*The Knowledge Engineering Review* (2023), **38**, e8, pp. 1–24
doi:10.1017/S0269888923000085

The Knowledge Engineering Review

CAMBRIDGE UNIVERSITY PRESS

RESEARCH ARTICLE

# Using active learning and an agent-based system to perform interactive knowledge extraction based on the COVID-19 corpus

Yao Yao[1], Junying Liu[2] and Conor Ryan[1]

[1]Lero–Science Foundation Ireland Research Centre for Software, Department of Computer Science and Information Systems, University of Limerick, Limerick, Ireland
[2]NatPro Center, School of Pharmacy and Pharmaceutical Sciences, Trinity College Dublin, Dublin 2, Ireland
**Corresponding author:** Yao Yao; Email: Yao.Yao@ncirl.ie

**Received:** 28 July 2022; **Revised:** 18 July 2023; **Accepted:** 9 October 2023

**Abstract**
Efficient knowledge extraction from Big Data is quite a challenging topic. Recognizing relevant concepts from unannotated data while considering both context and domain knowledge is critical to implementing successful knowledge extraction. In this research, we provide a novel platform we call Active Learning Integrated with Knowledge Extraction (ALIKE) that overcomes the challenges of context awareness and concept extraction, which have impeded knowledge extraction in Big Data. We propose a method to extract related concepts from unorganized data with different contexts using multiple agents, synergy, reinforcement learning, and active learning.

We test ALIKE on the datasets of the COVID-19 Open Research Dataset Challenge. The experiment result suggests that the ALIKE platform can more efficiently distinguish inherent concepts from different papers than a non-agent-based method (without active learning) and that our proposed approach has a better chance to address the challenges of knowledge extraction with heterogeneous datasets. Moreover, the techniques used in ALIKE are transferable across any domain with multidisciplinary activity.

## 1. Introduction

Since the outbreak of the COVID-19 pandemic, scientists globally have been engaged in surmounting this unprecedented crisis and curbing the spread of the virus. There has been an enormous amount of research that has been published or updated daily. However, many of these endeavours and much of their fruitful results have not been extensively exploited by the research community due to the immense amount of related literature information.

One of the main reasons for this is that researchers need more efficient automatic tools to process, verify and integrate the enormous amount of data created and then subsequently extract related knowledge from these data to help their studies. Therefore, to help researchers cope with the Big Data deluge of COVID-19, we urgently need to develop automatic tools to extract knowledge from heterogeneous datasets and properly integrate such knowledge. The process of knowledge integration can be divided into two steps. The first is to identify and extract the relevant knowledge from the corpus datasets, while the second is to use that extracted knowledge to establish a comprehensive knowledge base which is then made accessible to researchers and corresponding programs. In this paper, we focus on how to extract context-related knowledge efficiently and automatically from massive datasets. Efficiently extracting

**Cite this article:** Y. Yao, J. Liu and C. Ryan. Using active learning and an agent-based system to perform interactive knowledge extraction based on the COVID-19 corpus. *The Knowledge Engineering Review* 38(e8): 1–24. https://doi.org/10.1017/S0269888923000085

relevant knowledge from datasets is necessary for the development of a knowledge integration framework. Therefore, our work in this paper has the potential meaning to improve contingent knowledge integration in Big Data.

In terms of this study, knowledge extraction is the retrieval of knowledge from structured (i.e., relational databases, XML) and unstructured (i.e., text, documents, images) sources. The extracted knowledge needs to be machine-readable and can be directly used for further inferencing and modeling (Unbehauen *et al*., 2012). When tackling Big Data, knowledge extraction is usually implemented by automatic data mining programs on computational platforms (Cheng *et al*., 2018). The challenge of implementing automatic knowledge extraction with highly professional and interdisciplinary topics such as COVID-19 is essentially one of context-related pattern recognition based on specific expertise and concept extraction considering the context (Che *et al*., 2013; Weichselbraun *et al*., 2014). Taking the research of the COVID-19 pandemic as an example, due to the complexity of this viral pandemic, many relevant investigations about COVID-19 need to collect interdisciplinary knowledge from multiple sources and study their topics based on a comprehensive understanding across different domains. However, much-published information about COVID-19 tends to come from relatively specific areas rather than representing some inherent correlation between different domains separately discussed in each paper. (Costa *et al*., 2020). Verifying and extracting related concepts across several domain contexts and then reorganising them with appropriate expertise from different datasets becomes critical in determining the efficiency of knowledge extraction here.

Automatically perceiving and annotating the context in knowledge extraction usually are based on two main approaches: one is using predefined rules or schema (Nadgeri *et al*., 2021; Hürriyetoğlu *et al*., 2021), and another is machine learning-based classifiers (Kraljevic *et al*., 2021; Chen *et al*., 2022), requiring intensive manual effort and considerable training data, respectively, to support the establishment of a suitable model. These requirements mean neither approach may be feasible nor too expensive to achieve when we face massive and unorganized literature data with diverse contexts. Furthermore, the final model is usually based on a particular domain and training data. This means the predefined schema or learning models don't always work well when tasks are related to a new interdisciplinary background. In other words, the re-usability of generic experience for performing similar tasks has not been sufficiently explored. Such difficulty limits the potential of knowledge integration in Big Data and dramatically raises the cost of knowledge extraction when the data scale increases.

We provide a novel platform we call ALIKE (Active Learning Integrated with Knowledge Extraction) that overcomes the challenges of context-related pattern recognition and concept extraction, which have impeded knowledge extraction in Big Data. Using multiple agents, synergy, reinforcement learning, and active learning, we propose a method to extract related concepts from unorganized data with different contexts. Instead of learning the entire pattern from data directly, we propose to develop multiple agents that can learn identifiable features of patterns from experts before using machine learning-based models to reassemble these features on a common semantic platform for recognizing new patterns with less training data. Through this proposed approach, the system can extract fragmented knowledge (i.e., concepts) from discrete datasets. Experts will be intensively involved in our approach to give a semantic definition of these features and to label the most typical patterns with predefined feature tags through active learning. Such customized user-validated features can more efficiently represent the expertise under different contexts, increasing the re-usability of extant knowledge. The knowledge transfer from experts will significantly reduce the labeling and training workload and enhance the efficiency of pattern recognition (Settles, 2010; Springer, 2016; Kulikovskikh *et al*., 2020). Finally, the separated and concreted concepts of different datasets will be extracted by multiple individual agents and annotated with a particular context that is inherent in the corresponding data, whereby the system can integrate knowledge into a global knowledge base. With this framework, we also achieved a good integration between open and closed Information Extraction and solved the problem of lacking the schema (Dutta *et al*., 2013). The knowledge in the knowledge graph (KG) is dynamically updated and guides further knowledge extraction. This provides the necessary schema for knowledge extraction and helps to interpret the result.

In our experiments, after a brief training period, the agents in our system can learn how to detect and identify features in a similar way to experts. These agents can also detect the corresponding patterns on a larger scale by collaboration. Our platform's detection of predefined related concepts has been tested and shown to compare favourably with default approaches (keyword extraction based on term frequency–inverse document frequency (TF-IDF) and agent-based extraction without reinforcement learning). Based on the correlation with given semantic patterns, our result has an evident improvement in quality. By reorganizing reusable features of individual concepts, the system can automatically learn further patterns more easily with limited training data in subsequent learning processes. Meanwhile, by introducing context features, our approach can recognize a particular pattern by considering its corresponding context. This is reflected in the result of our platform, as it can more efficiently distinguish the inherent concepts from different papers examined in the experiments. This advantage allows our proposed approach to better address the discussed challenges of knowledge extraction with heterogeneous datasets.

We will discuss this research's relevant work and background in Section 2. Section 3 focuses on the details of the methodology of this framework. At last, we provide some preliminary results of this research in Section 4 and then conclude the impact of this research on the relevant future work.

## 2. Related work

The task of successfully implementing knowledge extraction in our framework relates to several existing research areas, including knowledge acquisition, concept identification, pattern recognition, active learning, and agent-based knowledge processing. This section briefly reviews this related research and draws distinctions between previous work and ours.

### *2.1. Knowledge acquisition*

Knowledge acquisition is the process used to define the rules and ontologies required for a knowledge-based system. The phrase was first used in conjunction with Expert Systems to describe the initial tasks associated with developing an expert system: finding and interviewing domain experts and capturing their knowledge via rules, objects, and frame-based ontologies. Due to the sophisticated context (i.e., the ambiguity of a concept with different modalities) and massive ontology entities that emerged from Big Data, knowledge acquisition became increasingly important to develop the knowledge and implement the necessary engineering processes on Big Data (Kendal & Creen, 2007). Currently, there are two main approaches to knowledge acquisition.

The first is to use natural language parsing (NLP) and generation to facilitate knowledge acquisition. By analyzing expert documents, an NLP program can manually or automatically initialize the ontologies for constructing knowledge (Potter, 2003; Gyrard *et al*., 2018). Another approach to knowledge acquisition is a reuse-based approach. Knowledge can be developed in predefined ontologies that conform to standards, such as the Web Ontology Language (OWL). In this way, knowledge can be standardized and shared across a broad community of knowledge workers. One example domain where this approach has been successful is bioinformatics (Goble & Stevens, 2008).

In our knowledge extraction process, we utilize both approaches to produce ontologies. Firstly, we predefine the essential ontologies and store them as reusable semantic concepts that can be accessed by autonomous agents; then, we analyze the related literature with the help of computational agents. Through an active learning process, agents constantly integrate the knowledge of highly skilled experts into the predefined semantic concepts and inspire experts to extend the ontologies based on the context of data analysis. The system finally uses such knowledge that exacts from the data and experts to indicate the autonomous agents for optimizing their performance in further tasks. Such a hybrid knowledge acquisition approach allows our system to have high-quality reusable knowledge while still being capable of extending such knowledge automatically in data analysis processes.

In this paper, context is represented as a combination of semantic concepts that the interaction of agents will establish. For example, we can define a concept as a virus transmission model called A. This concept may have a mutual relationship with other concepts such as temperature B, climate C, population E, and age F. Each relevant concept will be defined with a corresponding agent based on expert knowledge. When one of the concepts has been activated by finding its coherent features or patterns, the agent of this concept will check if any relevant concepts have also been activated simultaneously. In other words, if the temperature B concept has been activated in a paper because there is text discussing that particular temperature factor, the agent will check if the virus transmission model A concept has also been activated. If not, the agent of virus transmission model A concept will check other relevant concepts like climate, population, and age to see if all of them are activated. If all these concepts have been activated simultaneously, we say a particular context for virus transmission model A has formed. Each concept may have multiple relevant contexts, and each of them could have different coherence. Coherence means the measurement that describes how coherent two concepts are under a certain context. For example, temperature B and climate C could also have other common contexts related to climate change, and their coherence in that context could be different. When the corresponding context is formed, the agent may activate the concept (i.e., virus transmission model A), and the ratio is based on coherence. All these relevant concepts for virus transmission model A are necessary to form a context, and the identification process of the context is based on the interaction (connection or disconnection) of corresponding agents.

