Conversation-Driven Refinement of Knowledge Graphs: True Active Learning with Humans in the Chatbot Application Loop

Dominik Buhl[1], Daniel Szafarski[1], Laslo Welz[1], and Carsten Lanquillon[1]

Heilbronn University of Applied Sciences, 74076 Heilbronn, Germany
carsten.lanquillon@hs-heilbronn.de

Abstract. The value of knowledge-grounded cognitive agents is often limited by a lack of high-quality knowledge. Although advances in natural language processing have substantially improved knowledge-extraction capabilities, the demand for different types of knowledge fragments and the potential for error in extraction processes has created a next generation of the knowledge acquisition bottleneck. Instead of waiting for a perfect knowledge base, we propose a design for an agent that is aware of these issues and that actively seeks feedback from users in conversations to improve its knowledge and extraction processes. This approach allows for imperfection and incompleteness, and for the agent to improve over time. Any feedback provided by the users in this conversational application loop is used to not only refine the underlying knowledge graph, but also to improve the knowledge extraction processes. Eventually, the agent's knowledge and the quality of its answers rises while talking to its users.

Keywords: Conversational AI · Human-in-the-Loop AI · Knowledge-Grounded Cognitive Assistants · Knowledge Graph Refinement.

1 Introduction

Chatbots or, more specifically, task-specific cognitive agents, also referred to as intelligent virtual assistants, with natural language interfaces are steadily gaining attention and traction in many application domains [1, 2, 56]. This is largely due to the emergent abilities [55] of current transformer-based [51] large language models [45, 46, 59, 5, 10, 39, 38]. Impressive as their generative linguistic abilities may be, in disciplines other than creative ones their value is limited due to their tendency to hallucinate and, thus, due to lacking faithfulness and factuality [34, 20]. One way to address this critical issue is to enhance language models with internal or external knowledge bases [19] yielding knowledge-grounded cognitive agents or systems [16, 22, 1, 37]. Using internal domain- and organization-specific knowledge bases can not only enhance faithfulness and factuality, but also helps to protect confidential data and to support knowledge preservation and transfer among co-workers.

Yet, the value of knowledge-grounded cognitive agents is often limited due to a lack of relevant and accessible high-quality domain-specific knowledge. Since the manual construction of knowledge bases is time-consuming and tedious, we focus on automating and refining the process of constructing a knowledge graph (KG) as a common form of knowledge base [17]. Typically, knowledge fragments are extracted as relations between two entities, so-called RDF-triples, from available documents [49, 25] such as project artifacts and, also, from conversions among co-workers in project-specific communication channels and groups.

Advances in natural language processing (NLP) based on deep learning–in particular the prompting paradigm based on large language models supporting few shot and low resource learning [32]–have substantially pushed the limits of information extraction capabilities [49, 15, 11, 9, 23, 31] for an automated KG construction. Nevertheless, we are facing the next generation of the knowledge acquisition bottleneck. One reason is the ever-increasing demand for different types of knowledge fragments. Knowledge extraction approaches must learn to recognize new types of relations and entities over time. Another reason regards knowledge quality. The automatically extracted knowledge fragments are error-prone and, hence, answers derived from them are at peril of being incorrect.

Instead of waiting for a perfect knowledge base to be established, we propose to embrace imperfection and incompleteness and to design a knowledge-grounded cognitive agent that is aware of these issues and tries to improve over time. The agent should act just as humans do, when facing new and challenging situations. To achieve this, we propose to refine the KG while talking to the users: The cognitive system tries to learn from the humans in the loop (HitL) [54, 57]. For any uncertain or unknown knowledge fragments, the agent will actively ask users for feedback, who are engaged in conversations and are assumed to know possible answers. In addition, the agent's background system will keep track of uncertain or out-dated knowledge fragments based on age, confidence levels of the extraction processes, and feedback from its users. Any feedback provided by the users in this conversational application loop is used to not only refine the KG, but also to improve the knowledge extraction processes [54, 57]. Eventually, the agent's knowledge and the quality of its answers rise while talking to its users. Consequently, when using cognitive agents, refining the KG is a key issue that significantly affects the performance of the entire system [56].

We follow a problem-oriented design science research approach [43] focusing on the first four of its six phases. The following research work is structured according to these phases. First, the problem is identified by defining a problem statement and research question. Their relevance for research is validated based on a systematic literature review [4]. Three key sub-questions obtained using a deductive approach [35] will help to answer the main research question. In the design phase, an artifact is proposed and, subsequently, implemented as a proof of concept. To demonstrate its functionality, a prototype was implemented and evaluated for a university case study [14].

