2023

ArgFrame: A Multi-Layer, Web, Argument-Based Framework for Quantitative Reasoning

Lucas Rizzo
Technological University Dublin, Ireland, lucas.rizzo@tudublin.ie

Funder: This research received no external funding

Software Impacts 17 (2023) 100547

Contents lists available at ScienceDirect

Software Impacts

journal homepage: www.journals.elsevier.com/software-impacts

Original software publication

ArgFrame: A multi-layer, web, argument-based framework for quantitative reasoning

Lucas Rizzo

Technological University Dublin, Dublin, Ireland

ARTICLE INFO

Keywords:
Defeasible argumentation
Automated reasoning
Knowledge-based systems
Dung semantics
Data analysis

ABSTRACT

Multiple systems have been proposed to perform computational argumentation activities, but there is a lack of options for dealing with quantitative inferences. This multi-layer, web, argument-based framework has been proposed as a tool to perform automated reasoning with numerical data. It is able to use boolean logic for the creation of if-then rules and attacking rules. In turn, these rules/arguments can be activated or not by some input data, have their attacks solved (following some Dung or rank-based semantics), and finally aggregated in different fashions in order to produce a prediction (a number). The framework is implemented in PHP for the back-end. A JavaScript interface is provided for creating arguments, attacks among arguments, and performing case-by-case analyses.

Code metadata

Current code version	*v1.0*
Permanent link to code/repository used for this code version	https://github.com/SoftwareImpacts/SIMPAC-2023-305
Permanent link to Reproducible Capsule	
Legal Code License	GPL-3.0 license
Code versioning system used	Git
Software code languages, tools, and services used	PHP, JavaScript, SQL, Docker
Compilation requirements, operating environments & dependencies	Docker compose
If available Link to developer documentation/manual	https://github.com/LucasRizzo/argframe#readme
Support email for questions	lucasmrizzo@gmail.com

1. Introduction

Uncertainty associated with insufficient, inaccurate, or unreliable knowledge is unavoidable in everyday reasoning and in many real-world circumstances. Modelling such uncertainty has been addressed by many different approaches within Artificial Intelligence (AI). One of these is given by non-monotonic reasoning. Non-monotonicity, or defeasibility, represents the capacity for retracting a conclusion in light of new information. In order to formalise the practice of defeasible reasoning, computational argumentation, also referred to as defeasible argumentation [1,2], was proposed.

Several systems have been developed to perform computational argumentation activities [3], but there is a lack of options for dealing with quantitative reasoning. In this case, quantitative reasoning is understood as reasoning built upon domain knowledge and performed on quantitative data, thus being able to provide numerical inferences. Often, quantitative approaches in AI are deemed limited for their inability to provide justifiable conclusions [4]. Hence, this paper proposes a framework, $ArgFrame$, as a tool to perform automated reasoning with numerical data. Intuitively, it provides a higher degree of interpretability and transparency to a reasoning process performed over quantitative data. The reason for that is because models built with it employ human-defined arguments and, consequently, follow the way humans reason. This increases the models' explainability, which is essential for their adoption and usage in important areas like medical diagnosis, legal reasoning, and self-driving cars. The multi-layered structure depicted in Fig. 1 provides the basic building blocks for the creation of argument-based models of inference through $ArgFrame$.

The remainder of the manuscript is organised as follows: Section 2 describes the framework and its main features and functionalities. Section 3 outlines the framework's impact, applications, and scholarly publications that have employed it. Lastly, Section 4 provides limitations and future research directions.

E-mail address: lucas.rizzo@tudublin.ie.

https://doi.org/10.1016/j.simpa.2023.100547
Received 27 June 2023; Received in revised form 7 July 2023; Accepted 14 July 2023

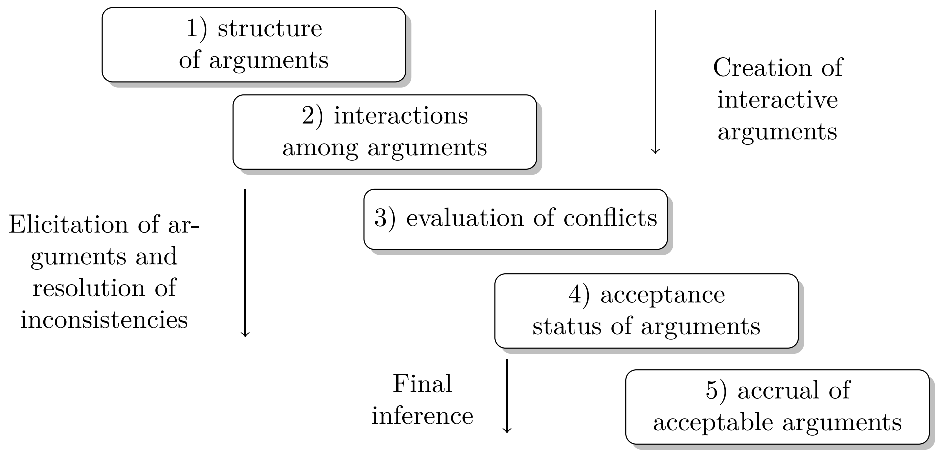

Fig. 1. Five layers structure [5] employed for the development of *ArgFrame*.