### *2.2. Concepts identification and pattern recognition*

Concepts usually refer to abstract ideas. In our research, a concept has been regarded as a customized semantic label that refers to a particular part of knowledge. Agents can access the required knowledge from the knowledge base based on a given concept. Experts will also use concepts to refer to the particular knowledge in their interaction with agents. To identify a concept, users need to define the relevant features or conditions related to it when they are defining the concepts. We call the related pattern for a concept the combination of these features and conditions. The pattern can be the particular network topology between other concepts, a set of data features, or any ontology objects that are accessible by agents. In other words, in our framework, a pattern is the predefined verification routine for an agent to identify a concept. Pattern information is given by the expert when the concept has been defined, but it can be optimized later by experts or machine learning models.

The identification of related concepts is a crucial step in knowledge extraction. Concepts are basic components of knowledge, and the combination of concepts and their connected relations constitute the entire knowledge. To extract knowledge during the data analysis, a system often needs to efficiently recognize the corresponding patterns of the given concepts that have been defined in knowledge acquisition. Our system uses various concept identification and pattern recognition methods to identify these concepts and their relations from data analysis.

The study of concept identification and pattern recognition is concerned with the automatic discovery of regularities in data through computer algorithms and using these regularities to take actions such as classifying the data into different categories (Bishop, 2006). Much previous research focuses on extracting concepts and knowledge from web-based data. Relevant frameworks such as DBPedia in Jens Lehmann *et al*. (2015), YAGO in Suchanek *et al*. (2007), Probase in Wentao and Wu (2012), ConcepT in Liu *et al*. (2019) have been well tested in practical tasks and shown good efficiency. Our research aims to continually improve knowledge extraction based on these previous studies by introducing the customized context and the relevant expertise during the analysis. We believe these added elements can facilitate the effectiveness of those given methods in knowledge extraction. In our study, we focus on using agents to collect expertise from experts to better support the knowledge extraction process rather than developing new algorithms. The system decomposes complicated knowledge into several simpler predefined concepts, each identified by a corresponding agent. The agent learns the necessary features and models from experts to recognize the related patterns of the corresponding concept from

data. The agents also interact with each other in a hierarchical structure and collectively represent comparatively complicated knowledge. Through this, our proposed framework allows the particular context and expertise to be sufficiently involved in knowledge extraction and evolved by the feedback of experts. A similar approach to unsupervised concept identification has been introduced in the previous research (Zhukova *et al*., 2021), and it showed a very good potential for context-aware recognition. Moreover, in our approach, we extract such knowledge of context-aware recognition and validate the knowledge with expert users. The validated knowledge will be stored in the knowledge base and will be available to be reused in other applications in the future.

### *2.3. Active learning*

As discussed in the previous section, in knowledge extraction, the agents are supported by the expertise that comes from human experts. Agents use such knowledge to identify the corresponding patterns or concepts that are included in data. Active learning is the main method that has been used to convey the expert's knowledge to the autonomous agents.

Active learning is a special case of machine learning in which a learning algorithm can interactively query a user (or some other information source) to label new data points with the desired outputs (Settles, 2010; Rubens *et al*., 2016; Das *et al*., 2016). For active learning, information sources such as experts are also denoted as 'teacher' or 'oracles.' This is useful because, in data analysis, there are situations where unlabeled data is abundant, but manual labelling is too expensive to be applied to all of it. In such a scenario, learning algorithms can actively query the expert for labels.

The main advantage of active learning is that it can effectively reduce the training data for learning a concept while optimizing the learning process by adding pertinent expertise. To avoid being overwhelmed by uninformative data in active learning, we use agents to help experts select our approach's most typical and valuable data for labeling. In this way, active learning is used in data analysis to help experts annotate the potential concepts in data. The whole active learning process can be divided into two steps.

Initially, the system will analyze the training data (i.e., literature data, text-based information). During this data analysis, active learning agents will ask the experts to define or discriminate any suspect concepts that are related to the given knowledge or model. The experts can initially customize such knowledge or model based on their interests and then embed this predefined knowledge into particular agents. This allows the experts to modify the embedded knowledge based on feedback during data analysis. Through the given annotated patterns from experts, agents can learn the knowledge from data more efficiently. Once the agent has learned enough knowledge, it will be able to identify concepts during data analysis in the same way experts did.

In a KG, knowledge is represented by semantic triples, each consisting of three parts: subject, predicate, and object. Each subject or object can be regarded as a conceptual node on KG corresponding to a particular concept, while the predicate describes the relationship between the subject and object. Identified concepts and their relationships will be used to activate the corresponding conceptual nodes and predicate on KG for completing the knowledge extraction. In our Active Learning Integrated with Knowledge Extraction (ALIKE) platform, active learning processes provide interactive dialogue between the system and experts when it is necessary. Through this, the knowledge of the system can be dynamically updated following the changes in data or constantly improved by the experts.

### *2.4. Agent-based modelling for knowledge graph development*

There has been much investigation into the use of multiple agents (sometimes using swarm robotics Coppola *et al*., 2019) to construct a collective system that combines knowledge representation and reasoning methods to acquire and ground knowledge (Rosenthal *et al*., 2010; Wei & Hindriks, 2012; Tenorth & Beetz, 2017). The combination of semantic techniques and robotics has shown great potential in knowledge processing.

In our study, a multiple-agent-based model has been developed to integrate fragmented knowledge from discrete datasets into a global KG. A multi-agent system (MAS) will select the corresponding concepts labelled by agents in data analysis to construct a global KG about the pandemic of COVID-19. Each selected concept will generate a corresponding agent as their digital twin in simulation, and, based on the context of data and the knowledge of experts, these digital agents will interact with each other and create semantic triples. Once an agent finds the related target (other related concepts in the context) in simulation, its corresponding concept will be written into the semantic triple as an entity or edge (relations). In our framework, knowledge has been used to describe the relationship between concepts, and it is stored in a hierarchy KG, meaning each part of the KG can be represented as a sub-KG (sub-KG). Agents can access their corresponding knowledge and convert it to rules to guide their behaviour. For example, the meta-knowledge about an edge (the relationship between concepts) can also be an embedded sub-KG. Each part of the domain-specific knowledge may have its own corresponding concepts, whereby the CI agents can recognize it. The relevant content of a concept, such as parameters and possible relations, is stored in JSON format files and linked with this concept. CI agents read the relevant sub-KG and then convert the content of relevant concepts as internal knowledge to conduct the agent's behaviours. All in all, any knowledge will be represented in semantic triple format and stored in a hierarchy structure. Each completed semantic triple will become the basic component of the global KG. This construction is based on the bottom-up cooperative behaviour of multiple agents, and the final knowledge base will be stored with a semantic data format as a KG. The KG will gradually self-organize dynamically by constantly accumulating semantic triples from simulation.

## 3. Methods and models

Our framework discussed in this paper is an agent-based system that aims to perform interactive knowledge extraction from heterogeneous datasets based on the expertise of users. This section will elaborate on and discuss the main components of this framework, respectively. First, it will briefly discuss the approach of using the agent-based model to represent knowledge in Section 3.1. In Sections 3.2 and 3.3, we give more details about how to use different agents to represent, identify and validate knowledge in various scenarios, and it elicits the description of the two main types of agents in our framework. Section 3.4 discussed how the reinforcement learning method had been applied to the agents of our framework in general. In the final Section 3.5, we try to combine all these discussed components and give a comprehensive view of how our framework works in a knowledge extraction task.

### *3.1. An agent-based model for knowledge representation*

In our study, knowledge refers to the representation of concepts and relations between concepts that describe the particular objects in data. For example, we can use a simple combination of concepts such as sky and blue to represent the colour of the sky. By extracting the relevant concepts and the relations between concepts from data analysis, we derive knowledge about a particular object from data, such as the colour of the sky. To extract knowledge better, it is necessary to identify the most pertinent concepts efficiently during data analysis. To achieve this goal, we utilize multiple agents to implement the concept identification task, respectively and specifically. By this dispersive approach, the complexity of identification can be effectively reduced in extraction. In addition, the system also adopts embedded reinforcement learning to facilitate the learning process of CI agents in order to improve identification. Our research separately tested the effect of the dispersed agent-based approach and the embedded reinforcement learning through a series of experiments, and the result have demonstrated an improvement in each of these settings.

### *3.2. Individual concept identification agent*

In the previous section, we discussed previous work and challenges related to concept identification. Extracted concepts must reflect particular domain expertise and users' cognitive perspective while remaining consistent with the context and given data. Traditional methods can usually process data efficiently but may ignore the context and necessary expertise during extraction. Imposing manual labelling can compensate for the deficiency of context and expertise during knowledge extraction to a certain extent. Still, the manual effort usually is too expensive to apply to Big Data analysis, and the given knowledge will not be easy to be generated automatically. This may lead to the update of knowledge being inefficient and knowledge becoming obsolete.

In our research, we use an agent-based approach to help people overcome the challenges in knowledge extraction. Based on the concept identification step, we applied an agent known as a Concept Identification (CI) agent to implement the tasks. A single CI agent is designed to identify latent patterns and extract related knowledge from data. It collaborates with other agents (as outlined below in the section titled 'Knowledge Extraction') to create a more efficient pattern recognition within a particular context. Generally, a CI agent will read knowledge (models) from KG and use the corresponding predefined knowledge to detect the latent data patterns corresponding to other known concepts. Several activated CI agents can interact with each other and form a particular context. The adapted single CI agent will finally find its niche in the context through this interaction. All CI agents can collectively establish knowledge of the KG by reassembling their corresponding concepts into various semantic triple expressions. The collaboration of CI agents can efficiently solve two main problems we often confront in knowledge extraction. One is an ambiguity caused by various contexts, and another is identifying latent associations between different domains or data sources. In the first case, the situation usually is that different terms in different domains may have the same meaning, and the same terms in different domains may have different meanings. For example, when people are reading a list, the number '3' may mean the third item on the list. Still, it could also be interpreted as a particular parameter value within a different context. Another example could be that 'the next line' means 'the third line' in the same paragraph. In the second case, we hope to be able to identify and extract the latent connection between concepts from different domains. For example, when we read a correlation between temperature rising and the contagion of diseases, we may hope to connect this concept with another concept of season change in the different domains across different reference documents. Through the interaction of agents, we can address the above problems. In our framework, when a concept such as 'temperature rising' has been detected, an existing agent will be activated and connected to this concept. This agent will continue to broadcast itself to other activated agents and find the relevant ones based on its possible relevant knowledge from the knowledge base. Assuming knowledge defines 'temperature rising' should be related to 'contagion of diseases' and 'seasons change,' the new agent will try to find both other agents there. If both these agents are activated, the new agent may set a connection between them and change their status into a more specific context. For example, the agent of 'seasons change' will be more specific to its knowledge related to epidemic prevention. Each new connection between agents will trigger both agents to do such a developmental process based on the previous common context. Such developmental behaviour driven by interaction is the main output of CI agents. Each agent can access the Knowledge base and finally represent a particular concept or context in the framework. When a CI agent becomes specific enough, it will be rewarded, and the connected network of such agents will be updated as extracted knowledge. With this, the interaction of multiple agents will be able to identify the current context more accurately and efficiently and introduce important concepts across different domains and data sources.

