Proceedings of AICOM track of the International Workshop on AI Value Engineering and AI Compliance Mechanisms (VECOMP 2024)

in association with
the 27TH EUROPEAN CONFERENCE ON ARTIFICIAL INTELLIGENCE (ECAI 2024)

AICOM track Co-chairs

Gauvain Bourgne, Sorbonne University, France
Jean-Gabriel Ganascia, University of Sorbonne, France
Adrian Paschke, Freie Universität Berlin and Fraunhofer FOKUS, Germany
Ken Satoh, Center for Juris-informatics, Japan

October 19, 2024

Preface

This volume contains the papers presented at the AICOM track of VECOMP2024: International ECAI 2024 Workshop on AI Value Engineering and AI Compliance Mechanisms (AICOM track) held on October 19, 2024 in Santiago de Compostela, Spain.

With the rapid evolution and spread of AI technologies in society, we can receive many benefits from AI. However, these same technologies also introduce new risks and negative consequences for individuals and society that threaten legal and ethical principles. Thus, we need to ensure that AI is compliant with these principles. This is a central concern that has become prominent both in public opinion and policy makers' agenda.

In the EU, there has been a proposal of "AI Act" to ban AI systems that have an unacceptably high risk to create a clear threat to society, livelihoods, and rights of people and strongly regulate AI systems that have a high risk to be used in critical infrastructures and systems influencing human rights. Other AI systems are not regulated by this Act but AI systems in general should be trustworthy, which means they should be lawful (respecting all applicable laws and regulations), ethical (adhering to ethical principles and values), and robust (both technically and considering their social environment). Therefore, technical solutions are needed to achieve this goal, and it is strongly believed that mechanisms addressing these issues should be embedded at the core of AI agent architectures.

The purpose of this workshop is to bring researchers together to present approaches to tackling legal/ethical AI compliance problems including the relationship between compliance and standards, legislation and regulation and to discuss selected challenges arising from AI compliance. We also solicit use cases related with AI compliance problems to create a basis to investigate common problems for future collaboration.

There were 6 submissions. The committee decided to accept all the 6 papers.

Last but not the least, we would like to thank all the authors who submitted papers and the members of PC and additional reviewers for reviewing the submitted papers.

October 19, 2024
Santiago de Conpostela, Spain

Ken Satoh
Gauvain Bourgne
Jean-Gabriel Ganascia
Adrian Paschke

Table of Contents

Program Committee

Gauvain Bourgne	CNRS & Sorbonnes Universités, UPMC Paris 06, LIP6
Marina De Vos	University of Bath
Wachara Fungwacharakorn	National Institute of Informatics, Sokendai University
Jean-Gabriel Ganascia	Pierre and Marie Curie University - LIP6
Randy Goebel	University of Alberta
Guido Governatori	Charles Sturt University
Hisashi Hayashi	Advanced Institute of Industrial Technology (AIIT)
Hiroshi Hosobe	Hosei University
Julian Padget	University of Bath
Adrian Paschke	Freie Universität Berlin
Pablo Rauzy	Université Paris 8 / LIASD
Livio Robaldo	Legal Innovation Lab Wales, University of Swansea
Giovanni Sartor	EUI/CIRSFID
Ken Satoh	Center for Juris-Informatics, ROIS, Japan
Ralph Schäfermeier	Leipzig University
Alexander Steen	University of Greifswald

Additional Reviewers

Zin, May Myo

Toward smooth integration of an online HTN planning agent with legal and ethical checkers

Hisashi Hayashi [a,*], **Yousef Taheri** [b,**], **Kanae Tsushima** [c,***], **Gauvain Bourgne** [b,****], **Jean-Gabriel Ganascia** [b,*****] **and Ken Satoh** [c,******]

[a]Advanced Institute of Industrial Technology, Tokyo, Japan
[b]Sorbonne University, Paris, France
[c]Center of Juris-Informatics, Research Organization of Information and Systems, Tokyo, Japan

Abstract. Owing to legal and ethical issues such as privacy, safety, and bias, it is crucial to adhere to the laws and respect the ethical guidelines of different countries when transferring or utilizing datasets via the Internet. Therefore, it is necessary to meticulously plan data transfer and utilization in compliance with local laws and ethical guidelines. Given the variability in legal and ethical norms across countries and the specialized knowledge required, we assume that legal and ethical checkers are implemented as independent modules that can be installed on different servers. In this study, we demonstrate how to integrate a planning agent, which utilizes an online HTN (hierarchical task network) planner, with legal and ethical checkers. We also introduce, evaluate, and compare three interaction modes between these modules, assessing the number of interactions and computation times using scenarios involving international data transfer and utilization.

1 Introduction

As data are transferred via the Internet to be used globally for numerous services, legal and ethical issues concerning privacy, security, and other factors have become central concerns. Numerous laws and ethical guidelines have been established to regulate data transfer and usage. A well-known set of data-protection regulations is the European General Data Protection Regulations (GDPR) [7]. Owing to the complexity of laws and ethical guidelines, research has focused on automated compliance checks for data transfer norms. In particular, the policy representation of the GDPR has been studied extensively [1, 4, 14, 22].

Planning the transfer and utilization of datasets is crucial because of the multistep nature of these processes. Moreover, compliance with laws and ethical guidelines is essential when constructing data transfer and utilization plans. Some studies focused on automated planning that considers ethical and legal norms [3, 9, 10, 19]. In particular, the studies [9, 10] utilized a general-purpose online HTN (hierarchical task network) planner for data transfer planning, adapting it to changing situations in which rules describing legal and ethical norms were included in the planning agent database.

Generally, owing to the complexity of legal and ethical norms, specialized expertise is required to conduct automated compliance checks across different countries. Thus, in this study, we proposed the use of a general-purpose online planner and independently developed the norm checkers. In particular, we developed a new architecture that integrates a planning agent with legal and ethical checkers implemented as separate modules. Each of the modules sharing the same interface can be implemented differently in the proposed architecture. However, when these modules are installed on separate servers, it is crucial to ensure that they use the same up-to-date information. This would ensure high efficiency through frequent interactions between these modules.

The contributions of this study are as follows: First, we propose a new architecture that integrates an online planning agent with a legal and an ethical checker. Next, we demonstrate efficiency improvement by changing the database locations and introducing the concept of fluent subscription. Finally, the efficiency gains in terms of the number of interactions between modules and their computation times are illustrated through simulations involving multiple scenarios of planning and replanning for data transfer and utilization.

The remainder of this paper is organized as follows. Section 2 presents a new architecture that integrates the three modules discussed earlier. Section 3 introduces the three interaction modes between the planning agent and the legal and ethical checkers. Section 4 explains the experimental procedure. The results thus obtained are presented and discussed in Section 5. Finally, Section 6 concludes this paper.

2 Overall architecture

This section introduces the overall architecture surrounding the planning agent as shown in Figure. 1. It includes a legal checker, an ethical checker, and an action executor. The planning agent features an online HTN planner that generates plans based on its beliefs and modifies them according to the changing states during plan execution. The agent sends action execution instructions to the action executor, and updates its beliefs and plans based on reports from the action executor. A legal checker evaluates each action in a plan based on legal norms to determine whether it is legal. The ethics checker selects the most ethical plan by comparing multiple plans based on various ethical norms. The action executor performs actions and reports

* Corresponding Author. Email: hayashi-hisashi@aiit.ac.jp
** Email: yousef.taheri@lip6.fr
*** Email: k_tsushima@nii.ac.jp
**** Email: gauvain.bourgne@lip6.fr
***** Email: jean-gabriel.ganascia@lip6.fr
****** Email: ksatoh@nii.ac.jp

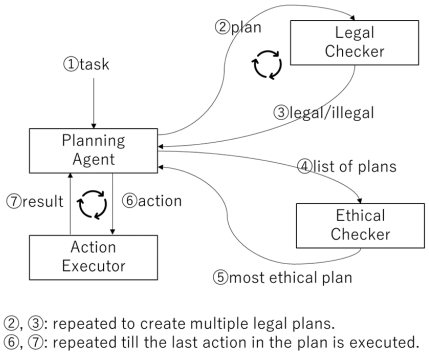

Figure 1. The flow of planning and execution in the proposed architecture.

the results to the planning agent. Sometimes, the executor recognizes unexpected changes in the world, such as changes in the activeness, safety level, or occupancy level of servers, and reports them to the planning agent.

Given task (①), the planning agent creates a least-costly plan using best-first search and sends it to the legal checker (②). The legal checker determines whether the plan is legal and reports the results to the planning agent (③). The planning agent then constructs the second least-costly plan and sends it to a legal checker (② in the second loop) for legal verification. This process is repeated (②–③) until a predefined number of legal plans are obtained or no more possible plans exist. the planning agent sends these low-cost legal plans to the ethical checker (④) and requests that it select the most ethical plan.

The ethics checker then selects the most ethical plan from the given legal plan and reports it back to the planning agent (⑤). At this point, the planning agent commits to the plan selected by the ethical checker. This plan is legal and the most ethical. It sequentially executes each action in the plan using the action executor (⑥). Following the action execution request from the planning agent, the action executor attempts to execute a specified action and/or conduct observations. The results are then reported to the planning agent (⑦), which updates its beliefs and plans based on action execution result and/or observations. When the current plan may become invalid or less cost-efficient, the action executor reports new observations to the planning agent, triggering replanning. Similar to initial planning, the planning agent calls on legal and ethical checkers during replanning (②-⑤).

2.1 Planning agent

The planning agent creates plans using a planner based on the online forward-chaining total-order HTN planning algorithm of Dynagent [11]. Similar to SHOP [13], a standard (offline) HTN planner, it creates plans through task decomposition using a best-first search to find the least-costly plan. The information used for planning is called belief and includes facts, task preconditions, action effects, task costs, and task decomposition rules (called methods in SHOP).

Because of the expressiveness of the planning domain heuristics, SHOP-like total-order HTN planners continue being utilized and studied to improve computational efficiency [2, 12, 18]. Another modern online forward-chaining HTN-like planner conducted a Monte Carlo tree search [15, 16] to find a plan in a large search space.

The planning agent also monitors and controls plan execution, incrementally modifying alternative plans during execution. A state change may affect certain task preconditions in the plans. Therefore, the planning agent checks the preconditions, deletes invalid plans, and adds new valid plans to adapt to a changing world. Moreover, the plan is adjusted if it becomes invalid or less cost-efficient.

The planning agent uses the action executor to perform actions in the current plan. Each time an action is successfully executed, the belief is updated based on the action's effects. The planning agent removes, first, the executed action from the head of each plan, and second, invalid alternative plans. It then adds new valid plans. If an action execution fails, the current plan becomes nonexecutable, and all plans with this failed action at their heads are removed from the alternatives.

As mentioned earlier, the planning agent also relies on legal and ethical checkers to filter out illegal plans and select the most ethical legal plan, respectively.

2.2 Legal checker

Various studies have been conducted on legal compliance using modal (deontic) logic [8, 21], natural language processing [6], and logic programming [5]. In addition, some languages have been introduced to represent legal rules, such as Proleg [17], which extends Prolog with exceptions to handle laws better. In this study, we used the logic programming language Prolog in the legal checker for the following reasons: First, it allows the logic of legal norms with exceptions to be expressed as "negation as failure." Second, because we implemented other parts of the system using Prolog, using the same language for the legal checker helps ensure a seamless implementation. However, each module can be implemented in any programming Language in theory.

The legal checker verifies whether the plan suggested by the planning agent is legal. Because a plan consists of a list of actions, the legal checker evaluates each action and deems the plan legal only if all actions are legal. In this study, the legal checker checks whether the given actions are legal according to GDPR based on the given database information. The database contains information about the permissions granted by of the data owners, countries in the EU, and nodes in the EU, and so on. For example, if a data owner does not grant permission to transfer the data outside the EU, the legal checker determines that it is illegal if the given plan uses the data and a route that goes outside the EU.

2.3 Ethical checker

The ethical checker is responsible for evaluating and selecting the best plan among the valid ones. Its evaluation mechanism was first introduced in [20] and is primarily an ordering process based on a model with multiple criteria. The ordering process considers different criteria which can stem from either moral or optimization considerations. Moral criteria refer to a certain harm or risk that can affect the individuals involved, while optimization criteria are necessary for system efficiency. Hence, they can be seen as either neutral or moral criteria that aim to promote good instead of preventing harm. For example, in the use case model described in Section 4, personal data are transferred through different nodes to be processed for a certain purpose. In this case, two criteria (among others) are used in order to select the best path: node safety and node occupancy. Transferring data through a safer node reduces the risk of a breach and subsequent harm to the subject. Data transfer through a less busy node is faster, which increases the overall system efficiency. Therefore, although

node occupancy is not directly related to risk, it affects system performance and user satisfaction.

The input plans are evaluated on each criterion using an ordinal scale : each criterion orders the plans according to its underlying standard. An ordinal scale helps avoid inconsistencies and improves expressivity of ethical evaluations. After ordering plans according to multiple criteria, they are aggregated to obtain a single order and the best plan is identified. We consider two types of aggregation behavior. First, an order may be (universally) superior to another, in which case, the aggregated order is similar to the superior one, and the inferior order is only considered when the two alternatives have an equal order. Second, when there is a type of reconciliation or trade-off between two (or more) orders instead of superiority, they are seen as votes and aggregated by a voting rule. Note that voting rules from computational social choice theory can be used in this case. Furthermore, the orders can be weighted to represent their importance during the aggregation. Finally, all orders are aggregated by specifying the superiority and trade-off relationships between their corresponding criteria. This specification serves as the ethical setting for the ethical checker, which is built on a relativist view, meaning that it does not judge which input plans are morally right or wrong; instead, it selects the best plan by identifying the one that is best aligned with the given ethical setting.

3 Interaction modes between modules

In this section, we introduce three interaction modes between the planning agent and the legal and ethical checkers.

As discussed in Section 2, the planning agent interacts with legal and ethical checkers during planning and replanning. We assumed that the planning agent, legal checker, and ethical checker are implemented as separate modules that can be installed on different servers. This assumption is natural, given that ethical and legal norms vary between countries. To achieve higher efficiency, we must reduce the number of interactions between these modules and decrease the computation time. Additionally, it is essential to ensure that the most recent information is reflected in the plans.

We introduced the following three interaction modes. 1: *default* mode, 2: *subscription* mode, and 3: *all-subscription* mode. The interaction modes are compared in Section 5 through experiments that evaluate the number of interactions between modules and the required computation time. The three interaction modes are described in the following subsections.

3.1 Default mode

The default mode is the simplest interaction design and serves as the baseline mode. Figure 2 shows interactions in the default mode. In this mode, the common knowledge of fluents describing the changing world is recorded in the planning agent database as a belief. Legal and ethical checkers query the planning agent regarding the truth value of a fluent whenever they need to evaluate a plan for legal or ethical checks, respectively.

Each time an action is executed or the truth value of a fluent is updated, the planning agent replans and updates multiple plans, the legal checker verifies the legality of each updated plan, and the ethical checker selects the most ethical plan from these updated legal plans.

This default interaction mode ensures that the most recent information is used for planning, replanning, and legal and ethical checks.

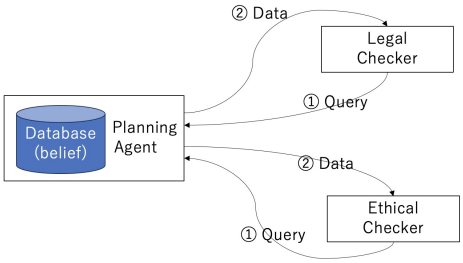

Figure 2. Default mode.

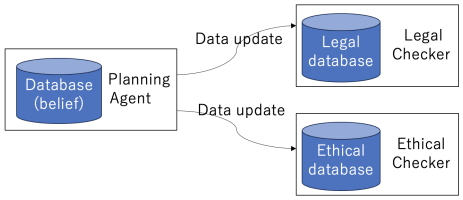

Figure 3. Subscription mode.

However, this is inefficient because the planning agent sometimes requests legal and ethical checks, even when unnecessary. In addition, legal and ethical checkers frequently query the planning agent for the truth value of a fluent, thereby increasing the number of interactions.

3.2 Subscription mode

The subscription mode was designed to improve interaction efficiency. Although the default model is simple and relatively easy to implement, it is inefficient for two reasons: First, the legal and ethical checkers frequently query the planning agent to check the truth value of a fluent, which is among the planning agent's beliefs. This significantly increases the number of interactions between these modules. Second, the planning agent sends requests to the legal and ethical checkers each time an action is executed, increasing unnecessary legal and ethical checks, number of interactions, and the required computation time. When an action in a plan is executed successfully, if the action execution does not change the truth values of the fluents that affect legal and ethical norms, it is unnecessary to modify the current plan and refer to legal and ethical checkers.

In the subscription mode, to address the first reason, the legal and ethical checkers declare the fluents that affect their norm checks as subscribed fluents. Figure 3 shows the interactions in subscription mode. The legal (or ethical) checker maintains a separate database of the subscribed fluents. Initially, the planning agent, legal checker, and ethical checker record the same truth values for each subscribed fluent in their databases.

To address the second problem, in the subscription mode, the planning agent omits legal and ethical checks when an action is successfully executed, provided that the action execution does not change the truth values of fluents that affect legal or ethical norms. However, if the action execution changes these truth values, the planning agent requests the legal checker to refilter the illegal plans and the ethical checker to select the most ethical legal plan.

If the truth value of a fluent is updated through observation, the validity of the plans may be affected. In such cases, the subscription mode is similar to the default mode, i.e., the planning agent replans and creates multiple plans, the legal checker verifies the legality of

each of these plans, and the ethical checker selects the most ethical legal plan.

3.3 All-subscription mode

The *all-subscription* mode is a special example of the subscription mode. In this mode, all fluents are subscribed by their legal and ethical standards. In this case, it is unnecessary to declare the subscribed fluents.

4 Use case model

In order to show the characteristics and efficiency of our proposed approach, we apply it in a data transfer and processing situation. A similar use case model has been used in [19] and [9, 10] as a demonstration of legal / ethical compliance of data manipulations. The model mainly includes multiple nodes that are used to transfer or process data and are connected as illustrated in Figure 4. Each node represents a section of a corporation that is located at a different location, which may be within the EU or outside the EU. Node 4 (marked as a square) is the central node that serves as a cloud server to process data for different purposes. Other nodes (marked as circles) are used to store and transfer data. In this use case, users' personal data are stored in circle nodes. Different sections may ask to apply a processing on data and receive the output of the processing at their corresponding node.

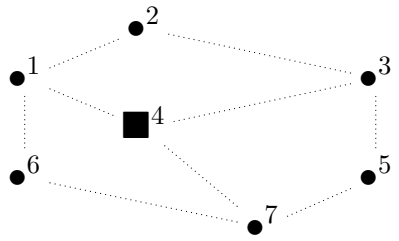

Figure 4. Nodes and connections in the network

In order to perform a task, the system locates the data, transfers them to the processing node, and applies a process with the corresponding purpose. After processing personal data, the system delivers the output to the requested node. The planner in our architecture generates possible plans to satisfy the given task, i.e. the possible paths to transfer data and process them in the network. Each possibility represents different behaviors of the system. According to this architecture, these behaviors are verified by the legal checker for any infringement of the (modeled) regulations. The legal checker rules out the illegal plans, and the remaining plans are ordered by the ethical checker based on their alignment with the ethical specification (cf. Section 2.3).

There is additional information on this use case that enables testing our architecture in different scenarios. Table 1 shows the information on the nodes. The *region* is the location of each node. Since our focus is particularly on GDPR, the regions are categorized as *EU* and *NonEU*. The region of the node is used in the legal verification process. Transferring personal data outside the legislative zone may have ethical implications for data subjects; it is also used in the ethical verification process. The *safety level* corresponds to the safety protocols supported by each node that can be high, medium, or low. Transferring data through more secure nodes is necessary to avoid any possible breach that harms user privacy. Thus, it is important in ethical checking process. The *occupancy level* indicates whether or not a node is busy. It is used to minimize data management time and improve the overall efficiency of the system.

Table 1. The attributes of each node

Node	Region	Safety Level	Occupancy Level
1	Non EU	medium	normal
2	EU	medium	normal
3	EU	medium	busy
4	EU	high	busy
5	EU	high	normal
6	Non EU	low	busy
7	Non EU	high	normal

Table 2. The information of available processing

Processing	Location	Purpose	Bias Level	Required Categories
p1	node 4	recom	2	[c1,c2,c3,c4]
p2	node 4	recom	1	[c2,c3,c5]
p3	node 4	recom	3	[c1,c3,c6,c7,c8]

Table 3. The information on personal data

Data	Category	Storage Location	Owner
du11	c1	node 1	u1
du12	c2	node 1	u1
...	...	...	...
du27	c1	node 2	u2
du28	c2	node 2	u2

Table 2 shows the processing available to apply on personal data. It includes information on the *location* of processing that is node 4 and the *purpose* that is *recommendation* for all processing in this case. The *bias level* shows the extent to which processing can be biased with respect to a certain group. We show this simply by positive integers. Each processing requires certain *categories* of data which are indicated by a list and the category name, e.g. c1, c2, etc.

Last but not least, Table 3 shows information on personal data. This includes their corresponding *category*, the node on which the data are *stored*, the data subject who is the *owner* of the personal data, and permission from the user to take the data out of the EU. Note that the permission may be changed by the data owner during execution.

We demonstrate the functionality of our architecture by testing it in some scenarios in the following section.

4.1 Scenario basecase

Scenario basecase is the baseline scenario. In particular, the situation remains unchanged and the job given to the planning agent is as follows: load the necessary data and process recommendations and deliver the results to node 7. As shown in the map, several routes can be considered. First, the planning agent creates several plans using different data and/or different routes. The legal checker performs the following checks on those plans: node 7 is outside the EU, and some data are prohibited from being taken out of the EU; thus, the plans containing prohibited data are rejected. The ethical checker chooses the best plan from the legal plans. When we ran our prototype, the chosen plan used the following route: node 1 $\rightarrow$ node 4 (recommendation process) $\rightarrow$ node 7.

4.2 Scenario precondition-replan

This scenario aims to show how the system reacts to physical changes in the operating environment, that is, changes in the use case of connected networks, which is explained in the previous section. The objective is to process the personal data of user $u2$ for recommendation purposes. The data are initially stored in a database at node 2 and the processing output is requested at the same node. The initially selected plan is to transfer the data to node 4 via node 1, apply processing $p2$, and send the output to node 2 via node 1. As shown in Table 1, nodes 1 and 3 have the same values for every attribute, except for the occupancy level, where node1 is less busy than node 3; therefore, node 1 is selected in the initial plan. During execution, when the data are loaded from the database, the system realizes that node 1 is suddenly deactivated. The planner replans and selects node 3 as an intermediate to both send data to node 4, where the processing is applied, and transfer it back to node 2. This new plan is executed step by step, and just after the processing in node 4, the system recognizes that node 1 has been reactivated. This new change is considered by replanning from the current stat and node 1 is chosen again as the intermediate node for sending the output back to node 2.

4.3 Scenario ethical-replan

Scenario ethical-replan illustrates how the system reacts to changes that affect the ordering of plans by the ethical checker. In this scenario, the task is to use $u1$'s personal data to create recommendations and deliver results at node 5. $u1$'s data is stored at node 1. To perform the task, the planner transfers personal data from node 1 to node 4 to run the selected process and chooses an intermediary node between nodes 3 and 7 to deliver the result to node 5. Because the safety level of node 7 is higher, the ethical checker initially selects a plan that transfers data through this node. However, just after processing the data at node 4, the system realizes that, owing to some external incidents, the safety level of node 7 has changed to *low*. A re-evaluation is then initiated by the system, and the ethical checker selects the path that passes through node 3 because it is now safer. In this scenario, the physical constraints are fixed; however, the properties that affect the ordering of the ethical checker, and consequently, the selected plan, are changed. The re-evaluation process demonstrates the functionality of our proposed architecture and the ethical checker component in similar situations.

4.4 Scenario legal-replan

In Scenario legal-replan, the planner discovers that a user has rewritten the permission information in the database during execution. The legal checker checks the legality again and finds that the chosen plan is currently not allowed. Thus, the planner re-creates different plans. Specifically, the initial plan selected the dataset [du21,du23,du26,du27,du28] and the route to obtain data from nodes 2 to 7 via the EU to achieve the goal. However, during execution, the permission information for du28 was rewritten to prohibit taking the data out of the EU, which illegalized moving the data through this route. Therefore, the planner uses another dataset [du22,du23,du25] to achieve this goal.

5 Experiments and discussions

Tables 4 and 5 show the executed results. All executions were performed using SWI-Prolog (threaded, 64 bits, version 9.0.4) on a computer: Mac Book Air running MacOS 14.4.1, Apple M2, 8 cores, and 24GB memory. All the runs used the same maximum number of plans, 16. This implies that the planner can create a maximum of 16 plans. The database information presented in Section 4 was almost the same; however, some parameters were modified to represent each scenario. Note that each module can be implemented in any programming language and installed on different servers as long as they can communicate with one another, for example, via remote procedure calls.

Table 4. Executed results: CPU time in seconds.

	default	all-subscription	subscription
basecase	1.349094	1.343983	0.465795
precondition-replan	2.80622	2.747913	1.578951
ethical-replan	2.714904	2.704922	1.313285
legal-replan	3.407112	3.370376	0.809458

Table 5. Executed results: the number of interactions.

	default	all-subscription	subscription
basecase	16916	84	25
precondition-replan	50357	140	51
ethical-replan	32760	121	43
legal-replan	41069	157	47

In our current implementation, we used SWI-Prolog to run three modules on a single computer. Therefore, the communication cost between the modules is minimal. However, these modules could be distributed across servers, increasing the communication cost between the modules. In this experiment, the communication cost was evaluated by counting the number of interactions.

Comparing the default and all-subscription modes, the computation times were almost equal but the number of interactions in the all-subscription mode was significantly lower.

In the default mode, the legal and ethical checkers are called whenever an action is executed. The all-subscription mode functions similarly because an action execution normally changes the truth values of some fluents, which are subscribed to by both checkers.

Furthermore, in the default mode, the legal and ethical checkers have to request the planning agent for the truth value of a fluent. Whereas, in the all-subscription mode, these checkers consult their own databases and need not consult the planning agent. This significantly reduces the number of interactions.

Considering the communication time required for each interaction, the impact of all-subscription mode is huge. Note that, although the communication times for interaction are not included in Table 4, it is possible to estimate them by multiplying the number of interactions and the approximated unit communication time.

In the subscription mode, the number of unnecessary legal and ethical checks are reduced. Compared with the all-subscription mode, both the number of interactions and the computation times is lower. This shows the considerable impact of the subscription mode on the system efficiency.

In any case, the subscription mode was the most efficient in terms of number of interactions and computation time.

6 Conclusion

This paper demonstrates the implementation of a planning agent that smoothly integrates an online planner, a legal checker, and an ethical checker. Moreover, we compared and evaluated three interaction modes and found that the fluent subscription technique works well and significantly reduces the number of interactions and computation time, which are vital for the smooth and efficient integration of

these modules. In future, we plan to improve our integration method for real-time computation of legal and ethical planning.

Acknowledgements

This work was partially funded by JST AIP Trilateral AI Research Grant No. JPMJCR20G4; JST Mirai Program, Grant No. JPMJMI23B1; and JSPS KAKENHI, Grant No. 22H00543 and 21K12144; and Agence Nationale de la Recherche (ANR, French Research Agency) project RECOMP (ANR-20-IADJ-0004).

