Explanatory Summarization with Discourse-Driven Planning

Dongqi Liu[Ω]*, Xi Yu[Ω], Vera Demberg[Ω], Mirella Lapata[Θ]
[Ω]Saarland University, Germany
[Θ]University of Edinburgh, UK
[Ω]{dongqi,xiyu,vera}@lst.uni-saarland.de
[Θ]mlap@inf.ed.ac.uk

Abstract

Lay summaries for scientific documents typically include explanations to help readers grasp sophisticated concepts or arguments. However, current automatic summarization methods do not explicitly model explanations, which makes it difficult to align the proportion of explanatory content with human-written summaries. In this paper, we present a plan-based approach that leverages discourse frameworks to organize summary generation and guide explanatory sentences by prompting responses to the plan. Specifically, we propose two discourse-driven planning strategies, where the plan is conditioned as part of the input or part of the output prefix, respectively. Empirical experiments on three lay summarization datasets show that our approach outperforms existing state-of-the-art methods in terms of summary quality, and it enhances model robustness, controllability, and mitigates hallucination. The project information is available at https://dongqi.me/projects/ExpSum.

1 Introduction

In the domain of lay summarization for scientific documents, the inclusion of explanatory content is beneficial for improving readability and accessibility, particularly when dealing with difficult concepts or complicated statements (Srikanth and Li, 2021; August et al., 2022; Luo et al., 2022; Goldsack et al., 2022). As an example, consider the target sentence being explained (highlighted in orange) within the lay summary in Figure 1. It describes the role of the cerebellum (a similar sentence can be found in the source document, also highlighted in orange) and is accompanied by an explanation highlighted in green, which compares the cerebellum to a coach. This analogy, which illustrates how the cerebellum adjusts the timing of actions based on previous movements, makes the preceding technical description easier to understand. Using simpler vocabulary or shorter sentences can make the summaries easier to read, but could also lead to misinterpretations (Liu and Demberg, 2023; Han et al., 2023; Wu et al., 2023b; Hewett et al., 2024). Thus, explanations are often used to avoid this, balancing accessibility with accuracy and cognitive load (Oksa et al., 2010).

Empirical analysis of lay summaries supports the observation that explanations are commonplace. Specifically, by using a discourse parser (Liu et al., 2020, 2021) to identify explanations in the eLife and PLOS (lay summarization) datasets (Goldsack et al., 2022), we find that explanations account for approximately 19.02% and 18.19% of summary sentences, respectively. This proportion of explanations in lay summaries is three times higher than in expert summaries (6.28% and 5.06%) and nearly four times greater than in the original academic papers (5.16% and 4.76%). In other words, explanations are significantly more common than other discourse relations like *Condition* and *Purpose*. This underscores their importance in structuring lay summaries and enhancing text accessibility.

Most lay summarization models follow an end-to-end approach (Goldsack et al., 2022; Liu et al., 2022; Goldsack et al., 2023; Liu et al., 2024) without *explicitly* accounting for explanations. As a result, explanations are often underrepresented in generated summaries, which in turn may be the reason why generated summaries lack the clarity and readability found in human-produced ones (Guo et al., 2021b; Goldsack et al., 2023; Tang et al., 2023; Liu and Demberg, 2023; Zhang et al., 2024; Wang et al., 2025). In this paper, we develop neural models that are capable of generating *lay summaries* with controlled explanatory content. We achieve this by *planning* the content of the summary, thus *directly* steering the model towards

* This work was conducted during the research visit at the University of Edinburgh.

Transactions of the Association for Computational Linguistics, vol. 13, pp. 1146–1170, 2025. https://doi.org/10.1162/tacl.a.30
Action Editor: Jessy Li. Submission batch: 10/2024; Revision batch: 4/2025; Published 9/2025.
© 2025 Association for Computational Linguistics. Distributed under a CC-BY 4.0 license.

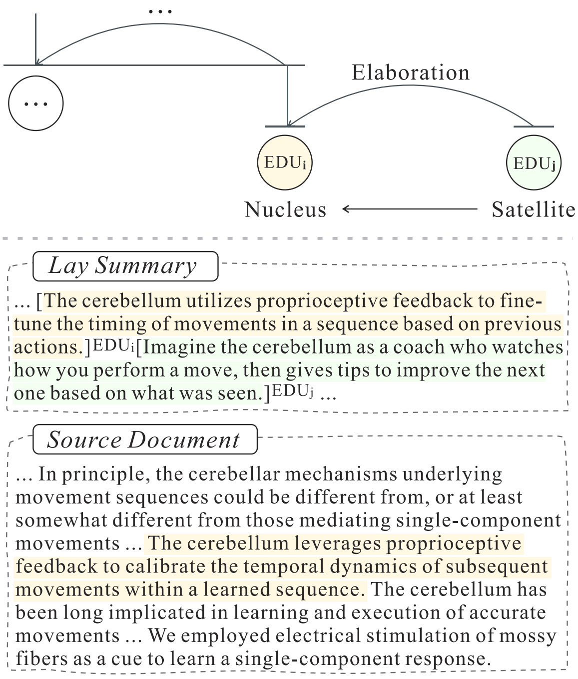

Figure 1: An excerpt of a source document paired with its summary (bottom); the explanatory sentence and its target are highlighted in green and orange, respectively. The RST tree (top) for the text corresponding to the two sentences shows they are linked by the rhetorical relation *Elaboration*.

generating explanations. Plan-based models are a suitable type of approach for lay summarization, as they are less prone to hallucinations, offer greater controllability (Puduppully et al., 2019; Narayan et al., 2021; Moryossef et al., 2023; Narayan et al., 2023; Huot et al., 2023, 2024; Liu et al., 2025), and can be flexibly adapted to different generation tasks depending on how plans are formulated.

Our method conceptualizes plans as a series of questions that trigger explanatory content, with their strategic placement learned from training data. This placement is not arbitrary; rather, it reflects where human writers tend to insert explanations to clarify complex claims or findings. Importantly, our approach not only models where to explain, but also what is being explained and how—e.g., through analogy, background, or causal reasoning. In addition, evaluating such explanations remains challenging, as automatic consistency detection metrics often misclassify useful additions as hallucinations (Cao et al., 2022). To address this, we incorporate external knowledge verification, but acknowledge that finer-grained human or LLM-based judgments are still needed for nuanced evaluation.

A common challenge with plan-based models is the scarcity of datasets with plan annotations. As a result, plans are often reverse-engineered from existing reference summaries through automatic means, e.g., by identifying entities (Narayan et al., 2021; Huot et al., 2024) or generating questions and their answers (Narayan et al., 2023). Our approach draws inspiration from Rhetorical Structure Theory (RST; Mann and Thompson 1987). RST represents texts as trees (see Figure 1), where the leaves correspond to Elementary Discourse Units (EDUs) and the nodes specify how these and larger units are connected through rhetorical relations (e.g., *Elaboration*). Discourse units are further characterized in terms of their text importance: *nuclei* represent central segments, whereas *satellites* denote peripheral ones. As depicted in the upper part of Figure 1, EDU_j acts as a satellite to EDU_i, supplying an explanation.

We automatically generate plans from reference summaries annotated with RST trees (Liu et al., 2020, 2021). We hypothesize that explanatory EDUs answer latent questions (Beaver et al., 2017), which we verbalize. Our plans are designed to pose these latent questions, but we do not state the answers explicitly (see Figure 2). We develop two model variants that differ in how they integrate planning in the summarization process, with plans either included as part of the source document or as part of the target summary. Empirical results using `Mistral` (Jiang et al., 2023) as backbone confirm that our discourse-driven planning approach significantly improves the quality of lay summaries but also has the potential to control the generation of explanatory content to a certain extent. Our key contributions are as follows:

- We introduce the task of *explanatory summarization*, which aims to generate lay summaries with controlled explanations.

- We propose two discourse-driven models that generate lay summaries without requiring manual annotations. Empirical results show improvements in summary quality and better alignment with human performance.

- Both qualitative and quantitative analyses indicate that our models mitigate content hallucinations and enhance factual accuracy. However, a gap still exists compared to human performance.

Planning questions

q1: How does the cerebellum use feedback to adjust the timing of movements in a sequence?
q2: How does the cerebellum use feedback from one blink to trigger the next in a sequence?
q3: How can a blink in one eye trigger a blink in the other eye?

Processed summary

[The cerebellum utilizes proprioceptive feedback to fine-tune the timing of movements in a sequence based on previous actions.]^{t_1} [Imagine the cerebellum as a coach who watches how you perform a move, then gives tips to improve the next one based on what was seen.]^{e_1} But how exactly does it achieve this? [To investigate, we trained rabbits to blink in response to an external cue and explored whether the cerebellum could use feedback from one blink to trigger the next.]^{t_2} [As expected, after learning the initial blink, the rabbits blinked again in response to their own first blink, creating a chain of movements.]^{e_2} Control experiments confirmed that each blink was initiated by the previous one rather than the original cue. Consistent patterns of brain activity during this process indicate that the cerebellum adjusts movement based on feedback from previous actions. [Building on this, we trained rabbits to blink on cue, and they learned to initiate additional blinks in response to earlier blinks in the sequence.]^{t_3} [We further found that the rabbits could use a blink from one eye as a cue to trigger a blink in the other eye, suggesting that the same mechanism governs these movements.]^{e_3} This raises the possibility that the cerebellum might also guide sequences of cortical activity during cognitive tasks, given its extensive connections to the cortex, a question future experiments should explore.

Target EDU   Explanatory EDU

Figure 2: We use `DMRST` (Liu et al., 2021) to extract explanatory (e) EDUs and their target (t) EDUs from reference summaries, and then feed this data into `GPT-4o` to generate plans (b).

2 Related Work

Lay Summarization Lay Summarization involves transforming complicated scientific texts into accessible summaries for non-experts, thereby enhancing public understanding of scientific research. Existing investigations have tackled several challenges, such as creating datasets and developing task-specific models. Chandrasekaran et al. (2020) pioneered the task of scientific lay summarization, while follow-up work focused on the development of larger datasets, including CDSR (Guo et al., 2021b), eLife, PLOS (Goldsack et al., 2022), and SciNews (Liu et al., 2024). Methodological advancements include integrating domain-specific knowledge graphs with encoder-decoder models (Goldsack et al., 2023) and developing methods like ATLAS (Zhang et al., 2024), which facilitate the generation of summaries with varying degrees of accessibility for lay audiences. However, none of the existing approaches emphasize discourse-driven mechanisms for modulating summary content, nor do they focus on evaluating the quality or proportion of explanatory content.

Interactive Summarization Interactive Summarization refers to the process of enabling users to engage with and influence the summarization process through interactive interfaces (Costa et al., 2018; Guo et al., 2021a; August et al., 2023; Kim an dLee, 2019; Fok et al., 2024). For example, August et al. (2023) introduced an interface that provides localized plain-language key points, definitions, and question-based navigation, allowing users to explore medical research more effectively. Similarly, Fok et al. (2024) proposed a design for recursively expandable abstracts, where users can click on keywords or phrases to access system-generated text and citation links. While interactive summarization is not the focus of our work, it is worth pointing out that users could choose to modify the plan and influence the generation process.

RST in Summarization Rhetorical Structure Theory (RST) is a discourse framework that analyzes the relations between text units, termed Elementary Discourse Units (EDUs), and their roles within the document (Marcu, 1997, 1999, 2000). RST categorizes connections among EDUs into different rhetorical or coherence relations, such as *Elaboration*, *Contrast*, and *Causality*. Within this framework, EDUs are arranged into a hierarchical tree structure, where nucleus EDUs are supported or elaborated by satellite EDUs. Early research (Marcu, 1997, 1999; Louis et al., 2010) has found that human-written summaries are often derived from nucleus EDUs, affirming RST's usefulness for summarization tasks. Several studies have explored ways to inject discourse structure into summarization models and demonstrated improved performance compared to models without such discourse injection (Xu et al., 2020; Dong et al., 2021; Chen and Yang, 2021; Liu et al., 2023; Liu and Demberg, 2024). Our work applies RST to discern explanatory EDUs, which elucidate or support propositions, and their related target (explained) EDUs within reference summaries.

