# Introduction to AI: Propositional Resolution

François Jacquenet

Laboratoire Hubert Curien
Université de Saint-Etienne
Francois.Jacquenet@univ-st-etienne.fr

# Outline

# Definitions

## Literal

A literal is either an atomic sentence or a negation of an atomic sentence

Example: if $p$ is a logical constant, the following sentences are both literals: $p$ and $\neg p$

## Clausal sentence

A clausal sentence is either a literal or a disjunction of literals.

Example: if $p$ and $q$ are logical constants, then the following are clausal sentences: $p$ and $\neg p$ and $\neg p \vee q$

## Clause

A clause is the set of literals in a clausal sentence

Example: the following sets are the clauses corresponding to the clausal sentences above: $\{p\}$ and $\{\neg p\}$ and $\{\neg p, q\}$

# Transformation to Clausal Form

Transforming implications

$$\begin{array}{lcl} \Phi \Rightarrow \Psi & \rightarrow & \neg\Phi \vee \Psi \\ \Phi \Leftarrow \Psi & \rightarrow & \Phi \vee \neg\Psi \\ \Phi \Leftrightarrow \Psi & \rightarrow & (\neg\Phi \vee \Psi) \wedge (\Phi \vee \neg\Psi) \end{array}$$

# Transformation to Clausal Form

Transforming negations

$$
\begin{array}{lcl}
\neg\neg\Phi & \rightarrow & \Phi \\
\neg(\Phi \wedge \Psi) & \rightarrow & \neg\Phi \vee \neg\Psi \\
\neg(\Phi \vee \Psi) & \rightarrow & \neg\Phi \wedge \neg\Psi
\end{array}
$$

Transformation 2 and 3 are called De Morgan's laws

# Transformation to Clausal Form

Distribution rules

$$
\begin{array}{lcl}
\Phi \vee (\Psi \wedge \Gamma) & \rightarrow & (\Phi \vee \Psi) \wedge (\Phi \vee \Gamma) \\
(\Phi \wedge \Psi) \vee \Gamma & \rightarrow & (\Phi \vee \Gamma) \wedge (\Psi \vee \Gamma) \\
\Phi \vee (\Phi_1 \vee \cdots \vee \Phi_n) & \rightarrow & \Phi \vee \Phi_1 \vee \cdots \vee \Phi_n \\
(\Phi_1 \vee \cdots \vee \Phi_n) \vee \Phi & \rightarrow & \Phi_1 \vee \cdots \vee \Phi_n \vee \Phi \\
\Phi \wedge (\Phi_1 \wedge \cdots \wedge \Phi_n) & \rightarrow & \Phi \wedge \Phi_1 \wedge \cdots \wedge \Phi_n \\
(\Phi_1 \wedge \cdots \wedge \Phi_n) \wedge \Phi & \rightarrow & \Phi_1 \wedge \cdots \wedge \Phi_n \wedge \Phi
\end{array}
$$

# Transformation to Clausal Form

Generating clauses

$$
\begin{array}{lcl}
\Phi_1 \vee \cdots \vee \Phi_n & \rightarrow & \{\Phi_1, \ldots, \Phi_n\} \\
\Phi_1 \wedge \cdots \wedge \Phi_n & \rightarrow & \{\Phi_1\}, \ldots, \{\Phi_n\}
\end{array}
$$

# Example

Consider translating the compound sentence : $a \wedge (b \Rightarrow c)$
Observe what the 4 steps produce:

| Action | transformed sentence |
|---|---|
| Transform implication | $a \wedge (\neg b \vee c)$ |
| Transform negation | $a \wedge (\neg b \vee c)$ |
| Distribution | $a \wedge (\neg b \vee c)$ |
| Generating clauses | $\{a\}$ |
| | $\{\neg b, c\}$ |

Using those rules we build the Conjunctive Normal Form (CNF) of a logical sentence.
It's a conjunction of disjunctions