2 Problem Identification and Objectives

As introduced above, the use of cognitive agents is currently frustrating for many users due to their insufficient knowledge. Moreover, to support further use cases, it is necessary to manually enhance existing data sets and fine-tune underlying language models [56]. Since machines are unlikely to have omniscient knowledge soon, the integration of humans into the process is essential to ensure high-quality standards [40, 57]. Although some approaches with human integration may successfully compensate for the missing knowledge, they usually do not enhance the underlying knowledge base [40]. In fact, the necessity to frequently integrate human assistance without learning from the feedback renders these approaches inefficient and expensive. Evidently, this conflicts with the common reasons to implement a chatbot solution. A useful cognitive agent should have the ability to acquire new knowledge based on feedback to continuously optimize itself [29]. The adaption of this concept, often achieved by approaches referred to as *human-in-the-loop (HitL) AI* or *machine learning* (see section 3 for more details), motivates our approach to optimize chatbot performance.

Since most traditional NLP pipelines are not designed to integrate humans in the loop [54], there are many open issues to be addressed in this context [57]. This motivates the research question to be pursued in this paper: *How can a system architecture for a cognitive agent be designed and implemented in which uncertain or unknown knowledge fragments are verified or provided by its users?*

Numerous open questions arise when designing and implementing such a system [54, 57]. Based on the literature, we have identified three fundamental sub-questions that we will address in the paper:

1. *How can existing weaknesses within a KG be identified?* Issues like incompleteness, incorrectness, and inconsistencies among entities and relations between entities in a KG may lead to potential problems [53]. Mostly, these weaknesses result from error-prone extraction processes or from manual user input. Therefore, it is important to develop mechanisms that flag critical objects. They are candidates for which the system may actively seek feedback.
1. *How can the right users to ask be identified?* For any candidate identified above, the system should determine user groups that are suitable to provide feedback that helps to solve the issue. Asking users arbitrarily without justification quickly leads to frustration and rejection of the system [7, 54].
2. *How can the responses be validated and used for KG refinement?* The feedback provided by the users has to be validated, and relevant information has to be extracted and incorporated in the KG. In this context, coping with partially noisy or misleading feedback is a key challenge [26, 54].

3 Background

The following section briefly introduces some basic concepts regarding the topics *human-in-the-loop (HitL) machine learning* and *conversational AI*.

3.1 Human-in-the-Loop (HitL) Machine Learning

Along the machine learning (ML) pipeline, many algorithms are used for a great variety of tasks. An algorithm that exploits human interaction to improve its output is defined as a HitL approach [18]. In particular, approaches may benefit from human assistance if the tasks are complex and deploy large-scale training data with higher quality for better ML model performance [57]. Human interaction is used for different steps, especially for data pre-processing [7], labeling [7] and model evaluation [41].

In HitL learning scenarios, typically there is a trade-off between cost and resources, human feedback quality and model performance. Issues like inferring the truth from multiple inputs, assigning the right tasks to the right persons, reducing the latency of the human feedback, and extending label generation with existing ML techniques have to be addressed [7].

3.2 Conversational Artificial Intelligence

As a subdomain of artificial intelligence (AI), conversational AI combines the use of chatbots and cognitive agents or systems with NLP techniques. The resulting AI-based systems interact with humans in written or spoken language. Currently, there are many reference architectures for chatbots [1]. A conversational AI systems typically comprises three main components: [27]

Natural Language Understanding (NLU) handles the users' input, detecting their intents and recognizing entities. An intent states what a user is trying to achieve with the conversation [44], whereas relevant context information is typically provided by entities such as by people, locations, or organizations [12].

Dialog Management (DM) takes care of the actions of the conversational agent and keeps track of the current state of the ongoing conversation. It can be *task-oriented* or *non-task-oriented* [8].

Natural Language Generation (NLG) is responsible for generating human understandable responses based on the results of the NLU and DM components. Approaches range from simple predefined templates to advanced deep learning models. Current large language models like ChatGPT can generate high-quality conversations [38]. NLG performance is crucial regarding the usability of conversational agents [27].

Fig. 1 shows a common system architecture of a chatbot based on the three components introduced above. The NLU component has a connection to a user interface that enables interactions between users and the bot. The intent and relevant entities are passed to the DM, which may access required additional information via data connectors to prepare content for the response. Based on the DM output, the NLG generates human understandable responses which are passed to the messaging backend.

4 Design and Development of the Artifact

The integration of HitL approaches into the chatbot system architecture shown in Fig. 1 results in changes which are explained in the following section. Design

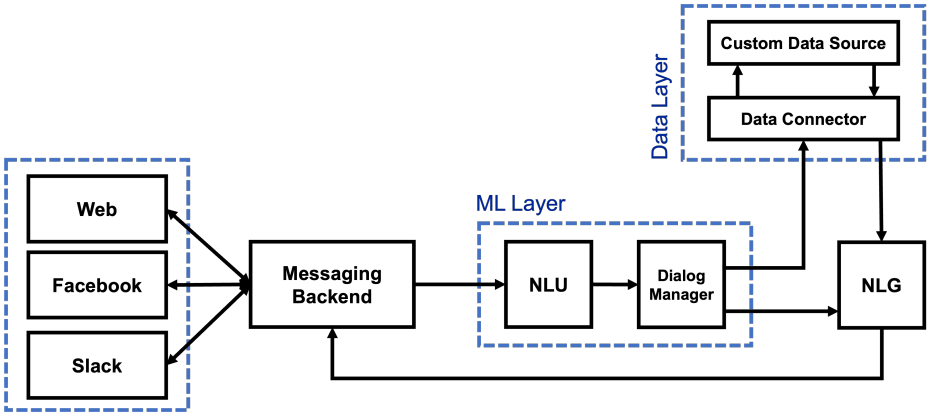

Fig. 1. Common chatbot system architecture based on [3].

decisions for the artifact are derived and explained based on the three sub-questions from the problem identification.