Planning in Summarization Text summarization with planning involves organizing content through structured representations prior to generating summaries. Existing work has primarily focused on *phrase-level* planning, which has been shown to improve summary quality and reduce hallucination (Narayan et al., 2021, 2023). Plans have previously taken the form of named entities (Narayan et al., 2021; Liu and Chen, 2021; Huot et al., 2024), keyword prompts (Creo et al., 2023), and question-answer pairs, referred to as blueprint (Narayan et al., 2023). Our work introduces *EDU-level* planning, placing emphasis on larger semantic units. Our plans take the form of questions (without answers), triggered by explanations identified through RST. This definition is more flexible (compared to question-answer pairs), as we do not require answers to be incorporated and are not limited to a specific style of answers or questions.

Question Under Discussion Question Under Discussion (QUD) conceptualizes discourse as a dynamic interaction of continuously posed and answered questions (Roberts, 2012; Benz and Jasinskaja, 2017; De Kuthy et al., 2020; Ko et al., 2023). It can be seen as a question generation strategy, where each sentence serves as a response to an implicit question arising from its preceding context, thereby delineating the text's intent and structure (Beaver et al., 2017; Wu et al., 2023a,b). Although QUD has been explored in discourse analysis (Ko et al., 2023; Wu et al., 2023a) and to interpret complex linguistic phenomena like presupposition and information structure (Beaver et al., 2017), its application in text generation and summarization has received limited attention. Wu et al. (2023b) leveraged manual annotation to recover implicit questions to guide sentence-level simplification. Our work draws inspiration from QUD in formulating plans as a series of questions. We work with more complex documents and rely on large language models, bypassing issues of scalability arising from manual annotations.

3 Explanatory Summarization

3.1 Task Definition

We formalize *explanatory summarization* as follows: Let $D = \{(x_1, s_1), (x_2, s_2), \ldots, (x_n, s_n)\}$ denote a dataset where each tuple (x_i, s_i) contains a document x_i and its corresponding lay summary s_i. We further assume s_i contains some explanatory elements e_i such as examples, contextual explanations, or background information. Our objective is to devise a summarization model that can model the generation of this explanatory content.

3.2 Model Overview

Inspired by the blueprint approach (Narayan et al., 2023), we propose a discourse-driven planning method to achieve the above goal. This involves transforming the original dataset $D = \{(x_1, s_1), (x_2, s_2), \ldots, (x_n, s_n)\}$ into $D' = \{(x_1, b_1, s_1), (x_2, b_2, s_2), \ldots, (x_n, b_n, s_n)\}$ by introducing planning component b. We then train a summarization model to learn the conditional probability distribution $P(S|X, B)$ or $P(B, S|X)$, where $S = \{s_1, s_2, \ldots, s_n\}$, $X = \{x_1, x_2, \ldots, x_n\}$, and $B = \{b_1, b_2, \ldots, b_n\}$. Each b_i in B consists of a series of ordered plan questions $(q_1, q_2, \ldots, q_n)$, which do not initially exist in the original dataset D.

Note that $P(S|X, B)$ assumes plan b is treated as part of the input sequence, whereas $P(B, S|X)$ treats b as a prefix to summary generation. The key challenge in training either model is that plan b is latent. We thus automatically augment the original training pairs (x, s) into triplets (x, b, s), with b acting as a conditional or control signal, offering an overarching plan for summary generation while also guiding the creation of explanatory sentences by prompting responses to the plan.

3.3 Explanatory Content Extraction

To obtain triplets (x, b, s), we apply `DMRST` (Liu et al., 2020, 2021), an RST-based parser,

Dataset	# Training	# Validation	# Test	Avg. Doc Tokens	Avg. Summ Tokens	Coverage	Density	Compression Ratio
SciNews	33,497	4,187	4,188	7,760.90	694.80	0.74	0.94	12.71
eLife	4,346	241	241	7,833.14	383.02	0.82	1.77	20.52
PLOS	24,773	1,376	1,376	5,340.58	178.66	0.07	0.90	36.06

Table 1: Descriptive statistics for SciNews, eLife, and PLOS datasets.

to automatically identify and extract all EDUs functioning as explanations in reference summary s (see Figure 2). Out of 18 rhetorical categories, we select four that are relevant to explanatory sentences: *Background*, *Elaboration*, *Explanation*, and *Comparison* (Table 5 in Appendix A has the full list). Given the directed nature of RST relations, we can identify discourse units that serve as explanations and the content they explain (see Figure 1). Note that the extracted EDU pairs act only as a silver standard proxy for the subsequent training phase of our models. While our approach is parser-agnostic, we also conduct experiments with alternative parsers (see Section 5 for details).

3.4 Explanatory Plan Generation

We propose that explanatory sentences can serve as responses to (implicit) questions raised by preceding target sentences, aligning with the theoretical framework of QUD. We employ `GPT-4o` (Achiam et al., 2023) to generate these questions from target sentences (attested in reference summary) and the preceding context, as illustrated in Figure 2. For example, the plan question q_2 is generated based on the target sentence t_2 and the sentences that precede it. We do not filter the generated questions, but we create the plan by following the order in which explanations appear in the reference. The prompt for generating plan questions can be found in Appendix Figure 7.

3.5 Explanatory Summarization Models

We propose two model variants. The first variant (`Plan-Output`) concatenates all plan questions b with the summary s to form a sequence $[b; s]$ for each document-summary pair. The training objective for this model is to generate both the plan and the summary based on the input document x.

The second variant (`Plan-Input`) is not an end-to-end model. Instead, it employs a Plan Generation (`PG`) module and a Summary Generation (`SG`) module, which are trained separately. The `PG` module learns to generate plan b from the input document x, and is trained on (x, b) pairs, where x is the input and b is the silver standard plan. The `SG` module is trained on $([x; b], s)$ tuples where $[x; b]$ represents the input document enriched with plan questions (we concatenate input document x with plan b). At test time, the trained `PG` module is used to predict plan $\hat{b}$ based on input document x. Generated questions $\hat{b}$ are then combined with x to form the input $[x; \hat{b}]$ for the `SG` module, which produces the final summary on the test set.

`Plan-Output` trains a single end-to-end model that learns to *jointly* generate the plan and the summary. This design reduces inference overhead, making it well-suited for real-time applications or resource-constrained environments (since only one inference call is made during testing). In contrast, `Plan-Input` adopts a modular approach, training the `PG` and `SG` components *separately*. This makes `Plan-Input` better suited to settings where task requirements evolve over time, allowing for the refinement or replacement of individual module components as needed.

Note that the original Blueprint approach of Narayan et al. (2023), relies on *phrase-level* control (the answers to the questions are mostly named entities) and, as such, cannot be readily used to control the generation of explanations that are typically longer and semantically richer. Moreover, the frequent repetition of entities (i.e., the same entity is often mentioned multiple times) adds unnecessary complexity to the planning process, necessitating the use of filtering algorithms to streamline it. In contrast, planning at the *EDU-level* provides a more cohesive structure and reduces redundancy.

4 Experimental Setup

Datasets We report results on three lay summarization datasets, namely SciNews (Liu et al., 2024), eLife (Goldsack et al., 2022), and PLOS (Goldsack et al., 2022). Table 1 presents different statistics for these datasets, which vary in terms of size, domain, and summary length. In the table, Coverage measures the extent to which the summary directly uses tokens from the source material, with higher coverage indicating that a greater proportion of the summary's tokens come from the source. DENSITY calculates the average length of the source text segments associated with each token in the summary, where higher density suggests the inclusion of longer continuous text segments in the summary. COMPRESSION RATIO represents the ratio between the length of the source document and the summary, with a higher compression ratio signifying a more concise summary.

Automatic Evaluation Metrics We report several complementary metrics aimed at assessing different facets of summary quality. We use ROUGE (Lin, 2004) and BERTSCORE (Zhang et al., 2020) to evaluate *informativeness* against the human references; we report F1 of Rouge-2 (R2) and Rouge-Lsum (RLSUM). We use D-SARI (Sun et al., 2021) and Flesch Reading Ease Formula (FRE, Kincaid et al., 1975) to measure summary *readability*. D-SARI and FRE scores range from 0 to 100, with higher values indicating easier-to-read material and lower scores reflecting more complex passages. Additionally, we compute EXPRATIO, which represents the *proportion of explanatory* EDUs to the total number of EDUs in the generated summary (explanatory and target EDUs always appear in pairs due to the RST structure). A higher EXPRATIO suggests the summary contains more explanations. We also report the generated summary *length*, which we calculate as the average number of summary tokens (AST) using spaCy.

We assess *factual consistency* between the source document and the generated summary using SUMMAC$_{Conv}$ (Laban et al., 2022), which determines whether summary sentences are entailed by the input. However, human-written lay summaries often include additional external information, which could be mistakenly classified as *hallucinations* when assessed automatically with original SUMMAC$_{Conv}$. This highlights the limitations of traditional consistency detection metrics for our task, where only extrinsic hallucinations, rather than all additional explanations, should be penalized.

To address this issue, we propose a new SUMMAC variant. Specifically, for summary sentences with entailment scores below 0.5 (i.e., not supported by the source text), we use the Wikipedia-API to retrieve relevant articles and re-evaluate these sentences using SUMMAC$_{Conv}$. This approach allows us to determine whether sentences unsupported by the source text can be validated using external knowledge bases. If the highest entailment score from the retrieved articles is higher than the original score, it replaces the original score for that sentence. We report both the original SUMMAC$_{Conv}$ and the proposed variant, which we call SUMMAC*. We also leverage VERISCORE (Song et al., 2024) to verify whether the claims in the generated summary align with objective facts by consulting external knowledge bases. For claim extraction and verification, we use `GPT-4o`.

Model Comparisons We build `Plan-Output` and `Plan-Input` (`PG` and `SG`) on top of the `Mistral-7B-Instruct-v0.3`, which we fully fine-tune on the above datasets (enriched with plan annotations). We compare our models against the following `Mistral` configurations: a) zero-shot setting (Mistral$_{ZS}$); b) in-context learning with one randomly selected demonstration from the training split (Mistral$_{ICL}$); and c) full parameter fine-tuning without planning (Mistral$_{FT}$).

We also re-implement the best-performing multi-task blueprint model (Blueprint$_{MT}$) from Narayan et al. (2023), which is optimized for two tasks: 1) question generation given source document and said answer, or 2) summary generation given the same document and its question-answer plan. We use `GPT-4o` to identify answers (while the original `Blueprint` model extracts answers using spaCy, we employ `GPT-4o` which leads to better performance – on average 2.21% higher RLSUM across datasets) and generate corresponding questions and fine-tune with the same backbone `Mistral` model on datasets enriched with question-answer plans.[1]

All comparison models follow identical hyperparameter settings detailed in Appendix B. We also include results with `GPT-4o` (Achiam et al., 2023) in zero-shot (GPT-4o$_{ZS}$) and in-context learning (GPT-4o$_{ICL}$) settings; we compare against the state-of-the-art (SOTA) methods on each dataset (results are directly taken from respective publications). For a fair comparison, we use the same prompt and/or selected sample for both Mistral$_{ZS/ICL}$ and GPT-4o$_{ZS/ICL}$. After inference, we discard all generated plans, retaining only the summaries for automatic evaluation.

[1]Prompts are offered in Appendix Figures 8, 9, 10 and 11.

