A Knowledge Base of Argumentation Schemes for Multi-Agent Systems

Carlos Eduardo A. Ferreira[1][a], Débora C. Engelmann[2][b], Rafael H. Bordini[3][c],
Joel Luis Carbonera[1][d] and Alison R. Panisson[4][e]

[1]*Informatics Institute, Federal University of Rio Grande do Sul, Porto Alegre, Brazil*
[2]*Integrated Faculty of Taquara (FACCAT), Taquara, Brazil*
[3]*School of Technology, Pontifical Catholic University of Rio Grande do Sul, Porto Alegre, Brazil*
[4]*Department of Computing, Federal University of Santa Catarina, Araranguá, Brazil*

Keywords: Multi-Agent Systems, Argumentation, Explainable AI.

Abstract: Argumentation constitutes one of the most significant components of human intelligence. Consequently, argumentation has played a significant role in the community of Artificial Intelligence, in which many researchers study ways to replicate this intelligent behaviour in intelligent agents. In this paper, we describe a knowledge base of argumentation schemes modelled to enable intelligent agents' general (and domain-specific) argumentative capability. To that purpose, we developed a knowledge base that not only enables agents to reason and communicate with other software agents using a computation model of arguments, but also with humans, using a natural language representation of arguments which results from natural language templates modeled alongside their respective argumentation scheme. To illustrate our approach, we present a scenario in the legal domain where an agent employs argumentation schemes to reason about a crime, deciding whether the defendant intentionally committed the crime or not, a decision that could significantly impact the severity of the sentence handed down by a legal authority. Once a conclusion is reached, the agent provides a natural language explanation of its reasoning.

1 INTRODUCTION

Multi-agent systems (MAS) are computational systems where autonomous intelligent entities share an environment (Wooldridge, 2009). This paradigm has become popular due to the increasing use of artificial intelligence (AI) techniques and the need for distributed intelligent systems such as smart homes, smart cities and personal assistants. Nowadays, MAS is a popular paradigm for implementing complex distributed systems driven by AI techniques.

MAS communication often uses argumentation-based approaches, allowing agents to communicate arguments that support their positions in dialogues. Arguments are built from reasoning patterns called argumentation schemes (Walton et al., 2008), representing reasoning patterns available for agents in that MAS (Panisson et al., 2021b).

Argumentation schemes are considered a central component in argumentation-based frameworks for multi-agent systems. They enable agents to reason and communicate arguments automatically, as demonstrated in recent practical approaches such as (Panisson et al., 2021b). They also provide explainability within MAS (Panisson et al., 2021a). Agents can translate arguments from a computational to a natural language representation using predefined templates for argumentation schemes. This makes argumentation and explainability in MAS dependent on a knowledge base containing argumentation schemes.

To illustrate this, we take the argumentation scheme `role_to_know`, from (Panisson and Bordini, 2020), as an example. This scheme was included in a knowledge base of 73 argumentation schemes proposed in this work. In Jason syntax (Bordini et al., 2007), this scheme can be represented as a defeasible rule, which goes as follows:

```
defeasible_rule(Conc,[role(Agent,Role),
   role_to_know(Role,Domain),
   asserts(Agent,Conc),
```

[a] https://orcid.org/0000-0002-2275-7966
[b] https://orcid.org/0000-0002-6090-8294
[c] https://orcid.org/0000-0001-8688-9901
[d] https://orcid.org/0000-0002-4499-3601
[e] https://orcid.org/0000-0002-9438-5508

Ferreira, C., Engelmann, D., Bordini, R., Carbonera, J. and Panisson, A.
A Knowledge Base of Argumentation Schemes for Multi-Agent Systems.
DOI: 10.5220/0012547800003690

In *Proceedings of the 26th International Conference on Enterprise Information Systems (ICEIS 2024) - Volume 1*, pages 587-594
ISBN: 978-989-758-692-7; ISSN: 2184-4992

```
   about(Conc,Domain)])
[as(role_to_know)].
```

The argumentation scheme can be explained in natural language as follows:

⟨"<Agent> is a <Role>, and <Role>s know about <Domain>. <Agent> asserts <Conc>, therefore we should believe that <Conc>."⟩[as(role_to_know)]

This explanation in natural language can be enabled in applications through the templates modelled with each argumentation scheme. It is important to provide a clear and concise explanation for each scheme to increase our understanding of the reasoning behind it.

