WIRTSCHAFTS
UNIVERSITÄT
WIEN VIENNA
UNIVERSITY OF
ECONOMICS
AND BUSINESS

MSc Thesis

RAG for Knowledge Extraction: Theoretical Framework and Applications in Research

Daniil Dobriy

Student ID: 11776408

Programme Code: 066/960

Advisor: Prof. Dr. Axel Polleres

Year of Submission: 2025

Department of Information Systems & Operations Management, Vienna University of Economics and Business, Welthandelsplatz 1, 1020 Vienna, Austria

DEPARTMENT FÜR WIRTSCHAFTSINFORMATIK
UND OPERATIONS MANAGEMENT DEPARTMENT
OF INFORMATION SYSTEMS AND OPERATIONS
MANAGEMENT

Figure 1: *Falke* by Philipp Ferdinand de Hamilton, end of 17^{th}/first half of 18^{th} century. Kunsthistorisches Museum, Vienna, Austria

Preface

This work builds upon two papers ([33] and [82]), as well as insights gained from co-organizing an academic workshop on Retrieval-Augmented Generation with Knowledge Graphs.[1] The research also incorporates findings from earlier investigations into applying LLMs for knowledge extraction [4, 35, 34].

[1] https://2024.rage-kg.org

Abstract

Structured knowledge extraction from natural language text is essential for systematic evidence synthesis and knowledge base construction. Traditionally, this process has been performed through manual feature extraction by human coders. However, this approach suffers from significant resource constraints, including time and labour costs, as well as reliability concerns due to inter-coder variability. Large Language Models (LLMs) demonstrate emergent text understanding capabilities, presenting a promising alternative to human coding. This work establishes foundations for the systematic application of LLMs in structured knowledge extraction from natural language text and introduces methodologies for reliability assessment through golden standard studies and human-LLM agreement evaluation. These methodologies are applied to knowledge extraction use cases in the research domain, demonstrating that LLMs can reliably replace and often exceed human performance in complex extraction tasks. This advancement enables scalable, consistent, and cost-effective knowledge extraction for research synthesis and knowledge base construction.

Contents

List of Figures

List of Tables

List of Abbreviations

CA Content Analysis
EEL Entity Extraction and Linking
FAIR Findable, Accessible, Interoperable, and Reusable
GenAI Generative AI
GT Grounded Theory
HITL Human-in-the-Loop
ICR Inter-coder reliability
ICT Information and Communication Technology
IE Information Extraction
IR Information Retrieval
IRI Internationalized Resource Identifier
KC Knowledge Codification
KE Knowledge Extraction
KG Knowledge Graph
KM Knowledge Management
KMS Knowledge Management Systems
KR Knowledge Representation
LD Linked Data
LLM Large Language Model
LOD Linked Open Data
ML Machine Learning
NED Named-Entity Disambiguation
NER Named-Entity Recognition
NLP Natural Language Processing
NLT Natural Language Text
NLU Natural Language Understanding
OBIE Ontology-based Information Extraction
OWL Web Ontology Language
QC Qualitative Coding
RAG Retrieval-Augmented Generation
RDF Resource Description Framework
RE Relation Extraction
RL Relation Linking
SHACL Shapes Constraint Language
SLR Structured Literature Review
SPARQL SPARQL Protocol and RDF Query Language
SW Semantic Web
WU Vienna Vienna University of Economics and Business

1 Introduction

This work explores the use of LLMs for automated extraction of structured knowledge from NLT. Its subject, therefore, lies at the intersection of KC [26], QC [115], IE [27] and KE [108], which can be regarded as implementations of a common task in different disciplines. Section 1 is structured as follows: First, the work reviews and compares the methodologies of KC, QC, IE and KE, highlighting their commonalities and distinctions. The analysis then examines the current state of QC tools and their limitations, particularly focusing on resource requirements and reliability challenges. Subsequently, the discussion explores recent advances in automated KE and IE, especially the emergence of LLMs and their potential applications. Finally, the section presents a research agenda for leveraging LLMs and RAG to address the identified challenges in QC and KC, outlining an approach to developing and validating a novel methodology that bridges established practices for manual knowledge extraction with emerging technologies.

1.1 Overview of Related Extraction Methodologies

KC is a prominent method of KM, which can be defined as the reduction and conversion of knowledge into information that enables efficient transmission, storage and reproduction, with the lowest level of codification resulting in unambiguous codes [30]. Minimally, the process entails the creation of (i) models, (ii) languages that codify the proposed models, and ultimately results in (iii) messages [29]. Through KC, the (a) knowledge acquires commodity properties enabling direct trade or value signalling, (b) it obtains non-standard commodity features like non-rivalry in use and low marginal reproduction costs, (c) enables modularization of knowledge leading to improved organizational efficiency, and (d) accelerates knowledge creation, innovation and economic growth [24]. Advances in ICT continuously contribute to diminishing costs of knowledge codification and increase the potential value of codified knowledge [24].

Another prominent method in social sciences and other disciplines, QC, is an essential methodological tool of qualitative research in which codes systematically assign summative, salient and essence-capturing attributes to portions of NLT or visual data [43]. QC primarily deals with research data sources, such as interviews and other text excerpts. It has has also benefited from the use of dedicated software and automated analysis which will be discussed below.

The advances in data processing, however, are most prominent in the tasks of KE and IE. IE is the process of automatically extracting structured

information from unstructured or semi-structured sources, typically involving the processing of NLT through the means of NLP [89], KE targets the transformation of both structured and unstructured sources into machine-readable and machine-interpretable data, which with regards to natural language text commonly relies on NLU, a subset of NLP targeting the extraction of semantics from text.

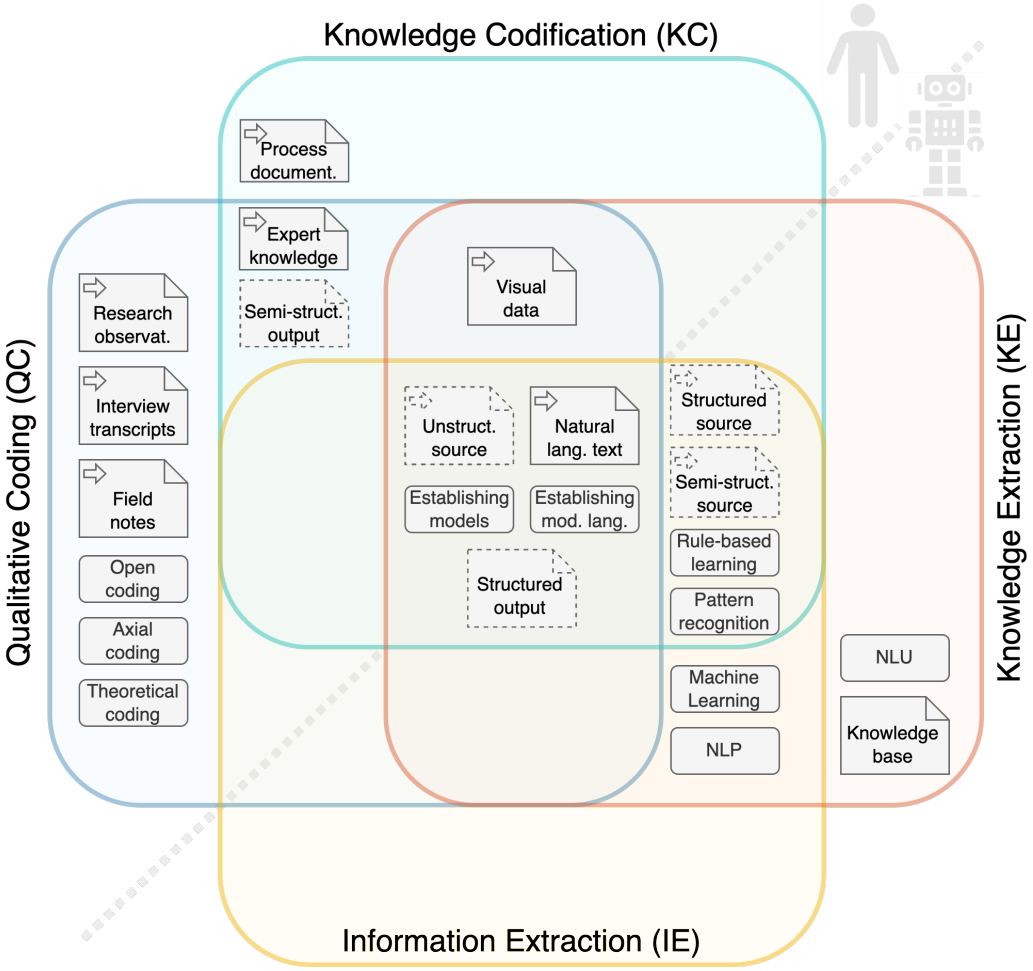


Figure 2: Comparing Knowledge Codification, Qualitative Coding, Knowledge and Information Extraction methodologies in terms of inputs, methods and artifacts

Figure 2 compares the methodologies in terms of inputs (), methods () and artifacts (). Classes shown with dashed borders represent more abstract, higher-level categories (e.g., "Unstructured source"), while solid borders indicate more specific, concrete classes (e.g., "Interview transcripts").

The analysis establishes notable commonalities between the approaches: all methodologies, similar to (i) - (iii) steps of KC, implement basic modelling steps and strive towards the creation of structured or semi-structured data artifacts as the result of analysis. Furthermore, all methodologies typically work with unstructured data, most commonly NLT, as input to analysis.

1.2 Tools and Costs of Manual Qualitative Coding

There is a number of software tools supporting the process of QC[2]. Review studies have analysed the use of such tools in QC exercises in the healthcare and medical education domains [104, 23]. Table 1 gives on overview of the widely-used tools supporting QC, their use in studies and main features.

		Capabilities [23]							
Tool	**Use, %** [104]	Code	Aggregate	Search	Visualize	Transcribe	Collaborate	Multilang.	Stat. anal.
ATLAS.ti	29.2	✓	✓	✓	✓	✓	✓	✓	
NVivo	20.8	✓	✓	✓	✓	✓	✓		
Ethnograph	4.2	–	–	–	–	–	–	–	–
MAXQDA	0.0	✓	✓	✓	✓	✓	✓	✓	✓
DeDoose	0.0	✓	✓	✓	✓		✓		
QDA Miner Lite	NS	✓	✓	✓					
Quirkos	NS	✓	✓	✓	✓		✓		

Note. Usage percentages based on analyzed studies (n=48). NS = Not specified in usage analysis. Checkmark (✓) indicates feature availability; dash (–) indicates no data available.

Table 1: Overview of Software Tools for Qualitative Coding

Most tools provide the core functionality assisting the manual QC process: document search, aggregation and coding. Similarly, most tools offer visualization of data (codes) and collaborative coding. Some tools also include transcription functionality and support multilinguality. Only MAXQDA also offers features implementing statistical analysis (which, however, have not been used in the studies reviewed). As this analysis shows, the tools available primarily support the users at manual QC, streamlining the process and offering collaboration functionality. They all rely on relegating the concep-

[2]The following analysis applies to KC in a similar fashion despite fewer tools available.

tualization aspects of QC to researchers and human coders, including the process of manually going over the data, assigning codes and features.

Consequently, despite advances in supporting software tools that assist the process, QC remains heavily dependent on manual labour. The manual sub-tasks of QC demand substantial resources in terms of time and skilled personnel. The time required for data analysis is a primary issue in QC, as conceptual tasks associated with qualitative analysis cannot be easily hastened [106]. Furthermore, the requirement for multiple coders to independently process the same material – a practice necessary for ensuring reliability – significantly multiplies the resource investment [23]. The SLR study, whose protocol is detailed in Appendix A, estimates high monetary costs associated with such studies based on the data for QC studies performed at the Vienna University of Economics and Business (WU Vienna). Notably, large-scale qualitative studies working with large datasets may employ teams of 10 or more coders, which requires extensive 1) recruiting and training, 2) compensation and incentives, 3) efforts maintaining data quality and coding reliability and 4) building up team coordination and morale [8].

1.3 Reliability of Qualitative Coding

Another challenge associated with qualitative research and QC studies is their reproducibility. Though qualitative methods often do not easily subject themselves to replication, aiming at the objectivity of research and striving towards reproducibility shall be nonetheless attempted [123]. On the other, the use of formalized methodologies, such as QC, shall enable reproduction and, therefore, cement the value of research results achieved by such studies. ICR is considered a good practice in qualitative research [94], but its use can still be associated with methodological and practical limitations, such as bias inherent in small or large but homogeneous coder teams. Achieving a high ICR score becomes ever more challenging as the conceptual complexity of codes increases [88]. And besides the practical difficulties of achieving conceptual agreement, coding protocols or essential elements of the process are occasionally simply not reported [94]. Even when reported, authors often have different interpretations of similar ICR metrics, and the percentage of data used in ICR tests varies across studies [19]. Thus, the issues associated with reproducibility in QC research contribute to the overall replication crisis [7, 88], especially in disciplines relying on qualitative methodologies, such as psychology [92], social sciences [14] etc.

Thus, QC, and KC in general, face two significant challenges: First, it remains highly resource-intensive, requiring substantial investments in time and skilled personnel. Second, the reproducibility of such studies is ham-

pered by methodological limitations, inconsistent reporting of coding protocols, varying interpretations of ICR metrics, and challenges in achieving conceptual agreement among coders, contributing to the broader replication crisis in science.

1.4 Automatic Information and Knowledge Extraction

While KE, IE, QC, and KC all process unstructured sources (particularly, NLT) to generate structured data, KE and IE differ fundamentally in their approach by primarily utilizing machine processing rather than human analysis. The reliability of proposed KE and IE pipelines is assessed in comparison to benchmarks (or, golden standard/ground truth data), which are created/-validated manually by humans. With the increasing permeation of LLMs, frameworks for evaluating their performance in IE tasks have been proposed [37].

In the recent time, pre-trained LLMs have demonstrated impressive emergent capabilities, especially in understanding context across different tasks and scenarios. In NLP, LLMs have achieved state-of-the-art performance across many tasks, becoming the de-facto baseline models [156]. LLMs also perform well in the variations of the Turing Imitation Game[3] [58, 57, 144]. And while some studies still report LLMs falling short of the baseline set by human participants (about 66%), mostly failing based on linguistic style [58], the fact that they have achieved human-level capabilities[4] in text generation in practice is evident in how LLM output is widely (and successfully) used to cheat on educational assignments [25].

LLMs are increasingly used for IE and KE tasks in various domains [47, 146, 152, 99]. They have also been proposed for document information extraction from visually-rich documents [96] and scientific texts [28]. The extraction process commonly employs carefully crafted prompts to specify both the desired information and its structural representation [132, 84].

1.5 Semantic Web and Retrieval-Augmented Generation

SW [9] provides a universal framework to KR, which allows publishing interconnected data on the web and enables machines to comprehend data meaningfully through the use (most notably, re-use) of ontologies. Thus, SW

[3]Widely known as the original "Turing Test."

[4]One could argue, perhaps inviting criticism from certain categorically-disposed readers, that these capabilities may be construed as encapsulating *intelligence* as such.

captures and links information on the Web, enriches it with semantics and enables reasoning on it [112]. SW standards also underlie KGs, facilitating data integration, powering search engines and capturing enterprise data [51]. Also notable is the synergetic application of SW in ML [12], where SW can provide actual semantics, grounding the ML methods and their outputs in meaning.

Since 2023, Gartner has placed KGs in the centre of crucial technologies enabling GenAI on its GenAI Impact Radar,[5] and placed KGs on the same level as GenAI on its general Impact Radar.[6] The trend has also been reflected in academic publications. On the one side, LLMs have spurned abundant research in the direction of KG construction [91, 130, 150, 15, 18, 154, 153, 81] including for service descriptions [151] and recommendations [139]. On the other side, research has focused on the major shortcomings of LLMs - namely, the hallucination of facts [124, 2, 120, 76] and solving such limitations with RAG approaches, by infusing LLMs with KGs and structured data in various ways: though knowledge injection in prompts [70, 53, 54, 77, 6, 141, 1] and outputs [45], during training and fine-tuning [148, 125, 55, 83], for reasoning over graphs [61, 135, 72, 119], query generation and question answering [67, 122, 149, 145] and other approaches [2, 69]. Recently, RAG has also been extended by agentic architectures. Agentic RAG incorporates planning, tool use, multi-agent collaboration in retrieval and contextual understanding, which improves the flexibility of the approach [114].

1.6 Research Agenda and Work Structure

Given the demonstrated capabilities of LLMs in text understanding and generation, coupled with their increasing application in IE and KE tasks across various domains, these models present a promising solution to the fundamental challenges facing QC and KC. In this regard, RAG presents a promising approach to eliciting the desirable output. Furthermore, SW standards and domain-specific ontologies could be used to capture and validate the output of analysis.

In this work, we will address three key challenges in leveraging LLMs and RAG for QC and KC. First, we will review the methodologies for QC, KC, IE, and KE and analyse the commonalities between them to establish a unified approach that utilizes LLMs' capabilities in text understanding

[5] See https://www.gartner.com/en/articles/understand-and-exploit-gen-ai-with-gartner-s-new-impact-radar

[6] See https://www.gartner.com/en/articles/30-emerging-technologies-that-will-guide-your-business-decisions

and structured information extraction. Second, we will adapt traditional reliability testing methods to the context of KE using an ontology-guided RAG pipeline, developing new frameworks for assessing extraction quality and consistency when automated systems replace or augment human coders. Third, we will investigate the inherent limitations of RAG-based KE, particularly focusing on potential biases and the implications for research validity. Through this systematic examination, we aim to develop a comprehensive methodology that bridges established research practices with emerging LLM capabilities while maintaining and improving standards for research quality and reliability. Thus, the research questions addressed in this work, are:

RQ1: How can Retrieval-Augmented Generation be used to automatically extract structured data that conforms to predefined ontologies from natural language text?

RQ2: How can reliability testing methodologies be adapted to evaluate ontology-guided knowledge extraction using Large Language Models?

RQ3: To what extent is ontology-guided knowledge extraction with Large Language Models reliable for practical applications in the research ecosystem?

The remainder of this work is structured as follows: Section 2 provides essential background on relevant extraction methodologies, reliability testing methods, and structured data representation. Section 3 reviews related work in KE using LLMs, with particular focus on recent advances in prompt engineering and RAG approaches. Section 4 presents the methodology for developing and evaluating LLM-based extraction pipelines. Section 5 establishes the theoretical framework for reliability testing of such pipelines. Section 6 demonstrates practical applications of our methodology in research ecosystems. Finally, Section 7 and Section 8 discuss our findings, highlight limitations, and outline directions for future research. At the very end, Appendix contains additional materials such as the protocol of the structured literature review quantifying resource-intensity of QC tasks.

2 Background

This section provides essential background information for understanding the methodological and technical foundations of this work. First, it examines QC and KC as research methodologies, including their established practices and core processes. Then, it explains reliability testing approaches widely used in

QC, IE and KE, focusing on golden standard studies, ground truth comparisons and benchmarking. It also details methodologies for ICR assessments. Subsequently, the section explores structured data representation, examining how SW and ontologies enable knowledge representation and reuse. Finally, it introduces RAG and its typical implementation patterns, establishing the technical framework for the methodology developed in this work. Understanding these components is crucial for developing an automated approach that maintains methodological rigour while leveraging emergent capabilities of LLMs.

2.1 Manual Extraction

The four extraction methodologies handled in this work - QC, KC, IE, and KE - have evolved distinct practices, principles, and validation approaches within their respective domains. While QC and KC emerged from social science and organizational research traditions, IE and KE developed from computer science and artificial intelligence fields. The following subsections detail the core components, established workflows, and mention quality assurance practices of each methodology as traditionally implemented in research practice.

2.1.1 Qualitative Coding

QC is "the simple operation of identifying segments of meaning in your data and labelling them with a code," [115] where a code is "a word or short phrase that symbolically assigns a summative, salient, essence-capturing, and/or evocative attribute for a portion of language-based or visual data." [110] . Initially, QC was established as the backbone of GT, a qualitative research methodology that formalizes theory generation from data [43, 118].

Characteristics QC is often understood as an iterative process. With each iteration, codes are refined in various ways: redefined, modified, combined, split, reassigned etc. Through multiple iterations, a better understanding of the underlying data is sought after, which includes uncovering potential connections and insights gained from it. This process can be deductive (predefined codes are applied to text, in a top-down approach) just as it can be inductive (codes are "learned" from text, in a bottom-up approach), and often QC application employs both approaches interchangeably.

Initial Coding Cycle In the first iteration, codes could be either assigned deductively, using a pre-defined codebook (Provisional, Hypothesis, Protocol

Coding etc.) based on a specific theoretical framework, previous research or otherwise elicited through a structured approach, yet still being attentive to potential other codes not considered initially, or inductively, through one of the approaches [110] described in Table 2 or other, domain and study-specific approaches.

Method Name	**Description**	**Examples** (code examples)
Descriptive Coding	Summarizes the primary topic of a data segment with a brief label.	Workplace dynamics Team communication
In Vivo Coding	Uses participants' own phrases as codes.	It's overwhelming Supportive team
Attribute Coding	Identifies and categorizes attributes of participants or data.	Age, gender, occupation
Simultaneous Coding	Applies multiple codes to a single data segment.	Labour-intensive work Financial cost
Subcoding	Adds detailed subcategories to primary codes for better organization.	Challenge: Time Challenge: Cost
Magnitude Coding	Assigns levels of intensity, frequency, or agreement.	Various scales possible
Domain and Taxonomic Coding*	Categorizes taxonomies within the data.	Coding method → First Cycle Method
Causation Coding	Explores cause-effect relationships in the data.	Lack of resources → Limited reliability
Process Coding	Captures ongoing actions/processes.	Teaching Learning Analyzing
Emotion Coding	Labels affective states or emotions.	Joy Rage Nostalgia
Values Coding	Captures participants' values and beliefs.	Importance of efficiency Always tell the truth

Note. Though `Domain and Taxonomic Coding` is primarily used in ethnographic research to discover the taxonomy of behaviours and interpretations of experiences, this work generalizes it to taxonomic relationships in data as such.

Table 2: Summary of Selected First Cycle Coding Methods

Further Coding Cycles Following the initial cycle of qualitative coding, subsequent cycles focus on refining, integrating, and abstracting the insights derived from the data. Prominent methods for second-cycle coding include Pattern Coding, Axial Coding, Focused Coding, Elaborative Coding and Longitudinal Coding [110]. These methods (with the exception of Longitudinal Coding, which relies on changes across time in panel data) are summarized in Table 3.

Theory-building After several iterations of coding and with the help of memos (i.e. notes made during the process and relating to specific codes

Table 3: Summary of Selected Second Cycle Coding Methods

Method Name	Description	Transformation Example
Pattern Coding	Groups similar codes into meaningful clusters (themes, explanaitions).	Workplace dynamics + Team communication → ORGANIZATIONAL DYNAMICS
Focused Coding	Identifies the most frequent or significant codes to develop major categories. Other sources include deductive elements in this approach.	It's overwhelming + Supportive team → WORKPLACE SUPPORT SYSTEM
Axial Coding	Reassembles data that was split in first cycle coding by relating categories to subcategories and specifying their properties and dimensions.	Teaching + Learning + Analyzing → KNOWLEDGE PROCESSES (incl. dimensions)
Theoretical Coding	Integrates and synthesizes all other codes and categories into a coherent theory by core category that explains the phenomenon.	Joy + Rage + Nostalgia → EMOTIONAL TRAJECTORY IN QC

or categories), the established categories are reflected upon with the goal of integrating them into an overarching theory, though no formalized approach exists that can be followed here [110].