Data	Model	R2	RLsum	BERTScore	D-SARI	FRE	ExpRatio	AST	SummaC/SummaC*	VeriScore
SciNews	Mistral$_{ZS}$	7.07	37.02	57.21	14.12	27.41	10.45	554.29	46.11/52.65	0.43
	Mistral$_{ICL}$	7.11	37.31	57.57	14.65	31.93	9.87	602.33	50.12/59.27	0.48
	GPT-4o$_{ZS}$	12.79	40.51	58.15	20.15	33.05	11.73	611.09	58.21/64.96	0.56
	GPT-4o$_{ICL}$	13.22	40.73	58.17	20.33	38.91	11.90	634.76	60.04/66.74	0.52
	Mistral$_{FT}$	15.56	46.12	64.38	30.11	38.25	13.61	669.27	64.10/69.73	0.56
	Blueprint$_{MT}$	15.62	46.10	64.33	32.35	40.01	15.03	688.93	72.35/75.18	0.62
	Plan-Output	15.73	46.30	**65.34**	36.23	42.25	17.51	685.33	72.29/75.03	0.67
	Plan-Input	**15.88**†‡	**46.41**†‡	65.32†‡	**37.18**†‡	**43.11**†‡	**17.68**†‡	**692.14**†‡	**72.40**†/**75.38**†‡	**0.71**†‡
	Liu et al. (2024)	14.92	45.19	62.80	—	—	—	—	—	—
	Reference Summary	—	—	—	—	42.23	18.91	694.80	45.08/81.94	0.81
eLife	Mistral$_{ZS}$	8.90	36.19	61.38	18.28	28.43	10.29	301.16	45.45/56.68	0.39
	Mistral$_{ICL}$	8.97	36.26	61.40	20.07	33.60	10.30	333.27	47.42/58.98	0.45
	GPT-4o$_{ZS}$	11.15	44.29	64.34	24.57	38.17	11.10	329.39	47.30/58.46	0.56
	GPT-4o$_{ICL}$	11.31	44.38	63.77	25.52	44.19	11.22	368.42	50.92/60.07	0.60
	Mistral$_{FT}$	14.40	47.68	86.87	32.60	48.42	13.56	377.59	58.31/62.33	0.58
	Blueprint$_{MT}$	14.85	48.09	87.22	35.31	50.33	14.51	375.47	60.31/65.44	0.62
	Plan-Output	14.91	48.27	87.57	38.77	**56.76**	17.71	377.61	61.33/68.21	0.70
	Plan-Input	**15.11**†‡	**48.64**†‡	**87.92**†‡	**39.22**†‡	55.48†‡	**17.73**†‡	**380.39**†‡	**61.35**†‡/**68.28**†‡	**0.75**†‡
	Goldsack et al. (2023)	14.24	45.71	85.40	—	—	—	—	—	—
	Liu and Demberg (2024)	14.92	48.21	87.81	—	—	—	—	—	—
	Zhang et al. (2024)	12.57	44.14	85.20	—	—	—	—	—	—
	Reference Summary	—	—	—	—	51.83	19.02	383.02	49.38/82.28	0.83
PLOS	Mistral$_{ZS}$	6.43	31.42	63.24	19.09	29.70	11.03	130.14	45.83/67.47	0.41
	Mistral$_{ICL}$	6.50	31.66	63.31	21.27	35.93	11.19	149.30	49.02/68.87	0.44
	GPT-4o$_{ZS}$	11.57	35.39	64.22	21.06	32.69	12.31	155.25	55.23/70.79	0.52
	GPT-4o$_{ICL}$	11.72	35.40	64.50	23.74	39.44	12.31	152.38	58.02/73.03	0.50
	Mistral$_{FT}$	14.31	40.22	87.69	32.39	36.45	13.70	158.93	60.02/75.25	0.57
	Blueprint$_{MT}$	14.20	41.35	88.04	33.48	36.04	14.67	166.54	62.22/77.67	0.63
	Plan-Output	15.28	41.33	88.17	39.22	40.01	**18.12**	167.77	62.25/77.79	**0.74**
	Plan-Input	**15.72**†‡	**41.64**†‡	**88.22**†‡	**40.07**†‡	**40.27**†‡	17.65†‡	**171.12**†‡	**62.27**†/**77.95**†‡	0.72†‡
	Goldsack et al. (2022)	13.52	38.63	—	—	—	—	—	—	—
	Zhang et al. (2024)	12.33	40.60	85.70	—	—	—	—	—	—
	Reference Summary	—	—	—	—	30.08	18.19	178.66	53.43/84.71	0.87

Table 2: Model performance on three lay summarization datasets. Bold numbers represent the best results achieved by the models in each test set, excluding human results from the comparison. ExpRatio is the number of explanatory EDUs over all EDUs in summaries. Symbols † and ‡ denote that Plan-Input is statistically significant ($p < 0.05$) against Mistral$_{FT}$ and Blueprint$_{MT}$ using paired t-test, respectively.

5 Results

Our results are summarized in Table 2, which consists of three main blocks corresponding to different datasets. Within each block, we compare models in zero-shot and in-context learning settings against fine-tuned systems and, where applicable, against the human reference summaries.

We find the performance of fine-tuned models to be superior. Across all metrics and datasets, Plan-Input emerges as the top-performing model, followed closely by Plan-Output (see R2 and RLsum). Recall that Plan-Input is a pipeline system where the plan and summary generation stages are trained separately, which we hypothesize makes the summarization task easier.

In comparison to the related Blueprint$_{MT}$ and other comparison models, our two discourse-driven systems produce summaries that are easier to read (see FRE and D-SARI), contain more explanations (see ExpRatio), are closer to human-level performance (see Reference Summary), and align more closely with human summaries in terms of length (AST). Since Blueprint$_{MT}$ relies on entities and phrases to generate plans, it lacks diversity in the types of questions it predicts; in fact, *what* and *who* questions are the greatest majority with an average of 82.2% across datasets. In contrast, our models produce a more balanced distribution, with *what* and *who* questions making up 23.2% of the plan, while *how* and *why* questions account for an average of 66.4%.

As human-authored summaries often include content that is not directly mentioned in the source document (Cao et al., 2022), we observe relatively low (original) SummaC scores. However, when external knowledge bases are considered, we find that human summaries exhibit the lowest level of extrinsic hallucinations. VERISCORE results further corroborate the observation that model-generated summaries lag behind human-authored ones regarding factual consistency. Nevertheless, fine-tuned models are overall more consistent, with plan-based models having an advantage against $\texttt{Mistral}_{FT}$ and state-of-the-art systems. Both `Plan-Input` and `Plan-Output` achieve the highest consistency scores across datasets and metrics, indicating that planning at the EDU-level not only enhances consistency compared to phrase-level planning but also enables the generation of summaries with more elaborations while maintaining lower levels of hallucinations.

Impact of Plan Generation Ablation Next, we ablate our plan generation strategy more closely, comparing it to naive alternatives. We apply `Plan-Input` as our main model, except when indicated otherwise.

1. Lead-3_Q and Lead-K_Q: Generate plans based on the first 3 or first K EDUs in the reference summary.
2. Tail-3_Q and Tail-K_Q: Generate plans based on the last 3 or last K EDUs in the reference summary.
3. Random-3_Q and Random-K_Q: Generate plans based on 3 random or K random EDUs in the reference summary.
4. All-EDUs_Q: Generate plans based on each of the EDUs in the reference summary.
5. NonExp-EDUs_Q: Generate plans based on each of the non-explanatory EDUs in the reference summary.
6. `PG` for zero-shot learning: Train a `PG` model and use the plan questions it generates as instructions to `Mistral` in a zero-shot setting.[2]

Here, K denotes the average number of explanatory EDUs parsed in training set summaries, rounded up to the nearest whole number ($K = 8$ for SciNews, $K = 4$ for eLife, and $K = 2$ for PLOS). For the first four experiments, we do not differentiate between explanatory and non-explanatory EDUs. Each selected EDU serves as

[2]The prompt can be found in Appendix Figure 11.

DATA	MODEL	R2	RLSUM	VERISCORE
SciNews	$\texttt{Mistral}_{FT}$	15.56	46.12	0.56
	Lead-3_Q	15.47	46.12	0.59
	Lead-K_Q	15.50	46.15	0.52
	Tail-3_Q	15.41	45.80	0.50
	Tail-K_Q	15.40	45.85	0.50
	Random-3_Q	15.35	45.88	0.48
	Random-K_Q	15.33	45.94	0.47
	All-EDUs_Q	15.48	46.11	0.52
	NonExp-EDUs_Q	15.42	46.03	0.49
	$\texttt{Mistral}_{ZS+PG}$	7.15	37.11	0.46
eLife	$\texttt{Mistral}_{FT}$	14.40	47.68	0.58
	Lead-3_Q	14.68	48.14	0.59
	Lead-K_Q	14.71	48.11	0.60
	Tail-3_Q	14.25	47.85	0.54
	Tail-K_Q	14.30	47.74	0.54
	Random-3_Q	14.65	48.05	0.56
	Random-K_Q	14.62	48.00	0.55
	All-EDUs_Q	14.62	48.10	0.56
	NonExp-EDUs_Q	14.56	48.02	0.52
	$\texttt{Mistral}_{ZS+PG}$	10.11	38.69	0.42
PLOS	$\texttt{Mistral}_{FT}$	14.31	40.22	0.57
	Lead-3_Q	14.69	40.66	0.59
	Lead-K_Q	14.57	40.59	0.56
	Tail-3_Q	13.78	40.12	0.52
	Tail-K_Q	13.72	40.18	0.51
	Random-3_Q	14.21	40.26	0.54
	Random-K_Q	14.14	40.20	0.53
	All-EDUs_Q	14.41	40.33	0.55
	NonExp-EDUs_Q	14.37	40.33	0.54
	$\texttt{Mistral}_{ZS+PG}$	6.74	32.44	0.48

Table 3: Comparison of different plan generation strategies using `Plan-Input` and `Mistral` in fine-tuned and zero-shot settings.

the target sentence; all preceding EDUs are considered context and are used to generate the corresponding plan questions. For instance, in the All-EDUs_Q experiment (all sentences from the reference summary are involved in the plan generation, rather than being limited to explanatory EDUs), the first sentence is treated as a target EDU (since there are no preceding EDUs, there is no context in this case) and the first plan question is generated. Next, the second EDU is treated as a target EDU, and the first EDU is used as context, and so on.

As shown in Table 3, Lead-3_Q, Lead-K_Q, and All-EDUs_Q strategies demonstrate marginal improvements over $\texttt{Mistral}_{FT}$ across all datasets, while all other strategies result in decreased performance. All strategies in Table 3 are worse than $\texttt{Blueprint}_{MT}$, `Plan-Input`, and `Plan-Output`, in terms of summary quality and factual consistency. This implies that heuristic ablations are less effective for the generation of plan questions. Results for NonExp-EDUs$_Q$ and All-EDUs$_Q$ also verify that modeling and generating plan questions for explanatory sentences is more valuable for lay summarization than modeling and generating plans for other sentences. Results for `Mistral`$_{ZS+PG}$ further show that a robust plan generator improves performance, even in zero-shot settings.

Impact of Discourse Parser We next investigate the extent to which the choice of RST parser influences summary output quality. Specifically, we compare the `DMRST` parser, which has been trained on the RST treebank, against results derived from `GPT-4o` and `Mistral` in zero-shot settings (the instruction is detailed in Appendix Figure 12). For `GPT-4o` and `Mistral`, we use these models to directly extract explanatory sentences and their target sentences, rather than performing RST parsing, as RST parsing is a more complex task.[3] Following Stede et al. (2017), we implement a rule-based method (RB) for extracting explanations (detailed in Appendix C). We also use the RST-Coref parser (Guz and Carenini, 2020) and LLaMA-based RST parser (Maekawa et al., 2024) for comparison. To simulate parser instability, we design a random replacement (RR) method, where explanatory EDUs identified by `DMRST` are randomly replaced with non-explanatory or non-target EDUs. The number of replacements varies from one to the total number of explanatory EDUs per article, mimicking parser inaccuracy in identifying explanatory content. Finally, we introduce full random replacement (FRR), a variant representing the worst-case scenario of parser inaccuracy, where all explanatory EDUs are replaced with randomly selected non-explanatory or non-target EDUs. All these models are trained using `Plan-Input` as the default model.

Figure 3 shows that summarization performance improves with parsing accuracy (see `DMRST` vs. RR/FRR). `DMRST` consistently outperforms more naive methods based on `GPT-4o` and `Mistral`, as well as the RST-Coref parser and the rule-based approach. Its performance is comparable to the LLaMA-based RST parser, but it has a smaller parameter size and incurs lower computational costs in parsing. We also

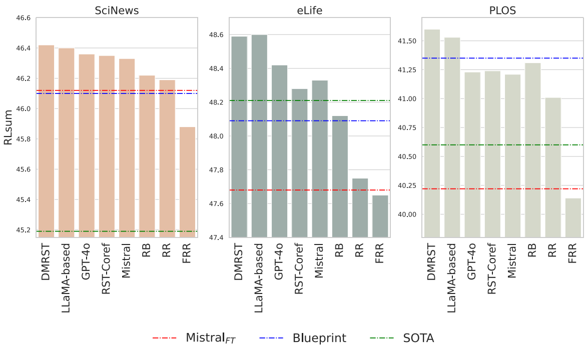

Figure 3: Summary quality as a function of different RST parsers.

find that RST-based models (irrespectively of the parser at hand) outperform the baseline fine-tuned (`Mistral`$_{FT}$) and SOTA models (the performance of the previous best-published model on RLSᴜᴍ for each dataset in Table 2), and frequently surpass the `Blueprint`$_{MT}$ model, as illustrated by the red, green, and blue dashed lines, respectively.