Figure 1 below gives a generic functional role of the CI agent in the framework. The system predefined a list of rules for each given concept to recognize its occurrence from the possible datasets. These initial rules include the relevant keywords (represent abstract concepts) and relations that the expert specified to identify the occurrence of the concept. Each of these rules can be represented as a semantic triple and we call these rules the identified data patterns. The experts will provide this knowledge when they create the new concept in the system, and the CI agent can use this knowledge to identify

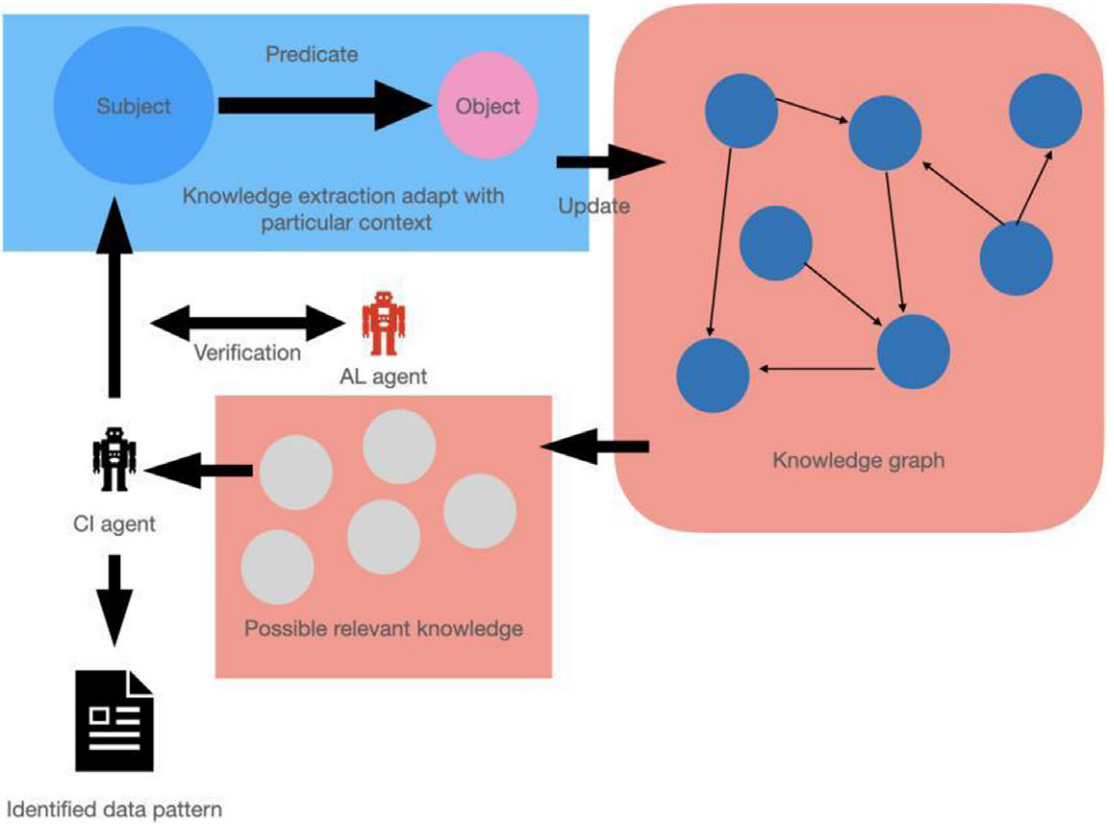


***Figure 1.*** *The generic functional role of a CI agent in the corresponding knowledge extraction.*

its corresponding concept afterward. This knowledge will lead CI agents automatically connect other activated concepts that have been identified in the same dataset based on the specified context conditions. Though connecting other given concepts, CI agents aim to be more specific and continually optimize the given identified data patterns in a particular context. This developmental process is under the surveillance of the AL agent. The AL agent will catch any predefined conflict in extracted knowledge and then send them to experts by query. The communication frame can be defined or adapted by experts manipulating the metadata templates. At the moment, the system provides fixed-format text prompts, multiple selections, and priority ratings as the main style of communication, but it can be easily extended by adding more metadata templates by users. The KG include these metadata as a part of the knowledge. Expert users can manually review or optimize any part of KG through a provided user interface or occasionally check the corresponding part of KG based on the query from AL agents.

### *3.3. Active learning agent*

In our research, we embed active learning into multiple computational agents. The potential users of this system need to predefine the related knowledge domains and the corresponding experts. According to different experts and their specialized knowledge, the system will assign corresponding agents to interact with these experts. This research's agents interacting with experts are active learning agents (AL agents).

The AL agent's essential functionality is to convert experts' specific knowledge and experience into a particular computational model, which is then stored in the KG or accessed by other computational

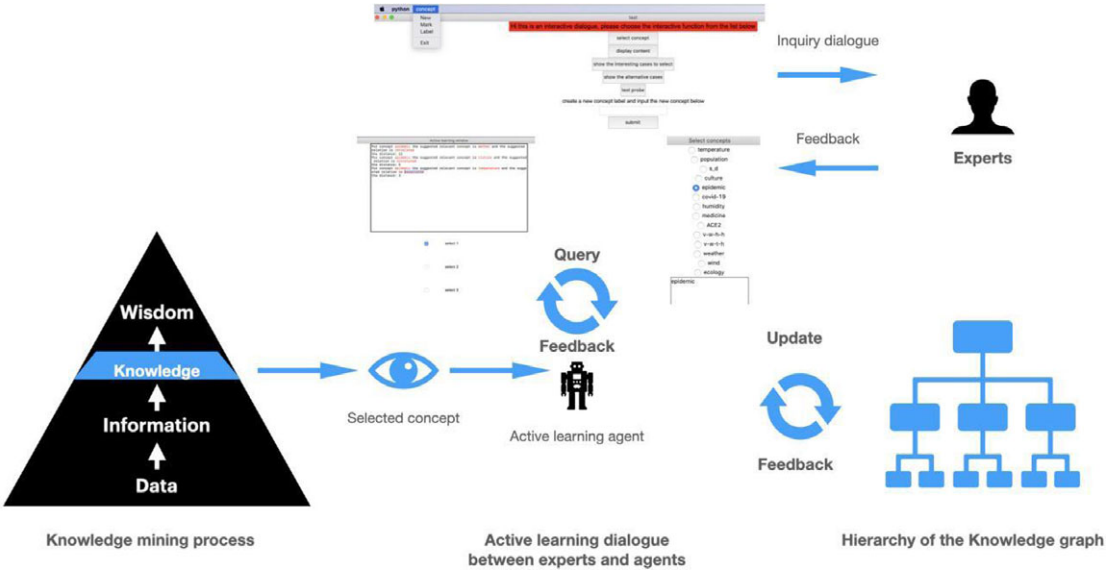


***Figure 2.*** *Illustration of the active learning dialogues for domain-specific expert users.*

agents. Such digitized knowledge will guide other agents to learn the target objectives more efficiently from limited annotated training data. We developed each AL agent for learning particular feature patterns and the corresponding knowledge models and assigned specific AL agents with particular concepts. Each concept has its own feature patterns that can be used to identify this concept from data. The concept also has a knowledge model reflecting the coherence to other relevant concepts. Unlike CI agents, AL agents do not focus on identifying the related patterns of concepts but on improving these patterns based on the user's interests. For example, the concept 'warm' has one feature pattern that indicates it needs a temperature record above 20 Celsius. CI agent will activate the 'warm' concept when the pattern has been found in the data. Still, the AL agent will ask the expert if there is a need to redefine that pattern when the activated 'warm' concept encounters a conflict based on its current knowledge model (i.e., encounter incompatible concept 'cold'). During the training phase, when a related concept has been identified, the corresponding AL agent could also be activated and contact the relevant experts for feedback or suggestion based on the current unknown context. There is an example shown in Figure 2 about how such communication has been implemented. In the framework, different users can access different layers of knowledge, and Figure 2 shows one example that domain-specific experts access the corresponding domain knowledge in the system.

During the knowledge extraction phase, the system requires domain-based expertise to help to select the target concepts and evaluate the quality of extracted knowledge. One of the main tasks of an AL agent is to apply its given expertise to these functions. Each of the AL agents involved in these processes corresponds to a particularly given rule that describes part of the measure logic of experts to identify the relevant knowledge. For example, in terms of modes of transmission of the virus causing COVID-19, one exemplary extracting rule from experts could be defined as that the relevant statement needs to discuss at least one of the predefined transmission ways (experts could specify these predefined transmission ways) and mention COVID-19 virus simultaneously in the same paragraph.

