

SVEUČILIŠTE U ZAGREBU  
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

SEMINAR  
PREDSTAVLJANJE ZNANJA U INFORMACIJSKIM  
SUSTAVIMA

# **Prikaz argumenata u informacijskim sustavima**

*Student: Filip Boltužić*

*Voditelj: Prof. dr. sc. Nikola Bogunović*

Zagreb, siječanj 2018.

# SADRŽAJ

<b>1. Uvod</b>	<b>1</b>
<b>2. Argument</b>	<b>2</b>
2.1. Vrste informacija u argumentaciji . . . . .	4
2.2. Agenti argumentacije . . . . .	5
2.3. Računalna argumentacija . . . . .	6
<b>3. Predstavljanje argumentacije</b>	<b>7</b>
3.1. Argument Interchange Format . . . . .	7
3.2. Specifikacija AIF-a . . . . .	8
3.2.1. Koncepti i relacije . . . . .	8
3.2.2. Čvorovi . . . . .	8
3.2.3. Bridovi . . . . .	8
3.3. Primjeri AIF dokumenta . . . . .	9
3.3.1. XML RDF primjer AIF dokumenta . . . . .	9
3.3.2. AML primjer AIF dokumenta . . . . .	11
3.3.3. RTNL primjer AIF dokumenta . . . . .	11
3.3.4. LKIF primjer AIF dokumenta . . . . .	12
3.3.5. DOT primjer AIF dokumenta . . . . .	13
3.3.6. SVG primjer AIF dokumenta . . . . .	14
3.4. AIFdb . . . . .	15
<b>4. Analiza argumentacije</b>	<b>17</b>
4.1. Kolaborativna analiza . . . . .	18
<b>5. Evaluacija argumentacije</b>	<b>20</b>
5.1. ASPIC+ . . . . .	21
5.2. TOAST . . . . .	23

<b>6. Zaključak</b>	<b>25</b>
<b>7. Literatura</b>	<b>26</b>

# 1. Uvod

Argumentiranje je obrazlaganje zauzetog stajališta s ciljem uvjeravanja publike (Walton, 1990). Ukoliko ovaj seminar nije dovoljno dobar za položiti predmet *Predstavljanje znanja u skupovima podataka*, mogu se poslužiti argumentima koji će tvrditi suprotno s ciljem uvjeravanja profesora. Pojavom interneta i društvenih mreža, argumentiranje je postalo pristupačnije no ikad; Reddit, jedna od najvećih platformi za rasprave, imala je preko 1500 (234 različitih) milijuna posjetitelja mjesečno u 2017. godini, koji diskutiraju o preko milijun različitih tematika (subreddita) <sup>1</sup>.

Argumentacija se javlja u raznim oblicima: online rasprave za ili protiv uvođenja valutne klauzule, nadmetanje odvjetnika za pobjedu u pravnim slučajevima ili debata oko originalnosti doprinosa znanstvenog rada. Neke rasprave, kao online rasprave, nemaju strukturu ili pravila, dok znanstvene rasprave predstavljaju potpunu suprotnost s dobro definiranom strukturom rasprave (motivacija, hipoteza, dokaz, eksperimenti, zaključak). Strukturiranje rasprava olakšava ulazak novih sudionika u raspravu, kao i kritičku analizu rasprave. Novi sudionik strukturirane rasprave lakše će uvidjeti i ocijeniti koji argumentima nedostaje dokaza ili tko je počinio grešku u zaključivanju (Rieppel, 1992). Kako izvora argumentacije ima mnogo, korištenje računala nameće se kao prirodan izbor za pomoć pri analizi. Računalna obrada argumentacije razvija se vrlo intenzivno u zadnjih 20 godina s ciljem strukturiranja rasprava, donošenja novih zaključaka i evaluacije valjanosti argumentacije.

U ostatku seminarskog rada, objasnit će se pojmovi vezani uz argument i argumentaciju (poglavlje 2), definirati što je računalna argumentacija i zašto je potrebna (odjeljak 2.3). Nakon uvoda u računalnu argumentaciju, govorit će se o dijelovima računalne argumentacije: predstavljanju argumentacije u računalu (poglavlje 3), analizi argumentacije (poglavlje 4 i evaluaciji argumentacije u računalu (poglavlje 5).

---

<sup>1</sup>Brojevi preuzeti 7.1.2018. s <https://en.wikipedia.org/wiki/Reddit> i <http://redditmetrics.com/history>

## 2. Argument

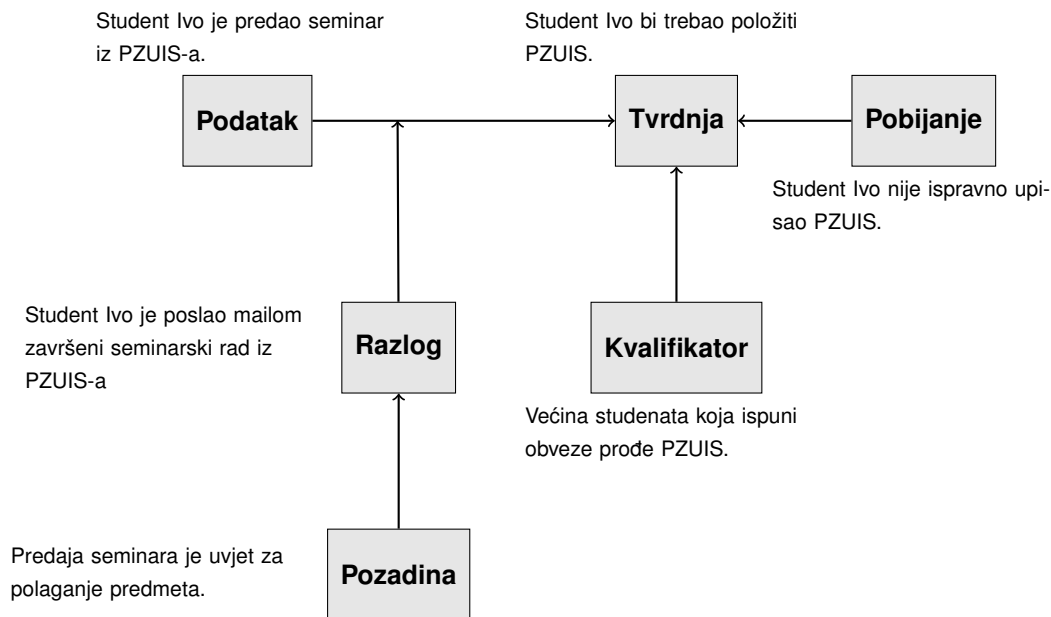
**Argument** je sredstvo razrješavanja sukoba oko kontroverzne izjave (engl. *claim*) (Walton, 1990) između dva ili više sudionika. Sastoji se od barem jedne pretpostavke ili **premise** (engl. *premise*) i **zaključka** (engl. *conclusion*) koji logički slijedi iz pretpostavki kroz **postupak zaključivanja** (engl. *reasoning*). Središnji dio argumenta je zaključak, koji se smatra kontroveznom izjavom ili tvrdnjom, a cilj iznošenja argumenata je uvjeriti publiku kako je argument ispravan (Walton, 1990). Premise (pretpostavke) opravdavaju (engl. *justify*) tvrdnju argumenta (Besnard i Hunter, 2008). Primjerice, argument *A* sadrži rečenice:

- (1) student Ivo je uspješno predao seminar iz predmeta *Predstavljanje znanja u skupovima podataka*,
- (2) svaki student koji uspješno preda seminar iz predmeta *Predstavljanje znanja u skupovima podataka* je položio predmet *Predstavljanje znanja u skupovima podataka*, i
- (3) student Ivo je položio predmet *Predstavljanje znanja u skupovima podataka*.

gdje su (1) i (2) premise argumenta *A* iz kojih slijedi tvrdnja (zaključak) (3).

Toulmin (1974), vrlo utjecajni filozof i retoričar 20. stoljeća, definirao je strukturu kvalitetnog argumenta. Specijalizirao je uloge premisa i tvrdnji tako što je definirao:

1. **tvrdnju** (engl. *claim*): središnju izjavu argumenta,
2. **podatke** (engl. *data*): činjenice i dokaze argumenta,
3. **razlog** (engl. *warrant*): logičku vezu između tvrdnje i podataka,
4. **pozadinu** (engl. *backing*): izjave koje podupiru razlog,
5. **kvalifikator** (engl. *qualifier*): uvjet istinitosti tvrdnje ili snaga tvrdnje, i
6. **pobijanje** (engl. *rebuttal*): izjave koje ističu uvjete pod kojima argument nije istinit.



**Slika 2.1:** Toulminov primjer kvalitetnog argumenta

Elementi Toulminovog argumenta i njihovi međudodnosi prikazani su na slici 2.1. Zbog svoje jednostavnosti, ali i lake mogućnosti primjene na stvarne argumentativne rasprave, Toulminovom strukturom argumenta (Toulminovim modelom) najčešće se modelira argumentacija u području umjetne inteligencije.