References

[1] Agarwal, S. Steyskal, F. Antunovic, and S. Kirrane. Legislative compliance assessment: Framework, model and GDPR instantiation. In *Annual Privacy Forum*, pages 131–149, 2018.

[2] G. Behnke, D. Höller, and S. Biundo. totSAT — totally-ordered hierarchical planning through SAT. In *International Conference on Autonomous Agents and Multiagent Systems*, pages 6110–6118, 2018.

[3] F. Berreby, G. Bourgne, and J.-G. Ganascia. Event-based and scenario-based causality for computational ethics. In *International Conference on Autonomous Agents and Multiagent Systems*, pages 147–155, 2018.

[4] P. A. Bonatti, S. Kirrane, I. M. Petrova, and L. Sauro. Machine understandable policies and GDPR compliance checking. *KI - Künstliche Intelligenz*, 34(3):303–315, 2020.

[5] F. Chesani et al. Compliance in business processes with incomplete information and time constraints: a general framework based on abductive reasoning. *Fundamenta Informaticae*, 161(1-2):75–111, 2018.

[6] G. Contissa et al. Claudette meets GDPR: Automating the evaluation of privacy policies using artificial intelligence. *https://ssrn.com/abstract=3208596*, 2018.

[7] European Commission. Regulation (EU) 2016/679 of the European Parliament and of the Council, 2016. URL http://data.europa.eu/eli/reg/2016/679/oj.

[8] G. Governatori et al. Designing for compliance: Norms and goals. In *International Joint Conference on Rules and Reasoning*, page 282–297, 2011.

[9] H. Hayashi and K. Satoh. Towards legally and ethically correct online htn planning for data transfer. In *International Conference on Agents and Artificial Intelligence*, volume 1, pages 154–164, 2023.

[10] H. Hayashi and K. Satoh. Online HTN planning for data transfer and utilization considering legal and ethical norms: Case study. In *International Workshop on Non-Monotonic Reasoning, Federated Logic Conference*, volume 3197 of *CEUR Workshop Proceedings*, pages 4–15, 2023.

[11] H. Hayashi, S. Tokura, T. Hasegawa, and F. Ozaki. Dynagent: An incremental forward-chaining htn planning agent in dynamic domains. In M. Baldoni, U. Endriss, A. Omicini, and P. Torroni, editors, *Declarative Agent Languages and Technologies III*, pages 171–187. Springer, 2006.

[12] M. C. Magnaguagno, F. Meneguzzi, and L. Silva. HyperTensioN: A three-stage compiler for planning. In *International Planning Competition: Planner and Domain Abstracts – Hierarchical Task Network Planning Track*, pages 5–8, 2021.

[13] D. Nau, Y. Cao, A. Lotem, and H. Munoz-Avila. SHOP: Simple hierarchical ordered planner. In *International Joint Conference on Artificial Intelligence*, volume 2, page 968–973, 1999.

[14] M. Palmirani, M. Martoni, A. Rossi, C. Bartolini, and L. Robaldo. Legal ontology for modelling GDPR concepts and norms. *Legal Knowledge and Information Systems*, pages 91–100, 2018.

[15] S. Patra, M. Ghallab, D. Nau, and P. Traverso. Acting and planning using operational models. In *AAAI Conference on Artificial Intelligence*, pages 7691–7698, 2019.

[16] S. Patra, J. Mason, A. Kumar, M. Ghallab, P. Traverso, and D. Nau. Integrating acting, planning, and learning in hierarchical operational models. In *International Conference on Automated Planning and Scheduling*, pages 478–487, 2020.

[17] K. Satoh and et. al. Proleg: An implementation of the presupposed ultimate fact theory of Japanese civil code by prolog technology. In *New Frontiers in Artificial Intelligence*, pages 153–164. Springer Berlin Heidelberg, 2011. ISBN 978-3-642-25655-4.

[18] D. Schreiber. Lilotane: A lifted sat-based approach to hierarchical planning. *Journal of Artificial Intelligence Research*, 70:1117–1181, 2021.

[19] Y. Taheri, G. Bourgne, and J.-G. Ganascia. A compliance mechanism for planning in privacy domain using policies. In K. Yada, Y. Takama, K. Mineshima, and K. Satoh, editors, *New Frontiers in Artificial Intelligence*, pages 77–92, Cham, 2023. Springer Nature Switzerland. ISBN 978-3-031-36190-6.

[20] Y. Taheri, G. Bourgne, and J.-G. Ganascia. Modelling integration of responsible AI values for ethical decision making. In *Workshop on Computational Machine Ethics, International Conference on Principles of Knowledge Representation and Reasoning*, 2023.

[21] M. B. van Riemsdijk et al. Agent reasoning for norm compliance: a semantic approach. In *International Conference on Autonomous Agents and Multiagent Systems*, pages 499–506, 2013.

[22] M. D. Vos, S. Kirrane, J. Padget, and K. Satoh. ODRL policy modelling and compliance checking. In *International Joint Conference on Rules and Reasoning*, pages 36–51, 2019.

An Argumentative Approach for Explaining Preemption in Soft-Constraint Based Norms

Wachara Fungwacharakorn[a,b,*], Kanae Tsushima[a], Hiroshi Hosobe[c], Hideaki Takeda[a,b] and Ken Satoh[a]

[a]Center for Juris-Informatics, Research Organization of Information and Systems, Tokyo, Japan
[b]National Institute of Informatics, Tokyo, Japan
[c]Faculty of Computer and Information Sciences, Hosei University, Tokyo, Japan
ORCID (Wachara Fungwacharakorn): https://orcid.org/0000-0001-9294-3118, ORCID (Kanae Tsushima): https://orcid.org/0000-0002-3383-3389, ORCID (Hiroshi Hosobe): https://orcid.org/0000-0002-7975-052X, ORCID (Hideaki Takeda): https://orcid.org/0000-0002-2909-7163, ORCID (Ken Satoh): https://orcid.org/0000-0002-9309-4602

Abstract. Although various aspects of soft-constraint based norms have been explored, it is still challenging to understand preemption. Preemption is a situation where higher-level norms override lower-level norms when new information emerges. To address this, we propose a derivation state argumentation framework (DSA-framework). DSA-framework incorporates derivation states to explain how preemption arises based on evolving situational knowledge. Based on DSA-framework, we present an argumentative approach for explaining preemption. We formally prove that, under local optimality, DSA-framework can provide explanations why one consequence is obligatory or forbidden by soft-constraint based norms represented as logical constraint hierarchies.

1 Introduction

In complex situations, norms can conflict, leading to challenges in normative systems. To address this, several studies have interpreted norms as *soft constraints* [6, 15, 16, 22]. Unlike *hard constraints*, which must be exactly satisfied, soft constraints are allowed to be relaxed, enabling agents to prioritize norms based on the context. This paper focuses on constraint hierarchies [2], a pioneering formalism for dealing with soft constraints. Prior work has investigated various aspects of maintaining norms represented as constraint hierarchies, including debugging norms based on user expectation [12], revising norms based on new information [13], and exploring connections with case-based reasoning [14]. However, a key challenge lies in understanding preemption. Preemption refers to a situation where higher-level norms override lower-level norms as more information becomes available. If preemption is not well-understood, it can undermine trust in the normative system. Imagine a situation where an agent expects a certain consequence to be obligatory, but it is ultimately forbidden due to preemption. Without explanation, the consequence becomes unexpected and this can erode trust in the system. Therefore, explaining preemption is critical for building trust in the system and allowing agents to understand the rationale of the normative system for handling norms and preferences.

To address this, we present a novel argumentative approach for explaining preemption. Argumentative approaches are widely used for explanations in various reasoning domains [4, 20]. Based on abstract argumentation framework [7], most approaches explored their own methods to build arguments, such as arguments built from precedent cases [4] or defeasible rules [19]. This paper proposes a derivation state argumentation framework (DSA-framework), with arguments built from derivation states and situational knowledge to understand preemption. Based on DSA-framework, we can provide explanations using dispute trees [9]. We formally prove that, if one consequence is obligatory or forbidden under local optimality, we can always find an explanation why it is.

This paper is structured as follows. Section 2 describes norm representation and some logical structures used in this paper. Section 3 proposes DSA-framework. Section 4 presents preemption explanations based on the proposed framework. Section 5 discusses limitations of the framework and suggestions for future work. Finally, Section 6 concludes this paper.

2 Preliminaries

In this paper, we consider representing norms as logical constraints. Let $\mathcal{L}$ be a classical logical language generated from a set of propositional constants in a standard way. We write $\neg$ for negation, $\rightarrow$ for implication, $\leftrightarrow$ for equivalence, $\top$ for a tautology, $\bot$ for a contradiction, and $\vdash$ for a classical deductive monotonic consequence relation. A constraint hierarchy is typically represented as $H = \langle H_1, \ldots, H_l \rangle$, where l is some positive integer, and each $H_i \subseteq \mathcal{L}$, called a *level*, is a finite subset of logical constraints. In original definitions of constraint hierarchies [2], there exists a level H_0 consisting of *required* (or *hard*) constraints that must be exactly satisfied. However, in this paper, we consider the level of hard constraints as a background theory T_0 to simplify other definitions. Each H_i consists of *preferential* (or *soft*) constraints that can be relaxed if necessary. A constraint hierarchy is totally ordered, which means that a preferential level H_i with smaller i consists of more important constraints.

Given a constraint hierarchy $H = \langle H_1, \ldots, H_l \rangle$ and a background theory T_0, we also treat H as the whole set of logical constraints, that is $H = \bigcup_{i \in \{1,\ldots,l\}} H_i$. With a general assumption that T_0 is consistent (i.e. $T_0 \nvdash \bot$), we say H is consistent if and only if $T_0 \cup H \nvdash \bot$. For example, given that T_0 is empty, $\langle \{p\}, \{q\} \rangle$ is

* Corresponding Author. Email: wacharaf@nii.ac.jp.

consistent but $\langle\{p\},\{\neg p,q\}\rangle$ is not. For $\Phi \subseteq \mathcal{L}$, we also say H is consistent with Φ if and only if $T_0 \cup H \cup \Phi \nvdash \bot$.

Applying the concepts of sub-bases [1] and maximal consistent sets [21] to constraint hierarchies, we say a constraint hierarchy $H' = \langle H'_1,\ldots,H'_l\rangle$ is a *sub-base* of $H = \langle H_1,\ldots,H_l\rangle$ if and only if H' must have the same number of levels as H and $H'_i \subseteq H_i$ for every $i \in \{1,...,l\}$. For example, $\langle\{p\},\{q\}\rangle$ is a sub-base of $\langle\{p\},\{\neg p,q\}\rangle$. Let H be a constraint hierarchy, a *sub-base space* of H is a pair $(\Delta,\geq)$ where Δ is the set of all possible sub-bases of H and $\geq$ is a partial order over Δ, representing the preference of soft constraint relaxations. The strict order $>$ associated with $\geq$ is defined as $\delta > \delta'$ if and only if $\delta \geq \delta'$ and it is not the case that $\delta' \geq \delta$ (for δ and $\delta' \in \Delta$). The maximal element of $\geq$ is H itself and the minimal element is the constraint hierarchy with the same number of levels as H but all of them are empty. Corresponding to a local comparator in constraint hierarchies [2], we say $\geq$ is a local preference when $\delta \geq \delta'$ (for δ and $\delta' \in \Delta$) if and only if there exists $k \in \{1,\ldots,l\}$ such that $\delta'_k \subsetneq \delta_k$ and for every $i \in \{1,\ldots,l\}$ $i < k$ implies $\delta'_i = \delta_i$. Given a sub-base space $(\Delta,\geq)$, we use the following notations:

- Δ^{Φ} (for $\Phi \subseteq \mathcal{L}$): the set of all sub-bases in Δ consistent with Φ,
- $max(D)$ (for $D \subseteq \Delta$): the set of all $\geq$-maximal elements of D.

In our setting, we represent a situation as a consistent set of formulas $\Pi \subseteq \mathcal{L}$, and we represent a consequence as a formula $\psi \in \mathcal{L}$ such that $T_0 \cup \Pi \nvdash \psi$ and $T_0 \cup \Pi \nvdash \neg\psi$. Now, we define the concept of obligation, adapted from [17], as follows.

Definition 1 (obligation)**.** *Let T_0 be a background theory, H be a constraint hierarchy corresponding with sub-base space $(\Delta,\geq)$. We say a consequence ψ is* obligatory *(resp.* forbidden*) by H with a situation Π if and only if, for every $\delta \in max(\Delta^{\Pi})$ $T_0 \cup \delta \cup \Pi \vdash \psi$ (resp. $\neg\psi$).*

Example 1 (overtaking)**.** *Considering the following norms regarding overtaking, prioritized from less important to more important.*

1. *Generally, drivers should not overtake the other car.*
2. *If the other car appears obstructed, drivers should overtake the other car.*
3. *If the other car is in a danger zone, drivers should not overtake the other car.*

Omitting the background theory in this example, the norms can be represented as the constraint hierarchy $H = \langle\{p \to \neg r\},\{q \to r\},\{\neg r\}\rangle$ where p represents "the other car is in a danger zone", q represents "the other car appears obstructed", and r represents "drivers should overtake the other car". The constraints are placed in a different order as constraint hierarchies prioritize constraints from left to right. There are eight sub-bases of H, ranked by the local preference as follows.

1. $\delta_0 = \langle\{p \to \neg r\},\{q \to r\},\{\neg r\}\rangle = H$
2. $\delta_1 = \langle\{p \to \neg r\},\{q \to r\},\{\}\rangle$
3. $\delta_2 = \langle\{p \to \neg r\},\{\},\{\neg r\}\rangle$
4. $\delta_3 = \langle\{p \to \neg r\},\{\},\{\}\rangle$
5. $\delta_4 = \langle\{\},\{q \to r\},\{\neg r\}\rangle$
6. $\delta_5 = \langle\{\},\{q \to r\},\{\}\rangle$
7. $\delta_6 = \langle\{\},\{\},\{\neg r\}\rangle$
8. $\delta_7 = \langle\{\},\{\},\{\}\rangle$

Suppose the situation is that another car appears obstructed and it is in a danger zone ($\Pi = \{p,q\}$). We have that $\Delta^{\Pi} = \{\delta_2,\ldots,\delta_7\}$ because δ_0 and δ_1 are not consistent with Π. We also have that r ("drivers should overtake the other car") is forbidden because δ_2 is the maximal element of Δ^{Π} and $\delta_2 \cup \Pi \vdash \neg r$.

3 Proposed Framework

To leverage an argumentation framework in the norm structure, we first define derivation states and derivation state spaces as follows.

Definition 2 (derivation state)**.** *Let T_0 be a background theory, $\delta \subseteq \mathcal{L}$, $\pi \subseteq \mathcal{L}$, $\psi \in \mathcal{L}$, and $\Sigma = \{\bot,+,-,n\}$ be the domain of derivation states. A derivation state (σ) of ψ with respect to δ and π is defined as follows.*

1. *$\sigma = \bot$ if δ is not consistent with π.*
2. *$\sigma = +$ if $T_0 \cup \delta \cup \pi \vdash \psi$ and $T_0 \cup \delta \cup \pi \nvdash \neg\psi$.*
3. *$\sigma = -$ if $T_0 \cup \delta \cup \pi \nvdash \psi$ and $T_0 \cup \delta \cup \pi \vdash \neg\psi$.*
4. *$\sigma = n$ if $T_0 \cup \delta \cup \pi \nvdash \psi$ and $T_0 \cup \delta \cup \pi \nvdash \neg\psi$.*

Definition 3 (derivation state space)**.** *Let H be a constraint hierarchy corresponding with sub-base space $(\Delta,\geq)$, Π be a situation, and ψ be a consequence. A* derivation state space *(denoted by Ω) of ψ with respect to H and Π is the set $\Omega = \{\langle\delta,\pi,\sigma\rangle \in \Delta \times 2^{\Pi} \times \Sigma \mid \sigma$ is a derivation state of ψ with respect to δ and $\pi\}$.*

Now, we define a *DS-argument* as follows.

Definition 4 (DS-argument)**.** *Let H be a constraint hierarchy corresponding with sub-base space $(\Delta,\geq)$ and Ω be a derivation state space. A* DS-argument *from Ω is an element $\langle\delta,\pi,\sigma\rangle$ of Ω that satisfies following conditions.*

1. *$\sigma \neq \bot$, that is δ needs to be consistent with π.*
2. *There is no $\langle\delta',\pi,\sigma'\rangle \in \Omega$ (π is fixed) such that $\delta' > \delta$ and $\sigma' \neq \bot$. In other words, δ is a maximal sub-base of Δ consistent with π, or formally speaking, $\delta \in max(\Delta^{\pi})$.*

For a DS-argument $\langle\delta,\pi,\sigma\rangle$, we call δ a corresponding sub-base, we call π a situational knowledge, and we call σ a derivation state.

Table 1 shows the derivation state space of a consequence r with respect to H from Example 1 and the situation $\Pi = \{p,q\}$ to find all DS-arguments. There are four DS-arguments from this setting: $\langle\delta_0,\{\},-\rangle$, $\langle\delta_0,\{p\},-\rangle$, $\langle\delta_1,\{q\},+\rangle$, and $\langle\delta_2,\{p,q\},-\rangle$, corresponding to the derivation states denoted by asterisks (*) in the table.

Table 1. Derivation state space in Example 1

$\pi\backslash\delta$	δ_0	δ_1	δ_2	δ_3	δ_4	δ_5	δ_6	δ_7
$\{\}$	$-^*$	n	$-$	n	$-$	n	$-$	n
$\{p\}$	$-^*$	$-$	$-$	$-$	$-$	n	$-$	n
$\{q\}$	$\bot$	$+^*$	$-$	n	$\bot$	$+$	$-$	n
$\{p,q\}$	$\bot$	$\bot$	$-^*$	$-$	$\bot$	$+$	$-$	n

Inspiring from abstract argumentation for case-based reasoning (AA-CBR) [4], this paper proposes a derivation state argumentation framework (DSA-framework) based on derivation states and incremental knowledge of the situation as follows.

Definition 5 (DSA-framework)**.** *Let H be a constraint hierarchy corresponding with sub-base space $(\Delta,\geq)$, Π be a situation, and ψ be a consequence with Ω as a derivation state space of ψ with respect to H and Π. A DSA-framework with respect to H, Π, and ψ is $(AR, attacks)$ satisfying the following conditions.*

1. *AR is the set of all DS-arguments from Ω.*
2. *For $\langle \delta, \pi, \sigma \rangle, \langle \delta', \pi', \sigma' \rangle \in AR$, $\langle \delta, \pi, \sigma \rangle$ attacks $\langle \delta', \pi', \sigma' \rangle$ if and only if*
   - *(change derivation state) $\sigma \neq \sigma'$, and*
   - *(gain more knowledge) $\pi' \subsetneq \pi$, and*
   - *(concise attack) $\nexists \langle \delta'', \pi'', \sigma \rangle \in AR$ with $\pi' \subsetneq \pi'' \subsetneq \pi$.*

From Example 1, the DSA-framework with respect to H, the situation $\Pi = \{p, q\}$, and a consequence r can be illustrated in Figure 1. The arrows represent attacks between arguments.

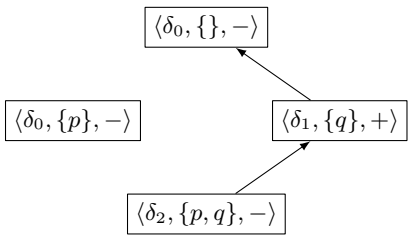

Figure 1. DSA-framework from Example 1

Proposition 1 shows general properties of DSA-framework.

Proposition 1. *DSA-framework has the following properties.*

1. *For every $\pi \subseteq \Pi$ (including $\{\}$ and Π), the DSA-framework has at least one argument with a situational knowledge π.*
2. *DSA-framework is well-founded (i.e., acyclic).*
3. *If ψ is obligatory (resp. forbidden) by H with a situation Π, every argument with the complete situational knowledge (Π) has a derivation state $+$ (resp. $-$).*
4. *If an argument $\langle \delta, \pi, \sigma \rangle$ attacks $\langle \delta', \pi', \sigma' \rangle$, $\delta \not\succ \delta'$.*

Proof.

1. Property 1 holds because we assume that the background theory and the situation are consistent and there exists a minimal sub-base in the sub-base space, which is the empty constraint hierarchy.
2. Property 2 holds because we derive attacks from subset relation, which is a partial order.
3. Property 3 follows Definition 1 and Definition 4 since for every $\delta \in max(\Delta^{\Pi}), T_0 \cup \delta \cup \Pi \vdash \psi$ so every $\langle \delta, \Pi, \sigma \rangle \in AR, \sigma = +$. The forbidden can be proved analogously.
4. Suppose an argument $\langle \delta, \pi, \sigma \rangle$ attacks $\langle \delta', \pi', \sigma' \rangle$ and $\delta > \delta'$. Since δ is consistent with π and $\pi' \subsetneq \pi$, δ is consistent with π'. Together with $\delta > \delta'$, we can conclude that $\delta' \notin max(\Delta^{\pi'})$, contradicting Definition 4.

□

Next, we consider a specific condition, called *local optimality*, defined as follows.

Definition 6 (local optimality)**.** *Let T_0 be a background theory. A constraint hierarchy H corresponding with sub-base space $(\Delta, \geq)$ is* locally optimized *for a consequence ψ with respect to a situation Π if and only if the following conditions hold.*

1. *$\geq$ is a local preference.*
2. *Every maximal consistent subset of $H \cup \Pi$ is decisive with respect to ψ, i.e. if $S \subseteq H \cup \Pi$ is consistent and no other consistent $S' \subseteq H \cup \Pi$ such that $S \subsetneq S'$ then either $T_0 \cup S \vdash \psi$ or $T_0 \cup S \vdash \neg\psi$.*

If ψ is obligatory (resp. forbidden) by H with Π and H is locally optimized for ψ with respect to Π, then we say ψ is locally optimally obligatory (resp. forbidden).

For example, the consequence r in Example 1 is locally optimally forbidden because we use the local preference and every maximal consistent subset is decisive with respect to r. Under local optimality, we can simplify arguments in DSA-framework based on derivation states as follows.

Proposition 2. *If a consequence ψ is locally optimally obligatory by a constraint hierarchy H with a situation Π, a DSA-framework with respect to H, Π, and ψ has the following properties.*

1. *All arguments with derivation states (n) do not attack any arguments.*
2. *All arguments with derivation states (n) are attacked by some arguments.*
3. *All arguments with derivation states $(-)$ are attacked by some arguments.*

Proof.

1. Suppose an argument $\langle \delta^n, \pi^n, n \rangle$ attacks an argument $\langle \delta, \pi, \sigma \rangle$. We have that $\sigma \neq n$ and $\pi \subsetneq \pi^n$. There must be a soft constraint $c \in \delta \setminus \delta^n$ at level l such that $T_0 \cup \{c\} \cup \pi \vdash \psi$ (if $\sigma = +$) or $T_0 \cup \{c\} \cup \pi \vdash \neg\psi$ (if $\sigma = -$). From Definition 2, we have that $T_0 \cup \delta^n \cup \pi^n \nvdash \psi$ and $T_0 \cup \delta^n \cup \pi^n \nvdash \neg\psi$.Therefore, adding c into the level l of δ^n gets δ', which is consistent with π^n and $\delta' > \delta^n$ under the local preference, contradicting the fact that $\delta^n \in max(\Delta^{\pi^n})$ according to Definition 4.
2. Suppose an argument $\langle \delta^n, \pi^n, n \rangle$ is unattacked, we have two cases:
   (a) $\forall \langle \delta^\sigma, \pi^\sigma, \sigma \rangle \in AR$ with $\sigma \neq n$ [$\pi^n \not\subset \pi^\sigma$ or $\pi^n = \pi^\sigma$]: This case contradicts the fact that DSA-framework has such $\langle \delta, \Pi, + \rangle \in AR$ and no such $\langle \delta, \Pi, n \rangle \in AR$ according to Proposition 1.
   (b) $\forall \langle \delta^\sigma, \pi^\sigma, \sigma \rangle \in AR$ with $\sigma \neq n$ [$\exists \langle \delta', \pi', \sigma \rangle \in AR$ with $\pi^n \subsetneq \pi' \subsetneq \pi^\sigma$]: This case contradicts the facts that $S = \{\pi \mid \langle \delta^\sigma, \pi, \sigma \rangle \in AR$ and $\sigma \neq n$ and $\pi^n \subsetneq \pi\}$ is not empty since $\langle \delta, \Pi, + \rangle \in S$ and $\pi^n \neq \Pi$, S has a minimal element π^σ with respect to the set inclusion, and no $\pi' \in S$ such that $\pi^n \subsetneq \pi' \subsetneq \pi^\sigma$.
3. Suppose an argument $\langle \delta^-, \pi^-, - \rangle$ is unattacked, we have two cases:
   (a) $\forall \langle \delta^+, \pi^+, + \rangle \in AR$ [$\pi^- \not\subset \pi^+$ or $\pi^- = \pi^+$]: This case contradicts the fact that DSA-framework has such $\langle \delta, \Pi, + \rangle \in AR$ and no such $\langle \delta, \Pi, - \rangle \in AR$ according to Proposition 1.
   (b) $\forall \langle \delta^+, \pi^+, + \rangle \in AR$ [$\exists \langle \delta', \pi', + \rangle \in AR$ with $\pi^- \subsetneq \pi' \subsetneq \pi^+$]: This case contradicts the facts that $S = \{\pi \mid \langle \delta^+, \pi, + \rangle \in AR$ and $\pi^- \subsetneq \pi\}$ is not empty since $\langle \delta, \Pi, + \rangle \in S$ and $\pi^- \neq \Pi$, S has a minimal element π^+ with respect to the set inclusion, and no $\pi' \in S$ such that $\pi^- \subsetneq \pi' \subsetneq \pi^+$.

□

Corollary 3. *If a consequence ψ is locally optimally forbidden by a constraint hierarchy H with a situation Π, a DSA-framework with respect to H, Π, and ψ has the following properties.*

1. *All arguments with derivation states (n) do not attack any arguments.*

2. *All arguments with derivation states (n) are attacked by some arguments*
3. *All arguments with derivation states $(+)$ are attacked by some arguments.*

4 Explaining Preemption

In this section, we focus on explaining preemption with DSA-framework. Since DSA-framework is a specific type of abstract argumentation framework, we provide explanations using dispute trees in the same manner of other abstract argumentation based systems [4, 8, 9]. Referring to the original abstract argumentation framework [7], we use a term *AA-framework*, denoted by a pair $(\mathcal{A}, \mathcal{R})$ where $\mathcal{A}$ is a set whose elements are called *arguments* and $\mathcal{R} \subseteq \mathcal{A} \times \mathcal{A}$. For $x, y \in \mathcal{A}$, we say x attacks y if $\langle x, y\rangle \in \mathcal{R}$. We follow the definitions of dispute trees in AA-CBR [4] as follows.