# Example

Consider translating the compound sentence : $\neg(a \wedge (b \Rightarrow c))$
Observe what the 4 steps produce:

| Action | transformed sentence |
|---|---|
| Transform implication | $\neg(a \wedge (\neg b \vee c))$ |
| Transform negation | $\neg a \vee \neg(\neg b \vee c)$ |
| | $\neg a \vee (\neg\neg b \wedge \neg c)$ |
| | $\neg a \vee (b \wedge \neg c)$ |
| Distribution | $(\neg a \vee b) \wedge (\neg a \vee \neg c)$ |
| Generating clauses | $\{\neg a, b\}$ |
| | $\{\neg a, \neg c\}$ |

# Exercice

Convert the following sentences to clausal form.

1. $p \wedge q \Rightarrow r \vee s$
2. $p \vee q \Rightarrow r \vee s$
3. $\neg(p \vee q \vee r)$
4. $\neg(p \wedge q \wedge r)$
5. $p \wedge q \Leftrightarrow r$

The main idea in the resolution principle is this one: given a clause containing a literal $\Gamma$ and another clause containing the literal $\neg\Gamma$, we can infer the clause consisting of all the literals of both clauses without the complementary pair.

Propositional Resolution also called Resolution Principle (this is an inference rule)

$$\frac{\begin{array}{c}\{\Phi_1, \ldots, \Gamma, \ldots, \Phi_n\} \\ \{\Psi_1, \ldots, \neg\Gamma, \ldots, \Psi_p\}\end{array}}{\{\Phi_1, \ldots, \Phi_n, \Psi_1, \ldots, \Psi_p\}}$$

# Examples

Example 1

$$\frac{\begin{array}{c}\{p, q, r\}\\ \{s, \neg q, t\}\end{array}}{\{p, r, s, t\}}$$

Example 2

$$\frac{\begin{array}{c}\{p, q\}\\ \{q, \neg p\}\end{array}}{\{q\}}$$

(remember a clause is a set so we have no duplicate)

# The empty clause

It may happen that we infer the empty clause

$$\frac{\begin{array}{c}\{p\}\\ \{\neg p\}\end{array}}{\{\}}$$

# Example

Consider again the sentences: If it's Monday then Mary loves Pat or Quincy, If Mary loves Pat then Mary loves Quincy.
Those sentences are formalized as:

1. $m \Rightarrow p \vee q$
2. $p \Rightarrow q$

1) may be transformed into the clausal form: $\{\neg m, p, q\}$
2) may be transformed into the clausal form: $\{\neg p, q\}$
Using the Resolution Principle :

$$\frac{\begin{array}{c}\{\neg m, p, q\}\\ \{\neg p, q\}\end{array}}{\{\neg m, q\}}$$

that is: $m \Rightarrow q$
which means: If it's Monday then Mary loves Quincy

# Reasoning with the Resolution Principle

Basics of reasoning with the Resolution Principle:

- start with premises
- apply the Resolution Principle to those premises
- apply the rule to the results of those applications
- and so forth until we get to our desired conclusion or we run out of things to do

A resolution derivation of a conclusion from a set of premises is a finite sequence of clauses terminating in the conclusion in which each clause is either a premise or the result of applying the Resolution Principle to earlier members of the sequence

# Example

Generally we can find resolution derivations of conclusions from premises

| | | |
|---|---|---|
| 1. | $\{\neg p, r\}$ | Premise |
| 2. | $\{\neg q, r\}$ | Premise |
| 3. | $\{p, q\}$ | Premise |
| 4. | $\{q, r\}$ | 1, 3 |
| 5. | $\{r\}$ | 2, 4 |

# Resolution is not generatively complete

BUT: it's important to understand that the resolution is not generatively complete

It is not possible to find resolution derivations for all clauses that are logically entailed by a set of premise clauses