This work includes a knowledge base with argumentation schemes that can be imported individually or grouped by domain. The natural language templates for each scheme enable agents to explain their reasoning to human users. The knowledge base is practically used in the law domain, where an agent uses these argumentation schemes to determine the potential intentionality of a crime and explain its conclusion using natural language. Overall, this knowledge base adds transparency to the reasoning process of MAS and covers general argumentation, as well as legal and hospital bed allocation domains.

Therefore, our work contributes to the field of argumentation and dialogue by using templates to generate natural language arguments and explanations. This differs from the modular platform introduced in (Snaith et al., 2020), which uses a *Dialogue Utterance Generator* to populate abstract moves. Our work is also distinct from the ones presented in (Lawrence et al., 2020; Lawrence et al., 2019; Visser et al., 2019), which focus on recognising and validating argumentation schemes from text or transcripts. Additionally, our work is inspired by previous studies that classify argumentation schemes according to different typologies and provide guidelines for their application in different domains (Visser et al., 2018; Macagno et al., 2017).

2 ARGUMENTATION SCHEMES AS A CENTRAL COMPONENT

Argumentation schemes (Walton et al., 2008) have been playing a central role on defining mechanism for agents reasoning, decision making and communication (Panisson et al., 2021b; Prakken, 2011), applied together with frameworks such as ASPIC+ (Modgil and Prakken, 2014), DeLP (García and Simari, 2014), and others. Most of those works focus on formal aspects of argumentation.

Recently, Panisson and colleagues (Panisson et al., 2021b) proposed an argumentation-based framework in which argumentation schemes are the central components, considering all aspects of these reasoning patterns for developing multi-agent systems. Therefore, in this work, we follow (Panisson et al., 2021b), describing this practical framework to implement multi-agent systems powered by argumentation techniques. We start by contextualising the use of the argumentation scheme in multi-agent systems and introducing this practical framework. Later, we describe how the framework has been used in Explainable Artificial Intelligence (XAI), providing agents with the capability of explaining their reasoning and decision-making.

2.1 Argumentation Schemes in Multi-Agent Systems

Argumentation schemes are patterns for arguments that capture the structure of typical arguments used in everyday discourse, as well as in specific contexts like legal and scientific reasoning (Walton et al., 2008). They represent forms of arguments that are *defeasible*[1], meaning that they may not be strong by themselves, but they may provide evidence that warrants rational acceptance of their conclusion (Toulmin, 1958). That means conclusions from argumentation schemes can be inferred in uncertain and knowledge-lacking situations. The reasoner must be open-minded to new evidence that can invalidate previous conclusions (Walton et al., 2008).

We use a first-order language for representing arguments, following (Panisson et al., 2021b), as most agent-oriented programming languages are based on logic programming. We use uppercase letters to represent variables — e.g., `Ag` and `R` in `role(Ag,R)` — and lowercase letters to represent terms and predicate symbols — e.g., `john`, `doctor` and `role(john,doctor)`. We use "¬" to represent strong negation, e.g., ¬reliable(pietro) means that pietro is *not* reliable. We also use *negation as failure* "not", e.g., not(reliable(pietro)) means that an agent does not know if pietro is reliable.

Let us go back to the example of the `role_to_know` argument discussed in the introduction. Imagine that there are two doctors you are consulting with. One of them, according the argument representing example, called john, says that "smoking causes cancer." However, another agent also playing the role of doctor, called pietro, asserts that "*smoking does not cause cancer*" —

[1] Sometimes called *presumptive*, or *abductive* as well.

