

Table 7: Ablation study results comparing sentence discourse roles. Scores closer to 0 are better for ratio difference.

| Metric | Full Model | ACPN only | GoT only |
|---|---|---|---|
| Dominant (P) | **0.917** | 0.776 | 0.827 |
| Dominant (R) | **0.928** | 0.704 | 0.749 |
| Dominant (F1) | **0.922** | 0.738 | 0.817 |
| Reference (P) | **0.917** | 0.684 | 0.717 |
| Reference (R) | **0.983** | 0.838 | 0.746 |
| Reference (F1) | **0.949** | 0.753 | 0.731 |

Table 8: Ablation study results for citation importance fidelity.

| Metric | Full Model | ACPN only | GoT only |
|---|---|---|---|
| Background (P) | **0.728** | 0.302 | 0.714 |
| Background (R) | 0.964 | 0.519 | **0.965** |
| Background (F1) | **0.829** | 0.382 | 0.821 |
| Differences (P) | **0.844** | 0.203 | 0.748 |
| Differences (R) | **0.960** | 0.464 | **0.960** |
| Differences (F1) | **0.898** | 0.283 | 0.841 |
| Extends (P) | **0.862** | 0.283 | 0.841 |
| Extends (R) | 0.980 | 0.700 | **1.000** |
| Extends (F1) | **0.917** | 0.400 | 0.836 |
| Future Work (P) | **1.000** | 0.000 | 0.750 |
| Future Work (R) | **1.000** | 0.000 | **1.000** |
| Future Work (F1) | **1.000** | 0.000 | 0.857 |
| Motivation (P) | 0.684 | 0.455 | **0.909** |
| Motivation (R) | 0.983 | 0.581 | **0.987** |
| Motivation (F1) | 0.806 | 0.510 | **0.947** |
| Similarities (P) | **0.955** | 0.680 | 0.711 |
| Similarities (R) | 0.955 | 0.674 | **0.959** |
| Similarities (F1) | **0.955** | 0.103 | 0.817 |
| Uses (P) | **0.991** | 0.821 | 0.958 |
| Uses (R) | **0.950** | 0.319 | 0.949 |
| Uses (F1) | **0.970** | 0.460 | 0.954 |

Table 9: Ablation study results for citation intent fidelity.

## E Comparison of Compressed and Final Drafts

In Tables 11-15, we see only small differences in metric performance between the Compressed and Final drafts produced by our Writer module. Specifically, the Final draft trades a slight dip in performance on text generation metrics for slightly improved citation intent fidelity, discourse ratios, and citation grouping/ordering.

Figure 6 shows the Compressed draft of the example RWS. We can see that the it uses some ambiguous or unexpected phrasing in order to reduce the overall word count, often falling into noticeably AI-like word choice, compared to the more expressive Final draft shown in Figure 5.

| Metric | Full Model | ACPN only | GoT only |
|---|---|---|---|
| Edge Jaccard | **0.847** | 0.665 | 0.689 |
| Kendall's $\tau$ | **0.886** | 0.662 | 0.666 |

Table 10: Ablation study results for citation grouping and ordering.

| Metric | Compressed | Final |
|---|---|---|
| ROUGE-1 (P) | **0.684** | 0.653 |
| ROUGE-1 (R) | 0.977 | **0.978** |
| ROUGE-1 (F1) | **0.794** | 0.771 |
| ROUGE-2 (P) | **0.676** | 0.644 |
| ROUGE-2 (R) | 0.973 | **0.974** |
| ROUGE-2 (F1) | **0.789** | 0.766 |
| ROUGE-L (P) | **0.679** | 0.648 |
| ROUGE-L (R) | **0.975** | **0.975** |
| ROUGE-L (F1) | **0.791** | 0.769 |
| BERTScore (P) | **0.750** | 0.738 |
| BERTScore (R) | **0.933** | 0.932 |
| BERTScore (F1) | **0.838** | 0.832 |
| BLEU (Macro) | **66.12** | 63.42 |
| BLEU (Micro) | **71.01** | 68.09 |
| METEOR (Macro) | **0.795** | 0.778 |
| METEOR (Micro) | **0.514** | 0.509 |

Table 11: Comparison of compressed and final drafts using traditional text generation metrics.