Impact of Plan Quality To evaluate how the quality of plan questions affects model performance, we employ `GPT-4o` in a zero-shot setting as a question generator (see Section 3.4). In addition, we apply the `Mistral` model and the recent question generation model (`RAST`), developed by Gou et al. (2023). Moreover, we implement another random replacement (RR) method, where questions generated by `GPT-4o` are randomly substituted with irrelevant ones.[4] The number of questions replaced varies from one to all for each article. Finally, we introduce another full random replacement (FRR), where questions generated by `GPT-4o` are all replaced with random irrelevant questions. Throughout these experiments, `Plan-Input` remains the default model.

Figure 4 reveals that plan questions have a direct impact on summary quality (compare `GPT-4o` vs. RR/FRR). Our method exhibits a certain degree of robustness, as it performs reasonably well even when the questions contain some degree of noise (RR vs. FRR). In the FRR condition, model performance degrades substantially, which suggests that relevant plan questions are instrumental in improving summary quality. Although questions generated by `Mistral` and `RAST` lead to lower performance gains compared to `GPT-4o`, these

[3]The performance is better in the former way–on average, the RLsᴜᴍ score is 7.82% higher than the latter across datasets when applied `Mistral` under zero-shot setting.

[4]The prompt can be found in Appendix Figure 13.

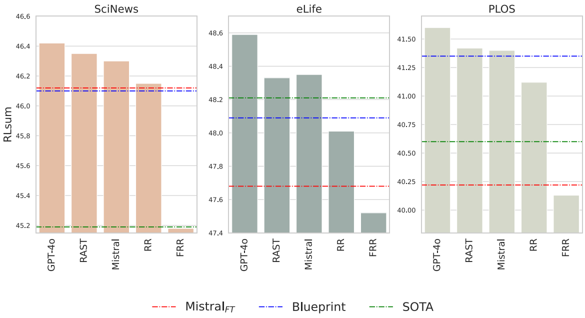

Figure 4: Summary quality as a function of different question generation methods.

Data	Relation	Plan-Output			Plan-Input		
		NoDel	Del	ΔRLSum	NoDel	Del	ΔRLSum
SciNews	Background	4.04	0.98	0.45	4.18	1.07	0.42
	Comparison	2.21	0.21	0.38	2.60	0.33	0.39
	Elaboration	6.47	1.25	1.56	6.39	1.20	1.50
	Explanation	7.09	1.69	1.62	7.88	2.11	1.56
eLife	Background	4.38	1.04	0.27	4.24	1.02	0.24
	Comparison	2.40	0.17	0.13	2.59	0.19	0.15
	Elaboration	5.46	1.16	0.46	5.42	1.15	0.40
	Explanation	6.07	1.72	0.58	6.11	1.81	0.52
PLOS	Background	4.33	0.97	0.20	4.09	0.92	0.22
	Comparison	2.02	0.09	0.11	2.12	0.13	0.13
	Elaboration	5.25	1.13	0.26	5.32	1.15	0.25
	Explanation	6.42	1.52	0.48	6.72	1.49	0.42

Table 4: Proportion of explanatory EDUs in predicted summaries for models with (Del) and without (NoDel) deletions in the plan. In the NoDel setting, the model regulates the proportion of explanatory content in the plan and summary. In the Del setting, the plan is manipulated by removing specific explanations. ΔRLsum = RLsum before plan deletion − RLsum after plan deletion.

open-source models offer a cost-effective alternative. Despite not reaching the highest performance, they still surpass the previous state-of-the-art results.

Impact of Plan Composition An advantage of plan-based models is their capacity to directly control the content of the output summary by simply altering the plan that gives rise to it. In this section, we investigate the extent to which changes in the formation of the plan questions can be observed in the generated summaries.[5] To this end, we conduct a controlled generation experiment. We investigate whether our models can be guided to avoid generating these particular types of explanations by eliminating questions from the plan that correspond to specific types of explanatory EDUs.

Specifically, we remove from the plan questions corresponding to *Background*, *Comparison*, *Elaboration*, and *Explanation* (EDUs are identified by the `DMRST` parser). Table 4 reports the proportion of explanatory EDUs in the output summaries for our two models when no manipulation takes place. These results are compared with summaries generated from plans in which specific types of explanations have been deliberately removed. By deleting plan questions related to certain types of explanations, we evaluate the models' capacity to refrain from generating particular types of EDUs. Results presented in Table 4 confirm that our discourse-driven planning approach is sufficiently expressive to control the explanations and their discourse function in the summary. For both `Plan-Input` and `Plan-Output`, we observe a close alignment between the questions in the plan and the types of explanations produced in the summary.

Amongst explanatory relations, *Elaboration* and *Explanation* emerge as the most significant contributors to overall summary quality. In the `Plan-Input` model, the deletion of *Elaboration* explanations consistently results in lower RLsum across datasets (e.g., a drop from 46.41 to 44.91 in SciNews and from 48.64 to 48.24 in eLife). In contrast, removing *Comparison* explanations has a minor effect on summary quality, as reflected in the relatively stable RLsum scores across datasets (e.g., 41.64 vs. 41.51 in PLOS and 48.64 vs. 48.49 in eLife). In the NoDel condition, we observe that *Elaboration* and *Explanation* relations are more prominent in the summary texts, which suggests that removing them will be more detrimental (e.g., compared to *Background* or *Comparison*).

6 Human Evaluation

To alleviate the limitations of automated metrics and more profoundly analyze the quality of model-generated summaries, we randomly select 5 instances from each of the SciNews, eLife, and PLOS datasets (a total of 15) for human evaluation. The evaluators are postgraduate or doctoral students specializing in Computer Science or Computational Linguistics with advanced proficiency in English. They are compensated at our university's standard hourly rate, and remain blind to the origin of each summary during the evaluation process. In this study, we compare `Mistral`$_{FT}$, `Blueprint`$_{MT}$, `Plan-Input`

[5] We modify the plan generated by the model and evaluate whether our modifications have any bearing on the output.

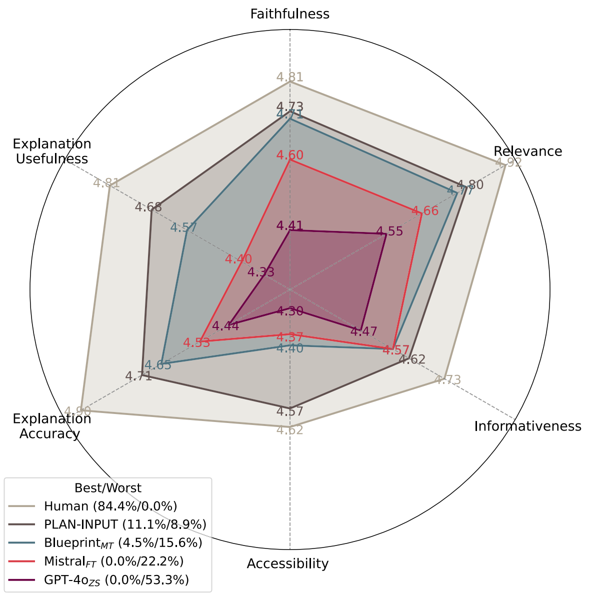

Figure 5: Human evaluation results along different dimensions of summary quality.

(our best performing model), GPT-4o$_{ZS}$, and human-authored reference summaries. Each summary is independently rated by three different evaluators, resulting in a total of 225 ($15 \times 5 \times 3$) evaluation samples.

Our judges rate each summary on a 1 to 5 Likert scale across the dimensions of Faithfulness, Relevance, Informativeness, Accessibility, Explanation Accuracy, and Explanation Usefulness, with higher scores indicating better quality. They are also tasked with ranking the summaries from highest to lowest based on their relative quality within the set of candidates. Raters highlight explanations while reading the summary and the source document. To assess the explanations' accuracy and usefulness, judges are allowed to consult external knowledge sources, such as books, academic papers, and Wikipedia, but not any AI tools. When they are not able to judge the accuracy or usefulness of explanations, raters are asked to skip them and flag them for expert review. We provide detailed evaluation guidelines in Appendix E (Figure 14) and discuss cases flagged for expert review in Appendix F.

Figure 5 reports the performance of each model across all summary samples, as well as the proportion of times a model is rated best (or worst). Fleiss' Kappa scores for Faithfulness ($\kappa = 0.712$), Relevance ($\kappa = 0.835$), Informativeness ($\kappa = 0.673$), Accessibility ($\kappa = 0.707$), Explanation Accuracy ($\kappa = 0.604$), and Explanation Usefulness ($\kappa = 0.633$) indicate substantial inter-rater agreement, with an average of $\kappa = 0.694$. Overall, we observe that human-written summaries outperform all neural summarization models in terms of quality, with particularly pronounced differences in the usefulness and accuracy of explanations. Notably, human-written summaries also perform best in terms of faithfulness.

Among the four neural models under consideration, GPT-4o$_{ZS}$ performs worst, while Mistral$_{FT}$ also obtains low results, with a 22.2% likelihood of being rated as worst. Models based on planning (Blueprint$_{MT}$ and Plan-Input) outperform Mistral$_{FT}$, with Plan-Input being superior across all metrics. On certain criteria, such as accessibility, Plan-Input is on par with human summaries. Compared to other neural summarization systems, it is also more likely to generate high-quality summaries.

7 LLM-as-Judge Evaluation

Due to the considerable length of the source documents and their summaries, it is not feasible to conduct extensive manual evaluations. Therefore, we also use an LLM-based evaluator (Liusie et al., 2024; Zheng et al., 2023; Liu et al., 2025) to perform large-scale comparisons of system outputs. We use the prompt targeting the same dimensions of summary quality adopted in our human evaluation and use GPT-4o as our evaluator (following the hyperparameter settings outlined in Appendix B). To avoid potential bias from previous interactions, we reset the conversation history before each query, making no further adjustments.

Firstly, we validate the agreement between GPT-4o and human ratings using the same set of 15 samples originally employed for human evaluation. We obtain a single rating per sample by averaging the scores of individual participants. We compute Fleiss' Kappa to measure the agreement between GPT-4o scores and the aggregated human ratings across evaluation dimensions. The resulting agreement scores are as follows: Faithfulness ($\kappa = 0.582$), Relevance ($\kappa = 0.643$), Informativeness ($\kappa = 0.633$), Accessibility ($\kappa = 0.624$), Explanation Usefulness ($\kappa = 0.615$), and Explanation Accuracy ($\kappa = 0.597$). We find that human raters and GPT-4o are in substantial agreement. Following this, we expand the evaluation to include all samples in the test sets across our three datasets. We present results on SciNews in

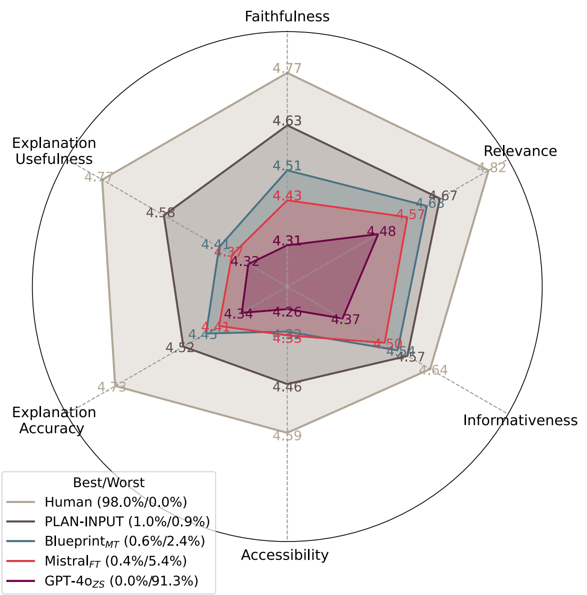

Figure 6: Evaluation with `GPT-4o` on SciNews.