Such rules are stipulated by experts and may be altered or extended during active learning. In the active learning process, when an AL agent finds a case that matches its rules close enough (based on a metric delineated in the section on relation verification), it will query experts for feedback or confirmation. Based on the experts' feedback, AL agents will evolve their behaviour (i.e., more sensitive or insensitive to similar cases) while updating the extracted knowledge as experts suggested. The AL agents can collectively interact with each other in the task to create more sophisticated rules, and the

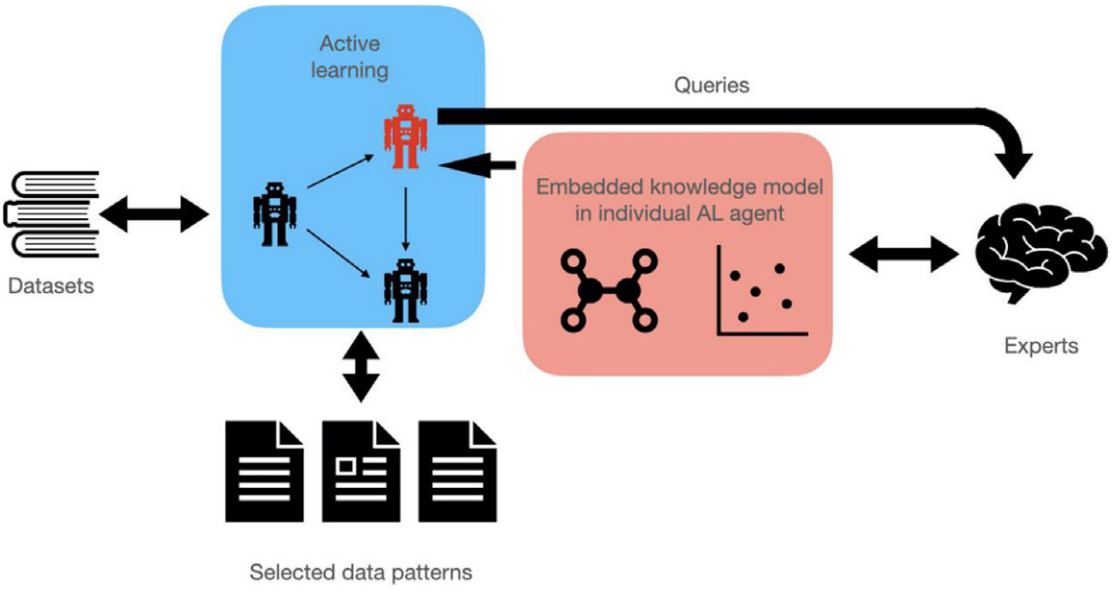


***Figure 3.*** *The interaction between AL agents and experts.*

feedback from experts will impact the relevant agents in a hierarchical manner. As Figure 2 shown, it is an example of an expert interacting with the AL agent at a domain-specific knowledge level. AL agents also can interact with different experts at different levels. For example, data analysts also can interact with the corresponding AL agents at the data level to optimize the knowledge of data integration or data format.

Figure 3 shows the generic pipeline of an AL agent about how to interact with other AL agents and experts. It takes the queries from its lower-level AL agents and integrates its own output into these queries. The final query will include all output of relevant agents and send it to the corresponding user.

### *3.4. Embedded reinforcement learning*

Based on the interaction between users and other agents, we applied the reinforcement learning method to optimize agents' behaviours. For both CI agents and AL agents, we applied embedded reinforcement learning to optimize the behaviour model in interaction. Reinforcement learning is based on Markov decision process (MDP), which is a discrete-time stochastic control process. MDP provides a mathematical framework for modelling decision-making in situations where outcomes are partly random and partly under the control of a decision-maker. Reinforcement learning aims for the agent to learn an optimal or near-optimal policy ($\pi$) that maximizes the reward function. Different agents have different reward functions based on the essentials of the task. CI agents aim to connect with as many relevant concepts as possible during the knowledge extraction and choose the efficient differentiation process to specify the context. By connecting with other concepts, the agent will better know the context for its corresponding concept, so each concept connection with itself or its connected concepts will reward the corresponding agents. To AL agents, the reward function is based on feedback from users. The acceptance of queries or updated knowledge will reward all relevant AL agents. Take a CI agent, A, as an example and assume it can only connect to the other two CI agents, B and C, based on the given knowledge. The observation space would be its all-possible connection status: A, A+B, A+C, A+B+C (based on its predefined relevance with other concepts), and the action space would be tuning operations (enhance or suppress the ratio for making a connection) to each of its relevant concepts. For CI

agents, the task is to represent their own concept and connect it with their relevant knowledge extraction concepts. Each time the agent connects with other concepts, the connection status will be updated (i.e., the connected concept number +1, the status for the connected concept becomes positive). During the knowledge extraction, each pair of relevant concepts has a ratio to decide if these two encountered concepts will connect to each other. That ratio can be changed by the tuning operations of agents. The policy mode of an agent is a matrix that specifies the possibility of actions under each status. The initial policy of the agent is based on the relevant predefined knowledge of KG that stipulates the relevance priority between concepts.

In our research, we regard embedded reinforcement learning as a form of model-based learning. It means the system will use predictive models to evaluate agents' actions during the learning process. These predictive models are part of the knowledge in the system. Our reinforcement learning also follows the MDPs to interact with the environment. The relevant components in a typical MDP are as below.

1. **S** represents a set of environment and agent states;
2. **A** represents the action set of agents;
3. **t** represents the time step;
4. **R**(s-s') represents the reward obtained by taking action in state s to state s';
5. $\boldsymbol{\gamma}$ is the attenuation factor ($\boldsymbol{\gamma} \in [0, 1]$) that determines the importance of future rewards;
6. $\pi$ is the following policy representing the agent's action selection model as a map.

Such a model of MDP will initially be generated from knowledge in KG, and the model can be updated based on the feedback in the task.

With the given model, we will be able to calculate the value of policy $\pi$:

$$\mathbf{V}_{\pi}\,(\mathbf{s}) = \mathbf{E}\,[\mathbf{R}] = \mathbf{E}\left[\sum_{\mathbf{t}=0}^{\infty} \gamma^{\mathbf{t}}\mathbf{r}_{\mathbf{t}} | \mathbf{s}_0 = \mathbf{s}\right]$$

With a learning process, the agent will find the optimal policy and put it in the KG. When the model has been changed based on the experts, agents will also re-learn to update the policy. The general purpose of reinforcement learning here is to make agents can connect with as many concepts as possible in interaction while keeping less inconsistency in its updated result. The possible connection options and actions constitute the learning model, and the loss function is the absolute loss function.

In our system, as discussed above, we decomposed concepts from knowledge into the particular context status and defined the particular behaviour of agents as actions. The rewards function depends on experts' feedback or the connection pattern between concepts. In other words, it can be various for different kinds of agents. Regards AL agents, the purpose is to provide the most typical queries for experts better annotate the concepts. Still, for CI agents, it is to extract new knowledge with more possible concepts. Agents accessing their relevant knowledge model in KG and using it to define their rewards functions and agent actions, respectively. For example, if the expert accepts a suggestion from an AL agent, a CI agent establishes a collaboration with other agents for extending semantic triples of knowledge, or if an agent detects a new concept during the data analysis, the status of related agents could change and generate corresponding rewards to those agents. In these examples, each agent has a corresponding status and action list that provides the agent with a model to learn the optimal policy. Such a mode is embedded in the corresponding knowledge. By running a model-based reinforcement learning, multiple agents can optimize their policies respectively in the tasks. In addition, agents can also constantly access the KG to update their models and find the most optimal policy based on the new knowledge model.

All agent models are maintained in KG, and AL agents can give suggestions to experts for updating these models based on the result of data analysis. With the collaboration between AL agents, CI agents and experts, the whole knowledge extraction model can be dynamically evolved to maximize the efficiency of user expectations.

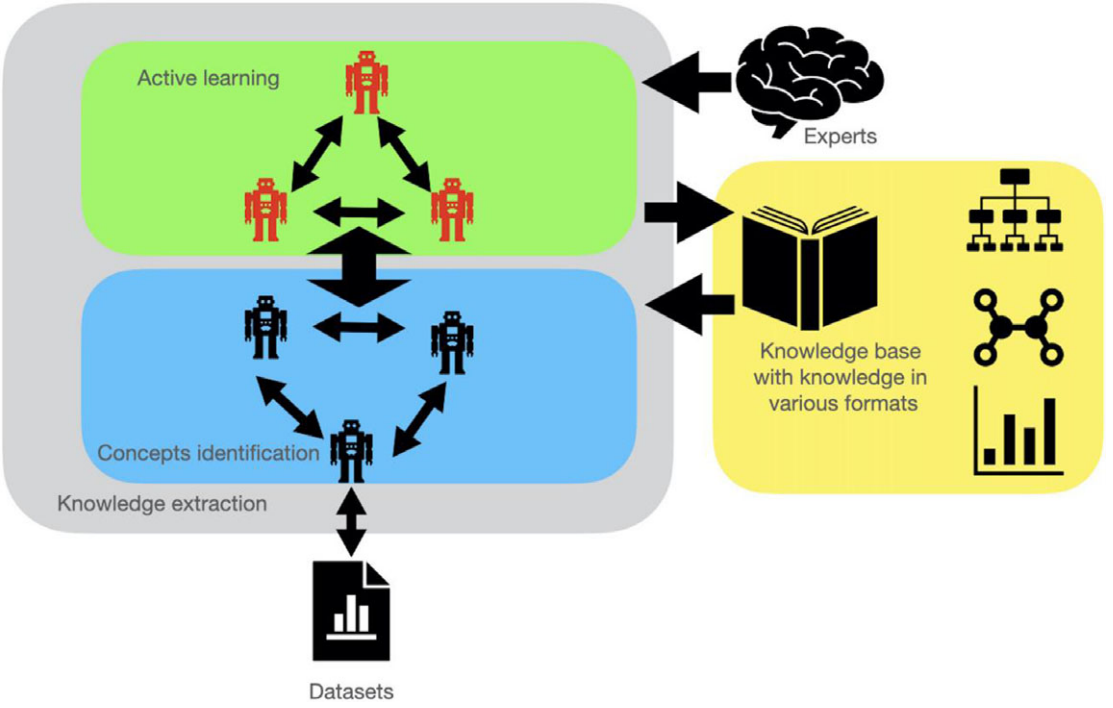


***Figure 4.*** *The interactive collaboration of agents in knowledge extraction. During this process, the agents also interact with experts for better validate extracted knowledge.*

### *3.5. Knowledge extraction*

In this paper, our research focuses on using multiple agent modelling and machine learning methods to implement knowledge extraction in a segmented manner. In this way, the system can more efficiently and accurately extract patterns and models from the data with complicated contexts. These identified patterns and expert annotations will be translated into semantic concepts and integrated into the KG. The entire knowledge extraction task in our system includes two key steps. The first is concept identification, while the second is pattern recognition.

In our system, as shown in Figure 4, we use CI agents to accomplish the task of concept identification and detect concepts that are related to the corresponding patterns in data. AL agents collect expertise from experts, which is used to help the system recognize patterns and concepts during the data analysis. All validated knowledge will be stored in the knowledge base that is represented as a hierarchical KG. In general, interaction between CI and AL agents achieves relation verification and pattern recognition. After the system has recognized the relevant patterns, the knowledge will be extracted from the combination of patterns before the knowledge is updated into KG. To be more concrete, we use an example to elaborate the pipeline of knowledge extraction in our framework.

During the data analysis, the system will use CI agents to check if the data has matchable concepts. When the agents detect the verifiable concepts, the corresponding agents will be rewarded through their reinforcement learning. Such iteration will allow the agent to become more sensitive to interaction with other rewarded agents at the same time to establish a relationship. An AL agent will simultaneously keep the interaction of agents and data analysis under surveillance and select the suspect or valuable cases to inquire the experts for feedback. The main task of an AL agent is to maintain and update the KG, which is accessible to other agents.

