

# Homework 9

CS 411 - Artificial Intelligence I - Fall 2019

Matteo Corain 650088272

November 7, 2019

## 1 Question 1

### 1.1 Syntax

The syntax of a logic defines how well-formed sentences should be structured. In propositional logic, valid sentences belong to one of the two categories:

- *Atomic sentences*: they represent statements that can be either true or false (depending on the particular model we are considering) and they are represented by simple propositional symbols (usually, an uppercase letter);
- *Complex sentences*: they are constructed from a number of atomic statements, which are linked through logical connectives (and possibly parentheses to define the order of the operations); in propositional logic, five connectives are defined:
  - *Negation*:  $\neg A$ ;
  - *Conjunction*:  $A \wedge B$ ;
  - *Disjunction*:  $A \vee B$ ;
  - *Implication*:  $A \Rightarrow B$ ;
  - *Biconditional*:  $A \Leftrightarrow B$ .

Where  $A$  and  $B$  represent arbitrary sentences (possibly complex as well).

### 1.2 Semantics

The semantics of a logic define the set of rules that are used to infer the truth value of complex sentences (the truth value of atomic sentences is fixed, given a certain model). In fact, in propositional logic each sentence has to have an associated truth value: this is indeed a strong limitation, since many sentences that may be expressed in natural language are not semantically valid in propositional logic since they do not have an associated truth value (e.g. questions, imperatives).

The semantic rules defined for propositional logic include the following:

- *Negation*:  $\neg A$  is true if and only if  $A$  is false in the considered model;

- *Conjunction*:  $A \wedge B$  is true if and only if  $A$  and  $B$  are both true in the considered model;
- *Disjunction*:  $A \vee B$  is true if at least one among  $A$  and  $B$  is true in the considered model;
- *Implication*:  $A \Rightarrow B$  is true if and only if  $A$  is false or  $B$  is true in the considered model;
- *Biconditional*:  $A \Leftrightarrow B$  is true if  $A$  and  $B$  are both true or both false in the considered model, or equivalently if both  $A \Rightarrow B$  and  $B \Rightarrow A$  are true in the considered model.

### 1.3 Modus ponens

Modus ponens is an inference procedure that allows to infer sentences from implications in the knowledge base. Formally, the modus ponens procedure states that, if implication  $A \Rightarrow B$  holds non-vacuously (that is to say,  $A$  is also true), then we can infer that  $B$  is true as well. In symbols, this inference procedure is written as:

$$\frac{A, A \Rightarrow B}{B}$$

The modus ponens inference technique is *sound*, that is to say that the set of models of the inferred (and thus entailed) sentence  $B$  is a superset of the one of the knowledge base:  $M(B) \supseteq M(KB)$ . If we consider only propositions in the described form, then this procedure is also *complete*; however, it is not complete in general: for example, this procedure cannot derive any proposition containing disjunction operations.

### 1.4 Monotonicity

Monotonicity is a property of logical systems which formalizes the concept that the set of entailed sentences can only increase as information is added to the knowledge base. Formally, if we have a knowledge base  $KB$  and a proposition  $\alpha$  such that  $KB \models \alpha$ , a system satisfies the property of monotonicity if, when we learn an additional proposition  $\beta$ , then we have also that:

$$(KB \wedge \beta) \models \alpha$$

Intuitively, the property of monotonicity states that, whenever we get to learn a new fact about the world, all the conclusions we were able to draw before knowing that facts remain true as well; in other words, the conclusion of the rule must follow regardless of what else is in the knowledge base. In terms of sets of models, monotonicity can be expressed as:

$$M(KB \wedge \beta) \subseteq M(KB) \subseteq M(\alpha)$$

Propositional logic satisfies the monotonicity property.

### 1.5 Proof by contradiction

Proof by contradiction is a technique that can be used to entail certain propositions by proving that it is impossible (i.e. it results in a contradiction of the axioms that are part of the considered knowledge base) for this entailment not to hold. Formally, if we have a knowledge base  $KB$  and a

sentence  $\alpha$  and we are able to prove that  $KB \wedge \neg\alpha$  cannot be true in the model we are considering, then proposition  $\alpha$  can be entailed from the knowledge base:  $KB \models \alpha$ .

Proof by contradiction is an essential tool that can be used to turn the resolution process complete (in general, resolution is a sound but not complete procedure): for this reason, we say that inference by resolution is *contradiction-complete*.

## 2 Question 2

Let us introduce the following atomic propositions:

- $A$ : “Ana eats”;
- $B$ : “Bret eats”;
- $C$ : “Charles eats”;
- $D$ : “Derek eats”;
- $E$ : “Earl eats”;
- $F$ : “Fred eats”.

Using those symbols, it is possible to translate the given sentences in formal propositional sentences in the following way:

- “If Ana eats, Bret eats”:  $A \Rightarrow B$ ;
- “Charles eats and Derek doesn’t eat”:  $C \wedge \neg D$ ;
- “Bret doesn’t eat”:  $\neg B$ ;
- “If Derek doesn’t eat, at least one among Ana, Earl and Fred eats”:  $\neg D \Rightarrow (A \vee E \vee F)$ ;
- “If at least one of Charles and Gary eats, Earl doesn’t eat”:  $(C \vee G) \Rightarrow \neg E$ .

## 3 Question 3

In order to prove that  $F$  is true, the following steps may be taken:

- From sentence 3, we know that  $\neg B$  is true, which means that  $B$  is necessarily false ( $\neg False = True$ );
- From sentence 2, we know that  $C \wedge \neg D$  is true, which means that  $C$  is necessarily true and  $D$  is necessarily false ( $True \wedge \neg False = True$ );
- From sentence 1, we know that  $A \Rightarrow B$  is true; since  $B$  is false, then  $A$  is necessarily false ( $False \Rightarrow False = True$ );

- From sentence 5, we know that  $(C \vee G) \Rightarrow \neg E$  is true; since  $C$  is true, then  $C \vee G$  is true independently of the value of  $G$ ; therefore,  $\neg E$  has to be true ( $True \Rightarrow True = True$ ) and, consequently,  $E$  is false;
- Finally, from sentence 4, we know that  $\neg D \Rightarrow (A \vee E \vee F)$  is true; since  $D$  is false, then  $\neg D$  is true, which means that  $A \vee E \vee F$  needs to be true as well ( $True \Rightarrow True = True$ ); however, since  $A$  and  $E$  are both false, then necessarily  $F$  is true, which concludes the proof.

The truth value of the atomic propositions is summarized in table 1.

Proposition	Value	From sentences
$A$	False	1,3
$B$	False	3
$C$	True	2
$D$	False	2
$E$	False	2,5
$F$	True	1,2,3,4,5
$G$	Undetermined	-

Table 1: Truth status of the atomic propositions