As shown in Fig. 2, the KG is supplemented by another database storing data items relevant for the HitL approach such as the weaknesses identified in the form of a question backlog, the user responses, and further information regarding the users and the system status.

4.1 KG Weakness Identification

As already discussed, the agent's knowledge base may contain incorrect and incomplete data. For our approach, we adapt the layer approach from the multi-hop architecture, which divides data into three layers depending on the processing status: The *bronze layer* stores raw data with low data quality, preprocessed and cleaned data is stored in the *silver* layer, and the *gold* layer stores high-quality data that is used in downstream applications [42]. Often, data in the gold layer has been manually validated [42].

While a common chatbot architecture should focus on high-quality knowledge stored in the gold layer, we focus on data with insufficient quality stored in the silver layer. Instead of manual quality validation, however, we rely on automated quality assessment based on data processing and usage statistics. High-quality objects are automatically transferred into the gold layer, while inferior objects with specific weaknesses remain in the silver layer. To identify weaknesses in the KG, data quality dimensions such as *accuracy, consistency, completeness, timeliness, and redundancy* must be considered in the silver layer [58]. Since the system architecture already with the KG already ensures *consistency* and *uniqueness* (absence of redundancy) [42], our HitL approach focuses on *accuracy, completeness*, and *timeliness*.

The improvement of a KG can be broken down into two main goals: [42]

(a) *Identifying incorrect information:* Incorrect information can be caused by either incorrect entities or relations between two entities in the KG or expiring validity. Therefore, important processing and usage statistics (metadata) is stored for each entity or relation object in the KG in addition to common attributes [21]. Based on the metadata, an aggregated quality score is calculated that reflects an entity's or relation's probability of being valid and current. Based on a provided threshold, relevant objects can be easily selected and stored in the HitL database as a backlog for user queries.
(b) *Supplementing missing knowledge:* Regarding *missing knowledge*, we focus on entities in the KG without any relations to other entities. Several approaches for predicting missing KG objects have been explored [42]. A common approach is to use traversal algorithms to identify gaps in RDF-triples [47].

4.2 Relevant User Identification

To receive high-quality feedback, it is important to ask the right users [7, 54]. Splitting the users into different roles or expert groups helps to reduce human errors and knowledge gaps in advance. [7]. Selecting users involves a trade-off between explicitly asking individual experts and including entire groups to ask for feedback [6]. The exact groups can be derived based on available entities and relevant attributes in the KG. In a corporate context, for example, the entity *department* could be used to create appropriate user groups. User groups with certain roles and properties will subsequently be mapped onto the corresponding questions regarding the candidate objects. Moreover, information that users provide may be used to describe their expertise and help narrow down the most appropriate user group. Finally, considering user statistics and behavior can ensure fast and regular feedback for HitL.

4.3 Answer Validation

Any user feedback received for specific questions should be validated before used to refine the KG. To support this, the users' responses must be collected and stored in a database with unique identifiers for later access. There are methods that *ensure correctness* based on uniqueness. For example, intentions (*affirm* or *deny* are set according to the pattern *yes* or *no*) that categorize answers. This is useful when verifying information. Another way to verify the answers is *truth discovery* from crowd-sourcing [7] where typically the most frequently mentioned answer is considered correct [41]. In case of closed-form questions with predefined answers, these methods work well. For open questions with free text input, validation is more complex since comparison between answers requires response harmonization [33]. In this context, fuzzy search has become a common approach: [36] Divergent answers that are similar in structure can be matched with entities from the KG as keywords. After successful validation, the KG can be refined. For this purpose, there are two main properties to be considered: We can choose to refine the KG *online*, i.e., immediately after input has been processed, or *offline* at specific intervals or events [54]. Regarding the values to