Definition 7 (dispute tree)**.** *Let $(\mathcal{A}, \mathcal{R})$ be an AA-framework. A* dispute tree *for an argument $x_0 \in \mathcal{A}$, is a (possibly infinite) tree $\mathcal{T}$ with the following conditions.*

1. *Every node of $\mathcal{T}$ is of the form $[L : x]$, with $L \in \{P, O\}$ and $x \in \mathcal{A}$ where L indicates the status of proponent (P) or opponent (O).*
2. *The root of $\mathcal{T}$ is $[P : x_0]$.*
3. *For every proponent node $[P : y]$ in $\mathcal{T}$ and for every $x \in \mathcal{A}$ such that x attacks y, there exists $[O : x]$ as a child of $[P : y]$.*
4. *For every opponent node $[O : y]$ in $\mathcal{T}$, there exists at most one child of $[P : x]$ such that x attacks y.*
5. *there are no other nodes in $\mathcal{T}$ except those given by 1-4.*

A dispute tree $\mathcal{T}$ is an admissible *dispute tree if and only if (a) every opponent node $[O : x]$ in $\mathcal{T}$ has a child, and (b) no $[P : x]$ and $[O : y]$ in $\mathcal{T}$ such that $x = y$. A dispute tree $\mathcal{T}$ is a* maximal *dispute tree if and only if for all opponent nodes $[O : x]$ which are leaves in $\mathcal{T}$ there is no argument $y \in \mathcal{A}$ such that y attacks x.*

As DSA-framework is an abstract argumentation based system, similar to AA-CBR [4], we adapt the definitions from AA-CBR to provide novel explanations for why a consequence is obligatory or forbidden as follows.

Definition 8 (explanation)**.** *Explanations for why a consequence ψ is obligatory by a constraint hierarchy H with a situation Π are:*

- *any admissible dispute tree for every argument $\langle \delta, \{\}, +\rangle$ and for every argument $\langle \delta', \pi, +\rangle$ that attacks $\langle \delta'', \{\}, n\rangle$, and*
- *any maximal dispute tree for every argument $\langle \delta, \{\}, -\rangle$ and for every argument $\langle \delta', \pi, -\rangle$ that attacks $\langle \delta'', \{\}, n\rangle$.*

Explanations for why a consequence ψ is forbidden by a constraint hierarchy H with a situation Π are

- *any admissible dispute tree for every argument $\langle \delta, \{\}, -\rangle$ and for every argument $\langle \delta', \pi, -\rangle$ that attacks $\langle \delta'', \{\}, n\rangle$, and*
- *any maximal dispute tree for every argument $\langle \delta, \{\}, +\rangle$ and for every argument $\langle \delta', \pi, +\rangle$ that attacks $\langle \delta'', \{\}, n\rangle$.*

Figure 2 illustrates an explanation for why r ("drivers should overtake the other car") is forbidden by H in Example 1 with the situation $\Pi = \{p, q\}$. It demonstrates the preemption through the constraint hierarchy and incremental knowledge of the situation. This explanation can be interpreted into the following dialogue.

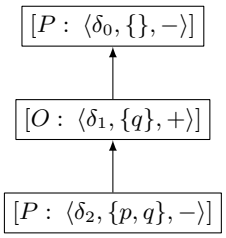

Figure 2. Explanation for why r is forbidden in Example 1 with the situation $\Pi = \{p, q\}$

P : Generally, drivers should not overtake the other car $\langle \delta_0, \{\}, -\rangle$
O : But, the other car appears obstructed in this situation so drivers should overtake the other car $\langle \delta_1, \{q\}, +\rangle$
P : But, the other car is in a danger zone in this situation so drivers still should not overtake the other car $\langle \delta_2, \{p, q\}, -\rangle$

On the other hand, Figure 3 illustrates an explanation for why r is obligatory by H in the same example but with the situation $\Pi = \{q\}$. This explanation is now a maximal dispute tree, unlike the previous explanation, which is an admissible dispute tree.

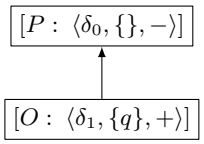

Figure 3. Explanation for why r is obligatory in Example 1 with the situation $\Pi = \{q\}$

Proving existence of explanations is based on several notions from abstract argumentation framework so we recap them as follows.

Definition 9 (from [7])**.** *Let $(\mathcal{A}, \mathcal{R})$ be an AA-framework, $E \subseteq \mathcal{A}$, and $x, y \in \mathcal{A}$.*

1. *E* attacks *x if some argument $z \in E$ attacks x.*
2. *E* defends *y if, for every $x \in \mathcal{A}$ that attack y, E attacks x.*
3. *E is* conflict-free *if no $x, y \in E$ such that x attacks y.*
4. *E is* admissible *if E is conflict-free and E defends every $z \in E$.*
5. *E is the* grounded extension *of the AA-framework if it can be constructed inductively as $E = \bigcup_{i \geq 0} E_i$, where E_0 is the set of unattacked arguments, and $\forall i \geq 0$, E_{i+1} is the set of arguments that E_i defends.*
6. *E is a* stable extension *of the AA-framework if it is a conflict-free set that attacks every argument that does not belong in E.*
7. *E is a* preferred extension *of the AA-framework if it is a maximal admissible set with respect to the set inclusion.*
8. *E is a* complete extension *of the AA-framework if it is an admissible set and every argument that E defends, belongs to E.*

Since DSA-framework is well-founded, the extension of the framework is unique, namely there is only one extension that is grounded, stable, preferred, and complete [7].

Proposition 4. *If a consequence ψ is locally optimally obligatory by a constraint hierarchy H with a situation Π, an extension E of DSA-framework $(AR, attacks)$ with respect to H, Π, and ψ has the following properties.*

1. *For every $\langle \delta, \{\}, +\rangle \in AR$, $\langle \delta, \{\}, +\rangle \in E$.*

2. *For every* $\langle \delta, \{\}, - \rangle \in AR$, $\langle \delta, \{\}, - \rangle \notin E$.
3. *For every* $\langle \delta, \{\}, n \rangle \in AR$,
   - *if it is attacked by* $\langle \delta^+, \pi^+, + \rangle \in AR$, $\langle \delta^+, \pi^+, + \rangle \in E$; *and*
   - *if it is attacked by* $\langle \delta^-, \pi^-, - \rangle \in AR$, $\langle \delta^-, \pi^-, - \rangle \notin E$.

Proof.

1. Suppose there is $\langle \delta, \{\}, + \rangle \in AR \setminus E$, there must be some $\langle \delta', \pi', - \rangle \in E$ that attacks $\langle \delta, \{\}, + \rangle$ because E is a stable extension. According to Proposition 2, $\langle \delta', \pi', - \rangle \in E$ must be attacked by $\langle \delta'', \pi'', + \rangle \in AR$. We have $\langle \delta'', \pi'', + \rangle \notin E$ since E is conflict-free: In this case, there must be $\langle \delta''', \pi''', - \rangle \in E$ attacks $\langle \delta'', \pi'', + \rangle$ and inductively we have that there must be some $\langle \delta^*, \pi^*, - \rangle \in E$ that is unattacked, contradicting Proposition 2.
2. Suppose there is $\langle \delta, \{\}, - \rangle \in E$. According to Proposition 2, there must be some $\langle \delta', \pi', + \rangle \in AR \setminus E$ that attacks $\langle \delta, \{\}, - \rangle$. Since E is admissible, there must be $\langle \delta'', \pi'', - \rangle \in E$ attacks $\langle \delta', \pi', + \rangle$ and inductively we have that there must be some $\langle \delta^*, \pi^*, - \rangle \in E$ that is unattacked, contradicting Proposition 2.
3. For every $\langle \delta, \{\}, n \rangle \in AR$, there are two cases according to Proposition 2:
   (a) It is attacked by $\langle \delta^+, \pi^+, + \rangle \in AR$. Suppose $\langle \delta^+, \pi^+, + \rangle \notin E$, we can prove in the same manner as 1. that there must be some $\langle \delta^*, \pi^*, - \rangle \in E$ that is unattacked, contradicting Proposition 2.
   (b) It is attacked by $\langle \delta^-, \pi^-, - \rangle \in AR$. Suppose $\langle \delta^-, \pi^-, - \rangle \in E$, we can prove in the same manner as 2. that there must be some $\langle \delta^*, \pi^*, - \rangle \in E$ that is unattacked, contradicting Proposition 2.

□

Corollary 5. *If a consequence* ψ *is locally optimally forbidden by a constraint hierarchy* H *with a situation* Π*, an extension* E *of DSA-framework* $(AR, attacks)$ *with respect to* H*,* Π*, and* ψ *has the following properties.*

1. *For every* $\langle \delta, \{\}, - \rangle \in AR$, $\langle \delta, \{\}, - \rangle \in E$.
2. *For every* $\langle \delta, \{\}, + \rangle \in AR$, $\langle \delta, \{\}, + \rangle \notin E$.
3. *For every* $\langle \delta, \{\}, n \rangle \in AR$,
   - *if it is attacked by* $\langle \delta^-, \pi^-, - \rangle \in AR$, $\langle \delta^-, \pi^-, - \rangle \in E$; *and*
   - *if it is attacked by* $\langle \delta^+, \pi^+, + \rangle \in AR$, $\langle \delta^+, \pi^+, + \rangle \notin E$.

Proposition 6. *If a consequence* ψ *is locally optimally obligatory by a constraint hierarchy* H *with a situation* Π*, there is an explanation for why* ψ *is obligatory by* H *with the situation* Π.

Proof. If ψ is obligatory by H with a situation Π and ψ is local optimal, every argument $\langle \delta, \{\}, + \rangle$ and every argument $\langle \delta', \pi, + \rangle$ that attacks $\langle \delta'', \{\}, n \rangle$ are inside the extension of DSA-framework with respect to H, Π, and ψ and every argument $\langle \delta, \{\}, - \rangle$ and every argument $\langle \delta', \pi, - \rangle$ that attacks $\langle \delta'', \{\}, n \rangle$ are outside the extension, according to Proposition 4. It is proved that there is an admissible dispute tree for every argument inside the extension and a maximal dispute tree for every argument outside the extension (see [5, 9]). □

Corollary 7. *If a consequence* ψ *is locally optimally forbidden by a constraint hierarchy* H *with a situation* Π*, there is an explanation for why* ψ *is forbidden by* H *with the situation* Π.

5 Discussion and Future Work

In section 3, we present an algorithm to find DS-arguments within the derivation state space. However, finding DS-arguments does not require exploring the entire space. Instead, we only need to find maximal sub-bases that are consistent with the current situational knowledge. The problem of finding such sub-bases is known as Partial MAX-SAT (PMSAT) [3]. PMSAT is a generalization of MAX-SAT problem [18] and decision versions of both problems are NP-complete [11]. Several PMSAT solvers have been developed to address this computational challenge [10, 11]. Following recent research [16] that explored norms as general constraint hierarchies, the problem in that setting would be more challenging. This is because general constraint hierarchies consider error functions that returns progressively larger values as satisfaction decreases [2]. This allows degrees of satisfaction rather than true or false, making the formalization of sub-bases more difficult than ours. This highlights extending DSA-framework to handle general constraint hierarchies, along with other representations of norms, as one interesting future work.

In section 4, we prove that if one consequence is locally optimally obligatory or forbidden, there is always an explanation for why it is. Unfortunately, the converse is not true. That is, if there is an explanation for why one consequence is obligatory or forbidden, it does not guarantee that the consequence is actually obligatory or forbidden. This behavior can arise due to conflicts between norms. Example 2 demonstrates one type of conflict where two norms within the same level have opposing enforcements on the same consequence.

Example 2. *Considering the constraint hierarchy* $H = \langle \{p \rightarrow \neg r, q \rightarrow r\} \rangle$ *and the situation* $\Pi = \{p, q\}$.

There are four sub-bases of H: (a) $\delta_0 = \langle \{p \rightarrow \neg r, q \rightarrow r\} \rangle = H$ (b) $\delta_1 = \langle \{p \rightarrow \neg r\} \rangle$ (c) $\delta_2 = \langle \{q \rightarrow r\} \rangle$ (d) $\delta_3 = \langle \{\} \rangle$ and under the local preference: $\delta_0 > \delta_1 > \delta_3$ and $\delta_0 > \delta_2 > \delta_3$. We have that $\Delta^{\Pi} = \{\delta_1, \delta_2, \delta_3\}$ because δ_0 is not consistent with Π. We also have that r is neither obligatory nor forbidden because $\delta_1, \delta_2 \in max(\Delta^{\Pi})$, $\delta_2 \cup \Pi \vdash \neg r$ and $\delta_3 \cup \Pi \vdash r$. the DSA-framework with respect to H, the situation $\Pi = \{p, q\}$, and a consequence r can be illustrated in Figure 4.

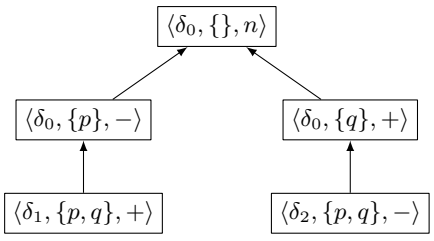

Figure 4. DSA-framework from Example 2

There is an explanation for why r is obligatory (Figure 5 left) as well as an explanation for why r is forbidden (Figure 5 right). However, r is neither obligatory nor forbidden as we have seen.

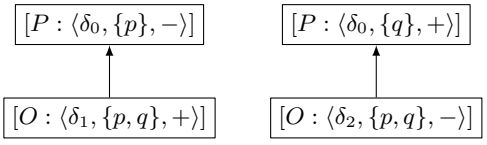

Figure 5. Explanations from Example 2

While this paper demonstrates the ability to explain why a consequence is obligatory, explaining why it is *not* obligatory remains

a challenge. This is because non-obligatory consequences can arise from either intentional permissions within the norms themselves or conflicts between norms. This highlights leveraging DSA-framework to automatically detect norm conflicts and explain non-obligatory consequences as another interesting future work.

6 Conclusion

This paper proposes the derivation state argumentation framework (DSA-framework) for explaining preemption in soft-constraint based norms represented as logical constraint hierarchies. The framework utilizes arguments that incorporate derivation states and the evolving knowledge of a situation. Under the local optimality, this approach guarantees explanations for why certain consequences are obligatory or forbidden, based on the properties of arguments within the DSA-framework and its extensions. Future research directions include leveraging DSA-framework to explain non-obligatory consequences, automatically detect norm conflicts, and extend its applicability to handle general constraint hierarchies and other normative representations.

Acknowledgements

This work was supported by JSPS KAKENHI Grant Number, JP22H00543, JST, AIP Trilateral AI Research, Grant Number, JPMJCR20G4, and the MEXT "R&D Hub Aimed at Ensuring Transparency and Reliability of Generative AI Models" project.

References

[1] S. Benferhat, D. Dubois, and H. Prade. Argumentative inference in uncertain and inconsistent knowledge bases. In *Uncertainty in Artificial Intelligence*, pages 411–419. Elsevier, 1993.

[2] A. Borning, B. Freeman-Benson, and M. Wilson. Constraint hierarchies. *LISP and symbolic computation*, 5(3):223–270, 1992.

[3] B. Cha, K. Iwama, Y. Kambayashi, and S. Miyazaki. Local search algorithms for partial maxsat. In *Proceedings of the fourteenth national conference on artificial intelligence and ninth conference on Innovative applications of artificial intelligence*, volume 263–268. Citeseer, 1997.

[4] K. Cyras, K. Satoh, and F. Toni. Abstract argumentation for case-based reasoning. In *Fifteenth International Conference on the Principles of Knowledge Representation and Reasoning*, pages 243–254, CA, USA, 2016. AAAI Press.

[5] K. Cyras, K. Satoh, and F. Toni. Explanation for case-based reasoning via abstract argumentation. In *Computational Models of Argument*. IOS Press, 2016.

[6] L. Dennis, M. Fisher, M. Slavkovik, and M. Webster. Formal verification of ethical choices in autonomous systems. *Robotics and Autonomous Systems*, 77:1–14, 2016.

[7] P. M. Dung. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. *Artificial intelligence*, 77(2):321–357, 1995.

[8] P. M. Dung, R. A. Kowalski, and F. Toni. Dialectic proof procedures for assumption-based, admissible argumentation. *Artificial Intelligence*, 170(2):114–159, 2006.

[9] P. M. Dung, P. Mancarella, and F. Toni. Computing ideal sceptical argumentation. *Artificial Intelligence*, 171(10-15):642–674, 2007.

[10] M. El Bachir Menai and T. N. Al-Yahya. A taxonomy of exact methods for partial max-sat. *Journal of Computer Science and Technology*, 28: 232–246, 2013.

[11] Z. Fu and S. Malik. On solving the partial max-sat problem. In *International Conference on Theory and Applications of Satisfiability Testing*, pages 252–265. Springer, 2006.

[12] W. Fungwacharakorn, K. Tsushima, and K. Satoh. Debugging constraint hierarchies representing ethical norms with valuation preferences. In *Proceedings of Workshop on AI Compliance Mechanism (WAICOM 2022)*, pages 114–124, 2022.

[13] W. Fungwacharakorn, K. Tsushima, and K. Satoh. Fundamental revisions on constraint hierarchies for ethical norms. In *Legal Knowledge and Information Systems*, pages 182–187. IOS Press, 2022.

[14] W. Fungwacharakorn, K. Tsushima, H. Hosobe, H. Takeda, and K. Satoh. Connecting rule-based and case-based representations of soft-constraint norms. In *Legal Knowledge and Information Systems - JURIX 2023*, volume 379 of *Frontiers in Artificial Intelligence and Applications*, pages 149–154. IOS Press, 2023.

[15] J. Greene, F. Rossi, J. Tasioulas, K. B. Venable, and B. Williams. Embedding ethical principles in collective decision support systems. In *Thirtieth AAAI conference on artificial intelligence*, 2016.

[16] H. Hosobe. and K. Satoh. A soft constraint-based framework for ethical reasoning. In *Proceedings of the 16th International Conference on Agents and Artificial Intelligence (ICAART 2024)*, pages 1354–1361, 2024.

[17] R. Kowalski and K. Satoh. Obligation as optimal goal satisfaction. *Journal of Philosophical Logic*, 47:579–609, 2018.

[18] M. W. Krentel. The complexity of optimization problems. In *Proceedings of the eighteenth annual ACM symposium on Theory of computing*, pages 69–76, 1986.

[19] H. Prakken, A. Wyner, T. Bench-Capon, and K. Atkinson. A formalization of argumentation schemes for legal case-based reasoning in ASPIC+. *Journal of Logic and Computation*, 25(5):1141–1166, 2015.

[20] T. Racharak and S. Tojo. On explanation of propositional logic-based argumentation system. In *Proceedings of the 13th International Conference on Agents and Artificial Intelligence (ICAART 2021)*, pages 323–332, 2021.

[21] K. Satoh and A. Aiba. Computing soft constraints by hierarchical constraint logic programming. *Transactions of Information Processing Society of Japan*, 34(7):1555–1569, 1993.

[22] K. Satoh, J.-G. Ganascia, G. Bourgne, and A. Paschke. Overview of recomp project. In *International Workshop on Computational Machine Ethics, International Conference on Principles of Knowledge Representation and Reasoning*, 2021.

Analyzing the baseline for harmonized standards – a systematic review of standards on bias and data quality

Anna Schmitz[a, b, *] **and Maximilian Poretschkin** [a, b, c]

[a]Fraunhofer-Institute for Intelligent Analysis and Information Systems IAIS, Sankt Augustin, Germany
[b]Lamarr-Institute for Machine Learning and Artificial Intelligence
[c]University of Bonn
ORCID (Anna Schmitz): https://orcid.org/0000-0001-8801-3700

Abstract. The recently adopted European AI Act mandates many AI providers to implement data quality and bias mitigation in their systems in order to safeguard fundamental rights, particularly non-discrimination. From a computer science perspective, however, the relevant requirements in the AI Act are not clearly linked to specific metrics or methods, highlighting the need for concrete interpretation within real-world applications. This issue might be partially solved by the formulation of ten harmonized standards which are requested by the European Commission in order to further specify the technical requirements and ensure legally compliant implementation in practice. Notably, the development of these standards is likely to leverage existing standardization results.

This paper presents a systematic review of all relevant international standards to explore how the requirements regarding fairness and non-discrimination outlined in the AI Act can be operationalized on this basis. We extracted from these standards specifications regarding data quality and bias concepts, guidance for their implementation and measurement, as well as indications for dealing with trade-offs between conflicting requirements. Our analysis confirms two prominent trends: i) group- and accuracy-focused bias measurement, ii) emphasis of the contextual considerations and stakeholder needs for operationalizing requirements. In addition, we observed a broad array of bias mitigation approaches, surpassing the AI Act requirements. However, we also identified several weaknesses such as inconsistencies across different standards. In conclusion, by giving a comprehensive overview of the current standardization landscape regarding bias and data quality, pointing out weaknesses therein and possible ways to address these, our review serves as a valuable resource for current standardization efforts in support of the European AI Act.

1 Introduction

1.1 The role of standards for implementing the European AI Act

The European AI regulation (AI Act) has recently come into force [38]. As stated in the first recital, it aims to ensure protection of health, safety and fundamental rights as enshrined in the EU Charter. This particularly includes the objective of fairness and non-discrimination (recital 27), which is the focus of this paper. Following the risk-based approach of the AI Act, potential adverse impact on non-discrimination is of particular relevance when classifying an AI system as "high risk" (rec. 48) and can also pose a systemic risk in general-purpose AI models (rec. 110). Furthermore, high-quality data is cited as a vital lever to ensure that AI systems do "not become a source of discrimination prohibited by Union law" (rec. 67). Notably, the recitals of the Act form the framework within which the requirements in the legal text are to be interpreted. The relevant regulatory requirements with regard to fairness and non-discrimination in the main part of the AI Act are accordingly closely related to data quality, see Table 1.

Table 1. Relevant requirements in the AI Act related to fairness and non-discrimination, extracted from [38].

Subject	Requirement	Article
training, validation and testing data sets of high-risk AI systems	shall be examined "in view of possible biases that are likely to [...] lead to discrimination prohibited under Union law, especially where data outputs influence inputs for future operations"	10, 2.f)
	"appropriate measures to detect, prevent and mitigate possible biases [shall be] identified"	10, 2.g)
	"shall be relevant, sufficiently representative, and to the best extent possible, free of errors and complete in view of the intended purpose"	10, 3.
instructions for use for deployers of high-risk AI systems	shall contain "when appropriate, [the AI system's] performance regarding specific persons or groups of persons on which the system is intended to be used"	13, 3.b.v)
high-risk systems that continue to learn during operation	"eliminate or reduce as far as possible the risk of possibly biased outputs influencing input for future operations ('feedback loops')"	15, 4.
Documentation of general-purpose AI models	shall contain information on "the data used for training, testing and validation (...) including the type and provenance of data and curation methodologies" as well as „methods to detect identifiable biases, where applicable"	Annex XI, Sec. 1, 2.c)

Technical standards create a basis for quality assurance and assessment of AI systems. Following alignment with the new legislative framework, also European product legislation emphasizes con-

* Corresponding Author. Email: anna.schmitz@iais.fraunhofer.de

formity assessment based on standards by accredited bodies as a common tool for internal market surveillance [36, 32]. Standardization correspondingly plays a key role for the implementation of the AI Act, both with respect to “high-risk” AI use cases and general-purpose models. Specifically, it can describe technical solutions to ensure compliance with the AI Act (see rec. 121 and Article 40 [38]). So-called "harmonized" standards which are adopted by the European Commission (EC) and published in the Official Journal of the EU [39] can be particularly effective in operationalizing the technical requirements contained in the AI Act. While compliance with harmonized standards is not mandatory, it does give rise to a presumption of conformity with the regulatory requirements covered (see Art. 40 and 41 [38]) and therefore offers greater legal certainty and is cheaper than other ways of demonstrating conformity [35, 32, 55].

The two European standardization organizations[1] CEN and CENELEC have set up the joint committee JTC 21 which is actively working on standardization deliverables in support of the AI Act as requested by the EC [37, 60]. The requirements in the AI Act related to fairness and non-discrimination will most probably be covered in the context of data governance and data quality, and possibly risk management, accuracy and robustness. It is anticipated that the efforts by JTC 21 will leverage existing results from international AI standardization [60, 59]. However, the adoption of the requested standardization deliverables as harmonized European standards depends on their ability to capture viable technical solutions and state-of-the-art practices ensuring compliance with the objectives of the AI Act [40]. Should there be concerns regarding safety or fundamental rights, for example, the EC may itself define so-called "common specifications" (see Art. 41 [38] and [35, 32]). In order to ensure the success of the current standardization work – namely, offering providers and users concrete, practical methods and criteria for implementing conformity with the AI Act requirements – it is therefore essential that the deliverables are consistent with and sufficiently safeguard (according to the state of the art) fairness and non-discrimination. To contribute a useful baseline, we aim to understand through a systematic analysis what content and guidance the existing standards provide in this regard.

1.2 Necessity and research questions for a systematic review of standards

The operationalization of fairness must take the entire life cycle into account: This includes defining a concept of fairness and identifying attributes by which potential discrimination must be avoided, examining data and models for bias, and monitoring application dynamics during operation, among other things [52, 29, 31, 57]. Furthermore, AI systems are typically embedded in organizations and specific application contexts, so that organizational and procedural measures also play an important role in ensuring fairness. These include, for example, the involvement of stakeholders in mapping the notion of fairness to specific metrics, options for complaints or redress, and the composition of diverse development teams [48, 50, 31, 29, 57].

As far as technical implementation is concerned, the requirements for data quality and bias mitigation in the AI Act (see Table 1) are broadly defined from a computer science perspective and, especially, not clearly linked to specific metrics or technical measures. Concretization of these requirements must be based on the specific AI application context [40, 35, 43]. However, various implementation and measurement methods exist which can differ greatly in their effects on the data, model quality, and bias mitigation.[2] This raises the question for developers and providers of regulated AI systems, as to which interpretation of the requirements is best aligned with the objectives of the AI Act. Further clarification of the fairness-related requirements and their targets of evaluation is needed. Overall, it would be desirable to identify specific measures for individual AI use cases with which the requirements can be achieved. It would also be helpful to be able to objectively determine the degree of conformity, ideally by using quality metrics. In the case of bias detection and mitigation decisions, however, these are normative questions in particular, which are not answered in the computer science literature. This leads us to the first research question (RQ) which aims to gain an insight into the content of the upcoming harmonized standards.

RQ1: *To what extent are the fairness-related requirements in the AI Act substantiated by existing standards, in the sense that a more precise explanation or concretization of target concepts is provided?*

Moreover, the available state of the art for detecting and mitigating bias should be leveraged as widely as possible in order to ensure legally compliant implementation in practice with regard to fairness and non-discrimination. In this respect, the focus of the fairness-related requirements on data quality may be a potential weakness of the AI Act.[3] Notably, there are no universal or generalizable findings on the exact effects of data quality measures in terms of mitigating model biases (nor on interactions with other model requirements such as accuracy, see below). In contrast, the literature shows that the effectiveness of different data pre-processing methods to avoid model biases varies and that these are rarely completely prevented or only in rare cases completely eliminated. The examination and, if necessary, mitigation of biases across different development phases such as design and modeling, on the other hand, are additional key building blocks to safeguard against undesirable biases. Harmonized standards can be an opportunity to emphasize this more clearly.

Furthermore, conflicts may arise between different requirements in the AI Act but it leaves largely open as to how these should be weighed up. Still, prioritization and trade-offs between system requirements often have to be made in the development process, which then significantly influence the choice of mitigation strategies for risks and thus the overall safeguarding of the AI application [58]. One specific example is the relationship between freedom from errors and bias mitigation.[4] Several measures against biases are based on the modification of the training data (e.g. by creating new data representations or changing labels in the data). Furthermore, their application typically also influences the accuracy of the resulting model if it is operated or tested on (unchanged biased) real data. It remains unclear whether modifications of the underlying data are permissible if this yields better bias reduction than other mitigation methods, for example, and how potential interactions between accuracy and freedom from bias should be weighed up. Corresponding guidelines are

[1] See 3.1 for a brief overview on standardization (organizations).