## F Example Related Work Sections

These example RWS are generated for "Parameter Sharing Methods for Multilingual Self-Attentional Translation Models" (Sachan and Neubig, 2018). This example also produced the GoT excerpt shown in Figure 2; it was chosen to be easily understandable by an NLP audience and may not demonstrate every distinction among approaches discussed in Section 5.

Note that the target RWS (Figure 4) has three sentences removed, due to their containing citations whose cited papers are not available in OARelatedWork.

| Metric | Compressed | Final |
|---|---|---|
| Single_summ | **0.011** | **0.011** |
| Multi_summ | 0.003 | **0.002** |
| Reflection | -0.003 | -0.003 |
| Narrative_cite | 0.001 | **0.000** |
| Transition | 0.002 | 0.002 |

Table 12: Comparison of compressed and final drafts on sentence discourse roles. Scores closer to 0 are better for ratio difference.

| Metric | Compressed | Final |
|---|---|---|
| Dominant (P) | **0.951** | 0.917 |
| Dominant (R) | **0.944** | 0.928 |
| Dominant (F1) | **0.951** | 0.922 |
| Reference (P) | **0.981** | 0.917 |
| Reference (R) | 0.945 | **0.983** |
| Reference (F1) | **0.963** | 0.949 |

Table 13: Comparison of compressed and final drafts on citation importance fidelity.

| Metric | Compressed | Final |
|---|---|---|
| Background (P) | 0.725 | **0.728** |
| Background (R) | 0.962 | **0.964** |
| Background (F1) | 0.827 | **0.829** |
| Differences (P) | 0.841 | **0.844** |
| Differences (R) | 0.960 | 0.960 |
| Differences (F1) | 0.890 | **0.898** |
| Extends (P) | 0.823 | **0.862** |
| Extends (R) | **1.000** | 0.980 |
| Extends (F1) | 0.903 | **0.917** |
| Future Work (P) | 0.750 | **1.000** |
| Future Work (R) | 1.000 | 1.000 |
| Future Work (F1) | 0.857 | **1.000** |
| Motivation (P) | **0.690** | 0.684 |
| Motivation (R) | 0.983 | 0.983 |
| Motivation (F1) | **0.811** | 0.806 |
| Similarities (P) | **0.959** | 0.955 |
| Similarities (R) | 0.955 | 0.955 |
| Similarities (F1) | **0.957** | 0.955 |
| Uses (P) | 0.982 | **0.991** |
| Uses (R) | 0.950 | 0.950 |
| Uses (F1) | 0.965 | **0.970** |

Table 14: Comparison of compressed and final drafts on citation intent fidelity.

| Metric | Compressed | Final |
|---|---|---|
| Edge Jaccard | 0.843 | **0.847** |
| Kendall's $\tau$ | 0.866 | **0.886** |

Table 15: Comparison of compressed and final drafts on citation grouping and ordering.

In this section, we will review the prior work related to MTL and multilingual translation.

Ando and Zhang (2005) obtained excellent results by adopting an MTL framework to jointly train linear models for NER, POS tagging, and language modeling tasks involving some degree of parameter sharing. Later, Collobert et al. (2011) applied MTL strategies to neural networks for tasks such as POS tagging, NER, and chunking by sharing the sequence encoder and reported moderate improvements in results. [Sentences removed due to cited papers missing from OARelatedWork.] MTL has also been widely applied to multilingual translation that will be discussed next.

On the multilingual translation task, Dong et al. (2015) obtained significant performance gains by sharing the encoder parameters of the source language while having a separate decoder for each target language. Later, Firat et al. (2016) attempted the more challenging task of many-to-many translation by training a model that consisted of one shared encoder and decoder per language and a shared attention layer that was common to all languages. This approach obtained competitive BLEU scores on ten European language pairs while substantially reducing the total parameters. Recently, Johnson et al. (2017) proposed a unified model with full parameter sharing and obtained comparable or better performance compared with bilingual translation scores. During model training and decoding, target language was specified by an additional token at the beginning of the source sentence. Coming to low-resource language translation, Zoph et al. (2016) used a transfer learning approach of fine-tuning the model parameters learned on a high-resource language pair of French→English and were able to significantly increase the translation performance on Turkish and Urdu languages. Recently, Gu et al. (2018) addresses the many-to-one translation problem for extremely low-resource languages by using a transfer learning approach such that all language pairs share the lexical and sentence-level representations. By performing joint training of the model with high-resource languages, large gains in the BLEU scores were reported for low-resource languages.