$\mathtt{asserts}(\mathtt{pietro}, \neg\mathtt{causes}(\mathtt{smoking}, \mathtt{cancer}))$. Any agent aware of both assertion, John's and Pietro's, is able to construct conflicting arguments for $\neg\mathtt{causes}(\mathtt{smoking}, \mathtt{cancer})$ and $\mathtt{causes}(\mathtt{smoking}, \mathtt{cancer})$. However, the agents can question whether $\mathtt{john}$ and $\mathtt{pietro}$ are reliable (trustworthy) sources, if they really play the role of $\mathtt{doctor}$, and the other questionable points indicated by the critical questions in the argumentation scheme used in order to check the validity of that particular conclusion. For example, imagine that an agent has the information that "*Pietro is not a reliable source*" — $\neg\mathtt{reliable}(\mathtt{pietro})$. In that case, that agent is not able to answer positively the critical question $\mathtt{reliable}(\mathtt{pietro})$, thus it is rational to think that instance of the argumentation scheme (i.e., that argument) might not be acceptable for that agent; the argument concluding $\neg\mathtt{causes}(\mathtt{smoking}, \mathtt{cancer})$ might not be an acceptable instance of the argumentation scheme *role to know* for such a rational agent.

2.2 Argumentation Schemes and Explainability

Argumentation schemes combined with natural language templates can be used to translate arguments from a computational representation to a natural language representation, supporting the development of sophisticated multi-agent applications capable of explaining their decision-making and reasoning (Panisson et al., 2021a). For example, the natural language template for the argumentation scheme `role_to_know` is as follows:

⟨ "<Agent> is a <Role>, and <Role>s know about <Domain>. <Agent> asserts <Conc>, therefore we should believe that <Conc>".⟩$[\mathtt{as}(\mathtt{role\_to\_know})]$

using the same unification function from the previous examples $\{\mathtt{Agent} \mapsto \mathtt{john}, \mathtt{Role} \mapsto \mathtt{doctor}, \mathtt{Domain} \mapsto \mathtt{cancer}, \mathtt{Conc} \mapsto \mathtt{causes}(\mathtt{smoking}, \mathtt{cancer})\}$, it is possible to build the following natural language argument:

⟨"`john` is a `doctor`, and `doctors` know about `cancer`. `john` asserts `smoking causes cancer`, therefore we should believe that `smoking causes cancer`".⟩$[\mathtt{as}(\mathtt{role\_to\_know})]$

Thus, when an agent needs to communicate an explanation in natural language, the agent uses the plan `+!translateToNL` that implements how agents translate arguments from a computational representation to natural language, then aggregating those natural language arguments into an explanation:

```
+!translateToNL([Rule|List],Temp,NLExpl)
<-  !translate(Rule, RT);
    .concat([RT],Temp,NewTemp);
    !translateToNL(List,NewTemp,NLExpl).

+!translateToNL([],Temp,NLExplanation)
<-  NLExpl=Temp.
```

Note that an explanation might be a sequence of arguments (also considered as a chained/complex argument). Thus `+!translateToNL` receives a list of one or more arguments (each of those arguments is an instance of an argumentation scheme). Then, it translates each computational argument to its corresponding natural language argument using the plan `+!translate`, which recovers the natural language template to the argumentation scheme used to instantiate that particular argument and returns its natural language representation.

For example, the plan `+!translate` presented below is used to translate arguments instantiated from the argumentation scheme `role_to_know` to a natural language argument, using the same unification function used to instantiate the argument being translated.

```
+!translate(defeasible_rule(Conc,
  [role(Ag,R), role_to_know(R,Domain),
  asserts(Ag,Conc),about(Conc,Domain)])
  [as(role_to_know)],TArg)
<-  .concat(Ag," is a ",R," and ",R,
    "s know about ", Domain". ", Ag,
    " asserts ",Conc, "therefore we
     should believe that ",Conc,".",TArg).
```

There will be *N* different plans `+!translate`, each one of them implementing a natural language template to translate arguments instantiated from the argumentation schemes available in that particular system to a natural language argument. Agents select them according to the unification of the defeasible rule corresponding to the argumentation scheme being translated. A free variable is used to return the concatenation of strings that is performed through the internal action `.concat`, i.e., `TArg` unifies with the natural language translation of that particular computational argument. Then, the execution returns to the `+!translateToNaturalLanguage` plan and all arguments used in that explanation are concatenated using the same internal action `.concat`, so that the resulting explanation available in `NLExpl` can be communicated by the agent to human users towards any human-agent interaction interface, for example, using chatbot technologies (Engelmann et al., 2021a).