The knowledge in KG supports the behaviour model of all agents in the framework. If a collective pattern emerges within a particular context in the interaction of CI agents, that pattern will be captured

by AL agents and updated in the KG as new knowledge for a further knowledge integration step. Each CI agent can access the related concepts and functions to identify the corresponding concepts in the data. These related models of concepts and functions will also be used to induct the calculation of the maximum expected future rewards for action at each state and provide the reinforcement learning method that will constantly optimize agents' behaviour.

In the KG, each entity corresponds to a concept, and each concept may connect with a few recognizable patterns stored in the KG nodes. The corresponding agents can discern these patterns after loading knowledge from KG. All successfully extracted knowledge in data analysis will be used to extend the KG and reinforce the knowledge extraction itself later. For example, agents can transform the knowledge about recognizing a particular concept from data or embed expert annotations into an executable computational model and update such model on the KG as a recognizable pattern. After verification, the knowledge or model will be accessible to the related CI agent that oversees discerning the concept in further data analysis and help the CI improve its efficiency. The pseudocode below shows the knowledge extraction process in a step-by-step manner.

1. Initialize the program.
2. Prompt the user for their task requirements.
3. Read and store the user's requirements. The relevant concepts and their CI agents will be activated.
4. Load the designated dataset and scan the data documents.
5. Perform the knowledge extraction task based on the user's requirements:
    a. Check for matchable concepts during the process.
    b. If a matched concept occurs:
        A. Activated the concept by creating the corresponding CI agent and AL agent.
        B. Check if there is connection can be confirmed with given possible relations between any two activated concepts.
        C. Continue with the main process.
    c. If no matched concepts occur, wake up all CI agents and iterate the loop on each agent, and proceed with the task.
    d. Every single sentence and document scan will be counted, and the number of scanned items will be used to check the lifecycle of CI agents. Based on the features of the corresponding concept, each agent has their own lifecycle. The activation will be deactivated when an agent reaches its own lifecycle.
6. After the scan of a document, store the relations of activated concepts and convert it as a KG. Comparing this KG with other KG from different document scan results and finding the overlapping parts.
7. After the task, conclude all KG and highlight the overlapping parts. Ask the user if they want to review and validated any newly extracted knowledge and determine the possible conflicts.
    a. If yes, prompt for knowledge validation dialogue and save the validated new knowledge.
    b. If no, skip this step.
8. End the program.

Thread to CI agent:

1. Initialize the agent and access the KG to retrieve the knowledge of the corresponding concept, Setup the life circle and start the loop:
    a. Based on the knowledge from KG, check if there is a related concept that can be found in the current data. If there is one, create the CI agent of the related concept as well.
    b. Check if there is given relation can be confirmed between activated concepts that share the same context. If so, connect with the CI agent of that concept.
    c. Perform interactions with connected CI agents.

   d. If there is any conflict or events requiring attention, wake up the corresponding AL agent and iterate the loop on the AL agent.
   e. Updating the life circle steps. If the life circle is done, remove the CI agent and the corresponding AL agent from the system.
   f. If the life circle is not completed, sleep and back to the main process from where it left off.
2. Release the memory and update to knowledge base

Thread to AL agent:

1. Initialize the agent and access the KG to retrieve the knowledge of the corresponding concept, start the loop:
   a. Based on the knowledge from KG, check if the current problem has a corresponding template.
   b. Using the predefined template to query the corresponding user or requiring users to suggest a template for the next communication.
   c. Collect the feedback from users based on the given problem.
   d. Update the new knowledge with related CI agents. If the problem solved, update the new knowledge into the log file, otherwise, iterate the above process again.
   e. Based on the user's feedback, optimise the behaviour model of AL agent.
   f. Sleep and back to where it left off.
2. Release the memory and update to knowledge base

Our current research aims to identify how given expertise can affect knowledge extraction through an agent-based framework and make the knowledge extraction more specific based on the expert's interest. The initial expertise includes general assumptions about the possible relevant factors and the contagion model. The knowledge is represented as a set of statements for the relation between concepts. For example, one of these statements is that COVID-19 contagion is affected by temperature change and demographic factors: age group, population density, and population mobility. These statements can be nested. Taking the above example, assuming another statement can be more population density increase the possibility of contagion and this new one can be embedded into the previous statement as a supplementary part. We collect these assumptions from the expert (Dr Junying Liu) who has studied in the research about SARS and COVID-19 for many years and make them the predefined knowledge to test with our system. The extracted knowledge is represented as sub-KGs describing each statement. These sub-KGs will be validated and then integrated into the hierarchical KG in the next knowledge integration process based on overlapping concepts.

Currently, we didn't involve the expert in our experiments all time but used predefined knowledge to mimic the experts' response. The knowledge is represented as an associated pattern pool, and AL agents can access it. Experts will affect the AL agents in the experiment by providing the predefined connect patterns between concepts and constraints for concepts connecting. The feedback is based on the consistency of queries to predefined patterns, and fewer violations can have better feedback. In the future study, we plan to develop a corresponding interface allowing users to simultaneously interact with AL agents and monitor the knowledge-optimizing processes during the tasks. Experts should verify and optimise knowledge quality during the interaction between AL agents in that extended framework. This paper only focuses on using predefined knowledge to give feedback to AL agents in knowledge extraction. The demonstration of knowledge validation and optimization will be left to future work with knowledge integration.

## 4. Results

Our experiments tested our framework with the corpus data that has been introduced above. This dataset is constituted by the literature related to COVID-19. Through analyzing massive data with our approach, we extracted abundant concepts and successfully established the relations between these concepts based on the inherent context of the corresponding literature. By comparing with the standard text mining

process (keyword extraction) and the applications that didn't exploit active learning agents, the result demonstrated a considerable potential and advantage of our proposed approach to knowledge extraction.

### *4.1. Datasets description*

In this paper, public corpus data (Kaggle, 2020) is used for training data and test data. The advantage of this data is threefold. First, it is comprehensive and covers the most recent literature, with over 500 000 scholarly articles, including over 200 000 with full text, concerning COVID-19, SARS-CoV-2, and related coronaviruses. Second, the data is presented in a standardized format (JSON files), which makes data access convenient. Third, the dataset is maintained and continually updated on Kaggle and is provided by the White House and a coalition of leading research groups in response to the COVID-19 pandemic. This freely available dataset is provided to the global research community to apply recent advances in natural language processing and other AI techniques to generate new insights supporting the ongoing fight against this infectious disease.

### *4.2. Individual concept identification with muti-agent reinforcement learning*

In the experimental scenario of concept identification, we use a basic keyword extraction method based on lexical analysis along with the TF-IDF technique as the counterpart of our agent-based approach in comparison. TF-IDF is a statistical measure that evaluates how relevant a word is to a document in a collection of documents. In our reference method, we use it as one of the metrics to identify the individual concepts. Other more sophisticated language representation models, such as Word2Vec (Goldberg & Levy, 2014) and BERT (Devlin *et al*., 2018), have also been considered, but the language model itself is not the focus of our experiments yet. Our experiments at this step aim to test how agents can help find the relevant concepts from unorganized data. In the agents, we didn't apply any language representation models except for the knowledge models learned from the agent's interactions. Keeping the language model simple will help to identify the agent's efficiency alone. However, the agents are compatible with any given language model, and the same agent-based framework can still improve the knowledge extraction based on these more sophisticated language models in the same way.

The agents in our framework will apply particular attention in concept identification based on the knowledge they learned from users and data. Such knowledge can also include any given language model such as Word2Vec or BERT. For a more focus on the agent-based framework test, we use a straightforward way to demonstrate the effectiveness of agents in individual concept identification. The reference method selects the relevant concepts from the text if the corresponding keywords reach the threshold of the top TF-IDF range while considering the attributes of the word. It means that concept identification only considers the basic syntax and frequency without any other special attention. The reference method is supported by scikit-learn. By comparing this reference method and our agent-based approach, we could examine the effect of a dispersed agent-based approach for concept identification. We also separately tested the CI agents with and without reinforcement learning in the same experimental scenario to evaluate the reinforcement learning. The reinforcement learning here has been used to optimize the selection of identified concepts.

#### *4.2.1. Evaluation of concept identification*

This section demonstrates the result of experiments on concept identification. Figure 5 illustrates the comparison based on three different approaches to concept identification for the same dataset. The metric for evaluating the efficiency of concept identification is the number of associated concepts during the process. We use a predefined associated pattern pool to evaluate the association between all provided concepts extracted in concept identification. The primary concern in knowledge extraction is to identify concepts related to each other and connect them as knowledge in a semantic manner. Therefore, retrieved concepts more strongly associated with each other in data to represent a certain knowledge are regarded

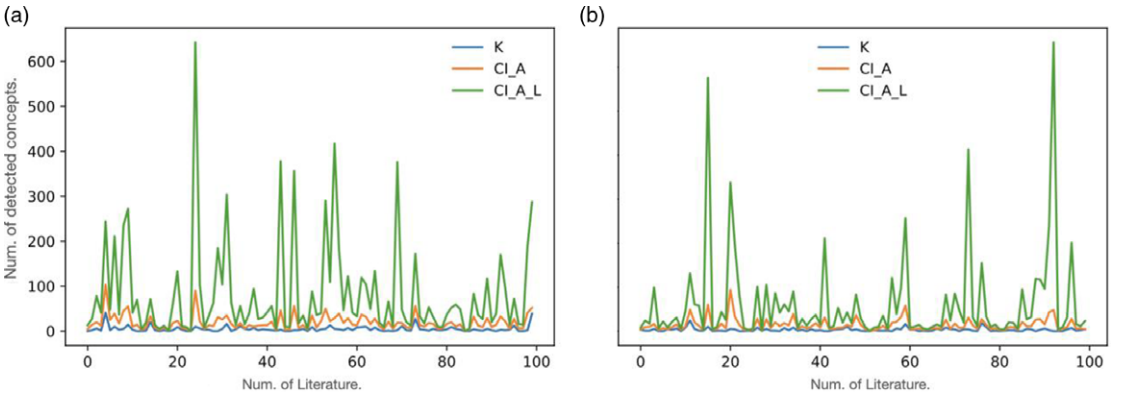


***Figure 5.*** *The comparison of three different approaches to concept identification in single experiment. (a) The comparison result using the same dataset in one run, while (b) illustrates the result obtained from another dataset in another run. The efficiency of the three different approaches may be slightly different based on different datasets, but the general tendency is clear for these three approaches.*

as a more valuable extraction result than those more loosely connected. Thus, we predefined a group of rules in the pattern pool to evaluate the association between concepts. One possible rule example is a simple lexical decision tree (based on the specific part-of-speech setting of subject, predicate and object and the relevant type of nouns). If the extracted concepts follow any of these exemplary patterns in the associated pattern pool, these concepts will be detected as associated concepts in our experiments.