2. Software description

ArgFrame is capable of: (1) defining categorical or continuous features; (2) using boolean logic to create if-then rules and attacking rules based on the defined features; (3) activating (or not) the defined rules/arguments using input data in CSV format; (4) solving attacks following certain Dung or rank-based semantics [6–8]; (5) and finally, aggregating accepted rules/arguments in different ways in order to generate a prediction (a number). Each of these steps will be further detailed.

Features and conclusions: Features are defined in the first stage, under the “Feature set” option in the top menu. Features can be created by interacting with the feature table. Each feature can have multiple levels or categories. For instance, if we want to model the feature “age” we need to define at least one level for the feature to exist. A simple example would be to have “age” with three different levels: “young” for ages 0 to 18, “adult” for ages 19 to 60, and “elderly” for ages 61 to 120. It is important to note that each level corresponds to a specific value range. If the levels are not continuous, such as “yes" or “no” then the ranges can be defined with the same lower and upper values. Conclusions or target features can also be defined in a similar manner at the bottom of the feature table. We should emphasise that the level names, like “young” or “yes” can aid in the user’s reasoning process, but only their respective numerical ranges are used to generate an inference.

Arguments and attacks: Once features and conclusions have been defined, it is possible to use the “Graphs” option in the top menu. On this screen, you can define arguments and their interactions. Arguments have a simple structure “$premises \rightarrow conclusion$”, where “conclusion” is a target feature and “premises” are features concatenated by `AND` and `OR` operators. For example, suppose a target feature “mortality” defined as a scale from 0 to 1. In this case, it is possible to model the argument “elderly [61, 120] age `AND` male [0, 0] sex → higher [0.75, 1] mortality”. If needed, you can also include parentheses to build more complex premises. All of these actions are performed through the interface, without requiring any coding expertise. Once an argument is completed, it is added to the current graph as a circular node in the interface. Finally, attacks between arguments can be defined in the graph interface. Arrows between nodes represent binary attacks, where one argument serves as the source attacking another target argument.

Generating inferences: Once an argumentation graph has been created, you can input data to generate case-by-case inferences. The data should be imported in CSV format, with headers matching the features of the arguments. Based on the input data, the boolean clauses (premises) of each argument can be evaluated as either true or false for each row in the imported data. Arguments whose clauses evaluate as true are activated, while those that evaluate as false are not considered when producing the final inference. This process generates a sub-graph with only the activated arguments for each row in the dataset. These sub-graphs, along with their activated arguments, can then be evaluated using a Dung or rank-based semantics, resulting in a set of accepted and rejected arguments. Finally, the accepted arguments can be aggregated, for example by averaging their conclusions or selecting the conclusion with the highest cardinality, to generate an inference.

2.1. Main features and functionalities

ArgFrame provides the following main features and functionalities::

- User login option to allow multiple users to use the same running instance of the application.
- Definition of categorical or continuous features.
- Creation of features through a JavaScript interface or by importing a JSON string.
- JavaScript interface for the creation of argumentation graphs, with interacting nodes and a hover option to visualise their internal structure.
- JavaScript interface to create arguments using `AND` and `OR` operators, parentheses to group clauses, and the ability to attach a conclusion (target feature).
- SQL database to store users’ information, features, arguments, and graphs. A PHPMyAdmin instance is also provided to manage the database.
- Implementation of the following Dung semantics in PHP: preferred, grounded, eager, ideal, stable, and semi-stable.
- Implementation of the categorizer [8] rank-based semantics in PHP.
- Several options for the aggregation of accepted arguments
- Export option to generate inferences for a whole dataset using any combination of semantics and aggregation strategies.

ArgFrame comes with a predefined set of features and arguments, as employed in [9]. The link for the employed data can also be seen in the “Compute graph” tab. This example can be used for learning and evaluation of the framework.

3. Software impacts and applications

ArgFrame has been developed as a preliminary research tool for the field of defeasible argumentation. It provides an automated option for creating and evaluating argumentation graphs built for producing numerical inferences and for reasoning with quantitative data. This multi-layer, web, argument-based framework has enabled various applications and experiments to be conducted, replicated, and compared.

