PeerJ Computer Science

Modelling and classifying the components of literature reviews: a novel annotation schema and evaluation of transformer models

Francisco Bolanos-Burgos[1], Angelo A. Salatino[1], Francesco Osborne[1,2] and Enrico Motta[1]

[1] Knowledge Media Institute, Open University, Milton Keynes, United Kingdom
[2] Department of Business and Law, University of Milano Bicocca, Milan, Italy

ABSTRACT

Previous work has demonstrated that artificial intelligence (AI) methods for analysing scientific literature benefit significantly from annotating sentences in articles according to their rhetorical roles, such as research gaps, results, limitations, extensions of existing methodologies, and others. Such representations also have the potential to support the development of a new generation of systems capable of producing high-quality literature reviews. However, achieving this goal requires the definition of a relevant annotation schema and effective strategies for large-scale annotation of the literature. This article addresses these challenges in two ways: (1) it introduces a novel, unambiguous annotation schema that is explicitly designed for reliable automatic processing, and (2) it presents a comprehensive evaluation of a wide range of large language models (LLMs) on the task of classifying rhetorical roles according to this schema. To this end, we also present *Sci-Sentence*, a novel multidisciplinary benchmark comprising 700 sentences manually annotated by domain experts and 2,240 sentences automatically labelled using LLMs. We evaluate 37 LLMs on this benchmark, spanning diverse model families and sizes, using both zero-shot learning and fine-tuning approaches. The experiments reveal that modern LLMs achieve strong results on this task when fine-tuned on high-quality data, surpassing 96% F1, with both large proprietary models such as GPT-4o and lightweight open-source alternatives performing well. Moreover, augmenting the training set with semi-synthetic LLM-generated examples further boosts performance, enabling small encoders to achieve robust results and substantially improving several open decoder models.

Subjects Artificial Intelligence, Computational Linguistics, Data Mining and Machine Learning, Digital Libraries, Natural Language and Speech

Keywords Literature review, Systematic literature review, Artificial intelligence, Large language models, Text classification, Literature review generation

Submitted 26 December 2025
Accepted 16 April 2026
Published 1 July 2026

Corresponding author
Francisco Bolanos-Burgos, francisco.bolanos-burgos@open.ac.uk

Academic editor
Ankit Vishnoi

Additional Information and Declarations can be found on page 37

DOI **10.7717/peerj-cs.3921**

Copyright
2026 Bolanos-Burgos et al.

Distributed under
Creative Commons CC-BY 4.0

OPEN ACCESS

INTRODUCTION

A literature review or related work section is a fundamental component of a research article, as it provides the necessary background, highlights the research gap, and justifies the research objectives (*Booth et al., 2021*). It also serves to summarise relevant literature in educational settings, aiding students and researchers in understanding the state of the art

regarding a certain topic. However, crafting a high-quality literature review remains a challenging task, even for experienced researchers. It requires comprehensive knowledge of the relevant literature, which is increasingly difficult to maintain due to the growing volume of published research (*Bornmann & Mutz, 2015*) and the continual need for updates to ensure relevance (*Moher et al., 2007*). In addition, it demands the ability to synthesise this information into a clear and structured discussion that highlights key research directions, theoretical frameworks, and open challenges in the field.

In recent years, research in artificial intelligence (AI) and natural language processing (NLP) has increasingly focused on the problem of automatically generating literature reviews (*Kasanishi et al., 2023*; *Bolanos et al., 2024*; *de Silva Júnior & Dutra, 2021*). More specifically, the objective is to develop systems capable of analysing the existing scientific literature, typically by accessing large repositories of scholarly publications, and producing a related work section comparable in quality to that of a well-written scientific article. This task involves several core challenges: accurately identifying and synthesising the relevant body of literature, presenting a coherent and comprehensive account of the state of the art, systematically comparing alternative approaches, and correctly citing the appropriate articles. Achieving this level of performance would have substantial practical value. For instance, high-quality, automatically generated literature reviews would support researchers and students in analysing and monitoring research fields, and assist stakeholders in the innovation ecosystem who require timely and reliable overviews of emerging areas. The rise of large language models (LLMs) has led to the emergence of various systems that claim to generate scientific analyses by leveraging relevant scholarly literature (*Shi et al., 2023*; *Zhang et al., 2024c*; *Li & Ouyang, 2024a*; *Martin-Boyle et al., 2024*; *Zhang et al., 2024a*; *Nishimura et al., 2024*; *Li et al., 2024d*; *Ma et al., 2025*; *Agarwal et al., 2025*; *Liu et al., 2025*; *Beger & Henneking, 2025*; *Wang et al., 2025*). Many of these systems focus specifically on generating related work sections. However, despite the surface coherence of the outputs produced by current LLM-based approaches, the quality of the resulting literature reviews remains limited (*Bolanos et al., 2024*). This limitation primarily stems from the fact that such outputs often consist of uncritical summaries of individual articles rather than structured and analytical reviews that take into account key elements such as research gaps and limitations (*Bolanos et al., 2024*). Previous studies (*Wang et al., 2022*; *Groza, 2013*; *Widyantoro et al., 2013*; *Teufel & Moens, 2002*) have shown that AI methods for analysing scientific literature benefit significantly from annotating sentences according to their rhetorical roles, such as research gaps, results, and limitations. Texts enriched with such annotations have been shown to facilitate the analysis of the evolution of scientific knowledge (*Wang et al., 2022*), as well as to support the identification (*Groza, 2013*; *Widyantoro et al., 2013*) and prediction (*Teufel & Moens, 2002*) of the significance of scientific concepts and contributions. In light of this evidence, we argue that supporting the development of systems capable of producing high-quality literature reviews requires the integration of rhetorical frameworks for literature analysis, which identify the functional role of each sentence, with the natural language generation capabilities of LLMs. Such an integration would enable systems to leverage a richer representation of the input

text, for example by retrieving all sentences that discuss research gaps within a specific domain in order to generate coherent overviews focused on future research directions.

The first step toward achieving this integration is to address the task of classifying text according to its rhetorical structure, thereby producing a representation that can support downstream systems for literature analysis and evaluation. In this study, we focus particularly on recent transformer-based architectures, which have demonstrated strong performance in related tasks involving the classification of technical text (*Fields, Chovanec & Madiraju, 2024*). This objective gives rise to two main research questions. First, what kind of annotation schema can enable a fully automatic characterisation of sentences in literature reviews? Second, how can current NLP technologies, particularly transformer models, reliably identify the rhetorical roles of sentences in research articles?

This article tackles these challenges by (1) introducing a novel annotation schema explicitly designed to support reliable automatic processing, and (2) performing a comprehensive evaluation of state-of-the-art transformer models on a human-curated benchmark constructed according to this schema. Portions of this text were previously published as part of a preprint (*Bolanos et al., 2025*).

As first step, we developed an annotation schema inspired by prior studies on rhetorical structure (*Khoo, Na & Jaidka, 2011*; *Jaidka, Khoo & Na, 2013*), which categorises scientific sentences into seven classes: `Overall`, `Research Gap`, `Description`, `Result`, `Limitation`, `Extension`, and `Other`. Unlike earlier approaches (*Khoo, Na & Jaidka, 2011*), which were primarily developed for manual annotation and used a few potentially ambiguous labels requiring contextual interpretation beyond the capability of automated classifiers, our schema defines clear and unambiguous categories that can be consistently annotated by machine learning models.

Based on this schema, we created a new publicly available resource, *Sci-Sentence*, a multidisciplinary benchmark that includes 700 sentences manually annotated by domain experts and 2,240 sentences automatically labelled using LLMs. We then evaluated several categories of transformer models on this dataset, covering encoder-only, encoder-decoder, and decoder-only architectures. The evaluation compared models of different sizes (1–3B, 7–8B, and large-scale LLMs), various prominent model families (*e.g.*, BERT, T5, Llama, Mistral, Gemma, Phi), and both proprietary and open-source variants. In total, we tested 37 models, organised into distinct categories, using both zero-shot and fine-tuning settings. The aim was to assess the overall performance of transformer-based models and to gain deeper insights into the most effective approaches and optimisation strategies for this challenging task. We also conducted an in-depth error analysis to identify the categories that pose the greatest challenges for automatic recognition across different model types. For example, we found that the `Limitation` and `Description` categories are particularly difficult, as they can sometimes appear in forms that may be misconstrued as other types. Furthermore, we compared two fine-tuning optimisation techniques, Low-Rank Adaptation (LoRA) and Noisy Embedding Instruction Fine-Tuning (NEFT), and observed that LoRA achieved superior results for large decoder models. Finally, we investigated the impact of training on semi-synthetic data across different model architectures.

The experiments yielded several novel insights that advance the state of the art in this challenging domain. First, the current generation of LLMs performs remarkably well on this task when fine-tuned on high-quality datasets such as *Sci-Sentence*, reaching performance levels above 96% F1. Second, while large proprietary models like GPT-4o (*Hurst et al., 2024*) achieve the best results, lightweight open-source alternatives, such as including *SuperNova-Medius* (https://huggingface.co/arcee-ai/SuperNova-Medius) and *Nemotron-8B* (*Zhang et al., 2023*), also demonstrate excellent performance. Third, although decoder-only models achieve the highest overall scores, small and scalable encoder-based models pre-trained on domain-relevant data, such as *SciBERT* (*Beltagy, Lo & Cohan, 2019*), also achieve solid performance. Therefore, they represent a practical solution for efficiently processing large volumes of text. Finally, enriching the training data with semi-synthetic examples generated by LLMs has proven very beneficial. This solution enables small encoders to achieve robust results and significantly enhances the performance of several open decoder models.

The remainder of this article is structured as follows. 'Related Work' reviews related work, including established literature review frameworks and existing approaches for classifying sentences in scientific articles. 'Framework' defines the task, describes the development of the annotation schema, and presents the novel benchmarks. 'Experimental Methodology' details the experimental methodology and describes the models and approaches used to classify scientific sentences. 'Results' presents the experimental results, with additional analysis on the performance of different methods, optimization techniques, and the effectiveness of semi-synthetic data. Finally, 'Conclusion' concludes the article and outlines potential directions for future research.

RELATED WORK

We review the current literature by focusing on two main research strands. First, we examine existing frameworks for categorising the content of a research article based on rhetorical roles and discourse analysis ('Frameworks based on the Rhetorical Structure of Scientific articles'). Second, we survey various NLP approaches for classifying scientific sentences in the context of generating related work sections ('Approaches for Classifying Sentences in Related Work Section Generation').

Frameworks based on the rhetorical structure of scientific articles

Rhetorical Structure Theory is a framework for text organisation that has inspired applications in discourse analysis, text generation, psycholinguistics, and computational linguistics (*Taboada & Mann, 2006*). It has also been extensively applied to the study, understanding, and generation of scientific and scholarly texts. In particular, three main types of analyses have emerged from its application to scientific and academic literature: *genre analysis* (*Swales & Swales, 1990*; *Kanoksilapatham, 2005*), *zoning analysis* (*Teufel, Carletta & Moens, 1999*; *Mizuta et al., 2006*; *Teufel, Siddharthan & Batchelor, 2009*; *Liakata et al., 2010*), and *discourse analysis* (*Mann & Thompson, 1988*; *Prasad et al., 2008*).

Genre analysis is commonly employed in the field of linguistics due to its effectiveness in manual rhetorical analysis and its pedagogical value in academic writing instruction

(*Swales & Swales, 1990*; *Swales & Feak, 2004*; *Kwan, 2006*; *Bastola & Ho, 2023*). It provides a structured guide to analysing and creating introductions of scientific articles or review chapters of doctoral theses. In contrast, zoning and discourse analyses are studied in NLP research, as they provide machine-interpretable categories and enable fine-grained, sentence-level annotation throughout the entire scholarly document. While zoning analysis focuses on identifying the rhetorical function of individual sentences within the scientific argument (*Teufel, Carletta & Moens, 1999*; *Teufel, Siddharthan & Batchelor, 2009*), discourse analysis examines the textual coherence, meaning, and structural relationships at the sentence level (*Khoo, Na & Jaidka, 2011*).

In the domain of related work generation only a limited number of studies have investigated genre or discourse analysis. *Wang et al. (2024b)*, inspired by the genre analysis, proposed the Create a Research Space (CaRS) model, which describes how academic writers structure introductions by establishing, justifying, and presenting their work. The authors also introduced RSGen, a transformer-based model that employs a two-step decoding process: first creating a rhetorical plan, then generating the related work content. However, RSGen achieves only moderate performance and suffers from issues such as error propagation in classification and limited generalizability to alternative rhetorical schemas. *Khoo, Na & Jaidka (2011)*, drawing on the discourse-based analysis, focused instead on the foundational task of understanding human-written literature reviews through manual analysis. They examined the macro-level discourse structure of literature reviews in information science journals, developing a coding schema with 12 categories. Nevertheless, the coding schema presents several challenges that hinder its suitability for fully automated adoption. First, some categories are defined ambiguously and can be easily misinterpreted even by human annotators, often requiring contextual knowledge that an automated classifier would not possess. Second, certain categories are overly broad, which may lead to confusion and reduced consistency in downstream systems. Finally, the schema was not originally designed for computational processing and therefore lacks reliable training datasets and clear, machine-readable classification guidelines.

In brief, existing research lacks a well-defined structure (*Shi et al., 2023*; *Zhang et al., 2024c*; *Li & Ouyang, 2024a*; *Martin-Boyle et al., 2024*; *Zhang et al., 2024a*; *Nishimura et al., 2024*; *Li et al., 2024d*; *Ma et al., 2025*; *Agarwal et al., 2025*; *Liu et al., 2025*; *Beger & Henneking, 2025*; *Wang et al., 2025*), and to address this, we adopt a discourse-based approach inspired by the findings of *Jaidka, Khoo & Na (2013)*, which emphasizes the critical role of sentences in the generation of related work section. In particular, we seek to enhance the coding schema introduced by *Khoo, Na & Jaidka (2011)*, with the objective of formulating machine-interpretable categories that facilitate their automation by AI systems.

Approaches for classifying sentences in related work section generation

The literature presents a variety of approaches for classifying sentences in scholarly articles. Several methods focus on the rhetorical structure of the article (*Khoo, Na & Jaidka, 2011*), most often targeting the discourse around citations, including citation

function (*Stevens, Giuliano & Garfield, 1965*; *Dong & Schäfer, 2011*; *Teufel, Siddharthan & Tidhar, 2006*; *Jurgens et al., 2018*; *Tuarob et al., 2019*; *Zhao et al., 2019*), citation intent (*Cohan et al., 2019*; *Lauscher et al., 2021*), and citation sentiment (*Athar, 2011*; *Athar & Teufel, 2012*; *Ravi et al., 2018*). Other approaches aim to associate sentences or text segments with relevant topics (*Salatino, Osborne & Motta, 2022*; *Masoumi & Khajavi, 2023*), often selected from one of the many knowledge organisation systems used to categorise scientific literature (*Salatino et al., 2025*). Another category of systems focuses on extracting research entities (*e.g.*, tasks, methods, materials) linked by semantic relations (*Dessí et al., 2022a*). The richer representations of the literature produced by these systems are often encoded in knowledge graphs (*Peng et al., 2023*) (*e.g.*, SemOpenAlex (*Färber et al., 2023*), ORKG (*Jaradeh et al., 2019*), AI-KG (*Dessì et al., 2020*), CS-KG (*Dessí et al., 2022b*), Nano-publications (*Kuhn et al., 2016*)) and have been shown to support effectively scientometric analyses (*Salatino, Osborne & Motta, 2020*; *Salatino et al., 2026*), intelligent systems for exploring the literature (*Angioni et al., 2021*), and, increasingly, conversational systems (*Meloni et al., 2023*) and question-answering methods (*Auer et al., 2023*; *Lehmann et al., 2024*; *Banerjee et al., 2023*). However, as noted by several recent studies (*Li & Ouyang, 2022*, *2024b*; *Bolanos et al., 2024*), the output of existing approaches does not adequately support the generation of related work sections. Therefore, in this article, we propose a sentence classification approach based on a novel schema explicitly designed to support related work analysis and generation.

To the best of our knowledge, only two approaches for related work generation incorporate sentence classification into their pipeline (*Xing, Fan & Wan, 2020*; *Ge et al., 2021*), and both specifically focus on characterising citations. Before discussing these methods, it is useful to first introduce the strategies employed in related work generation. These strategies are commonly classified as either extractive or abstractive. Extractive systems utilise sentences as their units of analysis, producing a paragraph by concatenating selected sentences in a specific order (*Hoang & Kan, 2010*; *Hu & Wan, 2014*; *Wang, Liu & Gao, 2019*; *Chen & Zhuge, 2019*; *Wang et al., 2019*; *Deng et al., 2021*). The paragraph does not have any division and lacks the structure of a human-written literature review (*Li & Ouyang, 2024b*). In contrast, abstractive systems process input such as excerpts or paragraphs generating fluent paragraphs (*AbuRa'ed et al., 2020*; *Xing, Fan & Wan, 2020*; *Ge et al., 2021*; *Chen et al., 2021*; *Luu et al., 2020*; *Jung et al., 2022*; *Li, Mandal & Ouyang, 2022*; *Chen et al., 2022*; *Li, Lee & Ouyang, 2023*; *Liu et al., 2023*; *Gu & Hahnloser, 2024*; *Li & Ouyang, 2024a*; *Martin-Boyle et al., 2024*). However, their outputs often exhibit deficiencies, such as the absence of transitional sentences, improper citation ordering, or as in extractive approaches the lack of structure (*Li & Ouyang, 2024b*).

The two approaches that rely on sentence classification are both abstractive methods. *Xing, Fan & Wan (2020)* trained a Bidirectional Encoder Representations from Transformers (BERT) (*Kenton & Toutanova, 2019*) model to classify sentences as explicit citations, which directly name the source, or implicit citations, which refer to prior work without explicitly mentioning it. *Ge et al. (2021)* fine-tuned SciBERT (*Beltagy, Lo & Cohan, 2019*) to categorise citation sentences as positive, negative, neutral, or mixed, depending on whether they emphasise contributions, highlight shortcomings, or provide objective

Table 1 Sentence classification approaches applied to related work generation.

Study	Classification focus	Strengths	Limitations
Xing, Fan & Wan (2020)	Explicit *vs.* implicit citation detection (BERT-based).	Scalable citation identification model leveraging contextualised BERT representations.	Moderate performance (≈67% F1); does not model rhetorical roles.
Ge et al. (2021)	Citation function classification (positive, negative, neutral, mixed) using SciBERT.	Scalable domain-specific encoder (SciBERT) tailored to scientific text; high classification performance (> 91% F1).	Reduces discourse structure to four 'citation function' categories; does not capture additional rhetorical roles.
Proposed approach	Discourse-level sentence classification for Related Work generation.	Novel schema designed for Related Work generation; supports multiple transformer architectures (encoder, decoder, and encoder–decoder); strong empirical performance (> 96% F1).	Requires high-quality annotated data for fine-tuning; scalability and computational cost depend on the chosen transformer architecture.

descriptions of the cited work. Table 1 summarises these approaches and compares them with the method proposed in this study. The primary limitation of these previous methods lies in the narrow scope of their classification schemas. In particular, *Xing, Fan & Wan (2020)* focus exclusively on distinguishing explicit from implicit citations, whereas (*Ge et al., 2021*) consider only a limited set of citation function categories. Although effective for citation-level sentence analysis, these schemas do not account for several discourse-level functions that are essential for comprehensive related work generation, such as identifying research gaps, describing methodologies, discussing findings, or providing high-level overviews of a research topic. A second limitation concerns their reliance on encoder-based architectures. While such models are highly scalable and well-suited to classification tasks, modern LLMs increasingly dominate this area, often providing greater flexibility and stronger performance in complex discourse modelling tasks.