2.1.2 Content Analysis

While QC can support theory-building, in practice many studies rely on coding primarily as input for qualitative and quantitative CA [65, 138, 116, 52]. Here, the codes serve as systematic categorizations [79] that enable both qualitative interpretation and quantitative analysis of patterns and frequencies in the data. This approach allows researchers to examine the presence, meanings and relationships of concepts in texts, relying on the systematic coding process.

As a pendant to inductive vs. deductive approaches in QC, CA differentiates between conventional, summative and directed approaches [52]. Figure 3 illustrates the classification [52] based on the initial source of keywords/codes.

The selection of a content analysis approach aligns with specific research objectives. Conventional Analysis is most appropriate when investigating novel phenomena where existing frameworks may constrain discovery of emergent patterns. QC with a Directed Analysis serves to extend established theoretical constructs into new domains or validate existing frameworks. The Summative Analysis, in contrast, is applied when research questions centre on examining the general or contextual usage and implicit meanings of specific terms across a corpus of text, thus constituting a primarily descriptive approach.

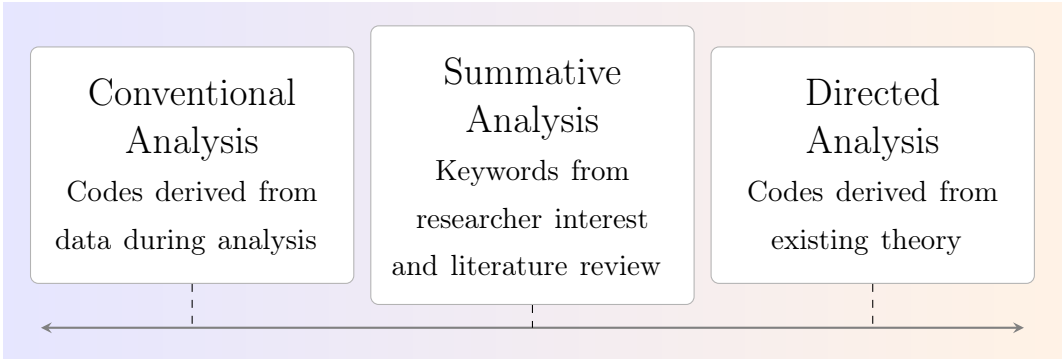


Figure 3: Content Analysis Approaches by Source of Codes

Qualitative Content Analysis The approaches of CA mirror the approaches to coding iterations. Table 4 establishes parallels between qualitative CA approaches [79] and QC methods from different stages of the process. Similarly to QC, mixed-method qualitaive CA combines deductive and inductive approaches iteratively or in parallel. The central analytical method, Deductive Category Assignment follows a strict deductive approach, basing the assignment of categories on definitions from a (theoretically-informed) codebook. Summarization approach aims to iteratively reduce the variability of descriptive codes through generalization and reduction, implementing various Open Coding methods. Inductive Category Formation is a streamlined version of the Summarization approach which establishes certain preselection rules and the reduction level to simplify the approach and insure its termination with one or few iterations. Finally, Explicational Content Analysis can be viewed as informed transcription, standardizing the initial data.

Quantitative Content Analysis Quantitative CA aims at applying statistical methods to test content-related hypotheses utilizing the coded data. By systematically analysing the coded representations, it facilitates the identification of patterns, relationships, and trends within textual data. Quantitative CA extends the scope of qualitative methodologies by incorporating analytical testing, often leveraging statistical tools for hypothesis validation. This segment outlines key techniques, emphasizing their analytical focus, required bases of analysis, and interpretive capabilities. Table 5 summarizes these approaches [65], highlighting the applicable types of input codes as the basis of analysis and interpretation of results.

QCA Approach	QC Pendant (if applicable)	Description
Deductive Category Assignment	Provisional Coding (and deductive coding methods)	The central content-analytical method with the goal of extracting certain pre-defined structures from data (based on coding rules) [79]. This approach is also known as Directed CA [52]
Summarizing	Descriptive Coding Pattern Coding Focused Coding	Data segments are coded into increasingly abstract representations, aiming at achieving uniform levels of greater abstraction at each step through paraphrasing, generalization and reduction [79].
Inductive Category Formation	Fcoused Coding	Pre-screened (based on a selection rule) segments are summarized to categories using pre-defined reduction level [79].
Explicational Content Analysis	No exact pendant	Unclear text segments are explicated, i.e. explained suing internal (text) and external (metadata) references [79]. Such explications are then inserted back into the text to check coherence .

Table 4: Comparing Qualitative Content Analysis to Coding Methods

Conclusion Quantitative content analysis offers a systematic approach to examining symbolic and numerical code data through quantitative techniques [65]. These methods are rooted in a robust methodological framework that emphasizes reproducibility, ensuring stable inferences from coded data. By enabling structured analysis of text sources, quantitative CA approaches complement qualitative methods and expand the analytical capabilities of researchers, particularly when it comes to robustly validating (theoretically informed) hypotheses based on available data, leading to the wide adoption of such methods in modern research ecosystem.

2.1.3 Knowledge Codification

Knowledge codification refers to the process of converting tacit, unstructured, or implicit knowledge into explicit, structured, and transferable forms. It involves reducing knowledge into symbolic representations, such as documents, algorithms, or codes, making it amenable to dissemination, replication, and control [26]. Codification underpins the development of dynamic capabilities by documenting experiential learning and formalizing knowledge for future use [155]. Articulation and codification facilitate innovation, division of labor, and replication, forming the foundation for systematic knowledge management [46].

The process is driven by the benefits of articulation: accelerating innovation, enabling precise communication, and ensuring organizational memory. However, the extent of codification depends on its cost-effectiveness and the suitability of available codes, tools, and theoretical underpinnings [26, 155, 46].

Technique	Description	Basis of anal.	Interpretability
Tabulation	Summarizes data by counting occurrences of codes.	Categorical codes	• Descriptive distribution of categories
Cross-Tabulation	Identifies relationships between coded variables through co-occurrence frequencies.	Categorical or Ordinal codes for multiple variables	• Hypothesis testing of associations
Correlation, Multivariate Techniques	Measures the strength and direction of (complex) relationships between independent /dependent variables.	Ordinal or Continuous numerical codes	• (Multifactorial) hypothesis testing of correlations • Causality through additional considerations
Factor Analysis, MDS	Reduces dimensionality by grouping correlated variables into abstract constructs.	Continuous or Interval-level numerical data.	• Identify factors explaining data variance with min. loss of explanatory power.
Semantic Network Analysis	Examines relationships between coded concepts based on their co-occurrence in text or networks, graph-based metrics.	Categorical codes, Associations	• Identifies central concepts within networks • Semantic or structural relationships between concepts
Clustering	Groups codes into clusters based on similarity.	Categorical or numerical codes.	• Identifies natural, thematic groupings of codes

Table 5: Overview of Selected Quantitative Content Analysis Techniques

Codification Steps The methodologies of codification involve several iterative steps, often cyclically refining tacit insights into actionable frameworks:

1. **Identifying Knowledge:** Recognize and define the tacit knowledge to be codified. This step involves analyzing processes, tasks, and capabilities where knowledge resides [155].

2. **Articulation:** Express tacit knowledge in an explicit form via notes, diagrams or structured documentation. Tools like templates, blueprints, and documentation are helpful at this step [46].

3. **Creating Codes:** Develop or apply standardized symbolic representations to structure knowledge into transferrable formats, such as manuals, algorithms, or digital repositories [26].

4. **Validating Knowledge:** Ensure the accuracy and usability of codified knowledge by involving domain experts.

5. **Storage and Dissemination:** The codified knowledge shall be stored in accessible repositories and disseminated within or across organizations using KMSs [46].

6. **Continuous Updating:** The codified knowledge should be periodically updated to reflect evolving knowledge [56].

2.2 Automatic Extraction

Automatic extraction methods represent a significant advancement over manual approaches in terms of reducing resource requirements such as time and skilled personnel at the actual extraction stage, but potentially not at (pre-)training as they could require extensive data management and supervision. While all the described methodologies (QC, KC, IE, KE) share the common goal of transforming unstructured data (in particular, NLT) into structured formats, IE and KE fundamentally differ by relying on machine processing rather than human evaluation.

It must be noted that such reliance could introduce biases, which human coders, in theory, should aim to avoid or, at the very least, be conscious about during the extraction process as well as when evaluating results. Furthermore, QC and KC imply domain expertise on the part of human coders as well as an implicit ability to solve natural language ambiguities. Finally, human coders performing their tasks conscientiously are presumed to be able to justify their annotations. This section introduces IE and KE as extraction methodologies.

2.2.1 Information Extraction

Initially, IE has been defined in contrast to IR as the process that goes beyond merely retrieving relevant documents, but instead identifying and isolating only the text fragments that contain information of interest, extracting the specific structured facts from these fragments and placing these facts into a consistent framework ("a template") [27]. "With large amounts of potentially useful information in hand, an IE system can then transform the raw material, refining and reducing it to a germ of the original text" by "find[ing] and link[ing] relevant information while ignoring extraneous and irrelevant information" [27]. Here, one could already draw parallels to deductive approaches of QC and Directed Analysis in CA. IE research can be characterized along multiple dimensions [111], as illustrated in Table 6.

Dim.	Aspect	Description	Examples
Type of Info Extracted	Entities	Named entities or domain-specific objects	People, locations, proteins, chemicals
	Relations	Binary or n-ary relations or events	`[Person]-employedBy-[Org]`
	Attributes	Qualities describing entities	Polarity, numeric quantity
	Higher-order	More complex info (lists, tables, ontologies)	Lists, taxonomies, hierarchies
Input	Granularity	IE on short snippets vs. entire documents/corpora	Single-sentence vs. multi-document
	Document Type	Templatized text vs. heterogeneous free-form text	Classified ads vs. Web pages
Input Resources	Structured DBs	Pre-existing KBs or gazetteers for validating/alignment	Geographic DBs, enterprise KBs
	Labeled Data	Manually annotated or distant/self-supervised data	Training corpora for CRFs, neural models
	Preprocessing	Shallow (POS tagging) or deep (dependency parsing)	POS tagger, parser, chunker
Extraction Methods	Rule-Based	Expert-crafted pattern rules	Regex, dependency templates
	Learning-Based	Supervised or semi-supervised	CRFs, neural networks, bootstrapping
	Hybrid	Combination of rule-based & statistical scoring	Large-scale or domain-rich IE
Output Structure	Annotated Text	Marking entity boundaries & linking them to relations	Named-entity spans plus relation labels
	Relational Tuples	Storing facts as (Subject, Relation, Object)	`([Person], worksFor, [Company])`
	Contextual/ Nested	Handling attribution, negation, nested args	Factuality, modality, conditionals

Table 6: Dimensions of Information Extraction

One implementation of IE is the OBIE systems, which align the IE pipeline with Semantic Web standards [75]. Figure 4 illustrates the main steps of an IE pipeline and summarizes the ways ontologies inform them [75]. Most crucially, ontologies can be used to: a) pre-filter inputs, b) guide tokenization by providing labels, c) for entity disambiguation and classification, d) for relationship extraction and validation, e) as the output of the IE pipeline and finally, f) for output validation and further inference.

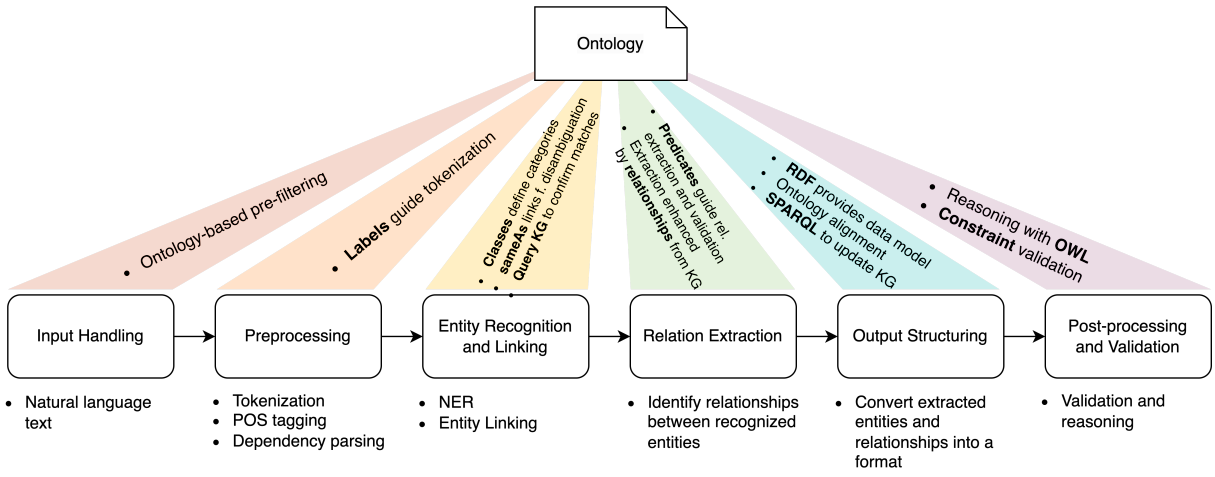


Figure 4: Ontology-enhanced Information Extraction Pipeline

In OBIE systems, ontologies are systematically integrated throughout each step of the IE process: during input handling, ontologies guide the processing of NLT through ontology-based pre-filtering to identify relevant text segments; in preprocessing, they enhance tokenization with ontology labels and can assist POS tagging, and dependency parsing; for entity recognition and linking, ontological classes define the categories for NER, with equivalence links guiding disambiguation and KGs queries confirming entity matches; during relation extraction, ontological predicates facilitate identifying relationships between recognized entities and validate extractions, further enhanced by existing relationships from KGs; in output structuring, they inform how to convert extracted entities and relationships into a format. e.g., using RDF as a data model, supporting ontology alignment or re-using the ontology; and finally, during post-processing, ontologies enable validation and reasoning over the extracted information through OWL (or other implementations of) inferencing and constraint validation, e.g., with SHACL. Thus, ontologies can underpin the entire extraction process to produce consistent structured information. This IE model will inform the design of the generalized KE pipeline described in Section 5.

2.2.2 Knowledge Extraction

While KE possesses many commonalities with IE, the distinguishing principle is that in KE either "existing formal information must be reused (e.g., in KGs or knowledge bases) or a schema must be derived from the source data" [108]. Thus, KE places an even greater emphasis on the tighter integration of ontologies into the pipeline. Similarly to IE, KE relies on NER and NED as well as RE and RL. Here, KGs and ontologies provide the structured knowledge and schemas for the mapping of the extracted knowledge [108].

The method is frequently employed in research ecosystems to extract knowledge from both academic literature [42, 93] and historical sources [44]. KE is particularly valuable for texts containing rich knowledge which remains effectively inaccessible when shrouded by the "shimmer" of NLT when it is approached as an unstructured data source. Through systematic KE, this knowledge can be transformed into formats that enable broader accessibility through standardization and linking, and automatic analysis. For scientific articles, there exist dedicated ground truth datasets [126], although such resources primarily focus on metadata.

2.3 Reliability Testing Methodologies

Reliability testing methodologies aim to validate extraction quality and consistency across different extraction approaches. These methodologies span ground truth and golden standard studies for evaluating automated systems, benchmarking studies for comparing different tools and approaches, and ICR assessments for measuring agreement between (human) annotators. While they share common goals of ensuring reproducibility and quality, each methodology employs distinct validation techniques addressing its specific requirements.

2.3.1 Ground Truth and Golden Standard Studies

One of the methodologies for evaluating reliability and performance is ground truth studies. Also referred to as Gold Standard Evaluations [64], they are used evaluate the reliability of IE and KE pipelines. The ground truth study methodologies differentiate the design of ground truths, collaborative labelling practices and quality assurance [86].

Design of Ground Truth There are three [86] key approaches to designing ground truths which mirror the inductive and deductive approaches in QC and CA: in (a) Principled Design, ground truth is created using predefined

guidelines and labeling vocabularies (this approach is similar to deductive QC and theory-driven CA); in (b) Iterative Design, ground truth is refined over successive cycles (this approach is similar to mixed/multi-cycle QC and summative/mixed-method CA); in (c) Improvisational Design, labels emerge through adaptive, on-the-fly processes. The latter approach bears similarities to inductive QC and data-driven CA.

Labelling and Quality Assurance In terms of labelling practices, there are different approaches to achieving consensus: either multiple annotators could be called upon to coordinate on a common label or there exists a hierarchical structure to resolve ambiguities. There may also be differences to how subsets of data or labelling categories are split between annotators. Here, greater redundancy improves the validity of reliability testing, while at the same time driving costs. Quality in ground truth is achieved through redundant labelling, the implementation of a consensus mechanism and detailed annotation guidelines (which could be iteratively refined).

Limitations Still, there are notable limitations associated with ground truths studies [64]: (1) studies involving human annotators are more resource-intensive and costlier than golden standard studies, (2) human annotation is not perfectly reproducible, (3) ICR above 90% is practically unattainable for specific tasks, (4) evaluations with ground truths are limited to the specific dataset and the results cannot be generalized (e.g., for other applications) and (5) ground truth studies leave room for "cheating," e.g. by misrepresenting the pool of annotators, being "generous" in evaluation etc.

Despite these limitations, ground truth studies are often the only evaluation possible to assess the usefulness of a tool [64] – sometimes, "the gold standard is not the perfect test but merely the best available test" [129].

Metrics Table 7 gives an overview of metrics [102] commonly used for reliability evaluation in golden standard studies. The first two, Precision and Recall are the most commonly applied metrics to test the accuracy of positive predictions (Precision) and the completeness of predictions (Recall). F1-Score represents a balanced combination of two. While Specificity, Accuracy and further metrics are included for completeness, in IE and KE, the negative space is often unbounded when there is not a finite set of items to classify, and the metrics are not directly applicable without additional definition or reformulation of the task as a classification of a with a finite set of categories.

Name	Definition	Purpose	Interpretation
Precision	$\frac{\text{TP}}{\text{TP}+\text{FP}}$	Measures the accuracy of positive predictions	High Precision means most predicted positives are correct
Recall (Sensitivity)	$\frac{\text{TP}}{\text{TP}+\text{FN}}$	Measures the ability to identify all positives	High Recall means few actual positives are missed
F1-Score	$2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision}+\text{Recall}}$	Balances Precision and Recall	x High F1 indicates good balance between Precision and Recall
Specificity	$\frac{\text{TN}}{\text{TN}+\text{FP}}$	Measures the accuracy of negative predictions	High Specificity means most negatives are correctly classified
Accuracy	$\frac{\text{TP}+\text{TN}}{\text{TP}+\text{TN}+\text{FP}+\text{FN}}$	Measures overall classification correctness	High Accuracy means a high proportion of all predictions are correct
Fallout (False Positive Rate)	$\frac{\text{FP}}{\text{FP}+\text{TN}}$	Measures the rate of false alarms	High Fallout indicates many negatives are misclassified as positives
Miss Rate (False Negative Rate)	$\frac{\text{FN}}{\text{FN}+\text{TP}}$	Measures the rate of missed positives	High Miss Rate means many actual positives are misclassified
Jaccard Index	$\frac{\text{TP}}{\text{TP}+\text{FP}+\text{FN}}$	Measures overlap between predicted and actual positives	Higher Jaccard Index indicates better agreement
Matthews Corr. Coeff. (MCC)	$\frac{\text{TP} \cdot \text{TN} - \text{FP} \cdot \text{FN}}{\sqrt{(\text{TP}+\text{FP})(\text{TP}+\text{FN})(\text{TN}+\text{FP})(\text{TN}+\text{FN})}}$	Quantifies the quality of binary classifications	MCC near 1 means strong correlation, near 0 means weak correlation
Informedness	$\text{Recall} + \text{Specificity} - 1$	Indicates the probability of making an informed decision	High Informedness indicates that predictions are far better than chance
Markedness	$\text{Precision} + \frac{\text{TN}}{\text{TN}+\text{FN}} - 1$	Indicates the reliability of predictions	High Markedness means predictions align well with actual outcomes
Area Under Curve (AUC)	$\frac{\text{Recall}+\text{Specificity}}{2}$	Evaluates classifier performance	Higher AUC means better classification performance
Weighted Relative Accuracy (WRAcc)	$4c \cdot \frac{\text{Recall}-\text{Bias}}{1+c}$	Accounts for bias and imbalance in classification	Higher WRAcc indicates better adjusted accuracy
Chi-Square Statistic (χ^2)	$\frac{(\text{TP}-E(\text{TP}))^2}{E(\text{TP})} + \frac{(\text{FP}-E(\text{FP}))^2}{E(\text{FP})}$	Tests statistical significance of results	Higher values indicate significant deviations from the null hypothesis

Table 7: Common Metrics for Ground Truth Studies

ground truth studies are broadly applied to LLM and RAG-based pipelines [100], which imitate tasks normally performed by humans. In the context of SW, a crowdsourcing approach has been proposed for gathering ground truth datasets [36].

2.3.2 Benchmarking Studies

Another approach for performance and reliability testing is benchmarking studies. Benchmarking is a systematic process of comparing the performance of different computational methods using well-characterized reference datasets and evaluation criteria [137]. Unlike the ground truth evaluations that commonly focus on a particular system against a (bespoke) Gold Standard, benchmarking studies are designed to facilitate performance comparisons between multiple approaches. Thus, benchmark datasets can be seen as publicly available and broadly recognised ground truths.

Limitations A practical limitation, however, becomes the possibility of the benchmark being included into the training dataset for a model (i.e. benchmark contamination). The issue is particularly relevant with LLMs [147] trained on vast and commonly intransparently reported sets of openly accessible data. Because of the common lack of transparency with regards to the training dataset and the difficulty of reliably identifying test set contamination, a common approach to avoiding the issue is to evaluate models trained on data predating the publication of a given benchmark. Later in the work, Table 9 specifies the knowledge cut-off dates for major LLMs.

2.3.3 Inter-Coder Reliability

For reliability testing of extraction between single coders, ICR is the commonly used methodology. ICR is "a statistic commonly reported by researchers to demonstrate the rigour of coding procedures during data analysis" [20]. The following metrics are commonly reported in ICR studies:

Percent Agreement Proportion of the number of agreements to the total number of coding decisions [90]. This metric is straightforward and most reported, but does not account for agreement by chance, which is especially relevant when coding schemes have few categories.

$$\text{Percent Agreement} = \left(\frac{\text{Number of Agreements}}{\text{Total Number of Coding Decisions}} \right) \times 100$$

Cohen's Kappa (κ) This metric improves upon Percent Agreement for pairwise agreement between codes by considering the possibility of agreement by chance.

$$\kappa = \frac{P_o - P_e}{1 - P_e}$$

where:

P_o : Observed agreement (proportion of agreements over all annotations)
P_e : Expected agreement by chance based on marginal category frequencies.

However, expected high agreement on common categories may lead to perceived overcompensation and conservative κ values. Furthermore, where the set of possible annotations (e.g., categories) is unbounded, P_e is difficult to estimate. In such cases, a practical but limited solution is to report the metric for the actually observed set of categories, with the limitation that this could overestimate the probability of chance agreement.

Krippendorff's Alpha (α This metric enables reliability assessment across multiple coders and different data types (nominal, ordinal, interval, ratio). Unlike other coefficients, α accommodates missing values, handles multiple coders and considers different measurement levels. Again, extensible category sets where the set of possible values may be theoretically unbounded require additional considerations, similar to κ.

$$\alpha = 1 - \frac{D_o}{D_e}$$

where:

D_o : Observed disagreement as a weighted sum of annotation differences
D_e : Expected disagreement from the distribution of possible pairings.

