The Long and Winding Road: Forty Years of Argumentation

Trevor BENCH-CAPON
Department of Computer Science, University of Liverpool, UK

Abstract. In this paper I review my engagement with argumentation over the past forty years. I describe the perspective I brought from philosophy and the Civil Service, and consider a number of aspects of computational argumentation: knowledge based systems, explanation, context, audiences, schemes and models. A key feature of argumentation is that it is an activity which has to be actively engaged with, whereas a proof is an object to be understood and admired.

Keywords. argumentation, explanation, justification

1. Introduction

As a student I studied Philosophy. Thus while my mathematician wife became familiar with proofs and theorems, I encountered only arguments. The study of modern epistemology for, example, begins with Descartes *Argument from Illusion*, and takes us through Kant's *Transcendental Argument* to Wittgenstein's *Private Language Argument*. Similarly philosophical theology, the topic of my PhD, concerns arguments: the existence of God is discussed through the *Ontological Argument*, the *Cosmological Argument* and the *Teleological Argument*: to see these arguments as intended to be proofs is to make a significant blunder, as I argued in [14]. Whereas proofs are passive, things to be understood, arguments are things that must be engaged with, accepted, adopted, *bought*, as we used to say. A proof is complete in itself, an argument only becomes complete when an audience accepts it. Wittgenstein said that the purpose of philosophy was to show the fly the way out of the fly bottle. Not to remove the fly, or to break the bottle, but to *show the way*. To escape the fly must take the route for itself. So too, an argument has an effect only when it is used by its audience. Thus the Argument from Illusion can ensnare us, but the Transcendental Argument shows us how we can escape from scepticism, and the Private Language Argument can rescue us from solipsism if we let it.

Having completed my PhD, I went to work as a Civil Servant, as a trainee policy maker. In those days policy making was thought to be a rational activity and so civil servants would prepare sets of arguments, both for and against various policy proposals, which the Minister would consider and choose between. Of course these arguments were not always about questions of fact: there were political arguments and arguments designed to appeal to various interest groups as well. The decision was always made by the Minister, and would, properly, reflect the aspirations and interests of the party he or she represented. Moreover the argument that convinced the Minister, would not always be the argument the Minister used to sell the policy to the Public. This gave more useful lessons in practical argumentation, and in the crucial role of the audience and its preferences. For a variety of reasons I left work on policy and moved into computing, first as a programmer analyst and then looking at the potential for using knowledge based systems in Government. And this in turn took me back to academia, and Imperial College.

2. Knowledge Based Systems

At Imperial College the Logic Programming Group conceived of knowledge based systems as sets of *axioms* from which consequences could be *proved*. With my background they appeared somewhat differently. Essentially we had a set of heuristics gathered from an expert, and these heuristics would provide reasons to believe certain conclusions. The whole enterprise was thus based on a particular style of argument, namely Argument from Expert Opinion. While conclusions could be justified in terms of the rules in the program, the rules themselves could only be justified by the quality and authority of the expert. The use of Negation as Failure made relevant another form of argumentation, Argument from Ignorance, which when used improperly gives rise to the fallacy *argumentum ad ignorantiam*. The conditions for its proper use can be given a logical justification by completing the database, but the necessary Closed World Assumption was sometimes inappropriate for particular systems where it was, none the less, used. Moreover it is a feature of logic programs that they can generate justifications for propositions *and* their contraries. In argumentation terms this is a good thing - the program can be seen as an generating arguments both for and against propositions. So my picture of a legal knowledge based system was of a program to generate arguments for and against some claim, among which it was up to the users to choose what they believed. The lack of prescription and the responsibility of the audience were thus both respected. This view was expressed in [10], which suggested that what was needed for an intelligent system would be “a representation in computer intelligible terms of what it is that makes an argument persuasive”, reasons why an argument should be accepted or rejected by a given adjudicator. Generating the arguments was relatively straightforward: supporting the choice between them was where the challenge lay.

