

Formal semantics and distributional semantics

AnnoDeMo, RT LIFT, Banyuls-sur-Mer

Timothée Bernard

May 5th-7th 2024

Laboratoire de linguistique formelle, Université Paris Cité

Meaning, truth-conditions and entailments

Natural language semantics

How to interpret sentences?

And smaller items (e.g. words)?

Natural language semantics

How to interpret sentences?

And smaller items (e.g. words)?

- What is the interpretation of a linguistic item?

Natural language semantics

How to interpret sentences?

And smaller items (e.g. words)?

- What is the interpretation of a linguistic item?
- Good question. One has freedom here.

- What about a numeral, e.g. 5?

- What about a numeral, e.g. 5?
 - Some abstract atomic entity.

- What about a numeral, e.g. 5?
 - Some abstract atomic entity.
 - The collection of all sets of 5 elements.

- What about a numeral, e.g. 5?
 - Some abstract atomic entity.
 - The collection of all sets of 5 elements.
 - Some other entity?

- What about a numeral, e.g. 5?
 - Some abstract atomic entity.
 - The collection of all sets of 5 elements.
 - Some other entity?
- What about a name, e.g. *Marie Curie*?

- What about a numeral, e.g. 5?
 - Some abstract atomic entity.
 - The collection of all sets of 5 elements.
 - Some other entity?
- What about a name, e.g. *Marie Curie*?
 - A certain person.

- What about a numeral, e.g. 5?
 - Some abstract atomic entity.
 - The collection of all sets of 5 elements.
 - Some other entity?
- What about a name, e.g. *Marie Curie*?
 - A certain person.
 - A certain collection of properties (incl. being a woman, a physicist, a chemist, born in Warsaw).

- What about a numeral, e.g. 5?
 - Some abstract atomic entity.
 - The collection of all sets of 5 elements.
 - Some other entity?
- What about a name, e.g. *Marie Curie*?
 - A certain person.
 - A certain collection of properties (incl. being a woman, a physicist, a chemist, born in Warsaw).
 - Some other entity?

- What about a sentence, e.g. *Today is Wednesday*?

- What about a sentence, e.g. *Today is Wednesday*?
 - Its truth value.

- What about a sentence, e.g. *Today is Wednesday*?
 - Its truth value.
 - The collection of situation it which it is true.

- What about a sentence, e.g. *Today is Wednesday*?
 - Its truth value.
 - The collection of situation it which it is true.
 - Some other entity?

- What about a sentence, e.g. *Today is Wednesday*?
 - Its truth value.
 - The collection of situation it which it is true.
 - Some other entity?
- What about a verb phrase, e.g. *eats a cake*?

- What about a sentence, e.g. *Today is Wednesday*?
 - Its truth value.
 - The collection of situation it which it is true.
 - Some other entity?
- What about a verb phrase, e.g. *eats a cake*?
 - The collection of every person that eats a cake.

- What about a sentence, e.g. *Today is Wednesday*?
 - Its truth value.
 - The collection of situation it which it is true.
 - Some other entity?
- What about a verb phrase, e.g. *eats a cake*?
 - The collection of every person that eats a cake.
 - The collection of every event of someone eating a cake.

- What about a sentence, e.g. *Today is Wednesday*?
 - Its truth value.
 - The collection of situation it which it is true.
 - Some other entity?
- What about a verb phrase, e.g. *eats a cake*?
 - The collection of every person that eats a cake.
 - The collection of every event of someone eating a cake.
 - Some other entity?

- Model: mathematical representation of the universe.
→ the metaphysical/conceptual universe, more than the physical one
- A model contains all the information required to determine the truth value of any (unambiguous) sentence (in addition to grammar of the language/an understanding of the sentence).