Concept identification requires extracting high-quality concepts (more relevant concepts with an appropriate relation) from data. The number of extracted concepts in and of itself can't prove the quality of the concepts despite a certain quantity of extracted concepts being a premise of efficient knowledge extraction. To better evaluate the efficiency here, we count the number of associated concepts rather than extracted concepts. Only those concepts matched with other concepts can be counted as detected in our result. In other words, we count the number of concepts and consider the relation between concepts in our evaluation. A larger number of detected concepts doesn't simply mean more extracted concepts but more extracted concepts that have a given relation and have synchronously appeared in the same paper. We assume false-positive cases will be harder to have a given relation while appearing together because they do not represent true knowledge.

In our results, we compared the results of three approaches to examine the effectiveness of CI agents and the functionality of reinforcement learning embedded in the agents. Each run covers 100 papers, and we applied three approaches on the same subset to contrast for the difference. One approach is the basic keyword extraction method (K), a second use CI agents without reinforcement learning (CI_A) and a third employs the CI agents with reinforcement learning (CI_A_L). Figure 5 presents two examples of comparing these three approaches, which come from different runs, of the comparison for these three approaches. The $X$-axis here represents the sequence of literature data based on individual papers. In contrast, the $Y$-axis represents the detection of associated concepts in each corresponding individual paper.

#### *4.2.2. Evolution of CI agents with reinforcement learning*

In our experiments, we also tested the effect of reinforcement learning. The fitness of the reinforcement learning in our current experiments is to detect the maximum associated concepts from each literature. Each CI agent focuses on the identification of a particular concept. When the CI agent identifies its corresponding concepts in data and the concept also has been confirmed by the learning program as associated concepts in the given associated pattern pool, reinforcement learning will reward the agents

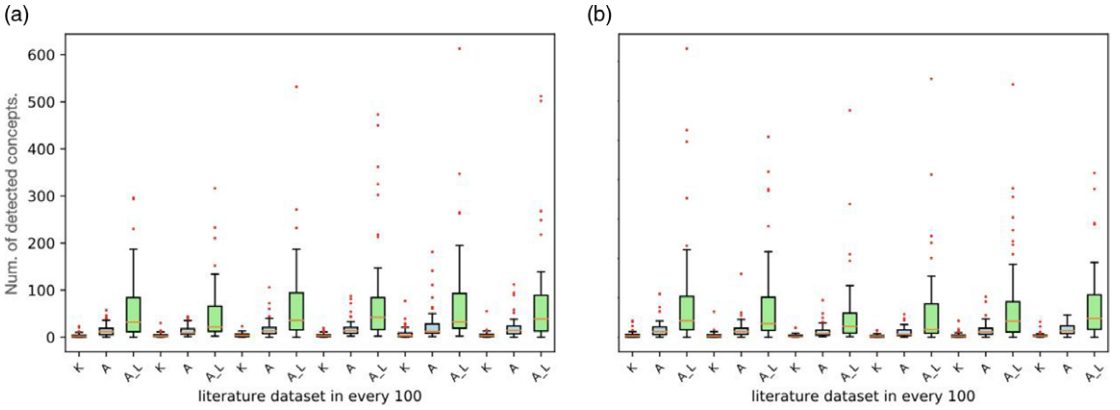


***Figure 6.*** *Multiple comparisons based on three different approaches in multiple runs. (a) The comparison result using the different datasets in the different runs, while (b) illustrates the result obtained from the same experimental conditions with a piece of different initial knowledge. In the experiments, different datasets and initial knowledge both can vary the efficiency of these approaches although CI agents with reinforcement learning always tend to be the most optimal method on all these cases. We separately applied the Mann–Whitney U test and proved the average efficiency of CI agents with the reinforcement learning method is significantly greater (p-value <0.05) than the average efficiency of other methods.*

and increase the sensitivity of the corresponding behaviour. If the identified concepts have not been confirmed as associated concepts or the agents fail to identify any concept in the data analysis, the agent's sensitivity will be decreased while the agent behaviour model will receive a stronger evolutionary pressure. Figure 5 shows that the CI agents typically identified more associated concepts when reinforcement learning is applied to them (the green curve represents the method with active learning and reinforcement learning; the orange curve represents the method with reinforcement learning, but the blue curve represents the method without any leaning approach). A more quantitative comparison is discussed next, as illustrated in Figure 6.

*4.2.3. Comparison of the result across multiple literature datasets*

To provide a more general and comprehensive view, we continuously compared the three approaches on a large scale. Figure 6 presents examples of the comparison using a box plot. Each box in the diagram shows the result of a subset of 100 papers. The *X*-axes labels show the corresponding approaches that have been applied to that subset: Basic keyword extraction method (K), CI agents without reinforcement learning (A) and CI agents with reinforcement learning (A_L).

Figure 6 shows the CI agents with reinforcement learning achieved the highest number of detected concepts of the three approaches. The fluctuations between datasets are due to the deficiency of particular concepts in the associated pattern pool. In our experiments, the given predefined patterns in the pool are not comprehensive and cannot even cover all inherently associated concepts in every literature sample. Due to these uncovered concepts, the system cannot fully evaluate the result leading to poorer detection in some datasets (see more detail on the comparison between Figure 6(a) and (b)). This issue could be resolved in the future by adopting a more comprehensive pattern pool. More details about this future improvement will be discussed in the later section.

Figure 6 also shows the comparison across multiple literature datasets. The general tendency of Figure 6 is similar to previous examples in Figure 5 but includes more comprehensive data and shows a variation between different data. With the result on a larger scale, we still observed that CI agents with reinforcement learning effectively improve the identification of associated concepts in the given data

analysis. In addition, the agent-based system also slightly improves concept identification in contrast to the non-agent-based group.

The advantages of the agent-based system rest on the embedded specific identification model for each particular concept at the individual agent level and the interaction between agents. Unlike keyword extraction, the agents in our framework can adopt sophisticated interactive models to detect the given target concept in data while considering multiple possible contexts. For example, suppose one concept has been identified. In that case, the corresponding agents will be able to recommend highly related concepts based on the previous learning record of these agents by enhancing the sensitivities of these given concepts in identification. By adding reinforcement learning to optimize such a model, the evolvable model can better support identification based on specific feedback from experts. Comparing our implemented experiments suggested great potential for this approach in concept identification.

### *4.3. Relation verification and pattern recognition with active learning*

The previous section considered the issue of concept identification in knowledge extraction. In this section, we show the potential of our framework in subsequent relation verification and new pattern recognition. After a successful concept identification process, complete semantic triples which represent the particular knowledge must be extracted. The system must also verify the new relation between concepts and discern the inherent similarities of the particular contexts within this knowledge. In our framework, this task is implemented by various agents adopted from active learning methods. The connection between this section and the previous section is the associated pattern (knowledge), although we employ different methods in these two sections. In the previous section, we used concept patterns in the associated pattern pool as metrics to select the concepts and measure the efficiency of concept identification. In this section, we recruit the new concept patterns as knowledge based on experts' feedback. With this, we use active learning to optimize the associated pattern pool for improving knowledge extraction. Therefore, we focus on optimising the concept patterns in this section, and we compare the optimization methods (two setups: with or without active learning) instead of comparing the concept identification methods (three setups in the previous section).

After these verified patterns of semantic triples have been confirmed, they can assist the fitness functions in evaluating the quality of concept identification or inducting further searches to complete possible relevant knowledge in data analysis. More importantly, the corresponding semantic triples will finally be used to construct the KG based on the given datasets.

In the active learning implemented by agents, the expertise of experts and previously learned knowledge are used as references to select the interesting cases that will be presented for verification by experts. Once experts verify these interesting cases, the system will integrate new knowledge from the learning result and use this new knowledge to extend the references of active learning at a later time. This means that learning efficiency can also be improved following the accumulation of knowledge.

One example of this relates to improving concept identification in the framework. In the previous section, we discussed the associated pattern pool, where we predefined the models of interesting patterns and demonstrated how active learning could help extend the associated pattern pool constantly. When we manually provided initial patterns for evaluating our results, our proposed framework also uses an automatic way to constantly communicate with experts and recruit the expert-verified patterns dynamically through the whole data analysis process. A larger associated pattern pool will improve concept identification and provide better-identified concepts for optimizing pattern recognition and active learning. Through continuous optimization, experts can view the knowledge extraction and calibrate relevant models based on their expertise. AL agents support the interactive communication between the system and experts. Here, the main goal of AL agents is to transform the expertise of experts into computationally accessible knowledge represented as the format of available semantic triples.

*4.3.1. Evaluation of the effect of knowledge on improving pattern recognition*

The aim of active learning in our framework is to convert human expertise into computationally accessible knowledge. As discussed above, one of the uses of the knowledge in our experiments is to improve pattern recognition. As such, we deploy pattern recognition as the scenario to evaluate the effectiveness of active learning here. Regarding pattern recognition, the value of the learned knowledge through active learning is based on whether the knowledge can help extract meaningful semantic triples more efficiently from data analysis. Here, we assume that more meaningful semantic triples are more likely to be rediscovered in the literature than random and meaningless semantic triples. With this assumption, the amount of successful detection of repetitive semantic triples in different independent literature datapoint (academic papers or articles) could be regarded as an important criterion to evaluate the value of learned knowledge.

In our experiments, we use this metric to test the effect of active learning in improving pattern recognition. We understand that the discussed metric cannot comprehensively represent the quality of learned knowledge since there is a certain complexity in evaluating knowledge. Still, it can provide a statistical overview of the comparison and demonstrate the potential of active learning in pattern recognition. This section aims to give examples of the potential of active learning in knowledge extraction. In future work, we plan to add more criteria to make the evaluation of active learning more comprehensive. We analyze the same datasets in the current experiments through our knowledge extraction process with (AcL) and without active learning agents (non_AcL). Following the increased scanned literature, both methods derived more and more knowledge which reoccurred in different literature. Still, it was evident that the agent-based knowledge extraction with active learning can locate more repetitive knowledge than without active learning.

We compared the extracted knowledge (represented as semantic triples), which are advised by AL agents, to those selected based on fixed selection rules to make inferences about how much variation between them is due to active learning. The setting of the scenario is as follows:

With an identical training dataset which includes 20 randomly selected papers, we set up two different pattern pools. One pool has been built based on the fixed selection rules (according to the importance based on TF-IDF and position in literature), and the other one's construction is based on active learning results, which can be regarded as including the expert's feedback. After both pools have been constructed, we use another identical test dataset to test the quality of the patterns with each pool. Comparing the test results is based on the number of semantic triples on repetitive detection against the number of literature data.

