Defeasible AceRules: A Prototype

Martin Diller Adam Wyner* Hannes Strass

Vienna University of Technology *University of Aberdeen
Leipzig University

Contracts and Computation Workshop
University of Gothenburg
02.11.2017
http://remu.grammaticalframework.org/contracts/workshop/

Motivation

The argument-mining - formal argumentation gap:

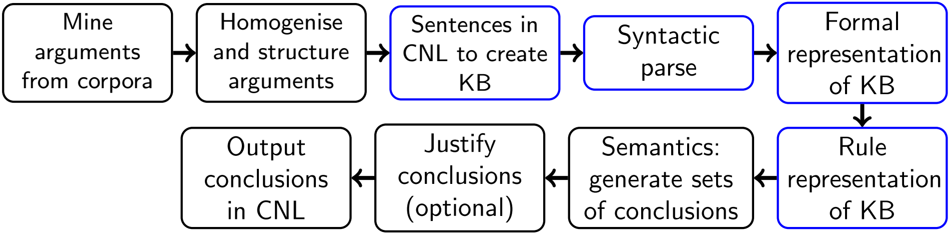
- Argument mining: attempts to extract arguments from large textual corpora and structure them (Lippi and Torroni, 2016).
  - Drawback: lacking fine-grained structured representations that can be used for inference.
- Formal argumentation: has focused on giving dialectical characterisations of reasoning in defeasible knowledge bases (Dung, 1995; Prakken, 2010, ...).
  - Drawback: abstraction from linguistic information limits applicability.

Contracts: has focused on finding conflicts, but not necessarily reasoning further with them, though conflicts in and between contracts are widespread. Need to treat conflict to reason further. Goal to identify alternative sets of compatible clauses and draw inferences.

Motivation (cont.)

A long-term vision for solving the argument-mining - formal-argumentation gap:

- A controlled natural language (CNL) as a middle-layer between natural-language and formal-argumentation.
    - A *CNL* is a subset of a natural language, restricted in lexicon, grammar; can have a fixed semantic representation; can translate to executable rules.
    - Eliminates ambiguity and reduces complexity of unrestricted NL.



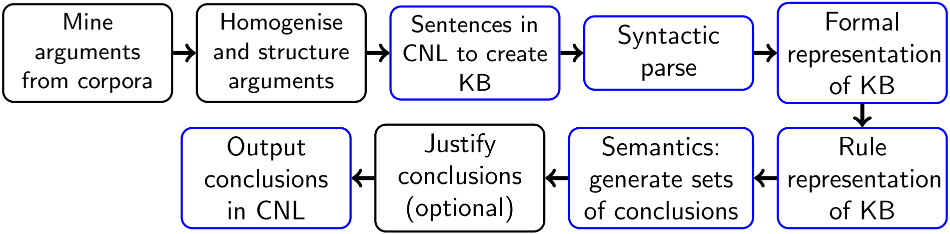
Motivation (cont.)

This work:

- A first experiment in using a CNL as a natural-language interface to formal-argumentation.
- We extend an existing controlled natural language, Attempto Controlled English (ACE), with means for expressing generic generalisations ("it is usual that..."). **Need to comment about interpretations of 'usual' and how it is used. We have taken it as given, but it might not be. When is the conflict between 'it is usual that P' and 'it is usual that not P'.**
- Building on tools for ACE, we develop a prototype reasoner for defeasible rules expressed in natural language.
- We employ a novel argumentation-inspired semantics.
  - Allows for transparent, linguistically accessible reasoning with incomplete/incremental and inconsistent knowledge bases.
  - Circumvents problems in more standard-approaches to reasoning about defeasible KBs using argumentation.

Motivation (cont.)

This work:



Motivation

- Central topics in the study of argumentation:
  - Identification and extraction of arguments in natural langauge.
  - Evaluation of the cogency of arguments.
- Roughly, the topic of argument-mining and formal models (structured or graph-based) of argumentation.
- Substantial gap between advances in both areas limits applicability of formal models and makes only “shallow” forms of inference amenable to argument-mining.

Motivation (cont.)

- Controlled natural languages (CNLs): purposefully selected sub-sets of a natural language that can be extended and its formal semantics corrected in light of theoretical and empirical studies.
- CNLs can serve as modifiable links in an engineering approach to close the gap between informal and formal argumentation.
- Crucial added benefit: connect developments in computational linguistics (e.g. mapping syntax to semantics, anaphora resolution, presupositions, dynamics,...) with related work in computational models of argumentation.