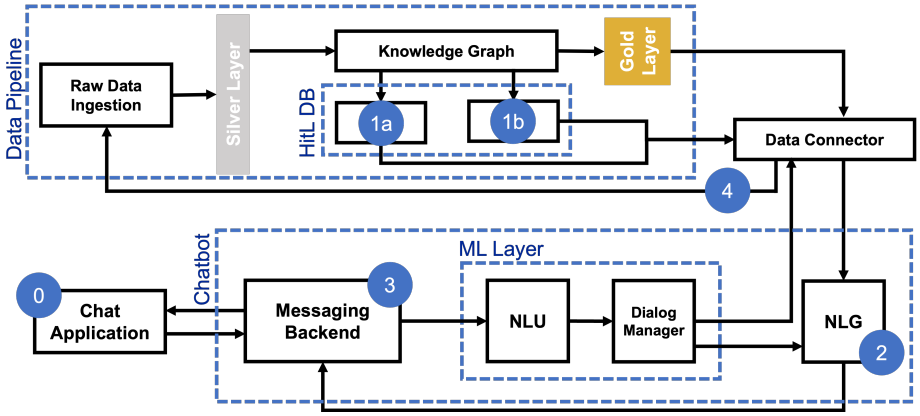

Fig. 2. HitL-based chatbot architecture (own illustration).

be updated, it is important to know whether a new entity or relation has to be inserted or the quality score of an entity or relation has to be updated. To update existing objects, they have to be discovered with regular or fuzzy search first.

5 Demonstration

Finally, the prototypical implementation and development of the architecture is presented based on the components used.

5.1 Implementation

The prototypical implementation of the system architecture is based on the open-source framework Rasa version 3, which is widely used for the development of chatbots in practice. The main component of the Rasa core is the NLU component, which can be trained using various language models. For our prototype, we selected the large German Spacy NLP model. The Rasa actions as well as separate scripts like the initial user contact and various database and KG queries are implemented in Python. The KG is created with ArangoDB. Simple CRUD operations can be made with the query language AQL, which greatly simplifies data extraction and updates. For the storage of relevant HitL data, a light-weight relational database is set up with SQLite. The database contains three tables for the backlog, users, and answers.

To drive adoption of HitL, the chatbot should be shared and used as early as possible via an existing, user-friendly, and widely used chat application. Therefore, instead of the standard command line usage, the Rasa system is directly connected to WebEx Teams as our default chat application via an API.

5.2 Case Study

We evaluate our prototype based on a case study in our university context. Traditionally, universities are organized into faculties with various programs of study. Often, the faculties are quite diverse and interact only to a limited extent [52]. This may put the innovative capacity of universities at risk [28, 52]. Improving technological support is one approach that can help to resolve this issue [50]. The chatbot application presented above makes teaching and research topics and contact information easily accessible for interested parties like faculty members, administrative staff, and students.

In the following, specific implementation details and their implications regarding the three sub-questions introduced in section 2 will be discussed. The basic functionality is shown in Fig. 2. The description references parts of the figure using the numbers given in blue circles.

The chatbot can initiate a conversation via *WebEx Teams* as our default messaging application. If the backlog is not empty and relevant users are active, the system asks them if they are willing to answer questions. Whether a question regarding a particular weakness from the backlog is relevant for a user is simply determined based on affiliation with faculties and our programs of study, assuming the members of a faculty or a program of study are likely to answer questions regarding the respective unit *(step 0)*.

If a user agrees, a question is selected from the backlog in the HitL database. A background process for scanning the KG for weaknesses is scheduled regularly. It fills the backlog with weakness of types (a) and (b) according to section 4.1 as follows. Each object in the KG has a quality score that is initiated based on the confidence of the associated extraction process and the credibility of the source, and will be discounted based on age and frequency of access. In addition, the quality score will be further discounted in case spelling errors are detected. If the quality score of an object drops below a threshold of 0.5, it is stored in the backlog as weakness of type *uncertain (step 1a)*. Positive feedback from users may increase the quality score again. To identify missing knowledge or gaps, a traversal algorithm is applied over all edges of the KG in the prototype [47]. As the KG is a directed graph, this process is triggered from both the regular direction and also the inverse direction. Any objects lacking relevant information are stored in the backlog labeled as *lack (step 1b)*.

The chatbot will continue the conversation and ask the active user a question about a suitable object drawn from the backlog with probabilities complimentary to their quality score. As the set of entity and relation types is fixed, we create suitable question based on simple templates with relevant names or descriptions from the objects which are affected filled into their slots. This approach works well, but obviously lacks variation in the chatbots utterances *(step 2)*.

Any answers from the users need to be parsed and validated. Validation depends on the type of weakness. For possibly incorrect objects, the chatbot asks closed-form questions and has to recognize whether the user states the object is correct or not. For lacking objects, the chatbot uses its NLU component to recognize entities or relations mentioned in the answer *(step 3)*.

If the answer passes the plausibility check, the KG has to be refined. For possibly incorrect objects, their quality score will be updated according to the answer. If an object is confirmed, its quality score is increased. With enough confirmation, high-quality objects can be deleted from the backlog and transferred from the silver to the gold layer. If the user believes the object is incorrect, the quality score is further discounted, which will eventually cause the object to be invalidated or removed from the KG. In case the user provides missing information, we need to distinguish between inserting a missing relation between existing entities and inserting new entities, which may entail new questions. Currently, processing the user feedback is run in offline mode because it is easier to control and inspect the processes. All user feedback is stored in the HitL database to allow for manual inspection and adaptation of the systems in case processes do not work as expected *(step 4)*.