2.4 Knowledge Representation

The Semantic Web [9] provides a set of standards for human-understandable and machine-processable knowledge representation. At its core, it relies on RDF as a data model, which represents knowledge as `subject-predicate-object` triple statements that use unique identifiers for concepts. An RDF triple is formally defined as:

$$(s, p, o) \in (U \cup B) \times U \times (U \cup B \cup L)$$

where:

- U is the set of all IRI,

- B is the set of all blank nodes (without an identifier),
- L is the set of all literals.

These statements collectively form directed labelled graph structures ultimately known as KGs [51]. Such graph data can be queried with SPARQL. In this context, *ontologies* serve as schemas that define the relationships and constraints among entities, enabling reasoning and data integration (through re-use of the same concepts). In the context of Semantic Web, OWL is commonly used to define ontological semantics and SHACL to define constraint shapes.

2.5 Retrieval-Augmented Generation

LLMs are neural networks, predominantly based on the transformer architecture [128], that have been trained on vast corpora of text to generate human-like responses to prompts. These models process text as sequences of tokens and, as noted in Section 1, have demonstrated remarkable capabilities in understanding and generating NLT, including, initially, translation, but also creative content generation, becoming the backbone technology of GenAI. However, LLMs face inherent limitations: their knowledge is bounded by their training data (with a knowledge cut-off date), they sometimes generate incorrect information (hallucinate), and they lack direct access to external, continuously updated information sources.

RAG is a systematic approach to language modelling that addresses the limitations of LLMs by integrating a corpus or a database for retrieval [68]. This approach combines the capabilities of GenAI with the reliability of retrieval modules, providing a solution to the incompleteness of knowledge inherent in purely parametrized models [41]. This subsection presents important aspects of RAG, following the previously proposed schema [41], illustrated in Table 8. The presented dimensions discuss various implementation aspects of RAG systems, including architectural paradigms for RAG, various augmentation stages, types of data sources, retrieval processes and evaluation methods.

Dim.	Aspect	Description	Examples
RAG Paradigms	Naive RAG	Traditional indexing, retrieval, and generation process	Basic "retrieve–read" approach
	Advanced RAG	Optimized or specialized pre/post-retrieval methods	Fine-grained segments, metadata integration
	Modular RAG	Flexible architecture with additional functional modules	Search module, memory module, alignment module
Augmentation Stage	Pre-training	Enhancing pre-trained language models with retrieval	-
	Fine-tuning	Fine-tuning retrievers and generators for specific tasks	-
	Inference	Adding retrieval capabilities during inference	-
Augmentation Data Sources	Unstructured Data	Textual data from pure text corpora	Web documents, articles, other documents
	Structured Data	Knowledge stored in structured formats	Knowledge graphs, tables, databases
	LLM Generated	Content generated by LLMs themselves	LLM-generated contexts
Retrieval Process	Once Retrieval	Single retrieval-generation process	Text-based, semantic, vector, hybrid search
	Iterative Retrieval	Multiple retrievals based on previous results	Following multiple levels of references
	Adaptive Retrieval	Dynamic retrieval based on model judgments	Additional prompting in the retrieval process
Evaluation Methods	Independent	Separate evaluation of retriever and generator	Hit Rate, MRR, (Normalized) DCG, Precision
	End-to-End	Evaluation of final model response	Faithfulness, answer and context relevance
	Ability Testing	Testing specific capabilities for RAG	Noise robustness, negative rejection, performance in information integration

Table 8: Dimensions of Retrieval-Augmented Generation

RAG Paradigms RAG has evolved through three main paradigms [41]. The "naive" RAG represents the earliest methodology gaining prominence after the widespread adoption of ChatGPT, involving traditional indexing, retrieval, and generation processes in a basic "retrieve–read" framework. Advanced RAG, in comparison, proposes improvements by incorporating pre-retrieval and post-retrieval methods, optimizing indexing approaches, fine-grained segmentation, and including metadata. On the other side, modular RAG breaks away from the traditional framework, offering greater flexibil-

ity by integrating various methods to expand functional modules, such as incorporating search capabilities, memory and alignment modules.

Augmentation Stages The augmentation stage dimension in Table 8 illustrates when retrieval can be incorporated in the model lifecycle [41]. RAG systems could incorporate retrieval mechanisms (1) from the ground up for training, (2) in the fine-tuning stage or (3) at the inference stage/on-the-fly without modifying the underlying model parameters. The architecture proposed in this work demonstrates the application of pre-trained LLMs for KE, therefore relying on the inference-stage retrieval. Future work (see Section 8) includes details on the extension of the approach to (2) and (1), which is expected to further improve performance.[7]

Augmentation Data Sources RAG systems can utilize various types of knowledge repositories [41] including: unstructured data in pure NLT corpora (incl. web documents, articles etc.), structured data (incl. KGs, tables, and databases). Some RAG implementations even leverage content generated by LLMs themselves, creating a self-reinforcing knowledge loop.

Retrieval Process The retrieval process can be (i) singular, (ii) iterative and (iii) adaptive [41]. Once retrieval involves a single retrieval-generation process using text-based, semantic, vector search, or other search techniques. Iterative retrieval includes multiple levels of retrieval based on previous results. Adaptive retrieval dynamically adjusts the retrieval process based on model judgments implementing a more context-aware retrieval logic.

Evaluation Methods The final dimension outlines different general approaches to evaluating RAG systems [41]. Independent evaluation separately assesses the retriever and generator using metrics like hit rate, measures for assessing ranking quality (e.g., Mean Reciprocal Rank and Discounted Cumulative Gain) and Precision (see Table 7). End-to-end evaluation, on the other hand, examines the final model response. Another approach is ability testing, which focuses on *specific* capabilities crucial for RAG, such as noise robustness and negative rejection (e.g., performance and rejection in case of semantically similar documents, which are, however, not helpful for question answering), and information integration(e.g., to answer complex questions requiring integration of multiple data sources).

[7]Huggingface LLM Leaderboard suggests that fine-tuning pre-trained LLMs significantly improves performance for specific tasks, see: https://huggingface.co/spaces/open-llm-leaderboard/open_llm_leaderboard.

Notable Benefits Through implementation of methods described above, RAG improves performance of language models [41] for certain tasks: (1) by integrating provenance into the system, RAG reduces hallucinations and improves explainability of results and, therefore, trust; (2) reliance on external knowledge base or corpus addresses the knowledge cut-off and resource-intensive fine-tuning when integrating additional or actively evolving data. (3) Retrieval also allows models to be tailored to different domains or changing use cases by including new domain knowledge. The latter benefit is especially relevant in this work, as it allows to change the ontology based on the codebook or feature catalogue for a particular extraction task.

The flexibility and modular nature of RAG, as presented in Table 8, make it a powerful approach for adapting LLMs to specific tasks and use cases. Nonetheless, the provided flexibility also places a greater emphasis on rigorous evaluation and adapting evaluation approaches to particular tasks as will be discussed in Section 4.2.

3 Related Work

LLMs have emerged as powerful tools for IE and KE with a number of studies looking into applying LLMs to various parts of IE and KE pipelines. In this context, one particular area of application gaining prominence is KE from scientific texts. This section gives an overview of relevant literature on structured IE, KE with LLMs as well as the role of Semantic Web in the field. Thus, it focuses on the intersection between (a) IE and KE, (b) LLMs and (c) Semantic Web, LD and KGs.

LLM-supported Information Extraction A study published in Nature Communications [28] fine-tuned GPT-3 and Llama-2 to extract specialised knowledge in the chemistry domain, outputting results as English sentences and in a structured format (as JSON objects). Precision, recall and F1-Scores (as described in Table 7) were computed using an exact word-matching approach: i.e., each entity was first converted into a set of words. For example, `Bi2Te3 thin film` becomes {`‘‘Bi2Te3’’, ‘‘thin’’, ‘‘film’’`}. For relationships between entities, triplets of the form `(word1, word2, relationship_type)` were extracted. The scores were then computed by comparing the sets of true and predicted triplets (triple-level comparison). While the approached showed promise, reliability remained a concern, with F1-Scores generally around 0.5 except for one task reaching the score of 0.821, which, however, still could not be considered reliable.

LLM-based exaction has found applications in several specialized fields:

- Scientific Literature: LLMs are being used [136] to extract metadata from scientific literature, enabling scalable processing and analysis of large volumes of research. With research output (as a total number of articles published and available in bibliographic databases) following an exponential growth of approx. 5.6% annually [11], automatic processing of scientific literature becomes increasingly relevant.

- Clinical and Biomedical Domain: Pipelines in the clinical [3, 142] and biomedical domain [16, 22] have been proposed for IE from NLT.

- Material Science: and material science [98]. Here, zero-shot and few-shot extraction IE approaches for various tasks have been evaluated, including metadata extraction from study reports [134, 71, 113, 143, 121].

Evaluation in LLM-supported Information Extraction Broad application of LLMs in various fields has prompted the development of new evaluation metrics. Specifically, the issues [38] of (1) semantic consistency between model outputs and ground truth, whereby model might generate close, but not literally matching annotations, and (2) incompleteness of broadly used benchmarks, whereby a model might generate correct annotations, which, however, were not previously included in a benchmark.

Approaches based on LLMs and have been proposed to address these issues [38]. While such approaches effectively address erroneously-attributed underperformance of LLM-powered IE pipelines by proposing advanced metrics (`SQC-Score`), though the use of LLMs in evaluation, they add a level of methodological complexity which potentially limits the validity of results.

Information Extraction in the Context of Semantic Web The intersection of IE and Semantic Web represents a particularly rich area of research, where information is not only extracted, but also represented in machine-readable and interoperable format, following LD [10] and FAIR data principles [48].

Before the advent of LLMs, the popularity of such approaches increased immensely with the availability of Wikipedia-related KGs such as DBPedia [80] and Wikidata [133]. One of the most prominent approaches for IE with Semantic Web follow the EEL architecture [75].

In EEL, the IE process is comprised in two conceptual steps: entity recognition and linking. First, entity mentions are identified in the text using string/token and NER-based approaches [75]. Then, in the disambiguation

phase, entities are linked to external reference through a wide range of approaches [75].

LLM-supported Knowledge Extraction Recently, LLMs have demonstrated potential in KE, extracting LD across various domains. In this context, most approaches target NLT as input data for extraction (e.g., [40]).

- **Biomedical domain:** LLMs have been extensively applied to extract LD in the biomedical domain [109, 17, 60].
- **Conversational UI:** Recent work [87] explores using LLMs to extract structured knowledge from conversational interfaces and integrating it with existing KGs.
- **Cultural heritage:** The application of LLMs for KE extends to the cultural heritage domain [85, 74].

Beyond these specific domains, LLMs have been applied to extract knowledge from general NLT across diverse contexts. Annotated datasets have been proposed for evaluation of transformer-based extraction from scholarly publications [97]. Furthermore, LLMs have demonstrated applied potential for KE in the manufacturing domain [78], where they can extract technical specifications, process parameters, and domain-specific relationships.

Ontology Learning While this work focuses on using existing ontologies to guide KE, with defining broader categories to enable inductive QC, the complementary field of OL targets the automatic creation of the ontologies themselves. A number of approaches [59] could potentially feed into the ontology-guided extraction pipeline proposed in our work. OL creates a potential positive feedback loop where extraction benefits from better ontologies, and the extracted knowledge could, in turn, enable ontology refinement. This connection between OL and LLM-based KE represents a promising direction for creating adaptive knowledge systems that evolve their ontology based on encountered information, further enhancing the extraction performance.

4 Methodology

This section discusses the methodology for: (1) creating a automated extraction pipeline, in Section 4.1; (2) evaluating the extraction reliability, in Section 4.2; (3) evaluating the extraction pipeline on concrete use cases in

research ecosystem, in Section 4.3. In describing the requirements, this section refers to specific aspects of related methodologies described in Section 2. It also establishes parallels to existing pipelines described in Section 3.

4.1 Creating the Extraction Pipeline

This section will review the requirements towards a pipeline that could replace manual feature extraction approaches. Here, we refer to both *coding* (i.e., assignment of codes in the sense of QC) and feature extraction in the sense of IE and KE under the broader term of *feature extraction*. In this sense, feature extraction includes both assignment of salient categories to NLT (as classifications of entities identified in the text) as well as extraction of quantitative data as annotations and extraction of semantics according to pre-defined ontological patterns.

Requirements The pipeline should enable both deductive and inductive approaches for QC (see Section 2.1.1) and CA (see Section 2.1.2), thus enabling both explorative extraction without pre-defined categories or with the possibility of extending an incomplete/transient set of categories (open schema, see Table 2) as well as theoretically-guided extraction (fixed schema) of a predefined, bound set of categories as well as annotations following theoretically-informed constraints. Furthermore, the approach should enable iterative category evolution (see Table 3) and qualitative CA (see Section 2.1.2). To enable quantitative CA, the approach should also support extraction of different numerical values (incl. datatypes) (see Figure 3).

Input Granularity Requirements The pipeline must flexibly handle various levels of input granularity, ranging from individual labels (such as headlines in the example from SLR, see Appendix A) and phrases to larger segments (e.g., paragraphs, posts, interview transcripts) and entire documents. This flexibility ensures that extraction can work with fine-grained annotations as well as broader document-level analysis, depending on the specific extraction need. Such granularity range enables both detailed annotation similar to manual coding of specific phrases in QC and broader analysis of text segments.

Extraction Mode Requirements The pipeline should support both deductive (top-down) and inductive (bottom-up) approaches. In deductive approaches, the system should apply predefined categories and annotations to the data, as in Directed CA. With inductive approaches, the system discovers

categories and patterns emerging from the data without a fixed framework of categories, similar to Open Coding in GT. Mixed approaches, which combine elements of both, should also be supported to accommodate the iterative nature of qualitative research. This requirement enables both exploratory research (where patterns emerge from the data) and theory-driven research (where feature extraction is driven by established frameworks).

Output Structure Requirements The pipeline must produce structured output implementing structures associated with QC, CA, but also KE. They include categorical assignments (classifying text into defined categories), numerical values with their associated units of measurement, and relationships informed by ontologies (incl. taxonomic hierarchies). This diversity of output formats supports both qualitative and quantitative analysis of the extracted information, and required a data model capable of flexibly representing data structures.

Provenance Requirements For transparency and validation purposes, the pipeline must maintain clear links to the source material through text citations (also referred to as *evidence* in this work). Each extraction should be traceable back to its origin in the text, allowing verification of the extraction's accuracy and validity. This requirement is crucial for maintaining the reliability and reproducibility, but is rarely contained in research data published as a result of QC studies.

Process Requirements The extraction process must be iterative, allowing for refinement and evolution of the extraction model. This iterative architecture supports the progressive improvement of results and aligns with established practices in QC and CA, where understanding deepens through multiple analytical passes. The iterative nature of the process also facilitates the kind of conceptual/theory development that occurs in the transition from initial cycle to further cycles of coding in traditional QC (see Section 2.1.1).

Ontological Grounding Since complex relationships are part of the output scope, we will specify valid RDF annotations as output format in line with OBIE as illustrated in Figure 4. To follow the expected structure of the output, the pipeline should accept a domain-specific ontology defining features of interest, patterns and constraints, e.g. for datatypes as well as complex shapes.

Consequently, ontological grounding not only enables the automatic structuring of results as LD – facilitating seamless integration with LOD and cre-

ating a readily publishable, self-descriptive resource beneficial to researchers – but also supports automatic validation, inferencing, and, potentially, repairs.

Pipeline Modules Following these requirements, we will adapt and extend appropriate components of RAG as described in Section 2.5. We will demonstrate, how different modules of the pipeline fulfil the requirements and integrate ontologies in various stages of the process. Most importantly, we will detail, how the modules in this hybrid approach enable effective and diverse feature extraction.

Language Models For evaluation purposes, the pipeline must accommodate multiple LLMs, including both open-source and commercial models. The architecture should facilitate straightforward substitution of language models without necessitating alterations to the pipeline's implementation.

Overview of Language Models The choice of language models define inputs that could be used for evaluation. Table 9 gives an overview of a selection of recent pre-trained LLMs, their context window size and knowledge cut-off dates. In this comparison, Gemini 2.0 Pro (Experimental) has the most recent cut-off date (August 2024).

Model	Version	Context Window	Max Output	Training Cutoff	Source
Gemini 2.0 Pro[a]	gemini-2.0-pro-exp-02-05	2M	8,192	Aug 2024	Closed
DeepSeek-R1[b]	deepseek-r1	128K	32,768	July 2024	Open
Claude 3.5 Sonnet[c]	claude-3-5-sonnet-20241022	200K	8,192	April 2024	Closed
Llama 3.3 70B[d]	llama-3.3-70b-instruct	128K	2,048	Dec 2023	Open
GPT-4o[e]	gpt-4o-2024-11-20	128K	16,384	Oct 2023	Closed

[a] See https://blog.google/technology/google-deepmind/gemini-model-updates-february-2025/
[b] See https://huggingface.co/deepseek-ai/DeepSeek-R1
[c] See https://docs.anthropic.com/en/docs/about-claude/models
[d] See https://huggingface.co/meta-llama/Llama-3.3-70B-Instruct1
[e] See https://platform.openai.com/docs/models#gpt-4o

Table 9: Comparison of Large Language Models

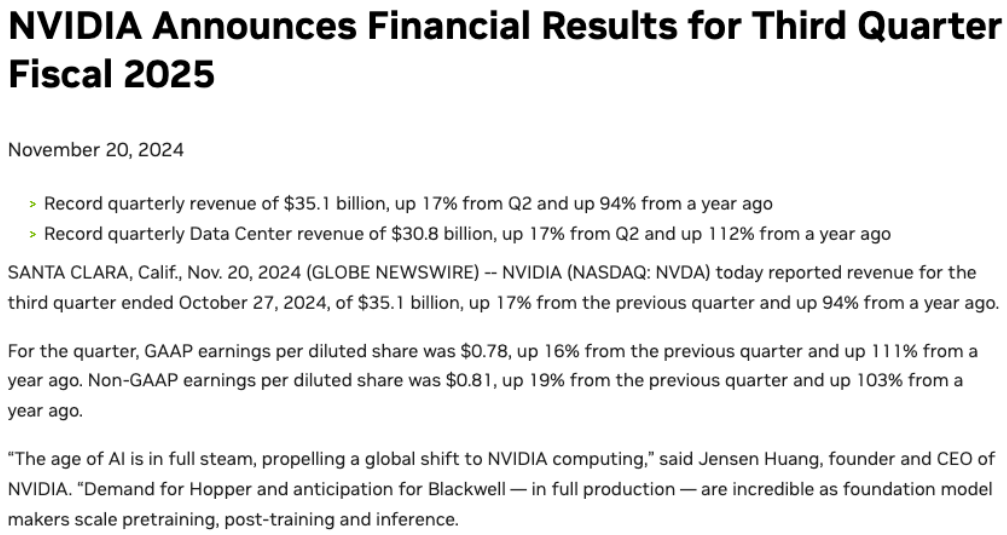
Running Examples As a running example, this work uses Bram Stoker's *Dracula* [117]. This example merely serves as an illustration and, due to the work potentially being included in the knowledge before the cut-off date of LLMs, we will not use this example for evaluation. However, the journal-like format of Dracula resembles field notes, which are a common input source for QC. As an applied task, we are interested in extracting features of journal entries (author, location, time) from chapter. Figure 5 gives an example for an excerpt from a chapter.

> *JONATHAN HARKER'S JOURNAL (Kept in shorthand.)*
> *3 May. Bistritz. – Left Munich at 8:35 P.M., on 1st May, arriving at Vienna early next morning; should have arrived at 6:46, but train was an hour late. Budapest seems a wonderful place, from the glimpse which I got of it from the train and the little I could walk through the streets. I feared to go very far from the station, as we had arrived late and would start as near the correct time as possible. The impression I had was that we were leaving the West and entering the East; the most western of splendid bridges over the Danube, which is here of noble width and depth, took us among the traditions of Turkish rule. [...]*

Figure 5: Excerpt from the 1st Chapter of *Dracula* by Bram Stoker, 1897.

As the second running example which lies beyond the knowledge cut-off date of LLMs mentioned in Table 9, we will use the use the Financial Results Year Press Release of NVIDIA (Q3 of the fiscal year 2025).[8] Figure 6 illustrates the press release from NVidia's website.

[8] See https://nvidianews.nvidia.com/news/nvidia-announces-financial-results-for-third-quarter-fiscal-2025



NVIDIA Announces Financial Results for Third Quarter Fiscal 2025

November 20, 2024

- Record quarterly revenue of $35.1 billion, up 17% from Q2 and up 94% from a year ago
- Record quarterly Data Center revenue of $30.8 billion, up 17% from Q2 and up 112% from a year ago

SANTA CLARA, Calif., Nov. 20, 2024 (GLOBE NEWSWIRE) -- NVIDIA (NASDAQ: NVDA) today reported revenue for the third quarter ended October 27, 2024, of $35.1 billion, up 17% from the previous quarter and up 94% from a year ago.

For the quarter, GAAP earnings per diluted share was $0.78, up 16% from the previous quarter and up 111% from a year ago. Non-GAAP earnings per diluted share was $0.81, up 19% from the previous quarter and up 103% from a year ago.

"The age of AI is in full steam, propelling a global shift to NVIDIA computing," said Jensen Huang, founder and CEO of NVIDIA. "Demand for Hopper and anticipation for Blackwell — in full production — are incredible as foundation model makers scale pretraining, post-training and inference.

Figure 6: Press Release of Financial Results for Third Quarter of NVIDIA's Fiscal Year 2025

In this example, in comparison to the Dracula example, we will be more interested in extracting numerical values. Specifically, the example will describe a task of extracting key financial performance indicators and CEO's quotes from the press release, which, for example, could be valuable for further aspect-based sentiment analysis.

4.2 Requirements for Reliability Assessments

This section reviews the requirements for adapting reliability assessment methodologies to the ontology-driven KE pipeline. A comprehensive evaluation framework must address two fundamental dimensions: (1) the pipeline's extraction accuracy when measured against a complete ground truth dataset, and (2) its performance relative to human annotators and other LLMs. The former establishes objective performance metrics for the extraction process, allowing conclusions on the absolute performance of the pipeline while the latter contextualizes these results in comparison to human annotators as a potential co-annotator or replacement for manual annotation. Crucially, the methodological adaptations required for each dimension must account for the graph structure of ontology-guided extractions while maintaining comparability with established reliability metrics in both QC and IE domains.

Adapting Ground Truth Evaluation For deductive feature extraction, the adapted approach must accommodate the hierarchical nature of ontology-

based annotations. Furthermore, the methodology must establish clear metrics for assessing the completeness of extraction relative to ground truth datasets and provide strategies for reformulating the unbounded tasks in such a way as to enable evaluation.