Motivation (cont.)

- Our work: first steps towards a CNL-interface to argumentation.
- A prototype for argumentation-based evaluation of CNL-knowledge bases consisting of strict and defeasible rules.
- We extend an existing CNL, ACE, with a linguistic marker to indicate defeasibility ("it is usual that...").
- We build on an existing reasoner for a subset-of ACE, AceRules.
- We link the output of a parser of ACE, APE, with answer-set-programming encodings of an argumentation-based semantics for knowledge bases consisting of strict and defeasible rules.

Background to this work

- Wyner, Bench-Capon, and Dunne. On the instantiation of knowledge bases in abstract argumentation frameworks. CLIMA 2013: 34-50.
- Strass. Instantiating Knowledge Bases in Abstract Dialectical Frameworks. CLIMA 2013: 86-101
- Wyner, Bench-Capon, Dunne, and Cerutti. Senses of 'argument' in instantiated argumentation frameworks. Argument & Computation, 6(1):50-72, 2015.
- Strass and Wyner, On automated defeasible reasoning with controlled natural language and argumentation, in Proceedings of the Second International Workshop on Knowledge-based Techniques for Problem Solving and Reasoning (KnowProS), Feb. 2017.
- Wyner and Strass: dARe - Using Argumentation to Explain Conclusions from a Controlled Natural Language Knowledge Base. IEA/AIE (2) 2017: 328-338.
- Diller, Wyner, Strass. Defeasible AceRules: A Prototype. International Conference on Computational Semantics (IWCS). 2017. Accepted.

1. Introduction
   - Motivation
   - Background to this work
   - Outline
2. Extending ACE
   - ACE
   - AceRules
   - Adding generic generalisations to AceRules
   - Alternatives
3. Direct-stable semantics
   - Motivation
   - Definition
4. Our prototype
   - Architecture
   - Description
5. An extended example
6. Conclusions

ACE: Attempto Controlled English

attempto.ifi.uzh.ch

- CNL for the English language developed at University of Zurich (Fuchs et al, 2008).
- Vocabulary comprises predifined function words (e.g. determiners, conjunctions, prepositions), predefined phrases (there is / are, it is false that ...), and an extendable set of content-words (nouns, verbs, adjectives, adverbs).
- Grammar supports (among others): quantification, negation, logical connectives, modality, active & passive voice, singular & plural, relative clauses , etc.

ACE: Attempto Controlled English (cont.)

attempto.ifi.uzh.ch

- Semantics given in terms of discourse representation structures (DRSes): account for linguistic phenomena as anaphora, tense and, more generally, presuppositions. In ACE only anaphora resolution is supported.
  - DRSes are constructed dynamically (anaphora resolution).
  - Complete DRSes (all co-references are resolved) have a model-theoretic semantics and can be translated to FOL.
  - Reasoning (more or less) with FOL expressed in natural language input and output.

- Many tools available for ACE, including the open-source parser APE.
- Also constructs DRSes, offers translations from DRSes to other languages (e.g. FOL, OWL, ...), and does paraphrasing.

AceRules

- ACE-based interface to formal rule systems (Kuhn, 2007).
- Support for logic programs under the stable (Gelfond and Lifschitz, 1990) and courteous (Grosof, 1997) semantics.
- Strict negation ("John is not a customer", "nobody knows John", ..) and negation as failure ("A customer is not provably trusworthy", "it is not provable that John has a card").
- Checks whether DRSes generated from input text by APE conform to the required rule language.
- Transforms DRSes in some cases in which the DRS does not conform syntactically, but can be made to conform ("intelligent grouping").
- Relies on external solvers for the stable semantics; native implementation of the courteous semantics.

Generics in ACE/AceRules vs. our treatment

- Example from (Kuhn, 2007).

Strict rules in ACE/AceRules:

John owns a car.
Bill does not own a car.
If someone does not own a car then he/she owns a house.

Generics in ACE/AceRules vs. our treatment

Generic generalisations in ACE/AceRules:

John owns a car.
Bill does not own a car.
If someone does not own a car and it is not provable that he/she does not own a house then he/she owns a house.

Generics in ACE/AceRules vs. our treatment

Our treatment of generic generalisations:

John owns a car.
Bill does not own a car.
If someone does not own a car then it is usual that he/she owns a house.

