



6th International Conference on AI in Computational Linguistics

# Using LLMs and ontologies to extract causal relationships from medical abstracts

Alexandru Lecu<sup>a,b</sup>, Adrian Groza<sup>a</sup>, Lezan Hawizy<sup>b</sup>

<sup>a</sup>Technical University of Cluj-Napoca, Cluj-Napoca, Romania

<sup>b</sup>Digital Science & Research Solutions Ltd, London, United Kingdom

---

## Abstract

The substantiation of the causal relationships behind its development is very important in identifying possible interventions and early treatment. Knowledge Graphs (KG) play a crucial role in the medical research domain by organizing data into interconnected structures that represent relationships between entities such as disease, treatments, and progressions. This paper shows a complete workflow that demonstrates the extraction of causal relationships from medical abstracts using a fine-tuned GPT-based model and the integration of these relationships into a KG.

© 2024 The Authors. Published by Elsevier B.V.

This is an open access article under the CC BY-NC-ND license (<https://creativecommons.org/licenses/by-nc-nd/4.0>)

Peer-review under responsibility of the scientific committee of the 6th International Conference on AI in Computational Linguistics, ACLing 2024

**Keywords:** Causal Relation Extraction; Knowledge Graphs; Large Language Models; Age-Related Macular Degeneration.

---

## 1. Introduction

Large language models (LLMs) are considered the best technology for processing and transforming textual data. When fine-tuned to the medical language and causal relationships, these models do improve their performance. However, the advantages of LLMs should be complemented by structured representation of knowledge. Here, Knowledge Graphs (KG) provide a way in which the extracted data can be explored and used. We argue that the interleaving of LLMs and KGs brings benefits in terms of reliability and explainability.

We exemplify the interleaving of LLM with KG in medical abstracts related to Age-related Macular Degeneration (AMD). AMD affects mainly people over 50 and it is one of the leading causes of vision loss and blindness in older adults. The pathogenesis of AMD involves a combination of genetic, environmental and lifestyle factors, including smoking, hypertension, obesity, and prolonged exposure to sunlight. Early detection is critical in slowing its

---

\* Corresponding author. Tel.: +40 723 783 776

E-mail address: [Alexandru.Lecu@cs.utcluj.ro](mailto:Alexandru.Lecu@cs.utcluj.ro)

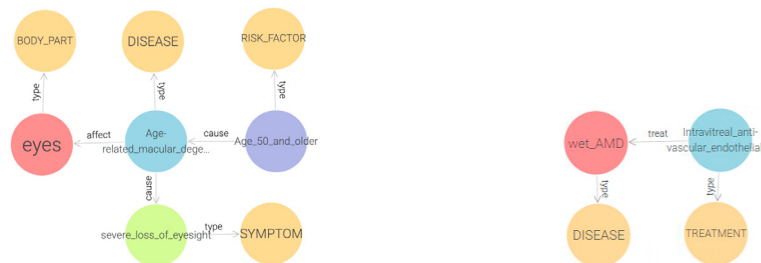


Fig. 1: Knowledge Graph example with cause and affect relations (Left) and treat relation (Right)

progression, and the substantiation of the causal relationships behind its development is very important in identifying possible interventions and early treatment. In the context of AMD, various Artificial Intelligence techniques have been applied [20]. For instance, knowledge graphs emerge as a possible tool to integrate and analyze diverse data types. A knowledge graph for AMD can facilitate understanding of the etiology, progression, and potential therapeutic targets of the disease by mapping the relationships between various entities involved in AMD.

Extracting causal relations from medical abstracts presents different challenges. Medical language is often filled with terms with multiple interpretations, acronyms, and specialized terminology. For example, words such as "anti-VEGF" are common in the treatment of wet AMD. Without a background in the treatment modalities of AMD, these terms are confusing. The language models applied to domain-specific medical texts must have the ability to understand medical contexts and the ability to disambiguate terms. Furthermore, the relations between entities are often implied, not explicitly stated, which requires context understanding and inference capabilities. There is also the issue of data quality and availability. The number of annotated datasets designed for relation extraction is relatively low, even more so for AMD disease, so fine-tuning a model effectively could be more difficult. The technical task is to extract both causal relations (RE) and named entities (NER) from medical abstracts. The investigation domain for named entities is restricted to 12 entities: (i) disease, (ii) symptom, (iii) treatment, (iv) risk factor, (v) test/diagnostic, (vi) gene, (vii) biomarker, (viii) complication, (ix) prognosis, (x) comorbidity, (xi) progression, (xii) body part. We consider 8 causal relations: (i) cause (causes or contributes to the occurrence of another entity), (ii) treat (is a treatment or intervention for another entity), (iii) present (is a symptom or manifestation of another entity), (iv) diagnose (is a test or diagnostic tool), (v) aggravate (worsens or exacerbates another entity), (vi) prevent (prevents or reduces the occurrence or development of another entity), (vii) improve (enhances or improves the condition or treatment of another entity), (ix) affect (affects a certain body part).

**Example 1 (Cause).** Given the text "Age-related macular degeneration is the most common cause of severe loss of eyesight among people 50 and older", the system extracts the KG in Fig. 1. The NER component identified AMD as a Disease, severe-loss-of-eyesight as a Symptom, eyes as a Body Part and age 50 and older as a Risk factor. The RE identified two relations of type cause and one relation of type affect.