3. Explanation

The importance of the user choosing between the pro and con arguments generated by the program, meant that explanation of the reasoning - the provision of the arguments - moves from a nice additional feature to the core of knowledge based systems. But the state of the art in explanation in 1990 had barely moved on from MYCIN: the question *how?* posed of a conclusion of the system would elicit the rules and facts used in its derivation. Moving from proof to argument meant moving the user from a *passive* consumer of proofs to a *proactive* participant in an argument, and this meant engaging in a dialogue. The basis for such dialogues was available in the logical dialogue games of Mackenzie [19] and Hamblin [17][1]. These, however, were games based on natural de-

[1] I am grateful to David Moore for introducing me to this work and its potential for application to explanation of KBS.

duction proofs and the resulting dialogues [6] were consequently rather stilted. More natural dialogues were produced [7] by basing the dialogue of the Argumentation pattern[2] identified by Stephen Toulmin [28]. Investigation of Toulmin further led to the insight that the various clauses in the body of a Prolog rule played very different roles. Consider

```
old(X):-man(X), age(X,A), A > 75,
        not has_drunk(X,elixir_of_youth).
```

Here the first clause establishes the *sort* of thing X is: men, dogs and houses all become old at different ages, and so require different rules. The second clause simply retrieves the age of X: it is not expected to fail. The third clause, is interesting not because it tests whether the number to which A has been instantiated is greater than 75 (its apparent purpose), but because it provides the key piece of information, namely the age at which a man becomes old. And the fourth clause provides an exception, since drinking the Elixir of Youth means that a person never grows old[3]. Recognition that the roles of the premises of the argument are not homogeneous is the key motivation for the use of argumentation schemes, which is now common place in computational argumentation. In this case, by annotating the clauses with these four roles, a Prolog meta interpreter could be used to produce the output from a logic program as a set of relations describing a set of linked Toulmin structures [9]. These structures could then be navigated using a set of performatives to request movement to different elements in the structure, (e.g. *why?* elicits the data, *presupposing?* solicits the sortal) giving rise to reasonably natural dialogues [8], in which users can establish what is needed for them to draw the conclusion, given their particular current background knowledge, and to explore the reasons for exceptions and counterarguments.

4. Context

Perhaps the most significant development in computational argumentation in this period was the introduction of abstract Argumentation Frameworks (AF) by Dung [15]. The important point here is that we always talk about *sets* of arguments, so that the acceptability of an argument is always dependent on the context formed by the other arguments in the framework. In an AF arguments are entirely abstract, related only by an attack relation. To be acceptable, an argument must either not be attacked, or in a position to defeat all its attackers. Attackers can be defeated by the attacked argument being a member of set which contains arguments that defeat them. A set in which none of the members attack one another and which contains at least one argument able to defeat any attacker of a member of the set is termed *admissible*, and maximal admissible sets *preferred extensions*. Acceptability can either be *credulous* if an argument is in at least one preferred extension, or *sceptical* if an argument is in all preferred extensions. As well as preferred semantics [15] discussed grounded and stable semantics, and since then many other different semantics for acceptability have been developed. The key point, however, is not

[2]Today we might term this a "model" or an "argumentation scheme", but "pattern" was used by Toulmin to introduce it.

[3]In contrast, the *struldbrugs* in Swift's *Gulliver's Travels* are immortal, but age normally and so are forever old, suffering all the increasing indignities and infirmities of extreme old age. A struldbrug is deemed legally dead at the age of eighty, so that wealth can be inherited, and so they are forever impoverished also.

any particular flavour of semantics, but that the acceptability of an argument depends on what *other* arguments there are in the context (both the set of arguments in the AF and the subsets which form admissible sets): this determines both the attackers that need to be defeated and the arguments which are available to defeat them. Acceptability is not a property inherent in an argument, but one which is only possessed with reference to a particular context. Computationally AFs can be readily linked to knowledge based systems and logic programs: the arguments and attacks can be generated from the program, and then evaluated in an AF.