[2] For example, potentially conflicting interpretations exist of data representativeness. One related concept is the similarity of the distributions of training and production data, another is the broadest or most diverse possible coverage of different groups or segments of the application domain. While the first can strengthen model performance within the production distribution (as the model learns majority groups particularly well), the latter can improve model quality for marginalized groups and reduce "overfitting". For a comprehensive literature survey of the operationalization of data relevance, completeness, representativeness, “freedom from errors”, and bias examination in AI systems, see section VI.4. in [40].

[3] For a detailed discussion, please refer to section VIII.1. in [40].

[4] For high-risk systems, the AI Act requires freedom from errors with regard to the data (Article 10) and the model in the sense of an "appropriate level of accuracy” (Article 15). For a detailed discussion of the potential conflict with bias mitigation, please refer to section VIII.2. in [40]

currently missing that may ensure legally compliant implementation in practice. The described weaknesses of the AI Act lead us to the second research question.

RQ2: *To what extent do the existing standards provide (sufficient) safeguards and technical solutions for fairness and bias mitigation in AI systems?*

In this paper, we present the results of a systematic review of standards that we have conducted in order to address the two research questions outlined above. Based on the review, we aim to understand in advance how the requirements in the AI Act with regard to fairness and protection against discrimination could be operationalized on the basis of existing standardization results. In this way, we provide an outlook on the possible content of upcoming harmonized standards and, at the same time, help to identify and address possible gaps or inconsistencies with the orientation of the AI Act at an early stage.

2 Related Work

Bias and data quality is an active field of research. The related work encompasses several surveys that compile and structure the state of the art in terms of metrics and implementation methods [52, 62, 29, 42, 40], as well as systematic reviews of existing guidelines and frameworks for trustworthy AI [31, 48, 51, 58]. These overviews are particularly valuable in identifying common elements, best practices and possible gaps in the various technical approaches. Furthermore, there is relevant interdisciplinary work that examines selected technical standards or methods with regard to conformity with EU law and analyses alignment challenges, in part specifically for fairness requirements [44, 63, 46, 45, 47, 41]. More fundamentally, there is also research on the role and processes of standardization with regard to the operationalization of certain aspects of the AI Act, in particular investigating their legitimacy and ability to ensure ethical requirements and fundamental rights (e.g. in terms of the stakeholders and expertise involved) [43, 55, 50, 30, 35]. However, to the best of our knowledge, there has been no systematic analysis of which of the existing definitions and measures for bias and data quality are actually reflected in current standards.

In addition, we identified some interesting non-peer-reviewed resources that are related to our work. The report [60] analyzes the existing standards considered by JTC 21 as part of its preliminary AI standardization work plan with regard to their coverage of the European Commission's draft standardization request. Especially, it points to alignment challenges such as the risk notion used in international standards with the risk notion in the AI Act.

Besides that, we identified several relevant reviews of standards. The report [54] provides a "standardization landscape". Its analysis is to a large part conducted by deriving keywords from the „high-risk" requirements in the draft AI Act and calculating an operationalization index based on the co-occurrence of these keywords in each standard. The report further records characteristics such as domain generality and maturity of a standard, that may indicate its suitability for operationalizing the AI Act requirements. As a result, it shows which standards are particularly relevant for each "high-risk" requirement and what gaps exist (in terms of a lack of standards that mention essential topics). An "Update" of this Standardization Landscape was published in 2023 that deals with eight selected IEEE standards [61]. The analysis of these standards is based on expert judgement and covers additional criteria such as "AI coverage" and "fit within standardization landscape". For three standards with the highest coverage of AI Act requirements, the report discusses how they may complement relevant ISO/IEC standards and provides a dedicated section on bias, see 5.1 in [61]. The review [28] categorizes AI standards according to their relevance for selected AI application domains such as HR- and talent management and also addresses non-discrimination. Lastly, standardization roadmaps e.g. [33] and other reports based on the consultation of standardization experts e.g. [27, 53] also provide overviews of existing standards.

While the aforementioned reviews are not fully systematic, they highlight the need for research, identify gaps and potential for standardization, and formulate recommendations in this regard. Also, they are broader in scope and their research question is more general compared with the review presented in this paper. Specifically, our analysis focuses on fairness-related requirements and, instead of examining whether all aspects of these requirements are mentioned or discussed in principle in the existing standards, we analyze the quality and the way in which they are discussed or specified.

3 Method

The study has been undertaken as a systematic review of standards based on the guidelines for systematic reviews as proposed by Kitchenham [49]. For simplicity, we use the term „standard" in the following to summarize different types of standardization documents (published by a standardization organization), such as technical standards, technical specifications, technical reports and draft versions of standards currently under development.

3.1 Scope of the review

To justify our approach of selecting relevant standards, we start with an overview of standardization and its main actors. From a European perspective, standardization work can be carried out at national, European and international level, depending on its relevance (e.g. economic) and target group. In all three scenarios, the standards are developed by experts from national standardization organizations, who may be send by their organizations to work at higher levels. The major organizations which provide valid standards (i.e., developed in a consensus-based process with stakeholder participation) are CEN, CENELEC and ETSI at European level and ISO, IEC and ITU at international level. In addition, there are several international consortia (e.g. IEEE, CSA, OGC) that develop relevant technical specifications or industry standards while not in a fully consensus-based process. For a comprehensive overview of the players and interrelationships in standardization, please refer to section 3.2 and Table 15 in [33].

The central body for European AI standardization is the joint committee CEN/CENELEC JTC 21 "Artificial Intelligence". In particular, CEN and CENELEC were mandated by the European Commission to develop ten standardization deliverables in support of the AI Act [37] and the work of these deliverables is coordinated within JTC 21 [33, 60]. It is anticipated that JTC 21 will leverage results of existing international standardization efforts to address the request [60, 59]. Notably, valid international standards bear the advantage that their content is already internationally aligned. Since ETSI is excluded from the Commission's request [37], we only consider the European standards bodies of CEN and CENELEC in scope of this review. As ITU is the analog of ETSI at international level [33], we therefore only consider international standards by ISO and IEC.

Having determined the relevant bodies, we finally narrow down the type of the standards we aim to analyze. Firstly, we naturally focus on such standards that contain specifications or guidance with respect to (some aspect of) the fairness-related requirements in the AI Act. As explained in 1, this especially comprises data quality for AI

systems, assessment, testing and mitigation of bias in data or models, as well as prioritization or trade-offs related to bias mitigation and other AI trustworthiness requirements. Notably, the AI Act pursues a horizontal approach, meaning that its requirements are supposed to be applied to AI systems regardless of the specific use case or domain in which they are deployed. Therefore, we secondly exclude all standards which refer to specific domains or use cases.

3.2 Search process and selection of standards

In order to identify standards that may operationalize the fairness-related requirements in the AI Act, we conducted a keyword search. We used Nautos to this end, which is a standards management and search service. It comprises, among others, filtering options for regularly updated versions of the entire bodies of European standards, ISO and IEC standards [34], thus covering our scope as described in 3.1. We used 11 keywords (and combinations) for the search. The keywords are in part derived from our research questions (i.e., AI bias, risk, assessment, data quality, trade-off) and complemented with more general keywords associated with the topics of AI fairness and bias (i.e., AI quality, trustworthiness, impact, evaluation, fair, ethic) in order to identify all standards possibly relevant to the research questions, even if they use different wording. All search results were manually filtered based on title and abstract to determine whether they fall within our scope. In particular, we excluded non-horizontal documents which refer to specific use cases (e.g. biometric identification, image recognition) or domains (e.g. health-care). A dedicated quality assessment during the search and selection process as suggested in [49] was not conducted, since we aim to review all relevant, existing standards. Our research questions relate precisely to the quality of the content of the identified standards, see 4. The search and selection of standards was conducted and completed in March 2024. Eventually, 35 standards were selected for in-depth analysis and data extraction.

3.3 Data extraction and analysis

As a basis for analyzing the content of the standards with regard to RQ1, we extracted from each standard any:

- Specifications of the bias notion; definition of target concepts for bias mitigation,
- Specifications of data quality; definition of target concepts for representativeness, relevance, freedom from errors, completeness,
- Guidance or instructions for the selection of (bias or data) metrics.

Regarding the analysis of RQ2, we recall that this is motivated in 1.2 by the focus of the AI Act on data quality as a specific lever to mitigate biases. This naturally emphasizes the data as main subject and the inception and training of an AI system as main lifecycle stages where mitigation measures should be taken. However, the state of the art is much broader (see 1.2 and the references in 2) and conflicts between bias mitigation and other system requirements may need to be dealt with. In order to investigate RQ2, we therefore extracted from each standard the information it provides on:

- Subject of the measurement or testing of bias (e.g. data, model)
- Stage/phase in the AI life cycle in which the (bias or data quality) measurement or evaluation should be carried out
- Subject of the bias mitigation measures (e.g. data, model)
- Guidance or instructions for the selection of implementation measures (for bias mitigation or data quality)
- Stage/phase in the AI life cycle in which the (bias or data quality) measures should be implemented
- Guidance on trade-offs between requirements or on their approval

To synthesize and evaluate the manually extracted information, we used a tabular representation. We also took into account the different categories of standards e.g. whether they are AI-specific standards or standards from the classic context of software engineering or IT products, or whether they relate specifically to bias, data quality, general AI development/management processes or even originate from the safety/robustness context.

3.4 Limitations

JTC 21 can of course incorporate its own content to the standardization deliverables, for example contributed by European national activities, and does not necessarily have to draw on standards which are internationally aligned already. It is a natural limitation of this review that this cannot be anticipated by the review itself. Another limitation which affects the presentation and easy traceability of our results is that the standards analyzed are not freely accessible for copyright reasons [30, 53].

4 Results

We first summarize in 4.1 our findings regarding the specification of bias and data quality concepts in the existing standards according to RQ1. With regard to RQ2, we then outline in 4.2 the guidance we have identified for their implementation, including prioritization in the case of conflicting requirements.

4.1 Specification of target concepts

We have discerned two overarching trends within the current standards regarding the process of defining specific requirements for bias mitigation and data quality. Firstly, there is a consistent emphasis on the importance of consulting or at least considering various stakeholders in the identification of requirements [4, 1, 3, 24, 17, 18, 22, 12, 22, 26, 16]. Secondly, there is a consensus that the application context and domain (comprising factors such as the operating conditions and target population) must be taken into account when determining what constitutes bias in an AI system and selecting appropriate metrics [4, 1, 24, 26] . Similarly, the specific data quality characteristics depend on the particular purpose and context of use [1, 10, 20, 22, 12, 24] . In the following subsections, we go into more detail on the notion of bias and the requirements of "freedom from errors", completeness, relevance, and representativeness of data stated in the AI Act, see Table 1.

4.1.1 Bias

The "systematic difference in treatment of certain objects, people, or groups in comparison to others" is the predominant definition of bias in ISO/IEC standards, see Table 2. Only some additional (non-defining) explanations put more emphasis on the impact of bias on the individual (e.g. in [4] or in the annex of [26] which requires a "denial of opportunity and/or assignment of an undesirable outcome for the user" so that a system is biased). Beyond a general bias definition, many standards additionally describe various, more specific types, sources and impacts of bias [4, 1, 3, 15, 26]. Here, in many instances a connection with data characteristics such as representativity and coverage is established (e.g. sampling bias, selection

bias, unequal representation of different scenarios or input conditions [4, 1, 25]). Moreover, demographics such as age, sex or residence, as well as ethnicity are repeatedly cited in the relevant standards as concrete examples of such biases. Still, "at-risk groups" can also encompass features not overtly evident in the data [4, 1], or a combination of multiple sensitive features which is referred as "intersectional fairness" [4]. Overall, the existing standards show a clear emphasis on the concept of group-fairness [62] to identify or measure biases [4, 1, 5, 8, 24, 18]. In addition, decreased accuracy or functional correctness of the AI system (for different groups) is consistently mentioned as a major or even main issue of bias [1, 13, 5, 17, 25, 8, 4], which further indicates a tendency towards performance-/accuracy-related fairness-concepts. Even with respect to ensuring equal allocation of opportunity [18, 26], the equal opportunity metric described in [4] essentially requires for a classification model that its true positive rates are equal across demographic groups. Few standards highlight other target concepts for bias mitigation, such as demographic parity [4] or the minimization of stereotyping [18, 4].

Table 2. Bias definitions in existing standards

„systematic difference in treatment of certain objects, people, or groups in comparison to others“	[4, 1, 13, 8, 19]
"favouritism towards some things, people or groups over others“	[5]
"measure of the distance between the predicted value provided by the ML model and a desired fair prediction"	[15]

4.1.2 Data quality

Data quality in the context of software products is defined as the "degree to which the characteristics of data satisfy stated and implied needs when used under specified conditions" [10]. The AI Act requirements of "freedom from errors" (in terms of accuracy) and completeness of data have already been established in this conventional data quality model [10]. Representativeness and relevance, in contrast, have only become important in AI-specific standards,[5] see in particular the 5259 series of standards on data quality for Machine Learning [20, 21, 22, 23]. It is worth noting that the definition of data quality is also changed in AI-specific standards to "characteristic of data that the data meet the organization's data requirements for a specified context" [20, 1]. Moreover, some standards on AI have extended the conventional data quality characteristics with additional interpretations (e.g. completeness in the sense of “data coverage”), as explained below. Lastly, although standards do exist that describe metrics or measurement functions for data quality properties [21, 11], it should be remarked that these typically still need to be tailored to the specific context of use. For instance, as a measure of feature relevance, the ratio between the number of features deemed relevant and the total number of features in the dataset is cited [21]. The following paragraphs give an overview of further explanations and interpretations that we identified in our review with respect to each of the four requirements.

"Freedom from errors" is treated in existing standards in terms of data accuracy. This is specified as the degree to which data values "correctly represent the true value of the intended attributes" [10] or, simpler, to which each data item has "the correct data value" [21]. AI-specific standards consistently emphasize the labels and the labeling process as a major subject of errors [4, 23, 15, 25, 19]. Taking into account that there might be uncertainty about the "true" value of a data item, quantification approaches such as the inter-annotator variation are suggested [19]. In addition, data accuracy can be affected by timeliness [4, 1] and outliers [10, 11, 2]. Especially, [2] mentions different sources for inaccurate data such as measurement and recording errors, and elaborates on the relationship between errors and outliers and their potential effects on the accuracy of a model.

Completeness of data is predominantly described in the classical sense as the opposite of missing values e.g., within data records, features or labels [4, 1, 10, 11, 21, 15]. In addition, completeness can be considered at the level of features i.e., whether critical features are generally missing in the dataset [19, 11], and, similarly, at the level of records [11]. Other interpretations of completeness we identified, refer to i) the *coverage* of all "expected" data values [11] (e.g. whether records in a creditscoring dataset cover all possible income cohorts), as well as ii) the "expected" *occurrence* of a given data value for the domain [21] (e.g. whether the proportion of female records is as expected for loan applications). The first corresponds to the concept of domain coverage, requiring a complete as possible context description from which training or test data can be derived to include all relevant scenarios [13, 3, 25]. The second refers to the data distribution. Notably, several standards point to connections between completeness and representativeness of data through formulations such as „complete (broad and varied) to be representative of the expected production data“ [24], or „incomplete representation of input domain is tied to data drift“ [25]. However, a clear and consistent specification of this connection is not recognizable across the reviewed standards (see also the last paragraph).

Relevance as a data property is the least clearly specified of the four considered AI Act requirements. Only [21] gives an explicit definition of data relevance as "the degree to which a dataset (...) is suitable for a given context". Beyond that, the term is mainly explained implicitly or by way of example, whereby we have identified three basic interpretations across the various standards. Firstly, relevance is mentioned in the context of (global) feature importance and feature selection with the aim that features facilitate good predictive power for the AI system [21, 4, 3]. In this respect, statistical correlation tests of features with the target variable [21] and feature selection methods such as different types of regression or random forests are mentioned [4]. Second, in the context of robustness- or safety-specific standards, feature relevance is also considered locally in terms of which inputs are most significant for the model output [7, 25]. This interpretation falls into the area of explainability methods (e.g. heatmaps), which can be used for model validation. The third interpretation relates to entire data sets and aims to ensure that an ML model learns all "relevant content" in relation to the context of use and that this can be tested [19, 6] . Relevance here means that the data spans the target distribution and covers all critically important scenarios that can be expected during operation (e.g. relevant adversarial examples, attacks, or distributional shifts) [6] . While no general measurement approaches are pointed out in this regard, reference is made to human assessment of relevance and techniques using intermediate representations [6].

Representativeness is distinguished as one of the most crucial data properties for Machine Learning [20]. It is defined as the degree to which the dataset reflects the target population under study, with AI-specific standards interpreting this predominantly in terms of the reflection of production data [21, 7, 5, 20, 1, 24, 15, 19]. On the one hand, several standards refer specifically to *distribution* properties in this respect or suggest comparing the distributions of training or test data and production data [1, 24, 15, 19, 25]. In addition, the

[5] Similar to other characteristics such as data balance, coverage, and various types of data bias as outlined in 4.1.

data sampling and selection procedures are mentioned as potential sources of non-representativeness and model bias, e.g. if the data are not sampled uniformly at random [4, 23, 1]. On the other hand, several standards pay attention to the coverage of all important *attributes* and different groups of the population represented in the production data e.g. geographical or demographic [23, 8, 18, 1, 21]. Thus, unlike the similarity of distributions, the proportion of attributes in the data compared to all relevant attributes of the population is also described as a measure of representativeness [21]. Especially, [1] associates the notion of representative training and test data with the aim to achieve and verify "an acceptable level of functional correctness for the target population". Similarly, the formulation of representativeness is also used in relation to the operational domain of use, which is likewise characterized by attributes of the environment in order to capture the input space on which the AI system is supposed to work well [17, 7, 25]. In this spirit, some standards mention the deviation from the operational distribution (e.g. up-sampling of underrepresented groups in the training data) as a possible measure for reducing model bias [4, 20, 15] and indicate that representativeness can be affected by data balance (i.e., even distribution of label-values or samples accross groups) [1, 21, 6]. In synthesis, the objective of bias mitigation through representative data can mean, according to some standards, that certain groups should be represented more strongly in the training data if this enables a more balanced performance of the model on all relevant groups. In many cases, however, this could contradict the similarity of training and production data distributions, as described in other standards under the term representativeness. Finally, we have identified a last interpretation in the sense of "representative examples" [19, 26], which is largely independent of the concepts already discussed. Here, representativeness is seen as an approach to explainability in order to mark data points that provide an insight into the nature of the (training or operational) data or illustrate typical errors of the model for human reviewers. Overall, we have extracted various concepts of representativeness from the standards and there is currently no consistent direction for implementing this property. Harmonized standards therefore have the potential to create much more clarity for providers, for example by further specifying which representativeness concept is most suitable for which AI technologies or tasks, see 5.

4.2 Guidance on implementation

Compared to the data focus of the AI Act, the existing standards pay additional attention to the mitigation and examination of biases in the AI system itself. Moreover, trade-offs between specific requirements are barely discussed, but some guidance is provided on requirements identification and acceptance, predominantly in the sense that stakeholders may be involved in the process in some way but not necessarily in the final approval. The following sections give more detailed insight into the corresponding explanations in the relevant standards.

4.2.1 Methods and Measurement

Regarding the mitigation of biases, the relevant standards provide a broad overview of various implementation approaches with different target entities across the life cycle; for a list of specific methods, see particularly [4, 1]. The overall picture of the existing standards makes evident the need to consider bias mitigation in inception, design and data preparation including labeling [4, 1, 8, 24, 15, 25], in the training algorithm and modeling [4, 1, 5, 8, 25, 26], in the post-processing of trained models [4, 1], as well as during operation e.g. in the form of re-training or re-engineering of an AI system [1, 24, 17, 15, 25]. It is also noted that the removal of sensitive features usually does not provide sufficient mitigation, for example due to proxies in the data [24, 15, 4]. Lastly, while the standards typically summarize (established) approaches and procedures according to the state of the art, we have also identified two aspects in which further specification would be helpful. First, FMEA or PFMEA with regard to bias is mentioned in two standards [9, 25], while it is not obvious from a computer science perspective how to break to down biases and their effects to the (smallest) component-level of an AI system (e.g. a neuron) nor how to interpret the outcome of this [56, 57]. Second, one standard states that being transparent about bias instead of removing it may be a possible mitigation measure [15]. In this aspect, it is questionable whether this is consistent with the objectives of the AI Act and it would be helpful to clarify through harmonized standards the extent to which this interpretation may be permissible, for example due to trade-offs with other system requirements 4.2.2.

The implementation of data quality is naturally described in the context of data preparation and system development (e.g. adaptations to the test data) [1, 20, 22, 23, 13, 5, 9, 24, 17, 25, 19, 2, 26] and only few standards mention a potential need for quality improvement measures during operation [13, 25]. In addition to conventional categories of data quality measures (e.g. data cleaning, removal of duplicates or outliers, imputation of missing values, etc.) [20, 23, 5, 24, 17, 15, 19, 2], specifically the augmentation of data [22, 25, 24, 23, 4, 1, 13], as well as the possibility to remove missing, faulty or biased data [26, 24, 2, 23, 20, 4, 1] are consistently mentioned in the relevant standards. Furthermore, the visualization of data, e.g. to identify outliers and potential errors [20, 2], and human review of the data quality requirements are suggested [22, 18, 15, 2].

Also regarding the examination in view of possible biases, the existing standards overcome the data focus of the AI Act. Clearly, the data is frequently mentioned as one target of bias evaluation [1, 8, 9, 24, 18, 15, 25, 23] and some standards specifically highlight the labels in this regard [1, 23]. In addition, another emphasis in the bias testing is consistently on the AI system and its outputs [4, 1, 13, 5, 8, 24, 15, 17, 18, 3], with [1] highlighting that the whole system as well as individual components may be considered. In particular, some standards even focus on the model as the target of bias examination (e.g., [13, 5], and [4] which specifies concrete metrics only with respect to model outputs). Correspondingly, the measurement of bias is consistently required throughout development and operation (i.e. monitoring) [4, 1, 13, 5, 8, 9, 24, 17, 15, 25, 3], with less emphasis on the measurement during the inception stage [1, 24, 15, 25].

The measurement of data quality characteristics (certainly overlapping with data bias measurement) is, again, naturally mentioned with respect to the stage of data preparation i.e., prior to modeling [10, 11, 5, 25, 19, 2, 20, 21, 22, 23], but also in the context of AI development [21, 22, 23, 25, 19], This leaves open to a certain extent, whether all aspects of data quality can and should be evaluated independently from the resulting AI system. Furthermore, some standards point to the monitoring of data properties during operation e.g. with regard to distribution shifts [22, 25] .

4.2.2 Trade-offs and approval

Among the reviewed standards, we identified few direct references to potential conflicts between different AI system or data properties [4, 22, 21, 12, 5, 14, 13, 15, 26]. Although some specific examples of trade-offs are given, such as bias-accuracy trade-offs [5], guidance or

considerations are described only sporadically that could be used to address them in relation to specific system requirements. Especially in the context of data quality requirements, prioritization is suggested according to the organization's objectives and business needs (or the user's needs which is typically the organization) [21, 20, 1, 11]. In addition, the effort, cost and impact for data quality measures should be considered e.g., with respect to maintenance or necessary rework in the future [21, 12]. Likewise, the AI risk management standard describes a prioritization of mitigation measures based on the risks to organizational objectives, as well as risk-benefit analyses [3]. Other standards, by contrast, propose to include stakeholders in the definition and prioritization of requirements [12, 1, 24], or to prioritize requirements based on the identified (general) risk [26].

In addition to these directions and considerations for trade-offs, we also analyzed how the standards specify the approval of requirements. Regarding the involvement of stakeholders, [12] and [26] provide most comprehensive guidance among the standards subject to our review. In particular, [12] includes a stakeholder-target matrix and additional references regarding stakeholder requirements elicitation, and requires approval of the data quality requirements by all stakeholder groups (notably, a similar requirement exists also with respect to data bias analyses in [9]). Also [26] includes detailed process-level guidance on stakeholder involvement for the elicitation of relevant values, as well as on considerations regarding feasibility. However, an ethics expert does not necessarily need to be involved for achieving compliance with the standard, but it defines the role of a "top management champion" who resolves conflicts and prioritizes values. Final approval is subject to management and selected stakeholders. Similarly, the requirement of management approvals and sign-offs regarding AI properties can be found in other standards [17, 18]. Lastly, several standards suggest ethical review boards or procedures to be responsible for acceptance decisions [4, 16, 8].

5 Conclusion

Based on our analysis, the body of relevant standards provides useful definitions and a variety of technical solutions and processes that users can apply to identify and mitigate biases in their AI systems. We consider it particularly beneficial that the existing standards go beyond the focus on data quality as set out in the AI Act and also highlight model outputs and the overall AI system as crucial objects of bias measurement and mitigation. Regarding the operationalization of the AI Act with the stated aim of protecting against discrimination, we have also identified some aspects in which further clarification by the regulator itself or through harmonized standards would be desirable.

With regard to the specification of target concepts, we have recognized a tendency towards group- and accuracy-based approaches. Group-fairness metrics represent a natural (in view of sensitive attributes on the basis of which discrimination is not permitted) and, in particular, feasible approach to detecting biases in data and models. In addition, measuring the AI system's performance regarding specific persons or groups of persons is consistent with the specifications in Article 13 and Annex IV, 3. [38]. However, we noted that the existing standards (and the bias metrics included) mostly focus on classification tasks and lack specific guidance with respect to other state of the art techniques such as foundation models, large language models or reinforcement learning. It would be helpful to shed more light on the respective approaches, e.g. how to measure stereotyping which is especially relevant in the processing of textual data. Furthermore, a balanced model accuracy in relation to different groups can be in conflict with the concept of demographic parity, which plays a correspondingly minor role in the current standards. Still, the (conditional) demographic parity approach is highlighted in some legal studies as particularly important for the assessment of discrimination [63] and it would therefore be beneficial to examine the relevance of this metric for harmonized standards.

Regarding the data quality requirements, we have extracted several interpretations from the existing standards. A key point that would bring further clarification in some aspects is whether the data quality requirements may/should be measured and assessed against the resulting model (e.g. by whether the model quality is balanced for different groups), or whether they should be considered on their own, independently of the modeling (e.g. assessment of feature relevance based on factual, contextual relationships or statistical tests). Given the distributed value chains of AI applications, it would be practical to define clear transition points between the different actors (e.g. clarify which model-agnostic data properties should be implemented by data providers and handed over to the AI developer, for example documented as data sheet). Furthermore, it would also be useful to describe the selection of appropriate bias mitigation approaches as a dedicated process, which should include specifically the consideration and comparison of different possible methods (including measures in the model) to reflect the fact that the general effectiveness of existing measures is not yet well understood and may vary depending on the use case.

Along with the target concepts comes the question of who should determine compliance and according to which evaluation standards. Naturally, horizontal standards cannot define concrete target values e.g. for bias mitigation, as the variety of possible AI use cases is simply too broad. At the same time, there is currently a lack of extensive experience in the application and assessment of fairness-related requirements in real AI systems. Notably, the AI auditing and testing market which is already emerging in the unregulated area still faces many challenges. Further research and empirical validation regarding the effectiveness of the various implementation methods in practice is therefore necessary (horizontally and vertically) in order to be able to determine and further develop best practices, possibly with sector- or use case-specific evaluation standards (see also the discussions in [55, 43]). In addition, the general rationales for trade-offs and prioritization mentioned in the current standards are in part strongly based on the business objectives of the organization (see 4.2.2 and the AI-specific definition of data quality in 4.1.2), indicating an alignment challenge with the EU AI Act similar to those discussed in [60]. It would therefore be helpful to clarify which deviations from the fairness-related requirements are permitted and under which circumstances to further reduce uncertainty among providers (e.g. similar to the explicit exception in the AI Act that sensitive features can be used for bias mitigation, possibly in conflict with data protection).