**Example 2 (Treat).** Given the text, "There are drugs that treat but don't cure wet AMD. They include anti-vascular endothelial growth factor (anti-VEGF) injections.", the system extracts the knowledge graph from Figure 1. Note that the extracted relation is not directly specified in text; instead our system was able to identify it based on the context.

The contributions are: (i) a corpus of medical abstracts annotated with named entities and causal relations for AMD condition; (ii) the CausalAMD ontology used in the annotation process and automatically building the LLMs prompt; (iii) a tool able to extract causal relations from medical abstracts; (iv) a chatbot guided with RAG.

## 2. Related Work

Integrating causality, ontologies, and large language models (LLMs) creates new opportunities for advancing natural language understanding.

Large Language Models (LLMs) began with early statistical and neural network models, progressed significantly with the introduction of transformer[26] architectures like BERT[4] and GPT-2, and has seen tremendous growth

with the advent of models like GPT-3 and beyond, this opened up novel pathway for investigating causal inference in scientific literature. LLMs demonstrate promising capabilities in capturing and understanding causal relationships within text data, however once the task extends beyond language understanding, a deeper domain knowledge and interdisciplinary approach is required. Gao et al[6] evaluated ChatGPT for causal reasoning and found that it was prone to hallucinations. Additionally, they observed that ChatGPT demonstrates proficiency in discerning explicit causal relationships rather than implicit ones. Moreover, its performance is notably higher in sentences where events are closer together in proximity. Zečević et al[28] arrived at a similar conclusion that current LLMs are considered weak ‘causal parrots’ that at best recite data that is already embedded in the text. Jin et al [9] created a new benchmark task to explore whether causality can be inferred from correlation and came to the conclusion that it can only perform causal inference in in-distribution settings when variable names and textual expressions used in the queries are similar to those in the training set. Joshi et al[10] examined the limitations of LLMs in accurately performing causal reasoning and demonstrate that these models often struggle with distinguishing causation from mere correlation and tend to make errors in scenarios requiring causal inference. The study involved fine-tuning Llama2 models using these scenarios and observing their performance in inferring causal links. Despite the fine-tuning, the models frequently exhibited fallacies, indicating a fundamental challenge in causal reasoning. Key findings include: (i) LLMs often fail to distinguish between causally related events and those merely correlated; (ii) the presence of positional biases significantly affects the models’ performance in temporal relation tasks; (iii) the models struggle with counterfactual reasoning, frequently making errors in hypothetical scenarios. These findings underscore the importance of developing more robust methods, such as incorporating knowledge graphs, to improve the accuracy of causal inference in training large language models (LLMs).

Knowledge graphs have their origins in graph theory and logic from the 1950s and 1960s. During this period, foundational ideas about representing relationships between entities were explored through mathematical graphs. Researchers such as John McCarthy [15] and Marvin Minsky [19] laid the groundwork for knowledge representation in AI. The development of semantic networks in the 1960s and 1970s, including work by Quillian and Collins [3], provided early methods for modeling knowledge using interconnected nodes and relationships. These frameworks evolved into more sophisticated systems like frames and ontologies, which provided structured ways to represent complex knowledge. The 1990s brought a major shift with the introduction of the Semantic Web concept by Tim Berners-Lee (<https://www.w3.org/2001/sw/>). In 1999, Berners-Lee’s vision aimed to make data on the web machine-readable and interoperable through the use of technologies like RDF (Resource Description Framework) and ontologies [2]. This vision set the stage for the modern understanding and application of knowledge graphs by emphasizing the importance of structured, semantically-rich data on the web. Contemporary knowledge graphs have greatly enhanced search technology by deepening the contextual understanding of relationships between entities. Leading technology companies, including Google [24], Microsoft [16], and Facebook [5], have utilized knowledge graphs to improve search functionalities, recommendation systems, and user interactions.

In recent years, knowledge graphs have expanded their role into various domains beyond search engines. They are now integral to applications in healthcare, finance, and customer service, where they support advanced data analysis and personalized services[22]. The integration of knowledge graphs with machine learning and AI technologies has further enhanced their capabilities, making them crucial tools for intelligent systems and data-driven decision-making[8]. Salnikov et al[23] explored using knowledge graphs against pretrained LLMs to answer factoid questions and observed boosts of 4-6% while Yang et al[27] reviewed studies on enhancing pretrained language models (PLM) with knowledge graphs (KG) developing knowledge graph-enhanced large language models (KGLLMs). Additionally, projects like Text2Bench[18] utilize an ontology-driven approach to employ LLMs for populating knowledge graphs. Other developments include the application of Retrieval-Augmented Generation (RAG) and advanced prompting techniques in conjunction with knowledge graphs, as seen in projects like KRAGEN [14], to reduce hallucinations and misinformation. Lewis et al[11] also found that RAG trained on domain specific data generate more specific, diverse and factual language than a state-of-the-art parametric-only seq2seq baseline for language generation tasks. Given the novelty of this approach, there exists considerable scope for exploration and investigation, especially as tools continue to advance rapidly.