In recent years, LLMs have emerged as a prominent solution for text classification (*Fields, Chovanec & Madiraju, 2024*; *Zhang et al., 2025*). These models have demonstrated remarkable capabilities in assigning text segments to diverse categories, enabling the accurate extraction and synthesis of fine-grained information from heterogeneous sources, including research articles (*Cadeddu et al., 2024*), patents (*Kosonocky et al., 2024*), medical records (*Omiye et al., 2024*), social media posts (*Yang et al., 2024b*), news articles (*Motta et al., 2024*), technical documentation (*Cascini, Fantechi & Spinicci, 2004*), legal texts (*Savelka & Ashley, 2023*), and financial documents (*Wu et al., 2023*; *Birti, Maurino & Osborne, 2025*; *Li et al., 2023a*). In parallel, substantial research has investigated the interpretability of LLM-based classifiers, showing that these models can often provide meaningful and structured explanations for their predictions (*Song et al., 2025*). Within the specific domain of research article analysis, recent studies further demonstrate that LLMs are capable of encoding complex scientific constructs, including abstract dimensions such as research quality (*McCoubrey et al., 2026*).

However, to the best of our knowledge, LLMs have not yet been applied to the challenging task of identifying rhetorical roles within scientific sentences. This task raises two significant challenges for modern LLMs. First, the focus on scientific articles introduces substantial linguistic complexity. Such texts rely heavily on specialised

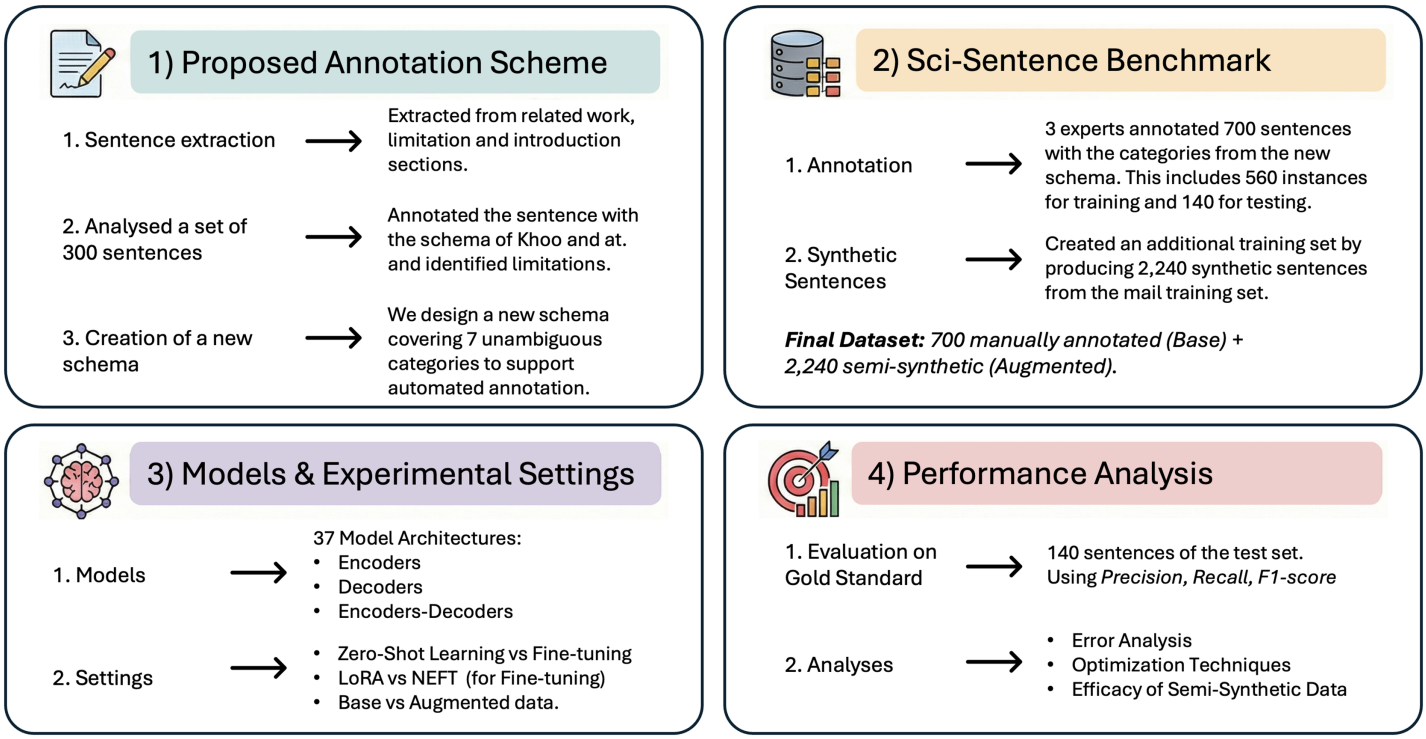

Figure 1 **Workflow of our experimental process.** Full-size DOI: 10.7717/peerj-cs.3921/fig-1

terminology and domain-specific phrasing that may be underrepresented in the training data of general-purpose LLMs. Indeed, scientific literature has consistently posed difficulties for these models, motivating extensive research aimed at improving their robustness in this domain (*Bolanos et al., 2024*). Second, the target categories correspond to rhetorical entities that involve a high level of abstraction. Accurate classification requires a nuanced understanding of how these roles are expressed in scholarly discourse and of the varied linguistic forms they may assume.

In this article, we substantially advance the state of the art in the classification of sentences according to their rhetorical roles in literature reviews by (i) introducing a novel classification schema specifically designed to support systems for related work generation, and (ii) investigating the use of LLMs to automatically label sentences at scale.

FRAMEWORK

This section presents and justifies the theoretical framework and the dataset used for the annotation of research articles to support the analysis and generation of related work. Figure 1 illustrates the complete experimental workflow. We begin by formally defining the task and outlining the classification categories ('Task definition and Annotation Schema'), followed by a detailed description of the Sci-Sentence Benchmark development ('The Sci-Sentence Benchmark'). Finally, 'Experimental Methodology' details the remaining two stages of our process.

Table 2 Proposed annotation schema.

Category	Description	Association
Overall	It is a sentence describing, introducing, classifying, or defining a research topic often based on the discussion of multiple previous studies together.	Topic
Research gap	It is a sentence highlighting the need for new research in a topic due to absence of information, insufficient information or contradictory information.	Topic
Description	It is a sentence describing the objective, methodology or design of a previous study.	Study
Result	It is a sentence presenting the findings of a previous study.	Study
Limitation	It is a sentence describing any factor that can affect the validity or reliability of the previous study regarding its methodology.	Study
Extension	It is a sentence describing how the current study addresses or extends previous studies by stating the overall idea, contrasting ideas or elaborating further ideas.	Study
Other	This denotes a sentence that does not fit within the above categories.	None

Task definition and annotation schema

To support the development of automatic systems capable of generating more structured literature reviews, we aim to characterise sentences in related work sections according to specific rhetorical types. In particular, we assign each sentence to a single, well-defined rhetorical category.

We formulate this task as a single-label multi-class classification problem, in which each sentence is mapped to the most appropriate category. Although some sentences may plausibly belong to more than one rhetorical type, which would motivate a multi-label classification setting, in this study, we follow the approach adopted in prior work (*Teufel, Siddharthan & Batchelor, 2009*) and restrict the task to single-label classification. The exploration of multi-label approaches is left for future work.

After analysing existing classification schemes (*Khoo, Na & Jaidka, 2011*; *Wang et al., 2024b*), we opted to develop a new open annotation schema optimised for automatic labelling. The schema was designed to include only well-defined and unambiguous categories that can be reliably identified by text classification models. The resulting scheme includes seven categories: `Overall`, `Research Gap`, `Description`, `Result`, `Limitation`, `Extension`, and `Other`. These categories are defined in Table 2.

This novel annotation schema was developed by building on the theoretical framework proposed by *Khoo, Na & Jaidka (2011)*, who introduced an influential coding schema for annotating the macro-level discourse structure of related work sections. Their annotation schema comprised twelve categories. In the present study, we focus on eight of these twelve categories, which are described in detail in Table 3. These categories served as the foundation for our schema, as discussed in detail in the following section.

Although influential, the schema proposed by *Khoo, Na & Jaidka (2011)* was not designed to support automated processing and is therefore not suitable for direct integration into machine learning classifiers. Some categories have soft boundaries that partially overlap, and their definitions depend on substantial contextual understanding, which hinders both consistent human annotation and automated interpretation. In addition, several categories are overly broad, introducing potential ambiguity for

Table 3 ***Khoo, Na & Jaidka (2011)*** **annotation schema.** We report the descriptions from the original article.

Name	Description
What	"Encloses a short statement of the topic or study. It is a mandatory element embedded within a topic or a study element. It has a type attribute used to specify the higher level genre of the study, for example, algorithm, model, survey, *etc*"
Description	"Encloses more details of the topic or study element within which it is embedded. It must be preceded by a what element. It also contains other embedded elements that belong within a topic or a study element"
Meta-critique	"Encloses the author's critique of cited studies. It may contain justification for the current study, critical comparison or an indication that the current study is a follow-up of a previous study and may be embedded within the description element of a topic"
Meta-summary	"Encloses an objective, noncritical summary of cited studies, constructed by the author and may be embedded within a description element of a topic"
Brief-topics	"Encloses text which has several studies cited within the span of a single sentence"
Current-study	"Encloses references to the author's current work (being reported in the article)"
Method	"Encloses information about the research method or procedure used in a cited study, including sampling information. It is embedded within the description element of a study"
Result	"Encloses the results of the cited study, including evaluation results if any. It is embedded within the description element of a study and is usually preceded by a method element"

downstream automatic systems that rely on precise classification. For example, the "what" category may refer either to a general concept in the field or to a specific methodology introduced in the same article.

To address these limitations, we developed a revised schema that refines and simplifies the original categories, retaining only those that are unambiguous and can be reliably processed by automatic classifiers. Following the protocol described in *Khoo, Na & Jaidka (2011)*, we conducted an iterative annotation study on more than 300 sentences extracted from research articles in Computer Science, Business, Medicine, and Psychology. We initially adopted the coding schema proposed in *Khoo, Na & Jaidka (2011)*. We then systematically refined it by merging overlapping categories and separating those that exhibited internal ambiguity. Through successive rounds of annotation and revision, this process yielded a final set of clearly defined, mutually distinct categories suitable for robust automatic classification.

The main change concerns the introduction of a clear distinction between (i) sentences that describe the overall research topic and (ii) sentences that refer to individual studies. These two levels are not explicitly addressed in the original framework by *Khoo, Na & Jaidka (2011)*, yet they are practically important for supporting both human and automated annotation. For example, the categories *what* and *description* in the original schema can apply to either the topic level or the study level, and their function varies significantly depending on the context.

In our classification schema, each category is restricted to one of these two levels in order to avoid ambiguity. The topic level is represented by the categories `Overall`, which provides a general overview of the research area, and `Research Gap`, which highlights unresolved issues or open questions in the field. The study level includes four categories that are specific to individual studies: `Description`, `Result`, `Limitation`, and `Extension`. This two-level structure facilitates both automatic annotation systems and

Table 4 Comparison between the two coding schemas.

Proposed schema	Association	*Khoo, Na & Jaidka (2011)* schema
Overall	Topic	What, Description, Meta-summary, Brief-topics
Research gap	Topic	Meta-critique
Description	Study	What, Description, Method
Result	Study	Result
Limitation	Study	Meta-critique
Extension	Study	Current-study
Other	–	–

human annotators by providing a clear conceptual distinction between general and study-specific statements. For example, a sentence describing a limitation of an approach would be classified as `Research Gap` if it concerns the broader research area, whereas it would be labelled as `Limitation` if it refers to a shortcoming of the specific study under discussion. We also introduced the category `Other` because an automatic classifier sometimes encounters sentences that do not match any predefined category. This additional category prevents the model from assigning an incorrect label when its confidence is low or when the text is unrelated to the existing categories.

Table 4 compares the coding schema proposed by *Khoo, Na & Jaidka (2011)* with our revised schema. Notably, we deconstruct *Khoo, Na & Jaidka*'s *(2011)* broad *meta-critique* category, which may conflate critiques of either a topic or a specific study, into two categories: `Research Gap`, which signals the need for further research at the topic level, and `Limitation`, which identifies methodological or conceptual shortcomings in a particular study. Our schema also resolves the ambiguity in *Khoo, Na & Jaidka*'s *(2011)* interchangeable use of *what* and *description* across both levels. We use `Overall` exclusively for sentences that describe the research area as a whole, and reserve `Description` for sentences detailing individual studies. The `Overall` category also consolidates several of *Khoo, Na & Jaidka*'s *(2011)* categories (namely *what*, *description*, *meta-summary*, and *brief-topics*) into a single, more coherent label for topic-level summaries. The `Extension` category has also been reconceptualised. While *Khoo, Na & Jaidka (2011)* use *current-study* for sentences that refer to the current work in a general manner and can therefore be easily conflated with `Description` or `Limitation`, our definition captures the motivations underlying the current work. This includes the articulation of new ideas, the identification of contrasting perspectives, and the elaboration of existing approaches.

The sci-sentence benchmark

To evaluate the ability of modern LLMs to classify sentences according to the annotation schema introduced earlier, we developed the *Sci-Sentence Benchmark*. This open benchmark is designed both to facilitate the training of models on our schema and to enable reliable and reproducible evaluation. In addition, it serves as a reusable resource for systematically assessing future approaches to scientific sentence annotation.

Sci-Sentence comprises 700 sentences manually annotated by domain experts and 2,240 automatically labelled sentences. The sentences were extracted from the introduction, related work, and limitations sections of scientific articles. All articles were published in English, and the corresponding metadata were manually collected from the respective publishers. The disciplinary distribution is as follows: Computer Science (50%), Medicine (20%), Business (10%), Education (10%), and Psychology (10%). The dataset is also balanced by publication type, with 50% of the sentences drawn from conference proceedings and 50% from journal articles.

Sci-Sentence was developed in three phases. First, we conducted an internal annotation session involving three domain experts from our university to define the annotation guidelines, annotate an initial sample, and evaluate inter-annotator agreement. This phase produced an initial set of 140 annotated sentences. After confirming a high level of agreement, the same experts annotated an additional 560 sentences, resulting in a total of 700 manually annotated sentences. Finally, we leveraged Sonnet 3.0 (*Anthropic, 2024*) to expand the training and validation sets by generating four alternative versions of each of the 560 sentences. This strategy enabled the use of a substantially larger dataset for training automatic methods, while preserving a fully manually annotated test set for evaluation. In the following, we provide a more detailed description of the development process.

The initial coordination and annotation session lasted approximately 4 h. The three annotators are senior researchers, two in Computer Science and one in Biology, each with more than 10 years of experience in literature synthesis. For the purposes of this study, extensive subject-matter expertise across all covered disciplines was not strictly required, as the annotation scheme was deliberately designed to be clear and unambiguous, thereby facilitating its adoption by automated methods. The resulting dataset comprised 140 sentences, selected from a larger pool of 300 annotated sentences, with 20 sentences for each of the seven categories. This sampling strategy was necessary because the categories `Result` and `Limitation` occurred relatively infrequently in the original dataset and were therefore underrepresented.

To ensure that the annotators produced consistent and reliable annotations, we then assessed inter-rater agreement on this dataset. Specifically, we employed two metrics: Fleiss' Kappa (*Fleiss, 1971*) and Gwet's AC1 (*Gwet, 2008*). Fleiss' Kappa was used to measure the overall agreement among the three raters across the entire annotation exercise (*Fleiss, 1971*). In contrast, Gwet's AC1 was applied to evaluate agreement at the category level. This choice was motivated by the fact that Gwet's AC1 is designed to overcome certain limitations of Fleiss' Kappa, providing a more robust and stable measure of inter-rater reliability under varying prevalence and marginal distribution conditions (*Gwet, 2008*).

A Fleiss' Kappa of 0.90 was achieved for the overall agreement among the three raters, indicating a high level of inter-annotator reliability (*Landis & Koch, 1977*). Table 5 reports Gwet's AC1 scores for category-specific agreement. In most cases, the agreement is above 0.80. The only exceptions are the `Research Gap` and `Limitation` categories, with agreement values of 0.78 and 0.75, respectively, which still represent substantial agreement according to established guidelines (*Landis & Koch, 1977*). This strong general and

Table 5 Average Gwet's AC1 per category.

Category	Gwet's AC1
Overall	0.89
Research gap	0.78
Description	0.89
Result	0.89
Limitation	0.75
Extension	0.93
Other	0.97

category-specific agreement indicates that the annotation task is well-defined and that the experts were able to produce consistent labels. In turn, this suggests that the Sci-Sentence benchmark can serve as a high-quality resource for training downstream applications.

After confirming the strong inter-annotator agreement, the three annotators proceeded independently with the annotation process. Each annotator was assigned a distinct subset of sentences and followed the original annotation guidelines. Annotation proceeded until each category was expanded by an additional 80 sentences, resulting in a total of 560 new annotated sentences. As a result, the complete dataset now contains 700 sentences, which are split into 70% for training (490 sentences), 10% for validation (70 sentences), and 20% for testing (140 sentences).

Given the labour-intensive and time-consuming nature of the annotation process, we also explored the use of semi-synthetic data. Unlike fully-synthetic samples, which are generated by mimicking the statistical properties of a dataset, semi-synthetic data refers to data generated considering the representation of real-world objects, such as original sentences (*Joshi et al., 2024*). Specifically, we aimed to generate artificial sentences that replicate the characteristics of those found in the original dataset (*Marwala, Fournier-Tombs & Stinckwich, 2023*). This approach has gained considerable momentum with the advent of LLMs (*Liu et al., 2024a*). The literature provides compelling evidence that semi-synthetic data can enhance dataset diversity (*He et al., 2021*), support threats detection in security (*Myneni et al., 2023*), address missing values (*Berti-Équille et al., 2018*), mitigate algorithmic bias (*Li, De-Arteaga & Saar-Tsechansky, 2022*), and support privacy-preserving data sharing (*Yale et al., 2019*). In line with recent studies (*Kaddour & Liu, 2023*; *Li et al., 2023b*, *2024c*), our experimental findings (see 'Results') confirm the effectiveness of this strategy.

To this end, we employed Sonnet 3.0 (accessed *via* Amazon Bedrock) to generate four semi-synthetic sentences for each original sentence in the training and validation sets only, leaving the test set unaltered (*Stan et al., 2025*). We subsequently evaluated the generated sentences to verify that they were sufficiently syntactically distinct from their corresponding source sentences. For this purpose, we used the normalised Levenshtein distance (see 'Augmented Data'), which measures the similarity between two sentences on a scale from 0 (identical) to 1 (completely different). Specifically, we computed the normalised Levenshtein distance between each original sentence and its four new variants,

as well as the average distance between each new sentence and the remaining three. If any of these distances fell below or equal to 0.20, indicating insufficient syntactic variation, these sentences were discarded and regenerated using the language model until they met the threshold.

'Augmented Data' includes the prompt used to generate the semi-synthetic sentences and a table reporting the normalised Levenshtein distances between sentences, grouped by category. This process resulted in 1,960 new sentences for the training set and 280 for the validation set, bringing the total number of sentences to 2,450 and 350, respectively, when combined with the manually annotated data.

We released the Sci-Sentence benchmark in two versions: (1) the base version, which contains 700 manually annotated sentences; and (2) the augmented version, which includes a total of 2,940 sentences comprising both the manually annotated data and additional semi-synthetic examples. As previously mentioned, in both versions the test set consists exclusively of manually annotated sentences to ensure a fair evaluation. The benchmark is available on GitHub under a CC-BY license (https://github.com/fcobolanos/Classifying-the-Components-of-a-Literature-Review/tree/main/datasets).

EXPERIMENTAL METHODOLOGY

This article aims to investigate the capability of AI models to annotate the rhetorical roles of sentences within research articles. It has two main objectives: (i) to introduce a new annotation schema and a relevant dataset, detailed in 'Framework'; and (ii) to examine whether current language models can accurately and efficiently perform this task at scale, as well as to identify which architectures are most effective.