Argumenti se mogu međusobno povezivati, uspoređivati i vrednovati. Kažemo da argument *A* pobija (engl. *rebutts*) argument *B* ukoliko argument *A* sadrži tvrdnju koja je kontradiktorna tvrdnji ili bilo kojoj premisi iz argumenta *B* (Besnard i Hunter, 2008). Primjer argumenta *B* u kontradikciji s argumentom sa slike 2.1 ima tvrdnju:

- (4) student Ivo nije obavio predaju seminara iz PZUIS-a na vrijeme.

gdje je tvrdnja (4) iz argumenta *B* u kontradikciji s premisom (1) argumenta sa slike 2.1.

Argumenti se međusobno povezuju postupcima zaključivanja (engl. *reasoning*). Izvođenje zaključaka ili tvrdnji iz premisa može se raditi pod različitim pravilima logike. Tako razlikujemo logiku prvog reda, predikatnu logiku, neformalnu i brojne druge. Neformalna logika (engl. *informal logic*) formalizira, razvija metode evaluacije i vrednovanja argumentacije svakodnevnog razgovora (Fogelin, 1985; Blair i Johnson, 2000). Upravo se na neformalnu logiku naslanja moderna teorija argumentacije procjenjuje valjanost argumenata u raspravama pravilima neformalne logike. Više o zaključivanju i valjanosti argumenata u poglavlju 5.

## 2.1. Vrste informacija u argumentaciji

Besnard i Hunter (2008) razlikuju subjektivne, objektivne ili hipotetske informacije u argumentaciji. S obzirom na izrazitost, informacije mogu biti sigurne (engl. *certain*) i nesigurne (engl. *uncertain*). **Sigurne** (kategoričke) informacije smatraju se uvijek istinitima. Primjeri takvih informacije su matematičke formule ili zdravorazumske činjenice prema kojima je *Zagreb glavni grad Hrvatske*. Također tu pripadaju i informacije čija je vjerojatnost istinitosti izrazito visoka, primjerice: *danas navečer zalazi sunce*. Svaka informacija koja nije sigurna je **nesigurna** informacija. Stupanj sigurnosti informacije ovisi o kontekstu u kojem se procjenjuje, primjerice: *Hrvatska ima jako dobru poljoprivredu, student Ivo ima 20 godina, tipkovnica ima 104 tipke...* Postoje još brojne druge dimenzije informacije, kao što su neodređenost ili vjerojatnost informacije, o kojima je moguće više pronaći u literaturi posvećenoj formalnom zaključivanju (Wang et al., 2004).

**Subjektivne informacije** ukorjenjene su u mišljenjima, stavovima i vjerovanjima ljudi koje ih iznose. Odabir studija za studenta često uključuje procjenu subjektivnih informacija, kao što su *osobni profesionalni interesi, težina fakultetskih programa, mogućnosti zaposlenja nakon završetka fakulteta*. Ukoliko isti izbor gledamo više sa **objektivnog stajališta**, tada ćemo se služiti egzaktnim, provjerenim i pouzdanim informacijama, kao što su: *90% ljudi koji završe FER nađu posao u 6 mjeseci nakon diplomiranja, prosječno trajanje studiranja na FER-u je 6 godina...* Sadržaj **hipotetskih informacija** se pretpostavlja i ne ispituje, već služi kao priprema za argumentaciju koja slijedi. Moguće je da hipotetska informacija nije točna, niti će ikad biti, ali pretpostavljanjem njene istinitosti istražuju se posljedice hipotetske izjave. Primjerice, *recimo da student Ivo uspješno položi PZUIS, student Ivo bi tada položio sve predmete na doktorskom studiju i ispunio jedan od preduvjeta završetka studija..* Hipotetski dio argumenta *recimo da student Ivo uspješno položi PZUIS* služi nam kako bismo pokazali posljedice ostvarivanja istinitosti te informacije, ali pritom ne ispituje istinitost same hipotetske izjave.

Je li informacija objektivna, subjektivna ili hipotetska, te koliko je njen stupanj sigurnosti uvelike ovisi o kontekstu (Oren et al., 2007). U oblikovanju konteksta veliku ulogu ima izvor informacije — agent.

## 2.2. Agenti argumentacije

Agent (engl. *agent*) je autonoman, proaktivan i nezavistan sustav s određenom ulogom u argumentaciji (Besnard i Hunter, 2008). Agenti mogu biti sudionici debate, pripadnici porote, liječnici, odvjetnici, ali i **računalni sustavi**. Agenti imaju razne uloge u sklopu argumentacije. Uloga odvjetnika može biti dokazivanje krivnje osumnjičenog kroz argumentaciju. Kvalitetu argumentacije odvjetnika evaluiraju agenti — članovi porote koji donose odluku o presudi. S obzirom na broj agenata involviranih u argumentaciju, razlikujemo argumentaciju jednog agenta — **monološku argumentaciju** (engl. *monological argumentation*) i više agenata — **dijalošku argumentaciju** (engl. *dialogical argumentation*).

U **monološkoj argumentaciji** jedan agent iznosi argumente za i protiv određene tvrdnje. Ukoliko student Ivo piše blog na temu: *Koji fakultet upisati?*, on se može služiti objektivnim informacijama (*90% ljudi se zaposlilo nakon što je završilo moj faks*), subjektivnim informacijama (*profesori na faksu su ljubazni*) ili hipotetskim informacijama (*da sam upisao drugi faks ...*). Izrečene izjave strukturiraju se kroz premise i zaključke te formiraju konačan stav prema temi (*student Ivo je upisao ispravan fakultet*). Neki praktični primjeri monološke argumentacije su eseji, blogovi i kritike.

**Dijaloška argumentacija** uključuje više od jedne osobe koje argumentiraju za ili protiv zadane teme ili tvrdnje. Agenti mogu imati međusobno suprotstavljene stavove, međusobno se uvjeravati (engl. *persuade*) u ispravnost stavova. Rasprava *Koji faks upisati?* može se odvijati putem online foruma gdje student Ivo i ostali studenti sveučilišta uvjeravaju gimnazijalce koji fakultet odabrati. Naglasak u dijaloškoj argumentaciji je na procesu iznošenja argumenata agenata i formiranju konačnih stavova agenata. Neki primjeri dijaloške argumentacije su rasprave odvjetnika u sudnici, rasprave u online diskusijama ili pregovaranje cijene automobila.

Bez obzira radi li se o monološkoj ili dijaloškoj argumentaciji, agenti analiziraju i vrednuju argumentaciju prema brojnim kriterijima kao što su logička valjanost postupaka zaključivanja, kvaliteta subjektivnih argumenata ili pouzdanost objektivnih argumenata. Analizom argumentacije često se bavimo nesvjesno samim time uvjerava li nas argumentacija da *upišemo FER zbog argumenata studenta Ive, kupimo auto marke BMW zbog njegovih karakteristika* ili *glasamo za određenu stranku zbog njihovog karizmatičnog čelnika*.



## 2.3. Računalna argumentacija

Analiza argumentacije može biti zahtjevna, kompleksna, posebice za pojedinca. Analizom složenijih argumentacija bave se skupine ljudi, koji za to koriste pomoć računala. Idejni začetnik povezivanja računala i argumentacije je Dung (1995). Dung je prvi formalizirao sustav koji koristi oborivu (engl. *defeasible*) logiku zaključivanja, jer je smatrao da je takva logika prikladnija u argumentaciji prava i medicine od dotad klasične logike (engl. *classical logic*). Dungov rad smatra se začecem nove grane umjetne inteligencije — **računalne argumentacije** (engl. *computational argumentation*), područje posvećeno računalnoj obradi argumentacije.

Jedan od glavnih ciljeva računalne argumentacije je oblikovanje **argument weba**, velike mreže povezanih korisnički izraženih argumenata na Webu. Ostvarivanje argument weba dijeli se na (Reed et al., 2017):

1. predstavljanje argumenatacije u računalu (engl. *argument representation*),
2. analizu i vizualizaciju argumentacije (engl. *argument visualization and analysis*),
3. podučavanje argumentacije (engl. *argument pedagogy*),
4. evaluaciju argumentacije (engl. *argument evaluation*) i
5. dubinsku analizu argumentacije (engl. *argument mining*).

U argument webu ne postoje nestrukturirane rasprave, već svaka rasprava slijedi logičku strukturu te je moguće lako saznati koji je dokaz *globalnog zatopljenja* ili *koja istraživanja tvrde da cjepivo nije dobro za ljude*. U nastavku seminara, reći ćemo nešto više o predstavljanju argumentacije u računalu u poglavlju 3, o analizi i vizualizaciji argumentacije bit će govora u poglavlju 4, dok će evaluacija argumentacije bit će obrađena u poglavlju 5. Podučavanje argumentacije nastoji učenike i studente učiti kritičkom mišljenju i konstrukciji kvalitetnih argumenata, o čemu neće biti govora u ostatku seminara, dok je dubinska analiza argumentacije najnovije multidisciplinarno područje koje pokušava metodama obrade prirodnog jezika (engl. *natural language processing*) ekstrahirati argumente iz nestrukturarnog teksta o čemu ćemo se osvrnuti u zaključku kao posljednjem, i najtežem, koraku ostvarivanja argument weba.