Example (ACE)

Problem with ‘it is usual that’ expressed as ’not provably not’

Example (Moustache Murder; paraphrased from Pollock, 2007)

```
Jones is a person.
Paul is a person.
Jacob is a person.
If X is a person and it is not provable that X is not
reliable then X is reliable.
If Jones is reliable then the gunman has a moustache.
If Paul is reliable then Jones is not reliable.
If Jacob is reliable then Jones is reliable.
```

Observation: Clearly not both Paul and Jacob can be reliable, and any semantics should be able to provide a choice of options.

Example (ACE)

```
1 person(jones).  person(paul).  person(jacob).
2 has(gunman,moustache) :- reliable(jones).
3 -reliable(jones) :- reliable(paul).
4 reliable(jones) :- reliable(jacob).
5 reliable(X) :- person(X), not -reliable(X).
```

The literal `-reliable(jacob)` cannot ever be derived from the program, so `reliable(jacob)` must be in every answer set by (5) and (1); similarly for `reliable(paul)`. `reliable(jones)` must be in every answer set by (4) and `reliable(jacob)`. `-reliable(jones)` must be in every answer set by (3) and `reliable(paul)`. Consequently, any answer set would have to contain both `reliable(jones)` and `-reliable(jones)`. This is inconsistent, and there is no set.

Why not....

- negation-as-failure, circumscription, default logic, abstract dialectical frameworks, preferences, ASPIC+/logic-based, and so others
- we are conservatively close to what is explicitly used in natural language versus:
  - two negations in natural language - classical and negation-as-failure
  - ab predicates (where do they come from?)
  - defeaters (where do they come from and they have a special status)
  - conditions on arcs (ad hoc? unconstrained?)
  - additional constraints (linguistic evidence?)
  - precompile "arguments" (blow-up, opaque, requires complete arguments)

Motivation behind the direct-stable semantics

Motivations behind direct-stable semantics (Strass and Wyner, 2017):

- Time-honored interpretation of strict rules as holding in all possible worlds, defeasible rules in all non-exceptional possible worlds (Poole, 1988), ...
- All benefits of argumentation (justification, defeasible reasoning, ...), while avoiding pitfalls in "standard" approaches to reasoning on defeasible KBs using "structured" argumentation:
  - Rationality postulates (Caminada and Amgoud, 2007) often violated,
  - exponential-overgeneration of arguments,
  - opacity of attacks,
  - ...
- Main difference: arguments not directly computed upon, but can be generated on demand for justification.

Defeasible Theories

- Direct-stable semantics for general defeasible theories (with first-order features) derived from definition for propositional defeasible theories.

Defeasible theories: propositional case

- Basis: set $\mathcal{P}$ of propositional variables
- Literals $L := \mathcal{P} \cup \{\neg p \mid p \in \mathcal{P}\}$
- strict rules: $b_1, \ldots, b_m \rightarrow h$; also (B, h) $(B = \{b_1, \ldots, b_m\})$
- defeasible rules: $b_1, \ldots, b_m \Rightarrow h$; also (B, h).
- A defeasible theory is a triple $\mathcal{T} = (\mathcal{P}, \mathcal{S}, \mathcal{D})$ consisting of literals, strict rules, and defeasible rules.

Direct Semantics: Possible Sets

Sets of consistent conclusions

Definition (Possible Sets)

Let $\mathcal{T} = (\mathcal{P}, \mathcal{S}, \mathcal{D})$ be a defeasible theory.
A set $M \subseteq \mathcal{L}_{\mathcal{P}}$ of literals is a *possible set* for $\mathcal{T}$ if and only if there exists a set $\mathcal{D}_M \subseteq \mathcal{D}$ such that:

1. M is consistent;
   - If $p \in M$ then $\neg p \notin M$ and viceversa
2. M is closed under $\mathcal{S} \cup \mathcal{D}_M$;
3. $\mathcal{D}_M$ is $\subseteq$-maximal with respect to items 1 and 2.

- $\mathcal{D}_M$ are the defeasible rules that hold in M.

Small Example

Example

Defeasible theory $\mathcal{T} = (\{a, b\}, \emptyset, \{a \Rightarrow b, \quad b \Rightarrow a\})$ has seven possible sets:

- $M_1 = \emptyset$,
- $M_2 = \{\neg a\}$,
- $M_3 = \{\neg b\}$,
- $M_4 = \{\neg a, \neg b\}$,
- $M_5 = \{a, \neg b\}$,
- $M_6 = \{\neg a, b\}$,
- $M_7 = \{a, b\}$.