Figure 6 and in the Appendix for the other two datasets (see Figures 15 and 16).

`GPT-4o` assigns the lowest scores to its own answers compared to fine-tuned models. Human-written summaries consistently receive the highest rating and are generally regarded as of the highest quality. In line with our human evaluation, `GPT-4o` also recognizes that `Plan-Input` is better than $\texttt{Blueprint}_{MT}$ and $\texttt{Mistral}_{FT}$. We discuss the statistical significance of our results in Appendix H.

8 Qualitative Analysis

We conduct a qualitative analysis in Appendix I by examining two case studies (see Tables 7 and 8), with a particular focus on comparing outputs generated by humans and machines. Through this analysis, we observe differences in hallucination patterns across different models, and our findings suggest that incorporating structured planning into the generation process can effectively reduce certain types of errors.

Specifically, plan-based models demonstrate improved factual consistency compared to `GPT-4o` and $\texttt{Mistral}_{FT}$ by better organizing information before generation. For instance, when describing how the cerebellum adjusts movements based on feedback (see Table 7), `GPT-4o` states, ‘‘The cerebellum acts as the brain’s command center for motor control.’’ At first glance, this phrasing may seem reasonable, but it inaccurately represents the cerebellum’s role and could lead to misconceptions. Instead of commanding movements, the cerebellum fine-tunes them using feedback from previous actions, adjusting their timing and coordination. When summarizing a document focusing on Musashi-1, a protein that binds to molecules of RNA and helps to promote cell growth during development (see Table 8), $\texttt{Mistral}_{FT}$ incorrectly attributes certain properties to Musashi-1, such as it being ‘‘permanently destroyed Musashi-1’’ by oleic acid. In reality, Musashi-1 is not permanently destroyed but rather inhibited or down-regulated by compounds like oleic acid. These mistakes suggest a tendency to overgeneralize patterns or infer causal relations that are not supported by evidence. In contrast, plan-based models (`Plan-Output` and `Plan-Input`) avoid these types of errors.

Unlike `GPT-4o` and $\texttt{Mistral}_{FT}$, which often produce fragmented or overly general statements, our models demonstrate a better grasp of causal and logical connections within the scientific narrative. For example, in the cerebellum case (see Table 7), our models successfully preserve the logical flow between experimental observations and their implications, while `GPT-4o` distorts causal relations by overgeneralizing. This is because structured planning forces the model to first outline key arguments and causal relations before generating text. This process reduces the risk of introducing spurious causal or conceptual connections, which are particularly important for explanation-oriented text.

Plan-based models are not without limitations; for instance, we occasionally observe misattributions, such as linking Musashi-1 to kidney cancer (see Table 7). However, they offer an improvement over models like GPT-4o in both factual accuracy and overall reliability. In sum, the qualitative analysis underscores the potential of plan-based models to enhance lay summarization quality while also highlighting areas for further refinement.

9 Conclusion

In this paper, we propose a planning-driven method designed for lay summarization. Our approach leverages insights from Rhetorical Structure Theory and the Question Under Discussion framework to create question-based plans tailored to explanatory content. We develop two models that differ in how they integrate planning in the summarization process (plans are part of the source document or the target summary) and investigate their impact on overall performance. Empirical studies demonstrate that our models improve summarization performance compared to vanilla fine-tuning and surpass previous state-of-the-art methods. Furthermore, unlike phrase-based planning models, our analysis confirms the effectiveness of utilizing discourse knowledge for high-level planning, resulting in more factually consistent and higher-quality summaries.

Acknowledgments

This project has received funding from the European Research Council (ERC) under the European Union's Horizon 2020 Research and Innovation Programme (grant agreement no. 948878). We acknowledge the inspiring environment of TRR 248 funded by DFG (German Research Foundation) – project number 389792660. Lapata acknowledges the support of the UK Engineering and Physical Sciences Research Council (grant EP/W002876/1). We are grateful to the anonymous reviewers and action editor (Junyi Jessy Li) for their exceptionally detailed and helpful feedback.

European Research Council
Established by the European Commission

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, Red Avila, Igor Babuschkin, Suchir Balaji, Valerie Balcom, Paul Baltescu, Haiming Bao, Mohammad Bavarian, Jeff Belgum, Irwan Bello, Jake Berdine, Gabriel Bernadett-Shapiro, Christopher Berner, Lenny Bogdonoff, Oleg Boiko, Madelaine Boyd, Anna-Luisa Brakman, Greg Brockman, Tim Brooks, Miles Brundage, Kevin Button, Trevor Cai, Rosie Campbell, Andrew Cann, Brittany Carey, Chelsea Carlson, Rory Carmichael, Brooke Chan, Che Chang, Fotis Chantzis, Derek Chen, Sully Chen, Ruby Chen, Jason Chen, Mark Chen, Ben Chess, Chester Cho, Casey Chu, Hyung Won Chung, Dave Cummings, Jeremiah Currier, Yunxing Dai, Cory Decareaux, Thomas Degry, Noah Deutsch, Damien Deville, Arka Dhar, David Dohan, Steve Dowling, Sheila Dunning, Adrien Ecoffet, Atty Eleti, Tyna Eloundou, David Farhi, Liam Fedus, Niko Felix, Simón Posada Fishman, Juston Forte, Isabella Fulford, Leo Gao, Elie Georges, Christian Gibson, Vik Goel, Tarun Gogineni, Gabriel Goh, Rapha Gontijo-Lopes, Jonathan Gordon, Morgan Grafstein, Scott Gray, Ryan Greene, Joshua Gross, Shixiang Shane Gu, Yufei Guo, Chris Hallacy, Jesse Han, Jeff Harris, Yuchen He, Mike Heaton, Johannes Heidecke, Chris Hesse, Alan Hickey, Wade Hickey, Peter Hoeschele, Brandon Houghton, Kenny Hsu, Shengli Hu, Xin Hu, Joost Huizinga, Shantanu Jain, Shawn Jain, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Tal August, Katharina Reinecke, and Noah A. Smith. 2022. Generating scientific definitions with controllable complexity. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8298–8317, Dublin, Ireland. Association for Computational Linguistics. https://doi.org/10.18653/v1/2022.acl-long.569

Tal August, Lucy Lu Wang, Jonathan Bragg, Marti A. Hearst, Andrew Head, and Kyle Lo. 2023. Paper plain: Making medical research papers approachable to healthcare consumers with natural language processing. *ACM Transactions on Computer-Human Interaction*, 30(5):1–38. https://doi.org/10.1145/3589955

David I. Beaver, Craige Roberts, Mandy Simons, and Judith Tonhauser. 2017. Questions under discussion: Where information structure meets projective content. *Annual Review of Linguistics*, 3(Volume 3, 2017):265–284. https://doi.org/10.1146/annurev-linguistics-011516-033952

Anton Benz and Katja Jasinskaja. 2017. Questions under discussion: From sentence to discourse. *Discourse Processes*, 54(3):177–186. https://doi.org/10.1080/0163853X.2017.1316038

Meng Cao, Yue Dong, and Jackie Cheung. 2022. Hallucinated but factual! Inspecting the factuality of hallucinations in abstractive summarization. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3340–3354, Dublin, Ireland. Association for Computational Linguistics. https://doi.org/10.18653/v1/2022.acl-long.236

Muthu Kumar Chandrasekaran, Guy Feigenblat, Eduard Hovy, Abhilasha Ravichander, Michal Shmueli-Scheuer, and Anita de Waard. 2020. Overview and insights from the shared tasks at scholarly document processing 2020: CL-SciSumm, LaySumm and LongSumm. In *Proceedings of the First Workshop on Scholarly Document Processing*, pages 214–224, Online. Association for Computational Linguistics. https://doi.org/10.18653/v1/2020.sdp-1.24

Jiaao Chen and Diyi Yang. 2021. Structure-aware abstractive conversation summarization via discourse and action graphs. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1380–1391, Online. Association for Computational Linguistics. https://doi.org/10.18653/v1/2021.naacl-main.109

Felipe Costa, Sixun Ouyang, Peter Dolog, and Aonghus Lawlor. 2018. Automatic generation of natural language explanations. In *Companion Proceedings of the 23rd International Conference on Intelligent User Interfaces*, pages 1–2. https://doi.org/10.1145/3180308.3180366

Aldan Creo, Manuel Lama, and Juan C. Vidal. 2023. Prompting llms with content plans to enhance the summarization of scientific articles.

Kordula De Kuthy, Madeeswaran Kannan, Haemanth Santhi Ponnusamy, and Detmar Meurers. 2020. Towards automatically generating questions under discussion to link information and discourse structure. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 5786–5798, Barcelona, Spain (Online). International Committee on Computational Linguistics. https://doi.org/10.18653/v1/2020.coling-main.509

Yue Dong, Andrei Mircea, and Jackie Chi Kit Cheung. 2021. Discourse-aware unsupervised summarization for long scientific documents. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 1089–1102, Online. Association for Computational Linguistics. https://doi.org/10.18653/v1/2021.eacl-main.93

Raymond Fok, Joseph Chee Chang, Tal August, Amy X. Zhang, and Daniel S. Weld. 2024. Qlarify: Recursively expandable abstracts for dynamic information retrieval over scientific papers. In *Proceedings of the 37th Annual ACM Symposium on User Interface Software and Technology*, UIST '24, New York, NY, USA. Association for Computing Machinery. https://doi.org/10.1145/3654777.3676397

Tomas Goldsack, Zhihao Zhang, Chenghua Lin, and Carolina Scarton. 2022. Making science simple: Corpora for the lay summarisation of scientific literature. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 10589–10604, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics. https://doi.org/10.18653/v1/2022.emnlp-main.724

Tomas Goldsack, Zhihao Zhang, Chen Tang, Carolina Scarton, and Chenghua Lin. 2023. Enhancing biomedical lay summarisation with external knowledge graphs. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 8016–8032, Singapore. Association for Computational Linguistics. https://doi.org/10.18653/v1/2023.emnlp-main.498

Qi Gou, Zehua Xia, Bowen Yu, Haiyang Yu, Fei Huang, Yongbin Li, and Nguyen Cam-Tu. 2023. Diversify question generation with retrieval-augmented style transfer. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 1677–1690, Singapore.

Association for Computational Linguistics. https://doi.org/10.18653/v1/2023.emnlp-main.104

Bin Guo, Hao Wang, Yasan Ding, Wei Wu, Shaoyang Hao, Yueqi Sun, and Zhiwen Yu. 2021a. Conditional text generation for harmonious human-machine interaction. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 12(2):1–50. https://doi.org/10.1145/3439816

Yue Guo, Wei Qiu, Yizhong Wang, and Trevor Cohen. 2021b. Automated lay language summarization of biomedical scientific reviews. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(1):160–168. https://doi.org/10.1609/aaai.v35i1.16089

Grigorii Guz and Giuseppe Carenini. 2020. Coreference for discourse parsing: A neural approach. In *Proceedings of the First Workshop on Computational Approaches to Discourse*, pages 160–167, Online. Association for Computational Linguistics. https://doi.org/10.18653/v1/2020.codi-1.17

HyoJung Han, Jordan Boyd-Graber, and Marine Carpuat. 2023. Bridging background knowledge gaps in translation with automatic explicitation. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 9718–9735, Singapore. Association for Computational Linguistics. https://doi.org/10.18653/v1/2023.emnlp-main.603

Freya Hewett, Hadi Asghari, and Manfred Stede. 2024. Elaborative simplification for German-language texts. In *Proceedings of the 25th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 29–39, Kyoto, Japan. Association for Computational Linguistics. https://doi.org/10.18653/v1/2024.sigdial-1.3

Fantine Huot, Joshua Maynez, Chris Alberti, Reinald Kim Amplayo, Priyanka Agrawal, Constanza Fierro, Shashi Narayan, and Mirella Lapata. 2024. μPLAN: Summarizing using a content plan as cross-lingual bridge. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2146–2163, St. Julian's, Malta. Association for Computational Linguistics. https://doi.org/10.18653/v1/2024.eacl-long.131

Fantine Huot, Joshua Maynez, Shashi Narayan, Reinald Kim Amplayo, Kuzman Ganchev, Annie Priyadarshini Louis, Anders Sandholm, Dipanjan Das, and Mirella Lapata. 2023. Text-blueprint: An interactive platform for plan-based conditional generation. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations*, pages 105–116, Dubrovnik, Croatia. Association for Computational Linguistics. https://doi.org/10.18653/v1/2023.eacl-demo.13

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. Mistral 7b.