Overall, the processes for stakeholder participation described in the dedicated standards appear to be a helpful basis, at present, to carry out the evaluation or approval of fairness requirements in specific AI applications in a (process-)standardized manner. Similar approaches have also been proposed in scientific work e.g. [50, 47, 64]. A specification in harmonized standards according to which criteria and level of granularity stakeholders should be involved in the acceptance of fairness impacts and requirements would support consistent implementation. Evaluation standards may then emerge bottom-up in relation to use cases or sectors. In future work, we aim to investigate whether existing vertical standards from the medical domain already contain more specific guidance compared to horizontal standards and if so, how this could be extended to other domains.

Acknowledgements

This research has been funded by the Ministry of Economic Affairs, Industry, Climate Action and Energy of the State of North Rhine-Westphalia as part of the flagship project ZERTIFIZIERTE KI, as well as by the Federal Ministry of Education and Research of Germany and the state of North Rhine-Westphalia as part of the Lamarr Institute for Machine Learning and Artificial Intelligence. The authors thank both consortia for the successful cooperation.

References

[1] ISO/IEC DTS 12791:2023, Treatment of unwanted bias in classification and regression machine learning tasks.
[2] ISO 16269-4:2010, Statistical interpretation of data - Part 4: Detection and treatment of outliers.
[3] ISO/IEC 23894:2023, AI - Guidance on risk management.
[4] ISO/IEC TR 24027:2021, Bias in AI systems and AI aided decision making, .
[5] ISO/IEC TR 24028:2020, Overview of trustworthiness in AI, .
[6] ISO/IEC 24029-1:2021, Assessment of the robustness of neural networks - Part 1: Overview, .
[7] ISO/IEC 24029-2:2023, Assessment of the robustness of neural networks - Part 2: Methodology for the use of formal methods, .
[8] ISO/IEC TR 24368:2022, Overview of ethical and societal concerns.
[9] ISO/IEC 24668:2022, Process management framework for big data analytics.
[10] ISO/IEC 25012:2008, Software product Quality Requirements and Evaluation (SQuaRE) - Data quality model, .
[11] ISO/IEC 25024:2015, SQuaRE - Measurement of data quality, .
[12] ISO/IEC 25030:2019, SQuaRE — Quality requirements framework, .
[13] ISO/IEC TS 25058:2024, SQuaRE – Guidance for quality evaluation of AI systems, .
[14] ISO/IEC 25059:2023, SQuaRE – Quality model for AI systems, .
[15] ISO/IEC TR 29119-11:2020, Guidelines on the testing of AI-based systems.
[16] ISO/IEC 38507:2022, Governance implications of the use of AI by organizations.
[17] ISO/IEC 42001:2023, AI – Management system, .
[18] ISO/IEC DIS 42005, AI system impact assessment, .
[19] ISO/IEC TS 4213:2022, Assessment of machine learning classification performance.
[20] ISO/IEC DIS 5259-1, Data quality for analytics and ML – Part 1; Overview, terminology, and examples, .
[21] ISO/IEC DIS 5259-2, Data quality for analytics and ML – Part 2; data quality measures, .
[22] ISO/IEC DIS 5259-3, Data quality for analytics and ML – Part 3; Data quality management requirements and guidelines, .
[23] ISO/IEC DIS 5259-4, Data quality for analytics and ML – Part 4; Data quality process framework, .
[24] ISO/IEC 5338:2023, AI system life cycle processes.
[25] ISO/IEC TR 5469:2024, Functional safety and AI systems.
[26] ISO/IEC/IEEE 24748-7000:2022, Standard model process for addressing ethical concerns during system design.
[27] A. Balahur et al. Data quality requirements for inclusive, non-biased and trustworthy AI. Technical report, Joint Research Centre, 2022.
[28] N. Becker et al. KI in der Arbeitswelt: Übersicht einschlägiger Normen und Standards. Berlin: Gesellschaft für Informatik e.V., 2021.
[29] R. Bommasani et al. On the opportunities and risks of foundation models. *arXiv preprint arXiv:2108.07258*, 2021.
[30] M. Cantero Gamito et al. Artificial intelligence co-regulation? The role of standards in the EU AI Act. *International Journal of Law and Information Technology*, 32(1):eaae011, 2024.
[31] A. de Hond et al. Guidelines and quality criteria for ai-based prediction models in healthcare: a scoping review. *NPJ digital medicine*, 5, no. 1: 2, 2022.
[32] S. de Vries et al. Internal market 3.0: The old "new approach" for harmonising ai regulation. *European Papers - A Journal on Law and Integration*, 2023(2):583–610, 2023.
[33] DIN e.V. & DKE. Standardization Roadmap Artificial Intelligence, 2nd edition, 2022.
[34] DIN Software. NAUTOS & DITR, Collection of technical rules. URL https://www.dinsoftware.de/resource/blob/320500/d5ed4d3642120dc4ad1c1bf8293a7f3f/perinorm-ditr-regelwerke-data.html.
[35] M. Ebers. Standardizing AI - The Case of the European Commission's Proposal for an Artificial Intelligence Act. *The Cambridge handbook of artificial intelligence: global perspectives on law and ethics*, 2021.
[36] European Commission. New legislative framework. https://single-market-economy.ec.europa.eu/single-market/goods/new-legislative-framework_en.
[37] European Commission. Draft standardisation request to the European Standardisation Organisations in support of safe and trustworthy AI. https://ec.europa.eu/docsroom/documents/52376, 2022.
[38] European Parliament and Council. Regulation (EU), No 2024/1689 (Artificial Intelligence Act), .
[39] European Parliament and Council. Regulation (EU) No 1025/2012, .
[40] J. Feldkamp et al. Rechtliche Fairnessanforderungen an KI-Systeme und ihre technische Evaluation – Eine Analyse anhand ausgewählter Kreditscoring-Systeme unter besonderer Berücksichtigung der zukünftigen europäischen KI-Verordnung. *Zeitschrift für Digitalisierung und Recht*, Heft 1:60–117, 2024.
[41] L. Floridi et al. CapAI-A procedure for conducting conformity assessment of AI systems in line with the EU AI Act. *SSRN 4064091*, 2022.
[42] S. Friedler et al. A comparative study of fairness-enhancing interventions in machine learning. In *Proceedings of the conference on fairness, accountability, and transparency*, pages 329–338, 2019.
[43] T. Goodman. Thinking Outside the Technical Standardisation Box: The Role of Standards Under the Draft EU Artificial Intelligence Act, 2023.
[44] P. Hacker. Teaching fairness to artificial intelligence: existing and novel strategies against algorithmic discrimination under eu law. *Common market law review*, 55(4), 2018.
[45] P. Hacker et al. AI Compliance - Challenges of Bridging Data Science and Law. *ACM Journal of Data and Inf. Quality*, 14(3):1–4, 2022.
[46] M. P. Hauer et al. Legal perspective on possible fairness measures–a legal discussion using the example of hiring decisions. *Computer Law & Security Review*, 42:105583, 2021.
[47] M. P. Hauer et al. Assuring fairness of algorithmic decision making. In *2021 IEEE International Conference on Software Testing, Verification and Validation Workshops (ICSTW)*, pages 110–113. IEEE, 2021.
[48] A. Jobin et al. The global landscape of ai ethics guidelines. *Nature machine intelligence*, 1(9):389–399, 2019.
[49] B. Kitchenham. Procedures for undertaking systematic reviews. Technical report, Joint Technical Report, Computer Science Department, Keele University (TR/SE-0401) and National ICT Australia Ltd. (0400011T.1), 2004.
[50] J. Laux et al. Three pathways for standardisation and ethical disclosure by default under the european union artificial intelligence act. *Computer Law & Security Review*, 53, 2024.
[51] Q. Lu et al. Towards a roadmap on software engineering for responsible ai. In *Proceedings of the 1st International Conference on AI Engineering: Software Engineering for AI*, pages 101–112, 2022.
[52] N. Mehrabi et al. A survey on bias and fairness in machine learning. *ACM computing surveys (CSUR)*, 54, no. 6:1–35, 2021.
[53] H.-W. Micklitz. The Role of Standards in Future EU Digital Policy Legislation. Technical report, ANEC and BEUC, 2023.
[54] S. Nativi et al. AI Standardisation Landscape: state of play & link to the EC proposal for an AI regulatory framework. 2021. ISSN 1831-9424.
[55] K. Prifti et al. Towards experimental standardization for ai governance in the eu. *Computer Law & Security Review*, 52:105959, 2024.
[56] S. Rismani et al. From plane crashes to algorithmic harm: Applicability of safety engineering frameworks for responsible ml. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*, pages 1–18, 2023.
[57] A. Schmitz et al. The why and how of trustworthy ai: An approach for systematic quality assurance when working with ml components. *at-Automatisierungstechnik*, 70(9):793–804, 2022.
[58] A. Schmitz et al. A global scale comparison of risk aggregation in AI assessment frameworks. *AI and Ethics*, pages 1–26, 2024.
[59] G. Sharkov et al. Strategies, policies, and standards in the eu towards a roadmap for robust and trustworthy ai certification. *Information & Security*, 50(1):11–22, 2021.
[60] J. Soler Garrido et al. Analysis of the preliminary AI standardisation work plan in support of the AI Act. Technical report, JRC, 2023.
[61] J. Soler Garrido et al. AI Watch: Artificial Intelligence Standardisation Landscape Update. Technical report, Joint Research Centre, 2023.
[62] S. Verma and J. Rubin. Fairness definitions explained. In *Proceedings of the international workshop on software fairness*, pages 1–7, 2018.
[63] S. Wachter et al. Why fairness cannot be automated: Bridging the gap between eu non-discrimination law and ai. *Computer Law & Security Review*, 41:105567, 2021.
[64] R. Zicari et al. Z-inspection®: a process to assess trustworthy ai. *IEEE Transactions on Technology and Society*, 2(2):83–97, 2021.

An Automated Arbitrator for Contesting Dialogues

Christodoulos Ioannou[a,*] **and Loizos Michael**[a,b,**]

[a]Open University of Cyprus, Nicosia, Cyprus
[b]CYENS Center of Excellence, Nicosia, Cyprus

Abstract. We present an automated system for arbitrating contesting dialogues, built on top of an argumentation-based reasoning engine. The arbitrator is introduced through the use-case scenario of a loan applicant contesting a bank's decision to reject their application. During their ensuing dialogue, the two parties exchange arguments, whose relative priorities are assumed to be known to the arbitrator when the latter determines the "winning" party during each round of the dialogue. Towards providing a natural interaction with humans, the arbitrator system provides a simple natural language interface.

1 Introduction

The increasing complexity of decision-making processes across domains has led to the demand for a structured approach to contest decisions [13]. With AI systems playing an increasingly significant role in decision making, there is an imperative need for a mechanism that allows humans to engage with these systems [7] in a human-centric way [1] that is both robust and understandable, in order to challenge and verify their alignment with regulatory and ethical standards.

The General Data Protection Regulation (GDPR) [8] of the European Union grants individuals the right to understand, access, correct, and contest decisions made using their personal data, particularly in the context of automated decision-making. Organizations that adopt automated decision-making must ensure transparency, accountability, and appropriate mechanisms to facilitate these legal rights.

Consequently, AI systems must be capable of comprehending, reasoning about, and effectively applying regulatory and ethical standards, ensuring their behavior is aligned with those standards. Computational argumentation [6] plays a crucial role in bridging the gap between human and machine reasoning by providing a structured framework for contesting decisions, aiming to create AI systems that are accountable, transparent, logical and accessible to humans [14].

The importance of contesting decisions made by automated systems is highlighted in various works, which include designing frameworks for contestable AI systems and mechanisms for human review [2], socio-technical approaches that combine software engineering practices with rule-based methodologies [1], and discussions on the transparency of embedded values in algorithmic systems [22]. Furthermore, some studies emphasize the need to incorporate dialectical processes into AI systems to protect against automatic decision making [21]. These studies contribute to a growing body of work that seeks to operationalize contestability in AI systems through the use of various technical and procedural strategies [15, 10, 3, 9].

* Corresponding Author. Email: christodoulos.ioannou@st.ouc.ac.cy.
** Corresponding Author. Email: loizos@ouc.ac.cy.

In this work, we start by presenting a use-case contesting dialogue, which we use to motivate and introduce certain key ideas towards operationalizing contestability. Based on these key features, we demonstrate how the exchange of arguments during the use-case contesting dialogue can be represented in the argumentation-based reasoning language Prudens [16]. Importantly, we show how the priorities between arguments, assumed to be known to the arbitrator but not necessarily to the two intelocutors, can be easily encoded in Prudens, and how then Prudens can determine the "winner" after each round of the contesting dialogue. Finally, we show how this arbitration process can be complemented with a user interface towards allowing the exchange of arguments in unconstrained natural language.

2 Contesting Dialogues and Arguments

To motivate and ground our presentation, we consider a use-case dialogue between two interlocutors, a "Bank Officer" and a "Loan Applicant", over the decision of a bank regarding the latter's loan application. The "Loan Applicant" interlocutor contests the banking institution's decision, and the "Bank Officer" interlocutor, who acts on behalf of the institution, must justify its decision. For the purposes of this use-case, we assume that the "Bank Officer" interlocutor has outdated information on the applicant's credit score and that the "Loan Applicant" knows the correct score. The use-case dialogue is below:

(N01) Bank Officer: "Your loan application is rejected."
(N02) Loan Applicant: "Why is my loan application rejected?"
(N03) Bank Officer: "Your loan application has been rejected because your care-giving obligations are considered high and your credit score is low."
(N04) Loan Applicant: "My loan application should not have been rejected because I am a good existing customer: I own an account for a long time and I make frequent transactions."
(N05) Bank Officer: "You are not qualified as a good existing customer because your account balance is low for more than one year."
(N06) Loan Applicant: "Why is my credit score low?"
(N07) Bank Officer: "Your credit score is considered low because it is 582."
(N08) Loan Applicant: "My credit score is 590."
(N09) Bank Officer: "Your credit score is considered low because it is below 600."
(N10) Loan Applicant: "Why are my care-giving obligations considered high?"
(N11) Bank Officer: "Your care-giving obligations are considered high because you are female and have two children."

(N12) Loan Applicant: "Gender should not be used to determine care-giving obligations."

Although presenting a full formalization for arbitrary contesting dialogues is not the goal of this paper, we still need to identify certain key features of the arguments exchanged during the use-case dialogue in order to allow automated reasoning by a system [23, 19, 4].

With the use-case dialogue as a reference, we observe that a dialogue comprises a set of concepts C (e.g., *"reject loan application"*, *"credit score is low"*, *"credit score value is 582"*). Some of these concepts are opposite to other concepts. A concept is considered to be opposite to some other concept when it represents the negation of the other concept (e.g., *"do not reject loan application"* is opposite to *"reject loan application"* and *"credit score is not low"* is opposite to *"credit score is low"*). Two opposite concepts are considered to be in conflict with each other. Two concepts can be conflicting even if they are not opposite to each other (e.g., *"credit score value is 590"* and *"credit score value is 582"*) if they cannot hold simultaneously.

The arguments A in the dialogue are then formed from these concepts C (e.g., *"credit score is considered low because it is 582"*). In general, A includes all the arguments formed by all possible combinations of the concepts in C, even if not all such arguments end up being invoked in any particular dialogue. Looking at our use-case dialogue, we can identify the following three types of arguments:

- *perception*, corresponds to a concept that is perceived as being true by at least one of the interlocutors; e.g., *"credit score value is 590"* or *"customer has two children"*.
- *supposition*, corresponds to a concept that is provisionally assumed to be true to facilitate the flow of the dialogue; e.g., *"I am a good existing customer"*.
- *association*, corresponds to an implication, the premise of which is a set of concepts and the conclusion a single concept; e.g., *"credit score is considered low because it is below 600"*.

In the course of a contesting dialogue, the arguments presented by the two interlocutors may be conflicting, in that their conclusions are conflicting. We assume that there exists a ground truth in terms of the relative priority between all pairs of conflicting arguments, that is specific to each use-case, and is known to the arbitrator, but not necessarily to the two interlocutors. Any such ground truth, however, needs to respect the following constraints: *perception* arguments have the highest priority, *supposition* arguments have the lowest priority, and *association* arguments have a priority that lies between *supposition* arguments and *perception* arguments.

At any stage of a contesting dialogue, an interlocutor can claim that any concept in C or its opposite holds, without being required to fully support such a claim at that stage, simply by presenting a *supposition* argument. By contrast, a *perception* argument can be presented only if it corresponds to a concept that is held to be true by the interlocutor; we are assuming that the arbitrator can check whether this condition holds, or that the interlocutor has to present some outside evidence in support of any perception argument put forward.

Effectively, then, a contesting scenario consists of a set of concepts C, a set of arguments A, a conflict relation between concepts, a priority relation between all conflicting arguments, and two sets stipulating the precepts of the two interlocutors. The arbitrator is assumed to have full knowledge of the contesting scenario, but each of the two interlocutors might have only a partial view. The two interlocutors in a dialogue take turns introducing new arguments from A, with these arguments persisting across all subsequent rounds. After each round, the arbitrator reasons with the available arguments to determine whether the original decision under contestation is entailed. At the end of the dialogue, the decision is upheld if and only if it is entailed by the set of all arguments put forward during the dialogue.

A specific contesting scenario that could have led to our use-case dialogue is illustrated in Figure 1. We annotate each natural language argument in the use-case dialogue with its corresponding number (e.g., *N05*) and we illustrate it with a coloured sketched oval which encapsulates the structured arguments that it comprises of. We represent concepts with solid line ovals. A concept may belong to more than one natural language argument. We represent structured arguments using directional solid line arrows which connect a premise concept to a conclusion concept. We number each argument (e.g., *N05*, *S02*, *P01*) and denote different types of arguments with different shapes. For clarity, we do not illustrate the premise concept for *supposition* and *perception* arguments, since in the former case it is assumed to always be satisfied, and in the latter case its satisfaction is based on extra-conceptual conditions. The different shapes used in the diagram are described in the legend accompanying the diagram.

Conflicts between conflicting arguments and their corresponding relative priorities are illustrated with thick dotted line arrows. For clarity, only those conflicts that are pertinent to the use-case dialogue are illustrated. A *perception* argument can attack a *supposition*, an *association*, or a *perception* argument with lower priority. An *association* argument can attack a *supposition* or an *association* argument with lower priority. A *supposition* argument can attack only a *supposition* argument. Attacks between *supposition* arguments are symmetric, and are illustrated with a two-way thick dotted line arrow.

Note that a "why" question by an interlocutor is effectively represented as a conflicting *supposition* argument to the *supposition* argument of the other interlocutor that is being questioned.

3 Arbitrator Representation and Reasoning

In this section we use Prudens [16], a declarative programming language, to represent the contesting scenario from Figure 1, with emphasis on the representation of arguments and their priorities, in a way that reasoning with Prudens can be used effectively to determine the "winning" interlocutor after each round of the use-case dialogue.

Prudens employs a prioritized rule-based logic, and supports efficient deduction with explanations for its inferences. The core syntax of Prudens includes *rules* that connect a premise to a conclusion (e.g., `fly(A), has(A, wing) implies is(A, bird)`). The list of *rules* on which the reasoning process is applied is called a *policy*. Resolution between conflicting *rules* in the *policy* is achieved by implicit order-based prioritization or explicit *rule* priorities. Prudens supports basic logic programming elements like *constants* (e.g., `wing`), *variables* (e.g., `A`), *predicates* (e.g., `is(A, bird)`), and the use of *negation*. A *literal* is either a *predicate* or a *negated predicate*. A *predicate* and its corresponding *negated predicate* are conflicting (e.g., `fly(penguin)` conflicts with `-fly(penguin)`). A list of non-conflicting *literals* that are provided as input to the reasoning process is called a *context*. Additionally, Prudens includes extended features such as mathematical operations, custom functions, explicit conflicts and conflict semantics [17, 16], not all of which we employ here.

Concepts in our contesting scenario are represented through predicates (e.g., `reject_loan_application` for the concept *"reject loan application"*, and `credit_score(low)` for the concept *"credit score is low"*). Negated predicates are used for the opposite concepts (e.g., `-good_existing_customer`). We have identified the following set of concepts C (and their negations, where appropriate) in the use-case dialogue, which are shown in the syntax of Prudens:

```
credit_score_value(582),
```

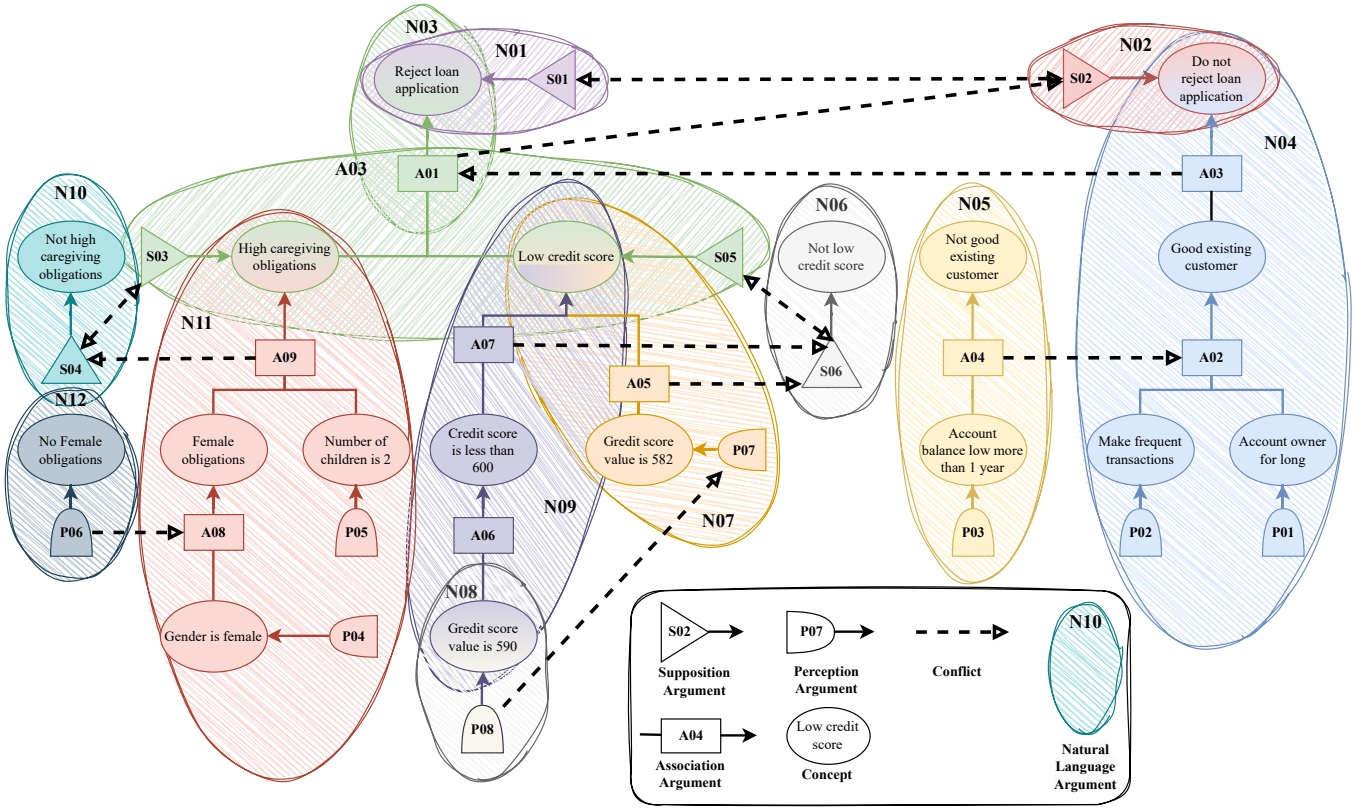

Figure 1. Representation of the contesting scenario considered in this work.

```
credit_score_value(590),
account_owner_for_long,
transaction_frequency(high),
account_balance_low_for(1, year),
gender(female),
have(child, 2),
female_obligations,
-female_obligations,
credit_score_less_than(600),
credit_score(low),
-credit_score(low),
good_existing_customer,
-good_existing_customer,
caregiving_obligations(high),
-caregiving_obligations(high),
reject_loan_application,
-reject_loan_application
```

Conflicts between non-opposite concepts are represented using the special conflict expressions available in Prudens. We have identified only one such conflict between two concepts that appear in the use-case dialogue, represented by the following conflict expression:

```
credit_score_value(582) # credit_score_value(590);
```

Rules with the premise `perceive` are used to represent *perception* arguments (e.g., `perceive implies gender(female)`).[1] *Rules* with the premise `suppose` are used to represent *supposition* arguments (e.g., `suppose implies good_existing_customer`). We represent an *association* argument by a *rule* the premise of which is a set of concepts from C, as illustrated in the example below:

```
account_owner_for_long,
transaction_frequency(high) implies
  good_existing_customer;
```

We shall not enumerate all possible arguments A that derive from concepts C. Instead, below we present only those *association* arguments that are pertinent given the natural language arguments in the use-case dialogue (cf. Table 1), along with all *perception* and *supposition* arguments for every concept in C and their opposites. Arguments are named with an index prefixed by a letter indicating the type of the argument, with these names also being used in Figure 1.

```
S01 :: suppose implies
         reject_loan_application | 00;
S02 :: suppose implies
         -reject_loan_application | 00;
S03 :: suppose implies
         caregiving_obligations(high) | 00;
S04 :: suppose implies
         -caregiving_obligations(high) | 00;
S05 :: suppose implies
         credit_score(low) | 00;
S06 :: suppose implies
         -credit_score(low) | 00;
S07 :: suppose implies
         good_existing_customer | 00;
S08 :: suppose implies
         -good_existing_customer | 00;
S09 :: suppose implies
         -account_owner_for_long | 00;
S10 :: suppose implies
```

[1] In doing so, we are sidestepping the question of how the arbitrator can determine whether a perception argument is indeed justified given the held beliefs of an interlocutor. This can be handled more fully by replacing the common premise with an expression unique to each *perception* argument.

```
        -transaction_frequency(high) | 00;
S11 :: suppose implies
        -account_balance_low_for(1, year) | 00;
S12 :: suppose implies
        -gender(female) | 00;
S13 :: suppose implies
        -have(child, 2) | 00;
S14 :: suppose implies
        -female_obligations | 00;
S15 :: suppose implies
        -credit_score_value(582) | 00;
S16 :: suppose implies
        -credit_score_value(590) | 00;
A01 :: caregiving_obligations(high),
       credit_score(low) implies
        reject_loan_application | 11;
A02 :: account_owner_for_long,
       transaction_frequency(high) implies
        good_existing_customer | 12;
A03 :: good_existing_customer implies
        -reject_loan_application | 13;
A04 :: account_balance_low_for(1, year) implies
        -good_existing_customer | 14;
A05 :: credit_score_value(582) implies
        credit_score(low) | 15;
A06 :: credit_score_value(590) implies
        credit_score_less_than(600) | 16;
A07 :: credit_score_less_than(600) implies
        credit_score(low) | 17;
A08 :: gender(female) implies
        female_obligations | 18;
A09 :: have(child, 2), female_obligations implies
        caregiving_obligations(high) | 19;
P01 :: perceive implies
        account_owner_for_long | 21;
P02 :: perceive implies
        transaction_frequency(high) | 22;
P03 :: perceive implies
        account_balance_low_for(1, year) | 23;
P04 :: perceive implies
        gender(female) | 24;
P05 :: perceive implies
        have(child, 2) | 25;
P06 :: perceive implies
        -female_obligations | 26;
P07 :: perceive implies
        credit_score_value(582) | 27;
P08 :: perceive implies
        credit_score_value(590) | 28;
```

To encode the priorities between conflicting arguments, and without over-complicating the policy by including explicit priorities between every pair of conflicting arguments, we have chosen to associated each argument with a numerical value, using the corresponding syntax supported by Prudens. In case two arguments are conflicting, this Prudens construct implies that the one with the higher value takes priority over the one with the lower value. Other than determining these pairwise priorities, the actual values are not consequential.

