Logiques et représentations des connaissances

Charles Vin

S1-2022

1 Introduction à la logique des propositions et des prédicats du premier ordre

Définition 1.1. Quelques définitions
- Expression : une description logique. Elle est vrais ou fausse
- Meaning : le sens qui relis l'expression vers sa référence
- Référence : représente le sens

Le langage des propositions :
- Atome
- Connecteur :
  - Binary : $\neg$ = "et", $\wedge$ = "Ou" $\vee$ = négation
  - DIAPO

Table de vérité :
- Un ligne = une "interprétation"

Une formule est
- Satisfiable : si vrais dans une interprétation
- Valide : si vrais dans toute les interprétation
- Unsatisfiable : faux sur toutes les interprétations

On peut avoir des fonctions qui représente une expression logique.
- Un terme représente une fonction qui renvoie un paramètre ou le tuple de paramètre d'une fonction.
- Un atome est une fonction qui renvoie un booléen.

En d'autre mot ça semble être en fonction de ce que la fonction renvoie un booléen ou un paramètre.

Variable est :
- Libre : si elle est "muette", c'est à dire que si on la change elle ne change pas le résultat
- Liée : inverse je crois?

Diapo interpretation : rien compris

A formula is "**valid**" if it is true in **all** the interpretation of **all** domains.

Exemple 1.1.
- Valide
- DIAPO

Un modèle est un couple $\mathcal{M} =< \mathcal{D}, i >$. Une valuation est une fonction $v : \mathcal{V} \to \mathcal{D}$. $I_m(F)$ is the truth value. Y'a des propriétés de la truth value dans le DIAPO.

A est une conséquence sémantique de B si A est vrais pour toutes les interpretations où B est vrais i.e. for all models if $I_{m\mathcal{V}}(B)$ then $I_{m\mathcal{V}}(A)$. Exemple $A \to B$ est une conséquence sémantique. Je crois que le symbole c'est $B \models A$ == A conséquence logique de B.

Définition 1.2. Un système formel est composé d'un langage formel, un ensemble d'axiom et de règle d'inférence.

Exemple 1.2. Exemple dans le fiapo avec le système formel de Hilbert.

$\vdash$ représente une dérivation. On peut prouver A à partir de B.

Définition 1.3 (Théorème). Any formula which is derived from the axoms by iteratively applying inference rules is a **theorem**.
Notation : $\vdash A$ means A is a theorem

Exemple 1.3. Exemple : $\vdash (A \to A)$

Preuve : Voir diapo c'est drôle □

Définition 1.4 (Démonstration)**.** A **proof** of a theorem A is a finite sequence of formulae $F_0, ..., F_n$ such that .... DIAPO mais osef un peu

Définition 1.5 (Symbolic system)**.**
— Consistency : Each description of the symbolic system corresponds to an object in the reality i.e. DIAPO
— Completeness : each object of the reality can be described in the symbolic system $\forall A$ if $\models A$ then $\vdash A$

1.1 Automatic theorem proving

1.1.1 Tableau method

1. Normalisation : Transformation into **NNF - Negative Normal Form** : **The negations occurs only before atomic propositions**. i.e il faut développer les négations au max.
2. Build a tableau :
   — Root : The formula under NNF.
   — Build successors of T using two rules $R_\wedge$ et $R_\vee$ VOIR LE DIAPO POUR LEUR DEFINITIONS
   — On arrête lorsque l'on ne peux plus appliquer les règles.

Définition 1.6. Un tableau peut être
— Contradictory
— ... DIAPO

Exemple 1.4 (de la methode)**.** DIAPO MDR OU YOUTUBE PLUTOT

L'avantage de cette méthode par rapport au tableau de vérité c'est qu'on a pas besoin de faire tous les cas possibles.

Généralisation Si je comprend bien on peut prendre des raccourcis avec des nouvelles règles. Pour la règles α on met dans le même tableau. Pour les règles β on met dans deux tableaux différents. **Apprendre les tableau de règles**

1.1.2 Resolution in propositional logic

Définition 1.7 (Une clause)**. Ca tombe à l'exam d'après le prof**
— Un literal is either an atom or its negation
— Une clause is a disjunction of literals

Remarque. A clause is a logical entailment (implication) because $(\neg A \vee B)$ is equivalent to $(A \supset B)$

Exemple 1.5. $even(X) \supset odd(successor(X))$

Théorème 1.1. *Any claused formula (sans variable libre) F can be transformed into a logically equivalent conjunction of clauses*

Exemple 1.6.

Définition 1.8 (Règle de résolution)**.** S'applique uniquement sur les clauses. Pour prouver que $S \models A$ il suffit de montrer que $S \cup [negA]$ est vide. Voir les exemples du DIAPO

1.1.3 Unification

Définition 1.9 (Substitution)**.** A **substitution** is characterized by a infinite set of "

On peut **composer** plusieurs substitution

Définition 1.10 (term instance)**.**

1.1.4 Resolution in First Oder Logic