So far, *ArgFrame* has been used in more than 10 different scholarly publications in different fashions:

- In [9–14], *ArgFrame* has facilitated the quantitative comparison of defeasible argumentation with other techniques for inference in the field of AI, such as fuzzy reasoning and expert systems. By using *ArgFrame*, the authors were able to produce several argument-based models of inference with a diverse number of fine-tuned parameters, such as argumentation semantics, accrual strategy and topology of argumentation graphs. Multiple domains of application were employed, demonstrating a strong generalisability of defeasible argumentation for real-world problems. Without *ArgFrame* automation, demonstrating such a generalisability would demand prohibitive time.
- In [15] the authors have employed *ArgFrame* for a qualitative investigation of the degree of explainability of defeasible argumentation. Through the web interface, *ArgFrame* allows the inferences of models built with it to be better understood. The paper shows how defeasible argumentation can lead to the construction of models of inference with a higher degree of explainability compared to other common techniques of inference in AI, such as fuzzy reasoning.
- In [16,17], *ArgFrame* has facilitated the computational modelling of knowledge bases in the field of health-case. By implementing the arguments and attacks designed by domain experts, it was possible to produce models of inference capable of aiding in the reasoning process of such experts. Since models in *ArgFrame* can be easily modified and evaluated, it also enabled knowledge bases to be improved.
- In [18,19] *ArgFrame* was employed for different knowledge representation problems. The authors implemented several argument-based models for the inference of ill defined constructs in psychology. Results were compared to other techniques of inference from the domains of application, showing that models bult with *ArgFrame* were better able to handle conflicting information and to produce more robust inferences.
- In [20,21], the authors employed a customized version of *ArgFrame* to produce argument-based models of inference aimed at the comparison with other data driven techniques of inference. It was claimed that user studies show that the explainability of argument-based models are statistically similar to the other compared techniques.

In summary, *ArgFrame* has proven to be instrumental in showcasing the benefits of defeasible inference. By automating reasoning tasks over incomplete and inconsistent data, as well as fragmented and vague knowledge, *ArgFrame* has demonstrated its ability to be a useful tool in the domain of defeasible argumentation. It stands out as the only framework known to enable the generation of a large number of quantitative inferences using defeasible argumentation, all within a reasonable time frame and without the need for coding expertise. Moreover, its user-friendly nature has facilitated the execution of automated reasoning activities in scholarly publications for purposes such as knowledge discovery, acquisition, and refinement.

4. Limitations and future research directions

ArgFrame has some limitations that can be overcome through future endeavours:

- The web interface has certain limitations when it comes to evaluating massive data sets. Since all data and semantics are computed locally, it is likely that the browser running the application will be unable to import a massive data set or to perform computations of non-polynomial semantics (such as the preferred semantics) for large graphs. Therefore, *ArgFrame* is more useful for problems of reasonable size where visualisations can assist human reasoners. Data sets of up to 1 GB have been used successfully.
- Currently, *ArgFrame* only supports quantitative and structured arguments. It would be beneficial to have the option to employ abstract argumentation for the analysis of preliminary results or the study of the topology of argumentation graphs.
- There are many other semantics that could be implemented. The ones currently available are those employed in previous studies.

Finally, the framework also has the potential to be extended in the following directions:

- The structure of arguments is currently implemented using boolean logic. Adding other logics or fuzzy options could expand the applicability of the framework to different areas.
- The design and evaluation of argumentation graphs are currently carried out in separate stages. Performing a real-time evaluation of the inferences produced by such graphs during their design process could enhance the reasoning process and facilitate the analysis of different knowledge bases.
- Integrating a machine learning module into the framework for automatic rule extraction could provide a starting point for an argumentation graph. Users could then enhance the graph based on their domain knowledge.
- Enabling the collaboration of multiple users in defining an argumentation graph could allow for richer arguments and attacks to be formulated. The framework could be extended to support real-time interaction with the same graph by multiple users.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

I would like to extend my heartfelt appreciation to Dr. Luca Longo for his collaborative ideas that helped me develop this framework. His expertise, insightful discussions, and constructive feedback have greatly enhanced the quality and impact of this work.

Appendix A. Supplementary data

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.simpa.2023.100547.