## 3. Predstavljanje argumentacije

Predstavljanje argumenata u računalima moguće kroz standardne serijalizacijske formate, kao što su XML (engl. *Extensible Markup Language*) ili JSON (engl. *JavaScript Object Notation*). No, radi se generičkim formatima, koji ne poznaju specifičnosti strukture argumenata. Nezavisno razvijeni alati za analizu argumentacije kao Araucaria (Reed i Rowe, 2004), Rationale (Van Gelder, 2007) ili Carnedeas (Gordon i Walton, 2006) su razvili vlastiti serijalizacijski format pohrane argumenata. Kako je potreba za integracijom između tih alata rasla, tako je i rasla potreba za ujedinenim formatom za pohranjivanje argumenta.

### 3.1. Argument Interchange Format

Standardizacija definiranja računalnog jezika bila je nužna zbog tri glavna razloga:

1. razmjena dokumenata kroz različite programske agente (primjerice korištenje dokumenata kreiranih u Araucarii u Rationaleu i obratno),
2. kompatibilnost argumentacijskih teorija u različitim programskim agentima (primjerice, Araucaria koristi Toulminovu teoriju, dok Rationale samo poznaje Waltonove argumentacijske teorije (Walton, 2012), te
3. potrebe da se automatski procesiraju logičke izjave

Prije AIF-a bilo je nekoliko pokušaja stvaranja zajedničkog jezika, ponajprije Araucarin AML (engl. *Argument Markup Language*). AML, baziran na XML-u (engl. *Extensible Markup Language*) osmišljen je za označavanje i analizu argumentacije u prirodnom jeziku. Sintaksa AML jezika specificirana je pomoću DTD (engl. *Document Type Definition*) strukturalnih ograničenja. 2005. nastao je prvi prijedlog za **AIF** (engl. *Argument Interchange Format*) jezik (Chesñevar et al., 2006). Baziran na dobro poznatom RDF jeziku, ideja AIF-a je bila ponuditi dovoljnu ekspresivnost kojom bi se obuhvatile sve općeprihvaćene teorije argumentacije i efikasno pohranili argumenti u

računalima. AIF je 2008. dobio proširenje za dijalog (Reed et al., 2008). Začetak AIF-a se smatra prvim korakom prema ostvarivanju argument web-a.

## 3.2. Specifikacija AIF-a

AIF odlikuju 1. sintaksa razumljiva programskim agentima, 2. eksplicitna semantika, 3. koncepti i proširenja te 4. objedinjen apstraktni model koncepata i relacija između koncepata.

### 3.2.1. Koncepti i relacije

Objekti argumentacije predstavljaju se kao skup čvorova povezanih usmjerenim grafom. Neformalno se takav usmjereni graf u kontekstu argumentacije naziva argumentativnom mrežom (engl. *argument network*) **AN**. Ne postoje nikakva ograničenja na oblik grafa koji može poprimiti AN.

### 3.2.2. Čvorovi

Razlikujemo dvije osnovne vrste **čvorova**: informacijske čvorove (engl. *information nodes*) I-čvorove i shematske (engl. *scheme nodes*) S-čvorove. I-čvorovi predstavljaju sadržaj izjava i čvrsto su povezani s temom argumentativne rasprave, S-čvorovi predstavljaju primjenu obrazaca u argumentiranju i smatraju se neovisnim o argumentativnoj raspravi. Postoje tri osnovna tipa obrasca u argumentiranju: posljedica (engl. *inference*), preferiranje (engl. *preference*) i sukob (engl. *conflict*). Primjena sheme radi se kroz S-čvor koji sukladno vrstama obrazaca u argumentiranju može biti čvor primjene logičke posljedice (engl. *inference application node*) (**RA-čvor**), čvor primjene preferiranja (engl. *preference application node*) (**PA-čvor**) te čvor primjene sukoba (engl. *conflict application node*) (**CA-čvor**).

### 3.2.3. Bridovi

**Čvorovi** su povezani usmjerenim bridovima. Kažemo da brid povezuje čvorove  $A$  i  $B$  tako da ide iz početnog čvora  $A$  u odredišni čvor  $B$ . Razlikujemo shematske i podatkovne bridove. Početne točke shematskih bridova su S-čvorovi, dok su početne točke podatkovnih bridova I-čvorovi. Primjerice, čvorovi  $A$  i  $B$  su povezani usmjerenim bridom  $A \rightarrow B$ . Ukoliko je čvor  $A$  početna točka tipa S-čvor primjene logičke posljedice

RA-čvor onda je čvor *B* zaključak strukture čvora *A*. Čvor *B* može biti S-čvor ili I-čvor. Ukoliko je početni čvor I-čvor onda odredišni čvor može biti samo S-čvor. Ideja iza toga stoji u principu da nije moguće povezati dvije izjave bez da se specifiira relacija (S-čvor) između izjava. Chesñevar et al. (2006) navode sve moguće kombinacije S-čvorova i I-čvorova sa bridovima uz pripadajuće semantičko značenje.

### 3.3. Primjeri AIF dokumenta

AIF dokumente moguće je preslikavati u raznim formatima: **OWL/RDF**, **AML** (format kompatibilan sa sustavom Araucaria), **RTNL** (format kompatibilan sa sustavom Rationale), **LKIF** (format kompatibilan sa sustavom Carnedeas), **DOT** (engl. *graph description language*) i **SVG** (engl. *scalable vector graphics*) format. U nastavku će se prikazati jednostavan AIF dokument istovjetnog sadržaja u različitim formatima. AIF dokument sadrži tvrdnje:

- (5) *Student Ivo je predao seminar*
- (6) *Student Ivo nije pristupio ispitu iz PZUIS-a i*
- (7) *Student Ivo bi trebao položiti PZUIS.*

Tvrdnja (7) logički slijedi iz tvrdnje (5), dok tvrdnja (6) je u kontradikciji sa tvrdnjom (7). Između tvrdnji (5) i (6) nema logičke veze.

#### 3.3.1. XML RDF primjer AIF dokumenta

Primjer isječka AIF dokumenta u XML RDF formatu vidljiv je kodu 3.1. Tip čvora definiran je kroz *rdf:type* (I-čvor ili S-čvor), *aif:claimText* predstavlja sadržaj I-čvora, dok S-čvor ima različita svojstva (ovisno o tome radi li se o RA, PA ili CA čvoru). U ovom primjeru imamo po jedan S-čvor zaključivanja (RA-čvor) i S-čvor primjene sukoba (CA-čvor) koji spajaju dva čvora: *aif:Premise* čvor i *aif:Conclusion* čvor.

```
<?xml version="1.0"?>
<!-- http://www.arg.dundee.ac.uk/AIFdb/nodes/319262 -->
<NamedIndividual rdf:about="&http://www.arg.dundee.ac.uk/AIFdb/nodes/319262">
  <rdf:type rdf:resource="&http://www.arg.dundee.ac.uk/aif#I-node"/>
  <aif:claimText>Student Ivo bi trebao položiti PZUIS</aif:claimText>
  <aif:Conclusion rdf:resource="&http://www.arg.dundee.ac.uk/AIFdb/nodes/319264"/>
  <aif:Conclusion rdf:resource="&http://www.arg.dundee.ac.uk/AIFdb/nodes/319266"/>
  <aif:creationDate>2017-12-20 18:17:01</aif:creationDate>
```

```

</NamedIndividual>

<!-- http://www.arg.dundee.ac.uk/AIFdb/nodes/319263 -->
<NamedIndividual rdf:about="&http://www.arg.dundee.ac.uk/AIFdb/nodes/319263">
  <rdf:type rdf:resource="&http://www.arg.dundee.ac.uk/aif#I-node"/>
  <aif:claimText>Student Ivo je predao seminar</aif:claimText>
  <aif:Premise rdf:resource="&http://www.arg.dundee.ac.uk/AIFdb/nodes/319264"/>
  <aif:creationDate>2017-12-20 18:17:01</aif:creationDate>
</NamedIndividual>

<!-- http://www.arg.dundee.ac.uk/AIFdb/nodes/319264 -->
<NamedIndividual rdf:about="&http://www.arg.dundee.ac.uk/AIFdb/nodes/319264">
  <rdf:type rdf:resource="&http://www.arg.dundee.ac.uk/aif#RA-node"/>
  <aif:Premise rdf:resource="&http://www.arg.dundee.ac.uk/AIFdb/nodes/319263"/>
  <aif:Conclusion rdf:resource="&http://www.arg.dundee.ac.uk/AIFdb/nodes/319262"/>
  <aif:creationDate>2017-12-20 18:17:02</aif:creationDate>
</NamedIndividual>

<!-- http://www.arg.dundee.ac.uk/AIFdb/nodes/319265 -->
<NamedIndividual rdf:about="&http://www.arg.dundee.ac.uk/AIFdb/nodes/319265">
  <rdf:type rdf:resource="&http://www.arg.dundee.ac.uk/aif#I-node"/>
  <aif:claimText>Student Ivo nije pristupio ispitu iz PZUIS-a.</aif:claimText>
  <aif:Premise rdf:resource="&http://www.arg.dundee.ac.uk/AIFdb/nodes/319266"/>
  <aif:creationDate>2017-12-20 18:17:02</aif:creationDate>
</NamedIndividual>

<!-- http://www.arg.dundee.ac.uk/AIFdb/nodes/319266 -->
<NamedIndividual rdf:about="&http://www.arg.dundee.ac.uk/AIFdb/nodes/319266">
  <rdf:type rdf:resource="&http://www.arg.dundee.ac.uk/aif#CA-node"/>
  <aif:Premise rdf:resource="&http://www.arg.dundee.ac.uk/AIFdb/nodes/319265"/>
  <aif:Conclusion rdf:resource="&http://www.arg.dundee.ac.uk/AIFdb/nodes/319262"/>
  <aif:creationDate>2017-12-20 18:17:02</aif:creationDate>
</NamedIndividual>