To summarize, the case study demonstrates the positive effect of applying our approach. The users of the system were able to eliminate weaknesses through their input, thus contributing to the refinement of the KG.

6 Related Work

Several research papers discuss the use of *human-in-the-loop (HitL) frameworks* with chatbots or dialog-based systems. Li et al. have built a simulator to conduct a conversation with a chatbot agent based on reinforcement learning [29]. By applying feedback from real humans, the quality of their question-answering system is significantly improved. In a similar approach, Liu et al. have deployed a hybrid approach that introduces a human feedback pipeline on top of an agent-based model to further improve the performance of the agent's capabilities [30]. First, the model learns from its mistakes via an imitation learning from human teaching. Subsequently, human feedback is applied to reward or penalize completed conversations. Karmakharm et al. have conducted a study on human feedback for rumor analysis [24]. Users can commence feedback over a web-based application to further train the underlying machine learning model. Santos et al. have developed a human-supervised process model focusing on the chatbot life-cycle [48]. Humans can interact on several touch-points, like knowledge bases, model training and conversation history. Furthermore, the model assigns specific roles to the chatbot development team. HitL is also used regarding cyber-physical systems. Fernandes et al. discuss a platform concept that combines human interaction with several data sources from mobile devices and sensors [13].

7 Discussion, Limitations, and Future Work

We have presented a new system architecture for the integration of the HitL concept into a chatbot application using a design science research approach. The essential properties of the system have been derived based on existing approaches in scientific publications and enriched by own considerations addressing three

sub-questions that have also been identified as research gaps and, thus, underline the relevance of our contribution [57].

To generate a backlog of candidates with potential weaknesses within a KG on which our chatbot is grounded, it is important to consider each type of error or weakness separately. Within the scope of our case study, both incorrect and completely missing entities and relations were considered as weaknesses. Using quality scores derived from processing and usage metadata and stored with each KG object turned out to be a simple, but very valuable choice that we highly recommend. Further, collecting and using user properties such as affiliation with study programs or research projects to filter the right users to ask for feedback based on matching with fuzzy search and simple mappings onto entity types and characteristics worked well in our small case study. With an increasing number of entity and relation types, more sophisticated approaches will certainly be needed to further differentiate among user groups and roles.

Based on analysis of the user feedback received, we suggest to also consider the type of weakness when seeking appropriate user groups, as it may allow to better anticipate different user reactions. Furthermore, the identification of gaps has only been done by simple traversing and not by complex algorithms. As a result, not all the gaps that could have been inferred based on implicit correlations have been identified.

Despite successful validation of our design choice, it has to be noted that the evaluation is only a first step as it is based on a small case study with only a few entity and relation types and a set of well-known users that are willing to provide answers and restrain from harming the system with inappropriate answers. Consequently, it is not feasible to generalize the findings to general chatbot applications, and further research and evaluation is necessary to be able to generalize beyond our university domain.

Regarding the implementation, further research is also possible. For instance, regarding the identification of weaknesses in a KG, it should be investigated which models can be utilized to identify anomalies as errors or gaps. Furthermore, there is still a lack of research on the evaluation of KG completeness [58]. Regarding the validation of feedback and KG refinement, relevant processes should be further automated. Prompting large language models with specific tasks is a promising solution that should be investigated.

References

1. Adamopoulou, E., Moussiades, L.: An Overview of Chatbot Technology. In: Maglogiannis, I., Iliadis, L., Pimenidis, E. (eds.) Artificial Intelligence Applications and Innovations. pp. 373–383. Springer International Publishing, Cham (2020). https://doi.org/10.1007/978-3-030-49186-4 _31
2. Almansor, E.H., Hussain, F.K.: Survey on Intelligent Chatbots: State-of-the-Art and Future Research Directions, vol. 993. Springer International Publishing (2020). https://doi.org/10.1007/978-3-030-22354-0 _47
3. Ayanouz, S., Abdelhakim, B.A., Benhmed, M.: A Smart Chatbot Architecture Based NLP and Machine Learning for Health Care Assistance. In: Proceedings of