Dongwhan Kim and Joonhwan Lee. 2019. Designing an algorithm-driven text generation system for personalized and interactive news reading. *International Journal of Human–Computer Interaction*, 35(2):109–122. https://doi.org/10.1080/10447318.2018.1437864

J. Peter Kincaid, Robert P. Fishburne Jr, Richard L. Rogers, and Brad S. Chissom. 1975. Derivation of new readability formulas (automated readability index, fog count and flesch reading ease formula) for navy enlisted personnel. https://doi.org/10.21236/ADA006655

Wei-Jen Ko, Yating Wu, Cutter Dalton, Dananjay Srinivas, Greg Durrett, and Junyi Jessy Li. 2023. Discourse analysis via questions and answers: Parsing dependency structures of questions under discussion. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 11181–11195, Toronto, Canada. Association for Computational Linguistics. https://doi.org/10.18653/v1/2023.findings-acl.710

Philippe Laban, Tobias Schnabel, Paul N. Bennett, and Marti A. Hearst. 2022. SummaC: Re-visiting NLI-based models for inconsistency detection in summarization. *Transactions of the Association for Computational Linguistics*, 10:163–177. https://doi.org/10.1162/tacl_a_00453

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Dongqi Liu and Vera Demberg. 2023. ChatGPT vs human-authored text: Insights into controllable text summarization and sentence style transfer. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 4: Student Research Workshop)*, pages 1–18, Toronto, Canada. Association for Computational Linguistics. https://doi.org/10.18653/v1/2023.acl-srw.1

Dongqi Liu and Vera Demberg. 2024. RST-LoRA: A discourse-aware low-rank adaptation for long document abstractive summarization. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 2200–2220, Mexico City, Mexico. Association for Computational Linguistics. https://doi.org/10.18653/v1/2024.naacl-long.121

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, Gyeongju, Republic of Korea. Association for Computational Linguistics.

Dongqi Liu, Yifan Wang, and Vera Demberg. 2023. Incorporating distributions of discourse structure for long document abstractive summarization. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5574–5590, Toronto, Canada. Association for Computational Linguistics. https://doi.org/10.18653/v1/2023.acl-long.306

Dongqi Liu, Yifan Wang, Jia Loy, and Vera Demberg. 2024. SciNews: From scholarly complexities to public narratives – a dataset for scientific news report generation. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 14429–14444, Torino, Italia. ELRA and ICCL.

Dongqi Liu, Chenxi Whitehouse, Xi Yu, Louis Mahon, Rohit Saxena, Zheng Zhao, Yifu Qiu, Mirella Lapata, and Vera Demberg. 2025. What is that talk about? A video-to-text summarization dataset for scientific presentations.

Zhengyuan Liu and Nancy Chen. 2021. Controllable neural dialogue summarization with personal named entity planning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 92–106, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics. https://doi.org/10.18653/v1/2021.emnlp-main.8

Zhengyuan Liu, Ke Shi, and Nancy Chen. 2020. Multilingual neural RST discourse parsing. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 6730–6738, Barcelona, Spain (Online). International Committee on Computational Linguistics. https://doi.org/10.18653/v1/2020.coling-main.591

Zhengyuan Liu, Ke Shi, and Nancy Chen. 2021. DMRST: A joint framework for document-level multilingual RST discourse segmentation and parsing. In *Proceedings of the 2nd Workshop on Computational Approaches to Discourse*, pages 154–164, Punta Cana, Dominican Republic and Online. Association for Computational Linguistics. https://doi.org/10.18653/v1/2021.codi-main.15

Adian Liusie, Potsawee Manakul, and Mark Gales. 2024. LLM comparative assessment: Zero-shot NLG evaluation through pairwise comparisons using large language models. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 139–151, St. Julian's, Malta. Association for Computational Linguistics. https://doi.org/10.18653/v1/2024.eacl-long.8

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In

International Conference on Learning Representations.

Annie Louis, Aravind Joshi, and Ani Nenkova. 2010. Discourse indicators for content selection in summarization. In *Proceedings of the SIGDIAL 2010 Conference*, pages 147–156, Tokyo, Japan. Association for Computational Linguistics.

Zheheng Luo, Qianqian Xie, and Sophia Ananiadou. 2022. Readability controllable biomedical document summarization. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 4667–4680, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Aru Maekawa, Tsutomu Hirao, Hidetaka Kamigaito, and Manabu Okumura. 2024. Can we obtain significant success in RST discourse parsing by using large language models? In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2803–2815, St. Julian's, Malta. Association for Computational Linguistics. https://doi.org/10.18653/v1/2024.eacl-long.171

William C. Mann and Sandra A. Thompson. 1987. *Rhetorical Structure Theory: A Theory of Text Organization*. University of Southern California, Information Sciences Institute Los Angeles.

Daniel Marcu. 1997. From discourse structures to text summaries. In *Intelligent Scalable Text Summarization*.

Daniel Marcu. 1999. Discourse trees are good indicators of importance in text. *Advances in Automatic Text Summarization*, 293:123–136.

Daniel Marcu. 2000. *The Theory and Practice of Discourse Parsing and Summarization*. MIT Press. https://doi.org/10.7551/mitpress/6754.001.0001

Amit Moryossef, Mathias Müller, Anne Göhring, Zifan Jiang, Yoav Goldberg, and Sarah Ebling. 2023. An open-source gloss-based baseline for spoken to signed language translation. In *Proceedings of the Second International Workshop on Automatic Translation for Signed and Spoken Languages*, pages 22–33, Tampere, Finland. European Association for Machine Translation.

Shashi Narayan, Joshua Maynez, Reinald Kim Amplayo, Kuzman Ganchev, Annie Louis, Fantine Huot, Anders Sandholm, Dipanjan Das, and Mirella Lapata. 2023. Conditional generation with a question-answering blueprint. *Transactions of the Association for Computational Linguistics*, 11:974–996. https://doi.org/10.1162/tacl_a_00583

Shashi Narayan, Yao Zhao, Joshua Maynez, Gonçalo Simões, Vitaly Nikolaev, and Ryan McDonald. 2021. Planning with learned entity prompts for abstractive summarization. *Transactions of the Association for Computational Linguistics*, 9:1475–1492. https://doi.org/10.1162/tacl_a_00438

Annishka Oksa, Slava Kalyuga, and Paul Chandler. 2010. Expertise reversal effect in using explanatory notes for readers of shakespearean text. *Instructional Science*, 38:217–236. https://doi.org/10.1007/s11251-009-9109-6

Ratish Puduppully, Li Dong, and Mirella Lapata. 2019. Data-to-text generation with content selection and planning. *Proceedings of the AAAI Conference on Artificial Intelligence*, 33(01):6908–6915. https://doi.org/10.1609/aaai.v33i01.33016908

Craige Roberts. 2012. Information structure: Towards an integrated formal theory of pragmatics. *Semantics and Pragmatics*, 5:6–1. https://doi.org/10.3765/sp.5.6

Yixiao Song, Yekyung Kim, and Mohit Iyyer. 2024. VeriScore: Evaluating the factuality of verifiable claims in long-form text generation. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 9447–9474, Miami, Florida, USA. Association for Computational Linguistics. https://doi.org/10.18653/v1/2024.findings-emnlp.552

Neha Srikanth and Junyi Jessy Li. 2021. Elaborative simplification: Content addition and explanation generation in text simplification. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 5123–5137, Online. Association for Computational Linguistics. https://doi.org/10.18653/v1/2021.findings-acl.455

Manfred Stede, Maite Taboada, and Debopam Das. 2017. Annotation guidelines for rhetorical structure. Manuscript. *University of Potsdam and Simon Fraser University*.

Renliang Sun, Hanqi Jin, and Xiaojun Wan. 2021. Document-level text simplification: Dataset, criteria and baseline. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7997–8013, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics. https://doi.org/10.18653/v1/2021.emnlp-main.630

Chen Tang, Shun Wang, Tomas Goldsack, and Chenghua Lin. 2023. Improving biomedical abstractive summarisation with knowledge aggregation from citation papers. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 606–618, Singapore. Association for Computational Linguistics. https://doi.org/10.18653/v1/2023.emnlp-main.40

Haining Wang, Jason Clark, Hannah McKelvey, Leila Sterman, Zheng Gao, Zuoyu Tian, Sandra Kübler, and Xiaozhong Liu. 2025. Science out of its ivory tower: Improving accessibility with reinforcement learning. https://doi.org/10.47989/ir30iConf47530

Yating Wu, Ritika Mangla, Greg Durrett, and Junyi Jessy Li. 2023a. QUDeval: The evaluation of questions under discussion discourse parsing. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 5344–5363, Singapore. Association for Computational Linguistics. https://doi.org/10.18653/v1/2023.emnlp-main.325

Yating Wu, William Sheffield, Kyle Mahowald, and Junyi Jessy Li. 2023b. Elaborative simplification as implicit questions under discussion. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 5525–5537, Singapore. Association for Computational Linguistics. https://doi.org/10.18653/v1/2023.emnlp-main.336

Jiacheng Xu, Zhe Gan, Yu Cheng, and Jingjing Liu. 2020. Discourse-aware neural extractive text summarization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5021–5031, Online. Association for Computational Linguistics. https://doi.org/10.18653/v1/2020.acl-main.451

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 2020, Addis Ababa, Ethiopia, April 26–30, 2020*.

Zhihao Zhang, Tomas Goldsack, Carolina Scarton, and Chenghua Lin. 2024. ATLAS: Improving lay summarisation with attribute-based control. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 337–345, Bangkok, Thailand. Association for Computational Linguistics. https://doi.org/10.18653/v1/2024.acl-short.32

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. 2023. Judging LLM-as-a-judge with MT-bench and chatbot arena. In *Thirty-seventh Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.

A RST Labels

We will now explain in detail the semantics of the discourse relations listed in Table 5, and how they can serve as explanations. For instance, the *Background* relation is used to set the context for an event, such as ‘‘The storm caused massive flooding. As a result, the government declared a state of emergency.’’ Here, the background information helps to explain why a state of emergency is declared. Similarly, the *Elaboration* relation provides additional information to clarify a statement, as in ‘‘The new software version is more user-friendly. It includes a redesigned interface and improved performance,’’ where the additional details elaborate on why the software is considered user-friendly. Other elaboration types include general-specific, part-whole, process-step, object-attribute, set-member, and example, each providing specific details to support and explain a broader statement. For example, ‘‘Many animals hibernate during winter. For example, bears often

RST TYPE	RST LABEL
Background	Background, Circumstance
Elaboration	Elaboration-additional, Elaboration-general-specific, Elaboration-part-whole, Elaboration-process-step, Elaboration-object-attribute, Elaboration-set-member, Example, Definition
Explanation	Evidence, Explanation-argumentative, Reason
Comparison	Comparison, Preference, Analogy, Proportion and Topic-Comment

Table 5: RST explanatory relations.

find caves or dig dens to sleep through the cold month'' elaborates on a general statement with a specific example.

Explanation relations, such as *Evidence* and *Reason*, offer support or justification for a statement. For instance, ''Many species are endangered due to habitat destruction. Studies have shown that deforestation has led to significant population decline.'' uses evidence to support the claim about endangered species. Similarly, ''She missed the meeting because her car broke down'' provides a reason to explain her absence. *Comparison* relations like *Preference* and *Analogy*, help explain by highlighting similarities or differences, as in ''A computer's memory works like a human brain, storing information for later use,'' where the analogy clarifies the concept of computer memory by comparing it to a familiar idea.