5. Audiences and Their Preferences

The problem, however, with Dung's framework, was that it was coercive - if an attacker could not be defeated then the attacked argument fell. But in many contexts, such as law and politics, it may be necessary to choose between conflicting arguments, even when neither can be defeated. In other words, if it is acknowledged that different arguments have different strengths, it is important to be able to distinguish attack from *successful* attack, from *defeat*. A mechanism to make this distinction was proposed in [1], but that proposal simply assumed a preference relation between arguments, without any mechanisms to apply the preference systematically throughout the framework, explain the preference or argue about the preferences.

In order to address these aspects, inspiration was drawn from the work of Perelman[4] on audiences [23]. The idea is that an audience can be characterised by its ordering on values: a Greek hero like Achilles will prefer the value *Fame* to the value *Long Life*, whereas the reverse will be true of a twenty first century Health and Safety Official. King Solomon chose *Wisdom* rather than *Wealth*, or any other kingly attribute. Characterising audiences in terms of their ordering of values was made the basis of *Value-based Argumentation Frameworks* [11]. If we now relate arguments to a set of values, as in [12] for example which used the argumentation scheme for practical reasoning devised by Atkinson and her colleagues [4], we can now apply the preferences according to the ordering of values for a particular audience to determine the relative strength of arguments, and hence which attacks should succeed as defeats for that audience.

The approach was further enhanced by the development of *Extended Argumentation Frameworks* (EAF) by Sanjay Modgil [22][5]. In an EAF, preferences are expressed by allowing arguments to attack attacks, as well as other arguments. In this way preferences can themselves be the subject of argument. Now we can combine a set of arguments with a set of reasons why those arguments should be found persuasive, and so make our theory of persuasiveness explicit. EAFs can subsume both preference and value-based frameworks [21]. The meta level description of such an EAF forms a standard Dung-style AF and so we can apply the many results relating to standard AFs to arguments with strengths and audience specific preferences.

---

[4]I am grateful to Floriana Grasso for introducing me to the work of Perelman, and its importance for computational argumentation.

[5]Related ideas were explored earlier by Verheij (e.g [29]) and have been further developed in [5], which permits attacks on attacks on attacks, and so on.

6. Schemes and Models

For me, the most interesting question currently in computational argumentation is how precisely argumentation schemes can be exploited. The problem is this: one of the key features of argumentation schemes is that they have a number of different premises, and these premises are intended to play different roles in composing the argument, and in consequence the premises have different forms. But the logical study of schemes pulls us in a different direction: for such purposes if we can regard all schemes as defeasible rules, and the different premises as different antecedents in those rules, so that the whole structure becomes a cascade of arguments and sub-arguments, we have a nice structure, amenable to analysis, and a simple homogeneous notion of premises. But in doing this something is lost: the process is analogous to moving from model based reasoning from first principles [26], to a rule based system intended to package such reasoning for easy delivery, but not to support the reasoning activity itself, and it is important to remember that it *engagement in the activity* that is central to argumentation. The contrast is illustrated by work such as KARDIO [18] in which a model of the heart is used to generate a set of diagnostic rules which can be used to build a conventional KBS to identify heart problems. Importantly also, once schemes are homogenised in this way a lot of the importance of critical questions in identifying the characteristic ways in which arguments using different schemes can be challenged is also lost. Attacks are reduced to premise defeat, rebuttal and undercut (cf [30]), and again this is convenient for analysis, but at the expense of the diversity and richness of argumentation as we practice it.