3 A KNOWLEDGE BASE OF ARGUMENTATION SCHEMES

Normally, argumentation schemes used in a particular application are modelled according to the needs of that particular application domain. For example, in (Toniolo et al., 2014) argumentation schemes have been specified for analysing the provenance of information, in (Parsons et al., 2012) argumentation schemes have been specified for reasoning about trust, in (Tolchinsky et al., 2007) argumentation schemes have been specified for arguing about transplantation of human organs, in (Panisson et al., 2018) argumentation schemes have been specified for implementing data access control between smart applications, in (Walton, 2019) argumentation schemes have been specified for reasoning about the intention of executing some actions, and so forth. However, some literature has proposed a set of argumentation schemes for more general reasoning, for example, those compiled in Walton's book (Walton et al., 2008).

In this paper, we present an initiative to build a knowledge base of argumentation schemes for the argumentation framework (Panisson et al., 2021b) implemented in Jason (Bordini et al., 2007), according to the computational representation required by the framework presented in Section 2.1. Also, concurrently with the computational representation of argumentation schemes, we model a natural language template for each argumentation scheme in the proposed knowledge base, according to the approach presented in Section 2.2. Thus, when importing argumentation schemes from the proposed knowledge base, agents are automatically able to use information about its application domain to build arguments from those argumentation schemes available to them, using those arguments for reasoning and communication, as illustrated in Figure 1.

Furthermore, the proposed knowledge base with argumentation schemes can be imported by agents according to the application need, focusing on only the modules necessary (or interesting) for that particular multi-agent application. The knowledge base is organised towards a module hierarchy, starting from modules with unique argumentation schemes, modules with argumentation schemes grouped according to application domains (for example, argumentation schemes used by agents to reason about the domain of law), and the global module corresponding to the complete knowledge base, enabling agent with a large capability of argumentation over different domains.

Therefore, one of the central elements of this work consists of the conception of the knowledge base of argumentation schemes, which proved to be fundamental for evaluating the effectiveness of the argumentation framework approach in different domains. This knowledge base not only allows the implementation of argumentation schemes in the context of a multi-agent system but also provides an additional layer of understanding, facilitating the translation of these arguments into natural language. Consequently, users can lucidly and accurately track the flow of reasoning within the multi-agent system.

To demonstrate our approach, in particular the proposed knowledge base, and how it enables agents to become argumentative (with themselves or with others), in this paper, we are going to focus on how agents build arguments supporting their conclusions in a particular application domain we will present in Section 3.1, then translating arguments to natural language arguments, and providing those arguments as an explanation.

Currently, the knowledge base presented in this paper has about 73 argumentation schemes extracted from the following literature:

- 22 general argumentation schemes from Walton's book (Walton et al., 2008);
- 40 argumentation schemes for reasoning about the domain of bed allocation in hospital from (Engelmann et al., 2021b);
- 11 argumentation schemes about reasoning in legal cases (Walton, 2019; Gordon and Walton, 2009).

In order to demonstrate the proposed knowledge base, we present 3 argumentation schemes available, including their computational representation and natural language templates. These argumentation schemes are also used in the case study we present in Section 3.1. The argumentation schemes are: (i) *argumentation scheme from witness testimony* (Walton et al., 2008), *argumentation scheme for motive to intention* (Walton, 2019), and *argumentation scheme from bias*, adapted from (Walton et al., 2008).

Table 1: Argumentation Scheme from Witness Testimony.

Premise	Witness W is in a position to know about the domain D.
Premise	Witness W states that S is true.
Premise	The statement S is related to domain D.
Premise	Witness W is supposedly telling the truth (as W knows it).
Conclusion	Therefore, S may be plausibly taken to be true.