Ontologies are used to organise unstructured textual information in the medical domain [21], while extracting causal relations from medical texts is old hat in computer science [17]. There are different tools that can be used for knowledge graphs management. Instead of our solution based on Ontotext’s GraphDB, one can use AllegroGraph [1]

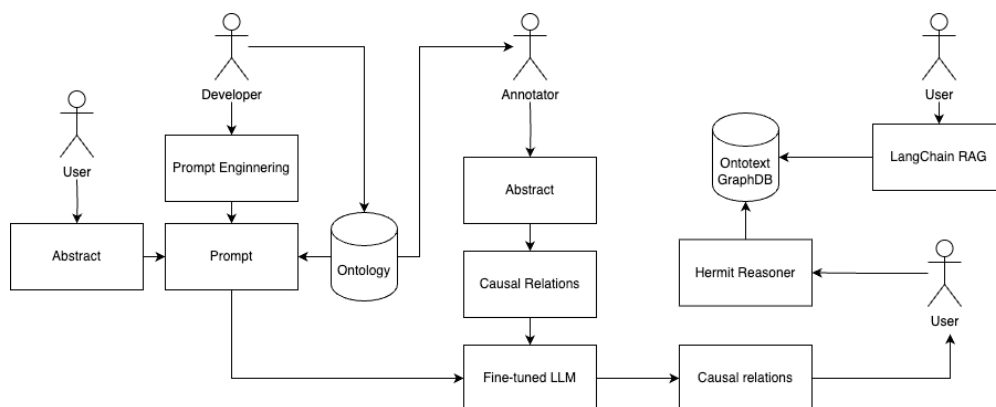


Fig. 2: System architecture

which also supports the integration of LLMs. LLMs have been used as a support instrument to extract knowledge from text and formalise it into an ontology, with the help of a plugin for the Protege ontology editor [13]. In our case, we found the integration of LLMs with Ontotext’s GraphDB more easier from the technical viewpoint, through its robust support of Python.

### 3. System architecture

Figure 2 shows the proposed solution. Firstly, an ontology that includes causal relations relevant for AMD is engineered using Protege editor. Secondly, we perform reasoning using Hermit and export all inferred knowledge into a KG managed within the Ontotext GraphDB tool. Thirdly, we fine-tuned GPT-3.5 Turbo to extract causal relations by providing annotated abstracts. Fourthly, using the Langchain library [25], we query the ontology in NL.

*Ontology engineering of causal relations for AMD.* This CausalAMD ontology serves as the foundational framework enabling LLMs and knowledge graphs to accurately interpret and link biomedical concepts. The ontology encapsulates different nuances of AMD, including risk factors, symptoms, treatments, and progression stages. The CausalAMD ontology was populated with data extracted from a corpus of medical abstracts, using a combination of automated LLM extraction methods and manual validation.

*Inferring knowledge from the ontology.* The next step uses the Hermit reasoner [7] to infer new knowledge. Reasoning increased the initial CausalAMD ontology from 489 to 549 logical axioms, from 246 to 248 class assertions, and from 236 to 294 object property assertions. Importing a reasoned ontology into GraphDB involves transferring the enriched knowledge base into the GraphDB environment, where it can be visualised in different formats

*Fine-tuning the LLM.* Extracting causal relations relies on an LLM. We fine-tuned the GPT-3.5-Turbo model and use it two folds: (1) to examine and extract relationships from text data; (2) to transform raw data into a KG. The model was trained on 31,743 tokens over 3 epochs with a batch size of 1 and a learning rate multiplier of 2. It cannot be shared publicly. By populating Ontotext GraphDB with relationships extracted by GPT, the system creates a navigable semantic network.

*Collecting and annotating the dataset.* Fine-tuning an LLM required an annotated corpus of medical abstracts. To collect a dataset of abstract for AMD, we use the Dimensions databases (<https://www.dimensions.ai>). We prioritize abstracts from clinical trials and observational studies published in the last ten years to ensure the relevance and currency of our data. We also excluded medical abstracts that were not written in English. The selection process involved screening of titles and abstracts. Finally, we selected 70 abstracts that contain a wide spectrum of terms related to AMD. The average number of words of abstracts in our corpus is around 350 words.

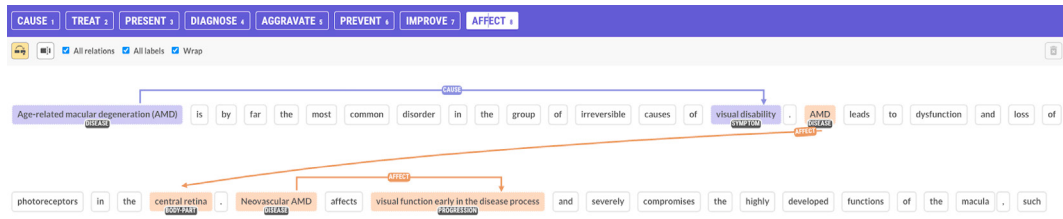


Fig. 3: Abstract annotation with causal relations from the CausalAMD ontology

For annotating the AMD abstracts, we developed a detailed annotation guideline to ensure consistency in identifying relations within the abstracts. The CausalAMD ontology, with the types of entity and the possible relations between them, served as a blueprint for the annotation guidelines. This ensured that the annotations were consistent with the ontology. It can also enhance the ability of the model to recognize and interpret causal relationships. We conducted a training session for our annotators to familiarize them with these guidelines and the annotation tool.