```

**Listing 3.1:** Primjer AIF RDF dokumenta

### 3.3.2. AML primjer AIF dokumenta

Primjer isječka AIF dokumenta u AML formatu vidljiv je kodu 3.2. U AML formatu čvorovi nisu eksplicitno definirani, već struktura ima oblik n-arnog stabla. Čvor stabla odgovara I-čvoru, dok je S-čvor definiran kroz tip veze koja povezuje čvor sa djetetom. U kodu 3.2 vidimo da je tvrdnja (7) korijen stabla povezan sa CA-čvorom (CA) prema djetetu (5) i RA-čvor (*REFUTATION*) vezom prema čvoru (6).

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE ARG SYSTEM "argument.dtd">
<ARG>
  <?Araucaria UTF-8?>
  <AU>
    <PROP identifier="A" missing="yes">
      <PROPTXT offset="-1">Student Ivo bi trebao položiti PZUIS</PROPTXT>
    </PROP>
    <REFUTATION>
      <AU>
        <PROP identifier="B" missing="no">
          <PROPTXT offset="274">Student Ivo nije pristupio ispitu iz PZUIS-a.</PROPTXT>
        </PROP>
      </AU>
    </REFUTATION>
  <CA>
    <AU>
      <PROP identifier="C" missing="no">
        <PROPTXT offset="99">Student Ivo je predao seminar</PROPTXT>
      </PROP>
    </AU>
  </CA>
</AU>
</ARG>
```

**Listing 3.2:** Primjer AIF AML dokumenta

### 3.3.3. RTNL primjer AIF dokumenta

Rationale sustav koristi format sličan Python kodu. Primjer AIF dokumenta u RTNL formatu vidljiv je u kodu 3.3. Slično kao u AML-u, u RTNL-u se gradi n-arno stablo gdje vrsta veze prema djeci definira tip S-čvora.

```
Node = None
```

```
def CreateChild(parent, type):
```

```
    global Node
```

```
    Node = app.CreateChild(parent, type)
```

```
    return Node
```

```
def SetText(text):
```

```
    app.SetText(Node, text)
```

```
map0_0 = Create("Claim")
```

```
SetText("Student Ivo bi trebao položiti PZUIS")
```

```
map1_1 = CreateChild(map0_0, "CompoundReason")
```

```
map1_2 = CreateChild(map1_1, "Claim")
```

```
SetText("Student Ivo je predao seminar")
```

```
map1_3 = CreateChild(map1_1, "Inference")
```

```
map1_4 = CreateChild(map0_0, "CompoundObjection")
```

```
map1_5 = CreateChild(map1_4, "Claim")
```

```
SetText("Student Ivo nije pristupio ispitu iz PZUIS–a.")
```

```
map1_6 = CreateChild(map1_4, "Inference")
```

**Listing 3.3:** Primjer AIF RTNL dokumenta

### 3.3.4. LKIF primjer AIF dokumenta

Primjer AIF dokumenta u LKIF formatu vidljiv je u kodu 3.4. Carnedeasov format je nešto sličniji AML-ovom i RDF XML AIF-ovom. Prvo se nabrajaju sve tvrdnje u *statements* oznaci (tagu), koje odgovaraju I-čvorovima, dok se kroz *arguments* definiraju S-čvorovi s pripadajućim tipom (*scheme*).

```
<?oxygen RNGSchema="../../../schemas/LKIF2.rnc" type="compact"?>
```

```
<lkif version="2.0.4">
```

```
  <argument–graphs>
```

```
    <argument–graph id="NewGraph" title="New Graph">
```

```
      <statements>
```

```
        <statement assumption="true" id="319262" standard="BA" value="unknown">
```

```
          <s>Student Ivo bi trebao položiti PZUIS</s>
```

```
        </statement>
```

```
        <statement assumption="true" id="319263" standard="BA" value="unknown">
```

```

        <s>Student Ivo je predao seminar</s>
    </statement>
    <statement assumption="true" id="319265" standard="BA" value="unknown">
        <s>Student Ivo nije pristupio ispitu iz PZUIS—a.</s>
    </statement>
</statements>
<arguments>
    <argument direction="pro" id="319264" scheme="Default Inference" weight="0.5">
        <conclusion statement="319262"/>
        <premises>
            <premise polarity="positive" type="ordinary" role="" statement="319263"/>
        </premises>
    </argument>
    <argument direction="con" id="319266" scheme="Default Conflict" weight="0.5">
        <conclusion statement="319262"/>
        <premises>
            <premise polarity="positive" type="ordinary" role="" statement="319265"/>
        </premises>
    </argument>
</arguments>
</argument-graph>
</argument-graphs>
</lkif>

```

**Listing 3.4:** Primjer AIF LKIF dokumenta

### 3.3.5. DOT primjer AIF dokumenta

DOT prikaz tipičan je za prikaz grafova, pa je stoga pogodan za prikaz argumentacije i mreže argumenata. Primjer AIF dokumenta u DOT formatu vidljiv je u kodu 3.5. Čvorovi u DOT formatu odgovaraju čvorovima. Oznaka čvora (*label*) označava tip čvora, s time da u DOT formatu nema hijerarhije vrsta čvorova (nema apstraktnih S-čvorova, već samo RA, PA ili CA-čvorova). Čvorovi su povezani jednostavnim usmjerenim vezama (označenim  $\rightarrow$ ).

```

digraph nodeset13108 {
    319262 [label="Student Ivo bi trebao\npoložiti PZUIS"];
    319263 [label="Student Ivo je predao seminar"];
    319264 [label="Default Inference"];

```



```

319265 [label="Student Ivo nije pristupio\nispitu iz PZUIS—a.");
319266 [label="Default Conflict"];
319264 --> 319262; 319263 --> 319264; 319266 --> 319262; 319265 --> 319266;

```

**Listing 3.5:** Primjer AIF DOT dokumenta

### 3.3.6. SVG primjer AIF dokumenta

SVG format definira vektorizirane objekte i većina preglednika slika podržava prikaz SVG datoteka. Primjer AIF dokumenta u SVG XML formatu vidljiv je u kodu 3.6. U slučaju SVG formata najteže je doći do argumentacijske mreže sa gledišta parsiranja dokumenta, ponajviše jer je prioritet SVG formata prikaz slike pa su podaci nisu strukturirani tako da se kroz jednu oznaku (tag) može pronaći I-čvor ili S-čvor.