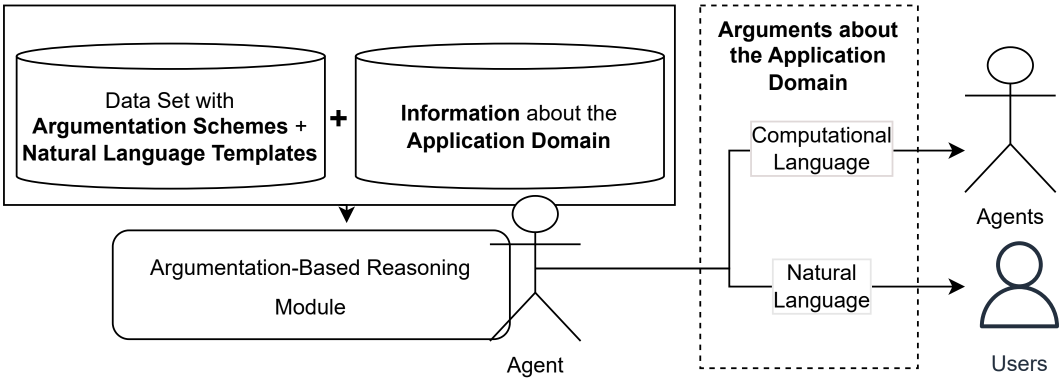

Figure 1: Our Approach for Argumentation-Based Reasoning and Communication using the Knowledge Base with Argumentation Schemes.

```
defeasible_rule(S,[position_to_know(W,D),
  states(W,S),is_related_to(S,D),
  is_telling_the_truth(W)])[as(asFrWT)].
cq(cq1,credible(W,D))[as(asFrWT)].
```

⟨"<W> is in position to know about <D>. <W> states that <S> is true. The statement <S> is related to <D>. <W> is supposedly telling the truth. <W> has credibility to state about <D>. Therefore, it is plausibly to conclude that the statement <S> is true."⟩[as(asFrWT)]

Table 2: Argumentation Scheme from Motive to Intention.

Premise	If agent *Ag* had a motive *M* to commit *A*, then *Ag* is more likely to have intentionally committed *A*.
Premise	*Ag* had *M* as a motive to commit *A*.
Premise	*Ag* committed *A*.
Conclusion	Therefore, *Ag* has intentionally committed *A*.

```
defeasible_rule(
  was_intentional(bring_about(Ag,A)),
  [had_motive_to(M,bring_about(Ag,A)),
  bring_about(Ag,A)])[as(asFrM2I)].
```

⟨ "<Ag> has <M> as a motive to commit <A>. <Ag> indeed committed <A>. Therefore, <Ag> has intentionally committed <A>" ⟩[as(asFrM2I)]

Table 3: Argumentation Scheme from Bias.

Premise	Agent *Ag* has no credibility about domain *D* when it is biased.
Premise	Agent *Ag* is biased about domain *D*.
Conclusion	Therefore, agent *Ag* is not credible about domain *D*.

```
defeasible_rule(¬credible(Ag,D),
  [is_biased_about(Ag,D)])[as(asFrBias)].
```

⟨"<Agent> is biased about <Domain>. Therefore, <Agent> is not credible about <Domain>."⟩[as(asFrBias)]

3.1 Case Study

To demonstrate our framework, we will describe a scenario inspired by the *Peter shot George* case presented in (Verheij, 2003). It is essential to emphasise that this scenario is entirely fictitious, crafted solely for illustrative purposes. It does not depict real events, individuals, or legal cases. By presenting this simplified case, we aim to demonstrate how our approach can be a valuable tool in decision-making processes, particularly in contexts where explicit reasoning and transparency are imperative.

In our scenario, a crime has been committed, and the testimony of two witnesses will be used to help decide whether the defendant intentionally committed the crime or not. Further, extending the analysis by (Verheij, 2003), we include the argument from motive to intention presented by (Walton, 2019), in which the agent considers information about a previous crime (which provides motives to commit the second crime), then inferring whether the second crime was intentional or not. In addition, a given character, John, is considered previously accused of stealing a chicken from another character, Joana's neighbour. The second crime, for which John is in the defendant's position, is the murder of Joana, where, in the absence of security camera records or an expert report, the testimonies of two witnesses who are in a position to know about this case will be decisive.

Given that Joana witnessed John's first crime and, consequently, would testify her witness against him about the chicken theft, there is an interpretation to

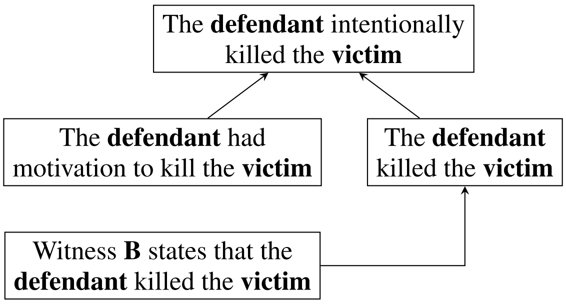

Figure 2: Preliminary inference for the case study.