I feel that if argumentation schemes are to be exploited fully it is important that their heterogeneous nature be somehow reflected in the knowledge representation. An example of this for the practical reasoning scheme of Atkinson and her colleagues can be found in [2], in which the scheme and its characteristic critical questions were related to models represented as Alternating Action Based Transition Systems. The use of models to support richer reasoning was further described in [3]. The benefits of doing so, in terms of the dialogues which can result, is reported in [31]. I believe that this may open up a whole new avenue for research into schemes, with much of the emphasis put back on *content* rather than *form*.

7. Modelling Real Arguments

When studying argumentation - even abstract argumentation - it is important to remember that arguments are always about something. Mathematics is not about the real world: it doesn't matter that we can never find *minus six pebbles* on the beach, once we have moved into the realm of mathematics such notions are fine. But arguments do not inhabit a Platonic realm: they need to be out there, in the midst of us, persuading and motivating people in the world. Losing sight of this can be dangerous. Content is important in real argumentation: over focussing on form is one way in which flies get into fly bottles.

Suppose we are told that residents of Liverpool are mostly poor, except for those who live in Woolton, which is where many Liverpool FC footballers live. Woolton is a relatively small district of Liverpool, so we might be tempted to write a strict rule and a defeasible rule:

If X lives in Woolton, X lives in Liverpool: $W \rightarrow L$

If X lives in Liverpool X is, defeasibly, poor: $W \Rightarrow P$

These rules could serve us very well as part of a larger system with many ways of establishing L, and where we use L for a range of purposes. The problems only come when we want to use both together. If we know that W is true, we might think we have an argument for P. But now the very reason for believing P is also an undercutter of the rule we use to derive P. W was used to derive the antecedent of the rule used to conclude P, but W is itself an exception to this rule. Stated as *Pepe Reina lives in Woolton so he is poor*, we can immediately see the problem: the compressed argument is the opposite of the information we were originally given. But once we start using symbols we can lose sight of this. What we must do when we fix this problem is to make sure that there is no argument from W to P at all. It is not enough that there is an argument which is defeated: there is no such argument. Living in Woolton provides no reason whatsoever to suppose that someone is poor. Clearly our defeasible rule should have been:

$L \wedge \neg W \Rightarrow P$

Two lessons: always use a couple of examples as a sanity check when reasoning with abstract symbols; and if there is a problem look to how the content has been represented before inventing any formal machinery.

Over the last thirty years I have tried to apply AI techniques to problems related to the law, and this has, of course, included modelling legal argumentation. One of the very first AI and law projects was Thorne McCarty's attempt to model the opposing arguments in the US tax law case of *Eisner v Macomber* e.g. [20]. But it is worth remembering that these arguments are in fact opinions of Supreme Court Justices: Pitney's is some 6,500 words and Brandeis' is some 5,500 words. I mention this to stress that a legal argument is often a far cry from abstract argumentation, or even the examples used to illustrate structured argumentation, which typically use no more than half a dozen rules, each with no more than two or three antecedents. The same, of course, applies to philosophical argument: Anselm's statement of the Ontological Argument in the *Proslogium* is only 323 words, but that is universally recognised to be a very succinct statement of the argument: over succinct since it has left several parts in need of elaboration.

In Computational Argumentation we find many tools for supporting the analysis of argument, such as Arucaria [25] and Cohere [27], with little support for computation, and many tools which generate and evaluate arguments where the structure appears rather simplified and idealised when set against the real examples from the Supreme Court or the philosophical literature. Both are useful, both are interesting and both have their place. But we need to remember that they are supposed to be connected. AI and Law has a tradition of work which looks at the complexity of real opinions from the perspective of computational models, and this has produced a lot of stimulating work of which the argumentation community should be aware, and which might be usefully imitated in other domains. Cases studied include *Eisner v Macomber*, *Carney v California* and related cases, the wild animals cases starting with *Pierson and Post* and ending with *Popov v Hayashi*. Recently the decision in *Popov v Hayashi*, which runs to something over 5,500 words, was the subject of a special issue of *AI and Law* journal in which the argumentation (or aspects of it) was modelled using dimensions and factors [13], as a Dung style argumentation framework with arguments generated using ASPIC+ [24] and using the Carneades framework [16]. This offers an instructive opportunity to compare techniques developed for computational argumentation applied to a real, substantial set of arguments in a context which permits a direct comparison between them.