To annotate medical abstracts, we rely on Prodigy (<https://prodi.gy/>), an interactive and adaptable annotation tool. The interface can also display suggestions for possible relations based on previous annotations, which the user can confirm or edit. There were two annotators who independently worked to identify causal relationships in our selected corpus of medical abstracts. To quantify the agreement between annotators, we used Cohen's kappa score ( $\kappa$ ), which measures the frequency with which several annotators agree with each other. Cohen's kappa score is calculated with  $\kappa = \frac{p_o - p_e}{1 - p_e} = 0.604$ , where  $p_o$  is the observed agreement, which is the proportion of instances where both annotators agreed, and  $p_e$  is the expected agreement, which is the probability that both annotators would choose the same label if they both just randomly guessed. The kappa score ranges from -1 to 1 where a score of 1 means perfect agreement and a score below 0.2 indicates that there is none to slight agreement. The calculated Cohen's kappa score was 0.604, indicating a substantial level of agreement among annotators.

**Example 3 (Annotating medical abstracts).** Figure 3, illustrates a segment of a medical abstract related to AMD. Entities such as "disease", "body part", and "progression" are highlighted in distinct colors corresponding to the respective classes in the ontology. Arrows, which represent types of causal relations, such as "cause" or "affect", show the interactions between these entities. Visual representation serves as a guide for annotators in ensuring consistency and accuracy in the annotation process.

The concepts in the Causal AMD ontology were mirrored in the annotation process. Each entity identified within the medical abstracts was annotated according to the predefined class in the ontology, ensuring a one-to-one link between the ontology class and the annotation process. Similarly, the relations defined in the ontology were strictly used when the annotations started. Using the same relationships, the annotators were able to consistently label the interactions between entities.

A number of 70 abstracts were annotated from different publications or clinical trials referring to AMD. After the annotation process, the data was adapted to fine-tune GPT4. This preparation included cleaning up the text and splitting the dataset into training and validation sets. A balanced representation of the different types of causal relationship in each set was ensured. To ensure the integrity of our dataset, a review session was held in which the two annotators verified the annotations and measured the quality of the results.

The quality of specificity of the annotations allowed for a more precise adjustment of the model to the task of extracting causal relations. Some results show a significant improvement in the model accuracy and reliability; this shows the importance of a well-formed dataset in the application of LLMs to specialized field such AMD.

Within these 70 abstracts, we annotated 326 relations, which we split into 250 relations for training and 76 for validation. This data was used to fine-tune our GPT-3.5 turbo model.

*Enriching Knowledge Graph using GPT.* Knowledge graphs play a crucial role in the medical research domain by organizing data into interconnected structures that represent complex relationships between entities such as disease, treatments, and progressions. The enrichment of these graphs involves the integration of new and relevant information. Using a Generative Pre-trained transformer (GPT) represents an approach to augment KGs, this way we could use advanced capabilities in language understanding and information generation.

GPT is able to extract information from different texts of the medical literature, even if the relationships are not directly specified in the text. After we fine-tuned GPT with our dataset, we will know even more how to extract relations related to AMD from medical abstracts. In this way, by processing multiple medical abstracts, GPT can identify new entities and their relations that are not currently present in the knowledge graph.

The methodology for enriching KG with GPT involves: (i) *Data Ingestion*: medical abstracts are processed with GPT, which analyze the text and extract relevant information about entities and relations; (ii) *Validation*: The information that comes from GPT is verified by the user to ensure the correctness of the data and relevance. (iii) *Integration into the KG*: data are parsed into relations like (entity<sub>1</sub>, relation, entity<sub>2</sub>) and using SPARQL queries is integrated in the KB; (iv) *Reason on ontology*: after new data are integrated, the Hermit reasoner infers new knowledge.

Training loss is an indicator of how well the model is learning during the training phrase. In our fine-tuning process, the GPT-3.5-Turbo model achieved a training loss of 0.0262. This low value suggests that the model has a high level of accuracy in its learning capability. In contrast, the validation loss assesses how well the model performs on a dataset that it has not been exposed to during training. The validation loss of our model is 0.5504, which gives insight into the performance of the model with new data. The significant discrepancy between the training and validation losses indicates possible overfitting, in which the model excels on the training data but struggles with new, unseen data.

We designed several prompts that guide the LLM to identify and classify entities according to predefined medical labels such as disease, symptom, and others. Additionally, the prompt instructs the model to extract specific relations among these entities. Until it reached its final version, we needed to adapt it several times due to its inconsistent responses or unrelated extracted relations.

The first prompt asked the model to identify entities and their relations from the text: *"Prompt1: Given the following text, identify and list all the explicit causal relationships mentioned. For each relationship, specify the causal agent (e.g., treatment) and the effect (e.g., symptom). Format your response as a series of statements in the form of causal agent - relation - effect.* As expected, the output was too general and did not contain specific relations. To increase the precision of the response, we added explicit instructions to identify entities belonging to specific categories and extract relationships between them. In the same way, a list of relationships that interest us was added to the prompt. Since the response of the model will be processed further to add the relations to the knowledge graph, we also introduced a structured format for the model to follow, ensuring consistency.

After testing with real medical abstracts, we could improve some areas. The response usually contained an introductory text before listing the extracted relations from the abstract or some enumeration, which did not help us to parse the text, and the extracted relations had to be in JSON format. Also, some entity names contain parentheses or commas, which influences the JSON format. So, instead of using commas, the model is now able to split the entity name into multiple ones and create relations accordingly.