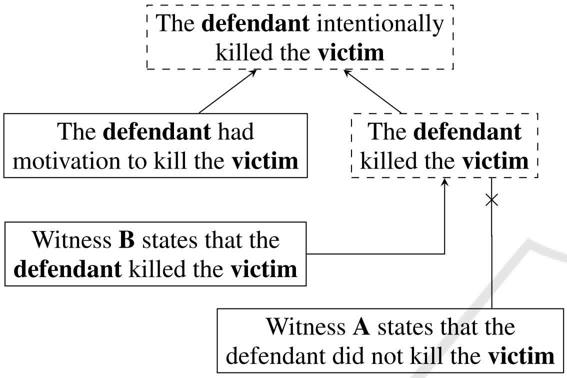

Figure 3: Preliminary inference for the case study, adding information about the witness.

infer that he wanted to prevent this from happening, as it would be negative for him. Thus, it is known that the defendant had the motivation to kill the victim[2], as shown in Figure 2. Under these circumstances, if the defendant is found guilty of Joana's murder, he will be in a very delicate position, given that the act he committed has great potential to be considered premeditated, and he can then be accused of intentional homicide.

Following our scenario, the information that the defendant killed the victim is supported by the testimony of the witness who saw the defendant kill the victim. Together, the motivation to kill the victim and the information that the defendant killed the victim (supported by a witness) support that the defendant intentionally killed the victim, an intentional homicide is suggested, as shown in Figure 2.

However, another witness contradicts the accusation, sating the defendant was with her at the crime time. With this new information, there is conflicting information about whether the defendant killed the victim or not, suggesting that there is no conclusive decision to support even that the defendant killed the victim, as shown in Figure 3.

[2]We will assume that there is already enough evidence for the current defendant to be found guilty in the first crime so that this information will be provided by a simple system input.

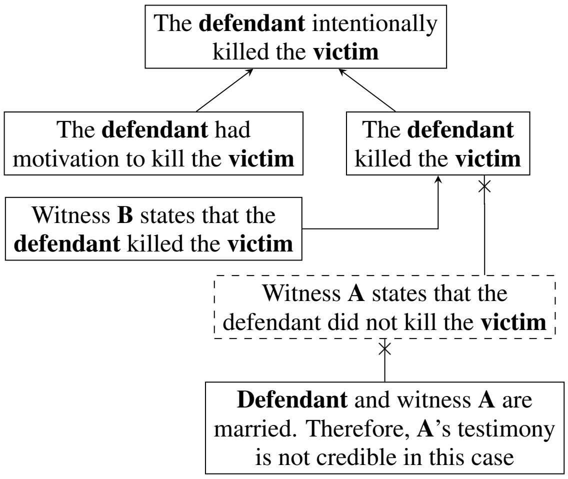

Figure 4: Final inference suggesting there was an intentional crime.

Finally, the agent receives the information that the witness is married to the defendant, thus updating their belief base to this case. It is crucial to emphasise that her testimony cannot be accepted as impartial and reliable evidence due to the clear conflict of interest arising from the marriage between the witness and John. The inherent personal and emotional bias within this relationship can significantly compromise the witness's objectivity when testifying on behalf of John. The credibility of any testimony largely depends on the witness's ability to provide impartial information free from external influences. However, in the present case, the intimate connection between the witness and John creates a natural predisposition to favour the spouse, thereby undermining the testimony's integrity.

Figure 5 shows the agent's final explanation in natural language to the proposed scenario. In this implemented example of ours, it's important to highlight that Witness B was instantiated as Scarlett and Witness A as Laura.