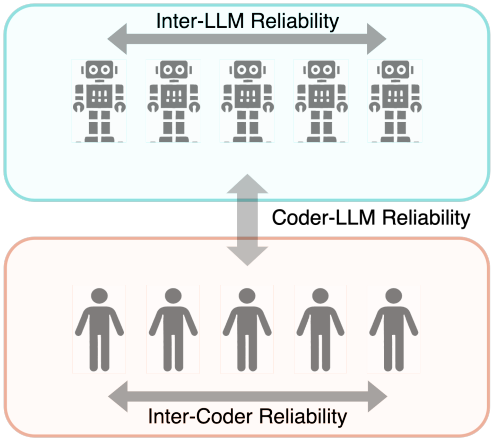


Figure 7: Reliability Assessment in Multiagent Context

Inter-LLM Reliability To assess extraction consistency across LLMs, this work adapts ICR to measuring agreement between the models. This approach will adapt established metrics (see Section 2.3.3) for graph-structured outputs, discussing blank node matching and semantic equivalence. The framework should evaluate agreement at multiple granularity levels and address methodological challenges when calculating chance-corrected agreement metrics. It should also establish approaches to building consensus (a procedure normally relying on coordination between human annotators) in case of differences in annotation. Figure 7 illustrates the concept of Inter-LLM Reliability.

Coder-LLM Reliability Coder-LLM reliability is evaluated to ultimately assess the applicability of LLM-supported extraction as compered to manual extraction. Therefore, for assessing human-LLM agreement, the human and LLM annotations are aggregated into two consolidated annotations respectively (as depicted in Figure 7 by blue and orange boxes). The final reliability scores are accordingly calculated based on those two consolidated annotations.

4.3 Application and Evaluation

To evaluate the application of the proposed pipeline to genuine research challenges in the scholarly ecosystem, this work will present results of extraction of conference-related knowledge. It will detail the evaluation across two use cases: (1) extracting conference-related knowledge from conference websites as demonstrated in the Semantic Observer framework [33], and (2) extracting comprehensive scholarly knowledge from websites and conference proceedings as implemented in the Scholarly Wikidata approach [82].

For each application, this work will discuss both extraction reliability and practical utility. Reliability assessment will apply the metrics established in 5.2, including Precision, Recall, and F1-Scores at the triple level. The practical utility evaluation will discuss the potential of the extracted knowledge to enhance research workflows through improved knowledge accessibility, searchability, and integration with existing KGs like Wikidata. This approach will provide concrete evidence of the LLM-supported KE's applicability to addressing real-world challenges in research knowledge management.

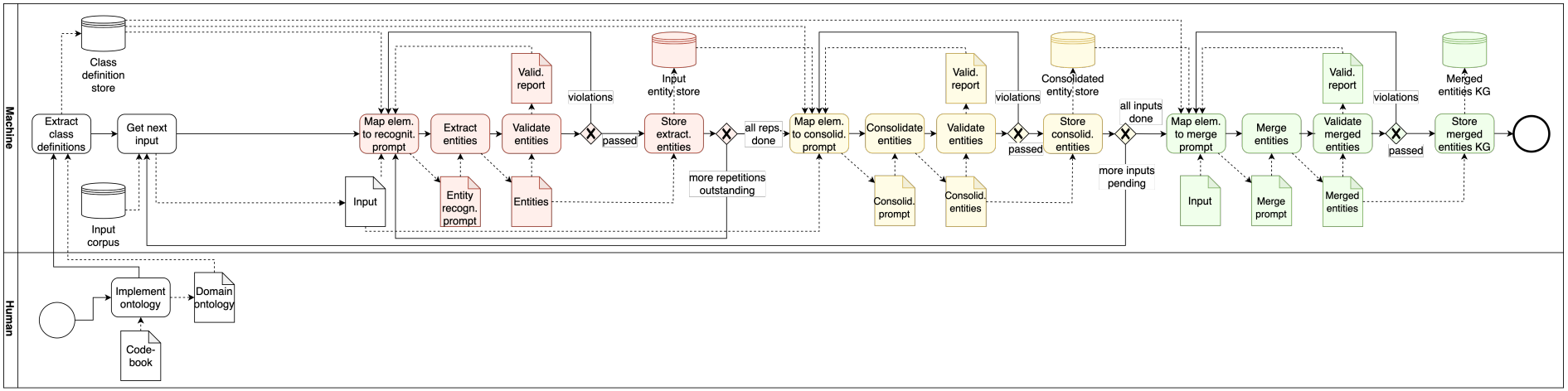
5 Theoretical Framework

The theoretical framework described in this section begins with the implementation details of the pipeline, then describes the methodology for LLM-specific golden standard studies, and, finally, approaches to calculating inter-LLM and human-LLM agreements.

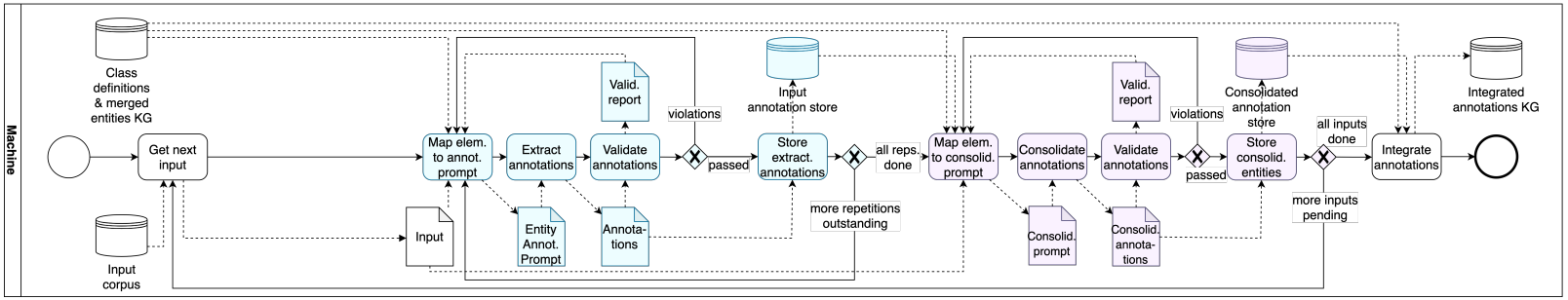
5.1 LLM-based Extraction Pipeline

The first part of this section – which describes the pipeline itself – is based on the requirements listed in Section 4.1. Figure 8 illustrates the simplified version of the pipeline: specifically, Figure 8a illustrates the first phase of the pipeline, the entity recognition phase, and Figure 8b illustrates the second, the annotation phase.

Phases The first phase of the pipeline, entity recognition, aims, based on the domain ontology, to identify and classify entities in each text input. The second phase, then, is tasked with annotating (incl. linking) the entities based on the definitions of the domain ontology and input text.



(a) Entity Recognition Phase



(b) Annotation Phase

Figure 8: BPMN Diagrams of Ontology-Guided Knowledge Extraction Pipeline

Module Overview In line with the modular RAG design, the pipeline introduces three module types (*Entity/Annotation Extraction*, *Entity/Annotation Consolidation* and *Entity Merge*) that can be combined to create increasingly sophisticated LLM-supported extraction pipelines. Figure 9 illustrates these modules with the same colour codes as the ones used to highlight the modules in Figure 8.

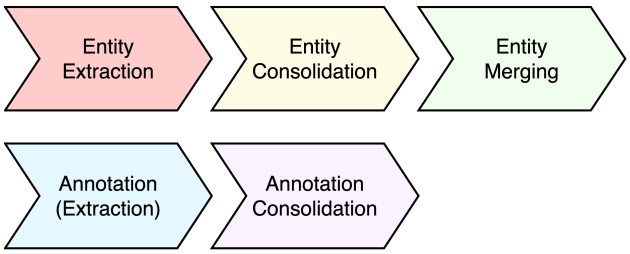


Figure 9: Core Modules of the Pipeline

Extraction Modules The Extraction Modules prompt the LLM to either extract entities from the text or annotate entities with specific property values from text. In the first case, the Entity Extraction Module instructs the LLM to identify entities corresponding to ontology-defined classes. The procedural implementation details of the module as well as relevant data sources are captured in Figure 8a. In the second case, the Annotation Extraction Module directs the LLM to simultaneously annotate multiple properties for these entities. The implementation details for the second module are presented in Figure 8b.

Consolidation Modules The Consolidation Modules process outputs from multiple validated extraction or annotation steps with identical inputs. These modules prompt the LLM to integrate the results by resolving duplicates and merging overlapping or complementary information, thereby producing a coherent, consolidated output.

Entity Merging The Entity Merging Module processes consolidated outputs from consecutive text inputs and integrates duplicates. If the number of input text segments exceeds the number of inputs simultaneously processed by the module, the outputs from each merge iteration become inputs for subsequent merging steps. This recursive merging continues until all entities extracted from the original segments are fully consolidated.

Validation Submodule Each module includes a validation submodule responsible for verifying the correctness of extraction results. If errors or inconsistencies are detected, the submodule generates a validation report. This report, together with the previous extraction output, is passed back to the module to facilitate corrections. Optionally, a *mandatory revalidation parameter* could be set, ensuring iterative re-validation even if the initial validation passes. Additionally, a *maximum retry parameter* may be specified to limit the number of re-validation attempts, ensuring that the validation process halts after a predefined number of unsuccessful retries.

5.1.1 Knowledge Extraction Process

This subsection illustrates the two-phase process (entity recognition and property annotation) applied to the running examples. It builds up the ontologies step-by-step, illustrating the important aspects that are required for the ontology to underlie the extraction process. Then, it illustrates how the modules discussed above produce incremental output, completing the generated RDF data.

Domain-Specific Ontology The first step in the extraction process is defining a domain-specific ontology that will be used for automatic extraction. In the context of QC, the ontology could be derived from existing codebooks or feature catalogues used in qualitative research and formalized using RDF and OWL following an established ontology engineering methodology [127], possibly involving ontology re-use.

For optimal extraction performance, the ontology should include rich textual descriptions: classes and properties require descriptive labels and detailed `rdfs:comment` annotations that provide (i) comprehensive definitions of concepts, (ii) examples, and (iii) crucial or complex constraints described in natural language. Classes are formally defined using `owl:Class` definitions, while properties are specified as either `owl:ObjectProperty` or `owl:DatatypeProperty` depending on their intended range.

Validation constraints are implemented through SHACL to enforce various types of constraints: `sh:NodeShape` shapes validate entity instances, while `sh:PropertyShape` shapes should be used to enforce cardinality, datatype restrictions, and value constraints on properties. Notably, OWL should not be used for expressing constraints due to its standard semantics being designed for inference rather than expressing constraints [63] and its inherent Open World Assumption impeding the use for validation.

The combination of semantic definitions and validation constraints creates an ontology that can guide LLM extraction process. For the Dracula run-

ning example, we will create an ontology focused on describing persons and locations, extending FOAF where appropriate.[9] First, the relevant classes with useful descriptions need to be defined (Figure 10).

```
@prefix : http://semantic.foundation/dracula# .
@prefix rdf: http://www.w3.org/1999/02/22-rdf-syntax-ns# .
@prefix rdfs: http://www.w3.org/2000/01/rdf-schema# .
@prefix xsd: http://www.w3.org/2001/XMLSchema# .
@prefix owl: http://www.w3.org/2002/07/owl# .
@prefix foaf: http://xmlns.com/foaf/0.1/ .
@prefix sh: http://www.w3.org/ns/shacl# .

:Person a owl:Class ;
    rdfs:subClassOf foaf:Person ;
    rdfs:label "Person"@en ;
    rdfs:comment A character in the novel 'Dracula' who is explicitly named. Examples:
        Jonathan Harker, Count Dracula, Mina Murray. Constraint: Must be a named
        individual, not a generic reference to a group of people.@en .

:Location a owl:Class ;
    rdfs:label "Location"@en ;
    rdfs:comment A named geographical place mentioned in the text, including cities,
        towns, regions, countries, or specific landmarks. Examples: London,
        Transylvania, Castle Dracula, Whitby. Constraint: Must be a specific named
        location, not a general area or direction like 'eastward' or 'the
        countryside'.@en .
```

Figure 10: Class Definitions in the Dracula Ontology

Then, the basic properties can be defined (Figure 11).

[9] See http://xmlns.com/foaf/spec/

```
:firstName a owl:DatatypeProperty ;
   rdfs:subPropertyOf foaf:firstName ;
   rdfs:label "first name"@en ;
   rdfs:comment The first or given name of a character. Examples: Andrew, Alexander,
       Maria. Constraint: Extract only the given name, not titles or surnames.@en ;
   rdfs:domain :Person ;
   rdfs:range xsd:string .

:surname a owl:DatatypeProperty ;
   rdfs:subPropertyOf foaf:familyName ;
   rdfs:label "surname"@en ;
   rdfs:comment The family or last name of a character. Examples: Smith. Constraint:
       Extract only the surname, not titles or first names.@en ;
   rdfs:domain :Person ;
   rdfs:range xsd:string .

:knows a owl:ObjectProperty ;
   rdfs:subPropertyOf foaf:knows ;
   rdfs:label "knows"@en ;
   rdfs:comment Indicates that one character knows another character personally.
       Constraint: Only mark a relationship when there is explicit evidence of personal
       acquaintance.@en ;
   rdfs:domain :Person ;
   rdfs:range :Person .

:visits a owl:ObjectProperty ;
   rdfs:label "visits"@en ;
   rdfs:comment Indicates that a character has physically traveled to and been present
       at a location. Examples: Maria visits London. Constraint: Must be an actual
       visit explicitly described in the text, not just a mention of a location.@en ;
   rdfs:domain :Person ;
   rdfs:range :Location .
```

Figure 11: Property Definitions in the Dracula Ontology

In addition to classes and properties, constraints should be defined for validation. For the Dracula ontology, the Turtle excerpt below defines SHACL constraints to enforce structural and datatype constraints.

```
:PersonShape a sh:NodeShape ;
   sh:targetClass :Person ;
   sh:property [
       sh:path :firstName ;
       sh:name "First Name Constraint" ;
       sh:description "Person's first name should be at least 2 characters long." ;
       sh:datatype xsd:string ;
       sh:minLength 2 ;
   ] .

:KnowsShape a sh:PropertyShape ;
   sh:path :knows ;
   sh:description "Validates that knows relationship points to valid Person" ;
   sh:class :Person ;
   sh:nodeKind sh:IRI .

:VisitsShape a sh:PropertyShape ;
   sh:path :visits ;
   sh:description "Validates that visits relationship points to valid Location" ;
   sh:class :Location ;
   sh:nodeKind sh:IRI .
```

Figure 12: SHACL Constraints for the Dracula Ontology

For the NVIDIA financial press release example, a simple financial reporting ontology focusing on key numerical data and CEO's statements is created. Here, it would be appropriate to reuse the FIBO ontology, though the example does not.[10] Note that this example defines a different default-/base namespace.

[10]See https://spec.edmcouncil.org/fibo/

```
@prefix : http://semantic.foundation/financial# .

:FinancialReport a owl:Class ;
   rdfs:label "Financial Report"@en ;
   rdfs:comment A corporate financial report. Constraint: Must be a press release about
       financial results.@en .

:quarterlyRevenue a owl:DatatypeProperty ;
   rdfs:label "quarterly revenue"@en ;
   rdfs:domain :FinancialReport ;
   rdfs:range xsd:decimal ;
   rdfs:comment The total revenue for the reported quarter in US dollars.@en ;

:netIncome a owl:DatatypeProperty ;
   rdfs:label "net income"@en ;
   rdfs:domain :FinancialReport ;
   rdfs:range xsd:decimal ;
   rdfs:comment The company's profit after all expenses in US dollars.@en ;

:ceoQuoteText a owl:DatatypeProperty ;
   rdfs:label "CEO quote"@en ;
   rdfs:domain :FinancialReport ;
   rdfs:range xsd:string ;
   rdfs:comment Direct quotation from the company's CEO. Constraint: Must be verbatim
       text with attribution.@en ;
```

Figure 13: Minimal Financial Reporting Ontology

For automatic validation, SHACL constraints should also be defined.

```
:FinancialReportShape a sh:NodeShape ;
    sh:targetClass :FinancialReport ;
    sh:property [
        sh:path :quarterlyRevenue ;
        sh:datatype xsd:decimal ;
        sh:minCount 1 ;
        sh:maxCount 1 ;
        sh:message "Each financial report must specify exactly one quarterly revenue as
            a decimal value." ;
    ] ;
    sh:property [
        sh:path :netIncome ;
        sh:datatype xsd:decimal ;
        sh:minCount 1 ;
        sh:maxCount 1 ;
        sh:message "Each financial report must specify exactly one net income as a
            decimal value." ;
    ] ;
    sh:property [
        sh:path :ceoQuoteText ;
        sh:datatype xsd:string ;
    ] .
```

Figure 14: SHACL Constraints for Minimal Financial Reporting Ontology

Input Corpus The input corpus is processed sequentially (though parallel processing is possible), with each document taken as a text input or split up into paragraphs. The choice of segmentation (e.g., by chapter, by entry, or by paragraph) depends on the specific requirements for feature extraction. A notable limitation constraining the input length is the context window (i.e., the whole composite prompt should be less than the context window). However, also generation limit poses a constraint – i.e., how many tokens can be generated at once. Together, it has to be considered that the results can be represented in the output of this length.

For the Dracula running example, the chapters are split into paragraphs, where each paragraph is presented as a single input to the extraction pipeline. The example paragraph has already been presented in Figure 5. For the NVIDIA running example, the press release excerpts as shown in Figure 6 are taken.

Class Definitions The pipeline extracts class definitions from ontologies by querying for entities of type `owl:Class` and `rdfs:Class`, collecting their labels, comments, and hierarchical relationships. For each class, it identifies associated properties through two mechanisms: first by finding tradi-

tional Semantic Web properties where the class appears as `rdfs:domain` or `rdfs:range`, and second by examining SHACL shape constraints that target the class, including relevant `sh:NodeShape` declarations and property paths.

Entity Extraction with Evidence The pipeline extracts entities from text using a prompt where the target class(es), class definition(s) (including properties and constraints), input text segment, and the latest output and validation report are included. For each identified entity, the extraction process captures supporting evidence directly from the text using `rdfs:comment` predicates. This evidence consists of exact text segments that justify the entity's classification – the same provenance information is also included in the output of other modules. Multiple evidence statements can be associated with a single entity, providing comprehensive justification for its identification. For example, when extracting instances of `foaf:Person` from Stoker's Dracula, the entity "Count Dracula" is identified with multiple supporting evidence statements:

```
[] a foaf:Person ;
    rdfs:label "Count Dracula" ;
    rdfs:comment "Count Dracula had directed me to go to the Golden Krone Hotel, which
        I found, to my great delight, to be thoroughly old-fashioned, for of course I
        wanted to see all I could of the ways of the country.",
        "I was not able to light on any map or work giving the exact locality of the
            Castle Dracula, as there are no maps of this country as yet to compare with
            our own Ordnance Survey maps; but I found that Bistritz, the post town
            named by Count Dracula, is a fairly well-known place." .
```

Figure 15: Entity Extraction with Evidence in the Dracula Example

This evidence-based approach ensures that each extracted entity is connected to the source text, facilitating validation and traceability.

```
[] a financial:FinancialReport ;
   rdfs:label "NVIDIA Q3 Fiscal 2025 Financial Report" ;
   rdfs:comment "NVIDIA (NASDAQ: NVDA) today reported revenue for the third quarter
       ended October 27, 2024, of $35.1 billion, up 17% from the previous quarter and
       up 94% from a year ago.",
       "For the quarter, GAAP earnings per diluted share was $0.78, up 16% from the
           previous quarter and up 111% from a year ago. Non-GAAP earnings per diluted
           share was $0.81, up 19% from the previous quarter and up 103% from a year
           ago." .
```

Figure 16: Entity Extraction with Evidence in the NVIDIA Example

Annotation with Evidence Annotation with evidence involves associating annotations explicitly with direct textual evidence, using the standard RDF reification [13]. While alternative, less verbose approaches have been proposed [49] including approaches specifically used in Wikidata [39], the standard RDF reification provides a mechanism to make statements about statements, allowing the attachment of evidence to specific triples. For instance, when annotating Jonathan Harker's acquaintance with Count Dracula in Stoker's *Dracula*, RDF reification is used as follows:

```
[] a rdf:Statement ;
    rdf:subject _:b1 ;
    rdf:predicate :knows ;
    rdf:object _:b2 ;
    rdfs:comment "(Mem., I must ask the Count all about them.)"^^xsd:string .

_:b1 a :Person ;
    rdfs:label "Jonathan Harker" ;
    rdfs:comment "JONATHAN HARKERS JOURNAL" .

_:b2 a foaf:Person ;
    rdfs:label "Count Dracula" ;
    rdfs:comment "Count Dracula had directed me to go to the Golden Krone Hotel, which
        I found, to my great delight, to be thoroughly old-fashioned, for of course I
        wanted to see all I could of the ways of the country.",
        "I was not able to light on any map or work giving the exact locality of the
            Castle Dracula, as there are no maps of this country as yet to compare with
            our own Ordnance Survey maps; but I found that Bistritz, the post town
            named by Count Dracula, is a fairly well-known place." .
```

Figure 17: Annotation with RDF Reification Providing Evidence for Jonathan Harker's acquaintance with Count Dracula

Here, the RDF reification technique is used to explicitly represent the relationship between Jonathan Harker and Count Dracula, associating it with supporting textual evidence. Note, the pipeline does represent all blank nodes in a skolemized manner [73] until the last processing step, when the skolemized entities are replaced by genuine blank nodes.

```
_:b1 a financial:FinancialReport ;
   rdfs:label "NVIDIA Q3 Fiscal 2025 Financial Report" ;
   financial:quarterlyRevenue "35100000000"^^xsd:decimal ;
   financial:netIncome "19309000000"^^xsd:decimal ;
   financial:ceoQuoteText "The age of AI is in full steam, propelling a global shift to
       NVIDIA computing. Demand for Hopper and anticipation for Blackwell  in full
       production  are incredible as foundation model makers scale pretraining,
       post-training and inference. AI is transforming every industry, company and
       country. Enterprises are adopting agentic AI to revolutionize workflows.
       Industrial robotics investments are surging with breakthroughs in physical AI.
       And countries have awakened to the importance of developing their national AI
       and infrastructure." ;
   rdfs:comment "NVIDIA (NASDAQ: NVDA) today reported revenue for the third quarter
       ended October 27, 2024, of $35.1 billion, up 17% from the previous quarter and
       up 94% from a year ago.",
       "For the quarter, GAAP earnings per diluted share was $0.78, up 16% from the
           previous quarter and up 111% from a year ago. Non-GAAP earnings per diluted
           share was $0.81, up 19% from the previous quarter and up 103% from a year
           ago.".

[] a rdf:Statement ;
    rdf:subject _:b1 ;
    rdf:predicate financial:quarterlyRevenue ;
    rdf:object "35100000000"^^xsd:decimal ;
    rdfs:comment "Record quarterly revenue of $35.1 billion, up 17% from Q2 and up 94%
        from a year ago."^^xsd:string .

[] a rdf:Statement ;
    rdf:subject _:b1 ;
    rdf:predicate financial:netIncome ;
    rdf:object "19309000000"^^xsd:decimal ;
    rdfs:comment "Net income $19,309 million, up 16% from the previous quarter and up
        109% from a year ago."^^xsd:string .
```

Figure 18: Annotation with RDF Reification Providing Evidence for NVIDIA's Financial Report

In this case, the structured RDF reification associates each annotation with textual evidence as well, providing explicit connection to the original source text and enhancing transparency for validation purposes.

The presented examples demonstrated the main aspects of how ontologies could be used to guide RAG-supported KE pipelines. Additional technical details and full documentation on the pipeline are made available online.[11]

[11] https://git.ai.wu.ac.at/dobriy/rag-ke

5.2 LLM-specific Golden Standard Studies

Golden standard studies focus on evaluating extractions against a human-validated ground truth. For evaluating LLM extraction performance against golden standards, we adapt traditional metrics while accounting for the semantic nature of RDF output. For QC, the analysis is performed on the level of annotated codes, which is a flat structure. For feature extraction in general, the most common level of analysis is key-value pairs. In LLM-supported KE, the analysis can be performed on various levels of granularity as illustrated in Table 10.