As Figure 7 shows, the patterns selected by active learning are more likely to detect semantic triples from test data. Testing with different training datasets has produced similar results. Further detail can be seen in Table 1 and Figure 8.

*4.3.2. The cluster distribution pattern of knowledge on the similar papers*

We also have compared the distribution patterns of the repetitive detection of two methods based on each particular data. Figure 9 shows the repetitive detections corresponding to each article in the test datasets. The red plot represents the method with active learning (AcL), and the blue plot represents the fixed rules selection (non_AcL). The $Y$-axis measures how many learned knowledge triples have been re-detected at each article in test datasets, and $X$-axis shows each of the articles.

The discrepancy between the two distribution patterns reflects an interesting tendency. In contrast to the non-active learning result, the distribution of detections with the active learning group seems more clustered, and the number of detection in a particular data usually attain much higher levels.

The cluster distribution pattern suggests that similar papers in datasets share certain common concepts and knowledge which could be understood as categorical to fully understand the data. Based on our results, we contend that active learning is more sensitive to exacting critical knowledge supported by experts and detects such knowledge more efficiently in the tests. However, due to the limited expertise, the knowledge extraction in our current experiments has an apparent bias for a comprehensive dataset,

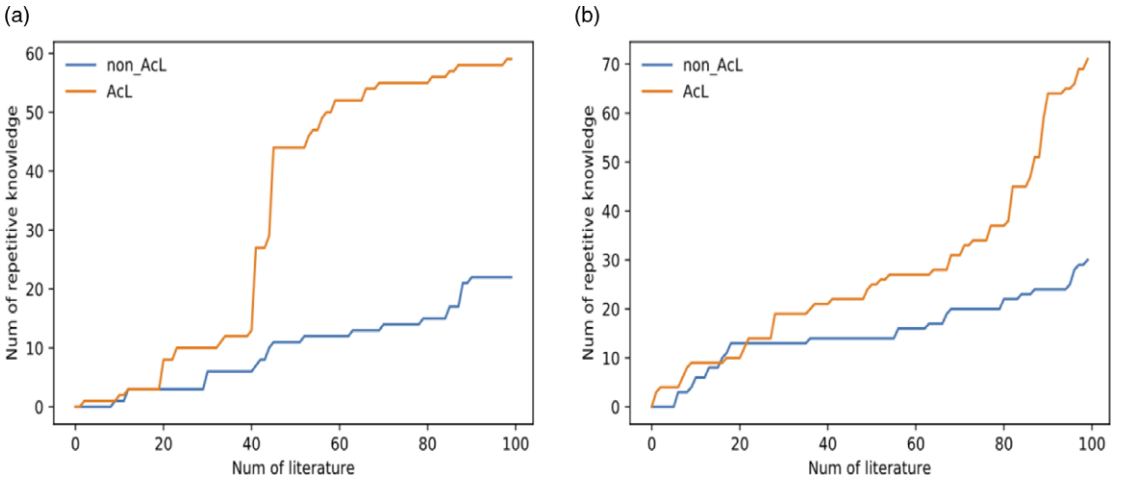


***Figure 7.*** *The comparisons between the knowledge extraction involved active learning agents and the ones without. (a) The comparison result using dataset A, while (b) illustrates the result obtained from dataset B. The application of active learning demonstrates evident benefits across diverse datasets, albeit with slight variations.*

***Table 1.*** *Relevant datasets in the experiment*

| Test datasets | Training data | Num. of the sample articles in the test dataset | Expert feedback | Corresponding figure |
|---|---|---|---|---|
| A | Sample training data 1(20 articles) | 100 | Feedback 1 | 7a |
| B | Sample training data 1(20 articles) | 100 | Feedback 1 | 7b |
| C | Sample training data 2(20 articles) | 100 | Feedback 2 | 8a,9a |
| D | Sample training data 2(20 articles) | 100 | Feedback 2 | 8b,9b |

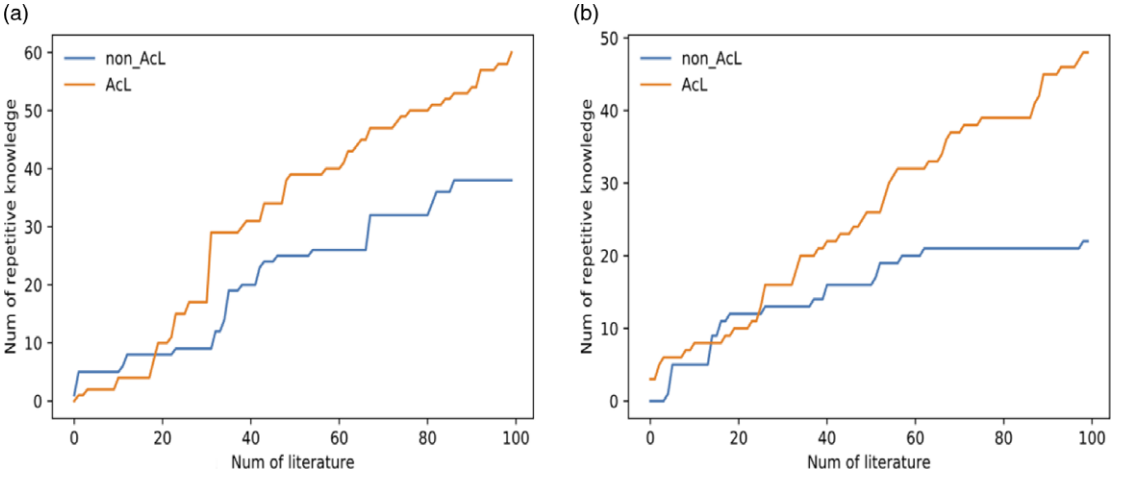


***Figure 8.*** *The similar tendency based on different training datasets.*

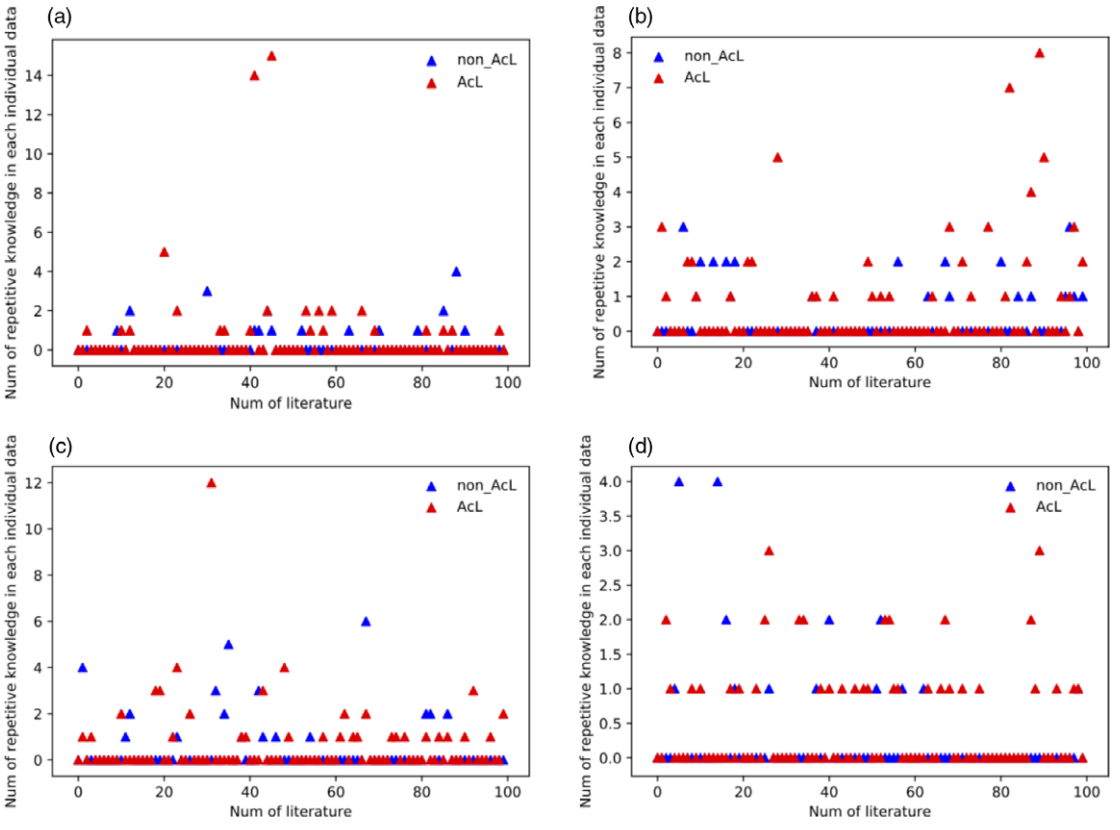


***Figure 9.*** *The distribution patterns of the repetitive knowledge detection.*

as we observed that most of the cases drop on the bottom line here. In other words, for some papers, the select metrics based on expertise can help to detect more knowledge, but such expertise would not work for all papers in all datasets. This issue should be resolved after accumulating more select metrics from an interdisciplinary expert consortium.

## 5. Discussion and conclusion

This paper presented an analytic study of knowledge extraction from unannotated and multidimensional Big Data. Learning processes usually require an intensive manual effort to extract organized knowledge from massive, unorganized data to define the related concepts and label the training data before benefiting from these data. This research aims to assist the exploration of interdisciplinary research in COVID-19 and to reduce the manual workload of data annotation to researchers by providing an automatic knowledge extraction approach based on active learning, reinforcement learning, agent-based modelling, and semantic techniques.

Our proposed approach adopts AL agents, which learn from interactive dialogues between expert users. The experts provide feedback throughout the data analysis and then annotate the result with particular concepts based on their research interests. This feedback and annotation from experts will be learned by the AL agents and converted into knowledge on KG. Such knowledge includes the necessary expertise for interpreting data and continually helps the system to extract similar conceptual patterns from data in the future.

This method can reduce the cost of annotating concepts by simplifying the training dataset and providing a certain level of transfer learning on knowledge extraction. It also could support an incremental

learning process necessary to maintain and optimize a comprehensive KG for Big Data in the long term. This automatic knowledge extraction system can effectively update its knowledge based on new data, while its KG will maintain the learned reusable knowledge in an integrated manner. The knowledge in our study consists of individual concepts and is represented by the combination of multiple agents during knowledge extraction. With expert verification and support, such a bottom-up knowledge model process certain flexibility to adaptation and explainability to interact with users. Overall, our proposed knowledge extraction framework has the potential to outperform traditional systems of processing heterogeneous datasets and the capability to cope more effectively with the rapid growth of unstructured Big Data.