```

<?xml version="1.0" encoding="UTF-8"?>
<svg xmlns="http://www.w3.org/2000/svg" xmlns:xlink="http://www.w3.org/1999/xlink" width="336pt" height="188pt" viewBox="0.00 0.00 336.00 188.00">
  <g id="graph1" class="graph" transform="scale(1 1) rotate(0) translate(4 184)">
    <title>nodeset13108</title>
    <polygon fill="white" stroke="white" points="-4,5 -4,-184 333,-184 333,5 -4,5" />
    <!-- 319262 -->
    <g id="node1" class="node">
      <title>319262</title>
      <polygon fill="#ebf3ff" stroke="#6666cc" points="220,-36 114,-36 114,-1.77636e-14 220,-3.55271e-15 220,-36" />
      <text text-anchor="middle" x="167" y="-23" font-family="FreeSans" font-size="10.00">Student Ivo bi trebao</text>
      <text text-anchor="middle" x="167" y="-9" font-family="FreeSans" font-size="10.00">položiti PZUIS</text>
    </g>
    <!-- 319263 -->
    <g id="node2" class="node">
      <title>319263</title>
      <polygon fill="#ebf3ff" stroke="#6666cc" points="154,-180 8,-180 8,-144 154,-144 154,-180" />
      <text text-anchor="middle" x="81" y="-160" font-family="FreeSans" font-size="10.00">Student Ivo je predao seminar</text>
    </g>
    <!-- 319264 -->
    <g id="node3" class="node">
      <title>319264</title>
      <polygon fill="#e6ffe6" stroke="#66cc66" points="81,-108 0.206152,-90 81,-72 161.794,-90 81,-108" />
      <text text-anchor="middle" x="81" y="-88" font-family="FreeSans" font-size="10.00">Default Inference</text>
    </g>
    <!-- 319263&#45;&gt;319264 -->
    <g id="edge4" class="edge">
      <title>319263-&gt;319264</title>
      <path fill="none" stroke="black" d="M81,-143.831C81,-136.131 81,-126.974 81,-118.417" />
      <polygon fill="black" stroke="black" points="84.5001,-118.413 81,-108.413 77.5001,-118.413 84.5001,-118.413" />
    </g>
    <!-- 319264&#45;&gt;319262 -->
    <g id="edge2" class="edge">
      <title>319264-&gt;319262</title>
      <path fill="none" stroke="black" d="M97.9908,-75.7751C109.26,-66.3402 124.348,-53.709 137.541,-42.6634" />
      <polygon fill="black" stroke="black" points="140.039,-45.1367 145.46,-36.0336 135.545,-39.7694 140.039,-45.1367" />
    </g>
  </g>

```