B Hyperparameter Settings

For all fine-tuning experiments, we utilize the AdamW optimizer (Loshchilov and Hutter, 2019) with $\beta_1 = 0.9$, $\beta_2 = 0.999$, $\epsilon = 10^{-9}$, and weight decay $= 0.1$, along with a warm-up ratio of 0.15. The initial learning rate is set to 5e-5, employing cosine learning rate scheduling. We set the random seed to 2,024 and the dropout rate to 0.1. For SciNews, we set the maximum number of new generation tokens to 1,024, for eLife to 512, and for PLOS to 256. All other parameters follow the default settings of the `Transformers` library. During training, we save the checkpoint with the highest Rouge-2 F1 on the validation set as the final checkpoint. All experiments run for 15 epochs with a batch size of 8, and early stopping is implemented to prevent overfitting, with all models converging before reaching 15 epochs. During inference, we employ beam search with size 4, length penalty of 3.0, and set the no-repeat n-gram size to 3.

For `GPT-4o`,[6] we use version *gpt-4o-2024-05-13* (our experiments took place May 1, 2024, to March 31, 2025) with the following hyperparameter settings: We set temperature to 1, top_p to 1, frequency penalty to 0.2, and presence penalty to 0.2. All other hyperparameters follow OpenAI's default values.

C Rule-based Extraction of Explanations

We develop a set of rules aimed at identifying words and phrases that indicate four types of explanations, which are in turn based on RST relations:

- **Background** includes *Background* and *Circumstance* relations, signalled by words such as ''historically'' and ''traditionally.''
- **Elaboration** covers various *Elaboration* relations (e.g., *Additional*, *General-Specific*, *Part-Whole*, *Process-Step*, *Object-Attribute*, and *Set-Member*), as well as *Example* and *Definition* relations, signalled by words like ''defined as'' and ''for example.''
- **Explanation** includes *Evidence*, *Explanation-Argumentative*, and *Reason* relations, signalled by words such as ''because'' and ''due to.''
- **Comparison** includes *Comparison*, *Preference*, *Analogy*, and *Proportion* relations, signalled by words like ''compared to'' and ''similarly.''

Table 6 presents the words and phrases associated with each of these relations. To identify these indicators within the text, we compile regular expression patterns tailored to each category. Explanatory content is collected on a sentence-by-sentence basis: If a signal word is detected within a sentence, that entire sentence is categorized accordingly. In constructing these patterns, we use the caret symbol ^ before certain words to restrict their occurrence to

[6] `https://platform.openai.com/docs/models/`.

<table>
<tr><th>Category</th><th>Signal Words and Phrases</th></tr>
<tr><td>Background</td><td>^historically, ^traditionally, ^previously, ^in the past, ^before, ^initially, ^once, ^earlier, ^in the beginning, ^at first, ^prior to, ^originally, ^at the outset, ^at the time, ^long ago, ^decades ago, ^in former times, ^previously mentioned, ^the history of, ^the origin of, ^in earlier times, ^from the outset, ^in the early days, ^over the years, ^long before, ^centuries ago, ^during the early stages, ^at that time, ^back then, ^once upon a time, ^throughout history, ^previously established, ^over the course of history, ^in ancient times</td></tr>
<tr><td>Comparison</td><td>compared to, compared with, ^similarly, likewise, in contrast, in comparison, in opposition, ^on the contrary, ^on one hand, ^on the other hand, ^conversely, rather than, different from, ^unlike, similar to, analogous to, contrary to, in contradistinction, distinct from, distinguishable from, as opposed to, in the same way, by comparison, comparable to, <code>differ(?:s|ed|ing)</code> from, diverg(e|es|ed|ing)? from, in a similar manner, in the same vein, on the flip side, correspondingly, on a different note, in opposition to, different from, in a contrasting way, in an analogous way</td></tr>
<tr><td>Elaboration</td><td>defined as, refer(?:s|red|ring)? to, mean(?:s|t)?, known as, definition, ^in other words, ^that is to say, ^that’s to say, ^this is to say, ^that means, ^this means, ^this implies, ^that implies, i.e., e.g., for example, for instance, such as, ^to clarify, ^to explain, whereas, ^to illustrate, ^to elaborate, ^specifically, ^particularly, in particular, as an example, by way of example, more precisely, ^to be specific, ^to exemplify, namely, by way of illustration, expounded upon, in more detail, one example, an example, ^to add to this</td></tr>
<tr><td>Explanation</td><td>because, due to, since, thanks to, owing to, for the sake of, stemming from, given that, in light of, for this reason, for that reason, for the reason that, for the purpose of, for this cause, the reason is, the reasons are, as a result, consequently, as a consequence, accordingly, with the result that, so that, such that, result(?:s|ed|ing)? in, , result(?:s|ed|ing)? from, lead(?:s|ed|ing)? to, which means, thereby, whereby, in consequence of, on account of, so as to, on the grounds that</td></tr>
</table>

Table 6: Words and phrases used to heuristically identify discourse relations signaling explanations.

sentence-initial positions. This method ensures that words like ‘‘historically’’ or ‘‘similarly’’ are recognized only when they function as introductory elements, which is common in explanations. Additionally, we utilize regular expressions such as `differ(?|ed|ing)` to capture different word forms. Once each explanatory sentence is identified, we then select the preceding sentence as its target sentence.

D Model Prompts

In this section, we present the various prompts used in our work.

Question Generation Prompt

Generate a question that is coherent and contextually relevant to the provided context and target sentence.
Context: {Context Text}
Target: {Target Sentence}
Question Sentence:

Figure 7: Prompt used by GPT-4o and Mistral to generate plan questions based on target sentences and their context.

Summary Generation Prompt

Generate a lay summary for the provided document.
Document: {Document Text}
Lay Summary:

Figure 8: Prompt used by $\texttt{Mistral}_{ZS}$, $\texttt{GPT-4o}_{ZS}$ and $\texttt{Mistral}_{FT}$ to generate summaries.

Summary Generation Prompt (ICL)

Document: {Document Text}
Lay Summary: {Summary Text}
Using the example above as a reference for structure and tone, generate a new lay summary for the following document. Ensure the summary is original and does not replicate phrases or content from the example.
Document: {Document Text}
Lay Summary:

Figure 9: Prompt used by $\texttt{Mistral}_{ICL}$ and $\texttt{GPT-4o}_{ICL}$ to generate summaries.

Named Entity Identification

Identify all named entities in the following text and return them as a list.
Document: {Document Text}
Return Format: [Entity 1, 2, 3]
List all identified entities in the order they appear in the text.
Identified Entities:

Figure 10: Prompt used by GPT-4o to identify named entities for Blueprint model.

Plan-based Summary Generation Prompt

Generate a lay summary for the following document based on the plan questions.
Document: {Document Text}
Planning Questions: {Questions}
Ensure that the generated summary sequentially answers the plan questions.
Lay Summary:

Figure 11: Prompt used by Plan-Input and Plan-Output models.

Direct Extraction of Explanatory Content

Parse the provided document to extract all explanatory sentences and the corresponding target sentences they explain. Return the parsing results as a list of dictionaries formatted as follows:

```
[
  {
    explanatory_sentence: 'XXX',
    target_sentence: 'XXX'
  },
  {
    explanatory_sentence: 'XXX',
    target_sentence: 'XXX'
  }
]
```

Ensure that each dictionary accurately pairs each explanatory statement with its respective target sentence.
Document: {Document Text}

Figure 12: Prompt used by GPT-4o and Mistral to directly extract explanatory content.

Irrelevant Question Generation

Randomly generate a question with a question mark.
Question Sentence:

Figure 13: Prompt used by GPT-4o to generate irrelevant questions.

E Human Evaluation Guidelines

Prerequisites Eligibility for this evaluation requires simultaneous fulfillment of two conditions: (1) being a Master's or Ph.D. student in Computer Science or Computational Linguistics, and (2) demonstrating greater than or equal to C2 English proficiency[a]. If you do not meet both criteria, we respectfully ask you to refrain from participating in this task. Those who qualify are encouraged to proceed and follow the instructions below.

Instructions Below is a detailed explanation of the metrics and evaluation criteria for our human evaluation process. Please carefully read the provided document along with the candidate summaries. After thoroughly examining each summary, evaluate them based on the following six criteria using a Likert scale from 1 to 5, with higher scores reflecting better quality:

- **Faithfulness:** This metric evaluates how accurately the summary reflects the information in the source document. A faithful summary should strictly adhere to the source material, avoiding any contradictions or unverified details.
- **Relevance:** This metric assesses how accurately the summary content reflects the topics covered in the source text. A relevant summary should include topics that are pertinent to the source document.
- **Informativeness:** This metric measures how well the summary conveys the key points and essential details from the source text. An informative summary should capture the main ideas, providing a clear and precise understanding of the source document's arguments and findings.
- **Accessibility:** This metric evaluates how easy the summary is to read and understand. An accessible summary should be well-structured and written in clear language. It should avoid unnecessary complexity and ensure that readers can follow the content without difficulty.
- **Explanation Accuracy:** This metric measures how factually correct the explanatory content is. Explanation accuracy requires that the information provided in the explanatory sentence is verifiable and aligns with established knowledge in the relevant field, avoiding any misleading or incorrect explanations.
- **Explanation Usefulness:** This metric assesses how helpful the explanatory content is in enhancing the reader's understanding of the subject matter. Explanation usefulness ensures that the content contributes to the reader's comprehension of complex ideas, offering valuable insights and clarifications.

Please note that you are required to highlight all explanatory content found in the summary before assigning evaluation scores. You are permitted to use external knowledge sources, such as books, academic papers, and Wikipedia, but you are not allowed to use any AI tools to assist in your judgment. If you encounter any specific explanatory content that you are unable to evaluate for accuracy or usefulness, you should skip that content and mark it (meaning it will not factor into your scoring, but you will still need to score based on the rest of the content). This will allow us to bring in external experts (PhD holders in the relevant field) to assess it.

Rating System For each criterion, use the Likert scale as follows:

- 1 (Worst): Very poor quality, does not meet the criteria at all.
- 2 (Poor): Subpar quality, meets the criteria to a minimal extent.
- 3 (Fair): Average quality, adequately meets the criteria.
- 4 (Good): Above average quality, meets the criteria well.
- 5 (Best): Excellent quality, fully meets the criteria.

Overall Ranking After rating the summaries based on the six criteria, you are also expected to rank the candidates from best to worst based on overall quality. Consider how well each summary performs across all criteria to determine the final ranking.

[a] https://en.wikipedia.org/wiki/C2_Proficiency

Figure 14: A snapshot of the experimental instructions seen by our human raters.

F Human Expert Evaluation Results

Our human evaluators identify six examples where they could not make a decision on the accuracy and usefulness of explanations. Three are generated by GPT-4o$_{ZS}$, two by Mistral$_{FT}$, and one by Blueprint$_{MT}$, covering fields such as Biology, Neuroscience, and Materials Science. To further assess these cases, we enlist experts with a Ph.D. degree in the relevant areas. Using the same guidelines, GPT-4o receives an average rating of 2.33 for explanation accuracy and 2.67 for explanation usefulness. For Mistral$_{FT}$, the average rating is 3.50 for both accuracy and usefulness, while Blueprint$_{MT}$ receives a rating of 4.00 for accuracy and 2.00 for usefulness. These expert evaluations align with the initial human assessment, reinforcing the consistency of the findings across the highlighted explanations.

G GPT-4o Evaluation Results

Figure 15 shows automatic evaluation results on the eLife dataset using GPT-4o as the judge. Results on the PLOS dataset are summarized in Figure 16, again using GPT-4o as the evaluator.

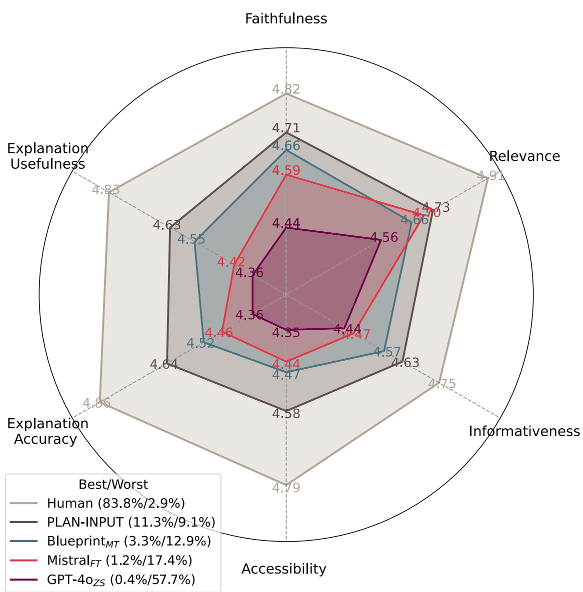

Figure 15: Evaluation results (dimensions of Faithfulness, Relevance, Informativeness, Accessibility, Explanation Accuracy, and Explanation Usefulness) on eLife using GPT-4o as a judge.