References

[1] Daniel Bryant, Paul Krause, A review of current defeasible reasoning implementations, Knowl. Eng. Rev. 23 (3) (2008) 227–260.
[2] Henry Prakken, Gerard Vreeswijk, Logics for defeasible argumentation, in: Handbook of Philosophical Logic, Springer, 2001, pp. 219–318.
[3] Federico Cerutti, Sarah A Gaggl, Matthias Thimm, Johannes Wallner, Foundations of implementations for formal argumentation, IfCoLog J. Log. Appl. 4 (8) (2017) 2623–2705.
[4] CarlosIván Chesñevar, Ana Gabriela Maguitman, María Paula González, Empowering recommendation technologies through argumentation, in: Guillermo Simari, Iyad Rahwan (Eds.), Argumentation in Artificial Intelligence, Springer US, Boston, MA, 2009, pp. 403–422.
[5] Luca Longo, Argumentation for knowledge representation, conflict resolution, defeasible inference and its integration with machine learning, in: Machine Learning for Health Informatics, Springer, 2016, pp. 183–208.
[6] Phan Minh Dung, On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games, Artif. Intell. 77 (2) (1995) 321–358.
[7] Yining Wu, Martin Caminada, Mikotaj Podlaszewski, A labelling-based justification status of arguments, Studia Logica 3 (4) (2010) 12–29.
[8] Philippe Besnard, Anthony Hunter, A logic-based theory of deductive arguments, Artificial Intelligence 128 (1–2) (2001) 203–235.
[9] Lucas Rizzo, Luca Longo, Comparing and extending the use of defeasible argumentation with quantitative data in real-world contexts, Inf. Fusion 89 (2022) 537–566, http://dx.doi.org/10.1016/j.inffus.2022.08.025.

[10] Luca Longo, Lucas Rizzo, Pierpaolo Dondio, Examining the modelling capabilities of defeasible argumentation and non-monotonic fuzzy reasoning, Knowl.-Based Syst. 211 (2021) 106514, http://dx.doi.org/10.1016/j.knosys.2020.106514.

[11] Lucas Rizzo, Evaluating the Impact of Defeasible Argumentation as a Modelling Technique for Reasoning under Uncertainty (Ph.D. thesis), Technological University Dublin, 2020.

[12] Lucas Rizzo, Luca Longo, An empirical evaluation of the inferential capacity of defeasible argumentation, non-monotonic fuzzy reasoning and expert systems, Expert Syst. Appl. 147 (2020) 113220, http://dx.doi.org/10.1016/j.eswa.2020.113220.

[13] Lucas Rizzo, Pierpaolo Dondio, Luca Longo, Exploring the potential of defeasible argumentation for quantitative inferences in real-world contexts: An assessment of computational trust, in: Luca Longo, Lucas Rizzo, Elizabeth Hunter, Arjun Pakrashi (Eds.), Proceedings of the 28th Irish Conference on Artificial Intelligence and Cognitive Science, Dublin, Republic of Ireland, December 7-8, 2020, in: CEUR Workshop Proceedings, vol. 2771, CEUR-WS.org, 2020, pp. 37–48, URL http://ceur-ws.org/Vol-2771/AICS2020_paper_13.pdf.

[14] Lucas Rizzo, Luca Longo, Inferential models of mental workload with defeasible argumentation and non-monotonic fuzzy reasoning: A comparative study, in: 2nd Workshop on Advances in Argumentation in Artificial Intelligence, 2019, pp. 11–26.

[15] Lucas Rizzo, Luca Longo, A qualitative investigation of the degree of explainability of defeasible argumentation and non-monotonic fuzzy reasoning, in: 26th AIAI Irish Conference on Artificial Intelligence and Cognitive Science, 2018, pp. 138–149.

[16] Lucas Rizzo, Ljiljana Majnaric, Luca Longo, A comparative study of defeasible argumentation and non-monotonic fuzzy reasoning for elderly survival prediction using biomarkers, in: Chiara Ghidini, Bernardo Magnini, Andrea Passerini, Paolo Traverso (Eds.), AI*IA 2018 – Advances in Artificial Intelligence, Springer International Publishing, Cham, 2018, pp. 197–209.

[17] Lucas Rizzo, Ljiljana Majnaric, Pierpaolo Dondio, Luca Longo, An investigation of argumentation theory for the prediction of survival in elderly using biomarkers, in: Lazaros Iliadis, Ilias Maglogiannis, Vassilis Plagianakos (Eds.), Artificial Intelligence Applications and Innovations, Springer International Publishing, Cham, 2018, pp. 385–397.

[18] Lucas Rizzo, Luca Longo, Representing and inferring mental workload via defeasible reasoning: a comparison with the NASA task load index and the workload profile, in: 1st Workshop on Advances in Argumentation in Artificial Intelligence, 2017, pp. 126–140.

[19] Ryan Kirwan, Comparing Defeasible Argumentation and Non-Monotonic Fuzzy Reasoning Methods for a Computational Trust Problem with Wikipedia (Ph.D. thesis), Technological University Dublin, 2019.

[20] Giulia Vilone, Luca Longo, A novel human-centred evaluation approach and an argument-based method for explainable artificial intelligence, in: Artificial Intelligence Applications and Innovations: 18th IFIP WG 12.5 International Conference, AIAI 2022, Hersonissos, Crete, Greece, June 17–20, 2022, Proceedings, Part I, Springer, 2022, pp. 447–460.

[21] Giulia Vilone, Luca Longo, A global model-agnostic XAI method for the automatic formation of an abstract argumentation framework and its objective evaluation, 2022.