Example: given $\{p\}$ and $\{q\}$ there is no derivation of $\{p, q\}$ despite the fact that $\{p, q\}$ is logically entailed by $\{p\}$ and $\{q\}$

| $p$ | $q$ | $p \vee q$ |
|---|---|---|
| 1 | 1 | 1 |
| 1 | 0 | 1 |
| 0 | 1 | 1 |
| 0 | 0 | 0 |

BUT: If a set $\Delta$ of clauses is unsatisfiable, then there is guaranteed to be a resolution derivation of the empty clause from $\Delta$. More generally, if a set $\Delta$ of Propositional Logic sentences is unsatisfiable, then there is guaranteed to be a resolution derivation of the empty clause from the clausal form of $\Delta$

Example: consider the clauses $\{p, q\}$, $\{p, \neg q\}$, $\{\neg p, q\}$, and $\{\neg p, \neg q\}$

| $p$ | $q$ | $p \vee q$ | $p \vee \neg q$ | $\neg p \vee q$ | $\neg p \vee \neg q$ |
|---|---|---|---|---|---|
| 1 | 1 | 1 | 1 | 1 | 0 |
| 1 | 0 | 1 | 1 | 0 | 1 |
| 0 | 1 | 1 | 0 | 1 | 1 |
| 0 | 0 | 0 | 1 | 1 | 1 |

We can see that there is no truth assignment that satisfies all four of these clauses

Now consider the resolution derivation

| | | |
|---|---|---|
| 1. | $\{p, q\}$ | Premise |
| 2. | $\{p, \neg q\}$ | Premise |
| 3. | $\{\neg p, q\}$ | Premise |
| 4. | $\{\neg p, \neg q\}$ | Premise |
| 5. | $\{p\}$ | 1, 2 |
| 6. | $\{\neg p\}$ | 3, 4 |
| 7. | $\{\}$ | 5, 6 |

We can use the relationship between unsatisfiability and logical entailment to produce a method for determining logical entailment

Recall that the Unsatisfiability Theorem introduced in the previous course tells that a set $\Delta$ of sentences logically entails a sentence $\Phi$ if and only if the set of sentences $\Delta \cup \{\neg\Phi\}$ is unsatisfiable

So, to determine logical entailment, all we need to do is to negate our goal, add it to our premises, and use Resolution to determine whether the resulting set is unsatisfiable

# Definitions

A resolution proof of a sentence $\Phi$ from a set $\Delta$ of sentences is a resolution derivation of the empty clause from the clausal form of $\Delta \cup \{\neg\Phi\}$.

A sentence $\Phi$ is provable from a set of sentences $\Delta$ by Propositional Resolution (written $\Delta \vdash \Phi$) if and only if there is a resolution proof of $\Phi$ from $\Delta$

# Example

Suppose we have $p$, $(p \Rightarrow q)$, $(p \Rightarrow q) \Rightarrow (q \Rightarrow r)$
and we want to prove $r$
We have to convert those sentences to clausal form, add $\neg r$ as a premise and then run a resolution proof

| | | |
|---|---|---|
| 1. | $\{p\}$ | Premise |
| 2. | $\{\neg p, q\}$ | Premise |
| 3. | $\{p, \neg q, r\}$ | Premise |
| 4. | $\{\neg q, r\}$ | Premise |
| 5. | $\{\neg r\}$ | Premise |
| 6. | $\{q\}$ | 1, 2 |
| 7. | $\{r\}$ | 4, 6 |
| 8. | $\{\}$ | 5, 7 |

# Proving the validity of a sentence

We can use resolution proof to prove a sentence is valid

Example: prove $(p \Rightarrow (q \Rightarrow p))$ is valid
We have to negate that sentence ( $\neg(p \Rightarrow (q \Rightarrow p))$ ) and convert it to clausal form

| Action | transformed sentence |
|---|---|
| Transform implication | $\neg(\neg p \vee (\neg q \vee p))$ |
| Transform negation | $\neg\neg p \wedge \neg(\neg q \vee p)$ |
| | $p \wedge (\neg\neg q \wedge \neg p)$ |
| | $p \wedge (q \wedge \neg p)$ |
| Distribution | $p \wedge q \wedge \neg p$ |
| Generating clauses | $\{p\}$ |
| | $\{q\}$ |
| | $\{\neg p\}$ |