By assigning a value of 00 to all *supposition* arguments, we readily capture the weak form of these arguments, and the mutual attack between any pair of such arguments. By assigning intermediate and high values to *association* and *perception* arguments, respectively, we capture the relative priorities between these types of arguments.

The precise values within each of these two categories of arguments correspond to priorities that are specific to the scenario that we are considering, as reflected in Figure 1. Recall that these priorities are not expressed in our use-case dialogue, and might, in fact, not be fully known to the interlocutors, but they are to the arbitrator.

Having described how our contesting scenario is represented in Prudens, we discuss how the arbitration process takes place during the use-case dialogue. We initialize Prudens with an empty *policy* and a *context* comprising the predicates `suppose` and `perceive`. At every round of the use-case dialogue, a natural language argument (or a "why" question) is put forward by one of the interlocutors. This corresponds to a collection of Prudens expressions, which are added into the current *policy*, without removing previous expressions. The current *policy* represents the arguments placed so far and it is used to infer the conclusion of the arbitrator at that point of the dialogue.

Note that certain structured arguments might be implied but not necessarily stated explicitly in a natural language argument. For instance, and as discussed earlier, a "why `X`" question is effectively a *supposition* argument for `-X`. Also, in the case that a natural language argument captures some form of association between concepts, then the premise concepts of the *association* argument are effectively assumed to be true. If these premise concepts cannot be established by other previously introduced arguments, then this implies that the corresponding *supposition* or *perception* (whenever applicable) arguments are also put forward automatically by the arbitrator. Finally, a conflict expression is added to the *policy* upon the first appearance of a concept that explicitly conflicts with a concept introduced earlier.

Table 1 shows the realization of the use-case dialogue in Prudens, where the arguments are presented in the same order as they appear in the natural language use-case dialogue. The *policy* representing the use-case dialogue at any specific point includes all the expressions in the *Prudens Expressions* column up to the respective line. The Prudens reasoning engine is invoked on every such *policy*, acting as an arbitrator to draw conclusions. The final and "winning" conclusion of the dialogue is the conclusion inferred on the final policy when no more arguments are put forward by any of the two interlocutors.

4 Prototype System and User Interface

As a proof of concept for the ideas presented herein, we have developed a prototype arbitrator for our use-case dialogue, available at: `https://cognition.ouc.ac.cy/contestability`. As per the use-case dialogue, the "Bank Officer" interlocutor presents arguments to justify the decision-making agent's original decision to *"reject loan application"*, whereas the "Loan Applicant" interlocutor presents arguments to contest that original decision. At each step of the dialogue, the natural language argument presented by one of the interlocutors is translated into Prudens expressions as described in Section 3, and these are added into the dialogue *policy*. Then, the Prudens reasoning engine is invoked on the current *policy* to infer the formal conclusions of the arbitrator at this specific point. The formal conclusions are, in turn, translated back into natural language.

If the inferences of the reasoning engine include the *predicate* `reject_loan_application`, which is the formal representation of concept *"reject loan application"* (the decision-making agent's original decision), then the arbitrator's determination is that the loan application rejection is considered upheld. On the other hand, if the *negated predicate* `-reject_loan_application` is included in the inferences, then the arbitrator's determination is that the loan application rejection is considered dismissed. In any of these cases, the key supporting arguments, which are consisted by the *association* arguments supporting the reasoning engine's inferences, are translated into natural language to provide a justification of the arbitrator's determination. If the system cannot infer any of these two conflicting predicates, then the decision-making agent (through the "Bank Officer") should support its decision to reject the loan application by presenting a new argument. Otherwise, the decision will be dismissed as not being justified, even if the opposite cannot be justified as well.

Figure 2 shows the output of the prototype system on three occasions during the dialogue, after arguments *N03*, *N04*, and *N08* have

Table 1. Our use-case dialogue realized in the Prudens language.

#	Prudens Expressions
N01	`suppose implies` `  reject_loan_application | 00;`
N02	`suppose implies` `  -reject_loan_application | 00;`

N03	`suppose implies`
	`  caregiving_obligations(high) | 00;`
	`suppose implies`
	`  credit_score(low) | 00;`
	`caregiving_obligations(high),`
	`credit_score(low) implies`
	`  reject_loan_application | 11;`

N04	`good_existing_customer implies`
	`  -reject_loan_application | 13;`
	`perceive implies`
	`  account_owner_for_long | 21;`
	`perceive implies`
	`  transaction_frequency(high) | 22;`
	`account_owner_for_long,`
	`transaction_frequency(high) implies`
	`  good_existing_customer | 12;`

N05	`perceive implies`
	`  account_balance_low_for(1, year) | 23;`
	`account_balance_low_for(1, year) implies`
	`  -good_existing_customer | 14;`

N06	`suppose implies` `  -credit_score(low) | 00;`

N07	`credit_score_value(582) #`
	`  credit_score_value(590);`
	`perceive implies`
	`  credit_score_value(582) | 27;`
	`credit_score_value(582) implies`
	`  credit_score(low) | 15;`

N08	`perceive implies` `  credit_score_value(590) | 28;`

N09	`credit_score_value(590) implies`
	`  credit_score_less_than(600) | 16;`
	`credit_score_less_than(600) implies`
	`  credit_score(low) | 17;`

N10	`suppose implies` `  -caregiving_obligations(high) | 00;`

N11	`perceive implies`
	`  gender(female) | 24;`
	`perceive implies`
	`  have(child, 2) | 25;`
	`gender(female) implies`
	`  female_obligations | 18;`
	`have(child, 2), female_obligations implies`
	`  caregiving_obligations(high) | 19;`

N12	`perceive implies` `  -female_obligations | 26;`

been put forward by one of the two interlocutors. The determination of the arbitrator after argument *N03* is that the loan application rejection is upheld since the *predicate* `reject_loan_application` is inferred. The determination of the arbitrator after argument *N04* is that the loan application rejection is dismissed since the *predicate* `-reject_loan_application` is inferred. Finally, the determination of the arbitrator after argument *N08* is that the loan application rejection requires further support by the "Bank Officer" interlocutor, since neither of the predicates `reject_loan_application` and `-reject_loan_application` is inferred. An appropriate justification is provided based on key supporting arguments when the determination is either to uphold or dismiss the loan application rejection.

To facilitate the natural interaction of humans in a contesting dialogue with an AI-based interlocutor, it is important to translate natural language text into the formal expressions recognized by Prudens. As a first step, our prototype system employs a translation map from the natural language arguments of the use-case dialogue to their corresponding Prudens expressions. This simple translation process allowed us to validate our approach for our specific use-case dialogue.

To further enhance the naturalness of the system, we employed ChatGPT 4o [18] during the translation process. Given the translation map created in the previous step, and given a natural language argument not necessarily found in the translation map, we asked ChatGPT 4o to return the formal expressions with the closest meaning to the input text. The consistency of the results was high even when the input text had significantly different syntax and wording from text found in the translation map, but conveyed the same underlying meaning. The ChatGPT-based version improved the system's ability to handle variations in syntax and wording, maintaining high accuracy in mapping natural language arguments to Prudens expressions.

5 Conclusions

We have considered the task of building an automated arbitrator for contesting dialogues. Without attempting to provide a full formalization of the key ideas behind our work, we have demonstrated the feasibility of building such an arbitrator, both in terms of representing and reasoning with the necessary knowledge that the arbitrator would need to work, but also in terms of developing a system that can support interaction with humans in natural language. Although our work has focused on a particular use-case dialogue and contesting scenario, we maintain that the ideas presented herein are sufficiently general to be applied to a wide range of contesting scenarios or be combined with other techniques for even wider applicability.

By demonstrating a practical implementation of an argumentation-based dialogue system, specifically tailored for contesting decisions, our work emphasizes the applicability and importance of argumentation in decision support systems [5, 14] and in the development of persuasive technologies [20]. By leveraging the Prudens language and reasoning engine [16], we aspire to provide a robust platform for formalizing contesting dialogues through an automated arbitrator.

Possible next steps for this work include enhancing the prototype system to allow interlocutors to present not only arbitrarily-worded arguments, but also arbitrarily-chosen arguments, towards supporting open-ended contesting dialogues that are not a priori determined and known. Enhancements will also aim to improve the system's ability to understand and process natural language, facilitating even more the natural human-machine communication. The further and more systematic use of Large Language Models (LLMs) during the translation process will leverage their substantial natural language understanding capabilities to provide a more natural and robust solution.

Zooming out to the bigger picture, future work will focus on the development of a more general, fully-fledged formal framework capable of handling a broader range of scenarios and more diverse types of contested decisions. This will require the automated consistent identification and generation of the set of concepts and arguments that comprise a dialogue, as the dialogue unfolds. Natural language

Figure 2. Dialogue and conclusions after arguments *N03*, *N04*, and *N08* have been put forward (from top to bottom).

parsers that are specifically designed for translating natural language into logic and are capable of identifying patterns in natural language text [11] can be used alongside LLMs for this task. This is especially true in domain-specific dialogues, where such parsers can be coached to identify important concepts [12, 17]. The integration of advanced natural language processing tools to enhance the system's generality and adaptability is necessary for the development of a methodology and a system that will be practically useful and will help promote transparency and fairness in automated decision-making systems.

Acknowledgements

This work was supported by funding from the EU's Horizon 2020 Research and Innovation Programme under grant agreement no. 739578, and from the Government of the Republic of Cyprus through the Deputy Ministry of Research, Innovation, and Digital Policy.

References

[1] A. Aler Tubella, A. Theodorou, V. Dignum, and L. Michael. Contestable Black Boxes. In V. Gutiérrez-Basulto, T. Kliegr, A. Soylu, M. Giese, and D. Roman, editors, *Rules and Reasoning*, pages 159–167, Cham, 2020. Springer International Publishing. ISBN 978-3-030-57977-7.

[2] K. Alfrink, I. Keller, G. Kortuem, and N. Doorn. Contestable AI by Design: Towards a Framework. *Minds and Machines*, 33(4):613–639, 2023. ISSN 1572-8641. doi: 10.1007/s11023-022-09611-z. URL https://doi.org/10.1007/s11023-022-09611-z.

[3] M. Almada. Human Intervention in Automated Decision-Making: Toward the Construction of Contestable Systems. In *Proceedings of the Seventeenth International Conference on Artificial Intelligence and Law*, ICAIL '19, page 2–11, New York, NY, USA, 2019. Association for Computing Machinery. ISBN 9781450367547. doi: 10.1145/3322640.3326699. URL https://doi.org/10.1145/3322640.3326699.

[4] P. Baroni, D. Gabbay, M. Giacomin, and L. van der Torre. *Handbook of Formal Argumentation*. College Publications, 2018. ISBN 9781848902756. URL https://books.google.com.cy/books?id=_OnTswEACAAJ.

[5] T. Bench-Capon and P. E. Dunne. Argumentation in Artificial Intelligence. *Artificial Intelligence*, 171(10):619–641, 2007. ISSN 0004-3702. doi: 10.1016/j.artint.2007.05.001. URL https://www.sciencedirect.com/science/article/pii/S0004370207000793.

[6] E. Dietz, A. Kakas, and L. Michael. Computational Argumentation and Cognition, 2021.

[7] E. Dietz, A. Kakas, and L. Michael. Argumentation: A Calculus for Human-Centric AI. *Frontiers in Artificial Intelligence*, 5, 2022. ISSN 2624-8212. doi: 10.3389/frai.2022.955579. URL https://www.frontiersin.org/articles/10.3389/frai.2022.955579.

[8] European Parliament and Council of the European Union. Regulation (EU) 2016/679 of the European Parliament and of the Council on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC (General Data Protection Regulation), 2016. URL https://data.europa.eu/eli/reg/2016/679/oj.

[9] C. Henin and D. Le Métayer. Beyond Explainability: Justifiability and Contestability of Algorithmic Decision Systems. *AI & SOCIETY*, 7(4):1397–1410, 2022. ISSN 1435-5655. doi: 10.1007/s00146-021-01251-8. URL https://doi.org/10.1007/s00146-021-01251-8.

[10] T. Hirsch, K. Merced, S. Narayanan, Z. E. Imel, and D. C. Atkins. Designing Contestability: Interaction Design, Machine Learning, and Mental Health. In *Proceedings of the 2017 Conference on Designing Interactive Systems*, DIS '17, page 95–99, New York, NY, USA, 2017. Association for Computing Machinery. ISBN 9781450349222. doi: 10.1145/3064663.3064703. URL https://doi.org/10.1145/3064663.3064703.

[11] C. Ioannou and L. Michael. Knowledge-Based Translation of Natural Language into Symbolic Form. In *Proceedings of the 7th Linguistic and Cognitive Approaches To Dialog Agents Workshop - LaCATODA 2021 (IJCAI 2021)*, volume 2935, pages 24–32. CEUR-WS, 2021.

[12] C. Ioannou and L. Michael. A Coachable Parser of Natural Language Advice. In *Proceedings of the 16th International Conference on Agents and Artificial Intelligence - Volume 2: ICAART*, pages 500–510. INSTICC, SciTePress, 2024. ISBN 978-989-758-680-4. doi: 10.5220/0012454500003636.

[13] M. E. Kaminski and J. M. Urban. The Right to Contest AI. *Columbia Law Review*, 121(7):1957–2048, 2021. ISSN 00101958, 19452268. URL https://www.jstor.org/stable/27083420.

[14] F. Leofante, H. Ayoobi, A. Dejl, G. Freedman, D. Gorur, J. Jiang, G. Paulino-Passos, A. Rago, A. Rapberger, F. Russo, X. Yin, D. Zhang, and F. Toni. Contestable AI needs Computational Argumentation, 2024.

[15] H. Lyons, E. Velloso, and T. Miller. Conceptualising Contestability: Perspectives on Contesting Algorithmic Decisions. *Proc. ACM Hum.-Comput. Interact.*, 5(CSCW1), apr 2021. doi: 10.1145/3449180. URL https://doi.org/10.1145/3449180.

[16] V. T. Markos and L. Michael. Prudens: An Argumentation-based Language for Cognitive Assistants. In *Proceedings of the 6th International Joint Conference on Rules and Reasoning (RuleML+RR'22)*, pages 296–304, Berlin, Germany, 2022. Springer.

[17] L. Michael. Machine Coaching. In *Proceedings of the IJCAI Workshop on Explainable Artificial Intelligence (XAI'19)*, pages 80–86, S.A.R. Macau, P.R. China, 2019.

[18] OpenAI. ChatGPT: OpenAI Language Model. https://www.openai.com/, 2023.

[19] H. Prakken. An Abstract Framework for Argumentation with Structured Arguments. *Argument & Computation*, 1(2):93–124, 2010. doi: 10.1080/19462160903564592. URL https://doi.org/10.1080/19462160903564592.

[20] C. Reed and F. Grasso. Recent Advances in Computational Models of Natural Argument. *International Journal of Intelligent Systems*, 22(1):1–15, 2007. doi: 10.1002/int.20187. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/int.20187.

[21] C. Sarra. Put Dialectics into the Machine: Protection against Automatic-Decision-Making through a Deeper Understanding of Contestability by Design. *Global Jurist*, 20(3), 2020. doi: doi:10.1515/gj-2020-0003. URL https://doi.org/10.1515/gj-2020-0003.

[22] K. Vaccaro, K. Karahalios, D. K. Mulligan, D. Kluttz, and T. Hirsch. Contestability in Algorithmic Systems. In *Companion Publication of the 2019 Conference on Computer Supported Cooperative Work and Social Computing*, CSCW '19 Companion, page 523–527, New York, NY, USA, 2019. Association for Computing Machinery. ISBN 9781450366922. doi: 10.1145/3311957.3359435. URL https://doi.org/10.1145/3311957.3359435.

[23] D. Walton, C. Reed, and F. Macagno. *Argumentation Schemes*. Cambridge University Press, 2008.

A Hate Speech Moderated Chat Application: Use Case for GDPR and DSA Compliance

Jan Fillies[a,b,*], **Theodoros Mitsikas**[a,c], **Ralph Schäfermeier**[d] **and Adrian Paschke**[a,b,e]

[a]Institut für Angewandte Informatik, Leipzig, Germany
[b]Freie Universität Berlin, Berlin, Germany
[c]National Technical University of Athens, Zografou, Greece
[d]Leipzig University, Leipzig, Germany
[e]Fraunhofer-Institut für Offene Kommunikationssysteme, Berlin, Germany
ORCID (Jan Fillies): https://orcid.org/0000-0002-2997-4656, ORCID (Theodoros Mitsikas): https://orcid.org/0000-0002-7570-3603, ORCID (Ralph Schäfermeier): https://orcid.org/0000-0002-4349-6726, ORCID (Adrian Paschke): https://orcid.org/0000-0003-3156-9040

Abstract. The detection of hate speech or toxic content online is a complex and sensitive issue. While the identification itself is highly dependent on the context of the situation, sensitive personal attributes such as age, language, and nationality are rarely available due to privacy concerns. Additionally, platforms struggle with a wide range of local jurisdictions regarding online hate speech and the evaluation of content based on their internal ethical norms. This research presents a novel approach that demonstrates a GDPR-compliant application capable of implementing legal and ethical reasoning into the content moderation process. The application increases the explainability of moderation decisions by utilizing user information. Two use cases fundamental to online communication are presented and implemented using technologies such as GPT-3.5, Solid Pods, and the rule language Prova. The first use case demonstrates the scenario of a platform aiming to protect adolescents from potentially harmful content by limiting the ability to post certain content when minors are present. The second use case aims to identify and counter problematic statements online by providing counter hate speech. The counter hate speech is generated using personal attributes to appeal to the user. This research lays the groundwork for future DSA compliance of online platforms. The work proposes a novel approach to reason within different legal and ethical definitions of hate speech and plan the fitting counter hate speech. Overall, the platform provides a fitted protection to users and a more explainable and individualized response. The hate speech detection service, the chat platform, and the reasoning in Prova are discussed, and the potential benefits for content moderation and algorithmic hate speech detection are outlined. A selection of important aspects for DSA compliance is outlined.

1 Introduction

"Content moderation is the organized practice of screening user-generated content" [25]. It is a highly sensitive issue that directly influences a person's online safety.

The Digital Services Act (DSA) was adopted in October 2022 and has been applicable since February 2024. The goal of the DSA is to define a comprehensive framework to counteract the dissemination of illegal and problematic content. It proposes a layered framework that defines different rules for different scopes. For a detailed view, refer to Husovec and Roche Laguna [10]. One of the key problems of the DSA is that it does not harmonize what content or behavior is considered illegal; this remains under the sovereignty of the member states [10].

Husovec and Roche Laguna [10] further states that a crucial aspect of the DSA is that online platforms accessible to minors must implement measures to ensure a high level of privacy, safety, and security. Additionally, they note that hosting providers must conduct fair content moderation. Uploaders are entitled to an explanation for the providers' actions, whether the action is based on legal violations or terms of use violations. These aspects are not applicable to all platforms and do not exhaustively cover everything that needs to be fulfilled, but they are key aspects of the new regulation.

The DSA Act lays the groundwork for any moderation system and significantly influences the future of online communication.

Another important legislation is the European GDPR (General Data Protection Regulation) which was introduced in 2016 to set guidelines for personal data protection. These guidelines cover all major areas of life, setting standards and rules for handling personal data. A key aspect of GDPR is the ability to consent to and revoke consent for data processing. In the case of data processing through a data controller, it is necessary for the user (data subject) to be able to consent to or reject the processing (European Commission, 2016, Article 6). Furthermore, the user must also have the right to access all collected personal data in a machine-readable format and be able to transfer it to a different data controller (right to data portability) (European Commission, 2016, Article 20).

In the sensitive field of online content moderation, data privacy is highly important. On the one hand, having access to certain personal information of stakeholders in an online conversation can enable an unbiased and reliable system to be more precise and fair in performing automated moderation, as well as providing effective countermeasures against hate speech. On the other hand, this personal information can be very sensitive, necessitating strict guidelines on how to handle it and in what context. Managing and moderating online

* Corresponding Author. Email: fillies@infai.org

written content requires robust procedures that adhere to GDPR regulations, ensuring that user data is protected while maintaining a safe environment. Online platforms are balancing between internal community guidelines, and the jurisdictions covering the users and organizations, these are drivers of the complexity of the problem.

If a moderation system needs to use personal data or online communication in general, it must comply with these regulations. Therefore, it should be able to implement different levels of policies and handle the range of complexities occurring within the system. This starts with the simple execution of ground rules and extends to instances where rules are overridden, establishing a hierarchy that prioritizes some rules over others depending on the situation.

This research establishes a system for GDPR-compliant content monitoring capable of representing non-monotonic states and fulfilling the mentioned key aspects of the DSA. To this end, we present two different use cases (UC) for hate speech detection in online chatrooms modeled using the rule language Prova, Solid Pods, GPT-3.5 based hate speech detection and personalized counter hate speech generation. The use cases include typical stakeholders such as users, the platform, and data controllers. Access control and moderation are realized using concepts such as user consent, the purpose of access, and the role of the party requesting access. Prova and Solid have been used in various domains and similar contexts, demonstrating their versatility and effectiveness in applications requiring compliance with data protection regulations. The system is combined and demonstrated in a prototype implementing GDPR-compliant data sharing for content monitoring, using personal attributes during moderation and automatic counter hate speech generation. The prototype is available for testing. [1]

The research established the following main research objectives:

1. A legal and ethical reasoning system for content moderation.
2. Counter hate speech generation based on personal attributes.
3. A chat platform for GDPR and DSA compliant content moderation.

The paper is organized as follows: Sections 2 presents related work. Section 3 details the technical preliminaries such as Prova and Solid. Followed by Section 4 outlining both use cases. Section 5 describes the implementation of the prototype. In Section 6 the work and ethical considerations are discussed. Followed lastly be the conclusion and future work in Section 7.

2 Related Work

In the field of hate speech detection, historically, transformer-based architectures [20] and fine-tuning of transformer-based models [5], specifically BERT [2], have yielded better performance compared to traditional machine learning models [16, 1, 18]. In recent years, pretrained large language models have gained traction [11, 15] due to their performance and simple setup. LLMs have also proven efficient in the field of counter hate speech generation [29], with the capability to effectively generate personalized counter hate speech [3].

In the area of compliance checking, many different works have been established in recent years [27, 7, 9]. Satoh et al. [27] proposed a legal reasoning system for decision-making by judges under incomplete information. Hayashi et al. [9] established a compliance mechanism for AI agent planning in a multi-agent setting. Goossens et al. [6] showed the possibilities of using GPT-3 in decision logic modeling, and Hayashi and Satoh [8] presented a planning method for legal and ethical norms.

Two main works in the field could be established. Firstly, Schäfermeier et al. [28] proposed a distributed data wallet use case that is GDPR-compliant, comparing two different approaches by applying AspectOWL and Prova for the modeling and implementation. AspectOWL is a monotonic contextualized ontology language that focuses on the representation of dynamic state transitions and knowledge retention by wrapping parts of the ontology in isolated contexts. In contrast, Prova handles state transitions at runtime using non-monotonic state transition semantics. They analyzed two use cases: one providing a personalized search and the other outlining the process of sharing pictures via a wallet-enabled sharing app. Both use cases were implemented and evaluated on aspects such as human and machine-readability, manageability, and the use of open standard technology. One of the findings was that AspectOWL is suitable for specifying the ontological domain model, while Prova is a more practical approach for real-world applications, including the interaction between involved parties.

The second research by Mitsikas et al. [19] presents a medical data access use case compliant with GDPR legal rules, also implemented using Prova. It demonstrates a scenario of a patient consenting to medical data sharing. The data is used for a specific purpose, and cases were considered where the typical rules are overridden, thereby adjusting the access rights.

This current research uses modern algorithms for hate speech detection and builds upon the works by Schäfermeier et al. [28] and Mitsikas et al. [19], but also others, as Hayashi et al. [9]. It follows existing research into designing a GDPR-compliant application, also choosing Prova for development due to its practicality and scalability. This research advances the field with two new highly important use cases and incorporates key aspects of the DSA legislation.

3 Technical Preliminaries

3.1 Prova

Prova is both a (Semantic) Web rule language and a distributed (Semantic) Web rule engine. It supports reaction rule based workflows, event processing, and reactive agent programming. It integrates Java scripting with derivation and reaction rules, and message exchange with various communication frameworks [14, 12, 21].

Syntactically, Prova builds upon the ISO Prolog syntax and extends it, notably with the integration of Java objects, typed variables, F-Logic-style slots, and SPARQL and SQL queries. Slotted terms in Prova are implemented using the arrow expression syntax '->'as in RIF and RuleML, and can be used as sole arguments of predicates. They correspond to a Java HashMap, with the keys limited to Stings [13].

Semantically, Prova provides the expressiveness of serial Horn logic with a linear resolution for extended logic programs (SLE resolution) [22], extending the linear SLDNF resolution with goal memoization and loop prevention. Negation as failure support in the rule body can be added to a rulebase by implementing it using the cut-fail test as follows:

```
not(A) :- derive(A), !, fail().
not(_).
```

Prova implements an inference extension called literal *guards*, specified using brackets. By using guards, we can ensure that during unification, even if the target rule matches the source literal, further evaluation is delayed unless a guard condition evaluates to

[1] http://81.169.159.230:7000/

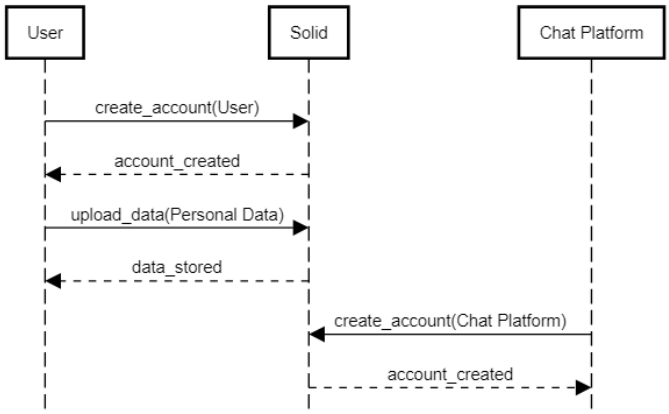

Figure 1. Sequence diagram of the initial account creation steps every user has to do in both use cases.

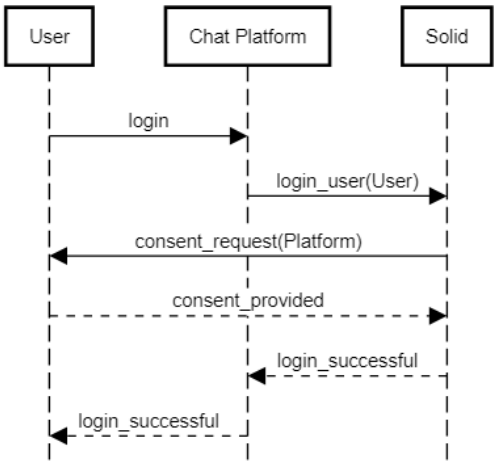

Figure 2. Sequence diagram of the steps done to join a chatroom. Every user has to do these steps in both use cases.