the 3rd International Conference on Networking, Information Systems & Security. NISS2020, Association for Computing Machinery, New York, NY, USA (2020). https://doi.org/10.1145/3386723.3387897
4. vom Brocke, J., Simons, A., Niehaves, B., Reimer, K., Plattfaut, R., Cleven, A.: Reconstructing the giant: On the importance of rigour in documenting the literature search process. ECIS 2009 Proceedings **161** (2009)
5. Brown, T.B., Mann, B., Ryder, N., et al.: Language models are few-shot learners. In: Proceedings of the 34th International Conference on Neural Information Processing Systems. Curran Associates Inc. (2020)
6. Budd, S., Robinson, E.C., Kainz, B.: A survey on active learning and human-in-the-loop deep learning for medical image analysis. Medical Image Analysis **71** (2021). https://doi.org/10.1016/j.media.2021.102062
7. Chai, C., Li, G.: Human-in-the-loop Techniques in Machine Learning. http://sites.computer.org/debull/A20sept/p37.pdf (2020), last access: 2023-02-08
8. Chen, H., Liu, X., Yin, D., Tang, J.: A Survey on Dialogue Systems: Recent Advances and New Frontiers. ACM SIGKDD Explorations Newsletter **19**(2), 25–35 (2017). https://doi.org/10.1145/3166054.3166058
9. Chen, X., Zhang, N., Xie, X., Deng, S., Yao, Y., Tan, C., Huang, F., Si, L., Chen, H.: KnowPrompt: Knowledge-aware Prompt-tuning with Synergistic Optimization for Relation Extraction. In: Proceedings of the ACM Web Conference 2022. ACM (4 2022). https://doi.org/10.1145/3485447.3511998
10. Chowdhery, A., Narang, S., Devlin, J., et al.: PaLM: Scaling Language Modeling with Pathways (2022), https://arxiv.org/abs/2204.02311
11. Cui, L., Wu, Y., Liu, J., Yang, S., Zhang, Y.: Template-Based Named Entity Recognition Using BART (2021), https://arxiv.org/abs/2106.01760
12. Dong, X., Qian, L., Guan, Y., Huang, L., Yu, Q., Yang, J.: A multiclass classification method based on deep learning for named entity recognition in electronic medical records. In: 2016 New York Scientific Data Summit (NYSDS). pp. 1–10 (2016). https://doi.org/10.1109/NYSDS.2016.7747810
13. Fernandes, J., Raposo, D., Sinche, S., Armando, N., Silva, J.S., Rodrigues, A., Macedo, L., Oliveira, H.G., Boavida, F.: A Human-in-the-Loop Cyber-Physical Approach for Students Performance Assessment. In: Proceedings of the Fourth International Workshop on Social Sensing. pp. 36–42. ACM, New York, NY, USA (2019). https://doi.org/10.1145/3313294.3313387
14. Gibbert, M., Ruigrok, W.: The "what" and "how" of case study rigor: Three strategies based on published work. Organizational Research Methods **13**(4), 710–737 (3 2010). https://doi.org/10.1177/1094428109351319
15. Giorgi, J., Wang, X., Sahar, N., Shin, W.Y., Bader, G.D., Wang, B.: End-to-end named entity recognition and relation extraction using pre-trained language models (2019), https://arxiv.org/abs/1912.13415
16. Guu, K., Lee, K., Tung, Z., Pasupat, P., Chang, M.W.: REALM: Retrieval-Augmented Language Model Pre-Training (2020), https://arxiv.org/abs/2002.08909
17. Hogan, A., Blomqvist, E., Cochez, M., D'amato, C., Melo, G.D., Gutierrez, C., Kirrane, S., Gayo, J.E.L., Navigli, R., Neumaier, S., Ngomo, A.C.N., Polleres, A., Rashid, S.M., Rula, A., Schmelzeisen, L., Sequeda, J., Staab, S., Zimmermann, A.: Knowledge Graphs. ACM Computing Surveys **54**(4), 1–37 (7 2021). https://doi.org/10.1145/3447772
18. Holzinger, A.: Interactive machine learning for health informatics: when do we need the human-in-the-loop? Brain Informatics **3**, 119–131 (6 2016). https://doi.org/10.1007/s40708-016-0042-6