Towards Explanations and Arguments

Justifying conclusions

Definition (Derivation)

Let $\mathcal{T} = (\mathcal{P}, \mathcal{S}, \mathcal{D})$ be a defeasible theory.
A *derivation in $\mathcal{T}$* (for z) is a set $R \subseteq \mathcal{S} \cup \mathcal{D}$ of rules with a partial order $\preccurlyeq$ on R such that:

1. $\preccurlyeq$ has a greatest element $(B_z, z) \in R$;
2. for each rule $(B, h) \in R$, we have: for each $y \in B$, there is a rule $(B_y, y) \in R$ with $(B_y, y) \prec (B, h)$ (where $\prec$ is the strict partial order contained in $\preccurlyeq$);
3. R is $\subseteq$-minimal with respect to items 1 and 2.

Small Example

Example

Defeasible theory $\mathcal{T} = (\{a, b\}, \emptyset, \{a \Rightarrow b, \quad b \Rightarrow a\})$ has no derivations. (Thus no justifiable conclusions.)

Example

Defeasible theory $\mathcal{T} = (\{a, b\}, \{\rightarrow a\}, \{a \Rightarrow b, \quad b \Rightarrow a\})$ has two derivations:

- $\rightarrow a$ is a derivation for a
- $\rightarrow a \quad \preccurlyeq \quad a \Rightarrow b$ is a derivation for b
- $\rightarrow a \quad \preccurlyeq \quad a \Rightarrow b \quad \preccurlyeq \quad b \Rightarrow a$ is not a derivation for a (since $\rightarrow a$ already is)

Direct Semantics: Stable Sets

Sets of justified conclusions

Definition (Stable Set)

Let $\mathcal{T} = (\mathcal{P}, \mathcal{S}, \mathcal{D})$ be a defeasible theory and $M \subseteq \mathcal{L}_{\mathcal{P}}$ be a possible set for $\mathcal{T}$. M is a *stable set for $\mathcal{T}$* iff for every $z \in M$ there is a derivation of z in $(\mathcal{P}, \mathcal{S}, \mathcal{D}_M)$.

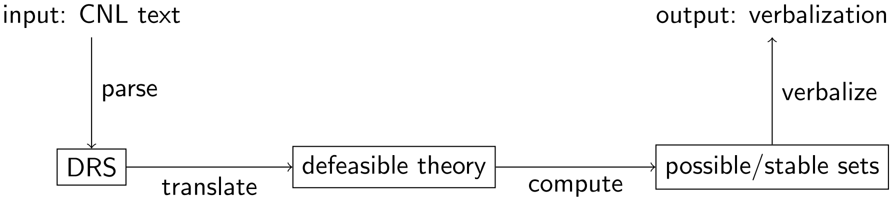
- Defeasible theories with variables: built from n-ary predicate symbols and variables as in logic-programming
  - Semantics via grounding
- See (Strass and Wyner, 2017) for results on expressivity and complexity of direct-stable semantics

Advantages of the approach

- incremental - adding statements and getting extensions without argument construction
- partial/incomplete KBs - answer sets even with missing premises
- no opacity of attack
- three senses of ‘argument’: (Wyner et al., 2015)
  1. argument (rule),
  2. case (derivation),
  3. debate (cases for x and $\neg x$).

Architecture

of our approach



A protoype

www.dbai.tuwien.ac.at/proj/adf/dAceRules/

- Currently we have an experimental adaptation of AceRules for our purposes.
    - i.e. supports defeasible rules using "It is usual that ..." in a rule.
- Interleaves calls to AceRules (and APE) parser, answer set programming (ASP) encodings of direct stable semantics of (and ASP solver), and APE paraphrasing for verbalisation of results.
- Tracks and processes defeasible rules externally.
- Ongoing work: develop an implementation that does not rely on AceRules.
    - Improve "intelligent grouping" mechanism

Extended example: generic generalisations in AceWiki

- AceWiki (Kuhn, 2009): prototype of an encyclopedia in Wikipedia-style, with articles written using ACE.
- `attempto.ifi.uzh.ch/acewiki`
- currently there are two demo wikis:
  - `attempto.ifi.uzh.ch/webapps/acewikigeo`
  - `attempto.ifi.uzh.ch/webapps/acewikiattempto`
- for complexity-reasons ACE used in AceWiki has been restricted to the fragment that can be translated to OWL 2 RL, and thus also, in principle, to that admitted by AceRules.
- we motivate the need for generic generalisations in the context of the Geo Wiki.