In this paper, we first experiment with the Transformer model for one-to-many multilingual translation on a variety of language pairs and demonstrate that the approach of Johnson et al. (2017) and Dong et al. (2015) is not optimal for all kinds of target-side languages. Motivated by this, we introduce various parameter sharing strategies that strike a happy medium between full sharing and partial sharing and show that it achieves the best translation accuracy.

Figure 4: Cleaned target related work section from “Parameter Sharing Methods for Multilingual Self-Attentional Translation Models" (Sachan and Neubig, 2018).

Recent advancements in neural machine translation (NMT) have prominently featured zero-resource and low-resource translation methods. Collobert et al. (2011) laid the groundwork for NMT by emphasizing the necessity of robust architectures to tackle diverse linguistic challenges. Firat et al. (2016) complemented this with a multi-way multilingual NMT model that employs a shared attention mechanism, improving efficiency and performance in zero-shot translation scenarios. Zoph et al. (2016) explored transfer learning strategies, highlighting how leveraging high-resource language pairs can considerably enhance BLEU scores for low-resource languages, effectively addressing data scarcity issues.

Further, Gu et al. (2018) reinforced the significance of shared representations in low-resource translations, aligning with the insights of Dong et al. (2015) regarding multi-task learning frameworks that optimize performance through the use of unlabeled data. Johnson et al. (2016) also advocated for semi-supervised learning techniques to boost model performance, echoing Zoph et al.'s findings. In contrast to these established approaches, our paper introduces a novel parameter sharing strategy within the Transformer architecture, strategically focusing on specific parameters to enhance multilingual translation accuracy across both closely and distantly related languages.

Figure 5: Related work section generated by pruned GRASP.

Recent advancements in neural machine translation (NMT) have prominently featured zero-resource and low-resource translation methods. Collobert et al. (2011) introduced foundational NMT techniques emphasizing robust architectures for diverse linguistic challenges. Firat et al. (2016) complemented this by proposing a multi-way multilingual NMT model with a shared attention mechanism, enhancing efficiency and competitive performance in zero-shot translation settings. Zoph et al. (2016) explored transfer learning strategies, demonstrating that high-resource language pairs can significantly boost BLEU scores for low-resource languages, thus addressing data scarcity.

Gu et al. (2018) further highlighted the importance of shared representations in low-resource translations, aligning with Dong et al. (2015) on multi-task learning frameworks that optimize performance using unlabeled data. Moreover, Johnson et al. (2016) indicated the potential of semi-supervised learning techniques to improve model performance, complementing the findings of Zoph et al. (2016). Contrasting with these approaches, our work proposes a novel parameter sharing strategy within the Transformer architecture, focusing on sharing specific parameters to enhance multilingual translation accuracy across diverse language pairs.

Figure 6: Compressed draft related work section generated by pruned GRASP.

Natural Language Processing (NLP) has seen significant advancements in recent years, particularly in the realm of Neural Machine Translation (NMT). The foundational work by Collobert et al. (2011) established key principles of NMT, emphasizing the simplicity of implementation and end-to-end training capabilities. This work laid the groundwork for subsequent research that explored more advanced architectures and methodologies.

One of the pivotal developments in NMT was the introduction of the Transformer model by Vaswani et al. (2017), which revolutionized the field by replacing recurrent neural networks (RNNs) with self-attention mechanisms. This transition, as noted in the current work, has facilitated faster processing and improvements in translation accuracy across multilingual tasks. Specifically, the work of Firat et al. (2016) exemplified the application of a shared attention mechanism in multilingual settings, supporting the notion that joint training on multiple languages can enhance translation capabilities.

Building on these ideas, our research presents a novel framework that incorporates a shared encoder with multiple decoders, maintaining some shared parameters while allowing flexibility for distinct target languages. This approach is a significant extension of the multi-task learning paradigm previously explored by Dong et al. (2015), who utilized a shared encoder for multiple target languages, albeit with separate decoders. By contrast, our model introduces a hybrid structure that optimally balances shared and unique parameters, demonstrating superior performance in translating unrelated language pairs, a challenge highlighted in the works of Gu et al. (2018) and Johnson et al. (2016).