Each version of the prompt was tested with different medical abstracts to ensure the precision and relevance of the response. The final version of our prompt (Listing 1) takes the medical abstract (line 1), concepts (line 2) and relations (line 4) directly obtained by querying the Causal AMD ontology. The LLM is asked to translate the output into a form understood by the Ontotext knowledge graph (lines 8-9).

```

1 Given the text {text} identify entities belonging to the following labels:
2 disease, symptom, treatment, risk_factor, test, gene, biomarker, complication, prognosis, comorbidity, progression, body-part.
3 Then, extract relationships among these entities based on the following relations: cause, treat, present, diagnose, aggravate, prevent, improve, affect.
4 When presenting entity names, ensure the names do not contain parentheses. If an entity's common name typically includes parentheses, rephrase or
5 abbreviate the name without using parentheses. Entity names must not contain commas. Instead, split entity and create separate relations.
6 Present only the relationships extracted, in the specified format, without any introductory text, summary, or enumeration. Use the format:
7 {{{'relation_type': 'relation type', 'entity1_type': 'entity1_type', 'entity1_name': 'entity1_name', 'entity2_type': 'entity2_type', 'entity2_name': '
entity2_name'}}}
IMPORTANT: Output must contain only the relations in the specified format, with no other text or numbers included.

```

Listing 1: Final version of the prompt takes the abstract with the entities and relations of interest directly retrieved from the Causal AMD ontology

Note that if the ontology is updated with new entities or relations, the prompt will sync with the ontology. Technically, the concepts and relations listed in the prompt are obtained by directly querying the ontology. Note also that this design of the prompt handles text containing enumerations (line 9).

**Example 4 (Dealing with enumerations).** Consider the text *Age-related macular degeneration affects the ability to read and drive a car, where it enumerates two distinct risk factors, an LLM must not only recognize that AMD affects multiple entities. This requires the model to parse and understand the enumeration and distinguish the different items. For this example, LLM returns two relations: AMD affects ability to read and AMD affects ability to drive a car.*

The prompt was used to extract relations from a corpus of AMD-related medical abstracts.

```

1 [{"entity1_name": "Age-related macular degeneration", "entity1_type": "disease",
2   "entity2_name": "blindness", "entity2_type": "symptom",
3   "relation_type": "cause"},
4 [{"entity1_name": "Age-related macular degeneration", "entity1_type": "disease",
5   "entity2_name": "eye", "entity2_type": "body_part",
6   "relation_type": "affect"},
7 [{"entity1_name": "smoking", "entity1_type": "risk_factor",
8   "entity2_name": "Age-related macular degeneration", "entity2_type": "disease",
9   "relation_type": "present"}]]

```

Listing 2: Extracted relations using fine-tuned GPT

*Querying the knowledge graphs.* The system incorporates the Retriever-Augmented Generation (RAG) technique. To show the benefits of the automatic extraction of relationships and entities from text inputs and the RAG system, we developed a web application using the VueJS framework. The user can give a text as input, see the relations extracted from that text, and add them to the graph database. After adding new data, the user can see the modified knowledge graph in a window. In addition, the user can interact with the Knowledge Graph by communicating with an AI agent in the context of the AMD condition.

#### 4. Running scenario and system evaluation

We present a complete workflow that demonstrates the extraction of causal relationships from AMD-related medical abstracts using the fine-tuned GPT and the integration of these relationships into a knowledge graph. Reasoning with Hermit and querying the enriched ontology in natural language based on Langchain will also be exemplified. Consider the abstract in Example 5.

**Example 5 (Running abstract on AMD).** *"Age-related macular degeneration (AMD) is one of primary blinding eye disease among people over 65 years in China. The anti-VEGF antibody treatment is proved useful for Neovascular Age-related Macular Degeneration (nAMD) by many studies. Bevacizumab is the only available low-cost type of anti-VEGF drug currently in China. This study is a multi-center, randomized trial of Bevacizumab effective dose and safety for nAMD. This study is to explore the effective therapeutic approach that the majority of patients in China can bear establishing a suitable treatment for China."* [12]

This abstract was used in the validation dataset when we fine-tuned the GPT model, and it was annotated with five relations in Listing 3. For instance, in line 4, the relation cause links AMD (which is an instance of Disease) with blinding (which is an instance of Complication).

```

1 {DISEASE: Age-related macular degeneration, RELATION: cause, COMPLICATION: blinding}
2 {DISEASE: Age-related macular degeneration, RELATION: affect, RISK_FACTOR: people over 65 years}
3 {TREATMENT: anti-VEGF antibody treatment, RELATION: treat, DISEASE: nAMD}
4 {DISEASE: AMD, RELATION: cause, COMPLICATION: blinding}
5 {TREATMENT: anti-VEGF antibody treatment, RELATION: treat, DISEASE: Neovascular Age-related Macular Degeneration}

```

Listing 3: Annotated relations from the abstract in Example 5

*Extracting relations.* The model extracted the relations shown in Listing 4. The relations 1, 2, and 3 were correctly extracted. The named entities were also correctly identified, with the exception of the term blinding annotated as Complication (line 1 in Listing 3) and identified by the model as Symptom (line 1 in Listing 4).