Evaluation Level	***Precision***	***Recall***	***F1-Score***	***Specificity***	***Accuracy***	***MCC***	***Informedn.***	***Markedn.***
Triple Level	✓	✓	✓	(✓)	(✓)	(✓)	(✓)	(✓)
Entity Level	✓	✓	✓	(✓)	(✓)	(✓)	(✓)	(✓)
Property Level	✓	✓	✓	(✓)	(✓)	(✓)	(✓)	(✓)
Value Level	✓	✓	✓	(✓)	(✓)	(✓)	(✓)	(✓)

Table 10: Evaluation Metrics for Golden Standard Studies by Different Levels of Evaluation

While metrics with ✓are always directly applicable (e.g., Precision, Recall, F1-Score), metrics with (✓) can be applied with limitations or meaningfully applied only if you the extraction task is framed as a bounded classification problem (e.g., Specificity, Accuracy, MCC, Informedness, Markedness and further metrics). The paragraphs below discuss the various levels of evaluation and describe how different levels of evaluation can be treated as bounded classification problems.

Triple-level Evaluation This level provides an overall assessment of the extraction reliability. It is more general than code-level evaluation in QC and key-value pair evaluation in IE and focuses on all, the correct underlying entity of description, the correct property for annotation and the correct extracted value. Thus, high reliability on this level shows the overall reliability of the pipeline as a whole.

Entity-level Evaluation This level of evaluation is similar to the code-level evaluation in QC and therefore should be used in comparisons to QC

study results. On this level, entity recognition can be evaluated separately. When a more general class is defined in the ontology, extracted items can be interpreted as open codes. Defining more precise categories transforms the extraction into a bounded classification task which allows application of advanced evaluation metrics. In general, this level evaluates the quality and comprehensiveness of class descriptions in the ontology.

Property-level Evaluation This level of evaluation assesses how accurately the models assigns properties and relationships to correctly-identified entities. It is not concerned with the correctness of value extraction or entity recognition. Property-level evaluation enables conclusions about extraction of characteristics of objects, which includes the pipeline's ability to recognize the presence of a characteristic and distinguish between similar properties. In general, this level evaluates the quality and comprehensiveness of property descriptions in the ontology, whereby potential under-extraction or over-generalization which could highlight the need for more concrete descriptions in the ontology.

Value-level Evaluation This evaluation level focuses explicitly on the reliability of entities, literal values and datatypes assigned to properties. This level of evaluation can distinguish the reliability of (a) entity linking – assigning entity values to properties, and (b) literal value and datatype extraction. This level of evaluation, together with property-level evaluation, could provide a pendant to property-value annotations. In such cases, the underlying entity is less important and a general class can be defined in the ontology to match a variety of entities. However it should be still evaluated that the general class matches all instances of property-value as a pre-requisite for evaluation.

Evidence Furthermore, on each level, evidence citations can be also evaluated together with the actual extractions. Despite evidence being collected for entity recognition and annotation processes, it can be used to support evaluation on every level: the evidence collected during entity recognition can be used to evaluate extraction on entity level and the evidence collected during the annotation process can be used at triple level, but also supporting property-level extraction and value-level extraction at the same time.

5.2.1 Evaluation Reporting

The multi-level evaluation approach combining different layers of extraction provides a comprehensive assessment of extraction quality while considering

the semantic nature of RDF output. The metrics can then be aggregated into a final score that weights different aspects according to task requirements.

5.2.2 Possible Interventions

Several interventions could be specified to improve pipeline performance. These interventions are divided into pipeline-level changes (targeting specific parameters of the extraction process) and ontology-level refinements (targeting specific parts of the guiding ontology).

Pipeline-level interventions focus on increasing the depth and granularity of the pipeline:

P1: Increase number of extraction runs per text: Performing multiple simultaneous extractions for each text segment can help to create a broader pool to aggregate from and improve overall extraction coverage.

P2: Increase number of extraction evaluations: Multiple evaluation cycles can enhance robustness and reliability of extraction by repeating evaluations and reducing random errors.

P3: Increase input text granularity: Processing smaller input segments or fewer paragraphs at once allows the model to focus on finer details and improve extraction precision and completeness. However, input segments that are too short could impede contextual understanding and make co-reference resolution during consolidation more error-prone.

P4: Decrease number of classes recognized per extraction run: Reducing the number of classes per run allows the model to focus on one entity type extraction, increasing accuracy and completeness per class.

P5: Decrease number of properties annotated per extraction run: Narrowing the scope of property annotation in each run can similarly enhance extraction accuracy and completeness per property.

Ontology-level interventions focus on changes to the ontology itself that guide more accurate and complete extraction:

O1: Improve class descriptions: Clarifying and enhancing textual definitions and labels for classes in the ontology can improve entity recognition accuracy and decrease the number of false positives. Textual constraints in the descriptions and examples could be helpful in further improving accuracy.

O2: Improve property descriptions: Refining textual descriptions and labels of properties can facilitate accurate assignment and decrease the number of false positives. Here, textual constraints in the descriptions and examples could also be helpful in further improving accuracy.

O3: Refine datatype constraints: Specifying more precise datatype restrictions can improve literal and datatype extraction reliability.

O4: Refine class (NodeShape) constraints: Adjusting or introducing stricter constraints to better validate entity classes and improve extraction of their instances.

O5: Refine property (PropertyShape) constraints: Improving constraints for properties guides accurate validation and improves extraction.

Effects on various errors The various interventions are theorized to have different effects on different types of errors. Section 5.2.2 below summarizes these effects. The direction of the effects are hypothesised based on the mechanism of respective interventions. These directions are represented with arrows: (↑) indicates an increase, (↓) indicates a decrease, and (↔) indicates no substantial change or a neutral effect. Entries marked with multiple directions suggest that the effect could be different based on the specifics of the implementation.

Intervention	**TP**	**FP**	**TN**	**FN**
P1: More extraction runs	↑	↑	↔	↓
P2: More extraction evaluations	↑	↓	↔	↓
P3: Increased granularity	↑	↔	↔	↓
P4: Fewer classes per run	↑	↑	↓	↓
P5: Fewer properties per run	↑	↑	↓	↓
O1: Improve class descriptions	↑	↓	↑	↓
O2: Improve property descriptions	↑	↓	↑	↓
O3: Refine datatype constraints	↑	↓	↑	↑
O4: Refine class constraints	↔/↓	↓	↑	↔/↑
O5: Refine property constraints	↔/↓	↓	↑	↔/↑

Table 11: Impact of Pipeline and Ontology Interventions on Confusion Matrix Elements

More extraction runs (P1) Increasing the number of extraction runs per text typically increases true positives (TP) by reducing the probability of missing entities or relationships (i.e., it is more probable that one run will catch them). However, it may also increase false positives (FP), as repeated extraction runs raise the probability of incorrectly identifying entities or relationships. True negatives (TN) usually remain unaffected or are not significantly affected. Finally, false negatives (FN) decrease since multiple extraction runs enhance coverage.

More extraction evaluations (P2) Increasing the number of extraction evaluations typically leads to an increase in true positives (TP), as repeated confirmations reduce uncertainty and inconsistencies. Additionally, false positives (FP) tend to decrease because repeated verification helps filter out incorrect initial extractions and hallucinations. True negatives (TN) would remain unchanged, as they are presumed to be unaffected by confirming positives. False negatives (FN) typically decrease, as repeated verification reduces the chances of overlooked extractions. Thus, it makes sense to apply P2 together with P1 because P1 would lead to considerably more TPs and less FPs (but also more FPs), while P2 can balance the negative effect out.

Increase input granularity (P3) Increasing input text granularity by processing smaller/less text segments would improve TPs by enabling more precise identification of entities and relationships in a smaller text. The effect on FPs and TNs is not clearly defined. Additionally, FNs decrease as smaller input segments reduce the chance of overlooking relevant entities or properties.

Fewer classes and properties per extraction run (P4, P5) Decreasing the number of classes/properties recognized in each extraction run typically leads to an increase in TPs by enabling the model to focus on fewer entity types/properties simultaneously. Additionally, FPs increase because the model tends to over-eagerly assign entities/properties to scope even when they should not be (yet, this behaviour can be mitigated by P2). TNs generally decrease for the same reason. Finally, FNs decrease as the narrowed extraction focus helps ensure that relevant entities/properties belonging to the fewer targeted classes are less likely to be overlooked.

Improve class and property descriptions (O1, O2) Improving class/property descriptions by clarifying and enhancing textual definitions typically

leads to increased TPs, as clearer descriptions help the model correctly identify relevant entities. Additionally, FPs significantly decrease because precise descriptions reduce ambiguity and incorrect matches. TNs generally increase due to better differentiation, and FNs decrease since improved descriptions and examples reduce the probability of overlooking entities/annotations that should match clearly defined classes/properties.

Refine datatype constraints (O3) Refining/providing datatype constraints typically increases TPs by ensuring correct identification of literal values and datatypes, which sometimes could involve transformations into the correct datatype. FPs decrease as clearer constraints prevent incorrect datatype annotation, which could be closer to input text. TNs generally increase for the same reason. Still, FNs could increase since the datatype closer to the annotation in the text could not match the required datatype and get overlooked.

Refine class and property constraints (O4, O5) Refining class (NodeShape) and property (PropertyShape) constraints typically do not affect TPs just leading to corrections in annotations or might actually decrease the TPs due to strict constraints. FPs, on the other hand, decrease because constraints limit incorrect matches. TNs generally increases too for the same reason. However, FN might slightly increase if constraints become overly strict, causing ambiguously annotated instances to be erroneously excluded if they are not corrected.

5.2.3 Effects of Interventions

Assessment of effects of interventions allows a systematic evaluation and identification of areas for improvement, which can be achieved through (i) changes to the pipeline parameters, (ii) changes to the underlying ontology. Section 5.2.2 suggests possible interventions to address various insufficiencies. To calculate those, we refer to metric definitions in Table 7 and identify effects of various interventions (Section 5.2.2) on metric terms (inter-alia). Table 12 illustrates the direction of effect of interventions on various metrics.

Intervention	*Precision*	*Recall*	*F1-Score*	*Specificity*	*Accuracy*	*Fallout*	*Miss Rate*	*Jaccard Index*	*MCC*	*Informedness*	*Markedness*	*AUC*	*WRAcc*	χ^2
P1	↔	↑	↔	↓	↔	↑	↓	↔	↔	↔	↔	↔	↔	↔
P2	↑	↑	↑	↑	↑	↓	↓	↑	↑	↑	↑	↑	↑	↑
P3	↑	↑	↑	↔	↑	↔	↓	↑	↑	↑	↑	↑	↑	↑
P4	↔	↑	↑	↓	↔	↑	↓	↔	↔	↔	↔	↔	↔	↔
P5	↔	↑	↑	↓	↔	↑	↓	↔	↔	↔	↔	↔	↔	↔
O1	↑	↑	↑	↑	↑	↓	↓	↑	↑	↑	↑	↑	↑	↑
O2	↑	↑	↑	↑	↑	↓	↓	↑	↑	↑	↑	↑	↑	↑
O3	↑	↔	↔	↑	↔	↓	↔	↔	↔	↔	↔	↔	↔	↔
O4	↔	↔/↓	↔/↓	↑	↔	↓	↔/↑	↔/↓	↔	↔	↔	↔	↔	↔
O5	↔	↔/↓	↔/↓	↑	↔	↓	↔/↑	↔/↓	↔	↔	↔	↔	↔	↔

Table 12: Impact of Interventions (Inter-alia) across Evaluation Metrics

One clear learning from the table is that including more evaluation steps and increasing granularity improves performance nearly across the board. Thus, similarly to the hawk depicted in Figure 1, the extraction pipeline should telescopically zoom in on one specific feature at a time, capturing it with singular precision.

Furthermore, improving ontology descriptions tends to have clear positive impact on all metrics. This might suggest that ontologies in general would profit from more descriptive, in-depth and example-rich annotations that can be automatically used to support LLM-driven extraction from NLT, suggesting a potential extension to ontology engineering methodologies and their goals. This finding is in line with the requirement of detailed annotation guidelines (see Section 2.3.1) for labelling and quality assurance.

5.3 Inter-LLM Agreement

Evaluating inter-LLM agreement is crucial to determine the generalizability of extraction tasks performed by different LLMs. Studies have shown that LLMs differ in their political affinities [105]. Furthermore, given variations in architectures, training datasets (see Table 9) and procedures, identifying consistency of extraction across multiple LLMs could help (a) assess the quality of extraction instructions (incl. ontology definitions), (b) evaluate the degree to which the variation in extraction results is dependent on model

characteristics. High inter-LLM agreement between multiple models reinforces confidence in extracted data and suggests that extraction results are reliably capturing intended semantics.

To assess extraction reliability across different LLMs, we adapt traditional inter-coder agreement metrics to reflect the semantic structure inherent in LLM-generated outputs. Unlike standard annotations, ontology-driven LLM extractions involve graph structures, enabling agreement analysis at various granularity levels, including triples, entities, properties, and values. As discussed in Section 5.2, various levels of analysis have parallels to other extraction approaches. While triple-level analysis provides an overall assessment of extraction reliability, based on the task, an entity-level evaluation (e.g., for classic QC tasks) and combined property and value-level evaluations (e.g., for key-value extraction) could be more applicable. Here, we will focus on triple-level evaluation although the descriptions would be just as applicable for other levels.

Percent Agreement This metric (as defined in Section 2.3.3) provides a straightforward way of capturing consistency between LLM extraction outputs. When applying percent agreement to extraction pipelines producing graph outputs, consistency can be assessed by directly comparing the RDF graphs generated by extraction pipelines powered by different LLMs. Specifically, percent agreement would quantify the proportion of triples identically extracted by multiple models relative to the total number of unique triples extracted across those models. In practice, comparing such graphs is connected to a number of considerations: (a) most extracted entities are instantiated as blank nodes, which requires matching based on the structure and annotations, (b) whenever the models generate labels or extract evidence excerpts supporting annotations, those might deviate slightly and require adapting string matching approaches.

Graph Isomorphism (a) In the first case, despite there existing no known polynomial-time algorithm for deciding if two graphs are isomorphic, the associated theoretical worst-case scenarios are unlikely to be encountered in practice [50]. Heuristics involving labels and text evidence excerpts could practically mitigate such eventualities.

String Variations (b) More common is the problem of marginal string variations that, nevertheless, do not produce exact matches. Here, in the case of evidence strings, partial matches and more advanced matching approaches could be applied in practice when matching evidence is also in scope of eval-

uation. For label strings, variability could be considerable despite multiple versions being appropriate (e.g., "Jonathan Harker" vs. "Narrator"). Thus, for comparisons, such annotations should be ignored.

Percent agreement allows rapid identification of the extent to which different LLM pipelines converge or diverge in extracting triples, thereby offering a straightforward initial indicator of pipeline reliability.

Pairwise Model Agreement Adapting Cohen's kappa for LLM pairs involves assessing the degree of agreement beyond chance between two extraction pipelines powered by different LLMs. To calculate this adapted kappa, each potential triple is treated as a binary classification—either extracted or not extracted by the models. Cohen's kappa then quantifies the extent to which two LLMs independently agree on both the presence (positive agreement) and absence (negative agreement) of each triple, adjusting for chance-level agreement.

Limitations Regarding Negative Space However, for triple-level extraction, explicitly assessing negative space can be problematic due to the inherently unbounded nature of potential triples. Since the set of all theoretically possible annotation triples that could be formulated is infinite, it becomes meaningless to define the negative space (i.e., triples intentionally not extracted by the model). Nevertheless, at the entity-level and property-level evaluations, by framing these evaluations as classification tasks where the presence or absence of specific class instances or properties is explicitly assessed, negative space becomes defined, allowing meaningful application of the metric.

Multi-Model Agreement Using Krippendorff's alpha for multiple LLMs simultaneously. Krippendorff's alpha encounters a similar limitation as Cohen's kappa. However, when applied to bounded classification tasks such as entity-level or property-level extraction – Krippendorff's alpha can effectively measure the degree of consensus among multiple models. In these cases, the metric captures nuanced variations in extraction consistency across many models simultaneously.

Additionally, multi-model percent agreement can be, similarly to percent agreement, calculated by identifying the proportion of triples consistently extracted by all LLMs relative to the total unique triples extracted across the models. High inter-LLM agreement, particularly among models with

different architectures and training sets, would suggest robust extraction that is less likely to reflect model-specific biases.

Other Interpretations While in the previous discussion the variability stemmed from different LLM architectures, it could also be systematically emulated with LLMs through using different prompting strategies with the same LLM, including potentially inducing similar priming effects as are common among humans [140, 131]. Such a technique could provide more robust reliability assessments by intentionally introducing controlled variation in the extraction process, similar to how different human coders bring diverse interpretive frameworks to manual coding tasks.

5.4 Human-LLM Agreement

To evaluate agreement between coordinated human annotations and LLM extractions, we first aggregate annotations separately from human annotators and LLM-based extraction pipelines.

Consolidating Human Annotations For human annotators, a collective decision-making process [32] can be followed to arrive at a final, consolidated annotation dataset. During this process, discrepancies or differences in coding between annotators are resolved through discussions, consensus-building, or adjudication by experts, ensuring a unified and consistent reference annotation.

Consolidating Extracted Graphs For LLM-based extractions, we similarly aggregate the outputs into a final consolidated graph. Such consolidation can take several forms: an *intersection approach*, including only entities and annotations appearing in all individual extractions; a *union approach*, combining all entities and annotations extracted by any model; or a *majority approach*, integrating only entities and annotations extracted by the majority of models.

Interchangeably, an *LLM-as-a-judge* process can be utilized, where multiple LLMs compare extraction outputs, assess associated evidence text excerpts, and collectively decide on the most accurate and representative final version. This approach leverages a dedicated consolidation module within the pipeline, retrieving relevant paragraphs from the evidence excerpts provided by each LLM. The final judgment produced through this process results in a unified graph representing the consensus extraction from multiple LLMs.

Evaluation Approaches Once both human and LLM outputs have been aggregated into their respective final annotation graphs, human-LLM agreement can be evaluated using two distinct approaches: (1) treating human annotations as ground truth, and (2) assessing ICR between human and LLM annotations. The two methods differ fundamentally: the ground truth approach evaluates LLM annotations explicitly against a standard of correctness defined by human experts, whereas the inter-coder reliability approach considers human and LLM annotations as equally valid, measuring the degree of mutual semantic understanding without assuming correctness.

Ground Truth In the ground truth approach, the human-generated graph serves as the definitive reference against which the LLM-generated graph is compared. This evaluation directly measures the accuracy and completeness of LLM-based extractions by assessing their conformity to the human-defined standard. Such analysis emphasizes precision, recall, and other metrics traditionally associated with correctness relative to a known and trusted reference.

Inter-Coder Reliability Conversely, the ICR approach does not assume either annotation set as inherently correct. Instead, it treats human and LLM outputs as independent codings of the same underlying data, evaluating their mutual consistency. Metrics such as Cohen's kappa, Krippendorff's alpha, and percent agreement are employed, taking into consideration previously noted limitations regarding negative space. Specifically, assessing negative agreement at the triple level can be challenging due to the infinite set of potential triples, whereas bounded entity-level or property-level evaluations can more effectively leverage ICR metrics by explicitly defining the evaluation space.

Comparative Assessment To systematically compare the performance of human annotations and LLM-based extractions, both should first be independently evaluated against a known external ground truth dataset, validated as authoritative. This approach establishes an objective baseline for accuracy and completeness. Subsequently, human and LLM-generated graphs can be evaluated through inter-coder reliability metrics, treating each as independent annotators. High inter-coder agreement would suggest that the LLM extraction pipeline performs comparably to human annotators, effectively serving as an additional annotator. This dual-layer evaluation thus provides a comprehensive understanding of the relative strengths and consistency of human versus LLM-driven extraction processes.

6 Applications in Research

Academic events, including scientific conferences and workshops, are central venues to share research findings and discuss research problems, which has lead to significant growth over the years [82]. To find orientation in this growing number of conferences often is proven difficult, especially for early-stage researchers and researchers working on interdisciplinary research topics [33].

A variety of platform collects information of different nature about academic conferences in computer science domain: conference rankings (CORE Ranking portal, QUALIS Conference Ranking, ERA's ranking etc.), collections of proceedings and basic metadata (DBLP, OpenResearch.org, IEEE Xplore,Scopus, Web of Science, Wikidata etc.) and calls for papers (WikiCFP) [33]. Furthermore, there has been a number of ontologies (Scholarly Data, SEO, SemWeb Conference, SciGraph, Schema.org, EVENTSKG, AceKG ontology, SEDE, Comprehensive Call Ontology for Research etc.) proposed for scholarly metadata as well as Knowledge Graphs collecting structured data about conferences (ORKG, Wikidata, SWDF, OpenResearch etc.) [33, 82].

Despite these resources, in practice, scholarly data remains sparse and inaccessible. In this regard, publication of scholarly data as embedded LOD [33], large-scale collection of metadata and sustainability of making such collected data available remain major challenges [82]. At the same time, the improving capabilities of LLMs in understanding NLT provide an opportunity towards automatic collection of metadata from unstructured sources.

6.1 Proposed Approaches

Recently, two complementary approaches have been proposed for collecting in-depth details about academic calls-for-papers from conference websites [33] as well as extracting and making available in-depth conference-related knowledge from proceedings and websites [82].

Semantic Observer Semantic Observer [33] applies an implementation of the methodology described in this work to extracting information about calls-for-papers from conference and workshop websites. The paper establishes an ontology for describing calls-for-papers and uses the subset of top 20 conferences in the field of Data Management and Data Science from the CORE ranking for evaluation. The notable difference to to approach described in this work lies in the input text collection.

Input Text Collection The architecture is extended by a website crawler. From the initial website URL which serves as an entry point, the crawler a) creates a sitemap through traversing the links, b) extracts embedded metadata in different formats and c) retrieves HTML contents of website pages. Then, this collected data is used as text input to the pipeline.

Scholarly Wikidata A similar approach has been further developed and applied to a broader corpus of input data [82]. Here, the corpus of input data also included structured sources (Scholarly Data and DBLP Knowledge Graph) and conference proceedings. The scope of extracted metadata has been broadly defined by comparing related ontologies for scientific data publishing. A notable further component going beyond the methodology described in this work has been the HITL integration pipeline with Wikidata.

Entity Linking and Human Validation Firstly, the created ontology has been integrated with Wikidata by comparing existing properties and qualifiers and proposing missing terms. Secondly, OpenRefine has been used to link extracted entities with existing entities on Wikidata and create import statements with QuickStatements. In this way, the extracted data can be validated by a human and reconciled with data already available in a knowledge base.

Exploration and Visualization Another notable extension of the pipeline relates to the broader accessbility of the extracted knowledge. To this end, the data has been integrated into Scholia[12] and Synia[13] platforms enabling visualization and exploration. Scholia is a web application that allows users to explore conferences, researcher profiles, publications and other scholarly metadata. It is a web application build on top of Wikidata query service. On the other hand, Synia is a wiki-based platoforms where users can not only explore scholarly metadata, but also create own visualization templates.