Furthermore, the concept of transfer learning for low-resource languages, as discussed by Zoph et al. (2016), aligns with our findings that partial sharing of parameters can be particularly beneficial when translating between languages from distant families. Our experimental results support this, showing that while full parameter sharing may yield satisfactory results for closely related languages, a more tailored approach is essential for achieving optimal accuracy in diverse linguistic contexts.

In summary, while prior research has extensively covered various architectures and training methodologies for NMT, the current work advances the understanding of parameter sharing strategies within the Transformer framework, thereby providing a more nuanced approach to multilingual translation. Our findings not only validate existing theories but also propose innovative solutions that enhance the efficacy of NMT systems in practical applications. This research contributes to the growing body of knowledge in NLP, particularly in the context of Arabic and multilingual texts, where effective translation strategies are paramount.

Figure 7: Related work section generated by unpruned GRASP.

In recent years, there has been substantial research into multilingual translation and the application of parameter sharing techniques. Early work by Ando and Zhang (2005) and Collobert et al. (2008) established the effectiveness of multi-task learning (MTL) frameworks, demonstrating that sharing parameters across tasks can enhance accuracy in various applications, including named entity recognition (NER) and part-of-speech (POS) tagging. These foundational studies laid the groundwork for subsequent investigations into multilingual neural machine translation (NMT).

Dong et al. (2015) explored parameter sharing within the context of multilingual translation, achieving performance gains by sharing encoder parameters while maintaining separate decoders for different languages. This approach was further advanced by Firat et al. (2016), who introduced a many-to-many translation model that shared encoders, decoders, and attention layers, resulting in competitive BLEU scores with a reduced parameter count. Building on this, Johnson et al. (2017) proposed a unified model that fully shared parameters across languages, demonstrating comparable performance to bilingual models and highlighting the potential of comprehensive parameter sharing in multilingual settings.

The challenge of low-resource languages also garnered attention, with Zoph et al. (2016) and Gu et al. (2018) employing transfer learning techniques to improve translation outcomes for languages with limited training data by leveraging resources from high-resource languages. Their work underscored the importance of parameter sharing in enhancing the robustness of translation systems in multilingual contexts.

Our study contributes to this growing body of work by examining various parameter sharing strategies specifically within self-attentional Transformer models. We investigate how different sharing approaches can optimize translation accuracy, particularly when dealing with languages from diverse families, thus addressing the limitations of full parameter sharing identified in earlier research. By integrating insights from previous studies, we aim to provide a balanced perspective on how to effectively leverage parameter sharing in multilingual translation tasks.

Figure 8: Related work section generated by the Li and Ouyang (2025) baseline.

The field of multilingual neural machine translation (NMT) has seen significant advancements through various architectural and training innovations aimed at improving performance across multiple language pairs. A key development in this domain is the proposal of a single neural translation model capable of handling multiple languages, as demonstrated by the work of Firat et al. (2016) and Johnson et al. (2016). Both studies utilize shared components, such as a unified attention mechanism and shared wordpiece vocabulary, to facilitate translation across diverse languages without altering the core architecture of the NMT system. These approaches stand in contrast to the more traditional encoder-decoder frameworks, which, as noted in Zoph et al. (2016), often fall short in low-resource scenarios. Here, transfer learning emerges as a crucial strategy, where parameters from high-resource language pairs are adapted to improve translation quality for low-resource pairs.

The challenge of translating a single source language into multiple target languages is addressed by Dong et al. (2015) through a multi-task learning model that leverages a shared encoder, illustrating the potential for improved translation quality via shared semantic and syntactic representations. This concept of parameter sharing resonates with the findings of Rie et al. (2005), which explore structural learning without empirical risk minimization, emphasizing the importance of structural parameters in enhancing predictive performance.

Further extending the paradigm of multilingual translation, Gu et al. (2018) introduces a transfer-learning approach that facilitates lexical and sentence-level sharing across languages with limited parallel data, underscoring the utility of universal representations in bridging resource gaps. This aligns with the broader trend towards reducing task-specific engineering, as advocated by Collobert et al. (2011), which promotes a unified network architecture across various natural language processing tasks, including translation.

Collectively, these studies highlight the efficacy of shared architectures and transfer learning in overcoming the limitations of multilingual NMT systems, demonstrating that strategic parameter sharing can significantly enhance translation performance across diverse language pairs.

Figure 9: Related work section generated by the Select, Read, Write (Liu et al., 2025) baseline.