The extraction missed two annotated relations (lines 4 and 5 in Listing 3: *AMD causes blinding* respectively *anti-VEGF antibody treatment treats nAMD*). We observed that the errors are related to the existence of abbreviations. To handle this, the knowledge graph includes the *sameAs* relations between nodes. Thus, despite the omission of such relations by the model, the knowledge graph bridges these gaps by recognizing *AMD* and *Age-related macular degeneration* as equivalent entities. Note that the extracted relations also include two new relations: (lines 4 and 5 Listing 4). This indicate a possible advantage of using LLMs in identifying additional relevant information that was not captured during manual annotation.

```

1 {DISEASE: Age-related macular degeneration, RELATION: cause, SYMPTOM: blinding}
2 {DISEASE: Age-related macular degeneration, RELATION: affect, RISK_FACTOR: people over 65 years}
3 {TREATMENT: anti-VEGF antibody treatment, RELATION: treat, DISEASE: Neovascular Age-related Macular Degeneration}
4 {DISEASE: Age-related macular degeneration, RELATION: present, BODY_PART: eye}
5 {TREATMENT: Bevacizumab, RELATION: treat, DISEASE: Neovascular Age-related Macular Degeneration}

```

Listing 4: Extracted relations using fine-tuned GPT

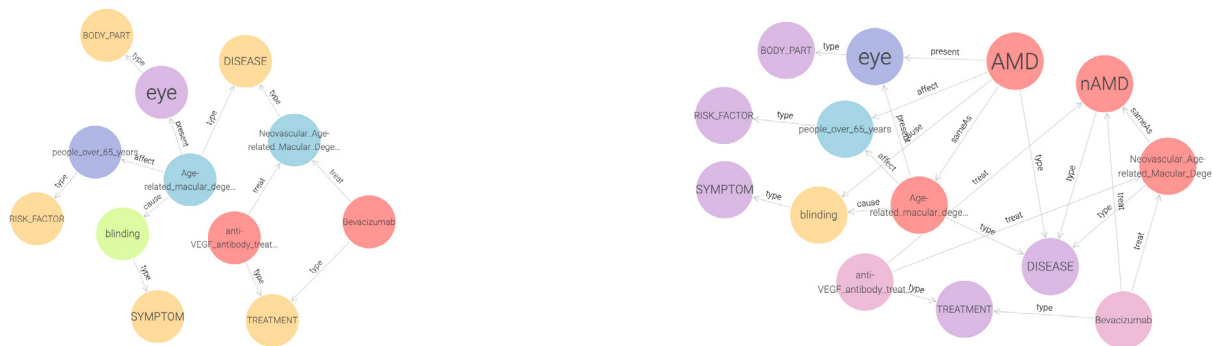


Fig. 4: Knowledge Graph resulted from the running scenario (left). Enriching the graph by reasoning with Hermit (right)

*Constructing the knowledge graph.* After the extraction process, the GPT results need to be parsed, isolating each entity and relation in a structured format. These relations are shown on the web interface, where the user can verify the relations and add them to the KG. If changes are needed, the user can also modify the extracted relations. This design is in line with the human-in-the-loop recommendation for AI-systems.

Technically, we constructed SPARQL queries to insert new entities and relations into KG. The queries are dynamically created based on the relations validated by the user. Using the SPARQLWrapper Python package, a Python wrapper around the SPARQL service, we execute these queries to enrich the Ontotext GraphDB. The generated KG (Figure 4) can be explored through a web interface.

*Reasoning with Hermit.* To show the advantages of reasoning on top of LLM outputs, we added two individuals in addition to those generated by GPT: *AMD* which is an abbreviation of age-related macular degeneration, and *nAMD*, which is an abbreviation of neovascular age-related macular degeneration. These individuals were connected to their original individual through the *sameAs* relation. With reasoning, the corresponding connections were also derived from the acronyms (Figure 4). Reasoning increases the initial CausalAMD ontology from 60 to 67 logical axioms, from 7 to 9 class assertions, and from 5 to 10 object property assertions. The inferred knowledge appears in Listing 5.

```

1  (DISEASE: AMD)
2  (DISEASE: nAMD)
3  (DISEASE: AMD, RELATION: cause, SYMPTOM: blinding)
4  (DISEASE: AMD, RELATION: present, BODY.PART: eye)
5  (DISEASE: AMD, RELATION: affect, RISK_FACTOR: people over 65 years)
6  (TREATMENT: anti-VEGF antibody treatment, RELATION: treat, DISEASE: nAMD)
7  (TREATMENT: Bevacizumab, RELATION: treat, DISEASE: nAMD)
    
```

Listing 5: Inferred relations

*Querying with Langchain.* Using the reasoned knowledge graph, the next step for the user is to query it using natural language. The user can initiate a dialogue with an AI agent based on our fine-tuned LLM trained in medical abstracts related to age-related macular degeneration. LangChain facilitates an interactive and informative dialogue between human agents and AI agents, specifically in the context of AMD medical queries. The goal is to reduce the disparity between structured and unstructured data, providing context to LLM to enhance the accuracy of the results.

Querying with LangChain involves a dynamic interaction in which the human agent asks questions related to age-related macular degeneration, and the AI agent powered by the LLM and informed by the knowledge graph provides specific answers. By utilizing SPARQL queries that are generated dynamically according to the user’s input, LangChain can effectively retrieve information from the graph and provide pertinent responses to the user. This setup allows the AI agent to query structured data with precision and also understand and interpret complex queries through natural language processing. The integration of LangChain with Ontotext GraphDB ensures that the knowledge graph can be dynamically updated with the latest research. Means that the AI agent becomes more proficient at answering more complex medical queries.