6.2 Evaluation Results

The evaluation results from the Semantic Observer [33] and Scholarly Wikidata [82] provide initial evidence for the effectiveness of LLM-based extraction pipelines in real-world scholarly KE tasks. The LLM-supported extraction approach demonstrated strong performance across multiple evalua-

[12] https://scholia.toolforge.org/
[13] https://synia.toolforge.org/

tion dimensions, with performance varying by task type within the Scholarly Wikidata [82].

The most notable success was seen in organization roles extraction and the programme committee member extraction, which achieved near-perfect results with F1-Scores of 0.98-1.00 and 0.99-1.00 respectively. Submitted and accepted papers extraction also performed very reliably with F1-Scores ranging from 0.89-0.95. These results demonstrate that the framework described in this work enables structured information extraction with high reliability.

Looking more closely, when comparing different models, the evaluation revealed significant task-dependant performance variations. While the evaluated models performed similarly well on organization roles and PC member extraction, GPT-4 consistently outperformed Claude-3 when extracting important dates from less structured sources. This difference highlights how model selection can impact extraction quality for certain tasks, particularly those involving temporal information and less standardized formatting, and suggests promising directions for future work.

Furthermore, input source type emerged as a factor affecting extraction reliability. Highly structured content like proceedings front matter consistently yielded more reliable extractions compared to websites with varied layouts. Nevertheless, domain-specific terminology and academic jargon in the conference context were well-handled by all models, indicating that LLMs can effectively navigate specialized language when properly guided by qualitative domain ontologies.

Another aspect that the both approaches relied on was a harmonization exercise for broadly-used domain ontologies. By analysing the common structures and relationships across different scholarly ontologies, a unified meta-ontology could be constructed that preserves domain-specific concepts while establishing standardized cross-domain mappings. The extraction results from both Semantic Observer and Scholarly Wikidata highlight how domain ontologies can be effectively augmented with rich descriptive annotations optimized for LLM comprehension, suggesting that future ontology engineering methodologies should *explicitly* incorporate LLM-friendly descriptions alongside formal semantic definitions. This approach would create a *virtuous cycle* where improved ontologies lead to better extraction, which in turn could, in an inductive manner, provide insights for further ontology refinement.

The practical impact of these approaches extends beyond demonstrating the feasibility of LLM-supported KE systems in practice. Another contribution of the approach was the successful reconciliation and integration of over 6,000 entities with existing knowledge in Wikidata, which suggests that ontology-based extraction approaches also facilitate interoperability. The semi-automated approach significantly reduced the manual effort required to

collect and structure conference metadata while maintaining high accuracy, addressing one of the core challenges identified in the early chapters and indicating potential to broaden the potential impact to productivity gains in QC and other methodological areas relying on manual extraction. These results collectively confirm that ontology-guided LLM extraction can effectively bridge the gap between unstructured scholarly content and structured, machine-readable knowledge.

7 Conclusion

This work has established a theoretical framework for ontology-guided knowledge extraction with LLMs and demonstrated the viability the approach on practical examples - open issues in research ecosystem.

Limitations of Manual Extraction First, this work discussed the structural issues associated with manual extraction approaches. On the one hand, manual extraction studies are plagued by limited reliability which significantly contributes to replicability crisis in science in general. On the other hand, such studies, especially when targeting high reliaiblity, are highly resource-intensive in terms of time and skilled personallel, which is associated with significant pesonnel and administration/overhead costs. Existing tools, while providing useful administration tools do not provide any automation features for the extraction approach. Therefore, extraction remains heavily dependent on manual labor.

Formulation as KE Tasks Manual extraction approaches can be formulated as knowledge extraction tasks. The work goes into detail discussing different approaches to QC and KC, highlighting essential, common elements of the approaches. Then, a list of requirements is formulated as the basis of formulating the knowledge extraction task and designing the knowledge extraction pipeline. The requirements cover both deductive and inductive QC approaches as well as common IE tasks. They also cover diversity of output data to enable a multitude of downstream qualitative (Table 4) and quantitative analysis approaches (Table 5).

Ontological Grounding One of the main approaches of QC and KC is model building. Ontologies capture semantics and are prime candidates for a formalized representation of such models. One the one hand, they are based on versatile and practically established standards for data modelling (RDF), capturing semantics and enabling inference (OWL), graph-based querying

(SPARQL) and enabling structural validation (SHACL), which power Knowledge Graphs.

On the other hand, such standards also enable seamless data integration and publishing as Linked Open Data and part of community-curated Open Knowledge Graphs. Capturing ontologies using these standards improves data availability, accessibility, integration, interoperability and re-use, enabling smart interoperable applications and versatile implementation in a multitude of other systems, including downstream applications tasked with analysis and in applied use cases profiting from the extracted data annotated using the ontology.

We have shown that qualitative descriptions as part of ontologies have a potential to universally improve metrics associated with extraction reliability, thus pointing at the need of re-formulation and extension of ontology engineering methodologies by LLM-friendly class and property descriptions, validation constraints as well as other useful annotations.

Extraction Pipeline Based on the requirements, we design a RAG pipeline for ontology-guided extraction utilizing LLMs. The pipeline consists of re-usable abstract modules implementing extraction, evaluation and consolidation for entity recognition and annotation. It follows the Divide-and-conquer problem-solving approach which is also underpinned by the finding that increased granularity of input and lower number of classes/properties to be extracted at each run. The modules can also be used in other LLM-supported tasks and make the system easy-to-extend. An implementation of the pipeline as Python library is made publicly available as a re-usable resource.

Evaluation Metrics and Datasets We have discussed evaluation approaches commonly used in QC and IE for testing reliability of the pipeline. here, we defined crucial metrics and discussed notable limitations of applying these metrics as well as mitigation approaches for those limitations. Since the pipeline produces higher dimensional annotations (graph structures) as output, we discussed different levels of evaluation: triple, entity, property and value-based evaluation and established parallels between levels and reliability studies conducted in QC and IE.

Performance-improving Interventions Besides evaluation levels, we have discussed possible pipeline-level and ontology-level interventions that could

improve extraction performance. On the pipeline level, such interventions relate to abstract modules and their parameters. Here, interventions increasing granularity, atomicity and validation of extraction lead to improved performance while other interventions (e.g., parallelization) have trade-offs and have to be used in combinations that balance opposite effects. On the ontology level, interventions target the quality of ontology descriptions. Here, improved descriptions have a clear positive impact on extraction performance while stringent constraints are great for eliminating FPs but should be used in combination with other interventions to offset their negative effects such as occasional filtering-out of potential TPs and the associated increase in FNs.

Inter-coder Reliability Furthermore, we have discussed common inter-coder reliability metrics and their limitations of evaluating chance annotation connected to unbounded negative space and mitigation approaches in context of LLM-supported and ontology-guided knowledge extraction. To apply metrics to LLM-based knowledge extraction contexts, we introduced inter-LLM and coder-LLM reliability as metrics that can be used to assess consistency of extraction between LLMs and compared to human coders. We also discussed minimal requirements for the LLM-supported extraction pipeline to be considered an additional annotator and evaluation strategy for assessing improved reliability compared to manual extraction.

Limited Availability of Knowledge in Research We discussed the limited availability and accessibility of scientific knowledge in research ecosystem. In particular, we showed the limited availability of knowledge relating to academic conferences. Here, despite a number of ontologies available for knowledge annotation, the reliance on manual annotation makes the approach infeasible in practice. Thus, automated knowledge extraction is a promising approach to capturing rich knwoledge associated with academic venues in research ecosystem.

Automatic Knowledge Extraction in Research Then, we demonstrated that ontology-driven knowledge extraction utilizing LLMs is a viable and reliable approach for such tasks. Applying the pipeline, two approaches could reliably extract conference knowledge from website text and conference proceedings, including calls-for-papers, programme and related academic events (e.g., workshops, challenges, invited talks etc.), various types of participants as well as more complex annotations such as acceptance rates.

Scalability Considerations Finally, we discussed extensions to the extraction pipeline that make it universally applicable for knowledge extraction at scale. On the one hand, web crawling and automatic platform discovery utilizing multiple search engines enable automatic discovery of relevant knowledge on the web as well as implement some aspects of agentic behaviour. In a sense, such functionality enables "intelligent agents" as proposed in the initial vision of Semantic Web [9]. On the other hand, we discuss and approach to integrating the extraction results with community-curated large Knowledge Graphs. This approach implements disambiguation and final validation though a tool-supported HITL process, although a fully automated implementation is also feasible if sufficiently high reliability can be demonstrated. Integrating the results with Open Knowledge Graphs ensures that the extracted knowledge is broadly accessible and re-usable, improving the value of extracted knowledge as an interoperable and highly available resource.

Together, this work has established a framework for the automation of manual extraction tasks and evaluation of its reliability. Furthermore, it has demonstrated reliability of the approach on a practical example of extracting knowledge in the research domain.

8 Discussion

This work has established a framework for ontology-guided KE with LLMs, bridging the gap between manual and automated extraction methods. It also provided reliability targets that have to be achieved if the extraction pipeline is to be applied as a superior extraction method to average human coders.

The framework addresses the significant challenges faced by traditional manual extraction approaches while leveraging the emergent capabilities of LLMs for understanding and processing NLT. By formulating manual extraction methodologies as knowledge extraction tasks and grounding them in ontological frameworks, we have created a systematic approach that aims to improve on the methodological rigour of traditional methods while reducing their resource intensity.

Theoretical Integration of Manual and Automated Approaches Our framework represents a unique integration of previously distinct methodologies. By analysing the commonalities between QC, KC, IE, and KE, we have demonstrated that these approaches, despite emerging from different disciplines, share fundamental extraction processes. This integration enables

practitioners from different domains to leverage extraction tools and insights across disciplinary boundaries.

The formulation of manual extraction methodologies as knowledge extraction tasks provides several key advantages. First, it enables a clear specification of the requirements for automating these processes. Second, it allows for the application of established evaluation metrics from IE and KE to assess the reliability of the automated extraction. Third, it facilitates the standardization (and re-use) of extraction models through ontologies and integration of the extracted knowledge.

Ontological grounding has proven particularly valuable in this context. By encoding models as ontologies using Semantic Web standards, we ensure that the extracted knowledge is not only structured but also semantically meaningful, interoperable, and reusable. This approach aligns with the original vision of the Semantic Web [9] as providing a framework for knowledge representation and sharing across systems and applications.

Addressing the Replication Crisis through Automated Extraction One of the most significant implications of our work relates to the replication crisis in scientific research. As discussed in the introduction, manual extraction methods face substantial challenges regarding reliability and replicability, contributing to broader issues in scientific reproducibility. Our ontology-guided extraction approach directly addresses these challenges in several ways:

1. By formalizing extraction categories and relationships in ontologies, defining explicit constraints and integrating qualitative, extraction-facilitating descriptions in ontologies, we create explicit, shareable models that allow for consistent application across studies.

2. The systematic extraction by LLMs reduces subjective interpretations that can vary among human coders. When a number of LLMs with different architectures and training datasets are applied, this can achieve higher objectivity of results. Furthermore, once validated, the extraction pipeline can be applied consistently across similar datasets, ensuring methodological consistency. Now, the replication can be performed automatically and is guided by concrete parameters: the underlying ontology, versions and hyperparameters of applied LLMs, and configuration parameters of the pipeline.

3. The automatic and systematic linking of extracted knowledge to textual evidence and provenance information provides transparent justification

for annotations, enabling easier verification. The justifications can be published together with results.

Integrating multiple LLMs in extraction tasks and evaluation of extraction reliability provide additional mechanism for improving the objectivity and replicability of the results.

Resource Efficiency and Scalability A primary motivation for this work was addressing the resource intensity of manual extraction approaches. Our evaluations suggest that the automated approach can significantly reduce the time and personnel resources required for knowledge extraction while maintaining acceptable reliability. This efficiency gain is particularly valuable in research contexts where resources are limited or where large-scale extraction is necessary. The scalability of our approach is demonstrated in the research ecosystem applications, where we successfully applied the framework to extract knowledge from conference websites, proceedings, and other scholarly sources. Such applications would be prohibitively resource-intensive using purely manual approaches, yet they provide valuable structured knowledge for the research community.

However, it is important to note that the resource efficiency of our approach does not (completely) eliminate the need for human involvement. Rather, it shifts the focus of human effort from exhaustive manual coding to more strategic tasks such as ontology design and interpretation of the extracted knowledge. In a way, the effects of changes to the model can be instantly transformed into extraction results and compared, paving the way of iteratively improving of models – extracted entities could be further categorized or broader categories could be extracted for scouting, in the same way, attributes with flexible definitions could be applied to explore data and, then, stringent definitions, datatypes and constraints can be defined for precise extraction. This human-in-the-loop approach ensures that domain expertise continue to guide the extraction process while reducing the repetitive aspects of manual coding, which is relegated to machines.

While public discourse often positions GenAI as primarily threatening to replace creative work, our findings demonstrate its effectiveness in automating structured manual tasks. This approach redirects human expertise toward strategic guidance rather than repetitive extraction, creating a complementary relationship that could challenge prevailing narratives about AI's impact on research methodologies.

Further Applications The framework developed in this work has significant potential across diverse domains:

- In SLRs and systematic mapping studies, the codebook and feature catalogue could be described with an ontology and then data could be automatically extracted and updated whenever new research is published. Ontologies could capture various aspects of the papers, including methodologies and the actual contents of papers.

- In financial domain, key data from reports, company websites and news (including social media posts) could be automatically extracted and then used to automate investment and risk management decisions.

- In education, knowledge from educational materials can be automatically extracted: topics, explanations, multimodal examples, exercises could be extracted, enabling automatic tutoring systems.

- In healthcare, knowledge from studies could be automatically extracted to enable semantic search and application of graph-based methods, potentially uncovering connections previously not explored. Crucial data from anamneses, subjective and objective clinical descriptions, clinical reports could be extracted to enable automatic inference, illuminating all potential connections, enabling diagnosis and treatment suggestions.

- In the work domain, automatically generated transcripts of meetings could be semantically annotated, allowing rule-based conclusions and summaries. Furthermore, knowledge from CVs and job descriptions could be automatically extracted and enable automatic matching.

- In public governance, crucial details from applicants' documents could be extracted, enabling automatic public governance services. In this context, applicant data could be securely stored in Personal Knowledge Graphs and automatically matched with bureaucratic requirements. Here, but also in other potential critical applications, HITL approaches are especially called for to ensure humanity.

- In legal studies, structured data could be extracted from (proposed) laws and regulations, enabling the identification of potential conflicts, inconsistencies, and gaps within legal frameworks. The approach proposed in this work could potentially serve as the foundation for ontology-based digital law, where legal concepts and their relationships are formally defined in ontologies, facilitating automated reasoning, compliance checking, and the development of more sophisticated legal information systems and innovative legal workflows.

- In politics, public communications from political figures could be annotated, enabling the creation of an overview of political positions and more informed decision-making on the part of citizens, potentially improving the functioning of democratic systems.

All these applications would necessitate further developments. Nonetheless, reliably, ontology-guided knowledge extraction remains the crucial component enabling those applications. Most importantly, the approach enables explainable, data-driven downstream applications, for which the quality of the underlying knowledge is paramount.

Broader Considerations The methodology established in this work potentially has transformative potential beyond individual studies. A centralized platform for qualitative research extraction could enable researchers to publish ontologies, extraction parameters, and reliability metrics alongside their studies, fostering transparency and reusability, similarly to how studies are pre-registered in biomedical domain.

Furthermore, LLMs could dynamically enhance ontologies based on extraction patterns and domain-specific corpora, creating a feedback loop of continuous improvement. This could eventually lead to automatically generated meta-ontologies optimized for maximum knowledge coverage while respecting pre-defined constraints on depth and complexity.

8.1 Limitations

Despite the promising results demonstrated by the framework, several important limitations must be acknowledged that affect both the theoretical foundations and practical implementation of the approach. Here, challenges that are actively mitigated by the pipeline and evaluation design (e.g., hallucinations) are not listed repeatedly.

Systematic Bias While using multiple LLMs can mitigate some biases, systematic biases remain embedded in the training data and architectures of these models. These biases may influence extraction results in subtle ways, particularly when extracting subjective information or working with contested knowledge domains. The framework's reliability assessments can identify consistency of annotation between models but are not targeting shared biases that persist across LLMs. Still, such biases could be evaluated in comparison to established ground truth datasets.

Ontological Consistency Ontologies and their resulting KGs are often not logically consistent [31], particularly when developed iteratively or when capturing complex (or many) domains. This inconsistency can lead to extraction ambiguities and validation challenges. While logical consistency is not one of the goals of extraction, the issue is well-known in Semantic Web community, although it has been given up with the advent of Knowledge Graphs, which often rely on the Open World Assumption. Grounding higher ontologies in foundational ontologies could improve consistency but is associated with increased complexity of modelling. This issue is further exacerbated by the evolution of Knowledge Graphs and ontologies [101, 95].

Institutional Adoption Barriers Significant institutional limitations exist in the adoption of automated extraction approaches.[14] Research communities have established practices and methodologies that researchers are accustomed to, and transitioning to new approaches requires not only technological solutions but also cultural and procedural changes within research institutions and publishing venues. Nonetheless, practical considerations, ease-of-use and good methodological foundation are aspects that could facilitate adoption.

8.2 Future Work

Several promising directions for future research emerge from this work, addressing both technical enhancements and methodological extensions to the ontology-guided knowledge extraction framework.

Ground Truths One avenue of future work is establishing qualitative ground truth datasets for KE. One notable domain for such exercise is scientific articles and research in general. Here, nanopublications [66] are a promising approach to representing extracted LD. However, nanopublications do not prescribe specific ontologies and despite a number of ontologies and schemas being proposed, the task of modelling the different increasingly complex aspects of the knowledge contained in scientific papers (e.g., methodologies, argumentation lines, findings) is increasingly urgent and relevant for advancing the field. Creating qualitative ground truth datasets would allow benchmarking of extraction approaches as a necessary step to proving the reliability of extraction.

[14] This limitation has been repeatedly identified in informal conversations with colleagues from social sciences, and became a key topic of discussion at the AI for the Research Ecosystem workshop 2024 in London (AI4RE, see https://ai4re.github.io/).

Ontology Learning Another promising direction is integrating OL methods with extraction pipelines. While this work relies on pre-defined ontologies to guide extraction, future research could integrate workflows where LLM-based extraction results inform ontology evolution directly, expanding the potential for theory-building. This would create an adaptive system where ontologies evolve based on the extraction results, and extraction is then improved through a refined ontology. Furthermore, LLMs themselves could be leveraged to propose ontology extensions based on patterns identified during extraction, with constraints both validating the extraction, and being dynamically defined and evaluated based on extraction results.

Linguistic Structure Utilization The proposed approach treats NLT primarily as an unstructured input source. However, natural language has syntactic patterns that could be exploited to enhance extraction of semantic annotations. Future work could integrate linguistic dependency and bottom-up semantic parsing to approach semantic annotation from the other direction. By mapping foundational ontologies and higher ontological relations to common syntactic structures (e.g., noun and verb phrases [21] as well as more complex reification), we could extract explicit semantic statements from text. Furthermore, co-reference resolution and other disambiguation approaches with the help of LLMs [107] could be incorporated.

Fine-tuning LLMs for Extraction While the current framework effectively utilizes pre-trained LLMs through prompting and RAG techniques, fine-tuning models specifically for structured extraction tasks represents a significant opportunity for improvement. For one, fine-tuning LLMs on domain-specific structured data could enhance their domain understanding and performance. While pre-trained general-purpose LLMs have demonstrated impressive capabilities in KE, architectural modifications specifically designed for structured information extraction could improve performance: specialized decoder heads optimized for graph-structured outputs, attention mechanisms working with graph structures or hybrid architectures that combine transformer models with graph neural networks.

References

[1] Hasan Abu-Rasheed, Mohamad Hussam Abdulsalam, Christian Weber, and Madjid Fathi. Supporting Student Decisions on Learning Recommendations: An LLM-Based Chatbot with Knowledge Graph Con-

textualization for Conversational Explainability and Mentoring. 2024. Publisher: arXiv Version Number: 3.

[2] Garima Agrawal, Tharindu Kumarage, Zeyad Alghami, and Huan Liu. Can Knowledge Graphs Reduce Hallucinations in LLMs? : A Survey. 2023. Publisher: arXiv Version Number: 1.

[3] Monica Agrawal, Stefan Hegselmann, Hunter Lang, Yoon Kim, and David Sontag. Large language models are few-shot clinical information extractors. In Yoav Goldberg, Zornitsa Kozareva, and Yue Zhang, editors, *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 1998–2022, Abu Dhabi, United Arab Emirates, December 2022. Association for Computational Linguistics.

[4] R. Alharbi, U. Ahmed, D. Dobriy, W. Łajewska, L. Menotti, M. J. Saeedizade, and M. Dumontier. Exploring the role of generative ai in constructing knowledge graphs for drug indications with medical context. In *SWAT4HCLS'24: The 15th International SWAT4HCLS Conference*, Leiden, The Netherlands, February 2024.

[5] Andrew Armitage and Diane Keeble-Ramsay. The rapid structured literature review as a research strategy. *US-China Education Review*, 6(4):27–38, 2009.

[6] Jinheon Baek, Alham Fikri Aji, and Amir Saffari. Knowledge-Augmented Language Model Prompting for Zero-Shot Knowledge Graph Question Answering, June 2023. arXiv:2306.04136 [cs].

[7] Monya Baker. 1,500 scientists lift the lid on reproducibility. *Nature*, 533(7604):452–454, May 2016.

[8] Melissa Beresford, Amber Wutich, Margaret V. Du Bray, Alissa Ruth, Rhian Stotts, Cindi SturtzSreetharan, and Alexandra Brewis. Coding Qualitative Data at Scale: Guidance for Large Coder Teams Based on 18 Studies. *International Journal of Qualitative Methods*, 21:16094069221075860, April 2022.

[9] Tim Berners-Lee, James Hendler, and Ora Lassila. The semantic web: A new form of web content that is meaningful to computers will unleash a revolution of new possibilities. In *Linking the World's Information: Essays on Tim Berners-Lee's Invention of the World Wide Web*, pages 91–103. 2023.

[10] Christian Bizer, Tom Heath, and Tim Berners-Lee. Linked data: Principles and state of the art. In *World wide web conference*, volume 1, page 40. Citeseer, 2008.

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

[12] Anna Breit, Laura Waltersdorfer, Fajar J Ekaputra, Marta Sabou, Andreas Ekelhart, Andreea Iana, Heiko Paulheim, Jan Portisch, Artem Revenko, Annette ten Teije, et al. Combining machine learning and semantic web: A systematic mapping study. *ACM Computing Surveys*, 55(14s):1–41, 2023.

[13] Dan Brickley, Ramanathan V Guha, and Brian McBride. Rdf schema 1.1. w3c recommendation. *World Wide Web Consortium*, 2, 2014.

[14] Colin F. Camerer, Anna Dreber, Felix Holzmeister, Teck-Hua Ho, Jürgen Huber, Magnus Johannesson, Michael Kirchler, Gideon Nave, Brian A. Nosek, Thomas Pfeiffer, Adam Altmejd, Nick Buttrick, Taizan Chan, Yiling Chen, Eskil Forsell, Anup Gampa, Emma Heikensten, Lily Hummer, Taisuke Imai, Siri Isaksson, Dylan Manfredi, Julia Rose, Eric-Jan Wagenmakers, and Hang Wu. Evaluating the replicability of social science experiments in Nature and Science between 2010 and 2015. *Nature Human Behaviour*, 2(9):637–644, August 2018.