This section describes the experimental methodology related to the second objective. Specifically, we conducted a comprehensive evaluation of a broad set of state-of-the-art LLMs (*Patel et al., 2023*; *Kasneci et al., 2023*) on the novel Sci-Sentence Benchmark, under both zero-shot learning (ZSL) and fine-tuning settings, as shown in Fig. 1. In particular, we fine-tuned the models using both the manually curated dataset and an augmented version of the training set, with the aim of assessing whether data augmentation leads to measurable performance improvements. We assessed 37 alternative solutions spanning a variety of model architectures, including encoder-only, decoder-only, and encoder–decoder models, and a wide range of parameter sizes. These included both open-source and proprietary solutions. All models were evaluated on the test set of the Sci-Sentence Benchmark, and their performance was measured using precision, recall, and F1-score.

Given the already substantial number of required experiments, we did not include in-context learning in our evaluation. In-context learning incorporates a small number of labelled examples directly into the prompt and has been shown to outperform zero-shot learning in several domains (*Song et al., 2026*). However, fine-tuning generally provides stronger and more stable performance in supervised classification tasks (*Fields, Chovanec & Madiraju, 2024*; *Li et al., 2023b*; *Zhang et al., 2025*). Furthermore, the limited number of target categories and the moderate size of the training set make fine-tuning computationally feasible. For these reasons, our analysis is restricted to zero-shot and fine-tuned configurations.

The following subsections present the experiments conducted using ZSL ('Zero-Shot Learning Settings') and fine-tuning ('Fine-Tuning Settings'). Finally, 'Overview of the Models' provides an overview of the employed LLMs. The code implementation for 'Zero-Shot Learning Settings' and 'Fine-tuning Settings' is available in the associated repository (https://github.com/fcobolanos/Classifying-the-Components-of-a-Literature-Review/tree/main/code).

Zero-shot learning settings

For the ZSL experiments, we evaluated a total of 12 decoder-only models. We excluded encoder-only and encoder-decoder architectures because they are generally not well-suited for ZSL tasks.

Among the 12 models, eight were open-source and varied in size from 2 billion to 123 billion parameters. The remaining four were proprietary models whose parameter counts are not publicly available but are widely believed to exceed 700 billion. To ensure a fair comparison, we applied the same prompt template across all models. This template, refined through iterative prompt engineering following best practices (*Berryman & Ziegler, 2024*), consists of three components: (1) an objective description to help the language model understand the task; (2) a precise explanation of the seven classification categories; and (3) a detailed procedure that includes the sentence to be classified along with a clearly specified output format to facilitate automated parsing. For transparency and reproducibility, the full prompt template is provided in 'Prompt for Zero-Shot Learning'.

The LLMs were executed using three different systems: (1) Amazon Bedrock (https://aws.amazon.com/bedrock/), (2) the OpenAI API (https://openai.com/api/), and (3) KoboldAI (https://github.com/KoboldAI/KoboldAI-Client). Amazon Bedrock provides a pre-configured generic wrapper that standardises interactions with various supported LLMs, such as the MistralAI family, Anthropic models, and Llama models. The OpenAI API was used to access ChatGPT. KoboldAI, an open-source tool based on llama.cpp, enables the local execution of LLMs and exposes them *via* an API endpoint. In our setup, we used KoboldAI to load quantised models on a Google Colaboratory instance equipped with Nvidia V100 and L4 GPUs. The parameter configurations of all models are detailed in 'Parameters Settings for Zero-shot Learning'.

Fine-tuning settings

For the fine-tuning experiments, we evaluated 25 models spanning decoder, encoder, and encoder-decoder architectures, aiming to identify the most effective approach for this task. Each model was fine-tuned twice: first using the base version of the Sci-Sentence benchmark and then using its augmented counterpart, in order to evaluate the impact of data augmentation.

We employed two distinct fine-tuning strategies, tailored to the specific architecture of each model. For encoder models, we converted the sentences and their corresponding categories into tensors ('List of Models', 'Encoder'). For decoder models ('Decoder-Small FT', 'Decoder-Medium FT', and 'Decoder-Large FT' of 'List of Models') and

encoder-decoder models ('List of Models', 'Encoder-Decoder'), we constructed prompts in a manner similar to Zero-Shot Learning, but without including category examples.

The models were fine-tuned using Google Colaboratory instances equipped with Nvidia V100 and L4 GPUs. For GPT-4o-mini, however, we relied on the OpenAI API. To mitigate the computational and memory constraints of Google Colaboratory, we adopted QLoRA (*Dettmers et al., 2024*), a method that quantizes model weights from high-precision (32-bit) to low-precision (8-bit) formats. This quantization significantly reduces both computational overhead and memory usage.

To further reduce the number of trainable parameters in our decoder models, we evaluated two optimisation techniques: Low-Rank Adaptation (LoRA) (*Hu et al., 2021*) and Noisy Embedding Instruction Fine-Tuning (NEFT) (*Jain et al., 2023*). LoRA introduces small, trainable matrices derived from a low-rank decomposition of weight updates. During inference, these updates are combined with the original weights to produce the final output. In contrast, NEFT adds random noise to embedding vectors during training. We selected LoRA due to its widespread adoption as a standard optimisation method (*Mao et al., 2024*), and NEFT for its demonstrated effectiveness in improving performance (*Zhao et al., 2024*; *Laurençon et al., 2024*; *Li et al., 2024e*; *Pradeep, Sharifymoghaddam & Lin, 2023*).

To accelerate the fine-tuning procedure, we employed Unsloth (https://github.com/unslothai/unsloth?utm_source=chatgpt.com) as it offers up to 30× faster training and a 90% reduction in memory consumption without compromising accuracy. In cases where Unsloth was not applicable (*e.g.*, unsupported models or decoder-small models not needing optimization), we used Hugging Face Transformers. 'Parameters Settings for Fine-Tuning' provides comprehensive details on parameter values and platforms for each model.

In Section 5 (Results), we will focus on the best configuration for each model in terms of training data (base *vs*. augmented) and optimisation technique (NEFT *vs*. LoRA). We report the complete set of results is available in the associated repository (https://github.com/fcobolanos/Classifying-the-Components-of-a-Literature-Review/tree/main/results).

Overview of the models

In summary, the experiments involved a total of 37 approaches, including 12 using zero-shot learning and 25 using fine-tuning. These approaches varied in training parameters, quantisation strategies, openness (*i.e.*, open-source *vs*. proprietary), fine-tuning methods, and model architectures. This comprehensive analysis enabled us to evaluate not only the performance of individual models but also the effectiveness of specific architectures and optimisation techniques for the task at hand.

To facilitate the analysis of the results, we divided the models into six categories based on their architecture (encoder, decoder, or encoder-decoder), training setting (ZSL *vs*. fine-tuning), and model size. These categories are: *Encoder* (four models), *Encoder-decoder* (four models), *Decoder-ZSL* (12 decoders in ZSL setting), *Decoder-Small-FT* (seven fine-tuned decoders with fewer than 4B parameters), *Decoder-Medium-FT* (six fine-tuned decoders between 4B and 8B), and *Decoder-Large-FT* (four fine-tuned decoders with more than 8B parameters).

Table 6 Main characteristics of the selected LLMs. The models are grouped by architectural type. M, million, B, billion, and T, trillion. **Context** length is measured in number of tokens. Not discl. means information not disclosed.

Category	Model name	# Param.	Context	Trained
Encoder	SciBERT	110 M	512	3.1 B
	BioBERT	110 M	512	18 B
	BigBird	25 M	4,096	1.5 B
	BERT	110 M	512	3.3 B
Decoder ZSL	Llama2 (Open-source full)	70 B	4,096	2 T
	Llama3 8b (Open-source full)	8 B	8,000	15 T
	Llama3 70b (Open-source full)	70 B	8,192	15 T
	Mistral (Open-source full)	7 B	2,000	Not discl.
	Mixtral (Open-Source Full)	46.7 B	32,000	Not discl.
	Mistral Large (Open-source full)	123 B	32,000	Not discl.
	Gemma (Open-source quantised)	2 B	8,192	2 T
	Orca (Open-source quantised)	13 B	4,096	Not discl.
	Sonnet (Proprietary)	Not discl.	200,000	Not discl.
	Haiku (Proprietary)	Not discl.	200,000	Not discl.
	GPT-3.5 (Proprietary)	Not discl.	16,000	Not discl.
	GPT-4 (Proprietary)	Not discl.	128,000	Not discl.
Decoder Small-FT	Gemma2-2B	2 B	8,192	2 T
	Olmo-1B	1 B	2,048	3 T
	SmolLM2	1.7 B	2,048	11 T
	TinyLlama	1.1 B	2,048	3 T
	Arcee-lite (Merged)	1.5 B	32,000	Not discl.
	Phi3.5	3.8 B	128,000	3.4 T
	Llama3.2-3B	3 B	8,000	15 T
Decoder Medium-FT	Nemotron-8B	8 B	4,096	3.5 T
	Olmo-7B	7 B	2,048	2.5 T
	Mistral-7	7 B	2,000	Not discl.
	SuperNova-Lite (Merged)	8 B	128,000	Not discl.
	Llama3-8B	8 B	8,000	15 T
	Arcee-Spark (Merged)	7 B	128,000	Not discl.
Decoder Large-FT	GPT-4o-mini	Not discl.	28,000	Not discl.
	SuperNova-Medius (Merged)	14 B	31,072	Not discl.
	Gemma2-9B	9 B	8,192	2 T
	Mistral-Nemo	12 B	128,000	Not discl.
Encoder decoder	T5 xxl	11 B	512	Not discl.
	T5 Large	770 M	512	34 B
	T5	222 M	512	1 T
	T5 xl	3 B	512	Not discl.

Table 6 presents all the models grouped by category, including their number of parameters, context window size, and the size of the datasets used during their original pretraining. Additional details about each model are provided in 'List of Models'.

With respect to model size, although the number of parameters is widely regarded in the literature as the standard metric, there is no clear consensus on the specific thresholds that distinguish small from large models (*Wang et al., 2024a*). For example, *Liu et al. (2024b)* define small models as those containing approximately one billion parameters, while (*Fu et al., 2023*) consider models with up to ten billion parameters to still fall within the small category. Due to this lack of agreement, we established our own thresholds to achieve a relatively balanced distribution of models across size categories.

Finally, we note that we also chose to include four merged models (Arcee-lite, Arcee-Spark, SuperNova-Lite and SuperNova-Medius) in our evaluation. These models are constructed by combining the weights or architectures of multiple pre-trained LLMs to leverage their complementary strengths. This model merging technique efficiently enhances LLMs by integrating specialized knowledge and capabilities from different models into a single, more robust, and adaptable system. In particular, Arcee Lite is derived from a Qwen2-based architecture and represents a distilled variant of the Phi-3-Medium model. Arcee-Spark is also derived from a Qwen2-based architecture, and distilled from the Qwen2-7B-Instruct model. The SuperNova-Lite model is constructed on the Llama-3.1-8B-Instruct24 architecture and results from the distillation of the more expansive Llama-3.1-405B-Instruct model. Furthermore, SuperNova-Medius employs the Qwen2.5-14B-Instruct architecture and incorporates distilled knowledge from both the Qwen2.5-72B-Instruct and Llama-3.1-405B-Instruct models.

RESULTS

In this section, we discuss the results of our experiments. As described in 'Experimental Methodology', all models were evaluated on the test set of the Sci-Sentence Benchmark, and their performance was assessed using precision, recall, and macro F1-score.

The first three subsections present an overall analysis of model performance. 'Zero-Shot Learning' reports the results obtained by the 12 ZSL decoder models. 'Fine-Tuned Models' describes the best-performing configurations for each of the 25 fine-tuned models. Finally, 'Performance across Rhetorical Categories' compares the top-performing models across the seven categories of the schema. The following three subsections ('Error Analysis through Confusion Matrices–Assessing the Efficacy of Semi-Synthetic Data') provide an in-depth analysis to interpret and contextualise the results presented earlier. Specifically, 'Error Analysis through Confusion Matrices' details the most prevalent model errors. 'Comparing Optimisation Techniques' investigates the effects of applying optimisation techniques (LoRA, NEFT) to the decoder models. Finally, 'Assessing the Efficacy of Semi-Synthetic Data' evaluates the impact of data augmentation on model performance across different optimisation strategies and training data solutions (base *vs.* augmented).

In the following analysis, we clearly distinguish between proprietary models, which are beyond our control and may involve additional undocumented processing steps, and open models, which were executed entirely within our configuration. While proprietary models may achieve superior performance, they are also less replicable. Therefore, we consider it important to evaluate them in a separate category.

Table 7 Precision, recall, and F1-score of experiments with zero-shot learning. PR, Precision, RE, Recall, F1, F1-score. Bold values indicate the best performance for each category.

MODEL		Llama2	Llama3 8b	Llama3 70b	Mistral	Mixtral	Mistral Large	Gemma	Orca	Sonnet	Haiku	GPT-3.5	GPT-4
PR	Average	0.542	0.471	**0.862**	0.760	0.737	0.797	0.181	0.678	**0.898**	0.733	0.624	0.824
	Overall	0.600	**1.000**	0.905	0.654	0.682	0.818	0.158	0.533	**1.000**	0.750	0.611	0.800
	Research gap	0.455	**1.000**	**1.000**	0.789	0.882	0.727	0.125	1.000	0.900	0.875	**1.000**	**1.000**
	Description	0.556	0.333	**1.000**	0.833	0.538	0.857	0.125	0.275	**1.000**	0.500	0.167	0.833
	Result	0.404	0.204	0.377	**0.773**	0.594	0.500	0.172	0.513	0.488	0.486	0.404	**0.556**
	Limitation	0.174	0.244	0.900	0.739	**0.909**	0.800	0.227	0.900	**1.000**	0.722	0.667	**1.000**
	Extension	0.667	0.000	0.850	0.640	0.682	**0.875**	0.059	0.667	**0.900**	0.800	0.516	0.809
	Other	0.938	0.513	**1.000**	0.895	0.870	**1.000**	0.400	0.857	**1.000**	**1.000**	**1.000**	0.769
RE	Average	0.477	0.363	0.713	0.736	0.720	**0.751**	0.157	0.561	**0.822**	0.707	0.573	0.770
	Overall	0.545	0.091	**0.864**	0.773	0.682	0.818	0.136	0.364	**0.818**	**0.818**	0.500	0.727
	Research gap	0.526	0.474	**0.842**	0.789	0.789	**0.842**	0.158	0.526	**0.947**	0.737	0.474	0.737
	Description	0.278	0.056	0.056	0.278	**0.389**	0.333	0.167	0.611	**0.611**	0.222	0.056	0.556
	Result	0.950	0.450	**1.000**	0.850	0.950	0.850	0.250	1.000	**1.000**	0.900	0.950	**1.000**
	Limitation	0.191	0.524	0.429	**0.809**	0.476	0.762	0.238	0.429	0.476	**0.619**	0.381	0.524
	Extension	0.100	0.000	**0.850**	0.800	0.750	0.700	0.050	0.400	**0.900**	0.800	0.800	0.850
	Other	0.750	0.950	0.950	0.850	**1.000**	0.950	0.100	0.600	**1.000**	0.850	0.850	**1.000**
F1	Average	0.455	0.312	0.694	0.726	0.710	**0.749**	0.154	0.567	**0.826**	0.701	0.553	0.768
	Overall	0.571	0.167	**0.884**	0.708	0.682	0.818	0.146	0.432	**0.900**	0.783	0.550	0.762
	Research gap	0.488	0.643	**0.914**	0.789	0.833	0.780	0.140	0.690	**0.923**	0.800	0.643	0.849
	Description	0.370	0.095	0.105	0.417	0.452	**0.480**	0.143	0.379	**0.759**	0.308	0.083	0.667
	Result	0.567	0.281	0.548	**0.809**	0.731	0.630	0.204	0.678	0.656	0.632	0.567	**0.714**
	Limitation	0.182	0.333	0.581	0.773	0.625	**0.780**	0.233	0.581	0.645	0.667	0.485	**0.688**
	Extension	0.174	0.000	**0.850**	0.711	0.714	0.778	0.054	0.500	**0.900**	0.800	0.627	0.829
	Other	0.833	0.667	**0.974**	0.872	0.930	**0.974**	0.160	0.706	**1.000**	0.919	0.919	0.870

Zero-shot learning

Table 7 presents the performance of the 12 LLMs in ZSL for classifying each of the seven categories, along with their average performance.

Sonnets achieves the highest overall F1-score (82.6%), followed by GPT-4 (76.8%) and Mistral Large (74.9%). These results confirm that the largest LLMs are capable of performing well on this task, although there is still considerable room for improvement.

When focusing on the open models (the first eight columns), the Mistral AI family, including Mistral Large, Mistral, and Mixtral, exhibits strong performance, achieving average F1-scores of 74.9%, 72.6%, and 71.0%, respectively. Notably, Llama 3 70B also achieves solid results (F1-score of 69.4%) and particularly excels in precision, reaching values above 85.0% in all categories except `Result`.

Regarding the proprietary models, as previously mentioned, Sonnet achieves the highest F1-score, closely followed by GPT-4. Both models demonstrate a well-balanced performance in terms of precision and recall across all categories. However, for certain categories, both are actually outperformed by the best open Mistral models. In particular, Sonnet is surpassed by Mistral in the `Result` category (65.6% *vs.* 80.9%), and by Mistral

Large in the `Limitation` category (64.5% *vs.* 78.0%). This suggests that, although these proprietary models perform best overall, they can still be challenged, and even outperformed, by open models in specific areas.

Finally, we can observe a recurring pattern in which the majority of the models exhibit low precision in the `Result` category, as well as low recall in the `Description` and `Limitation` categories. We will discuss more specific error patterns in the following sections.

Fine-tuned models

The fine-tuning experiments evaluated 25 models, grouped into the categories previously introduced in 'Experimental Methodology': *Encoder* (four models), *Encoder-decoder* (four models), *Decoder-Small-FT* (seven models), *Decoder-Medium-FT* (six models), and *Decoder-Large-FT* (four models). Note that *Decoder-ZST* is excluded here, as it was analysed in the preceding subsection.

As discussed in 'Fine-Tuning Settings', the fine-tuning experiments were performed using a range of configurations. In particular, each decoder was fine-tuned under four different settings, obtained by combining two training sets (base and augmented) with two optimisation methods (LoRA and NEFT). In contrast, encoders and encoder-decoder models were fine-tuned on both training sets using the standard fine-tuning procedure. To ensure clarity and avoid unnecessary detail, we report only the best-performing configuration for each model. A more detailed analysis of the effects of the optimisation techniques and training sets on performance is presented in 'Comparing Optimisation Techniques', and 'Assessing the Efficacy of Semi-Synthetic Data'.

Table 8 presents the F1-score, precision, and recall of the fine-tuned models. Within each architectural type, the models are listed in order of decreasing F1-score. In this section, we discuss their average performance across all rhetorical categories, while the next section examines the performance of the top models within each category.

The results are clearly superior to those obtained in the ZSL setting, confirming the value of the Sci-Sentence Benchmark training sets in enabling high-performance models for this task.

In terms of model type, decoder-based models outperform encoder-based ones, which in turn outperform encoder-decoder architectures. Among the decoder models, model size plays a significant role. *Decoder-Large-FT* achieved the best results, with GPT-4 reaching the highest F1-score of 96.4%, followed by *Decoder-Medium-FT* and *Decoder-Small-FT*. Notably, SuperNova-Medius, a merged open-source model with 14B parameters trained on the augmented dataset, obtained the second-best result overall, with an F1-score of 94.3%. This also represents a 17.5 percentage point improvement in F1-score over GPT-4 in the ZSL setting, which is commonly adopted as a default in many corporate solutions. These findings validate the importance of the novel Sci-Sentence datasets and demonstrate that open models can be highly competitive.