Analyzing the outputs, we can observe how the knowledge graph structure can be used to understand and generate responses to queries. In example 6, the human agent asks what affects AMD and the AI agent responds by using the *affect* property to relate AMD to *people over 65 years old*. This is directly supported by the ontology in which these two individuals are linked. The system’s output is limited by the information available in the knowledge graph, as it relies solely on the data within it.

When the user asks about potential treatments for nAMD, 7, the AI agent identifies *Bevacizumab*, an individual of the Treatment class, which is connected to *Neovascular Age-related Macular Degeneration or nAMD* via *treat* object property. In this case, the AI agent not only provides the correct outcome, but also establishes a connection that is not explicitly found in our knowledge graph, namely *Bevacizumab, an anti-VEGF antibody treatment*.

In Example 8 the AI agent is asked to define AMD. Even the knowledge graph provides some basic information about AMD, including symptoms like *blinding*, it may not contain all possible details or specific relations needed to fully understand the user question. In instances like these, when the knowledge graph is missing certain details or particular relationships, the AI's capacity to produce responses might be restricted, emphasizing the significance of consistently enhancing the knowledge graph.

**Example 6 (Sample of question answering based on RAG).** Human Agent: "What affect AMD?" AI Agent: "Based on the information provided from the SPARQL query result, age appears to be a factor that affects AMD (Age-related Macular Degeneration), specifically people over 65 years old. If you are looking for more detailed factors that affect AMD, I would need additional information from further data or research as the current data only indicates the age group impacted."

**Example 7 (Sample of question answering based on RAG).** Human Agent: "What are the potential treatments for neovascular age-related macular degeneration (nAMD)?" AI Agent: "One potential treatment for neovascular age-related macular degeneration (nAMD) is Bevacizumab, which is an anti-VEGF antibody treatment".

**Example 8 (Sample of question answering based on RAG).** Human Agent: "What is AMD?" AI Agent: "Based on the information provided, I don't have any details to share about AMD. If you have any specific questions or need information on a different topic, feel free to ask!"

*Evaluation* To evaluate the effectiveness of our relation extraction system utilizing large language models (LLMs), we performed a comparative analysis using a set of 20 abstracts. The evaluation compared the relations extracted by the GPT model with those annotated manually by a human.

The GPT model extracted a higher number of relations (226) compared to the annotated relations (160). This indicates that the GPT model is capable of identifying a larger volume of relational data from the abstracts. The model also identified a higher number of unique subjects (143) and unique objects (127) than human annotations (56 subjects and 74 objects, respectively). This suggests that the model has a broader scope in recognizing various entities within the text. The top subjects extracted by the GPT model and human annotations show some overlap but also differences. For instance, "Age-related macular degeneration" and its variations are prominently identified by both, but the GPT model also identified subjects like "iSONEP" and "Photodynamic therapy" more frequently.

To assess the correctness of the GPT-extracted relations, we manually compared the extracted relations with the annotated relations. The GPT model demonstrated a high degree of precision in identifying the most common types of relations such as "cause," "treat," and "affect." For example, the relation "cause" was identified 57 times by the GPT model, and it matched closely with the 66 instances identified by human annotation.

A false positive in this context is a relation identified by the GPT model that does not correspond to any relation in the annotated data. The presence of false positives indicates areas where the model may have misinterpreted the context or inferred relationships that were not intended. For example, the relation "diagnose" appeared 48 times in GPT-extracted relations but was not a top relation in human annotations, suggesting potential over-identification.

To quantify the precision, we calculate the precision, which is defined as  $precision = \frac{\text{Number of Correctly Identified Relations}}{\text{Total Number of Relations Extracted by GPT}} = \frac{174}{226} = 0.769$ . This indicates that approximately 76.9% of the relations extracted by the GPT model were correct compared to human annotations.

**Acknowledgment.** A. Groza as supported by the the Executive Agency for Higher Education, Research, Development and Innovation Funding (UEFISCDI)—grant PN-III-P2-2.1-PED-2021-2709. The code is available [Here](#).

## 5. Conclusion

The contributions are: (i) a corpus a medical abstracts annotated with named entities and causal relations for AMD condition; (ii) the CausalAMD ontology used in the annotation process and automatically building the LLMs prompt; (iii) a tool able to extract causal relations from medical abstract (iv) a chatbot guided with RAG.

The interested researcher can extend our work with several directions: First, we focused here on positive relations. However, in medical abstract we encountered negative relations (e.g. "there are drugs that treat but don't cure wet AMD"). One approach would be to consider "don't cure" relation as a new relation in the ontology, with the advantage of treating them in the same way as positive relations. Second, we considered only crisp relations. Uncertain or fuzzy relations (e.g. causes in many cases, causes in most cases, causes in few cases) can be considered, by relying for instance on Fuzzy Description Logic to engineer our CausalAMD ontology.