8. Conclusion

I have described part my journey (omitting some digressions into topics such as software engineering of KBS and ontology). The route has been winding and sometimes circular, but I feel I have made some progress. There is, however, a very long way still to go.

Logic provides an idealisation of reasoning. But most interesting reasoning problems contain one or more of the following features:

- There are reasons for believing both sides of the question
- The conclusion is relative to the context in which it is made
- The conclusion is relative to the audience which makes it
- The conclusion is chosen, not simply recognised
- The topic relates not simply to truth, but to values, aspirations and interests

Of course we can, and in AI we often do, abstract away from these elements. But for true intelligence we need to embrace them, and for this we need to *engage* in *argument*, not admire proofs.

Acknowledgements

I am very grateful to all those with whom I have discussed and written about argumentation over the past 40 years. These include: from my student days, Peter Hacker, Tony Kenny and Toomas Karmo; from my time at the DHSS, Bernard Cawley; from my time at Imperial, Marek Sergot and Tom Routen; various colleagues at Liverpool, especially Paul Dunne, Paul Leng, Katie Atkinson, Frans Coenen, Adam Wyner, Geof Staniford, Maya Wardeh, Alison Chorley, Fahd Nawwab, Rolando Medellin-Gasque, Duncan Lowes, Tony McEnery, John Henderson and Will Hounslow; other colleagues, especially Henry Prakken, Tom Gordon, Sanjay Modgil, Giovanni Sartor, Bart Verheij, Edwina Rissland, Sylvie Doutre, Floris Bex, and Doug Walton; and the most argumentative of my children, Michael. Particular thanks, as noted above, to David Moore and Floriana Grasso. Apologies to those I have missed out: there were, of course, arguments for including so many more of you.

References

[1] Leila Amgoud and Claudette Cayrol. Inferring from inconsistency in preference-based argumentation frameworks. *J. Autom. Reasoning*, 29(2):125–169, 2002.

[2] Katie Atkinson and Trevor J. M. Bench-Capon. Practical reasoning as presumptive argumentation using action based alternating transition systems. *Artif. Intell.*, 171(10-15):855–874, 2007.

[3] Katie Atkinson, Trevor J. M. Bench-Capon, Dan Cartwright, and Adam Zachary Wyner. Semantic models for policy deliberation. In *The 13th International Conference on Artificial Intelligence and Law, Proceedings of the Conference*, pages 81–90, 2011.

[4] Katie Atkinson, Trevor J. M. Bench-Capon, and Peter McBurney. Computational representation of practical argument. *Synthese*, 152(2):157–206, 2006.

[5] Pietro Baroni, Federico Cerutti, Massimiliano Giacomin, and Giovanni Guida. Afra: Argumentation framework with recursive attacks. *Int. J. Approx. Reasoning*, 52(1):19–37, 2011.

[6] TJM Bench-Capon, PE Dunne, and PH Leng. Interacting with knowledge-based systems through dialogue games. *Proceedings of the Eleventh International Conference on Expert Systems and Applications*, pages 123–140, 1991.

[7] TJM Bench-Capon, PE Dunne, and PH Leng. A dialogue game for dialectical interaction with expert systems. *Proc. 12th Annual Conf. Expert Systems and their Applications*, pages 105–113, 1992.

[8] T.J.M. Bench-Capon, T. Geldard, and P.H. Leng. A method for the computational modelling of dialectical argument with dialogue games. *Artificial Intelligence and Law*, 8:233–254, 2000.

[9] T.J.M. Bench-Capon, D. Lowes, and AM McEnery. Argument-based explanation of logic programs. *Knowledge-Based Systems*, 4(3):177–183, 1991.