In the *Decoder-Large-FT* category, GPT-4o-mini achieved the highest F1-score (96.4%), followed by SuperNova-Medius (94.3%) and Gemma2-9B (94.2%). Within the *Decoder-Medium-FT* category, the top-performing models showed very similar results.

Table 8 Ranking of models by type. In **Conf**iguration: **B,** Base version of benchmark; **A,** Augmented benchmark; **L,** LoRA; **N,** NEFT. Bold values indicate the highest F1-score within each model type category.

Model type	Model name	Conf.	Precision	Recall	F1-score
Encoder	SciBERT	A	0.929	0.928	**0.928**
	BioBERT	A	0.881	0.882	0.878
	BigBird	A	0.886	0.881	0.878
	BERT	A	0.871	0.866	0.861
Decoder-small	Gemma2-2B	AL	0.931	0.930	**0.928**
	Olmo-1B	BN	0.926	0.924	0.921
	SmolLM2	AN	0.923	0.916	0.914
	TinyLlama	AN	0.891	0.879	0.879
	Arcee-lite	BL	0.887	0.882	0.878
	Phi3.5	BN	0.872	0.870	0.869
	Llama3.2-3B	BL	0.859	0.856	0.857
Decoder-medium	Nemotron-8B	AL	0.940	0.937	**0.936**
	Olmo-7B	AN	0.938	0.938	0.935
	Mistral-7	BL	0.937	0.934	0.933
	SuperNova-Lite	BL	0.932	0.927	0.928
	Llama3-8B	BN	0.919	0.917	0.914
	Arcee-Spark	BN	0.890	0.887	0.886
Decoder-large	gpt-4o-mini	B	0.966	0.963	**0.964**
	SuperNova-Medius	AL	0.945	0.943	0.943
	Gemma2-9B	BL	0.943	0.943	0.942
	Mistral-Nemo	AL	0.933	0.927	0.929
Encoder-decoder	T5 xxl	B	0.910	0.898	**0.899**
	T5 Large	A	0.893	0.894	0.892
	T5	A	0.882	0.882	0.879
	T5 xl	A	0.860	0.858	0.856

Nemotron-8B achieved the best score (93.6%), followed closely by Olmo-7B (93.5%) and Mistral-7B (93.3%). The *Decoder-Small-FT* models performed only slightly worse than the medium-sized decoders, with Gemma2-2B reaching the highest score in this category (92.8%).

Notably, the *Encoder* category produced results comparable to the smaller decoder models. In particular, Sci-BERT, a BERT variant pre-trained on academic text and therefore well-suited for this task, achieved an F1-score of 92.8%, matching the performance of Gemma2-2B. This outcome is especially interesting because encoder models are generally faster and more scalable than small decoders. Thus, they offer an efficient solution for sentence classification with only a 1.5 percentage point drop in F1-score compared to the best-performing open model (SuperNova-Medius), and a 3.6 point drop compared to the top proprietary solution (GPT-4o-mini).

The encoder-decoder models did not perform particularly well, with the exception of T5-XXL, which achieved a solid 89.9% F1. This result suggests that this architecture may not be particularly well suited to the task.

Regarding the merged models, while SuperNova-Medius achieved an excellent result as the first open model, the other two merged models did not perform as well. SuperNova-Lite matched SciBERT's F1-score of 92.8%, whereas Arcee-Lite and Arcee-Spark reported lower F1-scores, both falling below 90%.

Performance across rhetorical categories

To assess how the categories in the annotation schema influence the performance of different classifier architectures, Table 9 reports the F1-score, precision, and recall of the best models for each architecture type (encoder, encoder–decoder, and small, medium, and large decoder) across the seven categories of the annotation schema. The table also includes the performance of the ZSL decoders (D-ZSL).

GPT-4o-mini and Nemotron-8B, the top-performing large and medium-sized models respectively, achieve the highest overall performance, with average F1-scores of 96.4% and 93.6%. In contrast, Sonnet, in the zero-shot learning (ZSL) setting, records the lowest average F1-score among the best models by type, underperforming by 13.8 and 11.0 percentage points compared to GPT-4o-mini and Nemotron-8B, respectively.

Notably, GPT-4o-mini achieves the highest absolute F1-score in five of the seven categories. Nemotron-8B matches GPT-4o-mini in the `Limitation` category and achieves the best result in `Result`. Finally, and perhaps surprisingly, the smaller SciBERT yields the best performance in `Extension`.

Focusing on category-specific performance, `Other` is clearly the easiest category to identify. All top models achieve perfect scores in this case. A closer examination suggests this is because classification errors tend to occur among semantically similar categories, such as confusing `Description` or `Limitation` with `Result`, whereas `Other` is semantically distinct enough to be reliably recognised by most systems.

By contrast, the categories `Limitation` and `Description` are the most challenging to classify. This is particularly evident from the performance of Sonnet in the ZSL setting, where it achieves a Recall below 61% for both categories. The category `Result` also yields a low F1-score, but for a different reason. Although it is identified with high accuracy, leading to a high Recall, it exhibits a relatively low Precision.

However, models fine-tuned on *Sci-Sentence* demonstrate a substantially better understanding. In particular, Nemotron-8B achieves F1-scores of 94.7% and 95.0% on `Limitation` and `Result`, respectively. A similar, though less pronounced, trend is observed for `Description`, where the best ZSL method attains an F1-score of 75.9%, while the best fine-tuned model reaches 94.1%.

In summary, we identify four key insights. First, the current generation of LLMs can perform exceptionally well on this task when fine-tuned on high-quality datasets, such as *Sci-Sentence*, achieving performance levels exceeding 96%. In contrast, ZSL produces substantially lower results, underscoring the critical role of domain-specific training data. Second, while large proprietary models such as GPT-4o attain the highest performance, lightweight open-source alternatives, such as SuperNova-Medius and Nemotron-8B, also yield excellent results. These models offer additional benefits in terms of scalability, reproducibility, and transparency. Third, although decoder-only models

Table 9 **PR**ecision, **RE**call, and **F1**-score of the best performing models by architectural type. In configuration: **B,** Base version of benchmark; **A,** Augmented benchmark; **L,** LoRA; **N,** NEFT. Bold values indicate the highest score for each evaluation category across model architectures.

Model type		Enc	D-ZSL	D-Sma	D-Med	D-Lar	Enc-Dec	
MODEL		SciBERT(A)	Sonnet	Gemma2-2B(L)	Nemotron-8B(L)	GPT-4o-mini	T5 xxl	Average
Configuration		A	B	BL	BL	B	B	–
PR	Average	0.929	0.898	0.931	0.940	**0.966**	0.910	0.929
	Overall	**1.000**	**1.000**	0.955	0.950	**1.000**	0.800	0.951
	Research gap	0.857	0.900	0.864	0.905	**0.950**	0.864	0.890
	Description	0.941	**1.000**	0.895	0.889	**1.000**	**1.000**	0.954
	Result	0.895	0.488	0.947	**1.000**	0.905	0.809	0.841
	Limitation	0.857	**1.000**	**1.000**	**1.000**	**1.000**	0.947	0.967
	Extension	**0.952**	0.900	0.857	0.833	0.909	0.950	0.900
	Other	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	1.000
RE	Average	0.928	0.822	0.930	0.937	**0.963**	0.898	0.913
	Overall	0.955	0.818	0.955	0.864	**1.000**	0.909	0.917
	Research gap	0.947	0.947	**1.000**	**1.000**	**1.000**	**1.000**	0.982
	Description	0.889	0.611	**0.944**	0.889	0.889	0.722	0.824
	Result	0.850	**1.000**	0.900	0.900	0.950	0.850	0.908
	Limitation	0.857	0.476	0.809	**0.905**	**0.905**	0.857	0.802
	Extension	**1.000**	0.900	0.900	**1.000**	**1.000**	0.950	0.958
	Other	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	1.000
F1	Average	0.928	0.826	0.928	0.936	**0.964**	0.899	0.914
	Overall	0.977	0.900	0.955	0.905	**1.000**	0.851	0.931
	Research gap	0.900	0.923	0.927	0.950	**0.974**	0.927	0.934
	Description	0.914	0.759	0.919	0.889	**0.941**	0.839	0.877
	Result	0.872	0.656	0.923	**0.947**	0.927	0.829	0.859
	Limitation	0.857	0.645	0.895	**0.950**	**0.950**	0.900	0.866
	Extension	**0.976**	0.900	0.878	0.909	0.952	0.950	0.928
	Other	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	1.000

achieve the best overall performance, encoder-based models pre-trained on relevant data, such as SciBERT, can still deliver very competitive results. Moreover, they offer significantly higher scalability, making them a practical alternative when processing large volumes of text. Finally, certain categories, notably `Limitation` and `Description` remain particularly challenging, especially in ZSL settings. However, the use of high-quality training data enables satisfactory performance even in these more difficult cases.

Error analysis through confusion matrices

Figure 2 presents the confusion matrices for the best-performing model of each of the six architectural types, providing a more detailed view of the results reported in Table 9. Among these, GPT-4o-mini achieved the highest performance, with only five misclassifications. In contrast, Sonnet in ZSL, which was the least accurate among the top

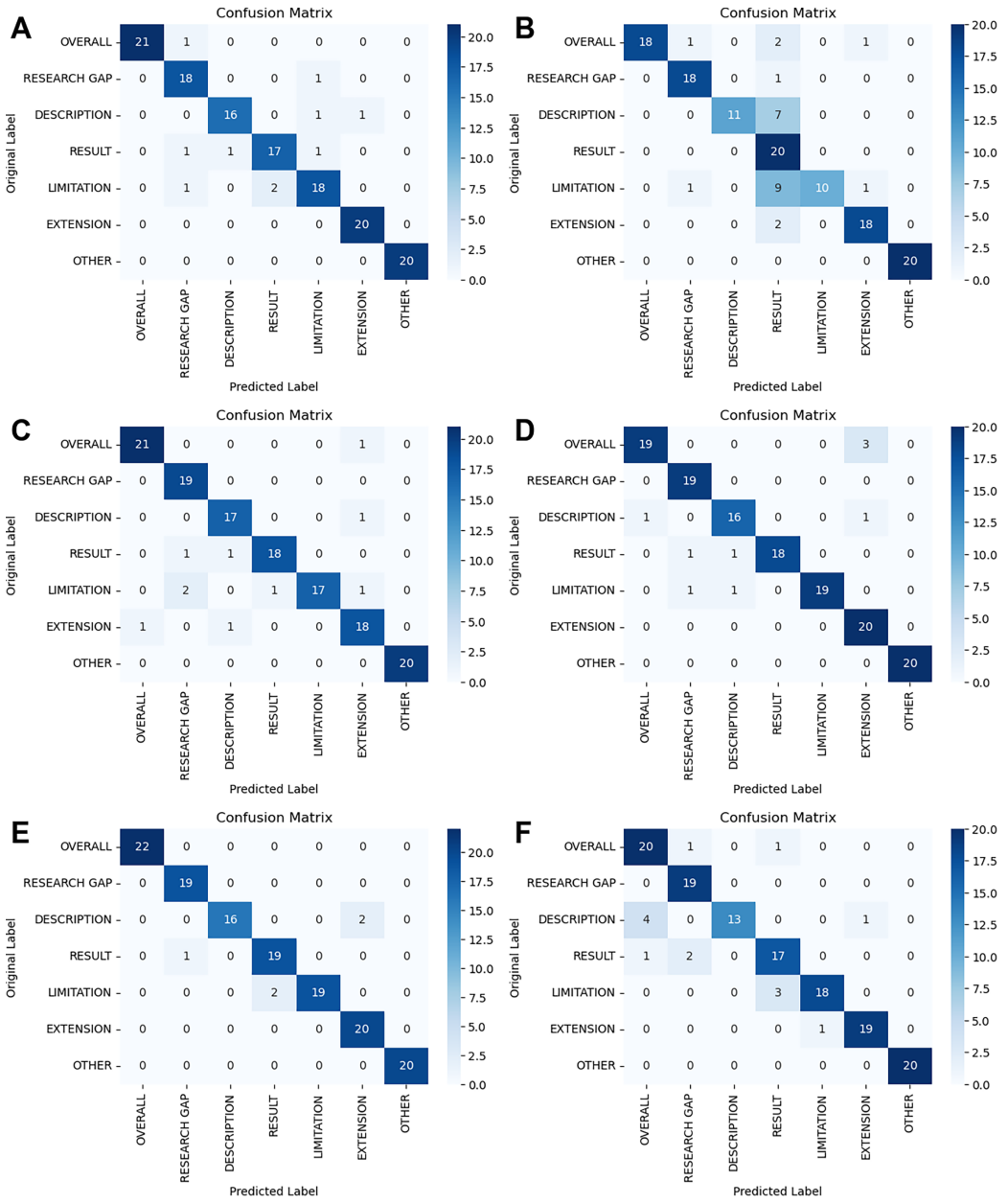

Figure 2 Confusion matrices of the best-performing models by model type. (A) Encoder: SciBERT (Augmented); (B) Decoder-ZSL: Sonnet; (C) Decoder Small: Gemma2-2B (LoRA-Augmented); (D) Decoder Medium: Nemotron-8B (LoRA-Augmented); (E) Decoder Large: GPT-4o-mini; (F) Encoder–Decoder: T5-XXL. Full-size DOI: 10.7717/peerj-cs.3921/fig-2

models, recorded 25 misclassifications. The remaining models had error counts ranging from nine to fourteen.

The majority of misclassification occurred within the `Limitation` and `Description` categories. Specifically, for the `Limitation` category, misclassification rates were 14% for SciBERT, 52% for Sonnet, 19% for Gemma2-2B, 10% for Nemotron3-8B, 10% for GPT-4o-mini, and 14% for T5. The `Description` category showed

misclassification rates of 11% (SciBERT), 39% (Sonnet), 11% (Nemotron3-8B), 11% (GPT-4o-mini), and 28% (T5).

Sentences annotated as `Limitation` in the gold standard were most often misclassified as `Research Gap` or `Result`.

A detailed analysis of these cases suggests that this occurs because instances of `Limitation` are often phrased in a way that implies a broader lack of knowledge. This phrasing can lead them to be interpreted as a `Research Gap`. For example, consider the sentence: *"Although the accuracy of current shape segmentation exceeds 90%, some models might still be poorly segmented and thus fail to be accurately landmarked"*. The intended meaning is to describe a limitation of the specific model being examined. However, the generalized expression "current shape segmentation" implies a shortcoming affecting the entire field rather than a precise constraint of the study. As a consequence, such statements may be misclassified as a `Research Gap`, since they seem to highlight an area where further research or improvement is needed across existing approaches instead of pointing to a limitation of the work at hand.

Moreover, some `Limitation` sentences explicitly refer to the results, which can lead to their misclassification as belonging to the `Result` category. For instance, the sentence *"Our encoding of traces does allow for uneven data samples, but it is probable that the performance will be reduced when sampling becomes irregular or infrequent"* can be misclassified as a `Result`, because it reports a finding about the model's ability to handle uneven data. However, it simultaneously conveys a limitation by acknowledging that performance may deteriorate under certain conditions, thus blurring the distinction between reporting results and discussing constraints. This characteristic of sentences belonging to multiple rhetorical types is a specific challenge that we aim to address in future refinement efforts.

On the other hand, sentences annotated as `Description` were most frequently misclassified as `Overall` or `Extension`. Our manual analysis indicates that some `Description` sentences can be confused with introductory clauses that appear to summarise the general status of a method or topic. This can lead to their incorrect assignment to the `Overall` category. This issue is exemplified by the sentence *"Specifically, as the first inductive kg embedding method, mean [8] learns to represent entities using their neighbors by simply mean-pooling the information of neighboring entity-relation pairs"*. The introductory phrase presents the method in a broad and contextual manner, highlighting its novelty or general contribution rather than providing a precise descriptive detail. This framing causes the sentence to resemble a high-level summary, which can lead to its misclassification as `Overall`. However, the latter part of the sentence (*"learns to represent entities using their neighbors…"*) in fact specifies how the method operates, which is characteristic of the `Description` category.

In other cases, `Description` outline the study's design in a way that resembles an expansion of previous methodology, aligning with the definition of the `Extension` category. For example, the sentence *"Telm [42] improves on TComplEx by utilizing a linear temporal regularizer and multi-vector embeddings to perform 4th-order tensor factorization of TKGs"* might appear to fit the `Extension` category because it references a previous

Table 10 Comparison of F1-scores when employing LORA or NEFT as optimisation techniques for optimizing decoder models. Bold values in the Diff. column indicate cases where NEFT achieved higher F1-scores than LoRA.

Model type	Model name	F1 (LORA)	F1 (NEFT)	Diff.
Decoder-small	Olmo-1B	0.902	0.921	**0.019**
	TinyLlama	0.552	0.702	**0.150**
	Arcee-lite	0.878	0.863	−0.015
	SmolLM2	0.796	0.782	−0.014
	Gemma2-2B	0.876	0.815	−0.061
	Llama3.2-3B	0.857	0.843	−0.014
	Phi3.5	0.861	0.869	**0.008**
Decoder-medium	Olmo-7B	0.900	0.929	**0.029**
	Mistral-7B	0.933	0.891	−0.042
	Arcee-Spark	0.872	0.891	**0.019**
	Llama3-8B	0.892	0.914	**0.022**
	Llama-3.1-SuperNova-Lite	0.928	0.922	−0.006
	Nemotron-8B	0.885	0.835	−0.050
Decoder-large	Gemma2-9B	0.942	0.891	−0.051
	Mistral-Nemo	0.907	0.89	−0.017
	SuperNova-Medius	0.902	0.87	−0.032

method. However, it should be classified as `Description`, since it does not explain how the current study elaborates on or extends the previous one. For a sentence to qualify as `Extension`, it must detail the nature of the elaboration, for example, by clarifying what limitations are addressed or what specific aspects of the prior study are developed further.

Comparing optimisation techniques

As discussed in 'Fine-Tuning Settings', the models were fine-tuned using two well-known techniques: LoRA and NEFT. In the previous sections, we always referred to the best-performing model between the two. However, it is also worth analysing in which cases one approach outperformed the other on Sci-Sentence benchmark.

Table 10 compares the F1-scores of the models trained with LoRA and NEFT. The "Diff." column reports the difference between the F1-score of the model using NEFT and that of the model using LoRA. Therefore, a positive value indicates that NEFT outperforms LoRA, whereas a negative value indicates that LoRA outperforms NEFT.

For large decoder models, LoRA consistently achieved slightly better results than NEFT. In the other categories, the outcomes are more mixed. Among medium decoder models, half of the models (Olmo 7B, Arcee-Spark, Llama3-8B) obtained F1-score improvements with NEFT, ranging from 0.019 to 0.029. For small decoder models, NEFT improved performance in three out of seven cases (Olmo 1B, TinyLlama, and Phi3.5), with TinyLlama exhibiting a notable F1-score increase of 0.150.

In conclusion, the evidence is insufficient to definitively establish the superiority of one optimization technique over the other for small and medium models. However, for large decoder models, LoRA produced more favourable results.

Assessing the efficacy of semi-synthetic data

Semi-synthetic training data produced by LLMs have proven to be very effective, but their performance is inconsistent across different classification tasks (*Li et al., 2023b*). One of our goals was to determine whether semi-synthetic data within *Sci-Sentence*, generated by creating alternative versions of manually classified sentences from the original data, could improve performance.

To this end, we trained each model on both the augmented and the original training data. As for the optimization techniques, in the previous section we always considered the best-performing model between the two configurations. Table 11 compares the F1-scores of the models trained on the augmented data and on the original data. The *Performance Gain* row refers to the difference between the F1-score obtained with the augmented data and the F1-score obtained with the original data. A positive value, therefore, indicates an improvement in performance due to the augmented data. It is important to note that Sonnet 3.0, which was used to generate the augmented datasets, was not included among the fine-tuning models in order to avoid potential issues of circularity (*Clarke & Dietz, 2024*).