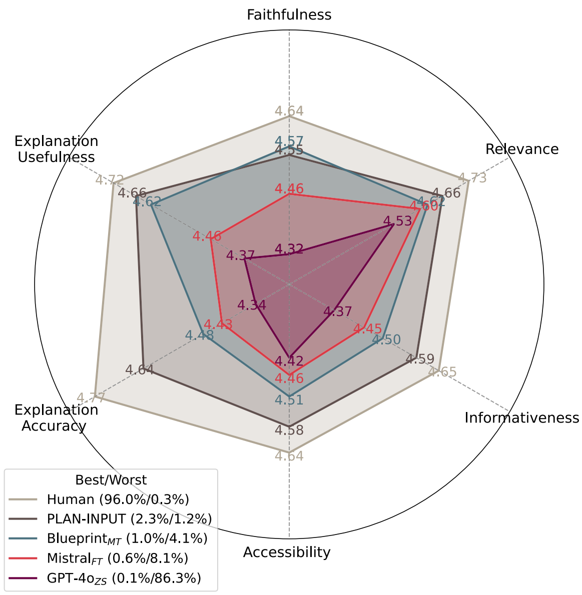

Figure 16: Evaluation results (dimensions of Faithfulness, Relevance, Informativeness, Accessibility, Explanation Accuracy, and Explanation Usefulness) on PLOS using GPT-4o as a judge.

H Significance Testing

We use paired t-tests to analyze the results of our human evaluation and automatic evaluation (LLM-as-Judge with GPT-4o). In the human evaluation, we find that human responses are perceived as significantly better against all baseline models across metrics; with regard to Faithfulness and Accessibility, differences against Plan-Input and Blueprint$_{MT}$ are not statistically significant. Plan-Input is significantly better in Explanation Usefulness and Accessibility compared to all neural network models (but not human answers).

In the LLM-as-Judge evaluation, human summaries outperform all neural models across all datasets and metrics with statistical significance. Plan-Input is significantly better over other neural network models on the SciNews dataset across all metrics except Relevance and Informativeness. On the eLife dataset, Plan-Input outperforms other neural network models on all metrics except Faithfulness, Explanation Usefulness, and Relevance. On the PLOS dataset, Plan-Input significantly outperforms competitor models in all metrics except Informativeness, Faithfulness, and Relevance.

These results are derived with a significance level of $p < 0.05$.

I Case Study

In this case study, we compare human-written lay summaries with those generated by neural models (see Tables 7 and 8). Human summaries go beyond simply listing experimental results; they provide well-structured interpretations of the research background, methods, and key findings, whereas neural models often generate content that appears *plausible but contains inaccuracies*.

Model	Output
Human Reference	The cerebellum utilizes proprioceptive feedback to fine-tune the timing of movements in a sequence based on previous actions. Imagine the cerebellum as a coach who watches how you perform a move, then gives tips to improve the next one based on what was seen. But how exactly does it achieve this? To investigate, we trained rabbits to blink in response to an external cue and explored whether the cerebellum could use feedback from one blink to trigger the next. As expected, after learning the initial blink, the rabbits blinked again in response to their own first blink, creating a chain of movements. Control experiments confirmed that each blink was initiated by the previous one rather than the original cue. Consistent patterns of brain activity during this process indicate that the cerebellum adjusts movement based on feedback from previous actions. Building on this, we trained rabbits to blink on cue, and they learned to initiate additional blinks in response to earlier blinks in the sequence. We further found that the rabbits could use a blink from one eye as a cue to trigger a blink in the other eye, suggesting that the same mechanism governs these movements. This raises the possibility that the cerebellum might also guide sequences of cortical activity during cognitive tasks, given its extensive connections to the cortex, a question future experiments should explore.
`GPT-4o`	The cerebellum acts as the brain's command center for motor control, directly issuing movement instructions and ensuring precise timing. To test this, scientists trained rabbits to blink in response to a vibration signal. Once trained, the rabbits began blinking in a continuous rhythm, even without an external cue, indicating that the cerebellum could independently generate movement sequences. Interestingly, when the researchers covered one of the rabbit's eyes, the uncovered eye still continued blinking, suggesting that the cerebellum maintains a direct, hardwired link between the two eyes. These findings demonstrate that the cerebellum is not only responsible for movement execution but may also function as a memory hub for motor programs, completely independent of the cerebral cortex.
`Mistral`$_{FT}$	The cerebellum helps coordinate movements by adjusting actions based on prior attempts. To study this, scientists trained rabbits to blink in response to a sound. Over time, the rabbits developed a habit of blinking twice in succession, suggesting that the cerebellum is responsible for generating repetitive movements. Further experiments showed that rabbits could use one blink as a cue for another, even switching between eyes. This suggests that the cerebellum not only controls basic motor sequences but also plays a role in synchronizing different body movements. It is even possible that the cerebellum helps regulate general cognitive functions, though more research is needed.
`Plan-Output`	q1: What can the role of the cerebellum in motor control be compared to? q2: How do rabbits form a sequence of consecutive blinks after training? q3: What does it suggest that rabbits can use a blink from one eye to trigger a blink in the other eye? The cerebellum acts like a coach, refining movements based on past experiences. To explore this, scientists trained rabbits to blink when they heard a tone. Over time, the rabbits learned to initiate a second blink in response to their own first blink, forming a sequence of movements. This suggests that the cerebellum uses sensory feedback to fine-tune motor control. Furthermore, the rabbits could blink one eye and then trigger a blink in the other eye, indicating that the cerebellum directly controls bilateral facial coordination. These findings highlight the cerebellum's essential role in sequencing motor actions and suggest that it may also contribute to higher-level cognitive processes.
`Plan-Input`	q1: What kind of role does the cerebellum play in motor control? q2: How do rabbits trigger a sequence of blinks from an initial blink after training? q3: What does it reveal about motor control mechanisms that rabbits can use a blink from one eye to trigger a blink in the other eye? The cerebellum functions as a predictive system that learns movement patterns and automatically sequences them. Researchers trained rabbits to blink when given an air puff. After sufficient training, the rabbits began to produce consecutive blinks, with each blink triggering the next. This suggests that the cerebellum can store movement chains and replay them when needed. Additionally, the researchers found that a blink in one eye could lead to a blink in the opposite eye, implying that the cerebellum might have specialized circuits dedicated to synchronizing both sides of the body. These insights open the possibility that the cerebellum also orchestrates sequences of thoughts and decisions, a topic for future research.

Table 7: Case study 1: We use red font to mark hallucinated information. This sample comes from the test split of the eLife dataset (Goldsack et al., 2022).

Model	Output
Human Reference	When an embryo is developing, stem cells must divide and develop into many specialized types of cells. However, if cell division doesn't stop, or if it restarts later in life, it can cause tumors to form. Musashi-1 is a protein that binds to molecules of RNA and helps to promote cell growth during development: mice that lack this protein have serious brain defects and die shortly after birth. Musashi-1 is usually turned off in adult cells that are not dividing. Sometimes, however, it remains active and contributes to the growth of cancers in the brain and the gut. Reducing Musashi-1 levels in colon tumors slows their growth and causes the cancer cells to die. To find a compound that would switch off Musashi-1, Clingman et al. screened more than 30,000 compounds and identified four inhibitors. One of these was oleic acid, a fatty acid that is found in olive oil and other animal and plant oils. Oleic acid interacts with Musashi-1 in a way that changes the shape of the protein. These changes mean that Musashi-1 is no longer able to regulate the genes that control cell proliferation. Clingman et al. also found that Musashi-1 promotes the activity of a particular enzyme that makes fatty acids; molecules that are needed in large quantities when cells are dividing. Musashi-1 appears to act as a nutrient sensor, turning down the activity of this enzyme in cells when levels of oleic acid are high, and turning up enzyme activity when oleic acid levels are low. The findings of Clingman et al. further reveal how our diets can affect gene expression, and may aid the development of new treatments against cancer.
GPT-4o	Musashi-1 is a protein that plays a crucial role in early development by directly binding to DNA and activating genes responsible for cell proliferation. In mice that lack Musashi-1, the protein is compensated by other molecular pathways, allowing them to develop normally with only minor cognitive impairments. To explore ways to inhibit Musashi-1, Clingman et al. analyzed a vast library of nearly 135,000 chemical compounds. They identified four potential inhibitors, including oleic acid, a fatty acid mainly found in red meat and dairy products. This compound works by permanently destroying Musashi-1, preventing it from affecting cell division. In cancer studies, researchers found that reducing Musashi-1 levels in colon tumors leads to uncontrolled tumor growth and increased metastasis. These findings suggest that Musashi-1 may function as a critical tumor suppressor, and future therapies should aim to enhance its activity in cancer patients rather than inhibit it.
Mistral$_{FT}$	Musashi-1 is a protein that regulates cell division by binding to RNA. It is active during embryonic development but usually turns off in adults. However, it can remain active and contribute to tumors in the brain and lungs. To inhibit Musashi-1, researchers screened over 30,000 compounds and found oleic acid, a substance present in coconut oil and dairy products. Oleic acid interacts with Musashi-1 and permanently destroys it, stopping it from regulating genes linked to cell growth. Additionally, high oleic acid levels completely shut down metabolism in cells, showing how diet influences gene regulation and cancer treatment.
Plan-Output	q1: What is the role of Musashi-1 in cell development and cancer growth? q2: How does oleic acid affect Musashi-1 and its function in cell proliferation? q3: How does Musashi-1 function as a nutrient sensor in relation to fatty acid synthesis? Musashi-1 is a protein that binds to RNA molecules and plays a crucial role in cell division and tissue development. During early embryonic growth, it ensures proper differentiation of stem cells. Mice lacking Musashi-1 experience severe neurological defects and typically survive for several months before succumbing to developmental issues. Normally, Musashi-1 is turned off in mature cells, but in some cases, it remains active and has been linked to tumor formation in the brain and kidneys. To counteract Musashi-1 activity, Clingman et al. screened over 30,000 potential inhibitors and identified four promising compounds. One of them, oleic acid–found in olive oil and fish oil–was particularly effective. It interacts with Musashi-1 in a way that causes the protein to break down, thereby preventing it from controlling genes involved in cell proliferation. Additionally, Musashi-1 functions as a metabolic sensor by regulating fatty acid synthesis. When oleic acid levels are low, Musashi-1 suppresses enzyme activity, reducing fatty acid production. However, when oleic acid concentrations rise, the enzyme becomes highly active, increasing fatty acid availability to support rapid cell growth. These findings illustrate the link between diet and gene regulation and could inform new therapeutic strategies for cancer treatment.
Plan-Input	q1: What happens to mice that do not have Musashi-1, and what does this indicate about its function? q2: How did researchers identify inhibitors of Musashi-1, and what is one of the inhibitors they found? q3: What effect does reducing Musashi-1 levels have on colon tumors, and what does this suggest for cancer treatment? Musashi-1 is a crucial protein that binds to RNA and plays a key role in cell division and development. When mice lack Musashi-1, they develop mild brain defects but survive into adulthood, though with noticeable neurological impairments. This suggests that Musashi-1 is essential for normal brain development and proper neural function. To identify potential inhibitors of Musashi-1, Clingman et al. conducted a large-scale screening of nearly 52,000 chemical compounds. Among the four inhibitors they discovered, one was oleic acid, a fatty acid commonly found in olive oil and coconut oil. Oleic acid interacts with Musashi-1 by binding to it directly, leading to its complete deactivation. Reducing Musashi-1 levels in colon tumors has been shown to accelerate their growth, making the tumors more aggressive. This suggests that Musashi-1 may have a dual role in cell regulation, potentially acting as both a tumor promoter and suppressor in different contexts. These findings provide new insights into the complex role of Musashi-1 in cancer and could inform the development of novel therapeutic strategies.

Table 8: Case study 2: We use red font to mark hallucinated information. This sample comes from the test split of the eLife dataset (Goldsack et al., 2022).