19. Hu, L., Liu, Z., Zhao, Z., Hou, L., Nie, L., Li, J.: A Survey of Knowledge-Enhanced Pre-trained Language Models. ArXiv (2022). https://doi.org/10.48550/ARXIV.2212.13428, https://arxiv.org/abs/2212.13428
20. Ji, Z., Lee, N., Frieske, R., Yu, T., Su, D., Xu, Y., Ishii, E., Bang, Y., Madotto, A., Fung, P.: Survey of Hallucination in Natural Language Generation. ACM Computing Surveys (2022). https://doi.org/10.1145/3571730
21. Jia, Y., Qi, Y., Shang, H., Jiang, R., Li, A.: A practical approach to constructing a knowledge graph for cybersecurity. Engineering **4**, 53–60 (2018). https://doi.org/10.1016/j. eng.2018.01.004
22. Kalo, J.C., Fichtel, L., Ehler, P., Balke, W.T.: KnowlyBERT - Hybrid Query Answering over Language Models and Knowledge Graphs. In: Pan, J.Z., Tamma, V., d'Amato, C., Janowicz, K., Fu, B., Polleres, A., Seneviratne, O., Kagal, L. (eds.) The Semantic Web – ISWC 2020. pp. 294–310. Springer International Publishing, Cham (2020). https://doi.org/10.1007/978-3-030-62419-4 _17
23. Kan, Z., Feng, L., Yin, Z., Qiao, L., Qiu, X., Li, D.: A Unified Generative Framework based on Prompt Learning for Various Information Extraction Tasks (2022). https://doi.org/10.48550/ARXIV.2209.11570
24. Karmakharm, T., Aletras, N., Bontcheva, K.: Journalist-in-the-Loop: Continuous Learning as a Service for Rumour Analysis. In: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP): System Demonstrations. pp. 115–120. Association for Computational Linguistics, Hong Kong, China (2019). https://doi.org/10.18653/v1/D19-3020
25. Kejriwal, M.: Domain-Specific Knowledge Graph Construction. Springer (2019). https://doi.org/10.1007/978-3-030-12375-8
26. Kreutzer, J., Riezler, S., Lawrence, C.: Offline reinforcement learning from human feedback in real-world sequence-to-sequence tasks (2020), http://arxiv.org/abs/2011.02511
27. Kulkarni, P., Mahabaleshwarkar, A., Kulkarni, M., Sirsikar, N., Gadgil, K.: Conversational ai: An overview of methodologies, applications & future scope. 5th International Conference On Computing, Communication, Control And Automation (ICCUBEA) pp. 1–7 (2019)
28. Lašáková, A., Ľubica Bajzíková, Dedze, I.: Barriers and drivers of innovation in higher education: Case study-based evidence across ten european universities. International Journal of Educational Development **55**, 69–79 (2017). https://doi.org/10.1016/j. ijedudev.2017.06.002
29. Li, J., Miller, A.H., Chopra, S., Ranzato, M., Weston, J.: Dialogue learning with human-in-the-loop. 5th International Conference on Learning Representations, ICLR 2017 - Conference Track Proceedings pp. 1–23 (2017)
30. Liu, B., Tur, G., Hakkani-Tur, D., Shah, P., Heck, L.: Dialogue learning with human teaching and feedback in end-to-end trainable task-oriented dialogue systems (2018), http://arxiv.org/abs/1804.06512
31. Liu, J., Chen, Y., Xu, J.: Low-Resource NER by Data Augmentation With Prompting. In: Raedt, L.D. (ed.) Proceedings of the 31st International Joint Conference on Artificial Intelligence, IJCAI-22. pp. 4252–4258. IJCAI Organization (2022). https://doi.org/10.24963/ijcai.2022/590
32. Liu, P., Yuan, W., Fu, J., Jiang, Z., Hayashi, H., Neubig, G.: Pre-train, Prompt, and Predict: A Systematic Survey of Prompting Methods in Natural Language Processing (2021), https://arxiv.org/abs/2107.13586

33. Ma, F., Li, Y., Li, Q., Qiu, M., Gao, J., Zhi, S., Su, L., Zhao, B., Ji, H., Han, J.: FaitCrowd: Fine Grained Truth Discovery for Crowdsourced Data Aggregation. In: Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. pp. 745–754. KDD '15, Association for Computing Machinery, New York, NY, USA (8 2015). https://doi.org/10.1145/2783258.2783314
34. Maynez, J., Narayan, S., Bohnet, B., McDonald, R.: On Faithfulness and Factuality in Abstractive Summarization. In: Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics. pp. 1906–1919. Association for Computational Linguistics (2020). https://doi.org/10.18653/v1/2020. acl-main.173
35. Mayring, P.: Qualitative content analysis. Forum: Qualitative Social Research **1** (2000). https://doi.org/10.17169/FQS-1.2.1089
36. Misargopoulos, A., Nikolopoulos-Gkamatsis, F., Nestorakis, K., Tzoumas, A., Giannakopoulos, G., Gizelis, C.A., Kefalogiannis, M.: Building a Knowledge-Intensive, Intent-Lean, Question Answering Chatbot in the Telecom Industry – Challenges and Solutions. In: Artificial Intelligence Applications and Innovations. AIAI 2022 IFIP WG 12.5 International Workshops. pp. 87–97 (2022). https://doi.org/10.1007/978-3-031-08341-9_8
37. Moiseev, F., Dong, Z., Alfonseca, E., Jaggi, M.: SKILL: Structured Knowledge Infusion for Large Language Models. In: Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies. pp. 1581–1588 (2022). https://doi.org/10.18653/v1/2022.naacl-main.113
38. OpenAI: ChatGPT: Optimizing Language Models for Dialogue (2022), https://openai.com/blog/chatgpt/, accessed: 2023-01-26
39. Ouyang, L., Wu, J., Jiang, X., Almeida, D., Wainwright, C.L., Mishkin, P., Zhang, C., Agarwal, S., Slama, K., Ray, A., Schulman, J., Hilton, J., Kelton, F., Miller, L., Simens, M., Askell, A., Welinder, P., Christiano, P., Leike, J., Lowe, R.: Training language models to follow instructions with human feedback (2022). https://doi.org/10.48550/ARXIV.2203.02155 , https://arxiv.org/abs/2203.02155
40. Paikens, P., Znotiņš, A., Bārzdiņš, G.: Human-in-the-loop conversation agent for customer service. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics) **12089 LNCS**, 277–284 (2020). https://doi.org/10.1007/978-3-030-51310-8 _25
41. Parameswaran, A., Sarma, A.D., Garcia-Molina, H., Polyzotis, N., Widom, J.: Human-Assisted Graph Search: It's Okay to Ask Questions. In: Proceedings of the VLDB Endowment. vol. 4, pp. 267–278. VLDB Endowment (2 2011). https://doi.org/10.14778/1952376.1952377
42. Paulheim, H.: Knowledge graph refinement: A survey of approaches and evaluation methods. Semantic Web **8**, 489–508 (2017). https://doi.org/10.3233/SW-160218
43. Peffers, K., Tuunanen, T., Rothenberger, M.A., Chatterjee, S.: A design science research methodology for information systems research. Journal of Management Information Systems **24**(3), 45–77 (2007). https://doi.org/http://doi.org/10.2753/MIS0742-1222240302
44. Qiu, L., Chen, Y., Jia, H., Zhang, Z.: Query intent recognition based on multi-class features. IEEE Access **6**, 52195–52204 (9 2018). https://doi.org/10.1109/ACCESS.2018.2869585
45. Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., Sutskever, I.: Language Models are Unsupervised Multitask Learners (2019), https://github.com/openai/gpt-2
46. Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., Liu, P.J.: Exploring the limits of transfer learning with a unified text-to-text transformer (2019), https://arxiv.org/abs/1910.10683