An interesting insight is that encoder models benefit the most from augmented data, with all models improving in performance, sometimes very significantly. For example, the original BERT achieves a gain of over 27 percentage points in F1-score. Even a domain-specific model such as SciBERT shows a clear improvement, increasing from 87.0% to 92.8% thanks to the augmented data. This finding has important implications, particularly for applications that require a lightweight model to scalably annotate a large number of research articles, where a lightweight encoder model may therefore be preferred.

Encoder-decoder models also tend to benefit from augmented data, especially in their smaller variants, but not to the same extent as encoder models. In sum, our results of semi-synthetic data increasing the performance of encoders or encoder-decoder architectures align with existing literature (*Fields, Chovanec & Madiraju, 2024*; *Chae & Davidson, 2023*; *Fatemi, Hu & Mousavi, 2024*).

For decoder models, the performance gains are more variable. This observation is consistent with the literature: while some studies report a positive effect (*Singh et al., 2023*; *Li et al., 2024b*; *Zhezherau & Yanockin, 2024*; *Chen et al., 2024*), others find little or no benefit (*Guo et al., 2023*; *Zhao, Yin & Durrett, 2024*; *Li et al., 2024a*; *Mecklenburg et al., 2024*; *Maheshwari, Ivanov & Haddad, 2024*). The greatest benefits are observed in the smaller variants: three out of seven (namely TinyLlama, SmolLM2, and Gemma2-2B) show notable improvements, with TinyLlama achieving a substantial F1-score increase of over 30%. Among the medium-sized models, only two exhibit improvements. It is interesting to note that one of them is Nemotron-8B, which is also the best-performing model in this category.

For the large models, two out of four (Mistral-Nemo and SuperNova-Medius) demonstrate enhanced performance. Notably, SuperNova-Medius achieves a significant improvement of 4.1 percentage points, making it the best open model among all those tested. Conversely, GPT-4o-mini does not benefits from augmented data.

Table 11 Comparison of F1-score of models trained on the base or augmented benchmark. L, LoRA; N, NEFT. Bold values in the Performance Gain column indicate positive performance improvements obtained using the augmented benchmark.

Archit. Type	Model name	F1-score Base	F1-score Augmented	Performance Gain
Encoder	BERT	0.590	0.861	**0.271**
	SciBERT	0.870	0.928	**0.058**
	BioBERT	0.731	0.878	**0.147**
	BigBird	0.646	0.878	**0.232**
Decoder small	Olmo-1B [N]	0.921	0.912	−0.009
	TinyLlama [N]	0.702	0.879	**0.177**
	Arcee-Lite [L]	0.878	0.863	−0.015
	SmolLM2 [N]	0.782	0.914	**0.132**
	Gemma2-2B [L]	0.876	0.928	**0.052**
	Llama3.2-3B [L]	0.857	0.852	−0.005
	Phi3.5 [N]	0.869	0.831	−0.038
Decoder medium	Olmo-7B [N]	0.929	0.935	**0.006**
	Mistral-7B [L]	0.933	0.893	−0.040
	Arcee-Spark [N]	0.886	0.806	−0.080
	Llama3-8 [N]	0.914	0.901	−0.013
	Llama-3.1-SuperNova-Lite [L]	0.928	0.927	−0.001
	Nemotron-8B [L]	0.885	0.936	**0.051**
Decoder large	Gemma2-9B [L]	0.942	0.929	−0.013
	Mistral-Nemo [L]	0.907	0.929	**0.022**
	SuperNova-Medius [L]	0.902	0.943	**0.041**
	GPT-4o-mini	0.964	0.892	−0.072
Encoder decoder	T5	0.849	0.879	**0.030**
	T5 Large	0.875	0.892	**0.017**
	T5 xl	0.839	0.856	**0.017**
	T5 xxl	0.899	0.892	−0.007

In conclusion, while the improvements in decoder models are not consistent across all cases, those that do benefit tend to gain a substantial margin, allowing them to outperform other open alternatives in the same category. Indeed, in all categories, the best-performing open model was trained on augmented data. We can therefore conclude that augmented data can be highly beneficial for decoder models as well, although careful consideration is required to identify which decoders are most likely to benefit.

CONCLUSION

In this article, we propose a novel framework for classifying sentences in the related work or literature review sections of research articles into seven categories, extending previous research in this area. The goal is to develop an automated method for identifying sentences that present research gaps, limitations, extensions of previous work, and similar aspects, in order to support advanced retrieval-based systems for question answering and literature

review generation. We conduct a comprehensive evaluation of a wide range of encoder, encoder-decoder, and decoder language models with different architectures on this task. To facilitate this evaluation, we create and publicly release the *Sci-Sentence* benchmark, which includes a base version with 700 manually annotated sentences and an augmented version with a total of 2,940 sentences, combining both manually annotated and semi-synthetic samples.

These experiments provided several novel insights that significantly advance the state of the art in this space. First, the current generation of LLMs can perform remarkably well on this task when fine-tuned on high-quality datasets such as *Sci-Sentence*, achieving performance levels above 96% F1. Second, although large proprietary models such as GPT-4o (*Hurst et al., 2024*) achieve the highest performance, lightweight open-source alternatives, including SuperNova-Medius (https://huggingface.co/arcee-ai/SuperNova-Medius) and Nemotron-8B (*Zhang et al., 2023*), also deliver excellent results. Third, while decoder-only models attain the best overall performance, small and scalable encoder-based models pre-trained on relevant data, such as SciBERT (*Beltagy, Lo & Cohan, 2019*), remain highly competitive. This makes them a practical choice for processing large volumes of text efficiently. Finally, augmenting the original data with semi-synthetic examples generated by LLMs for fine-tuning has proven effective, particularly by enabling small encoders to achieve robust results and substantially improving the performance of several open decoders. In summary, the proposed framework, the *Sci-Sentence* benchmark, and our experimental results together constitute an important step and a foundational contribution to the "Related Work Generation" task.

Beyond the automatic generation of related work sections, the proposed framework can be repurposed to support several applications that assist researchers and students in understanding and navigating the scientific landscape. For instance, it can enhance generative AI systems for academic writing, which require a robust understanding of the types of statements and rhetorical functions that characterise literature reviews (*Bolanos et al., 2024*). It can also support tools for the generation of educational material, which are increasingly adopted in both universities and commercial settings (*Huang, Wei & Huang, 2024*). Furthermore, the framework provides a foundation for systems that characterise citations according to their intent and semantic role, thereby improving citation recommendation methods (*Cohan et al., 2019*). It may also facilitate hypothesis generation by enabling large-scale analyses of research gaps reported in the literature (*Spangler et al., 2014*; *Borrego et al., 2025*). In conclusion, the framework offers a general and extensible foundation that can be adapted to a broad range of structured scientific writing and analysis tasks that extend beyond traditional literature review generation.

It is important to acknowledge a few limitations of this study, which we aim to address in future work. First, our dataset consists predominantly of sentences extracted from Computer Science articles. As a consequence, the empirical results reported in this work may also reflect domain-specific writing conventions. Additional experiments on corpora from other scientific disciplines are therefore necessary to evaluate the external validity and generalisability of our findings. Second, the field of artificial intelligence is evolving rapidly, with new models being released continuously, and future advances in LLMs are likely to

substantially improve classification performance. However, the primary contribution of this work does not lie in achieving state-of-the-art results with a particular model. Rather, it consists in the formalisation of a robust and practically applicable annotation schema, together with a systematic evaluation of model configurations that enable reliable automatic annotation. Finally, additional research is needed to improve classifiers' ability to identify the most complex and challenging categories, such as `Description` and `Result`.

As future work, we plan to advance our research on multiple fronts. First, we aim to extend the classifier from single-label to multi-label in order to better capture the multifaceted nature of complex sentences that may pertain to multiple categories. Second, we plan to investigate how to capture more elusive categories, such as critiques and interpretations of a piece of literature, which are crucial for incorporating critical evaluations, nuanced perspectives, and broader contextual understanding into our representation. Although *Khoo, Na & Jaidka (2011)* identify an author's 'interpretation' category, we argue that a more fine-grained approach is necessary to effectively support the automatic generation of literature reviews, instead of aggregating all interpretations into a single category. Third, we plan to expand both the data volume and the range of covered fields, enabling large-scale experiments that will further validate and generalise our findings. Finally, we intend to develop a novel framework for generating automatic literature reviews that integrates the framework presented in this article and moves beyond uncritical multi-document summarisation towards producing high-quality, in-depth analyses of the literature.

APPENDIX

AUGMENTED DATA

In Fig. A1, we present the prompt used to generate the semi-synthetic data. Table A1 reports the average syntactic similarity for the training and validation sets. For each generated sentence, we calculate its Levenshtein distance to the original sentence, and the average Levenshtein distance to all other generated sentences.

PROMPT FOR ZERO-SHOT LEARNING

Figure B2 shows the prompt used in the Zero-Shot Learning experiments.

PARAMETERS SETTINGS FOR ZERO-SHOT LEARNING

Table A2 summarizes the settings used in the Zero-Shot Learning experiments.

LIST OF MODELS

This appendix outlines the complete list of LLMs tested in our experiments, grouped into seven categories: Decoder-Zero Shot Learning, Decoder-Small, Decoder-Medium, Decoder-Large, Encoder-Decoder, and Encoder.

Text Paraphrasing Instructions:

PROCEDURE:

1. Rephrase the given text preserving the main ideas and context.

2. Gives 4 different answers. Each answer should start and end with *.

EXAMPLE:

Text: Today is a sunny day.

Output for the text: *The weather is bright and cloudless on this day.* *Sunlight bathes the landscape as we enjoy clear skies.* *Warm rays illuminate our surroundings on this radiant afternoon.* *The sun's cheerful presence defines the atmosphere today.*

Text to Paraphrase : [your sentence].

Figure A1 Instructional template illustrating a text paraphrasing procedure, including stepwise guidelines and an example demonstrating generation of four distinct rephrased outputs from a single input sentence. Full-size DOI: 10.7717/peerj-cs.3921/fig-3

Table A1 Average syntactic similarity for the training and validation sets. Original refers to the original sentence. Other refers to the other synthetic sentences. Syn, Synthetic.

Data type	Category	Syn1-original	Syn1-other averaged	Syn2-original	Syn2-other averaged	Syn3-original	Syn3-other averaged	Syn4-original	Syn4-other averaged
Training	Average	0.57	0.56	0.54	0.57	0.54	0.56	0.53	0.55
	Overall	0.57	0.54	0.54	0.55	0.54	0.55	0.53	0.53
	Research gap	0.61	0.61	0.55	0.60	0.53	0.59	0.54	0.58
	Description	0.51	0.53	0.51	0.54	0.53	0.54	0.52	0.54
	Result	0.60	0.56	0.57	0.57	0.55	0.56	0.56	0.54
	Limitation	0.60	0.59	0.55	0.60	0.57	0.60	0.55	0.56
	Extension	0.54	0.55	0.53	0.55	0.50	0.55	0.51	0.52
	Other	0.59	0.57	0.54	0.57	0.53	0.57	0.54	0.56
Validation	Average	0.57	0.56	0.54	0.56	0.52	0.55	0.54	0.54
	Overall	0.54	0.54	0.53	0.54	0.50	0.52	0.50	0.50
	Research gap	0.66	0.63	0.62	0.62	0.61	0.58	0.58	0.60
	Description	0.47	0.50	0.49	0.54	0.45	0.51	0.48	0.53
	Result	0.52	0.50	0.50	0.52	0.46	0.49	0.46	0.50
	Limitation	0.68	0.67	0.57	0.64	0.63	0.67	0.64	0.63
	Extension	0.54	0.52	0.48	0.52	0.48	0.52	0.57	0.52
	Other	0.58	0.55	0.56	0.55	0.50	0.54	0.53	0.53

Decoder-zero shot learning

Full models

Llama 2 Chat 70B (https://llama.meta.com/llama2/) (shortened as llama2) has been trained using a dataset comprising 2 trillion tokens derived from publicly accessible online sources. It has 70 billion parameters and a context length of 4,096 tokens.

Text Classification Instructions:

Objective: Classify excerpts from scientific articles into one of the following seven categories based on their content. Each category corresponds to a specific aspect of scientific discourse, either related to a topic or a study. A topic is defined as a scientific domain, such as “Computer Science” or “Machine Learning”. A previous study refers to a prior paper on the topic.

CATEGORIES:

1. OVERALL: Describes, introduces, classifies, defines or contrasts research topics often based on the discussion of multiple previous studies together (e.g., “Performance estimation strategies can be generally divided into various methods.”).

2. RESEARCH GAP: Highlights the need for further research within the topic due to absence of information, insufficient information or contradictory information (e.g., “The study of psychopathic traits in digital worlds is under-explored.”).

3. DESCRIPTION: Outlines the objectives, methodology, or design of one previous study, without mentioning results.The outlining of this category is based only in one previous study (e.g., “The author developed an interactive web tool to explore music genres.”).

4. RESULT: Describes specific findings, outcomes, or conclusions drawn from previous studies. This category includes empirical results, theoretical insights, and observed patterns reported by researchers. It often uses verbs like "showed," "found," "demonstrated," or phrases like "the findings indicate" (e.g., “The author found limited effects of nudging on user choices.”).

5. LIMITATION: Describes a constraint, challenge, or weakness inherent in the methodology of a previous study or topic that hinders generalizability or reliability in a particular topic (e.g., “However, their manual creation is labor-intensive and time-consuming, posing obstacles to effective knowledge dissemination.”).

6. EXTENSION: Describes how the current study addresses or extends previous studies by stating the overall idea, contrasting ideas or elaborating further ideas. It usually uses the words we or our (e.g., “We aim to simplify website design modification through user-described changes.”).

7. OTHER: Any text that does not fit the above categories (e.g., “Thanks to Naver AI Lab for their support.”).

PROCEDURE:

1. Determine if the text pertains to a topic or a study.

2. Identify the most suitable category based on the content. Do not create new categories. Use the categories given above.

3. Provide the category number that best fits the text. Just provide the category number without any explanation.

Text to Classify : [your text]

Figure B2 Schematic overview of the annotation framework used for text classification. Illustrating seven predefined discourse categories (*e.g.*, overall, research gap, result, limitation) along with their definitions and the stepwise procedure for assigning category labels to scientific text excerpts.
Full-size DOI: 10.7717/peerj-cs.3921/fig-4

Table A2 Zero-short learning settings.

Model	Temperature	Top_k	Top_p	Platform
Llama 2	0	NA	0	Amazon bedrock
Llama 3 8b	0	NA	0	Amazon bedrock
Llama 3 70b	0	1	0	Amazon bedrock
Mistral	0	1	0	Amazon bedrock
Mixtral	0	1	0	Amazon bedrock
Mistral large	0	1	0	Amazon bedrock
Gemma	0	1	0	KoboldAI
Orca	0	1	0	KoboldAI
Sonnet	0	1	0	Amazon bedrock
Haiku	0	1	0	Amazon bedrock
GPT-3.5	0	1	0	OpenAI API
GPT-4	0	1	0	OpenAI API

Llama 3 8B Instruct (https://ai.meta.com/blog/meta-llama-3/) (shortened as llama3 8B full) is an auto-regressive language model that employs an optimised transformer architecture with 8 billion parameters. It possesses a context length of 8,000 tokens and has been trained on a dataset consisting of 15 trillion tokens. The training process involved supervised fine-tuning (SFT) in conjunction with reinforcement learning from human feedback (RLHF) to align the model with human preferences for utility and safety. To enhance inference scalability, the model incorporates Grouped-Query Attention (GQA).

Llama 3 70b Instruct (https://ai.meta.com/blog/meta-llama-3/) (shortened as llama3 70b) is composed of 70 billion parameters and a context length of 8,192 tokens. This model has been fine-tuned and optimised specifically for dialogue and chat use cases based.

Mistral 7b Instruct (*Jiang et al., 2023*) (shortened as mistral) is a 7 billion parameter model with a context length of 32,000 tokens. It incorporates architectural innovations such as Sliding Window Attention mechanism, GQA, and Byte-fallback Byte Pair Encoding (BPE) tokenizer. The first architectural innovation, accommodates a context length of 8,000 tokens and features a fixed cache size, which theoretically enables it to process up to 128,000 tokens. The second innovation, enhances inference speed while reducing cache size. While the third innovation, ensures reliable character recognition without the need for out-of-vocabulary tokens.

Mixtral 8X7b Instruct (*Jiang et al., 2024*) (shortened as mixtral) consists of 46.7 billion parameters and is capable of processing a context length of 32,000 tokens. It is based on a Sparse Mixture of Experts (SMoE) architecture with open weights. This model employs the same innovative architecture as the Mistral 7b Instruct. However, the Sliding Window Attention mechanism restricts it to handling a context length of 8,000 tokens.

Mistral Large (https://mistral.ai/news/mistral-large/) features a context length of 32,000 tokens and employs 123 billion parameters. The model was trained using a heterogeneous

dataset that encompassed a substantial amount of code, multilingual information, and content across a broad spectrum of topics.

Quantised models

Gemma2-2B-Instruct (https://huggingface.co/google/gemma-2b-it-GGUF) (shortened as Gemma) is a quantized version of Google's Gemma-2b-it language model (*Mesnard et al., 2024*) converted to the GPT-Generated Unified Format (GGUF) file format with 2 billion parameters and context length of 8,192 tokens.

Orca-2-13B (https://huggingface.co/TheBloke/Orca-2-13B-GGUF) (shortened as Orca) It is a quantized version of Microsoft's Orca 2 (*Mukherjee et al., 2023*), converted to the GGUF format having 13 billion parameters and a context length of 4,096 tokens (*Mitra et al., 2023*). It was fine-tuned on LLama 2 13B base model.

Proprietary models

The precise specifications of these models are confidential. Consequently, we can only provide limited information about them. In particular, details about their parameters are not disclosed.

Sonnet 3.0 (Sonnet) and **Haiku 3.0** (Haiku) are part of the Claude 3 series, a family of LLMs developed by *Anthropic (2024)*. Both models were trained on a proprietary *corpus* derived from both public and private sources. They have a context window of 200,000 tokens. Their distinction resides in the fact that Haiku is designed for immediate responses, while Sonnet 3.0 possesses the capability to manage complex tasks thanks to its architectural framework. These models adhered to the Constitutional AI framework, ensuring their alignment with the principles of helpfulness, honesty, and non-harmfulness

GPT-3.5 Turbo (GPT-3.5) and **GPT-4 Turbo** (GPT-4), both developed by OpenAI, exhibit specific distinctions in their design and capabilities (*Achiam et al., 2023*). GPT-3.5 Turbo operates with a context window of 16,000 tokens, whereas GPT-4 Turbo features a larger context window of 128,000 tokens. Furthermore, GPT-4 Turbo is capable of processing both textual and visual inputs, whereas GPT-3.5 Turbo is restricted to textual data exclusively. Additionally, GPT-4 was trained on a larger and more heterogeneous dataset compared to GPT-3.5 Turbo. Despite this, GPT-3.5 Turbo continues to be a practical and economical choice for numerous applications because of its effective combination of performance and efficiency.

Decoder-small FT

The number of parameters in this category ranges from 1 billion to 3.8 billion. However, the context window and the number of training tokens differ across models. For instance, Olmo-1B-Instruct **(Olmo-1B)** (*Groeneveld et al., 2024*), with 1 billion parameters, and TinyLlama-1.1B-Chat-v1.0 **(TinyLlama)** (*Zhang et al., 2024b*), with 1.1 billion parameters, share the same context window size of 2,048 tokens and were trained on 3 trillion tokens. In contrast, **SmolLM2** (https://github.com/huggingface/smollm/blob/main/README.md), which has 1.7 billion parameters, also employs a context window of 2,048 tokens but

was trained on a substantially larger dataset of 11 trillion tokens. These models also vary in their training sources: Olmo-1B was trained on subsets of Dolma v1.7 (*Soldaini et al., 2024*), TinyLlama utilized the architecture and tokenizer of Llama 2 (*Touvron et al., 2023*), and SmolLM2 was trained on a diverse dataset that includes textbooks, web content, code, mathematics, and external data sources.