[15] Salvatore Carta, Alessandro Giuliani, Leonardo Piano, Alessandro Sebastian Podda, Livio Pompianu, and Sandro Gabriele Tiddia. Iterative Zero-Shot LLM Prompting for Knowledge Graph Construction. 2023. Publisher: arXiv Version Number: 1.

[16] Emanuele Cavalleri, Mauricio Soto-Gomez, Ali Pashaeibarough, Dario Malchiodi, Harry Caufield, Justin Reese, Christopher J Mungall, Peter N Robinson, Elena Casiraghi, Giorgio Valentini, et al. Spirex: Improving llm-based relation extraction from rna-focused scientific literature using graph machine learning. *Proceedings of the VLDB Endowment. ISSN*, 2150:8097, 2024.

[17] Pranav Chellagurki, Sai Prasanna Kumar Kumaru, Rahul Raghava Peela, Neeharika Yeluri, Carlos Rojas, and Jorjeta Jetcheva. Biomedical relation extraction using llms and knowledge graphs. In *2024 IEEE 10th International Conference on Big Data Computing Service and*

Machine Learning Applications (BigDataService), pages 60–69. IEEE, 2024.

[18] Jiao Chen, Luyi Ma, Xiaohan Li, Nikhil Thakurdesai, Jianpeng Xu, Jason H. D. Cho, Kaushiki Nag, Evren Korpeoglu, Sushant Kumar, and Kannan Achan. Knowledge Graph Completion Models are Few-shot Learners: An Empirical Study of Relation Labeling in E-commerce with LLMs. 2023. Publisher: arXiv Version Number: 1.

[19] Kason Ka Ching Cheung and Kevin W. H. Tai. The use of intercoder reliability in qualitative interview data analysis in science education. *Research in Science & Technological Education*, 41(3):1155–1175, July 2023.

[20] Kason Ka Ching Cheung and Kevin WH Tai. The use of intercoder reliability in qualitative interview data analysis in science education. *Research in Science & Technological Education*, 41(3):1155–1175, 2023.

[21] Noam Chomsky. *The minimalist program*. MIT Press, 2014.

[22] Olivier Cinquin. Chip-gpt: a managed large language model for robust data extraction from biomedical database records. *Briefings in bioinformatics*, 25(2):bbad535, 2024.

[23] Samuel O. Clarke, Wendy C. Coates, and Jaime Jordan. A practical guide for conducting qualitative research in medical education: Part 3—Using software for qualitative analysis. *AEM Education and Training*, 5(4):e10644, October 2021.

[24] P Cohendet. The codification of knowledge: a conceptual and empirical exploration. *Industrial and Corporate Change*, 9(2):195–209, June 2000.

[25] Debby RE Cotton, Peter A Cotton, and J Reuben Shipway. Chatting and cheating: Ensuring academic integrity in the era of chatgpt. *Innovations in education and teaching international*, 61(2):228–239, 2024.

[26] Robin Cowan, Paul A David, and Dominique Foray. The explicit economics of knowledge codification and tacitness. *Industrial and corporate change*, 9(2):211–253, 2000.

[27] Jim Cowie and Wendy Lehnert. Information extraction. *Communications of the ACM*, 39(1):80–91, 1996.

[28] John Dagdelen, Alexander Dunn, Sanghoon Lee, Nicholas Walker, Andrew S Rosen, Gerbrand Ceder, Kristin A Persson, and Anubhav Jain. Structured information extraction from scientific text with large language models. *Nature Communications*, 15(1):1418, 2024.

[29] Paul David, Dominique Foray, and W Edward Steinmueller. 13 the research network and the new economics of science: from metaphors to organizational behaviors. *The organization of economic innovation in Europe*, page 303, 1999.

[30] Paul A David and Dominique Foray. Accessing and expanding the science and technology knowledge base. *STI review*, (16), 1995.

[31] Thomas de Groot, Joe Raad, and Stefan Schlobach. Analysing large inconsistent knowledge graphs using anti-patterns. In *The Semantic Web: 18th International Conference, ESWC 2021, Virtual Event, June 6–10, 2021, Proceedings 18*, pages 40–56. Springer, 2021.

[32] Jessica Díaz, Jorge Pérez, Carolina Gallardo, and Ángel González-Prieto. Applying inter-rater reliability and agreement in collaborative grounded theory studies in software engineering. *Journal of Systems and Software*, 195:111520, 2023.

[33] D. Dobriy. Employing rag to create a conference knowledge graph from text. In *ESWC'2024: The 21st Extended Semantic Web Conference*, Hersonissos, Greece, May 2024.

[34] D. Dobriy, M. Beno, and A. Polleres. Smw cloud: A corpus of domain-specific knowledge graphs from semantic mediawikis. In A. Meroño Peñuela et al., editors, *The Semantic Web. ESWC 2024*, volume 14665 of *Lecture Notes in Computer Science*. Springer, Cham, 2024.

[35] D. Dobriy and A. Polleres. Crawley: Ein tool zur entdeckung von webplattformen. In I. Fundulaki, K. Kouji, D. Garijo, and J. M. Gomez-Perez, editors, *Proceedings der 22. International Semantic Web Conference: Poster, Demos und Branchen-Track*, CEUR Workshop Proceedings, Athen, Griechenland, November 2023.

[36] Anca Dumitrache, Oana Inel, Benjamin Timmermans, Carlos Ortiz, Robert-Jan Sips, Lora Aroyo, and Chris Welty. Empirical methodology for crowdsourcing ground truth. *Semantic Web*, 12(3):403–421, 2021.

[37] Yuchen Fan, Yantao Liu, Zijun Yao, Jifan Yu, Lei Hou, and Juanzi Li. Evaluating generative language models in information extraction as subjective question correction. *arXiv preprint arXiv:2404.03532*, 2024.

[38] Yuchen Fan, Yantao Liu, Zijun Yao, Jifan Yu, Lei Hou, and Juanzi Li. Evaluating generative language models in information extraction as subjective question correction. In Nicoletta Calzolari, Min-Yen Kan, Veronique Hoste, Alessandro Lenci, Sakriani Sakti, and Nianwen Xue, editors, *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 6409–6417, Torino, Italia, May 2024. ELRA and ICCL.

[39] Nicolas Ferranti, Jairo Francisco De Souza, Shqiponja Ahmetaj, and Axel Polleres. Formalizing and validating wikidata's property constraints using shacl and sparql. *Semantic Web*, 15(6):2333–2380, 2024.

[40] Michael Freund, Rene Dorsch, Sebastian Schmid, Thomas Wehr, and Andreas Harth. Enriching rdf data with llm based named entity recognition and linking on embedded natural language annotations. In *International Knowledge Graph and Semantic Web Conference*, pages 109–122. Springer, 2024.

[41] Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yi Dai, Jiawei Sun, Haofen Wang, and Haofen Wang. Retrieval-augmented generation for large language models: A survey. *arXiv preprint arXiv:2312.10997*, 2, 2023.

[42] Gabriel Lino Garcia, Joao Renato Ribeiro Manesco, Pedro Henrique Paiola, Lucas Miranda, Maria Paola de Salvo, and Joao Paulo Papa. A review on scientific knowledge extraction using large language models in biomedical sciences. *arXiv preprint arXiv:2412.03531*, 2024.

[43] Barney Glaser and Anselm Strauss. *Discovery of grounded theory: Strategies for qualitative research*. Routledge, 2017.

[44] Arianna Graciotti. Knowledge extraction from multilingual and historical texts for advanced question answering. In *DC@ISWC*, 2023.

[45] Xinyan Guan, Yanjiang Liu, Hongyu Lin, Yaojie Lu, Ben He, Xianpei Han, and Le Sun. Mitigating Large Language Model Hallucinations via Autonomous Knowledge Graph-based Retrofitting. 2023. Publisher: arXiv Version Number: 1.

[46] Lars Håkanson. Creating knowledge: the power and logic of articulation. *Industrial and Corporate Change*, 16(1):51–88, 2007.

[47] Ridong Han, Chaohao Yang, Tao Peng, Prayag Tiwari, Xiang Wan, Lu Liu, and Benyou Wang. An empirical study on information extraction using large language models. *arXiv preprint arXiv:2409.00369*, 2024.

[48] Ali Hasnain and Dietrich Rebholz-Schuhmann. Assessing fair data principles against the 5-star open data principles. In *The Semantic Web: ESWC 2018 Satellite Events: ESWC 2018 Satellite Events, Heraklion, Crete, Greece, June 3-7, 2018, Revised Selected Papers 15*, pages 469–477. Springer, 2018.

[49] Daniel Hernández, Aidan Hogan, and Markus Krötzsch. Reifying rdf: What works well with wikidata? *SSWS@ ISWC*, 1457:32–47, 2015.

[50] Aidan Hogan. Skolemising blank nodes while preserving isomorphism. In *Proceedings of the 24th International Conference on World Wide Web*, pages 430–440, 2015.

[51] Aidan Hogan, Eva Blomqvist, Michael Cochez, Claudia d'Amato, Gerard De Melo, Claudio Gutierrez, Sabrina Kirrane, José Emilio Labra Gayo, Roberto Navigli, Sebastian Neumaier, et al. Knowledge graphs. *ACM Computing Surveys (Csur)*, 54(4):1–37, 2021.

[52] Hsiu-Fang Hsieh and Sarah E Shannon. Three approaches to qualitative content analysis. *Qualitative health research*, 15(9):1277–1288, 2005.

[53] Jin Huang, Xingjian Zhang, Qiaozhu Mei, and Jiaqi Ma. Can LLMs Effectively Leverage Graph Structural Information: When and Why. 2023. Publisher: arXiv Version Number: 2.

[54] Yuxuan Huang, Lida Shi, Anqi Liu, and Hao Xu. Evaluating and Enhancing Large Language Models for Conversational Reasoning on Knowledge Graphs. 2023. Publisher: arXiv Version Number: 2.

[55] Zhang Jiawei. Graph-ToolFormer: To Empower LLMs with Graph Reasoning Ability via Prompt Augmented by ChatGPT. April 2023.

[56] Björn Johnson, Edward Lorenz, and Bengt-Åke Lundvall. Why all this fuss about codified and tacit knowledge? *Industrial and corporate change*, 11(2):245–262, 2002.

[57] Philip N. Johnson-Laird and Marco Ragni. What Should Replace the Turing Test? *Intelligent Computing*, 2:0064, January 2023.

[58] Cameron R. Jones and Benjamin K. Bergen. Does GPT-4 pass the Turing test?, April 2024. arXiv:2310.20216 [cs].

[59] Ahlem Chérifa Khadir, Hassina Aliane, and Ahmed Guessoum. Ontology learning: Grand tour and challenges. *Computer Science Review*, 39:100339, 2021.

[60] Edward Kim, Manil Shrestha, Richard Foty, Tom DeLay, and Vicki Seyfert-Margolis. Structured extraction of real world medical knowledge using llms for summarization and search. In *2024 IEEE International Conference on Big Data (BigData)*, pages 3421–3430. IEEE, 2024.

[61] Jiho Kim, Yeonsu Kwon, Yohan Jo, and Edward Choi. KG-GPT: A General Framework for Reasoning on Knowledge Graphs Using Large Language Models. 2023. Publisher: arXiv Version Number: 1.

[62] Barbara Kitchenham, Stuart Charters, et al. Guidelines for performing systematic literature reviews in software engineering version 2.3. *Engineering*, 45(4ve):1051, 2007.

[63] Holger Knublauch. Shacl and owl compared. *TopQuadrant*, August 2017. Available online at https://spinrdf.org/shacl-and-owl.html.

[64] Vojtech Kovár, Miloš Jakubıcek, and Aleš Horák. On evaluation of natural language processing tasks. In *Proceedings of the 8th International Conference on Agents and Artificial Intelligence*, pages 540–545, 2016.

[65] Klaus Krippendorff. *Content Analysis: An Introduction to Its Methodology*. SAGE Publications, Thousand Oaks, California, 4 edition, 2018.

[66] Tobias Kuhn, Albert Meroño-Peñuela, Alexander Malic, Jorrit H Poelen, Allen H Hurlbert, Emilio Centeno Ortiz, Laura I Furlong, Núria Queralt-Rosinach, Christine Chichester, Juan M Banda, et al. Nanopublications: a growing resource of provenance-centric scientific linked data. In *2018 IEEE 14th International Conference on e-Science (e-Science)*, pages 83–92. IEEE, 2018.

[67] Jens Lehmann, Preetam Gattogi, Dhananjay Bhandiwad, Sébastien Ferré, and Sahar Vahdati. Language Models as Controlled Natural

Language Semantic Parsers for Knowledge Graph Question Answering. In Kobi Gal, Ann Nowé, Grzegorz J. Nalepa, Roy Fairstein, and Roxana Rădulescu, editors, *Frontiers in Artificial Intelligence and Applications*. IOS Press, September 2023.

[68] Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in neural information processing systems*, 33:9459–9474, 2020.

[69] Xinze Li, Yixin Cao2, Liangming Pan, Yubo Ma, and Aixin Sun. Towards Verifiable Generation: A Benchmark for Knowledge-aware Language Model Attribution. 2023. Publisher: arXiv Version Number: 1.

[70] Yihao Li, Ru Zhang, Jianyi Liu, and Gongshen Liu. An Enhanced Prompt-Based LLM Reasoning Scheme via Knowledge Graph-Integrated Collaboration. 2024. Publisher: arXiv Version Number: 1.

[71] Jiayi Liu, Honghao Lai, Weilong Zhao, Jiajie Huang, Danni Xia, Hui Liu, Xufei Luo, Bingyi Wang, Bei Pan, Liangying Hou, et al. Ai-driven evidence synthesis: data extraction of randomized controlled trials with large language models. *International Journal of Surgery*, pages 10–1097.

[72] Linhao Luo, Yuan-Fang Li, Gholamreza Haffari, and Shirui Pan. Reasoning on Graphs: Faithful and Interpretable Large Language Model Reasoning. 2023. Publisher: arXiv Version Number: 1.

[73] Alejandro Mallea, Marcelo Arenas, Aidan Hogan, and Axel Polleres. On blank nodes. In *International semantic web conference*, pages 421–437. Springer, 2011.

[74] Mohammed Maree. Quantifying relational exploration in cultural heritage knowledge graphs with llms: A neuro-symbolic approach. *arXiv preprint arXiv:2501.06628*, 2025.

[75] Jose L Martinez-Rodriguez, Aidan Hogan, and Ivan Lopez-Arevalo. Information extraction meets the semantic web: a survey. *Semantic Web*, 11(2):255–335, 2020.

[76] Ariana Martino, Michael Iannelli, and Coleen Truong. Knowledge Injection to Counter Large Language Model (LLM) Hallucination. In Catia Pesquita, Hala Skaf-Molli, Vasilis Efthymiou, Sabrina Kirrane, Axel Ngonga, Diego Collarana, Renato Cerqueira, Mehwish Alam, Cassia Trojahn, and Sven Hertling, editors, *The Semantic Web: ESWC 2023 Satellite Events*, volume 13998, pages 182–185. Springer Nature Switzerland, Cham, 2023. Series Title: Lecture Notes in Computer Science.

[77] Ariana Martino, Michael Iannelli, and Coleen Truong. Knowledge Injection to Counter Large Language Model (LLM) Hallucination. In Catia Pesquita, Hala Skaf-Molli, Vasilis Efthymiou, Sabrina Kirrane, Axel Ngonga, Diego Collarana, Renato Cerqueira, Mehwish Alam, Cassia Trojahn, and Sven Hertling, editors, *The Semantic Web: ESWC 2023 Satellite Events*, volume 13998, pages 182–185. Springer Nature Switzerland, Cham, 2023. Series Title: Lecture Notes in Computer Science.

[78] Marvin Matthes, Oliver Guhr, Martin Krockert, and Torsten Munkelt. Leveraging llms for information extraction in manufacturing. In *IFIP International Conference on Advances in Production Management Systems*, pages 355–366. Springer, 2024.

[79] Philipp Mayring. *Qualitative content analysis: theoretical foundation, basic procedures and software solution*. SSOAR, Klagenfurt, Austria, 2014. Available online at https://www.ssoar.info/ssoar/handle/document/39517.

[80] Pablo N Mendes, Max Jakob, and Christian Bizer. *DBpedia: A multilingual cross-domain knowledge base*. European Language Resources Association (ELRA), 2012.

[81] Lars-Peter Meyer, Claus Stadler, Johannes Frey, Norman Radtke, Kurt Junghanns, Roy Meissner, Gordian Dziwis, Kirill Bulert, and Michael Martin. LLM-assisted Knowledge Graph Engineering: Experiments with ChatGPT. 2023. Publisher: arXiv Version Number: 1.

[82] Nandana Mihindukulasooriya, Sanju Tiwari, Daniil Dobriy, Finn Årup Nielsen, Tek Raj Chhetri, and Axel Polleres. Scholarly wikidata: Population and exploration of conference data in wikidata using llms. In *International Conference on Knowledge Engineering and Knowledge Management*, pages 243–259. Springer, 2024.

[83] Fedor Moiseev, Zhe Dong, Enrique Alfonseca, and Martin Jaggi. SKILL: Structured Knowledge Infusion for Large Language Models. 2022. Publisher: arXiv Version Number: 1.

[84] Max Moundas, Jules White, and Douglas C. Schmidt. Prompt patterns for structured data extraction from unstructured text. In *Proceedings of the 31st Conference on Pattern Languages of Programs, People, and Practices (PLoP 2024)*, pages 1–15. ACM, October 2024.

[85] M. Mountantonakis, M. Koumakis, and Y. Tzitzikas. Combining LLMs and hundreds of knowledge graphs for data enrichment, validation and integration case study: Cultural heritage domain. In *3rd International Conference On Museum Big Data*, MBD 2024, Athens, 2024.

[86] Michael Muller, Christine T Wolf, Josh Andres, Michael Desmond, Narendra Nath Joshi, Zahra Ashktorab, Aabhas Sharma, Kristina Brimijoin, Qian Pan, Evelyn Duesterwald, et al. Designing ground truth and the social life of labels. In *Proceedings of the 2021 CHI conference on human factors in computing systems*, pages 1–16, 2021.

[87] Omar Mussa, Omer Rana, Benoît Goossens, Pablo Orozco-terWengel, and Charith Perera. Towards enhancing linked data retrieval in conversational uis using large language models. In *International Conference on Web Information Systems Engineering*, pages 246–261. Springer, 2024.

[88] Nicole C. Nelson, Kelsey Ichikawa, Julie Chung, and Momin M. Malik. Mapping the discursive dimensions of the reproducibility crisis: A mixed methods analysis. *PLOS ONE*, 16(7):e0254090, July 2021.

[89] Christina Niklaus, Matthias Cetto, André Freitas, and Siegfried Handschuh. A survey on open information extraction. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 3866–3878, 2018.

[90] Alireza Nili, Mary Tate, and Alistair Barros. A critical analysis of inter-coder reliability methods in information systems research. In *ACIS 2017 Proceedings*, page 99. Australasian Conference on Information Systems, 2017.

[91] Janna Omeliyanenko, Albin Zehe, Andreas Hotho, and Daniel Schlör. CapsKG: Enabling Continual Knowledge Integration in Language Models for Automatic Knowledge Graph Completion. In Terry R.

Payne, Valentina Presutti, Guilin Qi, María Poveda-Villalón, Giorgos Stoilos, Laura Hollink, Zoi Kaoudi, Gong Cheng, and Juanzi Li, editors, *The Semantic Web – ISWC 2023*, volume 14265, pages 618–636. Springer Nature Switzerland, Cham, 2023. Series Title: Lecture Notes in Computer Science.

[92] Open Science Collaboration. Estimating the reproducibility of psychological science. *Science*, 349(6251):aac4716, August 2015.

[93] Wolfgang Otto, Andrea Zielinski, Behnam Ghavimi, Dimitar Dimitrov, Narges Tavakolpoursaleh, Karam Abdulahhad, Katarina Boland, and Stefan Dietze. Knowledge extraction from scholarly publications: The gesis contribution to the rich context competition. *Rich Search and Discovery for Research Datasets: Building the Next Generation of Scholarly Infrastructure, London, UK: Sage*, pages 107–126, 2020.

[94] Cliodhna O'Connor and Helene Joffe. Intercoder Reliability in Qualitative Research: Debates and Practical Guidelines. *International Journal of Qualitative Methods*, 19:1609406919899220, January 2020.

[95] Romana Pernisch, Daniel Dobriy, and Axel Polleres. The massive problem of remote changes in ontology reuse. 2025. The International World Wide Web Conference 2025.

[96] Vincent Perot, Kai Kang, Florian Luisier, Guolong Su, Xiaoyu Sun, Ramya Sree Boppana, Zilong Wang, Zifeng Wang, Jiaqi Mu, Hao Zhang, Chen-Yu Lee, and Nan Hua. LMDX: Language model-based document information extraction and localization. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar, editors, *Findings of the Association for Computational Linguistics: ACL 2024*, pages 15140–15168, Bangkok, Thailand, August 2024. Association for Computational Linguistics.

[97] Vayianos Pertsas, Marialena Kasapaki, and Panos Constantopoulos. An annotated dataset for transformer-based scholarly information extraction and linguistic linked data generation. In *Proceedings of the 9th Workshop on Linked Data in Linguistics@ LREC-COLING 2024*, pages 84–93, 2024.

[98] Maciej P Polak and Dane Morgan. Extracting accurate materials data from research papers with conversational language models and prompt engineering. *Nature Communications*, 15(1):1569, 2024.

[99] Fina Polat, Ilaria Tiddi, and Paul Groth. Testing prompt engineering methods for knowledge extraction from text. *Semantic Web*, pages 1–34, September 2024.

[100] Fina Polat, Ilaria Tiddi, and Paul Groth. Testing prompt engineering methods for knowledge extraction from text. *Semantic Web. Under Review*, 2024.

[101] Axel Polleres, Romana Pernisch, Angela Bonifati, Daniele Dell'Aglio, Daniil Dobriy, Stefania Dumbrava, Lorena Etcheverry, Nicolas Ferranti, Katja Hose, Ernesto Jiménez-Ruiz, et al. How does knowledge evolve in open knowledge graphs? *Transactions on Graph Data and Knowledge*, 1(1):11–1, 2023.

[102] David MW Powers. Evaluation: from precision, recall and f-measure to roc, informedness, markedness and correlation. *arXiv preprint arXiv:2010.16061*, 2020.

[103] Raminta Pranckutė. Web of Science (WoS) and Scopus: The Titans of Bibliographic Information in Today's Academic World. *Publications*, 9(1):12, March 2021.

[104] Ilana G. Raskind, Rachel C. Shelton, Dawn L. Comeau, Hannah L. F. Cooper, Derek M. Griffith, and Michelle C. Kegler. A Review of Qualitative Data Analysis Practices in Health Education and Health Behavior Research. *Health Education & Behavior*, 46(1):32–39, February 2019.

[105] David Rozado. The political preferences of llms. *PloS one*, 19(7):e0306621, 2024.

[106] Cynthia K Russell and David M Gregory. Issues for consideration when choosing a qualitative data management system. *Journal of Advanced Nursing*, 18(11):1806–1816, November 1993.

[107] Walid S Saba. Stochastic llms do not understand language: towards symbolic, explainable and ontologically based llms. In *International conference on conceptual modeling*, pages 3–19. Springer, 2023.

[108] Ahmad Sakor. *Knowledge Extraction from Unstructured Data*. PhD thesis, Leibniz Universität Hannover, Hannover, Germany, May 2023.