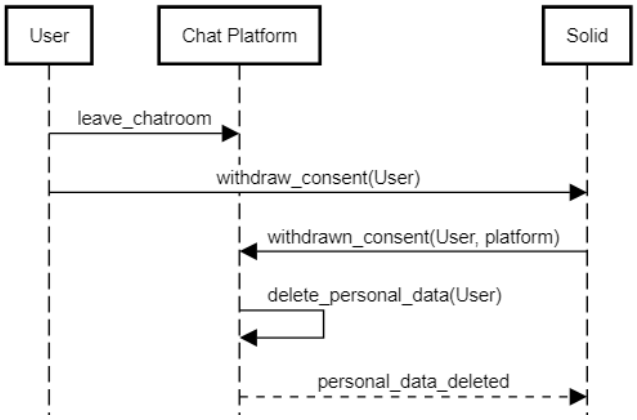

Figure 3. Sequence diagram of leaving a chatroom. Steps every user has to do in both use cases.

true. Guards can include arbitrary lists of Prova literals including Java calls, arithmetic expressions, relations, and even the cut operator. Prova guards play even a more important role in message and event processing, as they allow the received messages to be examined before they are irrevocably accepted. The guards are tested right after pattern matching but before a message is fully accepted, so that the net effect of the guard is to serve as an extension of pattern matching for literals [13, 23].

3.2 Solid

The Solid platform, first introduced by Sambra et al. [26], is a decentralized platform using W3C standards to create social applications based on linked data approaches. Linked data is a form of data interlinked with each other and accessible via semantic queries [24]. As described by Mansour et al. [17], following the concept of Solid, the data of each user is stored independently of the sources that created it, the data broker, and the end data consumer. Each user owns and manages their own personal online datastore (Pod) where all personal data is stored. A user is not limited to one Pod or one hosting provider, as they can self-host their Pods or choose between different hosting providers.

Applications that want to work with and access the data use protocols based on W3C standards. Mansour et al. [17] further states that a decentralized authentication and access control mechanism lays the groundwork for strong privacy protection. The decentralized architecture allows applications to access the user data independent of the hosting option, while users have full control over and access to their data at any point, with the possibility to switch providers or withdraw consent for sharing data.

4 Use Cases

Two data wallet use cases are described in terms of interaction sequences and data exchange between the different parties involved. The use cases involve data wallet owners sharing personal data using relaying parties that provide specialized applications, such as an Age-based Content Moderation application and a Hate Speech Classification combined with a Contextualized Semantic Counter Narrative Generation.

4.1 General Steps

Certain steps apply to both uses cases. Figure 1 refers to the initial steps a user and the platform has to perform to create and register with the Solid platform. The solid platform represents the data controller in this setting, storing the personal data and managing its access. Figure 2 depicts the process of a user login into the chatroom, providing consent for accessing personal data to the solid instance, and finally joining a specific chatroom. Lastly, Figure 3 is the process of leaving the chatroom, withdrawing the consent to user the personal data. In all figures, the requests are represented by solid arrows and the responses by dotted arrows.

4.2 Use Case 1: Age-based Content Moderation

The following use case outlines the scenario in which a platform needs to adjust the visibility of certain content (e.g., highly offensive content) as soon as adolescents enter their communication platform. The user and platform both need to be logged in to the data controller, and consent must be granted.

The Use Case: A minor (age 14) joins a chatroom. In Germany, at the age of 14 and younger, an individual is considered a child. The platform made the internal decision to protect the child by limiting all highly toxic statements (e.g., Holocaust denial) posted to the chat during the presence of a minor.

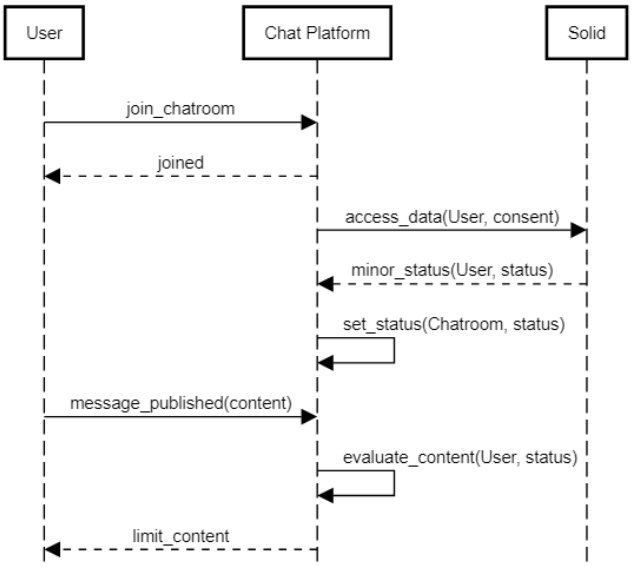

Figure 4. Sequence diagram of the UC 1.

4.2.1 Interaction sequences and data exchange:

See Figure 4 for the sequence diagram without the login, logout, or account creation.

- The user logs into the chat platform and provides consent for their data to be accessed and joins a chatroom.
- The chat platform requests the data controller to provide information if the user is a minor based on the country of origin and the personal age of the user.
- Whenever a message is sent, as long as the user is present in the chatroom, the platform can limit the posted content.
- When the user leaves the chatroom, withdrawing their consent for the data to be accessed, the chat is opened up for content suitable for adults.

4.3 Use Case 2: Contextualized Semantic Hate Speech Classification Combined with Counter Narrative Generation

This use case focuses on the moment an adult user of an online platform writes a problematic statement, such as denying the Holocaust, to other adults. In this setting, the platform needs to make multiple decisions. Firstly, is the message legal for the person to publish here? Secondly, is the statement against its internal guidelines? Thirdly, how to best address the statement.

After the message is classified as denying the Holocaust, to make a fitting legal decision, personal information such as the location of the user is needed. This information is obtained from the data controller. Now the statement can be evaluated against legal and ethical guidelines using a compliance check. Lastly, based on the personal information, a contextualized counter hate speech can be generated.

The Use Case: On a chat platform, a US citizen from California posts a statement denying the Holocaust. The platform can evaluate it based on the personal data of the user. In the US, this statement is covered under freedom of speech, making it legal for him to post. However, due to their internal guidelines, the platform still decides against the publication of the content. This reasoning is integrated into the created counter hate speech in English, explaining the reason

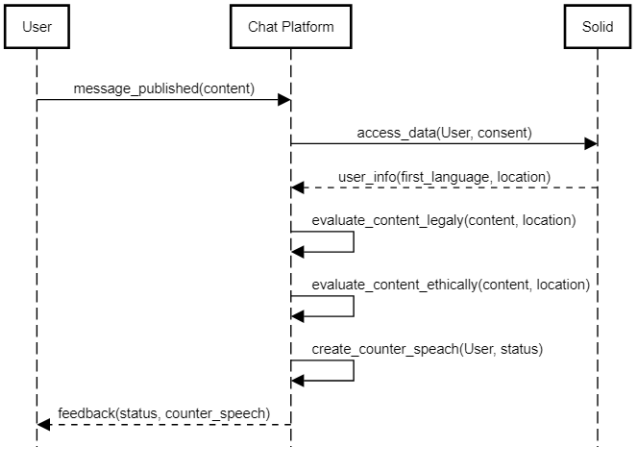

Figure 5. Sequence diagram of the UC 2.

for the message to be classified as problematic within the cultural context of America.

In the same session, a Greek user from Delphi publishes a statement also denying the Holocaust. The platform again evaluates it based on the personal data of the user. In Greece, it is not legal to deny the Holocaust. The platform therefore blocks the content from being posted and integrates this into the created counter hate speech in Greek, explaining the reason for the message to be classified as problematic within the cultural context of Greece.

4.3.1 Interaction sequences and data exchange:

See Figure 5 for the sequence diagram without the login, logout, or account creation.

- A harmful user posts a statement denying the existence of the Holocaust.
- The platform requests information from the data provider to determine if the statement violates their guidelines or local laws regulate such statements.
- Based on the input, the platform provides feedback that the statement is problematic not only based on internal guidelines but also due to local jurisdiction.
- The platform requests the first language and cultural background of the user from the data controller.
- With this information, the platform delivers understandable counter hate speech and provides context as to why the post was problematic.

5 Implementation

5.1 Architecture

As depicted in Figure 6, the prototype integrates the chat platform, the data controller (Solid), a hate speech detection service, and a Compliance Check implemented with Prova. All services are necessary to ensure safe and GDPR-compliant communication.

The chat platform serves as the interface for the user, supporting real-time messaging, and is designed to handle concurrent users. If the user sends a message to the platform (1. in Figure 6), the platform can request personal data from the user via the data controller, implemented as a Solid application (2.). After the controller provides the information (3.), the platform can forward the personal information

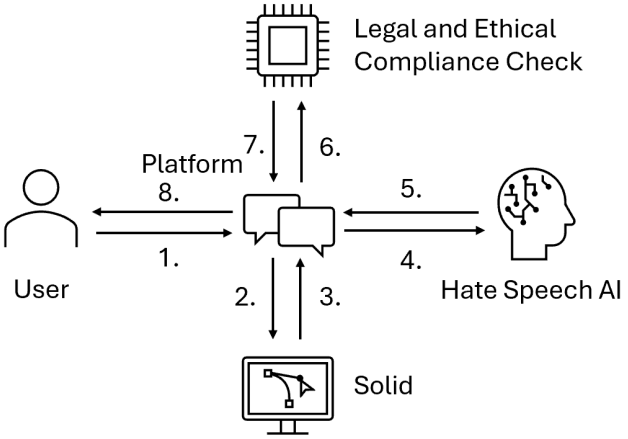

Figure 6. First rough architecture overview.

and the original message to the hate speech API (4.), which evaluates the content for hate speech, such as Holocaust denial, and generates personalized counter hate speech (5.). Based on the classification result, the platform can then request the Compliance Check (6.) to determine if the message violates legal or ethical standards (7.). Based on all responses, the platform can act accordingly and interact with the original message (8.).

5.2 *Solid*

The chat application has been implemented as a React[2] application. For the authentication and communication with the Solid platform, Inrupt's JavaScript Client Libraries and the React SDK[3] were used.

User data relevant to the use of the chat are stored in the user's profile (which is a standard Solid dataset and can be assumed to exist for every Solid user). The user's name, age and location of origin are stored as RDF triples using the vCard[4] and FOAF[5] vocabularies, respectively. User data retrieved from the Solid Pod is retained in the chat application for the duration of a chat session only. No personal data is permanently stored outside of the user's Solid Pod.

5.3 *Chat*

The chat application has been implemented using components from the Chat UI Kit[6]. The public demonstrator instance comes preconfigured with four chatrooms, each of them containing one virtual chat partner with different age and location of origin.

As soon as a new chat message is sent to any of the chatrooms, the text of the message is forwarded to the hate speech detection endpoint where it is being analyzed for hateful content (see also Section 5.4). The hate speech detection endpoint returns information about which kind of hate speech was detected (if any) and, if applicable, a numerical score ranging over 1-5 indicating the severity of the hate speech.

If the outcome is positive, a request containing the hate speech analysis result, age and location of the hateful comment's originator, as well as information about whether minors are present in the chatroom are sent to the legal and ethical compliance checker. The latter

[2] https://react.dev
[3] https://docs.inrupt.com/developer-tools/
[4] https://www.w3.org/TR/vcard-rdf/
[5] http://xmlns.com/foaf/spec/
[6] https://chatscope.io

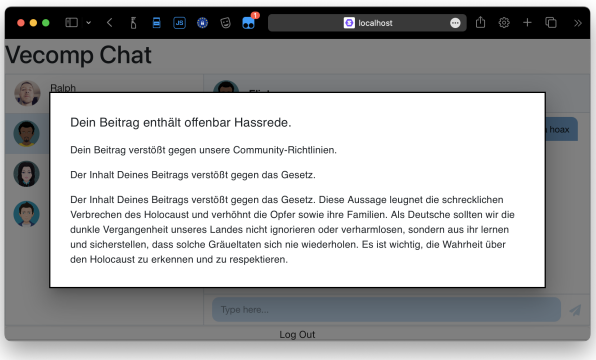

Figure 7. Screenshot of the chat application displaying a warning message in German that a user's post contains hate speech violating community guidelines and national law as well as a personalized counter hate speech message explaining the decision.

one decides whether the given instance of hate speech constitutes a violation against ethical norms (such as the chat service's community guidelines) or legal norms (considering the user's location) or both. See Section 5.5 for details on the compliance checking process.

In case of the presence of hate speech with ethical but without legal relevance, the hate speech mitigation endpoint is requested to generate appropriate counter speech.

Depending on the outcome, the chat application suppresses the message and presents the harmfully acting user with a warning message containing details about the reason for the intervention and the counter speech message. Figure 7 shows a screenshot of the application displaying a warning message about the violation of ethical and legal rules by one of the users' posts.

5.4 *Hate Speech Detection*

As shown by Kumarage et al. [15], OpenAI's LLMs are able to achieve stellar results in the tasks of hate speech detection. To build a simple but effective classifier, this research utilizes the OpenAI API with the GPT-3.5-turbo model as the foundation for hate speech detection. Due to its relatively recent release, size, economic considerations, and performance on academic benchmarks [15].

Ekin [4] suggests different approaches to prompt engineering. In his work, basic and advanced strategies are outlined. Due to hate speech detection not being the main focus of this work, only basic methods are used, applying templates and iterative testing and refinement. The advanced strategies would involve utilizing temperature and token control, prompt chaining, and adapting prompts. For this approach, templates were designed and then iteratively tested and refined until the results were satisfactory to the group of researchers.

Example prompt: "Analyze the following text for hate speech and respond with 'hate' or 'no-hate'. Provide a level of hate from 1-5, and indicate if Holocaust denial is present."

In the field of counter hate speech generation, Wang et al. [29] shows the strong capabilities of automatic counter hate speech generation utilizing GPT-3.5. They highlighted these capabilities while also raising concerns about the high need for well-designed prompts. Doğanç and Markov [3] demonstrate the possibility of creating high-quality counter hate speech statements by including personalized aspects. Based on these findings, this research again utilized templates and iterative testing and refinement to design the counter-narrative generation.

Example prompt: "Generate a counter speech for the following

text. You are an observer. The response should be between 50-100 words, contextualizing the problematic statements for someone of {national_origin} origin and providing the counter speech in {language}."

The hate speech detection and counter hate speech generation service is provided via a Swagger-based API interface. The underlying code is Python. The cost per call to the OpenAI API is $0.50 per 1M tokens[7].

5.5 *Compliance Checking with Prova*

The compliance checking mechanism, implemented in Prova, evaluates possible violations of legal and ethical standards. It functions as a service implemented in Prova and Java, that accepts HTTP requests of the following set of parameters: `user_location`, `user_age`, `chat_context`, `hate_speech_score`, and `hol`. The parameter `hol` characterizes the user message about the presence (or absence) of Holocaust denial, while the parameter `chat_context` denotes the general context of the chat, for example, if it is a chat where minors are participating. While not all parameters are used for this use case, they are included, aiming at future expansion.

The parameters (and their values) are converted to Prova slots (pairs of key and value), and are passed as messages to the two rulebases that perform, in turn, the legal and the ethical check.

First, the legal checker is invoked, using a subset of the slots, to check for potential legal violations of the user message. This depends on the message content, as well as the user location. The countries where Holocaust denial is a legal violation is provided through the `illCountry` predicate. In this case, three rule variants exist: 1. if the user message denies the Holocaust and the user location is in a country where Holocaust denial is illegal, 2. if the user message denies the Holocaust and the user location is not in a country where Holocaust denial is illegal, 3. if the user message does not deny the Holocaust

```
legalChecker() :-
     rcvMult(X,P,F,executionRequest,
         ↪{hol->hol_denial,user_location->L})
         ↪[illCountry(L)],
     spawn(X,$Service,result,
         ↪["legal_violation",
         ↪"Holocaust Denial"]),
     spawn(X,$Service,resume,[]).

legalChecker() :-
     rcvMult(X,P,F,executionRequest,
         ↪{hol->hol_denial,user_location->L})
         ↪[not(illCountry(L))],
     spawn(X,$Service,resume,[]).

legalChecker() :-
     rcvMult(X,P,F,executionRequest,{hol->H})
         ↪[not_equal(H,hol_denial)],
     spawn(X,$Service,resume,[]).
```

As shown above, the legal checker rulebase first selects the relevant messages through pattern matching over the slots (e. g., `user_location->L`), and then irrevocably accepts them if the guard (e. g., `[not(illCountry(L))]`) is satisfied, proceeding with calling outside Java methods that update the service's answer. In particular, `spawn(X,$Service,result,["...", "..."])` calls the Java method `result(String, String)`, which updates the answer with a kind of violation (`legal_violation`, or `ethical_violation`), while `spawn(X,$Service,resume,[])` invokes the method `resume()` that invokes a `notifyAll()` Java call. The latter is implemented for performance reasons, resuming the execution of the main Java thread (as Prova runs on different threads) as soon as Prova updates the answer. The third rule exists for the performance reasons mentioned above.

After the legal check and the potential update of the response, the ethical checker is called, to finalize the response. It contains analogous rules for the ethical checking, where the location of the user is not checked (Holocaust denial is unethical regardless of the user location), as well as a check for other ethical violations denoted by the parameter `hate_speech_score`.

The final response is provided in JSON form, for example

```
{
   "response":{
      "legal_violation":{
         "reason":"Holocaust Denial"
      },
      "ethical_violation":{
         "reason":"Holocaust Denial",
         "score":5
      }
   }
}
```

6 Discussion and Ethical Consideration

The created platform establishes the primary requirements of the GDPR (see Section 1) by separating the individual components into data subjects, data controllers, data processors, and an independent identification service. It provides users with a clear option to not only consent to their data sharing but also revoke access at any time. Additionally, by storing the shared data in their individual Solid Pods, which are provider-independent, users have full control over the type of data shared and all current access rights. While there are more aspects to the GDPR, this research covers the main parts and adopts a similar approach to other GDPR-compliant applications, such as in [28].

Regarding the DSA, three key aspects were introduced in Section 1. Firstly, the DSA does not harmonize what constitutes illegal content. The introduced compliance checker can include different legal and ethical definitions of hate speech. It is important that personal information needed to make these decisions can be shared securely and legally within this system, addressing a major open problem of the DSA. While the focus of this application was not on modeling all legislation regarding illegal content or ethical understanding of hate speech, the use cases were designed with a clear, small scope to show that the architecture and application can handle such a complex setting and can now be extrapolated and generalized to a broader spectrum.

Secondly, based on the DSA, online platforms that involve minors must take measures to ensure a high level of privacy, safety, and security. The proposed platform demonstrates this in the first use case, showing that the system can account for the presence of minors and adapt its behavior accordingly. A high level of data security is universally fulfilled with the proposed Solid infrastructure and GDPR-

[7] https://openai.com/api/pricing/

compliant structuring. Similar to the second point, the application does not introduce a complete and absolute solution on how to handle minors within a social platform but rather shows a way to generally provide privacy, safety, and security. This concept must now be adapted and fitted to more advanced features.

Thirdly, hosting providers must conduct fair content moderation. Users must be informed about moderation decisions, for example, whether the action is based on legal violations or violations of the terms of use. This key aspect is covered as shown in use case two. Here, the user is not only informed if their content was removed based on legal or ethical concerns but also receives a personalized response in their native language and with consideration of their social context, provided in natural language. Furthermore, it is important to mention that personal information is used in the ethical compliance checker to identify if local law was broken, making it a context-based hate speech detection system.

Content moderation is always a fine line between protecting people from online harm and limiting the ability to express oneself freely. This research developed a tool that supports content moderation, emphasizing that the researchers advocate for human-in-the-loop moderation approaches. It is possible that the LLM will make classification mistakes, this can only be ultimately solved by a human in the loop. Since both are prone to error, a mixed approach seems the best solution. This research is not intended to be viewed as a fully automatic solution. Furthermore, the proposed solution can include contextual personal information to make more informed legal and ethical decisions, distinguishing it from existing solutions that can handle either a mixture or just one type of information.

Using Large Language Models (LLMs) to generate counter speech based on personal attributes is a very young research discipline. While initial studies show that it is possible, no large-scale testing on the reliability or ethical aspects has been conducted. In this work, only language and country of origin are used for the generation process, both with explicit, always revocable consent. The "language" attribute is necessary to address the person in a format they understand best. The attribute "country of origin" could be more problematic, as it may result in unfitting counter hate speech. However, this risk is minimal considering that the LLMs used have safeguards in place to prevent discrimination and hate in their responses.

By introducing a legal and ethical compliance check, this research ensures a clear distinction between legal and ethical considerations, while also protecting legal and ethical statements. This work is in the public interest, focusing primarily on legislation such as the GDPR and DSA. The risk of sharing sensitive personal data is more manageable due to the full knowledge, control, and consent of the person sharing the data, in contrast to other standard data-sharing practices.

7 Conclusion and Future Work

The research outlines a GDPR-compliant application in the field of hate speech moderation. It lays the groundwork for key aspects of future DSA compliance. Two new use cases are introduced and implemented using Python, Prova, and Java. The first use case covers a key requirement regarding the protection of minors online introduced by the DSA. The second one shows the possibility of fair content moderation.

The architecture consists of four components: a platform that serves as the interface to the user and manages communication with the other tools, a Solid instance for access, permission, and storage of personalized user data, and identification of the users, an API that detects hate speech and Holocaust denial in text and generates counter hate speech based on personal attributes, and the Legal and Ethical Compliance Checker that evaluates specific instances based on Prova implementation for different legal and ethical scenarios. The compliance checker is able to contain formalized legislative rules and, based on the country of origin, identify if the given statement is considered illegal in a certain country (demonstrated in the case of Holocaust denial in Europe).

The architecture is clearly split into the different stakeholders required by the GDPR, and the required rights to consent and withdraw consent to data sharing are fulfilled.

In general, the application is the first known prototype to address these challenges arising with the DSA and GDPR in the context of content moderation. It provides a working demonstrator that shows the applicability and functionality of the proposed architecture and solutions.

In the future, more legal definitions need to be included in the compliance checker, extending on the one trial implementation. Furthermore, one of the strong suits that need to be explored is the possibility of using personal information directly in the LLM to identify context-based hate speech. The architecture could be expanded to also include a human-in-the-loop aspect for better safety and quality control. The proposed system needs to be further evaluated regarding its usability but also scalability and performance aspects. Lastly, the application could be expanded upon in the sense of other DSA aspects.

Acknowledgements

This work has been partially funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) project RECOMP (DFG – GZ: PA 1820/5-1) and by the German Federal Ministry of Education and Research, project "Terminology and Ontology-Based Phenotyping (TOP)" (grant number: 01ZZ2018).

References

[1] T. Caselli, V. Basile, J. Mitrović, and M. Granitzer. HateBERT: Retraining BERT for abusive language detection in English. In A. Mostafazadeh Davani, D. Kiela, M. Lambert, B. Vidgen, V. Prabhakaran, and Z. Waseem, editors, *Proceedings of the 5th Workshop on Online Abuse and Harms (WOAH 2021)*, pages 17–25, Online, Aug. 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.woah-1.3. URL https://aclanthology.org/2021.woah-1.3.

[2] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In J. Burstein, C. Doran, and T. Solorio, editors, *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1423. URL https://aclanthology.org/N19-1423.

[3] M. Doğanç and I. Markov. From generic to personalized: Investigating strategies for generating targeted counter narratives against hate speech. In Y.-L. Chung, H. Bonaldi, G. Abercrombie, and M. Guerini, editors, *Proceedings of the 1st Workshop on CounterSpeech for Online Abuse (CS4OA)*, pages 1–12, Prague, Czechia, Sept. 2023. Association for Computational Linguistics. URL https://aclanthology.org/2023.cs4oa-1.1.

[4] S. Ekin. Prompt engineering for chatgpt: A quick guide to techniques, tips, and best practices, 05 2023.

[5] J. Fillies, M. Hoffmann, and A. Paschke. Multilingual hate speech detection: Comparison of transfer learning methods to classify german, italian, and spanish posts. In *2023 IEEE International Conference on Big Data (BigData)*, pages 5503–5511, Los Alamitos, CA, USA, dec 2023. IEEE Computer Society. doi: 10.1109/BigData59044.2023.10386244. URL https://doi.ieeecomputersociety.org/10.1109/BigData59044.2023.10386244.

[6] A. Goossens, S. Vandevelde, J. Vanthienen, and J. Vennekens. GPT-3 for decision logic modeling. *Proceedings of the 17th International Rule Challenge and 7th Doctoral Consortium@ RuleML+ RR 2023 co-located with 19th Reasoning Web Summer School (RW 2023) and 15th DecisionCAMP 2023 as part of Declarative AI 2023*, 3485:1–14, 2023.

[7] G. Governatori, F. Olivieri, S. Scannapieco, and M. Cristani. Designing for compliance: Norms and goals. In *International Workshop on Rules and Rule Markup Languages for the Semantic Web*, pages 282–297. Springer, 2011.

[8] H. Hayashi and K. Satoh. Online htn planning for data transfer and utilization considering legal and ethical norms: Case study. In *ICAART (1)*, pages 154–164, 2023.

[9] H. Hayashi, T. Mitsikas, Y. Taheri, K. Tsushima, R. Schäfermeier, G. Bourgne, J.-G. Ganascia, A. Paschke, and K. Satoh. Multi-agent online planning architecture for real-time compliance. In *Proceedings of the 17th International Rule Challenge and 7th Doctoral Consortium RuleML+RR 2023*, volume 3485. CEUR, 2023. URL https://ceur-ws.org/Vol-3485/.

[10] M. Husovec and I. Roche Laguna. Digital services act: A short primer. *Martin Husovec and Irene Roche Laguna, Principles of the Digital Services Act (Oxford University Press, Forthcoming 2023)*, 2022.

[11] Y. Kim, S. Park, Y. Namgoong, and Y.-S. Han. ConPrompt: Pre-training a language model with machine-generated data for implicit hate speech detection. In H. Bouamor, J. Pino, and K. Bali, editors, *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 10964–10980, Singapore, Dec. 2023. Association for Computational Linguistics. URL https://aclanthology.org/2023.findings-emnlp.731.

[12] G. Kober, L. Robaldo, and A. Paschke. Modeling Medical Guidelines by Prova and SHACL Accessing FHIR/RDF. Use Case: The Medical ABCDE Approach. In *dHealth 2022*, pages 59–66. IOS Press, 2022.

[13] A. Kozlenkov. *Prova Rule Language version 3.0 User's Guide*, 2010. URL https://github.com/prova/prova/tree/master/doc.

[14] A. Kozlenkov, R. Penaloza, V. Nigam, L. Royer, G. Dawelbait, and M. Schroeder. Prova: Rule-Based Java Scripting for Distributed Web Applications: A Case Study in Bioinformatics. In T. Grust, H. Höpfner, A. Illarramendi, S. Jablonski, M. Mesiti, S. Müller, P.-L. Patranjan, K.-U. Sattler, M. Spiliopoulou, and J. Wijsen, editors, *Current Trends in Database Technology – EDBT 2006*, pages 899–908, Berlin, Heidelberg, 2006. Springer. ISBN 978-3-540-46790-8.

[15] T. Kumarage, A. Bhattacharjee, and J. Garland. Harnessing artificial intelligence to combat online hate: Exploring the challenges and opportunities of large language models in hate speech detection, 2024.

[16] P. Liu, W. Li, and L. Zou. NULI at SemEval-2019 task 6: Transfer learning for offensive language detection using bidirectional transformers. In J. May, E. Shutova, A. Herbelot, X. Zhu, M. Apidianaki, and S. M. Mohammad, editors, *Proceedings of the 13th International Workshop on Semantic Evaluation*, pages 87–91, Minneapolis, Minnesota, USA, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/S19-2011. URL https://aclanthology.org/S19-2011.