# Proving the validity of a sentence

Now we run a resolution proof

| | | |
|---|---|---|
| 1. | $\{p\}$ | Premise |
| 2. | $\{q\}$ | Premise |
| 3. | $\{\neg p\}$ | Premise |
| 4. | $\{\}$ | 1, 3 |

We infer the empty clause
which means $\neg(p \Rightarrow (q \Rightarrow p))$ is unsatisfiable
and then $p \Rightarrow (q \Rightarrow p)$ is valid

# Exercices

1. Given the premises $(p \Rightarrow q)$ and $(r \Rightarrow s)$ use Propositional Resolution to prove the conclusion $(p \vee r \Rightarrow q \vee s)$
2. Prove that $\neg(p \vee q) \Leftrightarrow (\neg p \wedge \neg q)$ is valid

# The language (syntax)

- Propositional constants
- Compound sentences
  - built using 5 operators $(\neg, \wedge, \vee, \Rightarrow, \Leftrightarrow)$
  - precedences:
    $prec(\neg) > prec(\wedge) > prec(\vee) > prec(\Rightarrow) = prec(\Leftrightarrow)$
  - right associativity (in this course)

# The language (semantics)

Wen can assign the value True or False (1 or 0) to proposition constants and then, using truth table, compute the truth value of a compound sentence

Let's recap the truth table for $\neg$, $\wedge$, $\vee$, $\Rightarrow$ and $\Leftrightarrow$

# Evaluation and satisfaction

Evaluation: given an assignment, process the truth value of a compound sentence (propose an example)

Satisfaction: given one or more compound sentences, is there an assignment that makes this sentence true?

# Validity, Unsatisfiability and Contingency

What is a valid (set of) sentence(s)?

What is an unsatisfiable (set of) sentences?

What is a contingent (set of) sentences?

# Equivalence, Entailment and Consistency

Draw a table that shows a situation of equivalence

Draw a table that shows a situation of entailment but not equivalence

Draw a table that shows a situation of consistency but not neither equivalence or entailment

# Clausal form

What does CNF mean?

Give the steps of the process to convert a sentence to CNF

# Resolution principle and reasoning

Give the inference rule called resolution principle

What is a resolution proof?

Given some premises, how can we try to prove a conclusion?

How can we prove a sentence is valid?

How can we prove a sentence is unsatisfiable?

# Exercise

Consider the following statements: *If John didn't meet Harry last night, then Harry is the murderer or John is a liar. If Harry is not the murderer, then John didn't meet Harry last night and the crime hold after midnight. If the crime hold after midnight, then either Harry is the murderer or John is not a liar*

1. Formalize these statements using propositional logic
2. Building a truth table, prove that *Harry is the murderer*
3. Using resolution reasoning, prove that *Harry is the murderer*

Try to solve this problem before looking at the next slides!!!

# Exercise (correction)

First formalize the sentences:

- let *jh* represent the possibility that John met Harry last night
- let *hm* represent the possibility that Harry is murderer
- let *jl* represent the possibility that John is a liar
- let *cam* represent the possibility that the crime hold after midnight

Second translate each sentence into propositional logic sentences

- *"If John didn't meet Harry last night, then Harry is the murderer or John is a liar"* is converted to $(\neg jh \Rightarrow hm \vee jl)$
- *"If Harry is not the murderer, then John didn't meet Harry last night and the crime hold after midnight"* is converted to $(\neg hm \Rightarrow \neg jh \wedge cam)$
- *"If the crime hold after midnight, then either Harry is the murderer or John is not a liar"* is converted to $(cam \Rightarrow hm \vee \neg jl)$