47. Ranganathan, V., Barbosa, D.: Hoplop: multi-hop link prediction over knowledge graph embeddings. World Wide Web **25**, 1037–1065 (3 2022). https://doi.org/10.1007/s11280-021-00972-6
48. Santos, G.A., de Andrade, G.G., Silva, G.R.S., Duarte, F.C.M., Costa, J.P.J.D., de Sousa, R.T.: A conversation-driven approach for chatbot management. IEEE Access **10**, 8474–8486 (2022). https://doi.org/10.1109/ACCESS.2022.3143323
49. Singh, S.: Natural Language Processing for Information Extraction (2018). https://doi.org/10.48550/ARXIV.1807.02383
50. Sohail, M.S., Daud, S.: Knowledge sharing in higher education institutions: Perspectives from malaysia. Vine **39**, 125–142 (2009). https://doi.org/10.1108/03055720910988841
51. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., Polosukhin, I.: Attention is All you Need. In: Advances in Neural Information Processing Systems. vol. 30. Curran Associates Inc. (2017)
52. Veiga Avila, L., Beuron, T.A., Brandli, L.L., Damke, L.I., Pereira, R.S., Klein, L.L.: Barriers to innovation and sustainability in universities: an international comparison. International Journal of Sustainability in Higher Education **20**, 805–821 (2019). https://doi.org/10.1108/IJSHE-02-2019-0067
53. Verint Systems Inc.: Conversational AI Barometer: Chatbots and Next-Gen AI (2021), https://www.verint.com/resources/conversational-ai-barometer-chatbots-and-next-gen-ai/, accessed: 2023-02-07
54. Wang, Z.J., Choi, D., Xu, S., Yang, D.: Putting Humans in the Natural Language Processing Loop: A Survey. Bridging Human-Computer Interaction and Natural Language Processing, HCINLP 2021 - Proc. of the 1st Workshop pp. 47–52 (2021)
55. Wei, J., Tay, Y., Bommasani, R., Raffel, C., Zoph, B., Borgeaud, S., Yogatama, D., Bosma, M., Zhou, D., Metzler, D., Chi, E.H., Hashimoto, T., Vinyals, O., Liang, P., Dean, J., Fedus, W.: Emergent Abilities of Large Language Models (2022), https://arxiv.org/abs/2206.07682
56. Meyer von Wolff, R., Hobert, S., Schumann, M.: Sorry, I Can't Understand You! – Influencing Factors and Challenges of Chatbots at Digital Workplaces, vol. 47. Springer (2021). https://doi.org/10.1007/978-3-030-86797-3_11
57. Wu, X., Xiao, L., Sun, Y., Zhang, J., Ma, T., He, L.: A Survey of Human-in-the-loop for Machine Learning. Future Generation Computer Systems **135**, 364–381 (10 2022). https://doi.org/10.1016/j.future.2022.05.014
58. Xue, B., Zou, L.: Knowledge graph quality management: a comprehensive survey. IEEE Transactions on Knowledge and Data Engineering (2022). https://doi.org/10.1109/TKDE.2022.3150080
59. Xue, L., Constant, N., Roberts, A., Kale, M., Al-Rfou, R., Siddhant, A., Barua, A., Raffel, C.: mT5: A massively multilingual pre-trained text-to-text transformer (2020), https://arxiv.org/abs/2010.11934