### *5.1. Impact of knowledge extraction and future work on knowledge integration*

Knowledge extraction is an essential process of data mining and knowledge integration, the efficiency and quality of which directly determine the result of knowledge integration. For the unorganized Big Data of diverse formats, knowledge extraction can be also challenging, mainly when identifying knowledge requires an understanding of a particular context.

Integrating the extracted knowledge with different contexts into a common knowledge base remains a conundrum for many data-mining projects today. Our research offers a solution to the current problem. By harnessing agent-based modelling and machine-learning methods improvements in knowledge extraction, we can easily see this framework's promising potential for dealing with knowledge based on various contexts. The system identifies particular concepts in our presented scenarios instead of extracting the entire conceptual model or constructing the complicated network. Based on the identified concepts, the autonomous agents use the learned expertise to verify the particular context and reassemble the relation between concepts.

Combined with the inherent hierarchical structure of KG, it allows knowledge to be developed in different contexts in the system. In addition, adopting the reinforcement learning inside of individual agents provides a potential capable of constantly optimizing the agent's behaviour in synergy. The experiments show that our approach can more efficiently identify and understand the context-related knowledge reflected from massive unannotated datasets than the flat text-based searching method. This work supports further knowledge integration by delineating context-aware and efficient knowledge extraction methods.

We believe such context-aware knowledge extraction and integration may have significant utility in many applications. For example, for the issue of COVID-19, the world pandemic emergency means tools are urgently needed to coordinate the efforts of reviving economies while minimizing the risk of further viral infection. Integrated and context-aware knowledge could be key to managing the crisis and rebuilding civil society. Due to the COVID-19 lockdown, many economic activities have been interrupted, and such an impact has caused a domino effect across multiple industries. Rebooting the stalled system needs not only the participation of related service providers but also the recovery of consumer confidence, stable support from upstream industries, and perfect timing. To achieve these, we must acknowledge every role within the system, provide pertinent information, and synchronize all behaviours with integrated knowledge. Ultimately, this study aims to develop a knowledge integration platform to provide context-aware knowledge in an integrated manner. This is a particularly daunting challenge for various situations since, due to timing and background, responses could vary greatly in terms of practical application. Due to such difficulties, social synergy remains relatively low despite enormous relevant data repositories. The dissemination of knowledge deserves better coordination.

Knowledge integration also has the potential to be applied in more general fields, such as the pharmaceutical industry. Pharmaceutical research has developed data digitization tremendously. This process motivates the pharmaceutical industry to employ AI to build systems and platforms that can interpret and learn to interface with experts and make independent decisions to achieve specific goals. Integrated knowledge can help AI applications make significant progress, and such a knowledge-driven platform can be a potent and useful tool for those who know how to use it. For example, knowledge-driven AI

can use patient-specific genome-exposure analysis to help select specific disease populations for recruitment in clinical trials; preclinical discovery of molecules based on the knowledge integration between Molecular Biology and Pathology can help early predict drug-likeness molecules that target selected patient populations; integrated electronic health records data help the administrators or clinicians of clinical trials closely monitor patients and help them follow the expected protocol of the clinical trial.

In our future work, we hope to adopt supplementary functions and APIs for processing heterogeneous datasets to enable this platform to collect the necessary data and share context-related knowledge across interdisciplinary domains. The knowledge integration program will provide such knowledge in KG to establish a comprehensive and integrated knowledge base. Our study in this paper represents an important step in knowledge integration. In future, based on the developed knowledge extraction approach, we aim to continuously extend the agent-based system to achieve comprehensive knowledge integration of Big Data.

**Acknowledgement.** This work was supported with the financial support of the Science Foundation Ireland grant 13/RC/2094_P2 and co-funded under the European Regional Development Fund through the Southern & Eastern Regional Operational Programme to Lero—the Science Foundation Ireland Research Centre for Software (www.lero.ie). This project has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement no. 754489. For the purpose of Open Access, the author has applied a CC BY public copyright licence to any Author Accepted Manuscript version arising from this submission.

## References

Bishop, C. M. 2006. *Pattern Recognition and Machine Learning*. Springer.

Che, D., Safran, M. & Peng, Z. 2013. From big data to big data mining: challenges, issues, and opportunities. In International Conference on Database Systems for Advanced Applications, Springer, 1–15.

Chen, X., Zhang, N., Xie, X., Deng, S., Yao, Y., Tan, C., . . . & Chen, H. 2022. Knowprompt: Knowledge-aware prompt-tuning with synergistic optimization for relation extraction. In Proceedings of the ACM Web Conference 2022, 2778–2788.

Cheng, Y., Chen, K., Sun, H., Zhang, Y. & Tao, F. 2018. Data and knowledge mining with big data towards smart production. *Journal of Industrial Information Integration* **9**, 1–13.

Coppola, M., Guo, J., Gill, E. & de Croon, G. C. 2019. Provable self-organizing pattern formation by a swarm of robots with limited knowledge. *Swarm Intelligence* **13** (1), 59–94.

Costa, J. P., Grobelnik, M., Fuart, F., Stopar, L., Epelde, G., Fischaber, S., . . . & Davis, P. 2020. Meaningful big data integration for a global COVID-19 strategy. *IEEE Computational Intelligence Magazine* **15**(4), 51–61.

Devlin, J., Chang, M. W., Lee, K. & Toutanova, K. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805.

Dutta, A., Meilicke, C., Niepert, M. & Ponzetto, S. P. 2013. Integrating open and closed information extraction: challenges and first steps. In NLP-DBPEDIA@ ISWC.

Elahi, M., Sugiyama, M. & Kaplan, D. 2016. Active learning in recommender systems. In *Recommender Systems Handbook*, Ricci, F., Rokach, L. & Shapira, B. (eds), 2nd edition. Springer US. doi: 10.1007/978-1-4899-7637-6. hdl:11311/1006123. ISBN 978-1-4899-7637-6.

Ghosh, S. & Ghosh, S. K. 2022. MANTRA: semantic mobility knowledge analytics framework for trajectory annotation. In IEEE INFOCOM 2022-IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS). IEEE, 1–2.

Goble, C. & Stevens, R. 2008. State of the nation in data integration for bioinformatics. *Journal of Biomedical Informatics* **41** (5), 687–693.

Goldberg, Y. & Levy, O. 2014. word2vec Explained: deriving Mikolov et al.'s negative-sampling word-embedding method. arXiv preprint arXiv:1402.3722.

Gyrard, A., Gaur, M., Padhee, S., Sheth, A. & Juganaru-Mathieu, M. 2018. Knowledge Extraction for the Web of Things (KE4WoT) WWW 2018 challenge summary. In Companion Proceedings of the The Web Conference 2018, 1935–1936.

Hürriyetoğlu, A., Yörük, E., Mutlu, O., Duruşan, F., Yoltar, Ç., Yüret, D. & Gürel, B. 2021. Cross-context news corpus for protest event-related knowledge base construction. *Data Intelligence* **3** (2), 308–335.

Kaggle. 2020. COVID-19 Open Research Dataset Challenge (CORD-19).

Kendal, S. L. & Creen, M. 2007. *An Introduction to Knowledge Engineering*. Springer, ISBN 978-1-84628-475-5, OCLC 70987401.

Kraljevic, Z., Searle, T., Shek, A., Roguski, L., Noor, K., Bean, D., . . . & Dobson, R. J. 2021. Multi-domain clinical natural language processing with MedCAT: the medical concept annotation toolkit. *Artificial Intelligence in Medicine* **117**, 102083.

Kulikovskikh, I., Lipic, T. & Šmuc, T. 2020. From knowledge transmission to knowledge construction: a step towards human-like active learning. *Entropy* **22** (8), 906.

Lehmann, J., Isele, R., Jakob, M., Jentzsch, A., Kontokostas, D., Mendes, P. N., Hellmann, S., Morsey, M., Van Kleef, P. & Auer, S., et al. 2015. DBpedia–a large-scale, multilingual knowledge base extracted from Wikipedia. *Semantic Web* **6**( (2), 167–195.

Liu, B., Guo, W., Niu, D., Wang, C., Xu, S., Lin, J., . . . & Xu, Y. 2019. A user-centred concept mining system for query and document understanding at tencent. In Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, 1831–1841.

Nadgeri, A., Bastos, A., Singh, K., Mulang, I. O., Hoffart, J., Shekarpour, S. & Saraswat, V. 2021. Kgpool: Dynamic knowledge graph context selection for relation extraction. arXiv preprint arXiv:2106.00459.

Potter, S. 2003. A survey of knowledge acquisition from natural language. TMA of Knowledge Acquisition from Natural Language.

Rosenthal, S., Biswas, J. & Veloso, M. M. 2010. An effective personal mobile robot agent through symbiotic human-robot interaction. In AAMAS, 10, 915–922.

Settles, B. 2010. *Active Learning Literature Survey*. Computer Sciences Technical Report 1648. University of Wisconsin–Madison. Retrieved 2014-11-18.

Shubhomoy, D., Wong, W.-K., Dietterich, T., Fern, A. & Emmott, A. 2016. Incorporating expert feedback into active anomaly discovery. In IEEE 16th International Conference on Data Mining, Bonchi, F., Domingo-Ferrer, J., Baeza-Yates, R., Zhou, Z.-H. & Wu, X. (eds). IEEE, 853–858. doi: 10.1109/ICDM.2016.0102. ISBN 978-1-5090-5473-2.

Suchanek, F. M., Kasneci, G. & Weikum, G. 2007. Yago: a core of semantic knowledge. In Proceedings of the 16th International Conference on World Wide Web. ACM, 697–706.

Tenorth, M. & Beetz, M. 2017. Representations for robot knowledge in the KnowRob framework. *Artificial Intelligence* **247**, 151–169.

Unbehauen, J., Hellmann, S., Auer, S. & Stadler, C. 2012. Knowledge extraction from structured sources. In *Search Computing*, 34–52.

Wei, C. & Hindriks, K. V. 2012. An agent-based cognitive robot architecture. In International Workshop on Programming Multi-Agent Systems. Springer, 54–71.

Weichselbraun, A., Gindl, S. & Scharl, A. 2014. Enriching semantic knowledge bases for opinion mining in big data applications. *Knowledge-based Systems* **69**, 78–85.

Wu, W., Li, H., Wang, H. & Zhu, K. Q. 2012. Probase: a probabilistic taxonomy for text understanding. In Proceedings of the 2012 ACM SIGMOD International Conference on Management of Data. ACM, 481–492.

Zhukova, A., Hamborg, F., Donnay, K. & Gipp, B. 2021. Concept identification of directly and indirectly related mentions referring to groups of persons. In International Conference on Information. Springer, 514–526.