And add $\{\neg hm\}$

# Exercise (correction)

| $jh$ | $hm$ | $jl$ | $cam$ | $\neg jh \Rightarrow hm \vee jl$ | $\neg hm \Rightarrow \neg jh \wedge cam$ | $cam \Rightarrow hm \vee \neg jl$ | $\neg hm$ |
|---|---|---|---|---|---|---|---|
| 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |
| 1 | 1 | 1 | 0 | 1 | 1 | 1 | 0 |
| 1 | 1 | 0 | 1 | 1 | 1 | 1 | 0 |
| 1 | 1 | 0 | 0 | 1 | 1 | 1 | 0 |
| 1 | 0 | 1 | 1 | 1 | 0 | 0 | 1 |
| 1 | 0 | 1 | 0 | 1 | 0 | 1 | 1 |
| 1 | 0 | 0 | 1 | 1 | 0 | 1 | 1 |
| 1 | 0 | 0 | 0 | 1 | 0 | 1 | 1 |
| 0 | 1 | 1 | 1 | 1 | 1 | 1 | 0 |
| 0 | 1 | 1 | 0 | 1 | 1 | 1 | 0 |
| 0 | 1 | 0 | 1 | 1 | 1 | 1 | 0 |
| 0 | 1 | 0 | 0 | 1 | 1 | 1 | 0 |
| 0 | 0 | 1 | 1 | 1 | 1 | 0 | 1 |
| 0 | 0 | 1 | 0 | 1 | 0 | 1 | 1 |
| 0 | 0 | 0 | 1 | 0 | 1 | 1 | 1 |
| 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 |

There is no truth assignment that satisfies both 4 sentences. The set of 4 sentences is unsatisfiable.

So the conclusion "Harry is the murderer" is true

# Exercise 2

Here are some informations about a simple world: *If Paul can not sleep then he will not pass his exam. When Jack wins to poker, he invites John and Mary for dinner. When Jack invites Mary for dinner, Paul is jealous and angry. If Jack invites John for dinner, Lucy cries or invites Mary for dinner. If Jack or Lucy invites Mary for dinner then Mary is happy. When Paul is angry he can't sleep. Jack wins to poker. Lucy doesn't cry*.

1. Model this universe using Propositional Logic, that is, provide judicious proposition constants and convert each sentence to proposition sentences.
2. Provide a resolution proof of: *Paul doesn't pass his exam and Mary is happy*.

# Exercise 3

Brown, Jones, and Smith are suspected of a crime. In front of a jury they testify as follows:

> Brown: "Jones is guilty and Smith is innocent."
>
> Jones: "If Brown is guilty then so is Smith."
>
> Smith: "I'm innocent and at least one of the others is guilty."

1) Model this universe using Propositional Logic, that is, provide three judicious proposition constants and convert those three testimonies to three proposition sentences.
2) Write a truth table for the three testimonies.
3) Use the above truth table to answer the following questions (explain your answer):

> (a) Are the three testimonies satisfiable?
>
> (b) The testimony of one of the suspects logically entails that of another. Say which one entails which one?
>
> (c) Assuming that everybody is innocent, who committed perjury?
>
> (d) Assuming that all testimonies are true, who is innocent and who is guilty?
>
> (e) Assuming that the innocent told the truth and the guilty told lies, who is innocent and who is guilty?

# Exercise 4

Here are some informations about a simple world:
*When Mary isn't sick, she sings, she dances with John, and Harry is jealous. When Lucy is sick and wants to run outside, John is afraid. When Mary is not happy, she cannot eat. When Mary is dancing with John or Harry, Lucy is sick. When John or Harry is jealous, Lucy is sick. Mary isn't sick. When John is afraid or Harry is jealous, Mary is not happy. When Mary sings, Lucy wants to run outside.*
Model this universe using propositional logic, and then provide a resolution proof of: *Harry is jealous and Mary cannot eat.*