On the other hand, Llama3.2-3-Instruct **(Llama3.2-3B)** (https://huggingface.co/meta-llama/Llama-3.2-3B-Instruct) was trained on a dataset comprising 15 trillion tokens, with a parameter count of 3 billion and a context window extending up to 8,000 tokens. In contrast, Phi3.5-mini-Instruct **(Phi3.5)** (https://huggingface.co/microsoft/Phi-3.5-mini-instruct) was trained on 3.4 trillion tokens, incorporating 3.8 billion parameters and a significantly larger context window of 128,000 tokens. Gemma2-2B-Instruct **(Gemma2-2B)** shares the same characteristics as those described in the category decoder OSL, differing only in its data source (https://huggingface.co/google/gemma-2-2b-it). The only merge model in this category is **Arcee-Lite** (https://huggingface.co/arcee-ai/arcee-lite) which have 1.5 billion parameters and was developed using Distilkit (https://github.com/arcee-ai/DistillKit). It supports a context window of 32,000 tokens and its distillation source is Phi-3-Medium (*Abdin et al., 2024*).

Decoder-medium FT

In this category the models have 7 billion or 8 billion parameters. For instance, Nemotron 3-8B-chat **(Nemotron3-8B)** (https://tinyurl.com/mw959vux) has 8 billion parameters and a context windows of 4,096 tokens. It was trained on 3.5 trillion tokens based on a large *corpus* of internet-scale data, including 53 languages and 37 coding languages. Similarly, models such as Mistral-7B-Instruct **(Mistral-7B)**, with 7 billion parameters, and Llama3-8-Instruct **(Llama3-8B)**, with 8 billion parameters, exhibit comparable features to their full-version counterparts but are distinguished by their quantized configurations. In the case of Olmo-7B-Instruct **(Olmo-7B)**, which also has 7 billion parameters, its distinction from Olmo-1B lies in its training dataset size of 2.5 trillion tokens.

For merged models, this category includes **Arcee-Spark** (https://huggingface.co/arcee-ai/Arcee-Spark) and Llama-3.1-SuperNova-Lite **(SuperNova-Lite)** (https://huggingface.co/arcee-ai/Llama-3.1-SuperNova-Lite). **Arcee-Spark**, initialized from the Qwen2-7B-Instruct (*Yang et al., 2024a*), comprises 7 billion parameters. On the other hand, **SuperNova-Lite**, with 8 billion parameters, is built upon the Llama-3.1-8B-Instruct (https://ai.meta.com/blog/meta-llama-3-1/) architecture and distilled from the Llama-3.1-405B-Instruct model. Both models have a context window of 128,000 tokens.

Decoder-large FT

This category includes models with parameters ranging from 9 billion to 14 billion. In this sense, Mistral-Nemo-Instruct-2407 **(Mistral-Nemo)** (https://mistral.ai/news/mistral-nemo/) has 12 billion parameters and a 128,000 token context window. Its training dataset incorporates a mix of multilingual text, code data, and conversational-style data to ensure high-quality input. Whereas, **SuperNova-Medius** (https://huggingface.co/arcee-ai/SuperNova-Medius) is a merged model of 14 billion parameters based on the

Qwen2.5-14B-Instruct architecture. It combines knowledge from both the Qwen2.5-72B-Instruct model and the Llama-3.1-405B-Instruct model through distillation. It has a context windows of 131,072 tokens. In contrast, Gemma2-9-Instruct **(Gemma2-9B)** has the same features as in Gemma, but differs on its source (https://huggingface.co/google/gemma-2-9b-it) and the number of parameters which are 9 billion. The only proprietary model is GPT-4o-mini-2024-07-18 **(GPT-4o-mini)** (https://openai.com/index/gpt-4o-mini-advancing-cost-efficient-intelligence/) which is part of the GPT-4o family and is designed to be a cost-efficient, high-performance AI model. It has multimodal capabilities and its context window of 128,000 tokens. It is designed to replace GPT-3.5 Turbo in ChatGPT due to its improved performance and cost-efficiency for various AI applications.

Encoder-decoder

For this category, we employed different versions of the **T5** (*Raffel et al., 2020*), including T5-base (222 million parameters), T5-large (770 million parameters), T5-xl (3 billion parameters), and T5-xxl (11 billion parameters). These models are built on a transformer architecture, wherein the encoder processes the input text, and the decoder generates the output text. T5 has been pretrained on the C4 *corpus*, a large dataset of text and code, using both supervised and self-supervised training methods. It has a context window of 512 tokens. Our decision to use T5 was driven by its superior performance compared to other encoder-decoder models (*Garg et al., 2021*; *Sarrouti, Tao & Randriamihaja, 2022*; *Kementchedjhieva & Chalkidis, 2023*).

Encoder

In this category, all models, with the exception of **BigBird** (*Zaheer et al., 2020*), exhibit identical features, including a number of parameters of 110 million, a context length of 512 tokens, and the utilisation of a full attention mechanism. In contrast, BigBird distinguishes itself with 125 million parameters, an extended context length of 4,096 tokens, and the use of a sparse attention mechanism.

BERT base (*Devlin et al., 2018*), hereafter referred to as **BERT**, was trained on a dataset containing 3.3 billion tokens sourced from Wikipedia and the Google Books *Corpus*. SciBERT case (**SciBERT**) (*Beltagy, Lo & Cohan, 2019*) was developed using 1.14 million scientific articles from Semantic Scholar (https://www.semanticscholar.org/), covering the biomedical and computer science domains, with a total of 3.1 billion tokens. **BioBERT** (*Lee et al., 2020*) was trained on an extensive collection of biomedical literature, including publications from PubMed (https://pubmed.ncbi.nlm.nih.gov/) and PMC (https://pmc.ncbi.nlm.nih.gov/), and employs WordPiece (*Wu, 2016*) tokenization to efficiently manage a large vocabulary. On the other hand, **BigBird** was trained on a dataset of 1.5 billion tokens drawn from the Books *Corpus* and Wikipedia.

PARAMETERS SETTINGS FOR FINE-TUNING

Table A3 reports the configuration settings employed for fine-tuning the models.

Table A3 Fine-tuning settings.

Model	LORA			NEFT	# Epochs	Platform
	r	Alpha	Droput	Alpha		
Olmo-1B	256	128	0.1	5	4	Hugging face
TinyLlama	256	128	0.1	5	1	Hugging face
Arcee-lite	256	128	0.05	5	1	Hugging face
SmolLM2	256	128	0.1	5	1	Hugging face
Gemma-2-2b	256	128	0.1	5	1	Unsloth
Llama3.2	256	128	0	5	1	Unsloth
Phi3.5	256	128	0	5	1	Unsloth
Olmo-7B	16	32	0.1	5	4	Hugging face
Mistral-7b	16	15	0	5	4	Unsloth
Arcee-Spark	256	128	0.1	5	1	Hugging face
Llama3-8b	256	128	0.1	5	1	Unsloth
Llama-3.1-SuperNova-Lite	256	128	0	5	1	Unsloth
Nemotron-8B	16	32	0.1	5	1	Hugging face
Gemma-2-9b	256	128	0.1	5	1	Unsloth
Mistral-Nemo	16	16	0	5	4	Unsloth
SuperNova-Medius-14B	256	128	0.1	5	1	Hugging face
gpt-4o-mini-2024-07-18	NA	NA	NA	NA	1	OpenAI API

ACKNOWLEDGEMENTS

During the preparation of this work, the authors used Grammarly, ChatGPT, and Gemini to check grammar and spelling. After using this tool/service, the authors reviewed and edited the content as needed and takes full responsibility for the content of the publication.

ADDITIONAL INFORMATION AND DECLARATIONS

Funding

The authors received no funding for this work.

Competing Interests

The authors declare that they have no competing interests.

Author Contributions

- Francisco Bolanos-Burgos conceived and designed the experiments, performed the experiments, analyzed the data, performed the computation work, prepared figures and/or tables, authored or reviewed drafts of the article, and approved the final draft.
- Angelo A. Salatino analyzed the data, prepared figures and/or tables, authored or reviewed drafts of the article, and approved the final draft.
- Francesco Osborne analyzed the data, authored or reviewed drafts of the article, and approved the final draft.

- Enrico Motta conceived and designed the experiments, authored or reviewed drafts of the article, and approved the final draft.

Data Availability

The following information was supplied regarding data availability:

The code and data are available at GitHub and Zenodo:

- https://github.com/fcobolanos/Classifying-the-Components-of-a-Literature-Review/tree/master.

- Francisco Bolanos. (2025). fcobolanos/Classifying-the-Components-of-a-Literature-Review: v1.0.0 (master). Zenodo. https://doi.org/10.5281/zenodo.17976195.

REFERENCES

Abdin M, Jacobs SA, Awan AA, Aneja J, Awadallah A, Awadalla HH, Bach N, Bahree A, Bakhtiari A, Behl HS, Benhaim A, Bilenko M, Bjorck J, Bubeck S, Cai M, Mendes CCT, Chen W, Chaudhary V, Chopra P, Giorno AD, de Rosa G, Dixon M, Eldan R, Fragoso V, Iter D, Goswami A, Gunasekar S, Haider E, Hao J, Hewett RJ, Huynh J, Javaheripi M, Jin X, Kauffmann P, Karampatziakis N, Kim D, Kim YJ, Khademi M, Kurilenko L, Lee JR, Lee YT, Li Y, Liang C, Liu W, Lin E, Lin Z, Madan P, Mitra A, Modi H, Nguyen AH, Norick B, Patra B, Perez-Becker D, Portet T, Pryzant R, Qin H, Radmilac M, Ren L, Rosset C, Roy S, Saarikivi O, Saied A, Salim A, Santacroce M, Shah S, Shang N, Sharma H, Song X, Ruwase O, Vaddamanu P, Wang X, Ward R, Wang G, Witte PA, Wyatt M, Xu C, Xu J, Yadav S, Yang F, Yang Z, Yu D, Yuan L, Zhang C, Zhang C, Zhang J, Zhang LL, Zhang Y, Zhang Y, Zhou X, Yang Y. 2024. Phi-3 technical report: a highly capable language model locally on your phone. ArXiv DOI 10.48550/arXiv.2404.14219.

AbuRa'ed A, Saggion H, Shvets A, Bravo À. 2020. Automatic related work section generation: experiments in scientific document abstracting. *Scientometrics* **125(3)**:3159–3185 DOI 10.1007/s11192-020-03630-2.

Achiam OJ, Adler S, Agarwal S, Ahmad L, Akkaya I, Aleman FL, de Almeida DM, Altenschmidt J, Altman S, Anadkat S, Avila R, Babuschkin I, Balaji S, Balcom V, Baltescu P, Bao H, Bavarian M, Belgum J, Bello I, Berdine J, Bernadett-Shapiro G, Berner C, Bogdonoff L, Boiko O, Boyd M, Brakman AL, Brockman G, Brooks T, Brundage M, Button K, Cai T, Campbell R, Cann A, Carey B, Carlson C, Carmichael R, Chan B, Chang C, Chantzis F, Chen D, Chen S, Chen R, Chen J, Chen M, Chess B, Cho C, Chu C, Chung HW, Cummings D, Currier J, Dai Y, Decareaux C, Degry T, Deutsch N, Deville D, Dhar A, Dohan D, Dowling S, Dunning S, Ecoffet A, Eleti A, Eloundou T, Farhi D, Fedus L, Felix N, Fishman SP, Forte J, Fulford I, Gao L, Georges E, Gibson C, Goel V, Gogineni T, Goh G, Gontijo-Lopes R, Gordon J, Grafstein M, Gray S, Greene R, Gross J, Gu SS, Guo Y, Hallacy C, Han J, Harris J, He Y, Heaton M, Heidecke J, Hesse C, Hickey A, Hickey W, Hoeschele P, Houghton B, Hsu K, Hu S, Hu X, Huizinga J, Jain S, Jain S, Jang J, Jiang A, Jiang R, Jin H, Jin D, Jomoto S, Jonn B, Jun H, Kaftan T, Kaiser L, Kamali A, Kanitscheider I, Keskar NS, Khan T, Kilpatrick L, Kim JW, Kim C, Kim Y, Kirchner H, Kiros JR, Knight M, Kokotajlo D, Kondraciuk L, Kondrich A, Konstantinidis A, Kosic K, Krueger G, Kuo V, Lampe M, Lan I, Lee T, Leike J, Leung J, Levy D, Li C, Lim R, Lin M, Lin SL, Litwin M, Lopez T, Lowe R, Lue P, Makanju A, Malfacini K, Manning S, Markov T, Markovski Y, Martin B, Mayer K, Mayne A, McGrew B, McKinney SM, McLeavey C, McMillan P, McNeil J, Medina D, Mehta A, Menick J, Metz L, Mishchenko A, Mishkin P, Monaco V, Morikawa E, Mossing DP, Mu T, Murati M, Murk O, M'ely D, Nair A, Nakano R,

Nayak R, Neelakantan A, Ngo R, Noh H, Long O, O'Keefe C, Pachocki JW, Paino A, Palermo J, Pantuliano A, Parascandolo G, Parish J, Parparita E, Passos A, Pavlov M, Peng A, Perelman A, de Avila Belbute Peres F, Petrov M, de Oliveira Pinto HP, Pokorny M, Pokrass M, Pong VH, Powell T, Power A, Power B, Proehl E, Puri R, Radford A, Rae JW, Ramesh A, et al. 2023. GPT-4 technical report. ArXiv DOI 10.48550/arXiv.2303.08774.

Agarwal S, Sahu G, Puri A, Laradji IH, Dvijotham KD, Stanley J, Charlin L, Pal C. 2025. LitLLMs, LLMs for literature review: are we there yet? *Transactions on Machine Learning Research* DOI 10.48550/arXiv.2412.15249.

Angioni S, Salatino A, Osborne F, Recupero DR, Motta E. 2021. AIDA: a knowledge graph about research dynamics in academia and industry. *Quantitative Science Studies* **2(4)**:1356–1398 DOI 10.1162/qss_a_00162.

Anthropic A. 2024. The claude 3 model family: opus, sonnet, haiku. Technical report, Anthropic.

Athar A. 2011. Sentiment analysis of citations using sentence structure-based features. In: *Proceedings of the ACL, 2011 Student Session*, 81–87.

Athar A, Teufel S. 2012. Context-enhanced citation sentiment detection. In: *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 597–601.

Auer S, Barone DAC, Bartz C, Cortes EG, Jaradeh MY, Karras O, Koubarakis M, Mouromtsev D, Pliukhin D, Radyush D, Shilin I, Stocker M, Tsalapati E. 2023. The SciQA scientific question answering benchmark for scholarly knowledge. *Scientific Reports* **13(1)**:7240 DOI 10.1038/s41598-023-33607-z.

Banerjee D, Awale S, Usbeck R, Biemann C. 2023. DBLP-QuAD: a question answering dataset over the DBLP scholarly knowledge graph. In: Frommholz I, Mayr P, Cabanac G, Verberne S, Brennan J, eds. *Proceedings of the 13th International Workshop on Bibliometric-enhanced Information Retrieval Co-located with 45th European Conference on Information Retrieval (ECIR 2023), Dublin, Ireland, April 2nd, 2023, Volume 3617 of CEUR Workshop Proceedings*. Germany: CEUR-WS.org, 37–51.

Bastola MN, Ho V. 2023. Rhetorical structure of literature review chapters in nepalese PhD dissertations: students' engagement with previous scholarship. *Journal of English for Academic Purposes* **65(1)**:101271 DOI 10.1016/j.jeap.2023.101271.

Beger C, Henneking C-L. 2025. Citegeist: automated generation of related work analysis on the arxiv corpus. ArXiv DOI 10.48550/arXiv.2503.23229.

Beltagy I, Lo K, Cohan A. 2019. SciBERT: a pretrained language model for scientific text. ArXiv DOI 10.48550/arXiv.1903.10676.

Berryman J, Ziegler A. 2024. *Prompt engineering for LLMs: the art and science of building large language model–based applications*. CA, USA: O'Reilly Media, Inc.

Berti-Équille L, Harmouch H, Naumann F, Novelli N, Thirumuruganathan S. 2018. Discovery of genuine functional dependencies from relational data with missing values. *Proceedings of the VLDB Endowment* **11(8)**:880–892 DOI 10.14778/3204028.3204032.

Birti M, Maurino A, Osborne F. 2025. Optimizing large language models for ESG activity detection in financial texts. In: *Proceedings of the 6th ACM International Conference on AI in Finance*, 856–863.

Bolanos F, Salatino A, Osborne F, Motta E. 2024. Artificial intelligence for literature reviews: opportunities and challenges. *Artificial Intelligence Review* **57(10)**:259 DOI 10.1007/s10462-024-10902-3.

Bolanos F, Salatino A, Osborne F, Motta E. 2025. Modelling and classifying the components of 1023 a literature review. ArXiv DOI 10.48550/arXiv.2508.04337.

Booth A, Martyn-St James M, Clowes M, Sutton A. 2021. *Systematic approaches to a 1025 successful literature review*. Third Edition. London: SAGE Publications Ltd.

Bornmann L, Mutz R. 2015. Growth rates of modern science: a bibliometric analysis based on the number of publications and cited references. *Journal of the Association for Information Science and Technology* **66(11)**:2215–2222 DOI 10.1002/asi.23329.

Borrego A, Dessì D, Ayala D, Hernández I, Osborne F, Recupero DR, Buscaldi D, Ruiz D, Motta E. 2025. Research hypothesis generation over scientific knowledge graphs. *Knowledge-Based Systems* **315(1)**:113280 DOI 10.1016/j.knosys.2025.113280.

Cadeddu A, Chessa A, De Leo V, Fenu G, Motta E, Osborne F, Recupero DR, Salatino A, Secchi L. 2024. A comparative analysis of knowledge injection strategies for large language models in the scholarly domain. *Engineering Applications of Artificial Intelligence* **133(2)**:108166 DOI 10.1016/j.engappai.2024.108166.

Cascini G, Fantechi A, Spinicci E. 2004. Natural language processing of patents and technical documentation. In: *International Workshop on Document Analysis Systems*. Cham: Springer, 508–520.

Chae Y, Davidson T. 2023. *Large language models for text classification: from zero-shot learning to fine-tuning*. Open Science Framework DOI 10.31235/osf.io/sthwk.

Chen X, Alamro H, Li M, Gao S, Yan R, Gao X, Zhang X. 2022. Target-aware abstractive related work generation with contrastive learning. In: *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 373–383.

Chen X, Alamro H, Li M, Gao S, Zhang X, Zhao D, Yan R. 2021. *Capturing relations between scientific papers: an abstractive model for related work section generation*. USA: Association for Computational Linguistics.

Chen H, Waheed A, Li X, Wang Y, Wang J, Raj B, Abdin MI. 2024. On the diversity of synthetic data and its impact on training large language models. ArXiv DOI 10.48550/arXiv.2410.15226.

Chen J, Zhuge H. 2019. Automatic generation of related work through summarizing citations. *Concurrency and Computation: Practice and Experience* **31(3)**:e4261 DOI 10.1002/cpe.4261.

Clarke CL, Dietz L. 2024. LLM-based relevance assessment still can't replace human relevance assessment. ArXiv DOI 10.48550/arXiv.2412.17156.

Cohan A, Ammar W, Van Zuylen M, Cady F. 2019. Structural scaffolds for citation intent classification in scientific publications. ArXiv DOI 10.48550/arXiv.1904.01608.

de Silva Júnior EM, Dutra M-L. 2021. A roadmap toward the automatic composition of systematic literature reviews. *Iberoamerican Journal of Science Measurement and Communication* **1(2)**:1–22 DOI 10.47909/ijsmc.52.

