multiple sentences solely through recognized named entities continues to be a challenge[16]. This limitation is particularly pronounced in technical and scientific documents, where critical information is often expressed in a non-standard and complex manner. In the domain of polymer science, NER-based extraction methods encounter additional specific challenges stemming from the expansive chemical design space of the materials and the utilization of non-standard nomenclature, including commonly used names, acronyms, synonyms, and historical terms[17].

Recently, large language models (LLMs) such as Generative Pretrained Transformer (GPT), Large Language Model Meta AI (LlaMa), Pathways Language Model, etc., have gained significant attention in the field of natural language processing[18,19]. These models have shown remarkable performance in handling various NLP tasks, showcasing their robustness and versatility[20], especially in high-performance text classification, NER, and extractive question answering with limited datasets[21]. A key factor contributing to the success of the LLMs is the vast amount of 'knowledge' these models gain during semi-supervised pre-training (e.g., using masked language modeling to predict the next token given a set of preceding tokens for context)[22]. In the pre-training phase, LLMs acquire a foundational comprehension of language semantics and contextual understanding through exposure to training datasets, which typically comprise texts from general science and scientific literature[23]. Subsequently, the pre-trained LLMs, also referred to as base models, undergo supervised fine-tuning to produce desired text outputs in response to specific prompts or instructions. Examples include OpenAI Codex and Code LlaMa, both of which are fine-tuned to generate code snippets based on a given natural language input[24]. Similarly, ChatGPT and LlaMa Chat models are language models fine-tuned to respond to user prompts or instructions conversationally while maintaining a history of previous interactions for added context for the conversation. A human-like understanding of the language semantics and subsequent instruction tuning thus enable the LLMs to perform in-domain tasks such as information extraction about a specific material class with no (zero-shot) to only a few task-specific examples (few-shots). Such ability offers excellent performance and eliminates the efforts needed to create a labeled dataset of significant volume and train or fine-tune a new model[25].

Despite the potential for many use cases including data extraction, the improved capabilities of the LLMs depend on access to significant computational resources. Using LLMs for inference incurs significant monetary costs, due to high demands of energy consumption, hardware or cloud computing time, and in terms of the environment, due to the carbon footprint of powering a number of modern tensor processing units[26,27]. Therefore, a data extraction pipeline aiming to efficiently utilize LLMs should extract the maximum amount of high-quality information and at the same time reduce the unnecessary prompting of the LLMs during the processing of millions of full-text scientific articles.

Limited prior works exist on the application of LLMs for data extraction in materials science. Dagdelen et al. fine-tuned GPT-3.5 and LlaMa 2 models to extract useful records of linking dopants and host metal-organic frameworks[28]. Zheng et al. developed a workflow utilizing ChatGPT as a collaborator for human chemists, extracting 26,257 distinct synthesis parameters of approximately 800 metal-organic frameworks from 228 articles[29]. Polak and Morgan proposed a similar workflow for metallic glasses and high entropy alloys, employing follow-up questions to GPT-4 to ensure correctness and address the issues of hallucinations with LLMs[30]. Similarly, Yang et al. used a repeated questioning strategy with GPT-4 for bandgap values, demonstrating reduced error rates and a more extensive dataset than human-curated databases[31]. GPT-based approach offered high-performance text classification, NER, and extractive question answering with limited datasets, and could reduce researcher workload by producing initial labelling sets and verifying human-annotations.

In this contribution, we present an approach to employing LLM- and NER-based pipelines, specifically designed to automate the extraction of property data of polymers from the full-text contents of journal articles. Our data extraction workflow, depicted in Fig. 1, processes a corpus of 2.4 million materials science journal articles published in the last two decades, from which, we identify and concentrate on 681,000 polymer-related articles. Subsequently, the paragraphs of the articles are processed through a dual-stage filtering scheme consisting of a 'heuristic filter' and a 'NER filter' to identify the most relevant paragraphs that contain extractable property data. The materials and properties are identified, relationships are established, and the information is extracted in a structured format using MaterialsBERT and GPT-3.5 models independently. Our pipelines extracted more than one million values of 24 selected properties from the full texts of the polymer-related articles. We have made the extracted data publicly available at polymerscholar.org (henceforth referred to as Polymer Scholar) where researchers can explore the distribution and relationships within the properties of polymers[15]. To identify the most efficient model, with a special focus on optimizing quality and costs, we evaluate three models – Materi-

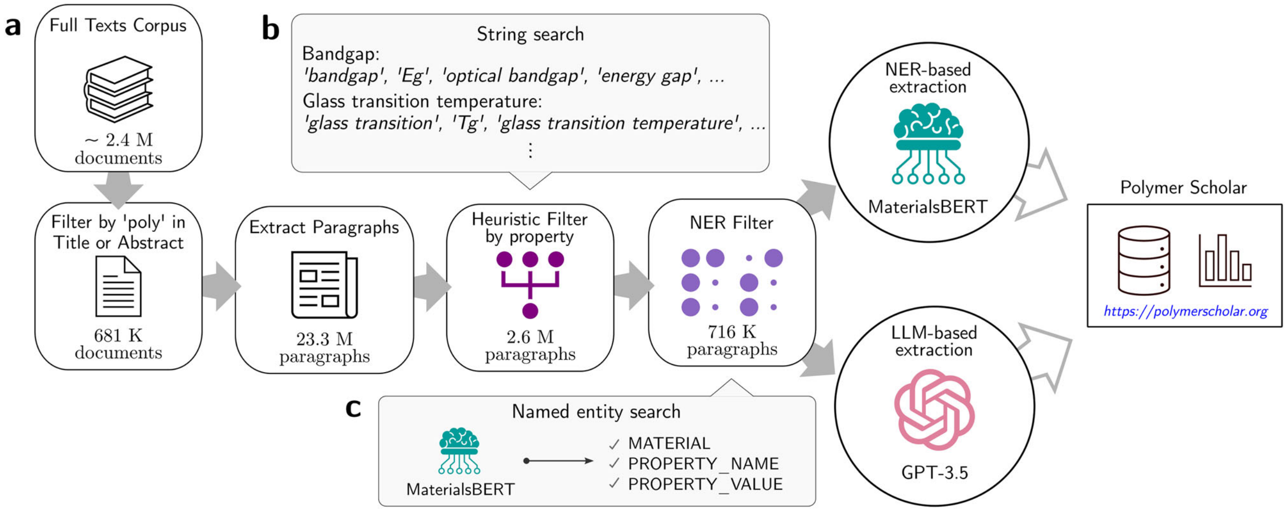

**Fig. 1 | Overall workflow to extract polymer property data.** **a** Polymer-specific documents are selected from a corpus of 2.4 million materials science journal articles. Multiple stages of filtering select the most relevant documents and paragraphs of the documents before performing data extraction by MaterialsBERT and GPT-3.5. Extracted data are finally deposited to a relational database of the Polymer Scholar web interface. **b** Property-specific paragraphs are selected by a heuristic filter based on string matching and dictionary lookup. **c** The NER filter identifies paragraphs with extractable named entities. The LlaMa-2 large language model was also evaluated, but was not used in the final data extraction pipeline due to comparatively low performance and long inference time, as described later in the text.