```

<!-- 319265 -->
<g id="node4" class="node">
  <title>319265</title>
  <polygon fill="#ebf3ff" stroke="#6666cc" points="316,-180 192,-180 192,-144 316,-144 316,-180" />
  <text text-anchor="middle" x="254" y="-167" font-family="FreeSans" font-size="10.00">Student Ivo nije pristupio</text>
  <text text-anchor="middle" x="254" y="-153" font-family="FreeSans" font-size="10.00">ispitu iz PZUIS-a.</text>
</g>
<!-- 319266 -->
<g id="node5" class="node">
  <title>319266</title>
  <polygon fill="#ffe6e6" stroke="#cc6666" points="254,-108 179.854,-90 254,-72 328.146,-90 254,-108" />
  <text text-anchor="middle" x="254" y="-88" font-family="FreeSans" font-size="10.00">Default Conflict</text>
</g>
<!-- 319265<#45;#gt;319266 -->
<g id="edge8" class="edge">
  <title>319265-&gt;319266</title>
  <path fill="none" stroke="black" d="M254,-143.831C254,-136.131 254,-126.974 254,-118.417" />
  <polygon fill="black" stroke="black" points="257.5,-118.413 254,-108.413 250.5,-118.413 257.5,-118.413" />
</g>
<!-- 319266<#45;#gt;319262 -->
<g id="edge6" class="edge">
  <title>319266-&gt;319262</title>
  <path fill="none" stroke="black" d="M236.812,-75.7751C225.411,-66.3402 210.148,-53.709 196.802,-42.6634" />
  <polygon fill="black" stroke="black" points="198.726,-39.713 188.791,-36.0336 194.263,-45.1057 198.726,-39.713" />
</g>
</g>
</svg>

```

**Listing 3.6:** Primjer AIF SVG dokumenta

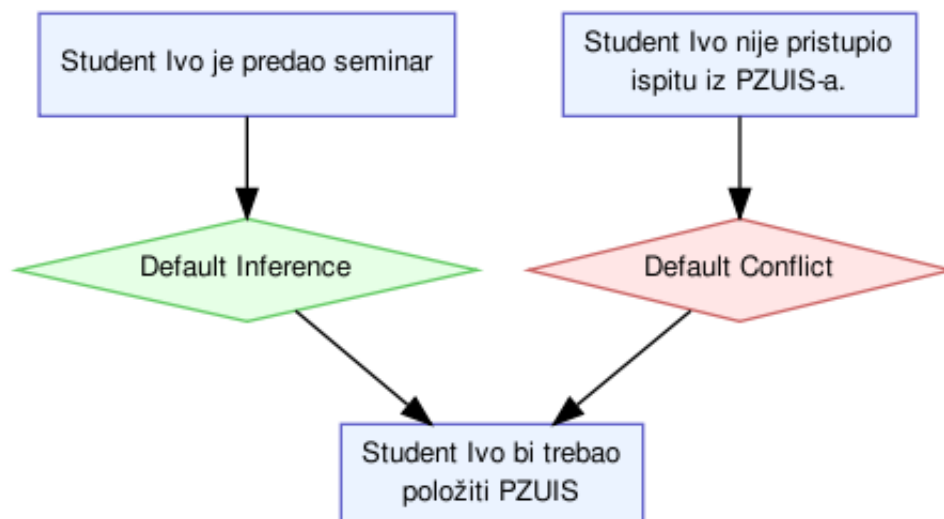
## 3.4. AIFdb

AIFdb<sup>1</sup> (Lawrence et al., 2012) je softversko rješenje za pretraživanje, pohranu i vizualizaciju AIF dokumenata te integraciju s drugim Argument Web alatima. Sastoji se od tri dijela: 1. korisničkog sučelja za vizualizaciju AIF dokumenata, 2. baze podataka koja pohranjuje AIF dokumente i 3. web servisa koji ima programski pristup bazi podataka. Baza podataka pohranjuje dokumente u AIF formatu. AIF dokumentima u bazi podataka moguće je pristupiti kroz korisničko sučelje kroz web preglednik, ili programski, koristeći sučelje web servisa. Putem web servisa moguće je pretraživati i pregledavati AIF dokumente. U AIFdb moguće je uvesti (engl. *import*) dokumente u formatima alata Carnedeas, Rationale i Araucaria te RDF-XML formatu, te izvoziti (engl. *export*) u formatima SVG (engl. *support vector graphics*), DOT (engl. *graph description language*) RDF te formatima alata Rationale i Carnedeas.

Korisnik može preko web sučelja dodavati AIFdb korpuse (Lawrence i Reed, 2014),

---

<sup>1</sup>Dostupan na [aifdb.org](http://aifdb.org)



**Slika 3.1:** Primjer jednostavnog AIF dokumenta u AIFdb korisničkom sučelju

koji grupiraju AIF dokumente. Jedan od korpusa dostupan online u AIFdb-u je i Aracaria korpus koji u trenutku pisanja sadrži 662 AIF dokumenta. Osim pohranjivanja, razmjene i vizualizacije AIF dokumenata, AIFdb je integriran s alatima koji evaluiraju argument u AIF formatu (više u poglavlju 5). Jednostavan primjer AIF dokumenta (iz odjeljka 3.3) pohranjen u AIFdb-u je na slici 3.1. Detalje poput sheme baze podataka moguće je pronaći u (Lawrence i Reed, 2014) i na web stranicama centra za tehnologiju argumentacije (engl. *Centre for Argument Technology*) ARG-tech<sup>2</sup>.

<sup>2</sup>Projekti ARG-techa dostupni na <http://www.arg-tech.org/index.php/projects/>

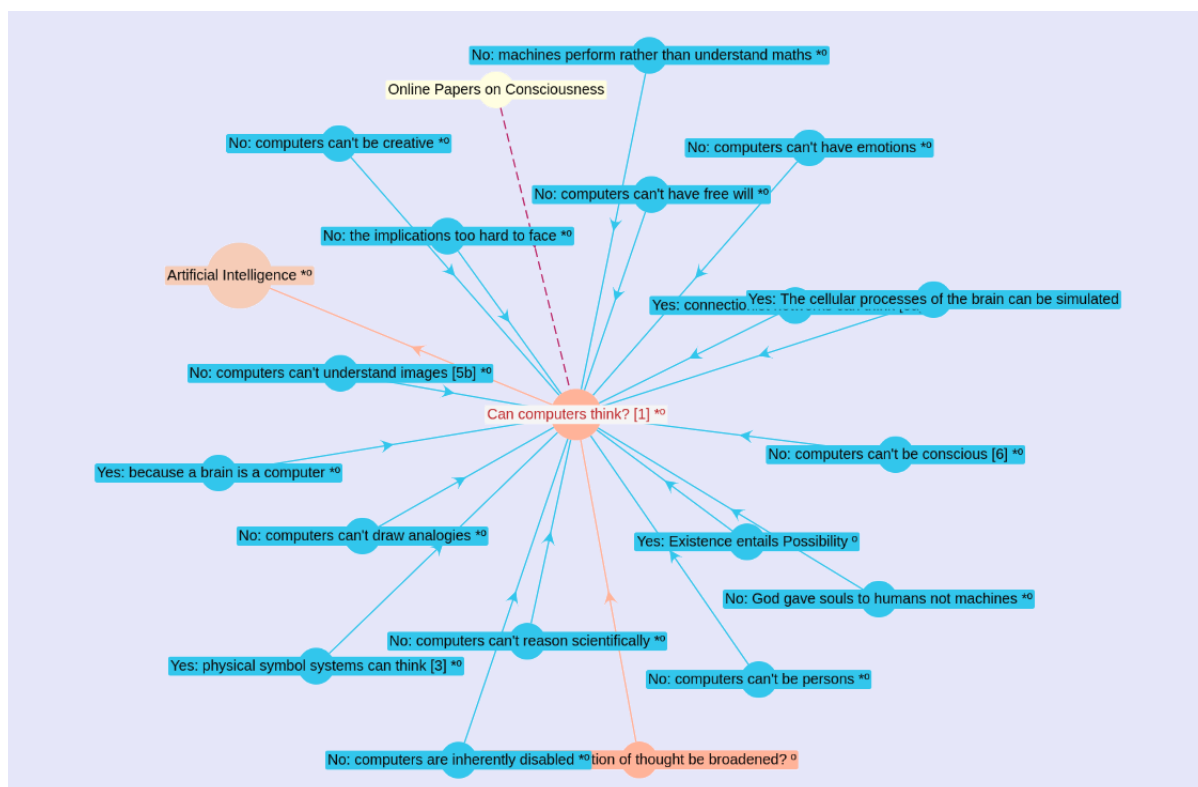
## 4. Analiza argumentacije

Prvi korak u analizi argumentacije je prepoznavanje, strukturiranje i izrada argumenata iz argumentativne rasprave (Scheuer et al., 2010; Prudencio i Toledo, 2005). Izrada argumenata iz teksta radi se izdvajajući tekst koji odgovara specifičnim dijelovima argumenata, povezujući dijelove argumenta ih odgovarajućim relacijama. Tako u tekstu *Student Ivo polaže PZUIS; Student Ivo mora samo položiti PZUIS prije nego doktorira; Student Ivo može doktorirati* možemo izdvojiti **premise** *Student Ivo polaže PZUIS;* i *Student Ivo mora položiti PZUIS prije nego doktorira* te **zaključak** *Student Ivo može doktorirati* te izvesti da iz premisa slijedi zaključak modus ponens pravilom zaključivanja. Analiza uključuje proučavanje valjanosti zaključivanja te kvalitete premisa i zaključka.

Razvojem softvera za argumentaciju, olakšala se i analiza argumentacije. **Araucaria** je najpopularniji softver za analizu argumentaciju s 10000 korisnika iz preko 80 zemalja između 2001. i 2010. godine (Reed et al., 2017). Araucariom se mogu

1. izrađivati dijagrami argumenata iz tekstnih datoteka,
2. uređivati korištene argumentacijske modele i
3. pohranjivati argumente (više u poglavlju 3)

Omogućavanjem uređivanja argumentacijskih modela, Araucaria je prvi alat agnostičan na model argumentacije pretpostavljajući Toulminov model argumenta, ali zadržavajući kompatibilnost s Wigmoreovim (Wigmore et al., 2016) i Freemanovim (Freeman, 1991) modelima. Zbog svoje kompatibilnosti s više različitih argumentacijskih modela, Araucia se smatra realizacijskim začetnikom argument weba. Araucaria je danas<sup>1</sup> dostupna online pod imenom OVA (engl. *Online Visualization of Argument*).



Slika 4.1: Analiza mogu li računala razmišljati na DebateGraphu

## 4.1. Kolaborativna analiza

Postavljanje sustava za analizu argumenata online olakšalo je suradnju na analizi argumentacije koja je neophodna u slučaju kompleksnije argumentacije. Uz OVA alat, pojavili su se i drugi online alati za analizu argumentacije. DebateGraph<sup>2</sup> omogućava korisnicima hijerarhijsku analizu teme kroz grafove. Primjer na slici 4.1 prikazuje analizu *Mogu li računala razmišljati* u sustavu DebateGraph. Svaki čvor u grafu moguće je dodatno otvoriti i zasebno istražiti.

Kolaborativna analiza argumentacije u stvarnom vremenu moguća je uz alat škotske grupe znanstvenika AnalysisWall<sup>3</sup>. Uz AnalysisWall moguće je analizirati debatu, prepoznavati i povezivati argumente koristeći video zaslon na dodir, kao što je učinjeno na slici 4.2.

<sup>1</sup>7.1.2018. dostupna na URL-u <http://ova.arg-tech.org/>

<sup>2</sup>7.1.2018. dostupna na <https://www.debategraph.com>

<sup>3</sup>7.1.2018. više o projektu na <http://www.arg-tech.org/index.php/projects/argument-analysis-wall/>



**Slika 4.2:** Anotacija pomoću AnalysisWall-a

## 5. Evaluacija argumentacije

Mogućnost analize nestrukturiranih rasprava pomoću računalnih alata uvelike pomaže stručnjacima prilikom vrednovanja argumentacije. Stručnjaci obrazovani u području argumentacije znaju prepoznati koje su prikladne logičke veze između tvrdnji. Stručnjaci koji se bave argumentacijom *neće reći da iz tvrdnje Student Ivo danas nije došao na fakultet slijedi tvrdnja Student Ivo nije položio niti jedan predmet, već će indukcijom zaključiti* kako *Student Ivo danas nije bio na fakultetu*. No, ono što intrigira je mogućnost računalne evaluacije argumentacije. Automatizacijom izvođenja zaključaka u argumentaciji, moguće je doći do novih, neizrečenih tvrdnji te dobiti **skupove prihvatljivih argumenata** (engl. *acceptable arguments*). Za gore naveden primjer argumenti *Student Ivo danas nije došao na fakultet* i *Student Ivo danas nije bio na fakultetu* formiraju skup prihvatljivih argumenata jer su povezani zaključivanjem. Računalni sustavi za zaključivanje i evaluaciju argumenata u argumentaciji zovu se **argumentacijska radna okruženja** (engl. *argumentation framework*), AF.