[17] E. Mansour, A. V. Sambra, S. Hawke, M. Zereba, S. Capadisli, A. Ghanem, A. Aboulnaga, and T. Berners-Lee. A demonstration of the solid platform for social web applications. In *Proceedings of the 25th international conference companion on world wide web*, pages 223–226, 2016.

[18] B. Mathew, P. Saha, S. M. Yimam, C. Biemann, P. Goyal, and A. Mukherjee. Hatexplain: A benchmark dataset for explainable hate speech detection. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(17):14867–14875, May 2021. doi: 10.1609/aaai.v35i17.17745. URL https://ojs.aaai.org/index.php/AAAI/article/view/17745.

[19] T. Mitsikas, R. Schäfermeier, and A. Paschke. Modeling medical data access with Prova. *New Frontiers in Artificial Intelligence*, page 35, 2024.

[20] M. Mozafari, R. Farahbakhsh, and N. Crespi. Hate speech detection and racial bias mitigation in social media based on bert model. *PLOS ONE*, 15(8):1–26, 08 2020. doi: 10.1371/journal.pone.0237861. URL https://doi.org/10.1371/journal.pone.0237861.

[21] A. Paschke. *Rules and Logic Programming for the Web*, pages 326–381. Springer, Berlin Heidelberg, 2011. ISBN 978-3-642-23032-5. doi: 10.1007/978-3-642-23032-5_6.

[22] A. Paschke and M. Bichler. Knowledge representation concepts for automated SLA management. *Decision Support Systems*, 46(1):187–205, 2008. ISSN 0167-9236. doi: 10.1016/j.dss.2008.06.008.

[23] A. Paschke and H. Boley. Reaction RuleML 1.0 for Distributed Rule-Based Agents in Rule Responder. In *Proceedings of the RuleML 2014 Challenge and the RuleML 2014 Doctoral Consortium, hosted by the 8th International Web Rule Symposium (RuleML 2014)*. CEUR.org, 2014.

[24] M. Ramachandran, N. Chowdhury, A. Third, J. Domingue, K. Quick, and M. Bachler. Towards complete decentralised verification of data with confidentiality: Different ways to connect solid pods and blockchain. In *Companion proceedings of the web conference 2020*, pages 645–649, 2020.

[25] S. T. Roberts. *Content Moderation*, pages 1–4. Springer International Publishing, Cham, 2017. ISBN 978-3-319-32001-4. doi: 10.1007/978-3-319-32001-4_44-1. URL https://doi.org/10.1007/978-3-319-32001-4_44-1.

[26] A. V. Sambra, E. Mansour, S. Hawke, M. Zereba, N. Greco, A. Ghanem, D. Zagidulin, A. Aboulnaga, and T. Berners-Lee. Solid: a platform for decentralized social applications based on linked data. *MIT CSAIL & Qatar Computing Research Institute, Tech. Rep.*, 2016.

[27] K. Satoh, K. Asai, T. Kogawa, M. Kubota, M. Nakamura, Y. Nishigai, K. Shirakawa, and C. Takano. PROLEG: an implementation of the presupposed ultimate fact theory of Japanese civil code by prolog technology. In *JSAI international symposium on artificial intelligence*, pages 153–164. Springer, 2010.

[28] R. Schäfermeier, T. Mitsikas, and A. Paschke. Modeling a GDPR compliant data wallet application in Prova and AspectOWL. In *RuleML+ RR (Companion)*, 2022.

[29] H. Wang, M. S. Hee, M. R. Awal, K. T. W. Choo, and R. K.-W. Lee. Evaluating GPT-3 generated explanations for hateful content moderation, 2023.

Knowledge-Augmented Reasoning for EUAIA Compliance and Adversarial Robustness of LLMs

Tomas Bueno Momcilovic[a,*], **Dian Balta**[a], **Beat Buesser**[b], **Giulio Zizzo**[c] **and Mark Purcell**[c]

[a]fortiss Research Institute of the Free State of Bavaria, Munich, Germany
[b]IBM Research Europe, Zurich, Switzerland
[c]IBM Research Europe, Dublin, Ireland

Abstract. The EU AI Act (EUAIA) introduces requirements for AI systems which intersect with the required processes for establishing adversarial robustness. However, given the ambiguous language of regulation and the dynamicity of adversarial attacks, developers of systems with highly complex models such as LLMs may find their effort to be duplicated without the assurance of having achieved either compliance or robustness. This paper presents a functional architecture that focuses on bridging the two properties, by introducing components with clear reference to their source. Taking the detection layer recommended by the literature, and the reporting layer required by the law, we aim to support developers and auditors with a reasoning layer based on knowledge augmentation (rules, assurance cases, contextual mappings). Our findings demonstrate a novel direction for ensuring LLMs deployed in the EU are both compliant and adversarially robust, which underpin trustworthiness.

1 Introduction

The European Union (EU) bases trustworthiness of artificial intelligence systems (AIS) on three properties: lawful, ethical, and robust [7]. The EU AI Act (EUAIA, [3]) is an upcoming regulation that sets obligations on the lawful design and implementation of AIS in the EU. Its content outlines the high-level requirements for improving the auditability of the AIS, whose generic descriptions are interepretable across contexts.

However, for properties such as adversarial robustness, providers of AIS with large language model-based (LLM) components are facing a difficult and highly dynamic challenge whose boundaries are not yet known. Providers in the EU who would like to ensure both compliance and robustness, are doubly burdened. First there is the need to constantly readapt their defenses against novel adversarial attacks [5], and second is the overhead for correctly interpreting "compliant robustness" with auditable evidence.

This paper presents a novel approach of knowledge augmentation for aligning adversarial robustness of LLMs with EUAIA compliance. By integrating detection, reasoning and reporting layers alongside the layer for interacting with users, we propose a comprehensive functional architecture as a reference for ensuring the AIS is dynamically protected and auditable. The research provides a framework for combining robustness and compliance activities while retaining the provenance to the requirements.

* Corresponding Author. Email: momcilovic@fortiss.org

Our roadmap centers on solution-oriented requirements engineering [12] and knowledge augmentation (i.e., knowledge representation and reasoning [9]) to develop the architecture of our prototype. This process of creating a blueprint of a compliant LLM defense against adversarial attacks involves three steps.

First, we extract the legal duties and relevant stakeholders from the EUAIA ([6, 14]; cf. [2] for the expanded list), and structure them into draft requirements in the next section. This approach takes inspiration from [4]. Second, concepts and relations surrounding LLMs are represented in a simple ontology [10]. State-of-the-art attacks and defenses in the context of natural language tasks are recovered from preprints [16, 15, 5]. The third step is a representation of the knowledge in a cyclical process model of actions between stakeholders and components, and the corresponding sources.

2 Requirements

The EUAIA places AI systems that are deployed in particular products or domains under categories of risk, where high-risk AI systems play a central role [3]. The regulation which has been adopted in 2024 places duties on stakeholders at design and runtime. Before standards are expected in the following years, these duties provide a basis for safety- and security-oriented requirements.

General-purpose AI models (GPAI, Art. 3, [3]) such as LLMs are not inherently high-risk. However, their broad capabilities and wide attack surface have over time crystallized similar requirements with respect to adversarial robustness. In examples provided by an ever-increasing body of work (cf. [16]), adversaries of an LLM can include third parties with malicious intentions, curious users who test the boundaries, and even completely benign users whose prompts elicit harmful or otherwise unintended output.

Based on an analysis of requirements in Table 1, EUAIA compliance and adversarial robustness are complementary properties, despite the difference in details. On the one hand, the requirements that are derived from the regulation (cf. [2] for expanded list) provide a generic description of stakeholders (R0), risk management (R3) and cybersecurity measures, and the need for human oversight (R10) and reporting (R12). On the other hand, the state-of-the-art literature introduces specific roles (R1), the detection of automated (R4), semi-automated (R5), and manual attacks (R15), and sustained coverage of these threats (R7). However, aside from the direct references to the term in EUAIA (R6, R12), the two sources emphasize different facets of a larger system - i.e., components for assuring the quality of

Table 1. Requirements related to adversarial robustness and their sources

id	Requirement	Source
R0	Include the following stakeholders: user; (malicious) third party; GPAI provider; AIS provider; GPAI or AIS deployer; national competent authority; market surveillance authority; AI office.	Art. 3 & Rec. 76 [3]; cf. [2]
R1	Include the following roles: user; developer (i.e., system or LLM engineer, researcher, scientist); auditor.	[13]
R2	Identify, evaluate and mitigate *reasonably foreseeable* risks of the system.	Art. 9 Para. 2 [3]
R3	Ensure appropriate and adequate risk management measures.	Art. 9 Para. 5 [3]
R4	Detect automated attacks such as prompts with randomized perturbations.	[16]
R5	Detect semi-automated attacks such as heuristic-based exploitation of the undertrained aspects of the model.	[5]
R6	Establish cybersecurity measures against adversarial and poisoning attacks.	Art. 15 Para. 5 [3]
R7	Achieve sustained coverage of detected and prevented attacks above a predefined threshold.	[1]
R8	Establish an appropriate level of robustness and cybersecurity.	Art. 15 Para. 1 [3]
R9	Provide information about robustness and cybersecurity (e.g., metrics) and their limitations in instructions for use.	Art. 13 Para. 3 & Annex IV [3]
R10	Design system for effective human oversight regarding safety monitoring and prevention/minimization of reasonably foreseeable misuse.	Art. 14 Para. 2 [3]
R11	Design appropriate functionalities for human overseers to monitor for "anomalies, dysfunctions and unexpected performance."	Art. 14 Para. 4 [3]
R12	Report on measures and tests used for adversarial testing, model alignment, and fine-tuning.	Art. 53 Para. 1 & Annex XI [3]; Art. 11 & Annex IV [3]
R13	Supply information on testing, safeguards and risk mitigation measures at the request of the AI Office.	Art. 92 Para. 5 & 7 [3]
R14	Establish and report on the definite, reasonably likely or suspected causal link between the system and a serious incident.	Art. 73 Para. 2-6 [3]
R15	Detect manual attacks based on patterns of persuasion (i.e., "jailbreaking").	[15]
R16	Notify supervising stakeholder of a serious incident.	Art. 73 Para. 1, 7-8 & 11 [3]

adversarial robustness beyond the purely functional components of an LLM-supported application. In the next section, we introduce one approach to satisfying both facets.

3 Functional Architecture and Workflow

3.1 Architecture

The functional architecture depicted in Figure 1 is composed of a cyclical workflow linking four stakeholders and four layers of components. Stakeholders and components are connected with arrows denoting action IDs (A), as described in Table 2, whereby each non-functional element of the architecture has a corresponding requirement ID (R) identified in Table 1.

The stakeholders include users, LLM developers, AIS developers and auditors, who represent the various roles involved in the design and implementation of AIS, with the corresponding EUAIA-defined role in parentheses. Auditors and users are external temporary roles, whereby a user can be benign, curious or malicious. Developers are internal and lasting roles, whose responsibilities depend on the access to the internal workings of an LLM and the system deploying it.

The layers involve the interaction layer which fulfils the functional requirements of an AIS, and detection, reasoning and reporting layers which fulfill the quality requirements underlying robustness. In other words, the first layer is enough to establish a fully working AIS, without special consideration for other properties. Interaction has a simple structure inspired by practice [11], containing the user-facing application (i.e., the interface between the user and the AIS) and the LLM.

Table 2. Actions and their descriptions.

id	Action
A0	Displays relevant information about the LLM, disclaimers, and limitations.
A1	Enters a prompt.
A2	Forwards the prompt and the metadata.
A3	Provides the first batch of classification using the deployed detectors.
A4	Provides the evaluation with the prompt (if benign) or warning (if malicious).
A5	Provides the generated result according to the evaluation.
A6	Displays the generated result.
A7	Provides data on the metrics and thresholds used for detectors.
A8	Displays the metrics and relevant data.
A9	Provides the second batch of classification using all relevant detectors and their combinations.
A10	Provides a counterfactual assessment comparing the coverage and accuracy of deployed and non-deployed detector combinations.
A11	Displays the counterfactual assessment.
A12	Reconfigures the detector combinations and their threshold values.
A13	Provides flagged LLM output (i.e., anomaly or incident) and the corresponding input prompt.
A14	Provides the data on the detected anomalies.
A15	Displays information about the individual or group of anomalies.
A16	Makes adjustments to the LLM based on the provided data.
A17	Provides data about the anomalies flagged as incidents.
A18	Displays information about the (serious) incidents.

3.2 Workflow

Detection is based on input and output classification, following the current paradigm of dealing with adversarial attacks [1]. Input detectors have thresholds based on some combination of single and n-pairs of metrics. Metrics denote ways of measuring particular properties of input prompts, examples including perplexity (pp; i.e., the extent to which the model is "surprised" by a prompt), context length (cl) and

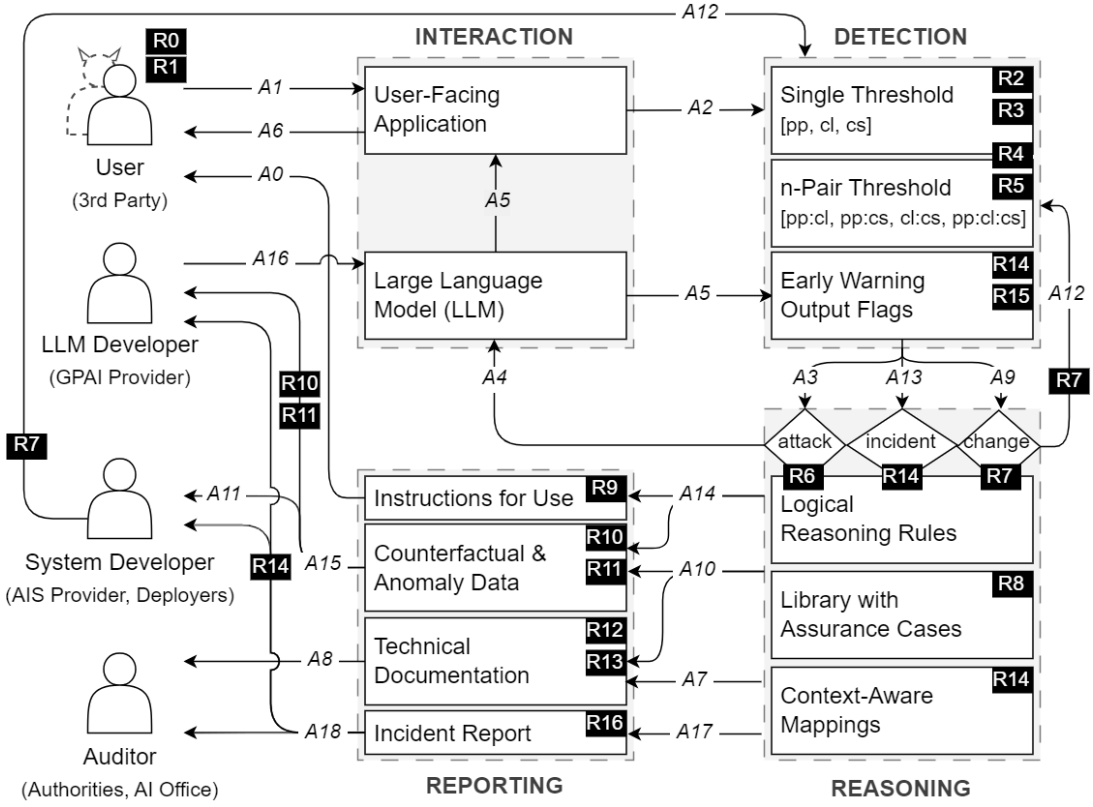

Figure 1. Reference functional architecture for compliant adversarial robustness of LLM-based AI systems.

character set size (cs). Output detectors attempt to detect unexpected LLM results which may be results of undetected attacks. They can be implemented similar as for inputs, but also using flags for harmful keywords to provide an early warning to the developer.

Reasoning serves as the middleware between other layers by decoupling the logic from detection, interaction and reporting activities. The layer provides a set of rules derived using deductive or inductive reasoning, which are behind decisions to classify an input as an attack, an output as an incident, or detector performance as a trigger for change. The library with assurance cases is a set of graphs connecting claims about satisfied requirements relating to compliance and robustness, with the evidence from chosen strategies. Given the adaptability of LLMs and the context-specific properties, context-aware mappings provide the needed metadata to separate the rules and assurance case elements to what they are appropriate. In addition, these mappings enable the variables in reports to be linked with actual values.

Finally, the reporting layer is primarily based on the EUAIA need for human oversight. Instructions for use and technical documentation are factsheets for users and auditors respectively. However, given the relevance of figures and test results to the monitoring of adversarial robustness, these components are useful to developers for AIS debugging and improvement as well. In addition, assessments based on counterfactuals and anomaly data allow the developers to monitor detectors with respect to needed changes. Incident reports are triggered by an event of a potentially successful attack; although primarily an EUAIA requirement for mandatory auditing of serious incidents, less critical but problematic anomalies provide an opportunity to developers to perform forensic analyses.

The Figure 2 depicts three main cyclic processes. The primary cycle is the simplest: a user enters a prompt into the application (A1), which is then forwarded to the input detectors (A2). The detectors' results are provided as input (A3) to a rule that classifies the prompt as safe or unsafe, passing on the prompt or the warning respectively to the LLM (A4). The LLM then generates instead elicits a warning to the user (A4, A5, A6). Relying only on this cycle would be a naive approach to handling adversarial attacks, whereby the developer would expect the detectors to perform well over time and prompts.

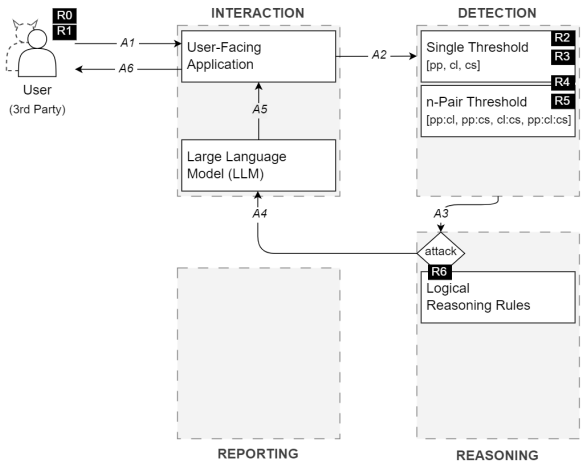

Figure 2. Primary cycle with interaction and basic attack detection.

The secondary cycle introduces the required auditability for EU-AIA compliance. The information about the deployed detectors is structured in assurance cases, which feed into the documentation (A7). This documentation provides an interface to the user to understand the model and its limitations before use (A0), and an interface to the auditor (A8) to establish a clear picture about the AIS.

This cycle also provides the basis for required dynamicity for adversarial robustness. Assurance cases are intended to provide the logic needed to evaluate detector performance. Given a number of prompts or some other triggering rule (A9), prompts would be processed through non-deployed detector combinations. This would provide the basis for counterfactually assessing the sustained robustness of the detectors (A10). This evaluation is initially be the responsibility of the AIS developer (A11), whose understanding of the context-sensitive performance and coverage would be needed to reconfigure the detectors (A12).

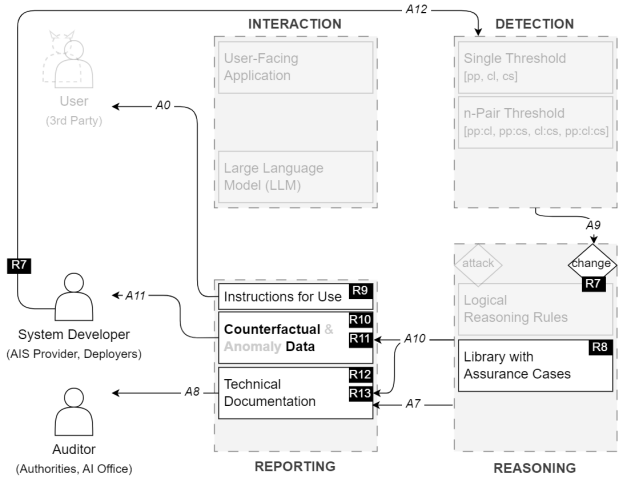

Figure 3. Secondary cycle with assurance, monitoring and reporting.

The tertiary cycle introduces mechanisms for handling failure systematically. An evaluation of the LLM output (A5), whether in real-time or delayed intervals, allows some successful attacks (i.e., incidents; A13) to be automatically detected. Here is context-specific information necessary to operationalize ambiguous EUAIA language: which risks or anomalies are "reasonably foreseeable" (A14) and worth exploring; which incidents are "serious" enough (A17) to demand contact with the auditor (A18); and when is a given risk management procedure not "suitable" anymore (A12). This cycle also proposes providing relevant information to the LLM developer, who may not be associated with the AIS directly, but nonetheless benefits from adversarially retraining the LLM, thereby making it more secure in the AIS as well.

4 Discussion and Conclusion

This paper introduces a knowledge-augmented framework designed to align the adversarial robustness of large language models with the EU AI Act compliance. By using a combination of detection, reasoning and reporting layers, we address the critical need for compliance and robustness in AI systems.

The functional architecture is meant as a reference for implementing physical components. For example, our early prototype implements simple detectors in Python, including n-pair detectors based on logistic regression classifiers pretrained on Hugging Face jailbreak data [8]. The reasoner is based on a combination of an assurance

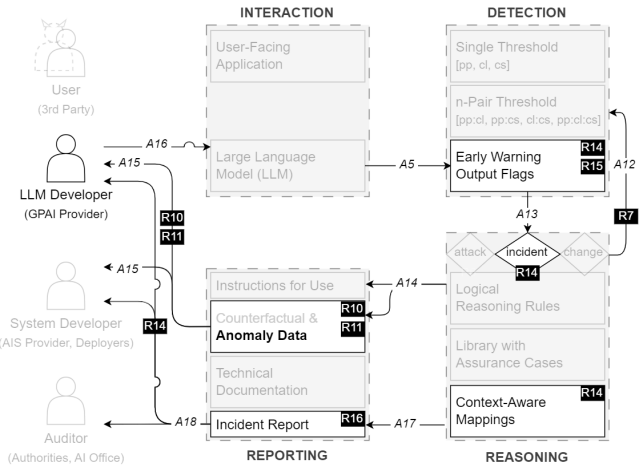

Figure 4. Tertiary cycle for advanced handling of failures.

case and an ontology, stored in the graph database, where evaluations are performed through queries. Additionally, graphical visualizations and textual data is generated in Jupyter Notebooks to provide clear and informative reporting. The interaction layer uses the streamlit package to provide a user-facing application, while GPT-2, accessed via the Hugging Face package, serves as the foundational LLM.

Our findings highlight a promising direction for developing resilient AI technologies capable of withstanding adversarial attacks while meeting regulatory standards. Future work will focus on the following aspects: (1) defining new detectors and combinations thereof, such as classifiers trained on larger samples of malicious and benign prompts; (2) expanding the reasoning based on the wider context, including computer language tasks (e.g., code translation); (3) evaluating the components of the architecture with respect to helping developers assure robustness and auditors determine compliance of the LLM-based systems.

Acknowledgements

This work was partially supported by financial and other means by the following research projects: DUCA (EU grant agreement 101086308), FLA (supported by the Bavarian Ministry of Economic Affairs, Regional Development and Energy), the DiProLeA (German Federal Ministry of Education and Research, grant 02J19B120 ff), as well as our industrial partners in the FinComp project. We thank the reviewers for their valuable comments.

References

[1] G. Alon and M. Kamfonas. Detecting language model attacks with perplexity. *arXiv preprint arXiv:2308.14132*, 2023.

[2] T. Bueno Momcilovic, B. Buesser, G. Zizzo, M. Purcell, and D. Balta. Towards assuring eu ai act compliance and adversarial robustness of llms. In *AI Act Workshop, 19. Internationale Tagung Wirtschaftsinformatik, 16. – 19. September (upcoming publication)*., 2024.

[3] European Parliament and Council of the European Union. Corrigendum to the position of the European Parliament adopted at first reading on 13 March 2024 with a view to the adoption of Regulation (EU) 2024/.....) of the European Parliament and of the Council laying down harmonised rules on artificial intelligence and amending Regulations (EC) No 300/2008, (EU) No 167/2013, (EU) No 168/2013, (EU) 2018/858, (EU) 2018/1139 and (EU) 2019/2144 and Directives 2014/90/EU, (EU) 2016/797 and (EU) 2020/1828 (Artificial Intelligence Act), 2024. URL https://www.europarl.europa.eu/doceo/document/TA-9-2024-0138-FNL-COR01_EN.pdf. Accessed: 2024-05-15.

[4] L. Floridi, M. Holweg, M. Taddeo, J. Amaya, J. Mökander, and Y. Wen. capai - a procedure for conducting conformity assessment of ai systems in line with the eu artificial intelligence act. March 23 2022. Available at SSRN: https://ssrn.com/abstract=4064091 or http://dx.doi.org/10.2139/ssrn.4064091.

[5] J. Geiping, A. Stein, M. Shu, K. Saifullah, Y. Wen, and T. Goldstein. Coercing llms to do and reveal (almost) anything. *arXiv preprint arXiv:2402.14020*, 2024.

[6] W. N. Hohfeld, D. Campbell, and P. A. Thomas. Fundamental legal conceptions as applied in judicial reasoning /. 2001. URL http://lawcat.berkeley.edu/record/1178561. First published 2001 by Ashgate Publishing.

[7] H.-L. E. G. O. A. INTELLIGENCE. Ethics guidelines for trustworthy AI. European Commission, 2019. URL https://ec.europa.eu/newsroom/dae/document.cfm?doc_id=60419.

[8] R. D. Jaramillo. Chatgpt jailbreak prompts, 2023. URL https://huggingface.co/datasets/rubend18/ChatGPT-Jailbreak-Prompts.

[9] C. M. Keet. *The What and How of Modelling Information and Knowledge: From Mind Maps to Ontologies*. Springer, Berlin, 2023.

[10] Ontotext. What is GraphDB? https://graphdb.ontotext.com/documentation/10.6/. Accessed: 2024/03/14.

[11] OpenAI. Introducing ChatGPT. https://openai.com/index/chatgpt, 2022. Accessed: 2024/02/25.

[12] K. Pohl. *Requirements Engineering: Fundamentals, Principles, and Techniques*. Springer, Berlin, Heidelberg, 2010. ISBN 9783642125775.

[13] H. Suresh, S. R. Gomez, K. K. Nam, and A. Satyanarayan. Beyond expertise and roles: A framework to characterize the stakeholders of interpretable machine learning and their needs. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*, New York, NY, USA, 2021. ACM.

[14] T. van Engers and R. van Doesburg. First steps towards a formal analysis of law. In G. G. Malzahn, D., editor, *eKNOW 2015: The Seventh International Conference on Information, Process, and Knowledge Management: February 22-27, 2015, Lisbon, Portugal*, pages 36–42, Wilmington, DE, 2015. IARIA.

[15] Y. Zeng, H. Lin, J. Zhang, D. Yang, R. Jia, and W. Shi. How johnny can persuade LLMs to jailbreak them: Rethinking persuasion to challenge AI safety by humanizing LLMs. *CoRR*, abs/2401.06373, 2024.

[16] A. Zou, Z. Wang, J. Z. Kolter, and M. Fredrikson. Universal and transferable adversarial attacks on aligned language models. *CoRR*, abs/2307.15043, 2023.

Author Index