4 RELATED WORK

In (Snaith et al., 2020), the authors describe a modular platform for argument and dialogue, in which they introduce the *Dialogue Utterance Generator* (DUG) component. DUG searches for propositional content to populate the abstract moves present in the modular platform using a template that fills those abstract moves. Moves include the "argue" move, in which an argument is filled. Our work differs from (Snaith et al., 2020) in that we use templates to generate natural language arguments in order to provide explana-

```
[judgeAgent]
   - scarlett is in position to know
     about john's crime judgment.
   - scarlett statement that john
     committed joana's murder is true.
   - The statement john committed
     joana's murder is related to
     john's crime judgment.
   - scarlett is supposedly telling the
     truth.
   - scarlett has the credibility to
     state about john's crime judgment.
   - Therefore, it's plausible to
     conclude that the statement john
     committed joana's murder is
     true.

[judgeAgent]
   - laura is in position to know
     about john's crime judgment.
   - laura statement that john
     committed joana's murder is false.
   - The statement john committed
     joana's murder is related to
     john's crime judgment.
   - laura is supposedly telling the
     truth.
   - laura has no credibility to
     state about john's crime judgment.

[judgeAgent]
   - john has 'avoid joana telling to
     police about your theft ector's
     chicken previous crime' as a
     motive to committed joana's
     murder.
   - john indeed committed joana's
     murder.
   - Therefore, john has intentionally
     committed joana's murder.
```

Figure 5: Jason agent system output.

tions.

In (Lawrence et al., 2020), the authors describe a decision tree for annotating argumentation schemes corpora, providing a heuristic decision tree that aims to clarify Walton's top-level taxonomy (Walton et al., 2008) of 60 schemes. That work is extended with an assistant to annotate argumentation schemes (Lawrence et al., 2019), and an annotated corpus of argumentation schemes is provided by (Visser et al., 2019). Different from our work, (Lawrence et al., 2020; Lawrence et al., 2019; Visser et al., 2019) present components for applications (for example, argument mining (Lawrence and Reed, 2016)) that focus on recognising argumentation schemes from text (or transcripts for natural language interactions), and validating arguments, instances of argumentation schemes according to the data available.

In (Visser et al., 2018), the authors revisited the computational representation of argumentation schemes, providing a guideline for the classification of schemes according to two authors, named Walton's typology (Walton et al., 2008) and Wagemans' Periodic Table of Arguments (Wagemans, 2016). Also, in (Macagno et al., 2017), the authors study the structure, classification and use of argumentation schemes.

Our work on grouping modules of argumentation schemes is inspired by such classification, in which (i) (Visser et al., 2018) aims at using such classification to corpus annotation, and (ii) (Macagno et al., 2017) provides a discussion on the application of argumentation schemes on different domains.

5 CONCLUSION

In this paper, we presented a knowledge base of argumentation schemes that, when provided to agents, allow them to become argumentative (with themselves during reasoning and with others during communication/dialogues). Also, when aware of such argumentation schemes and their respective natural language templates, agents are able to explain their reasoning and decision-making to human users using natural language arguments.

Currently, the knowledge base has about 73 argumentation schemes and their respective natural language templates and has been used to implement sophisticated multi-agent applications in which agents argue with each other and with human users, for example, (Engelmann et al., 2021b). The proposed knowledge base is organised by modules, ranging from individual argumentation schemes and their respective natural language templates to argumentation schemes grouped by application domains[3], for example, the law domain. In our future work, we intend to extend the proposed knowledge base by modelling a large number of argumentation schemes for diverse application domains.

[3]The knowledge base is available open-source at github.com/cadu08/AS_KB_ICEIS24

REFERENCES

Bordini, R. H., Hübner, J. F., and Wooldridge, M. (2007). *Programming Multi-Agent Systems in AgentSpeak using Jason (Wiley Series in Agent Technology)*. John Wiley & Sons.

Engelmann, D., Damasio, J., Krausburg, T., Borges, O., Colissi, M., Panisson, A. R., and Bordini, R. H. (2021a). Dial4jaca – a communication interface between multi-agent systems and chatbots. In Dignum, F., Corchado, J. M., and De La Prieta, F., editors, *Advances in Practical Applications of Agents, Multi-Agent Systems, and Social Good. The PAAMS Collection*, pages 77–88, Cham. Springer International Publishing.

Engelmann, D. C., Cezar, L. D., Panisson, A. R., and Bordini, R. H. (2021b). A conversational agent to support hospital bed allocation. In Britto, A. and Valdivia Delgado, K., editors, *Intelligent Systems*, pages 3–17, Cham. Springer International Publishing.

García, A. J. and Simari, G. R. (2014). Defeasible logic programming: Delp-servers, contextual queries, and explanations for answers. *Argument & Computation*, 5(1):63–88.