Deng Z, Zeng Z, Gu W, Ji J, Hua B. 2021. Automatic related work section generation by sentence extraction and reordering. In: *AII@ iConference*, 101–110.

Dessì D, Osborne F, Reforgiato Recupero D, Buscaldi D, Motta E, Sack H. 2020. AI-KG: an automatically generated knowledge graph of artificial intelligence. In: *The Semantic Web–ISWC 2020: 19th International Semantic Web Conference, Athens, Greece, November 2–6, 2020, Proceedings, Part II 19*. Cham: Springer, 127–143.

Dessí D, Osborne F, Recupero DR, Buscaldi D, Motta E. 2022a. SCICERO: a deep learning and NLP approach for generating scientific knowledge graphs in the computer science domain. *Knowledge-Based Systems* **258(10)**:109945 DOI 10.1016/j.knosys.2022.109945.

Dessí D, Osborne F, Reforgiato Recupero D, Buscaldi D, Motta E. 2022b. CS-KG: a large-scale knowledge graph of research entities and claims in computer science. In: *International Semantic Web Conference*. Cham: Springer, 678–696.

Dettmers T, Pagnoni A, Holtzman A, Zettlemoyer L. 2024. QLoRA: efficient finetuning of quantized LLMs. *Advances in Neural Information Processing Systems* **36**:10088–10115 DOI 10.52202/075280-0441.

Devlin J, Chang MW, Lee K, Toutanova K. 2018. BERT: pre-training of deep bidirectional transformers for language understanding. ArXiv DOI 10.48550/arXiv.1810.04805.

Dong C, Schäfer U. 2011. Ensemble-style self-training on citation classification. In: *Proceedings of 5th International Joint Conference on Natural Language Processing*, 623–631.

Fatemi S, Hu Y, Mousavi M. 2024. A comparative analysis of instruction fine-tuning large language models for financial text classification. *ACM Transactions on Management Information Systems* **16(1)**:1–30 DOI 10.1145/3706119.

Färber M, Lamprecht D, Krause J, Aung L, Haase P. 2023. SemOpenAlex: the scientific landscape in 26 billion RDF triples. In: *International Semantic Web Conference*. Cham: Springer, 94–112.

Fields J, Chovanec K, Madiraju P. 2024. A survey of text classification with transformers: how wide? How large? How long? How accurate? How expensive? How safe? *IEEE Access* **12**:6518 DOI 10.1109/ACCESS.2024.3349952.

Fleiss JL. 1971. Measuring nominal scale agreement among many raters. *Psychological Bulletin* **76(5)**:378 DOI 10.1037/h0031619.

Fu Y, Peng H, Ou L, Sabharwal A, Khot T. 2023. Specializing smaller language models towards multi-step reasoning. In: *International Conference on Machine Learning*. PMLR, 10421–10430.

Garg A, Adusumilli S, Yenneti S, Badal T, Garg D, Pandey V, Nigam A, Gupta YK, Mittal G, Agarwal R. 2021. News article summarization with pretrained transformer. In: *Advanced Computing: 10th International Conference, IACC 2020, Panaji, Goa, India, December 5–6, 2020, Revised Selected Papers, Part I 10*. Cham: Springer, 203–211.

Ge Y, Dinh L, Liu X, Su J, Lu Z, Wang A, Diesner J. 2021. BACO: a background knowledge-and content-based framework for citing sentence generation. In: *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, 1466–1478.

Groeneveld D, Beltagy I, Walsh P, Bhagia A, Kinney R, Tafjord O, Jha A, Ivison H, Magnusson I, Wang Y, Arora S, Atkinson D, Authur R, Chandu KR, Cohan A, Dumas J, Elazar Y, Gu Y, Hessel J, Khot T, Merrill W, Morrison JD, Muennighoff N, Naik A, Nam C, Peters ME, Pyatkin V, Ravichander A, Schwenk D, Shah S, Smith W, Strubell E, Subramani N, Wortsman M, Dasigi P, Lambert N, Richardson K, Zettlemoyer LS, Dodge J, Lo K, Soldaini L, Smith NA, Hajishirzi H. 2024. OLMo: accelerating the science of language models. ArXiv DOI 10.48550/arXiv.2402.00838.

Groza T. 2013. Using typed dependencies to study and recognise conceptualisation zones in biomedical literature. *PLOS ONE* **8(11)**:e79570 DOI 10.1371/journal.pone.0079570.

Gu N, Hahnloser R. 2024. Controllable citation sentence generation with language models. In: *Proceedings of the Fourth Workshop on Scholarly Document Processing (SDP 2024)*, 22–37.

Guo Y, Shang G, Vazirgiannis M, Clavel C. 2023. The curious decline of linguistic diversity: training language models on synthetic text. ArXiv DOI 10.48550/arXiv.2311.09807.

Gwet KL. 2008. Computing inter-rater reliability and its variance in the presence of high agreement. *British Journal of Mathematical and Statistical Psychology* **61(1)**:29–48 DOI 10.1348/000711006X126600.

He J, Zhou E, Sun L, Lei F, Liu C, Sun W. 2021. Semi-synthesis: a fast way to produce effective datasets for stereo matching. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2884–2893.

Hoang CDV, Kan M-Y. 2010. Towards automated related work summarization. In: *Coling 2010: Posters*, 427–435.

Hu EJ, Shen Y, Wallis P, Allen-Zhu Z, Li Y, Wang S, Wang L, Chen W. 2021. LoRA: low-rank adaptation of large language models. ArXiv DOI 10.48550/arXiv.2106.09685.

Hu Y, Wan X. 2014. Automatic generation of related work sections in scientific papers: an optimization approach. In: *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 1624–1633.

Huang C-Y, Wei J, Huang T-HK. 2024. Generating educational materials with different levels of readability using LLMs. In: *Proceedings of the Third Workshop on Intelligent and Interactive Writing Assistants*, 16–22.

Hurst A, Lerer A, Goucher AP, Perelman A, Ramesh A, Clark A, Ostrow A, Welihinda A, Hayes A, Radford A, Mkadry A, Baker-Whitcomb A, Beutel A, Borzunov A, Carney A, Chow A, Kirillov A, Nichol A, Paino A, Renzin A, Passos A, Kirillov A, Christakis A, Conneau A, Kamali A, Jabri A, Moyer A, Tam A, Crookes A, Tootoochian A, Tootoonchian A, Kumar A, Vallone A, Karpathy A, Braunstein A, Cann A, Codispoti A, Galu A, Kondrich A, Tulloch A, Mishchenko A, Baek A, Jiang A, toine Pelisse A, Woodford A, Gosalia A, Dhar A, Pantuliano A, Nayak A, Oliver A, Zoph B, Ghorbani B, Leimberger B, Rossen B, Sokolowsky B, Wang B, Zweig B, Hoover B, Samic B, McGrew B, Spero B, Giertler B, Cheng B, Lightcap B, Walkin B, Quinn B, Guarraci B, Hsu B, Kellogg B, Eastman B, Lugaresi C, Wainwright CL, Bassin C, Hudson C, Chu C, Nelson C, Li C, Shern CJ, Conger C, Barette C, Voss C, Ding C, Lu C, Zhang C, Beaumont C, Hallacy C, Koch C, Gibson C, Kim C, Choi C, McLeavey C, Hesse C, Fischer C, Winter C, Czarnecki C, Jarvis C, Wei C, Koumouzelis C, Sherburn D, Kappler D, Levin D, Levy D, Carr D, Farhi D, Mély D, Robinson D, Sasaki D, Jin D, Valladares D, Tsipras D, Li D, Nguyen PD, Findlay D, Oiwoh E, Wong E, Asdar E, Proehl E, Yang EM, Antonow E, Kramer E, Peterson E, Sigler E, Wallace E, Brevdo E, Mays E, Khorasani F, Such FP, Raso F, Zhang F, von Lohmann F, Sulit F, Goh G, Oden G, Salmon G, Starace G, Brockman G, Salman H, Bao H-B, Hu H, Wong H, Wang H, Schmidt H, Whitney H, woo Jun H, Kirchner H, de Oliveira Pinto HP, Ren H, Chang H-W, Chung HW, Kivlichan I, O'Connell I, Osband I, Silber I, Sohl I, Okuyucu I, Lan I, Kostrikov I, Sutskever I, Kanitscheider I, Gulrajani I, Coxon J, Menick J, Pachocki JW, Aung J, Betker J, Crooks J, Lennon J, Kiros JR, Leike J, Park J, Kwon J, Phang J, Teplitz J, Wei J, Wolfe J, Chen J, Harris J, Varavva J, Lee JG, Shieh J, Lin J, Yu J, Weng J, Tang J, Yu J, Jang J, Candela JQ, Beutler J, Landers J, Parish J, Heidecke J, Schulman J, Lachman J, Mckay J, Uesato J, Ward J, Kim JW, Huizinga J, Sitkin J, Kraaijeveld J, et al. 2024. GPT-4o system card. ArXiv DOI 10.48550/arXiv.2410.21276.

Jaidka K, Khoo CS, Na J-C. 2013. Literature review writing: how information is selected and transformed. *Aslib Proceedings* **65(3)**:303–325 DOI 10.1108/00012531311330665.

Jain N, yeh Chiang P, Wen Y, Kirchenbauer J, Chu H-M, Somepalli G, Bartoldson BR, Kailkhura B, Schwarzschild A, Saha A, Goldblum M, Geiping J, Goldstein T. 2023. NEFTune: noisy embeddings improve instruction finetuning. ArXiv DOI 10.48550/arXiv.2310.05914.

Jaradeh MY, Oelen A, Farfar KE, Prinz M, D'Souza J, Kismihók G, Stocker M, Auer S. 2019. Open research knowledge graph: next generation infrastructure for semantic scholarly knowledge. In: *Proceedings of the 10th International Conference on Knowledge Capture*, 243–246.

Jiang AQ, Sablayrolles A, Mensch A, Bamford C, Chaplot DS, de Las Casas D, Bressand F, Lengyel G, Lample G, Saulnier L, Lavaud LR, Lachaux M-A, Stock P, Scao TL, Lavril T, Wang T, Lacroix T, Sayed WE. 2023. Mistral 7b. ArXiv DOI 10.48550/arXiv.2310.06825.

Jiang AQ, Sablayrolles A, Roux A, Mensch A, Savary B, Bamford C, Chaplot DS, de Las Casas D, Hanna EB, Bressand F, Lengyel G, Bour G, Lample G, Lavaud LR, Saulnier L, Lachaux M-A, Stock P, Subramanian S, Yang S, Antoniak S, Scao TL, Gervet T, Lavril T, Wang T, Lacroix T, Sayed WE. 2024. Mixtral of experts. ArXiv DOI 10.48550/arXiv.2401.04088.

Joshi I, Grimmer M, Rathgeb C, Busch C, Bremond F, Dantcheva A. 2024. Synthetic data in human analysis: a survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **46**:4957 DOI 10.1109/TPAMI.2024.3362821.

Jung S-Y, Lin T-H, Liao C-H, Yuan S-M, Sun C-T. 2022. Intent-controllable citation text generation. *Mathematics* **10(10)**:1763 DOI 10.3390/math10101763.

Jurgens D, Kumar S, Hoover R, McFarland D, Jurafsky D. 2018. Measuring the evolution of a scientific field through citation frames. *Transactions of the Association for Computational Linguistics* **6(6)**:391–406 DOI 10.1162/tacl_a_00028.

Kaddour J, Liu Q. 2023. Synthetic data generation in low-resource settings via fine-tuning of large language models. ArXiv DOI 10.48550/arXiv.2310.01119.

Kanoksilapatham B. 2005. Rhetorical structure of biochemistry research articles. *English for Specific Purposes* **24(3)**:269–292 DOI 10.1016/j.esp.2004.08.003.

Kasanishi T, Isonuma M, Mori J, Sakata I. 2023. SciReviewGen: a large-scale dataset for automatic literature review generation. In: *Findings of the Association for Computational Linguistics: ACL 2023*, 6695–6715.

Kasneci E, Seßler K, Küchemann S, Bannert M, Dementieva D, Fischer F, Gasser U, Groh GL, Günnemann S, Hüllermeier E, Krusche S, Kutyniok G, Michaeli T, Nerdel C, Pfeffer J, Poquet O, Sailer M, Schmidt A, Seidel T, Stadler M, Weller J, Kuhn J, Kasneci G. 2023. Chatgpt for good? On opportunities and challenges of large language models for education. *Learning and Individual Differences* **103**:102274 DOI 10.1016/j.lindif.2023.102274.

Kementchedjhieva Y, Chalkidis I. 2023. An exploration of encoder-decoder approaches to multi-label classification for legal and biomedical text. ArXiv DOI 10.48550/arXiv.2305.05627.

Kenton JDM-WC, Toutanova LK. 2019. BERT: pre-training of deep bidirectional transformers for language understanding. In: *Proceedings of NAACL-HLT*. Vol. 1. Minneapolis, Minnesota, 2.

Khoo CS, Na J-C, Jaidka K. 2011. Analysis of the macro-level discourse structure of literature reviews. *Online Information Review* **35(2)**:255–271 DOI 10.1108/14684521111128032.

Kosonocky CW, Wilke CO, Marcotte EM, Ellington AD. 2024. Mining patents with large language models elucidates the chemical function landscape. *Digital Discovery* **3(6)**:1150–1159 DOI 10.1039/d4dd00011k.

Kuhn T, Chichester C, Krauthammer M, Queralt-Rosinach N, Verborgh R, Giannakopoulos G, Ngomo A-CN, Viglianti R, Dumontier M. 2016. Decentralized provenance-aware publishing with nanopublications. *PeerJ Computer Science* **2**:e78 DOI 10.7287/peerj.preprints.1760v1.

Kwan BS. 2006. The schematic structure of literature reviews in doctoral theses of applied linguistics. *English for Specific Purposes* **25(1)**:30–55 DOI 10.1016/j.esp.2005.06.001.

Landis JR, Koch GG. 1977. The measurement of observer agreement for categorical data. *Biometrics* **33(1)**:159–174.

Laurençon H, Tronchon L, Cord M, Sanh V. 2024. What matters when building vision-language models? ArXiv DOI 10.48550/arXiv.2405.02246.

Lauscher A, Ko B, Kuehl B, Johnson S, Jurgens D, Cohan A, Lo K. 2021. MultiCite: modeling realistic citations requires moving beyond the single-sentence single-label setting. ArXiv DOI 10.48550/arXiv.2107.00414.

Lee J, Yoon W, Kim S, Kim D, Kim S, So CH, Kang J. 2020. BioBERT: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics* **36(4)**:1234–1240 DOI 10.1093/bioinformatics/btz682.

Lehmann J, Meloni A, Motta E, Osborne F, Recupero DR, Salatino AA, Vahdati S. 2024. Large language models for scientific question answering: an extensive analysis of the SciQA benchmark. In: *European Semantic Web Conference*. Cham: Springer, 199–217.

Li Y, Bonatti R, Abdali S, Wagle J, Koishida K. 2024c. Data generation using large language models for text classification: an empirical case study. ArXiv DOI 10.48550/arXiv.2407.12813.

Li Y, Chen L, Liu A, Yu K, Wen L. 2024d. ChatCite: LLM agent with human workflow guidance for comparative literature summary. ArXiv DOI 10.48550/arXiv.2403.02574.

Li Y, De-Arteaga M, Saar-Tsechansky M. 2022. When more data lead us astray: active data acquisition in the presence of label bias. *Proceedings of the AAAI Conference on Human Computation and Crowdsourcing* **10(1)**:133–146 DOI 10.1609/hcomp.v10i1.21994.

Li X, Lee Y-H, Ouyang J. 2023. Cited text spans for citation text generation. ArXiv DOI 10.48550/arXiv.2309.06365.

Li B, Liang H, Li Y, Fu F, Yin H, He C, Zhang W. 2024a. Gradual learning: optimizing fine-tuning with partially mastered knowledge in large language models. ArXiv DOI 10.48550/arXiv.2410.05802.

Li X, Mandal B, Ouyang J. 2022. CORWA: a citation-oriented related work annotation dataset. ArXiv DOI 10.48550/arXiv.2205.03512.

Li X, Ouyang J. 2022. Automatic related work generation: a meta study. ArXiv DOI 10.48550/arXiv.2201.01880.

Li X, Ouyang J. 2024a. Explaining relationships among research papers. ArXiv DOI 10.48550/arXiv.2402.13426.

Li X, Ouyang J. 2024b. Related work and citation text generation: a survey. ArXiv DOI 10.48550/arXiv.2404.11588.

Li Y, Wang S, Ding H, Chen H. 2023a. Large language models in finance: a survey. In: *Proceedings of the Fourth ACM International Conference on AI in Finance*, 374–382.

Li Y, Wei F, Zhang C, Zhang H. 2024e. Eagle: speculative sampling requires rethinking feature uncertainty. ArXiv DOI 10.48550/arXiv.2401.15077.

Li J, Zhu X, Liu F, Qi Y. 2024b. Aide: task-specific fine tuning with attribute guided multi-hop data expansion. ArXiv DOI 10.48550/arXiv.2412.06136.

Li Z, Zhu H, Lu Z, Yin M. 2023b. Synthetic data generation with large language models for text classification: potential and limitations. ArXiv DOI 10.48550/arXiv.2310.07849.

Liakata M, Teufel S, Siddharthan A, Batchelor CR. 2010. Corpora for the conceptualisation and zoning of scientific papers. In: *International Conference on Language Resources and Evaluation*.

Liu X, Song R, Wang X, Chen X. 2025. Select, read, and write: a multi-agent framework of full-text-based related work generation. ArXiv DOI 10.48550/arXiv.2505.19647.

Liu R, Wei J, Liu F, Si C, Zhang Y, Rao J, Zheng S, Peng D, Yang D, Zhou D, Dai AM. 2024a. Best practices and lessons learned on synthetic data for language models. ArXiv DOI 10.48550/arXiv.2404.07503.

Liu J, Zhang Q, Shi C, Naseem U, Wang S, Tsang I. 2023. Causal intervention for abstractive related work generation. ArXiv DOI 10.48550/arXiv.2305.13685.

Liu Z, Zhao C, Iandola FN, Lai C, Tian Y, Fedorov I, Xiong Y, Chang E, Shi Y, Krishnamoorthi R, Lai L, Chandra V. 2024b. MobileLLM: optimizing sub-billion parameter language models for on-device use cases. ArXiv DOI 10.48550/arXiv.2402.14905.

Luu K, Wu X, Koncel-Kedziorski R, Lo K, Cachola I, Smith NA. 2020. Explaining relationships between scientific documents. ArXiv DOI 10.48550/arXiv.2002.00317.

Ma Y, Qing L, Kang Y, Liu J, Zhang Y, Cheng Q, Lu W, Liu X. 2025. Refinement and revision in academic writing: integrating multi-source knowledge and LLMs with delta feedback. *Expert Systems with Applications* **277(4)**:127226 DOI 10.1016/j.eswa.2025.127226.

Maheshwari G, Ivanov D, Haddad KE. 2024. Efficacy of synthetic data as a benchmark. ArXiv DOI 10.48550/arXiv.2409.11968.

Mann WC, Thompson SA. 1988. Rhetorical structure theory: toward a functional theory of text organization. *Text-Interdisciplinary Journal for the Study of Discourse* **8(3)**:243–281 DOI 10.1515/text.1.1988.8.3.243.

Mao Y, Ge Y, Fan Y, Xu W, Mi Y, Hu Z, Gao Y. 2024. A survey on lora of large language models. ArXiv DOI 10.48550/arXiv.2407.11046.

Martin-Boyle A, Tyagi A, Hearst MA, Kang D. 2024. Shallow synthesis of knowledge in GPT-generated texts: a case study in automatic related work composition. ArXiv DOI 10.48550/arXiv.2402.12255.