[109] Ahmad Sakor, Kuldeep Singh, and Maria-Esther Vidal. Biolinkerai: Capturing knowledge using llms to enhance biomedical entity linking.

In *International Conference on Web Information Systems Engineering*, pages 262–272. Springer, 2024.

[110] Johnny Saldaña. *The Coding Manual for Qualitative Researchers*. SAGE Publishing Inc., Thousand Oaks, California, 4th edition, 2021.

[111] Sunita Sarawagi et al. Information extraction. *Foundations and Trends® in Databases*, 1(3):261–377, 2008.

[112] Ansgar Scherp, Gerd Groener, Petr Škoda, Katja Hose, and Maria-Esther Vidal. Semantic Web: Past, Present, and Future. *Transactions on Graph Data and Knowledge (TGDK)*, 2(1):3:1–3:37, 2024. Artwork Size: 37 pages, 1598870 bytes Medium: application/pdf Publisher: Schloss Dagstuhl – Leibniz-Zentrum für Informatik.

[113] Xuanyu Shi, Wenjing Zhao, Ting Chen, Chao Yang, and Jian Du. A large language model approach to extracting causal evidence across study designs for evidence triangulation. *medRxiv*, pages 2024–03, 2024.

[114] Aditi Singh, Abul Ehtesham, Saket Kumar, and Tala Talaei Khoei. Agentic retrieval-augmented generation: A survey on agentic rag. *arXiv preprint arXiv:2501.09136*, 2025.

[115] Mai Skjott Linneberg and Steffen Korsgaard. Coding qualitative data: A synthesis guiding the novice. *Qualitative research journal*, 19(3):259–270, 2019.

[116] Steve Stemler. An overview of content analysis. *Practical Assessment, Research, and Evaluation*, 7(1):1–6, 2000.

[117] Bram Stoker. *Dracula*. Doubleday, Page & Co., Garden City, N.Y., 1920. [Pdf] Retrieved from the Library of Congress, https://www.loc.gov/item/21015337/.

[118] Anselm L. Strauss. *Qualitative Analysis for Social Scientists*. Cambridge University Press, Cambridge, UK, 1987.

[119] Jiashuo Sun, Chengjin Xu, Lumingyuan Tang, Saizhuo Wang, Chen Lin, Yeyun Gong, Lionel M. Ni, Heung-Yeung Shum, and Jian Guo. Think-on-Graph: Deep and Responsible Reasoning of Large Language Model on Knowledge Graph. 2023. Publisher: arXiv Version Number: 5.

[120] Kai Sun, Yifan Ethan Xu, Hanwen Zha, Yue Liu, and Xin Luna Dong. Head-to-Tail: How Knowledgeable are Large Language Models (LLM)? A.K.A. Will LLMs Replace Knowledge Graphs? 2023. Publisher: arXiv Version Number: 1.

[121] Zhuanlan Sun, Ruilin Zhang, Suhail A Doi, Luis Furuya-Kanamori, Tianqi Yu, Lifeng Lin, and Chang Xu. How good are large language models for automated data extraction from randomized trials? *medRxiv*, pages 2024–02, 2024.

[122] Tilahun Abedissa Taffa and Ricardo Usbeck. Leveraging llms in scholarly knowledge graph question answering. In *Proceedings of the First Scholarly QALD Challenge at the 22nd International Semantic Web Conference (ISWC 2023)*, pages 1–10, Athens, Greece, November 2023.

[123] Poorna TalkadSukumar and Ronald Metoyer. Replication and transparency of qualitative research from a constructivist perspective. *OSF Preprints*, 02 2019. Accessed March 6, 2025.

[124] Yiming Tan, Dehai Min, Yu Li, Wenbo Li, Nan Hu, Yongrui Chen, and Guilin Qi. Can ChatGPT Replace Traditional KBQA Models? An In-Depth Analysis of the Question Answering Performance of the GPT LLM Family. In Terry R. Payne, Valentina Presutti, Guilin Qi, María Poveda-Villalón, Giorgos Stoilos, Laura Hollink, Zoi Kaoudi, Gong Cheng, and Juanzi Li, editors, *The Semantic Web – ISWC 2023*, volume 14265, pages 348–367. Springer Nature Switzerland, Cham, 2023. Series Title: Lecture Notes in Computer Science.

[125] Yijun Tian, Huan Song, Zichen Wang, Haozhu Wang, Ziqing Hu, Fang Wang, Nitesh V. Chawla, and Panpan Xu. Graph Neural Prompting with Large Language Models. 2023. Publisher: arXiv Version Number: 2.

[126] Dominika Tkaczyk, Pawel Szostek, and Lukasz Bolikowski. Grotoap2-the methodology of creating a large ground truth dataset of scientific articles. *D-Lib Magazine*, 20(11/12), 2014.

[127] Tania Tudorache. Ontology engineering: Current state, challenges, and future directions. *Semantic Web*, 11(1):125–138, 2020.

[128] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[129] E Versi. " gold standard" is an appropriate term. *BMJ: British Medical Journal*, 305(6846):187, 1992.

[130] Blerta Veseli, Sneha Singhania, Simon Razniewski, and Gerhard Weikum. Evaluating Language Models for Knowledge Base Completion. In Catia Pesquita, Ernesto Jimenez-Ruiz, Jamie McCusker, Daniel Faria, Mauro Dragoni, Anastasia Dimou, Raphael Troncy, and Sven Hertling, editors, *The Semantic Web*, volume 13870, pages 227–243. Springer Nature Switzerland, Cham, 2023. Series Title: Lecture Notes in Computer Science.

[131] Luiz Victorino, Ronaldo Pilati, and Alexandre Linhares. Priming and prejudice: The bias effect of origin information on peer review, judgment and evaluation. *Avances en Psicología Latinoamericana*, 37(1):169–178, 2019.

[132] Aishwarya Vijayan. A Prompt Engineering Approach for Structured Data Extraction from Unstructured Text Using Conversational LLMs. In *2023 6th International Conference on Algorithms, Computing and Artificial Intelligence*, pages 183–189, Sanya China, December 2023. ACM.

[133] Denny Vrandečić, Lydia Pintscher, and Markus Krötzsch. Wikidata: The making of. In *Companion Proceedings of the ACM Web Conference 2023*, pages 615–624, 2023.

[134] Somin Wadhwa, Jay DeYoung, Benjamin Nye, Silvio Amir, and Byron C Wallace. Jointly extracting interventions, outcomes, and findings from rct reports with llms. In *Machine Learning for Healthcare Conference*, pages 754–771. PMLR, 2023.

[135] Kai Wang, Yuwei Xu, Zhiyong Wu, and Siqiang Luo. LLM as Prompter: Low-resource Inductive Reasoning on Arbitrary Knowledge Graphs. 2024. Publisher: arXiv Version Number: 1.

[136] Yu Watanabe, Koichiro Ito, and Shigeki Matsubara. Capabilities and challenges of llms in metadata extraction from scholarly papers. In *International Conference on Asian Digital Libraries*, pages 280–287. Springer, 2025.

[137] Lukas M Weber, Wouter Saelens, Robrecht Cannoodt, Charlotte Soneson, Alexander Hapfelmeier, Paul P Gardner, Anne-Laure Boulesteix, Yvan Saeys, and Mark D Robinson. Essential guidelines for computational method benchmarking. *Genome biology*, 20:1–12, 2019.

[138] Robert Philip Weber. *Basic Content Analysis*. Number 49 in Quantitative Applications in the Social Sciences. SAGE Publications, Newbury Park, 2 edition, 1990.

[139] Wei Wei, Xubin Ren, Jiabin Tang, Qinyong Wang, Lixin Su, Suqi Cheng, Junfeng Wang, Dawei Yin, and Chao Huang. LLMRec: Large Language Models with Graph Augmentation for Recommendation. 2023. Publisher: arXiv Version Number: 6.

[140] Evan Weingarten, Qijia Chen, Maxwell McAdams, Jessica Yi, Justin Hepler, and Dolores Albarracín. From primed concepts to action: A meta-analysis of the behavioral effects of incidentally presented words. *Psychological bulletin*, 142(5):472, 2016.

[141] Yilin Wen, Zifeng Wang, and Jimeng Sun. MindMap: Knowledge Graph Prompting Sparks Graph of Thoughts in Large Language Models. 2023. Publisher: arXiv Version Number: 4.

[142] Isabella Catharina Wiest, Fabian Wolf, Marie-Elisabeth Leßmann, Marko van Treeck, Dyke Ferber, Jiefu Zhu, Heiko Boehme, Keno K Bressem, Hannes Ulrich, Matthias P Ebert, et al. Llm-aix: An open source pipeline for information extraction from unstructured medical text based on privacy preserving large language models. *medRxiv*, 2024.

[143] JG Wrightson, P Blazey, D Moher, KM Khan, and CL Ardern. Gpt for rcts? using ai to determine adherence to reporting guidelines. *medRxiv*, pages 2023–12, 2023.

[144] Weiqi Wu, Hongqiu Wu, and Hai Zhao. Self-Directed Turing Test for Large Language Models, August 2024. arXiv:2408.09853 [cs].

[145] Yike Wu, Nan Hu, Sheng Bi, Guilin Qi, Jie Ren, Anhuan Xie, and Wei Song. Retrieve-Rewrite-Answer: A KG-to-Text Enhanced LLMs Framework for Knowledge Graph Question Answering. 2023. Publisher: arXiv Version Number: 2.

[146] Derong Xu, Wei Chen, Wenjun Peng, Chao Zhang, Tong Xu, Xiangyu Zhao, Xian Wu, Yefeng Zheng, Yang Wang, and Enhong Chen. Large Language Models for Generative Information Extraction: A Survey, October 2024. arXiv:2312.17617 [cs].

[147] Ruijie Xu, Zengzhi Wang, Run-Ze Fan, and Pengfei Liu. Benchmarking benchmark leakage in large language models. *arXiv preprint arXiv:2404.18824*, 2024.

[148] Linyao Yang, Hongyang Chen, Zhao Li, Xiao Ding, and Xindong Wu. Give Us the Facts: Enhancing Large Language Models with Knowledge Graphs for Fact-aware Language Modeling. *IEEE Transactions on Knowledge and Data Engineering*, pages 1–20, 2024. 0 citations (Crossref) [2024-03-04] Conference Name: IEEE Transactions on Knowledge and Data Engineering.

[149] Shuangtao Yang, Mao Teng, Xiaozheng Dong, and Fu Bo. LLM-Based SPARQL Generation with Selected Schema from Large Scale Knowledge Base. In Haofen Wang, Xianpei Han, Ming Liu, Gong Cheng, Yongbin Liu, and Ningyu Zhang, editors, *Knowledge Graph and Semantic Computing: Knowledge Graph Empowers Artificial General Intelligence*, volume 1923, pages 304–316. Springer Nature Singapore, Singapore, 2023. Series Title: Communications in Computer and Information Science.

[150] Liang Yao, Jiazhen Peng, Chengsheng Mao, and Yuan Luo. Exploring Large Language Models for Knowledge Graph Completion. 2023. Publisher: arXiv Version Number: 4.

[151] Shuang Yu, Tao Huang, Mingyi Liu, and Zhongjie Wang. BEAR: Revolutionizing Service Domain Knowledge Graph Construction with LLM. In Flavia Monti, Stefanie Rinderle-Ma, Antonio Ruiz Cortés, Zibin Zheng, and Massimo Mecella, editors, *Service-Oriented Computing*, volume 14419, pages 339–346. Springer Nature Switzerland, Cham, 2023. Series Title: Lecture Notes in Computer Science.

[152] Xie-Yun Zhang, Shi-Min Cai, Xiao-Rong Shen, Yang Han, Wen-Hao Hu, and Yan-Ru Zhang. Efficient Unified Information Extraction Model Based on Large Language Models, 2024.

[153] Yichi Zhang, Zhuo Chen, Wen Zhang, and Huajun Chen. Making Large Language Models Perform Better in Knowledge Graph Completion. 2023. Publisher: arXiv Version Number: 1.

[154] Yuqi Zhu, Xiaohan Wang, Jing Chen, Shuofei Qiao, Yixin Ou, Yunzhi Yao, Shumin Deng, Huajun Chen, and Ningyu Zhang. LLMs for Knowledge Graph Construction and Reasoning: Recent Capabilities and Future Opportunities. 2023. Publisher: arXiv Version Number: 2.

[155] Maurizio Zollo and Sidney G Winter. Deliberate learning and the evolution of dynamic capabilities. *Organization science*, 13(3):339–351, 2002.

[156] Arkaitz Zubiaga. Natural language processing in the era of large language models. *Frontiers in Artificial Intelligence*, 6:1350306, January 2024.

Appendix

A. Structured Literature Review Protocol

This protocol summarizes the main steps performed as part of the SLR aiming at approximating the costs of qualitative coding at WU Vienna. In this SLR, we follow the standard methodology [62] in line with using the SLR as a rapid research strategy [5].

Research Questions

This structured literature review addresses three research questions focused on understanding and quantifying qualitative coding practices, on the example of WU Vienna:

RQ1: What is the prevalence of qualitative coding in research projects at WU Vienna?

RQ2: What is the scale of the coding exercises in qualitative coding projects?

RQ3: What resources in terms of time, personnel and tools were required for qualitative coding projects?

Search Strategy

The search strategy employs a two-phase approach: (1) bibliographic database querying and (2) full-text retrieval for relevant papers.

Phase 1: Bibliographic Search The primary data source is Scopus, chosen for its its broad coverage [103]. Two queries have been constructed for the SLR:

1. A baseline query (Figure 19) to establish the overall publication landscape at WU Vienna, targeting English-language publications since 2010 across four document types: articles, conference papers, book chapters, and books.

2. A specialized query (Figure 20) focusing on qualitative research publications, extending the baseline query with methodological terms from commonly associated with qualitative research approaches.

Phase 2: Full-text Retrieval For identified publications, we implemented a systematic full-text retrieval process utilizing multiple services in sequence:

1. **Direct Links:** Initial attempt using any direct URLs provided in the bibliographic metadata, with content-type verification to ensure valid PDF retrieval.
2. **Unpaywall:** Secondary attempt through Unpaywall's API to locate legally available open access versions from institutional repositories and publisher platforms.
3. **Crossref:** Tertiary attempt utilizing Crossref's metadata API to locate authorized PDF versions through publisher-provided links.

In the retrieval process, retries for failed attempts and standardized file naming based on DOIs and normalized titles are implemented.

Baseline Query The baseline query establishes the overall publication landscape at WU Vienna. It covers core institutional affiliations using three common variations of the university's name. The query includes English-language publications from 2010 onwards, focusing on research publications (articles, conference papers, book chapters, and books).

```
(
    AF-ID("Vienna University of Economics and Business") OR
    AFFIL("WU Vienna") OR
    AFFIL("Wirtschaftsuniversitat Wien")
)
AND
(
    LANGUAGE(english)
)
AND
(
    DOCTYPE(ar) OR
    DOCTYPE(cp) OR
    DOCTYPE(ch) OR
    DOCTYPE(bk)
)
AND PUBYEAR > 2009
```

Figure 19: Baseline Scopus Query

Qualitative Research Query Building upon the baseline query, the search query shown in Figure 20 adds methodological terms commonly associated with qualitative research approaches. The terms were selected to capture various qualitative methods and techniques, including interviews, focus groups, case studies, and different analytical approaches like grounded theory and CA. Wildcards (e.g., `ethnograph*`, `phenomenolog*`) are used to capture variations in terminology.

Inclusion Criteria

Publications were included if they met all of the following criteria:

- published by authors affiliated with WU Vienna;
- published after 2009;
- published in English;
- document type is either article, conference paper, book chapter, or book;

```
(
    AF-ID("Vienna University of Economics and Business") OR
    AFFIL("WU Vienna") OR
    AFFIL("Wirtschaftsuniversitat Wien")
)
AND
(
    LANGUAGE(english)
)
AND
(
    DOCTYPE(ar) OR
    DOCTYPE(cp) OR
    DOCTYPE(ch) OR
    DOCTYPE(bk)
)
AND PUBYEAR > 2009
AND
(
    TITLE-ABS-KEY("qualitative" OR
                  "coding" OR
                  "interview" OR
                  "focus group" OR
                  "case study" OR
                  "ethnograph*" OR
                  "mixed-methods research" OR
                  "grounded theory" OR
                  "content analysis" OR
                  "thematic analysis" OR
                  "narrative analysis" OR
                  "phenomenolog*" OR
                  "participant observation" OR
                  "field research" OR
                  "field study" )
)
```

Figure 20: Qualitative Research Scopus Query

- contains studies following qualitative research methodologies;
- full text is available through any of the retrieval methods.

Exclusion Criteria

Publications were excluded if they met any of the following criteria:

- only mentions qualitative methods in passing without actually employing them;
- uses the term `qualitative` in a different context (e.g., `qualitative characteristics` in accounting);
- full text is not available through any of the retrieval methods;
- document is not a research publication (e.g., editorial, letter, or comment).

Data Extraction

Table 13 gives an overview of data points systematically extracted from identified publications.

Publication Information	
Basic Details:	· Title · Author(s) · Year · Document Type
Research Classification	
Methodology:	· Primary Approach · Research Design · Analysis Method
Coding Implementation	
Coders:	· Total Number · Roles · Expertise Level
Materials:	· Source Types · Volume · Format
Validation:	· Reliability Tests · Protocol · Quality Checks
Resource Allocation	
Requirements:	· Time · Personnel · Financial Cost

Table 13: Data Collection Framework for the Structured Literature Review

The extraction process was conducted through systematic full-text analysis of the retrieved publications, with particular attention to methodology

sections (e.g., `Materials and Methods`) where coding procedures and resource allocations are typically described.

Analysis

The following analysis quantifies the scale and implementation patterns of qualitative coding at WU Vienna, providing the foundation for estimating associated resource requirements.

Initial Dataset Coverage The initial Scopus query yielded 2127 publications. While this provides a substantial dataset for analysis, it is important to acknowledge the inherent limitations of relying solely on Scopus as a bibliographic database. A comparative analysis with the institutional research information system PURE[15] of the WU Vienna reveals a corpus of 11 384 publications for equivalent types of publications – 5.35 times larger, with the caveat that this comparison assumes no systematic differences in publication distribution patterns.

Identification of Qualitative Research The Scopus query for qualitative research papers lead to 310 results. From those, 264 or 85.16% have been identified as relevant in terms of containing qualitative research studies based on the title and abstract text. In total, 78 full papers could be retrieved – a corpus 3.38 times smaller than relevant papers considering no systematic differences in distribution. From these, 34 (44.16%) apply the QC methodology.

Coding Team Composition Among the 34 papers applying QC, 12 explicitly reported the number of coders. Of these, 41.7% (5/12) employed two coders, 33.3% (4/12) used three coders, 8.3% (1/12) utilized five coders, and 8.3% (1/12) engaged ten coders. Thus, the average number of coders per paper is slightly above 3 (3.08). The majority of papers reporting coder numbers relied on teams of coders. Of the 34 papers applying QC, only 9 (26.5%) explicitly reported the types of coders involved. In all cases but two, the coders were authors themselves, sometimes being joined by colleagues, native speakers and other groups.

Scale and Scope In 15 papers (44.1% of those applying qualitative coding), researchers reported the total number of items (unique artifacts) coded.

[15] See https://research.wu.ac.at/

The reported item counts varied widely, from as few as 12 to as many as 5000, with a mean of approximately 543 items and a median of 120 items.

Material Types Of the 34 papers applying QC, 19 (55.9%) reported the types of items coded. Interview transcripts were the most prevalent, used in 19 papers (55.9% of those reporting interview-related items). This included full transcripts, focus group discussions, semi-structured, in-depth, and cognitive debriefing interviews. Other item types included organizational and assessment materials (4 papers, 11.8%), websites (3 papers, 8.8%), and other specific materials like survey responses and mixed-method sources (2 papers, 5.9%).

Reliability Tests 24 papers (70.6%) reported ICR, with percent agreement ranging from 80% to 90% and in few cases coders spending time sharing impressions to achieve a consensual view. 20 papers (58.8%) made the coding protocols available.

Department Affiliations From the papers applying QC, four papers named affiliations to specific WU departments, with the departments distributed as follows: `Department of Management` named in two papers, `Department of Socioeconomics` in one paper and `Department of Foreign Language Business Communication` in one paper.

The systematic literature review process and its results are visualized in Figure 21, which shows the progressive filtering of publications from the ones officially reported to the final set of papers with detailed coding protocols.

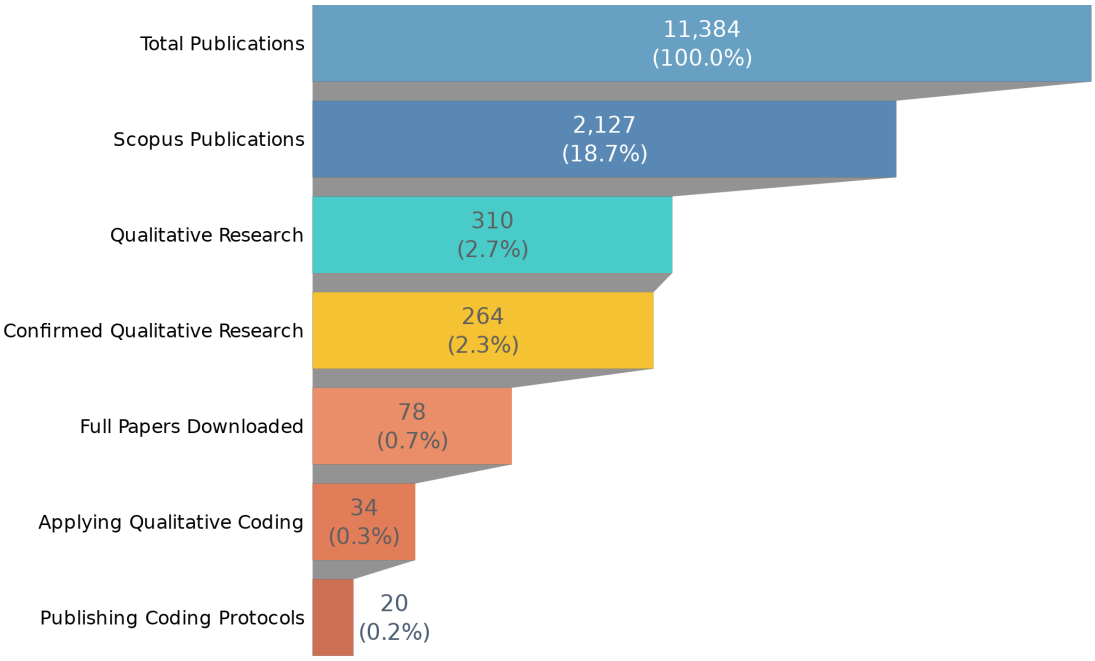


Figure 21: Structured Literature Review Funnel (Logarithmic Scale)

While few papers have reported the time and costs of the coding exercise, those could be presumed to be significant. The paper describing coding 5000 organization names reported 120 person-hours spend by a team of 3 coders for the task, yielding a circulatory rate of 2.08 items per minute (per person). As organizational names are fairly short and easy to directly annotate (not requiring any search through text), this can be considered the distant lower bound. Thus, an average study with the minimal conditions for reliability testing (3.08 coders) would require 13.4 person-hours.

At the current gross salary for full-time PhD students (EUR 68 880.00) without overhead costs, this would amount to the lower estimate of EUR 462.70 per QC task, not including overhead costs. In practice, the cost of similar tasks is likely to be far higher.