Kao što je spomenuto u odjeljku 2.3, Dung (1995) se prvi bavio evaluacijom argumenata analizirajući prihvatljivost argumenata kroz nededuktivnu (primjerice induktivnu ili oborivu) logiku. Dungov argumentacijski model povezuje argumente **binarnim relacijama napadanja** pravilima nededuktivne logike. Koristi se nededuktivna logika zbog njene učestalosti u svakodnevnom govoru. U svakodnevnom razgovoru ljudi su skloni analogijama i induktivnom zaključivanju, primjerice izjava *Student Ivo je položio skoro sve predmete na doktorskom studiju* ima za logičku posljedicu *Student Ivo će položiti PZUIS, predmet na doktorskom studiju*. Dakako, ova izjava nije valjana prema formalnoj logici, oblikovana je induktivnim argumentom analogije (Juthe, 2005). Dung ne pretpostavlja strukturu argumenata, tipove argumenata kao ni vrste binarnih relacija, što čini njegov model iznimno **apstraktnim**. Nakon Dungova rada, razvijena su argumentativna radna okruženja koja su proširila Dungov model. Dungov apstraktni model se specijalizirao strukturiranjem argumenata te specijalizacijom vrsta binarnih relacija između argumenata (Vreeswijk, 1993).

Najkorištenije argumentacijsko radno okruženje je **ASPIC+** okruženje<sup>1</sup>. Prvi računalni sustav, implementacija argumentacijskog radnog okruženja, je *Dung-O-Matic* (Snaith et al., 2010), no taj sustav nije kompatibilan s ostalim formatima zapisa argumenta. *TOAST* (engl. *The Online Argument Structures Tool*) sustav razvijen je s funkcionalnostima *Dung-O-Matic*-a, ali je kompatibilan s popularnim AIF formatom (poglavlje 3), kao i sa ASPIC+ argumentacijskim radnim okruženjem. Postoje još brojni drugi sustavi kao što su *Tweety* (Thimm, 2014) koji pokušava mapirati AIF strukturu u logički program (slično kao i *ASPARTIX* (Egly et al., 2008)), *ArgSemSAT* (Cerutti et al., 2014) koji koristi poznatu SAT (Moskewicz et al., 2001) tehniku kako bi odredio prihvatljivost argumenta u argumentacijskom radnom okruženju. U sljedećim odjeljcima поближе će se opisati ASPIC+ radno okruženje i ASPIC+ računalna implementacija *TOAST* sustav.

## 5.1. ASPIC+

ASPIC+ je radno okruženje nastalo iz Dungovog modela, s razlikom da ASPIC+ opisuje logička pravila zaključivanja, u što Dungov model ne ulazi detaljno. Grubo gledano, ulaz ASPIC+ sustavu su skup argumenata i relacije između tih argumenata koje ASPIC+ grupira u skupove prihvatljivih argumenata. Korisnici potom mogu raditi upite nad radnom okolinom kojima primjerice mogu provjeravati koji su argumenti međusobno prihvatljivi.

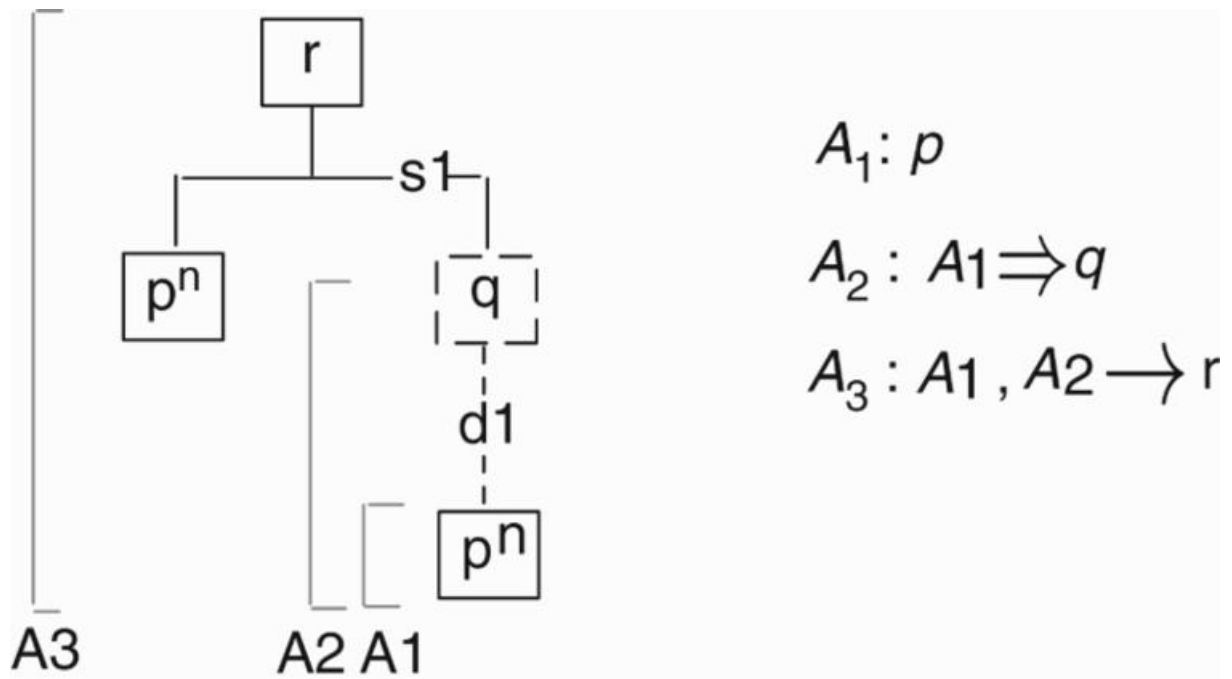
Definicija argumentacijskog radnog okruženja ( $AF$ ) istovjetna je za Dungov i ASPIC+ model.  $AF$  je par  $(A, D)$  gdje je  $D \subseteq A \times A$  binarna relacija napada (engl. *attack*) između argumenata  $A$ . Kažemo da  $A$  napada  $B$  ukoliko  $A$  napada  $B$  i  $B$  ne napada  $A$ .  $A$  predstavlja skupove argumenata, proširenja (engl. *extensions*) koji su koherentni i brane se od napada. Za svaki  $X \in A$ ,  $X$  je prihvatljiv s obzirom na  $S \subseteq A$  akko  $\forall Y$  takav da  $(Y, X) \in C$  implicira  $\exists Z \in S$  takav da  $(Z, Y) \in D$ . Ako je  $S \subseteq A$  bez konflikta (engl. *conflict free*), što znači da ne postoje  $A \in S, B \in S$  takvi da postoji  $(A, B) \in D$  onda za  $S$  vrijedi jedno od idućeg:

- $S$  je *prihvatljivo* proširenje akko  $X \in S$  implicira prihvatljiv  $X$  prema  $S$ ;
- $S$  je *kompletno* proširenje akko  $X \in S$  akko je  $X$  prihvatljiv prema  $S$ ;

---

<sup>1</sup>ASPIC+ okruženje razvilo se iz jednostavnog proširenja Dungovog modela – ASPIC okruženja (engl. *Argumentation service platform with integrated components*) nastalog u sklopu Europskog projekta ASPIC po kojem je dobio (pomalo neprikladno) ime. ASPIC se značajno razvio kroz brojna proširenja, od kojih je najznačajnije ASPIC+.





**Slika 5.1:** Primjer argumenta u ASPIC+ sustavu

- S je *preferirano* proširenje akko je uvrštenje skupa maksimalno potpuno proširenje;
- S je *osnovno* proširenje akko je uvrštenje skupa minimalno potpuno proširenje;
- i
- S je *stabilno* proširenje akko je preferirano i  $\forall Y \notin S, \exists X \in S$  takav da  $(X, Y) \in D$

Za  $T \in \{\text{kompletan, preferiran, osnovan, stabilan}\}$ ,  $X$  je *skeptično* opravdan ukoliko  $X$  pripada barem jednom  $T$  proširenju (Modgil i Prakken, 2014).

Za korištenje ASPIC+-a potrebno je odabrati negacijsko-zatvoreni logički jezik  $L$ , dva skupa strogih (engl. *strict*) i oborivih (engl. *defeasible*) pravila zaključivanja. Osim pravila, potrebno je specificirati bazu znanju (engl. *knowledge base*) koja sadrži dostupne informacije u obliku premisa. U bazi znanja razlikujemo obične, aksiome, te oborive premise, koje je moguće napadati. Baza znanja, pravila zaključivanja i logičkog jezika zajedno čine **argumentacijsku teoriju** (engl. *argumentation theory*). ASPIC+ radi nad argumentacijskom teorijom i omogućava upite nad njom. Nad zadanim argumentom moguće je dobiti premise koje ga čine kroz upit *Prem*, zaključak upitom *Conc*, njegove podargumente upitom *Sub*, oboriva pravila argumenta upitom *DefRules* te zadnje pravilo zaključivanja upitom *TopRule*.

Primjerice, baza podataka se sadrži od:  $p, q, r, s, t, u, v, w, x, d_1, d_2, d_3, d_4, d_5, d_6$  i

njihovih negacija, gdje je skup strogih pravila  $R_s = \{s_1, s_2\}$ , a skup oborivih pravila  $R_d = \{d_1, d_2, d_3, d_4, d_5, d_6\}$ . Sama pravila su:

$$\begin{array}{lll} d_1 : p \Rightarrow q & d_4 : u \Rightarrow v & s_1 : p, q \rightarrow r \\ d_2 : s \Rightarrow t & d_5 : v, x \Rightarrow \neg t & s_2 : v \rightarrow \neg s \\ d_3 : t \Rightarrow \neg d_1 & d_6 : s \Rightarrow \neg p & \end{array}$$

Funkcija  $n$  pridjeljuje oborivom pravilu  $d_i$  formulu  $d_i$ , dakle vrijedi:  $n(d_i) = d_i$ , prikazano na primjeru:  $n(p \Rightarrow q) = d_1$ . Ovako definiran argument prikazan je slikom 5.1: premise su na dnu slike, a zaključak na vrhu. Premise su označene eksponentom, a oborive premise i zaključivanja isprekidanom linijom. Sada je i upitima moguće doći do elemenata argumenata  $A_1, A_2$  i  $A_3$ . Tako je  $Prem(A_3) = \{p\}$ ,  $DefRules(A_3) = \{d_1\}$ . Još primjera i pojašnjenja ASPIC+ radne okoline moguće je pronaći u (Modgil i Prakken, 2014).

## 5.2. TOAST

TOAST (engl. *The Online Argument Structures Tool*) je implementacija ASPIC+ (Modgil i Prakken, 2014) argumentacijskog okruženja. TOAST je web servis u koji je moguće unijeti argumente i evaluirati ih. Korisnik TOAST-a može unositi izjave u bazu znanja (engl. *knowledge base*) i definirati vlastita pravila (engl. *rules*) (prema ASPIC+-u). U bazi znanja razlikujemo aksiome, premise i pretpostavke. Osim standardnih logičkih poveznica koje je moguće unositi između tvrdnji u bazi znanja, između premisa i pretpostavki moguće je definirati relacije preferencije. Pravila se unose u formatu:

[jedinstvena oznaka] {lista ancedensa} {implikacija} {konsekvens};

primjerice pravilo s dvije tvrdnje iz kojih logičkom implikacijom slijedi konsekvens:

[p] {Student Ivo studira na FER—u, FER je u Zagrebu} {→} {Student Ivo studira u Zagrebu};

Osim relacija klasične logike, moguće je preferirati tvrdnje (PA-čvor) u formatu:

[jedinstvena oznaka pravila a] < [jedinstvena oznaka preferiranog pravila b]

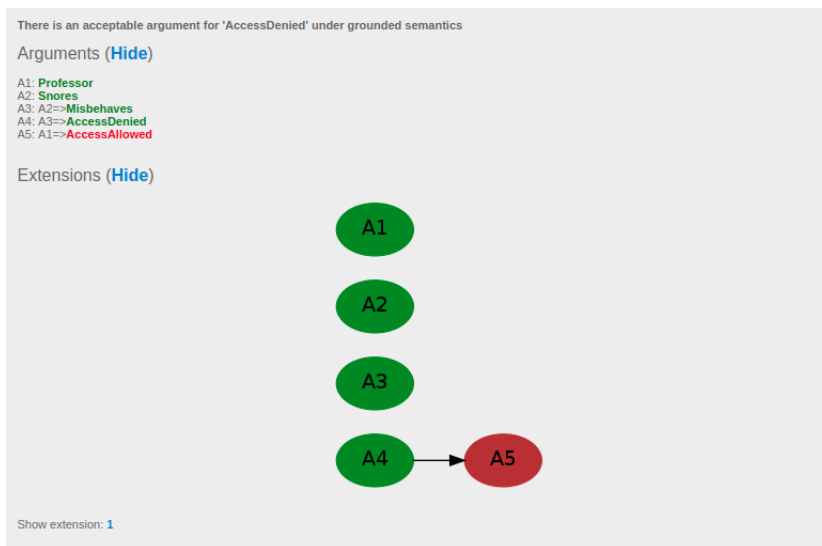
Primjeri korištenja TOAST-a vidljivi su na slikama 5.2 i 5.3. TOAST je u potpunosti integriran s AIFdb (više u odjeljku 3.4), stoga je moguće argument kreiran u AIFdb-u evaluirati kroz TOAST.

<b>Axioms:</b>	<b>Premises:</b> Snores; Professor;	<b>Assumptions:</b>	<b>Preferences:</b> Snores < Professor;
<b>Rules:</b> [r1] Snores=>Misbehaves; [r2] Misbehaves=>AccessDenied; [r3] Professor=>AccessAllowed;	<b>Rule Preferences:</b> [r1] < [r2]; [r1] < [r3]; [r3] < [r2];	<b>Contrariness:</b> AccessDenied-AccessAllowed	<b>Preference principle:</b> <input checked="" type="radio"/> Last link <input type="radio"/> Weakest link <b>Evaluation engine:</b> Dung-O-Matic ▾ <b>Semantics:</b> Grounded ▾ <input checked="" type="checkbox"/> Show islands in abstract framework?

[Add rule labels](#) | ☐ Close under transposition?

Query:   [Load an example](#) | [Help](#) | [API](#)

**Slika 5.2:** Unos pravila i baze znanja u TOAST



**Slika 5.3:** Evaluacija prihvatljivosti *AccessDenied* upita

## 6. Zaključak

U ovom seminarskom radu definirani su argumenti, argumentacija te je opisano kako su predstavljeni argumenti u računalnim sustavima u sklopu računalne argumentacije. Detaljnije je opisana računalna argumentacija te su obrazložene njene grane: predstavljanje argumentacije, računalna analiza argumenta te evaluacija argumentacije. Pokazano je na primjerima koje su prednosti korištenja računala pri analizi ili evaluaciji argumenata. Posljednja grana prema ostvarivanju potpunog argument weba je dubinska analiza argumentacije.

Dubinska analiza argumentacije, najmlađa grana računalne argumentacije pokušava iz nestrukturiranog teksta prepoznati dijelove argumenta (premise, zaključke) i logički povezati argumente. To se smatra izuzetno teškim problemom u sklopu računalne semantike, jer računalno mora *shvatiti* bit i implikacije argumenta. Održana su već četiri međunarodna znanstvena skupa *ArgMining* (2014. – 2017.) posvećenih novim metodama ekstrakcije argumenata iz zapisnika pravnih slučajeva, eseja ili online rasprava.

Razvoj semantičkog weba, strukturiranje online sadržaja kroz ontologije i baze znanja, inspiriralo je ekvivalentan pokret argument weba među znanstvenicima koji se bave argumentacijom. Računalna argumentacija aktivno radi na razvoju računalnih sustava za pomoć ljudima pri donošenju odluka s predumišljajem (engl. *informed decisions*), a jednog dana – možda niti čovjek neće uvijek biti potreban.

## 7. Literatura

- Philippe Besnard i Anthony Hunter. *Elements of argumentation*, svezak 47. MIT press Cambridge, 2008.
- J Anthony Blair i Ralph H Johnson. Informal logic: An overview. *Informal logic*, 20 (2), 2000.
- Federico Cerutti, Massimiliano Giacomin, i Mauro Vallati. Argsemsat: Solving argumentation problems using sat. *COMMA*, 266:455–456, 2014.
- Carlos Chesñevar, Sanjay Modgil, Iyad Rahwan, Chris Reed, Guillermo Simari, Matthew South, Gerard Vreeswijk, Steven Willmott, et al. Towards an argument interchange format. *The Knowledge Engineering Review*, 21(4):293–316, 2006.
- Phan Minh Dung. On the acceptability of arguments and its fundamental role in non-monotonic reasoning, logic programming and n-person games. *Artificial intelligence*, 77(2):321–357, 1995.
- Uwe Egly, Sarah Gaggl, i Stefan Woltran. Aspartix: Implementing argumentation frameworks using answer-set programming. *Logic Programming*, stranice 734–738, 2008.
- Robert Fogelin. The logic of deep disagreements. *Informal logic*, 7(1), 1985.
- James B Freeman. *Dialectics and the macrostructure of arguments: A theory of argument structure*, svezak 10. Walter de Gruyter, 1991.
- Thomas F Gordon i Douglas Walton. The carneades argumentation framework—using presumptions and exceptions to model critical questions. U *6th computational models of natural argument workshop (CMNA), European conference on artificial intelligence (ECAI), Italy*, stranice 5–13, 2006.
- André Juthe. Argument by analogy. *Argumentation*, 19(1):1–27, 2005.

- John Lawrence i Chris Reed. Aifdb corpora. U *COMMA*, stranice 465–466, 2014.
- John Lawrence, Floris Bex, Chris Reed, i Mark Snaith. Aifdb: Infrastructure for the argument web. U *COMMA*, stranice 515–516, 2012.
- Sanjay Modgil i Henry Prakken. The aspic+ framework for structured argumentation: a tutorial. *Argument & Computation*, 5(1):31–62, 2014.
- Matthew W Moskwicz, Conor F Madigan, Ying Zhao, Lintao Zhang, i Sharad Malik. Chaff: Engineering an efficient sat solver. U *Proceedings of the 38th annual Design Automation Conference*, stranice 530–535. ACM, 2001.
- Nir Oren, Timothy J Norman, i Alun Preece. Subjective logic and arguing with evidence. *Artificial Intelligence*, 171(10):838–854, 2007.
- Ramon Prudencio i S Toledo. Visualizing argumentation: Software tools for collaborative and educational sense-making, 2005.
- Chris Reed i Glenn Rowe. Araucaria: Software for argument analysis, diagramming and representation. *International Journal on Artificial Intelligence Tools*, 13(04): 961–979, 2004.
- Chris Reed, Simon Wells, Joseph Devereux, i Glenn Rowe. Aif+: Dialogue in the argument interchange format. *FRONTIERS IN ARTIFICIAL INTELLIGENCE AND APPLICATIONS*, 172:311, 2008.
- Chris Reed, Katarzyna Budzynska, Rory Duthie, Mathilde Janier, Barbara Konat, John Lawrence, Alison Pease, i Mark Snaith. The argument web: An online ecosystem of tools, systems and services for argumentation. *Philosophy and Technology*, 30(2): 137–160, 2017.
- Olivier Rieppel. Homology and logical fallacy. *Journal of evolutionary biology*, 5(4): 701–715, 1992.
- Oliver Scheuer, Frank Loll, Niels Pinkwart, i Bruce M McLaren. Computer-supported argumentation: A review of the state of the art. *International Journal of Computer-Supported Collaborative Learning*, 5(1):43–102, 2010.
- Mark Snaith, Joseph Devereux, John Lawrence, i Chris Reed. Pipelining argumentation technologies. U *COMMA*, stranice 447–453, 2010.

- Matthias Thimm. Tweety: A comprehensive collection of java libraries for logical aspects of artificial intelligence and knowledge representation. *KR*, 14:528–537, 2014.
- Stephen Toulmin. The uses of argument. 1958. *Cambridge: Cambridge UP*, 1974.
- Tim Van Gelder. The rationale for rationale™. *Law, Probability & Risk*, 6(1-4):23–42, 2007.
- Gerard Vreeswijk. The feasibility of defeat in defeasible reasoning. U *Diamonds and Defaults*, stranice 359–380. Springer, 1993.
- D. Walton. Argument mining by applying argumentation schemes. *Studies in Logic*, 4 (1):2011, 2012.
- Douglas N Walton. What is reasoning? what is an argument? *The Journal of Philosophy*, 87(8):399–419, 1990.
- Xiao Hang Wang, D Qing Zhang, Tao Gu, i Hung Keng Pung. Ontology based context modeling and reasoning using owl. U *Pervasive Computing and Communications Workshops, 2004. Proceedings of the Second IEEE Annual Conference on*, stranice 18–22. Ieee, 2004.
- John Henry Wigmore et al. *Wigmore on evidence*. JSTOR, 2016.