Marwala T, Fournier-Tombs E, Stinckwich S. 2023. The use of synthetic data to train ai models: opportunities and risks for sustainable development. ArXiv DOI 10.48550/arXiv.2309.00652.

Masoumi N, Khajavi R. 2023. A fuzzy classifier for evaluation of research topics by using keyword co-occurrence network and sponsors information. *Scientometrics* **128(3)**:1485–1512 DOI 10.1007/s11192-022-04618-w.

McCoubrey M, Salatino A, Osborne F, Motta E. 2026. How do LLMs encode scientific quality? An empirical study using monosemantic features from sparse autoencoders. ArXiv DOI 10.48550/arXiv.2602.19115.

Mecklenburg N, Lin Y, Li X, Holstein D, Nunes L, Malvar S, Silva BLB, Chandra R, Aski V, Yannam PKR, Aktas T, Hendry T. 2024. Injecting new knowledge into large language models via supervised fine-tuning. ArXiv DOI 10.48550/arXiv.2404.00213.

Meloni A, Angioni S, Salatino A, Osborne F, Recupero DR, Motta E. 2023. Integrating conversational agents and knowledge graphs within the scholarly domain. *IEEE Access* **11(12)**:22468–22489 DOI 10.1109/access.2023.3253388.

Mesnard T, Hardin C, Dadashi R, Bhupatiraju S, Pathak S, Sifre L, Rivière M, Kale M, Love JC, Tafti PD, Hussenot L, Chowdhery A, Roberts A, Barua A, Botev A, Castro-Ros A, Slone A, H'eliou A, Tacchetti A, Bulanova A, Paterson A, Tsai B, Shahriari B, Lan CL, Choquette-Choo CA, Crepy C, Cer D, Ippolito D, Reid D, Buchatskaya E, Ni E, Noland E, Yan G, Tucker G, Muraru G-C, Rozhdestvenskiy G, Michalewski H, Tenney I, Grishchenko I, Austin J, Keeling J, Labanowski J, Lespiau J-B, Stanway J, Brennan J, Chen J, Ferret J, Chiu J, Mao-Jones J, Lee K, Yu K, Millican K, Sjoesund LL, Lee L, Dixon L, Reid M, Mikuła M, Wirth M, Sharman M, Chinaev N, Thain N, Bachem O, Chang O, Wahltinez O, Bailey P, Michel P, Yotov P, Sessa PG, Chaabouni R, Comanescu R, Jana R, Anil R, Mcilroy R, Liu R, Mullins R, Smith SL, Borgeaud S, Girgin S, Douglas S, Pandya S, Shakeri S, De S, Klimenko T, Hennigan T, Feinberg V, Stokowiec W, Chen Y-H, Ahmed Z, Gong Z, Warkentin T, Peran L, Giang M, Farabet C, Vinyals O, Dean J, Kavukcuoglu K, Hassabis D, Ghahramani Z, Eck D, Barral J, Pereira F, Collins E, Joulin A, Fiedel N, Senter E, Andreev A, Kenealy K. 2024. Gemma: open models based on gemini research and technology. ArXiv DOI 10.48550/arXiv.2403.08295.

Mitra A, Corro LD, Mahajan S, Codas A, Simoes C, Agrawal S, Chen X, Razdaibiedina A, Jones E, Aggarwal K, Palangi H, Zheng G, Rosset C, Khanpour H, Awadallah A. 2023. Orca 2: teaching small language models how to reason. ArXiv DOI 10.48550/arXiv.2311.11045.

Mizuta Y, Korhonen A, Mullen T, Collier N. 2006. Zone analysis in biology articles as a basis for information extraction. *International Journal of Medical Informatics* **75(6)**:468–487 DOI 10.1016/j.ijmedinf.2005.06.013.

Moher D, Tsertsvadze A, Tricco AC, Eccles M, Grimshaw J, Sampson M, Barrowman N. 2007. A systematic review identified few methods and strategies describing when and how to update systematic reviews. *Journal of Clinical Epidemiology* **60(11)**:1095–e1 DOI 10.1016/j.jclinepi.2007.03.008.

Motta E, Osborne F, Pulici MM, Salatino A, Naja I. 2024. Capturing the viewpoint dynamics in the news domain. In: *International Conference on Knowledge Engineering and Knowledge Management*. Cham: Springer, 18–34.

Mukherjee S, Mitra A, Jawahar G, Agarwal S, Palangi H, Awadallah A. 2023. Orca: progressive learning from complex explanation traces of GPT-4. ArXiv DOI 10.48550/arXiv.2306.02707.

Myneni S, Jha K, Sabur A, Agrawal G, Deng Y, Chowdhary A, Huang D. 2023. Unraveled—a semi-synthetic dataset for advanced persistent threats. *Computer Networks* **227(2)**:109688 DOI 10.1016/j.comnet.2023.109688.

Nishimura K, Saito K, Hirasawa T, Ushiku Y. 2024. Toward structured related work generation with novelty statements. In: Ghosal T, Singh A, Waard A, Mayr P, Naik A, Weller O, Lee Y, Shen S, Qin Y, eds. *Proceedings of the Fourth Workshop on Scholarly Document Processing (SDP 2024)*. Bangkok, Thailand: Association for Computational Linguistics, 38–57.

Omiye JA, Gui H, Rezaei SJ, Zou J, Daneshjou R. 2024. Large language models in medicine: the potentials and pitfalls: a narrative review. *Annals of Internal Medicine* **177(2)**:210–220 DOI 10.7326/m23-2772.

Patel A, Bhattamishra S, Reddy S, Bahdanau D. 2023. MAGNIFICo: evaluating the in-context learning ability of large language models to generalize to novel interpretations. ArXiv DOI 10.48550/arXiv.2310.11634.

Peng C, Xia F, Naseriparsa M, Osborne F. 2023. Knowledge graphs: opportunities and challenges. *Artificial Intelligence Review* **56(11)**:13071–13102 DOI 10.1007/s10462-023-10465-9.

Pradeep R, Sharifymoghaddam S, Lin J. 2023. RankZephyr: effective and robust zero-shot listwise reranking is a breeze!. ArXiv DOI 10.48550/arXiv.2312.02724.

Prasad R, Dinesh N, Lee A, Miltsakaki E, Robaldo L, Joshi AK, Webber BL. 2008. The Penn discourse treebank 2.0. In: *International Conference on Language Resources and Evaluation*.

Raffel C, Shazeer N, Roberts A, Lee K, Narang S, Matena M, Zhou Y, Li W, Liu PJ. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research* **21(140)**:1–67 DOI 10.5555/3455716.3455856.

Ravi K, Setlur S, Ravi V, Govindaraju V. 2018. Article citation sentiment analysis using deep learning. In: *2018 IEEE 17th International Conference on Cognitive Informatics & Cognitive Computing (ICCI* CC)*. Piscataway: IEEE, 78–85.

Salatino A, Aggarwal T, Mannocci A, Osborne F, Motta E. 2025. A survey of knowledge organization systems of research fields: resources and challenges. *Quantitative Science Studies* **6(4)**:1–44 DOI 10.1162/qss_a_00363.

Salatino A, Osborne F, Motta E. 2020. ResearchFlow: understanding the knowledge flow between academia and industry. In: Keet CM, Dumontier M, eds. *Knowledge Engineering and Knowledge Management*. Cham: Springer International Publishing, 219–236.

Salatino A, Osborne F, Motta E. 2022. CSO classifier 3.0: a scalable unsupervised method for classifying documents in terms of research topics. *International Journal on Digital Libraries* **23(1)**:91–110 DOI 10.1007/s00799-021-00305-y.

Salatino A, Osborne F, Recupero DR, Angioni S, Motta E. 2026. Does diversity of expertise drive citation impact? Evidence from computer science. *Scientometrics* **131(2)**:1–28 DOI 10.1007/s11192-026-05560-x.

Sarrouti M, Tao C, Randriamihaja YM. 2022. Comparing encoder-only and encoder-decoder transformers for relation extraction from biomedical texts: an empirical study on ten benchmark datasets. In: *Proceedings of the 21st Workshop on Biomedical Language Processing*, 376–382.

Savelka J, Ashley KD. 2023. The unreasonable effectiveness of large language models in zero-shot semantic annotation of legal texts. *Frontiers in Artificial Intelligence* **6**:1279794 DOI 10.3389/frai.2023.1279794.

Shi Z, Gao S, Zhang Z, Chen X, Chen Z, Ren P, Ren Z. 2023. Towards a unified framework for reference retrieval and related work generation. In: *Findings of the Association for Computational Linguistics: EMNLP 2023*, 5785–5799.

Singh A, Co-Reyes JD, Agarwal R, Anand A, Patil P, Liu PJ, Harrison J, Lee J, Xu K, Parisi A, Kumar A, Alemi A, Rizkowsky A, Nova A, Adlam B, Bohnet B, Elsayed G, Sedghi H, Mordatch I, Simpson I, Gur I, Snoek J, Pennington J, Hron J, Kenealy K, Swersky K, Mahajan K, Culp L, Xiao L, Bileschi ML, Constant N, Novak R, Liu R, Warkentin T, Qian Y, Bansal Y, Dyer E, Neyshabur B, Sohl-Dickstein JN, Fiedel N. 2023. Beyond human data: scaling self-training for problem-solving with language models. *Transactions on Machine Learning Research* DOI 10.48550/arXiv.2312.06585.

Soldaini L, Kinney R, Bhagia A, Schwenk D, Atkinson D, Authur R, Bogin B, Chandu KR, Dumas J, Elazar Y, Hofmann V, Jha A, Kumar S, Lucy L, Lyu X, Lambert N, Magnusson I, Morrison JD, Muennighoff N, Naik A, Nam C, Peters ME, Ravichander A, Richardson K, Shen SZ, Strubell E, Subramani N, Tafjord O, Walsh P, Zettlemoyer LS, Smith NA, Hajishirzi H, Beltagy I, Groeneveld D, Dodge J, Lo K. 2024. Dolma: an open corpus of three trillion tokens for language model pretraining research. ArXiv DOI 10.48550/arXiv.2402.00159.

Song R, Li Y, Giunchiglia F, Xu H. 2026. Retrieval augmentation for out-of-distribution robustness in non-knowledge intensive in-context learning. *Science China Information Sciences* **69(2)**:122108 DOI 10.1007/s11432-025-4558-9.

Song R, Li Y, Tian M, Wang H, Giunchiglia F, Xu H. 2025. Causal keyword driven reliable text classification with large language model feedback. *Information Processing & Management* **62(2)**:103964 DOI 10.1016/j.ipm.2024.103964.

Spangler S, Wilkins AD, Bachman BJ, Nagarajan M, Dayaram T, Haas PJ, Regenbogen S, Pickering CR, Comer A, Myers JN, Stanoi I, Kato L, Lelescu A, Labrie JJ, Parikh N, Lisewski AM, Donehower LA, Chen Y, Lichtarge O. 2014. Automated hypothesis generation based on mining scientific literature. In: *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 1877–1886.

Stan GBM, Aflalo E, Madasu A, Lal V, Howard P. 2025. Learning from reasoning failures via synthetic data generation. ArXiv DOI 10.48550/arXiv.2504.14523.

Stevens ME, Giuliano VE, Garfield E. 1965. Can citation indexing be automated. In: *Statistical Association Methods for Mechanized Documentation, Symposium Proceedings*. Vol. 269, Citeseer, 189–192.

Swales JM, Feak CB. 2004. *Academic writing for graduate students: essential tasks and skills*. Vol. 1. MI: University of Michigan Press Ann Arbor.

Swales JM, Swales J. 1990. *Genre analysis*. UK: Cambridge university press.

Taboada M, Mann WC. 2006. Applications of rhetorical structure theory. *Discourse Studies* **8(4)**:567–588 DOI 10.1177/1461445606064836.

Teufel S, Carletta J, Moens M. 1999. An annotation scheme for discourse-level argumentation in research articles. In: *Ninth Conference of the European Chapter of the Association for Computational Linguistics*, 110–117.

Teufel S, Moens M. 2002. Articles summarizing scientific articles: experiments with relevance and rhetorical status. *Computational Linguistics* **28(4)**:409–445 DOI 10.1162/089120102762671936.

Teufel S, Siddharthan A, Batchelor C. 2009. Towards domain-independent argumentative zoning: evidence from chemistry and computational linguistics. In: *Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing*, 1493–1502.

Teufel S, Siddharthan A, Tidhar D. 2006. Automatic classification of citation function. In: *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*, 103–110.

Touvron H, Martin L, Stone KR, Albert P, Almahairi A, Babaei Y, Bashlykov N, Batra S, Bhargava P, Bhosale S, Bikel DM, Blecher L, Ferrer CC, Chen M, Cucurull G, Esiobu D, Fernandes J, Fu J, Fu W, Fuller B, Gao C, Goswami V, Goyal N, Hartshorn AS, Hosseini S, Hou R, Inan H, Kardas M, Kerkez V, Khabsa M, Kloumann IM, Korenev AV, Koura PS, Lachaux M-A, Lavril T, Lee J, Liskovich D, Lu Y, Mao Y, Martinet X, Mihaylov T, Mishra P, Molybog I, Nie Y, Poulton A, Reizenstein J, Rungta R, Saladi K, Schelten A, Silva R, Smith EM, Subramanian R, Tan X, Tang B, Taylor R, Williams A, Kuan JX, Xu P, Yan Z, Zarov I, Zhang Y, Fan A, Kambadur M, Narang S, Rodriguez A, Stojnic R, Edunov S, Scialom T. 2023. Llama 2: open foundation and fine-tuned chat models. ArXiv DOI 10.48550/arXiv.2307.09288.

Tuarob S, Kang SW, Wettayakorn P, Pornprasit C, Sachati T, Hassan S-U, Haddawy P. 2019. Automatic classification of algorithm citation functions in scientific literature. *IEEE Transactions on Knowledge and Data Engineering* **32(10)**:1881–1896 DOI 10.1109/tkde.2019.2913376.

Wang P, Li S, Tang J, Wang T. 2024b. What can rhetoric bring us? Incorporating rhetorical structure into neural related work generation. *Expert Systems with Applications* **251**:123781 DOI 10.1016/j.eswa.2024.123781.

Wang P, Li S, Zhou H, Tang J, Wang T. 2019. ToC-RWG: explore the combination of topic model and citation information for automatic related work generation. *IEEE Access* **8**:13043–13055 DOI 10.1109/access.2019.2959056.

Wang Y, Liu X, Gao Z. 2019. Neural related work summarization with a joint context-driven attention mechanism. ArXiv DOI 10.48550/arXiv.1901.09492.

Wang Y, Ma X, Nie P, Zeng H, Lyu Z, Zhang Y, Schneider B, Lu Y, Yue X, Chen W. 2025. ScholarCopilot: training large language models for academic writing with accurate citations. ArXiv DOI 10.48550/arXiv.2504.00824.

Wang X, Song N, Zhou H, Cheng H. 2022. The representation of argumentation in scientific papers: a comparative analysis of two research areas. *Journal of the Association for Information Science and Technology* **73(6)**:863–878 DOI 10.1002/asi.24590.

Wang F, Zhang Z, Zhang X, Wu Z, Mo T, Lu Q, Wang W, Li R, Xu J, Tang X, He Q, Ma Y, Huang M, Wang S. 2024a. A comprehensive survey of small language models in the era of large language models: techniques, enhancements, applications, collaboration with LLMs, and trustworthiness. *ACM Transactions on Intelligent Systems and Technology* **16(6)**:1–87 DOI 10.1145/3768165.

Widyantoro DH, Khodra ML, Riyanto B, Aziz EA. 2013. A multiclass-based classification strategy for rhetorical sentence categorization from scientific papers. *Journal of ICT Research and Applications* **7(3)**:235–249 DOI 10.5614/itbj.ict.res.appl.2013.7.3.5.

Wu Y. 2016. Google's neural machine translation system: bridging the gap between human and machine translation. ArXiv DOI 10.48550/arXiv.1609.08144.

Wu S, Irsoy O, Lu S, Dabravolski V, Dredze M, Gehrmann S, Kambadur P, Rosenberg D, Mann G. 2023. BloombergGPT: a large language model for finance. ArXiv DOI 10.48550/arXiv.2303.17564.

Xing X, Fan X, Wan X. 2020. Automatic generation of citation texts in scholarly papers: a pilot study. In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 6181–6190.

Yale A, Dash S, Dutta R, Guyon I, Pavao A, Bennett KP. 2019. Privacy preserving synthetic health data. In: *ESANN 2019-European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning*.

Yang A, Yang B, Hui B, Zheng B, Yu B, Zhou C, Li C, Li C, Liu D, Huang F, Dong G, Wei H, Lin H, Tang J, Wang J, Yang J, Tu J, Zhang J, Ma J, Xu J, Zhou J, Bai J, He J, Lin J, Dang K, Lu K, Chen K-Y, Yang K, Li M, Xue M, Ni N, Zhang P, Wang P, Peng R, Men R, Gao R, Lin R, Wang S, Bai S, Tan S, Zhu T, Li T, Liu T, Ge W, Deng X, Zhou X, Ren X, Zhang X, Wei X, Ren X, Fan Y, Yao Y, Zhang Y, Wan Y, Chu Y, Cui Z, Zhang Z, Fan ZW. 2024a. Qwen2 technical report. ArXiv DOI 10.48550/arXiv.2407.10671.

Yang K, Zhang T, Kuang Z, Xie Q, Huang J, Ananiadou S. 2024b. MentallaMA: interpretable mental health analysis on social media with large language models. In: *Proceedings of the ACM Web Conference 2024*, 4489–4500.

Zaheer M, Guruganesh G, Dubey KA, Ainslie J, Alberti C, Ontañón S, Pham P, Ravula A, Wang Q, Yang L, Ahmed A. 2020. Big bird: transformers for longer sequences. *Advances in Neural Information Processing Systems* **33**:17283–17297.

Zhang J, Chen J, Maatouk A, Bui N, Xie Q, Tassiulas L, Shao J, Xu H, Ying R. 2024a. LitFM: a retrieval augmented structure-aware foundation model for citation graphs. ArXiv DOI 10.48550/arXiv.2409.12177.

Zhang Z, Liu Y, Zhong S-H, Chen G, Yang Y, Cao J. 2024c. From references to insights: collaborative knowledge minigraph agents for automating scholarly literature review. ArXiv DOI 10.48550/arXiv.2411.06159.

Zhang Y, Wang M, Li Q, Tiwari P, Qin J. 2025. Pushing the limit of LLM capacity for text classification. In: *Companion Proceedings of the ACM on Web Conference 2025*, 1524–1528.

Zhang P, Zeng G, Wang T, Lu W. 2024b. TinyLlama: an open-source small language model. ArXiv DOI 10.48550/arXiv.2401.02385.

Zhang V, Verma S, Vaidya N, Sawarkar A, Saunders A. 2023. NVIDIA AI Foundation Models: Build Custom Enterprise Chatbots and Co-Pilots with Production-Ready LLMs. *Available at https://developer.nvidia.com/blog/nvidia-ai-foundation-models-build-custom-enterprise-chatbots-and-co-pilots-with-production-ready-llms/*.

Zhao H, Andriushchenko M, Croce F, Flammarion N. 2024. Long is more for alignment: a simple but tough-to-beat baseline for instruction fine-tuning. ArXiv DOI 10.48550/arXiv.2402.04833.

Zhao H, Luo Z, Feng C, Zheng A, Liu X. 2019. A context-based framework for modeling the role and function of on-line resource citations in scientific literature. In: *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 5206–5215.

Zhao X, Yin F, Durrett G. 2024. Understanding synthetic context extension via retrieval heads. ArXiv DOI 10.48550/arXiv.2410.22316.

Zhezherau A, Yanockin A. 2024. Hybrid training approaches for LLMs: leveraging real and synthetic data to enhance model performance in domain-specific applications. ArXiv DOI 10.48550/arXiv.2410.09168.