[10] Trevor Bench-Capon and Marek Sergot. Toward a rule-based representation of open texture in law. In Charles Walter, editor, *Computer Power and Legal Language*, chapter 6, pages 39–60. Quorum, New York, 1988. From the Second Annual Conference on Law and Technology, University of Houston, 24–28 June 1985.

[11] Trevor J. M. Bench-Capon. Persuasion in practical argument using value-based argumentation frameworks. *J. Log. Comput.*, 13(3):429–448, 2003.

[12] Trevor J. M. Bench-Capon, Katie Atkinson, and Alison Chorley. Persuasion and value in legal argument. *J. Log. Comput.*, 15(6):1075–1097, 2005.

[13] Trevor J. M. Bench-Capon, Katie Atkinson, and Peter McBurney. Using argumentation to model agent decision making in economic experiments. *Autonomous Agents and Multi-Agent Systems*, 25(1):183–208, 2012.

[14] Trevor J.M. Bench-Capon. Reinterpreting the proofs of the existence of god. *Religious Studies*, 16(03):299–306, 1980.

[15] 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):321–358, 1995.

[16] Thomas F. Gordon and Douglas Walton. A carneades reconstruction of popov v hayashi. *Artif. Intell. Law*, 20(1):37–56, 2012.

[17] C. L. Hamblin. *Fallacies*. Methuen, London, 1970.

[18] Nada Lavrac, Ivan Bratko, Igor Mozetic, Boian Cercek, Anton Grad, and Matija Horvat. Kardio-e: an expert system for ecg diagnosis of cardiac arrhythmias. *Expert Systems*, 2(1):46–55, 1985.

[19] J.D. Mackenzie. Question-begging in non-cumulative systems. *Journal of Philosophical Logic*, 8:117–1133, 1979.

[20] L. Thorne McCarty. An implementation of eisner v. macomber. In *Proceedings of the Fifth International Conference on AI and Law*, pages 276–286, 1995.

[21] S. Modgil and T.J.M. Bench-Capon. Metalevel argumentation. *Journal of Logic and Computation*, 21(6):959–1003, 2011.

[22] Sanjay Modgil. Reasoning about preferences in argumentation frameworks. *Artif. Intell.*, 173(9-10):901–934, 2009.

[23] Chaim Perelman and L. Olbrechts-Tyteca. *The New Rhetoric: A Treatise on Argumentation*. University of Notre Dame Press, June 1969.

[24] Henry Prakken. Reconstructing popov v. hayashi in a framework for argumentation with structured arguments and dungean semantics. *Artif. Intell. Law*, 20(1):57–82, 2012.

[25] Chris Reed and Glenn Rowe. Araucaria: Software for argument analysis, diagramming and representation. *International Journal on Artificial Intelligence Tools*, 13(4):983–, 2004.

[26] Raymond Reiter. A theory of diagnosis from first principles. *Artif. Intell.*, 32(1):57–95, 1987.

[27] Simon Buckingham Shum. Cohere: Towards web 2.0 argumentation. In Philippe Besnard, Sylvie Doutre, and Anthony Hunter, editors, *COMMA*, volume 172 of *Frontiers in Artificial Intelligence and Applications*, pages 97–108. IOS Press, 2008.

[28] Stephen Toulmin. *The Uses of Argument*. Cambridge University Press, Cambridge, England, 1958.

[29] Bart Verheij. Deflog: on the logical interpretation of prima facie justified assumptions. *J. Log. Comput.*, 13(3):319–346, 2003.

[30] Bart Verheij. Dialectical argumentation with argumentation schemes: An approach to legal logic. *Artif. Intell. Law*, 11(2-3):167–195, 2003.

[31] Adam Wyner, Katie Atkinson, and Trevor Bench-Capon. Model based critique of policy proposals. In *Proceedings of ePart 2012*, In Press, 2012.