## References

- [1] Alaoui, K., Bahaj, M.: Evaluation criteria for rdf triplestores with an application to allegrograph. *International Journal of Advanced Computer Science and Applications* **11**(6) (2020)
- [2] Berners-Lee, T.: Semantic web road map. W3C (1999), <https://www.w3.org/DesignIssues/Semantic.html>
- [3] Collins, A.M., Quillian, M.R.: Retrieval time from semantic memory. *Journal of Verbal Learning and Verbal Behavior* **8**(2), 240–247 (1969)
- [4] Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805* (2018), <https://arxiv.org/abs/1810.04805>
- [5] Facebook: Introducing graph search. Facebook Official Blog (2013), <https://about.fb.com/news/2013/01/introducing-graph-search/>
- [6] Gao, J., Ding, X., Qin, B., Liu, T.: Is ChatGPT a good causal reasoner? a comprehensive evaluation. In: Bouamor, H., Pino, J., Bali, K. (eds.) *Findings of the Association for Computational Linguistics: EMNLP 2023*. pp. 11111–11126. Association for Computational Linguistics, Singapore (Dec 2023), <https://aclanthology.org/2023.findings-emnlp.743>
- [7] Glimm, B., Horrocks, I., Motik, B., Stoilos, G., Wang, Z.: Hermit: an owl 2 reasoner. *Journal of automated reasoning* **53**, 245–269 (2014)
- [8] Hogan, A., Harth, A., Polleres, A.: Knowledge graphs: New opportunities and challenges. *ACM Computing Surveys* **54**(8), 1–36 (2021)
- [9] Jin, Z., Liu, J., Lyu, Z., Poff, S., Sachan, M., Mihalcea, R., Diab, M., Schölkopf, B.: Can large language models infer causation from correlation? (2023)
- [10] Joshi, N., Saparov, A., Wang, Y., He, H.: LLMs are prone to fallacies in causal inference (2024), <https://arxiv.org/abs/2406.12158>
- [11] Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., Yih, W.t., Rocktäschel, T., Riedel, S., Kiela, D.: Retrieval-augmented generation for knowledge-intensive nlp tasks. In: Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., Lin, H. (eds.) *Advances in Neural Information Processing Systems*. vol. 33, pp. 9459–9474. Curran Associates, Inc. (2020)
- [12] Li, X.: Bevacizumab for neovascular age-related macular degeneration, <https://clinicaltrials.gov/study/NCT01306591?a=1>, (2008, January - 2010, June)
- [13] Mateiu, P., Groza, A.: Ontology engineering with large language models. In: *2023 25th International Symposium on Symbolic and Numeric Algorithms for Scientific Computing (SYNASC)*. pp. 226–229 (2023)
- [14] Matsumoto, N., Moran, J., Choi, H., Hernandez, M.E., Venkatesan, M., Wang, P., Moore, J.H.: KRAGEN: a knowledge graph-enhanced RAG framework for biomedical problem solving using large language models. *Bioinformatics* **40**(6), btac353 (06 2024)
- [15] McCarthy, J.: Protocol for a logical calculus. MIT (1956), <https://www-formal.stanford.edu/jmc/1956/protocol.html>
- [16] Microsoft: Introducing bing's knowledge graph. Bing Blog (2011), <https://blogs.bing.com/search/2011/05/09/introducing-bings-knowledge-graph/>
- [17] Mihäilä, C., Ohta, T., Pyysalo, S., Ananiadou, S.: Biocause: Annotating and analysing causality in the biomedical domain. *BMC bioinformatics* **14**, 1–18 (2013)
- [18] Mihindukulasooriya, N., Tiwari, S., Enguix, C.F., Lata, K.: Text2kgbench: A benchmark for ontology-driven knowledge graph generation from text (2023)
- [19] Minsky, M.: *Semantic Information Processing*. MIT Press (1968)
- [20] Muntean, G., Marginean, A., Groza, A., Damian, I., Roman, S., Hapca, M., Muntean, M.V.; Nicoară, S.: The predictive capabilities of artificial intelligence-based oct analysis for age-related macular degeneration progression—a systematic review. *Diagnostics* **13**(14) (2023), <https://www.mdpi.com/2075-4418/13/14/2464>
- [21] Panzarella, G., Veltri, P., Alcaro, S.: Using ontologies for life science text-based resource organization. *Artificial Intelligence in the Life Sciences* **3**, 100059 (2023)
- [22] Paulheim, H.: Knowledge graphs: New opportunities and challenges. *ACM Computing Surveys* **54**(8), 1–36 (2017)
- [23] Salnikov, M., Le, H., Rajput, P., Nikishina, I., Braslavski, P., Malykh, V., Panchenko, A.: Large language models meet knowledge graphs to answer factoid questions (2023)
- [24] Singhal, A.: Introducing the knowledge graph: Things, not strings. Google Official Blog (2012), <https://www.blog.google/products/search/introducing-knowledge-graph-things-not-strings/>
- [25] Topsakal, O., Akinci, T.C.: Creating large language model applications utilizing langchain: A primer on developing llm apps fast. In: *International Conference on Applied Engineering and Natural Sciences*. vol. 1, pp. 1050–1056 (2023)
- [26] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L.u., Polosukhin, I.: Attention is all you need. In: Guyon, I., Luxburg, U.V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., Garnett, R. (eds.) *Advances in Neural Information Processing Systems*. vol. 30. Curran Associates, Inc. (2017)
- [27] Yang, L., Chen, H., Li, Z., Ding, X., Wu, X.: Give us the facts: Enhancing large language models with knowledge graphs for fact-aware language modeling. *IEEE Transactions on Knowledge and Data Engineering* **PP**, 1–20 (01 2024)
- [28] Zečević, M., Willig, M., Dhimi, D.S., Kersting, K.: Causal parrots: Large language models may talk causality but are not causal (2023)