- Example based on actual Wikipedia entries.

Example

Example

(1) Every island is a land-mass.
(2) If X is an island then a body-of-water surrounds X.

- Tied islands "are landforms consisting of an island that is connected to land only by a tombolo: a spit of beach materials connected to land at both ends." (`en.wikipedia.org/wiki/Tied_island`).

Example

`(1) Every island is a land-mass.`
`(2) If X is an island then a body-of-water surrounds X.`
(3) Every tied-island is an island.
(4) Every tied-island attaches-to a land-mass.

Example

(1) `Every island is a land-mass.`
(2) `If X is an island then a body-of-water surrounds X.`
(3) `Every tied-island is an island.`
(4) `Every tied-island attaches-to a land-mass.`
(5) Mainland-Shetland is an island.
(6) St-Ninians-Isle is a tied-island.
(7) St-Ninians-Isle is a part of the Shetland-Islands.

- “During the winter strong wave action removes sand from the tombolo that connects St. Ninian to Mainland Shetland such that the tombolo is usually covered at high tide and occasionally throughout the tidal cycle (...)” (en.wikipedia.org/wiki/St_Ninian’s_Isle).

Example

⋮

(8) It is usual that St-Ninians-Isle attaches-to Mainland-Shetland.

- Meanings of “being attached to a land mass”, “being surrounded by water”, and “being a part of”.

Example

⋮

9) If X attaches-to a land-mass then it is false that a body-of-water surrounds X.
(10) If a body-of-water surrounds X then it is false that X attaches-to a land-mass.
(11) If St-Ninians-Isle attaches-to Mainland-Shetland then St-Ninians-Isle is a part of Mainland-Shetland.
(12) If St-Ninians-Isle attaches-to Mainland-Shetland then St-Ninians-Isle attaches-to a land-mass.
(13) If St-Ninians-Isle attaches-to a land-mass then St-Ninians-Isle attaches-to Mainland-Shetland.

Example

```
(1) Every island is a land-mass.
(2) If X is an island then a body-of-water surrounds X.
(3) Every tied-island is an island.
(4) Every tied-island attaches-to a land-mass.
(5) Mainland-Shetland is an island.
(6) St-Ninians-Isle is a tied-island.
(7) St-Ninians-Isle is a part of the Shetland-Islands.
(8) It is usual that St-Ninians-Isle attaches-to
Mainland-Shetland.
⋮
```

Inconsistent!

- St. Ninian a tied-island $\Rrightarrow$ attached to a land-mass.
- St. Ninian a tied-island $\Rrightarrow$ St. Ninian an island $\Rrightarrow$ surrounded by a body of water.

Reason for the apparent contradiction: [d]epending on the definition used, St. Ninians is (...) either an island, or a peninsula. (`en.wikipedia.org/wiki/St_Ninian's_Isle`)

Example

```
(1) Every island is a land-mass.
(2) If X is an island then a body-of-water surrounds X.
(3) Every tied-island is an island.
(4) Every tied-island attaches-to a land-mass.
(5) Mainland-Shetland is an island.
(6) St-Ninians-Isle is a tied-island.
(7) St-Ninians-Isle is a part of the Shetland-Islands.
(8) It is usual that St-Ninians-Isle attaches-to
Mainland-Shetland.
⋮
```

Example

(1) Every island is a land-mass.

(2') If X is an island then it is usual that a body-of-water surrounds X.

(3') If X is a tied-island then it is usual that X is an island.

(4') If X is a tied-island then it is usual that X attaches-to a land-mass.

(5) Mainland-Shetland is an island.

(6) St-Ninians-Isle is a tied-island.

(7) St-Ninians-Isle is a part of the Shetland-Islands.

(8) It is usual that St-Ninians-Isle attaches-to Mainland-Shetland.