Gordon, T. F. and Walton, D. (2009). Legal reasoning with argumentation schemes. In *Proceedings of the 12th international conference on artificial intelligence and law*, pages 137–146.

Lawrence, J. and Reed, C. (2016). Argument mining using argumentation scheme structures. In *COMMA*, pages 379–390.

Lawrence, J., Visser, J., and Reed, C. (2019). An online annotation assistant for argument schemes. In *Proceedings of the 13th Linguistic Annotation Workshop*, pages 100–107. Association for Computational Linguistics.

Lawrence, J., Visser, J., Walton, D., and Reed, C. (2020). A decision tree for annotating argumentation scheme corpora. In *3rd European Conference on Argumentation: Reason to Dissent*.

Macagno, F., Walton, D., and Reed, C. (2017). Argumentation schemes. history, classifications, and computational applications. *History, Classifications, and Computational Applications (December 23, 2017). Macagno, F, Walton, D. & Reed, C*, pages 2493–2556.

Modgil, S. and Prakken, H. (2014). The aspic+ framework for structured argumentation: a tutorial. *Argument & Computation*, 5(1):31–62.

Panisson, A. R., Ali, A., McBurney, P., and Bordini, R. H. (2018). Argumentation schemes for data access control. In *Computational Models of Argument (COMMA)*, pages 361–368.

Panisson, A. R. and Bordini, R. H. (2020). Towards a computational model of argumentation schemes in agent-oriented programming languages. In *IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology (WI-IAT)*.

Panisson, A. R., Engelmann, D. C., and Bordini, R. H. (2021a). Engineering explainable agents: An argumentation-based approach. In *International Workshop on Engineering Multi-Agent Systems (EMAS)*.

Panisson, A. R., McBurney, P., and Bordini, R. H. (2021b). A computational model of argumentation schemes for multi-agent systems. *Argument & Computation*, 12(3):357–395.

Parsons, S., Atkinsonb, K., Haighc, K., Levittd, K., Rowed, P. M. J., Singhf, M. P., and Sklara, E. (2012). Argument schemes for reasoning about trust. *Computational Models of Argument: Proceedings of COMMA 2012*, 245:430.

Prakken, H. (2011). An abstract framework for argumentation with structured arguments. *Argument and Computation*, 1(2):93–124.

Snaith, M., Lawrence, J., Pease, A., and Reed, C. (2020). A modular platform for argument and dialogue. In *COMMA*, pages 473–474.

Tolchinsky, P., Atkinson, K., McBurney, P., Modgil, S., and Cortés, U. (2007). Agents deliberating over action proposals using the proclaim model. In *International Central and Eastern European Conference on Multi-Agent Systems*, pages 32–41. Springer.

Toniolo, A., Cerutti, F., Oren, N., Norman, T. J., and Sycara, K. (2014). Making informed decisions with provenance and argumentation schemes. In *Proceedings of the Eleventh International Workshop on Argumentation in Multi-Agent Systems, 2014*.

Toulmin, S. E. (1958). *The Uses of Argument*. Cambridge University Press.

Verheij, B. (2003). Artificial argument assistants for defeasible argumentation. *Artificial intelligence*, 150(1-2):291–324.

Visser, J., Lawrence, J., Wagemans, J., and Reed, C. (2018). Revisiting computational models of argument schemes: Classification, annotation, comparison. In *7th International Conference on Computational Models of Argument, COMMA 2018*, pages 313–324. ios Press.

Visser, J., Lawrence, J., Wagemans, J., and Reed, C. (2019). An annotated corpus of argument schemes in us election debates. In *Proceedings of the 9th Conference of the International Society for the Study of Argumentation (ISSA), 3-6 July 2018*, pages 1101–1111.

Wagemans, J. (2016). Constructing a periodic table of arguments. In *Argumentation, objectivity, and bias: Proceedings of the 11th international conference of the Ontario Society for the Study of Argumentation (OSSA), Windsor, ON: OSSA*, pages 1–12.

Walton, D. (2019). Using argumentation schemes to find motives and intentions of a rational agent. *Argument & Computation*, 10(3):233–275.

Walton, D., Reed, C., and Macagno, F. (2008). *Argumentation Schemes*. Cambridge University Press.

Wooldridge, M. (2009). *An introduction to multiagent systems*. John Wiley & Sons.