⋮

Example - Answer-sets

Two answer-sets, having in common:

```
There is a body-of-water X1.
St-Ninians-Isle is a tied-island.
St-Ninians-Isle is an island.
Mainland-Shetland is a land-mass.
Mainland-Shetland is an island.
St-Ninians-Isle is a part of Shetland-Islands.
The body-of-water X1 surrounds Mainland-Shetland.
It is false that Mainland-Shetland attaches-to a
land-mass.
```

Example - Answer-sets

First answer-set:

```
There is a body-of-water X1.
St-Ninians-Isle is a tied-island.
St-Ninians-Isle is an island.
Mainland-Shetland is a land-mass.
Mainland-Shetland is an island.
St-Ninians-Isle is a part of Shetland-Islands.
The body-of-water X1 surrounds Mainland-Shetland.
It is false that Mainland-Shetland attaches-to a
land-mass.
St-Ninians-Isle is a part of Mainland-Shetland.
St-Ninians-Isle attaches-to a land-mass.
St-Ninians-Isle attaches-to Mainland-Shetland.
It is false that a body-of-water surrounds
St-Ninians-Isle.
```

Example - Answer-sets

Second answer-set:

There is a body-of-water X1.
St-Ninians-Isle is a tied-island.
St-Ninians-Isle is an island.
Mainland-Shetland is a land-mass.
Mainland-Shetland is an island.
St-Ninians-Isle is a part of Shetland-Islands.
The body-of-water X1 surrounds Mainland-Shetland.
It is false that Mainland-Shetland attaches-to a land-mass.
There is a body-of-water X2.
St-Ninians-Isle is a land-mass.
The body-of-water X2 surrounds St-Ninians-Isle.
It is false that St-Ninians-Isle attaches-to a land-mass.

The example in AceRules

Example with negation as failure

(1) Every island is a land-mass.
(2) If X is an island and it is not provable that a body-of-water surrounds X then a body-of-water surrounds X.
(3) If X is a tied-island and it is not provable that X is not an island then X is an island.
(4) If X is a tied-island and it is not provable that it is false that X attaches-to a land-mass thenthen X attaches-to a land-mass.
(5) Mainland-Shetland is an island.
(6) St-Ninians-Isle is a tied-island.
(7) St-Ninians-Isle is a part of the Shetland-Islands.
(8) If it is not provable that it is false that St-Ninians-Isle attaches-to Mainland-Shetland, then St-Ninians-Isle attaches-to Mainland-Shetland.
...

For the example using negation-as-failure:

- No answer-sets under the stable-semantics for logic programs (Gelfond and Lifschitz, 1990).
- Rules are cyclic: no result under the courteous semantics (Grosof, 1997).

Conclusions

Current work:

- We have an approach and prototype for argumentation-inspired reasoning on defeasible ACE rule knowledge bases.

Ongoing work:

- Improve implementation.
- Also have support for justifications.

Future work

Future work / speculation:

- Other forms of generic/defeasible generalizations:
  - Lions have manes. (generic count noun)
  - Bill walks to work at 9:00... (progressives)
  - Water freezes at 0 degrees centigrade. (mass and simple present tense)
- Generic generalizations as the default, strict rules as the exception?
- Abduction
- Argumentation schemes
- Connect with results in argument-mining
- ...

The End

Defeasible Theories in Instantiated Argumentation

Basic workflow: construct arguments, detect conflicts, use conflict relation to evaluate the theory...

Argument construction

- if $b_1, \ldots, b_m \to h$ is a rule and we have arguments $A_1, \ldots, A_m$ for $b_1, \ldots, b_m$, then $[A_1, \ldots, A_m \to h]$ is an argument for h
- similar: defeasible rules lead to defeasible arguments
- argument is defeasible if some subargument is defeasible

Conflict detection

- argument for x ($\neg x$) attacks any defeasible argument with a subargument for $\neg x$ (x)

Why argumentation? Humans often communicate reasoning in form of pro/con arguments.

Defeasible Theories in Instantiated Argumentation

Basic workflow: construct arguments, detect conflicts, use conflict relation to evaluate the theory...

Evaluation of argumentation graph

- Conflict relation between arguments gives rise to an argumentation framework (Dung, 1995)
- Can be evaluated via semantics based on the reinstatement principle: roughly: an argument is to be accepeted as long as it can be defended from all attacks
- Semantics give rise to extensions: sets of arguments that can all be accepted together
- Extensions can then be mapped back to sets of atoms

Defeasible Theories in Instantiated Argumentation

Basic workflow: construct arguments, detect conflicts, use conflict relation to evaluate the theory...

(Some) problems:

- Potential exponential blowup of arguments
- Care that rationality postulates hold
  - consistency, closure, ...