Model (of a first-order language)

- $D = \{a, b, c, d, e, f\}$;
- $I(\textit{Sabine}) = a$; $I(\textit{Fred}) = b$; $I(\textit{Paris}) = c$; $I(\textit{Marseille}) = d$;
- $I(\textit{human}) = \{a, b\}$; $I(\textit{book}) = \{e, f\}$; $I(\textit{sleep}) = \{b\}$;
- $I(\textit{read}) = \{(a, e)\}$; $(x, y) \in I(\textit{read})$ iff x reads y
- $I(\textit{in}) = \{(a, c), (b, d)\}$. $(x, y) \in I(\textit{in})$ iff x is in y

Model (of a first-order language)

- $D = \{a, b, c, d, e, f\}$;
- $I(\textit{Sabine}) = a$; $I(\textit{Fred}) = b$; $I(\textit{Paris}) = c$; $I(\textit{Marseille}) = d$;
- $I(\textit{human}) = \{a, b\}$; $I(\textit{book}) = \{e, f\}$; $I(\textit{sleep}) = \{b\}$;
- $I(\textit{read}) = \{(a, e)\}$; $(x, y) \in I(\textit{read})$ iff x reads y
- $I(\textit{in}) = \{(a, c), (b, d)\}$. $(x, y) \in I(\textit{in})$ iff x is in y

- Model-theoretic semantics (MTS) interprets items (e.g. words, sentences) in terms of a model.
- Two main questions:
 1. For any given sentence, under which conditions is it *true*?
 2. For any given pair of sentences, does the first *entail* the second?

- Question No. 2 (about entailment) is *weaker* than question No. 1 (about truth conditions).
 - Given two sentences S_1 and S_2 , (by definition:) S_1 entails S_2 iff what makes S_1 true makes S_2 true too.

- Question No. 2 (about entailment) is *weaker* than question No. 1 (about truth conditions).
 - Given two sentences S_1 and S_2 , (by definition:)
 S_1 entails S_2 iff what makes S_1 true makes S_2 true too.
 - answering question No. 1 \Rightarrow answering question No. 2

- Question No. 2 (about entailment) is *weaker* than question No. 1 (about truth conditions).
 - Given two sentences S_1 and S_2 , (by definition:)
 S_1 entails S_2 iff what makes S_1 true makes S_2 true too.
 - answering question No. 1 \Rightarrow answering question No. 2
- For question No. 2, MTS does not seem necessary:
 - (1) a. Sabine is a human being and is reading a book.
 - b. Sabine is reading a book.
 - c. Sabine or Fred is reading a book

- Question No. 2 (about entailment) is *weaker* than question No. 1 (about truth conditions).
 - Given two sentences S_1 and S_2 , (by definition:)
 - S_1 entails S_2 iff what makes S_1 true makes S_2 true too.
 - answering question No. 1 \Rightarrow answering question No. 2
- For question No. 2, MTS does not seem necessary:
 - (1)
 - a. Sabine is a human being and is reading a book.
 - b. Sabine is reading a book.
 - c. Sabine or Fred is reading a book
- $[(1a)] \models [(1b)]$, $[(1c)]$, and $[(1b)] \models [(1c)]$

- MTS answers question (i) (and also (ii)) based on a notion of *reference*.
- $D = \{a, \dots\}$
 $I(\text{Sabine}) = a; \dots; I(\text{human}) = \{a, \dots\}; \dots$
- Natural language is arguably *grounded* in our extra-linguistic environment (Harnad [1990](#)).

- MTS answers question (i) (and also (ii)) based on a notion of *reference*.
- $D = \{a, \dots\}$
 $I(\text{Sabine}) = a; \dots; I(\text{human}) = \{a, \dots\}; \dots$
- Natural language is arguably *grounded* in our extra-linguistic environment (Harnad 1990).
(Grounding might be one of the missing key ingredients in AI systems; Bender and Koller 2020.)

- MTS answers question (i) (and also (ii)) based on a notion of *reference*.
- $D = \{a, \dots\}$
 $I(\text{Sabine}) = a; \dots; I(\text{human}) = \{a, \dots\}; \dots$
- Natural language is arguably *grounded* in our extra-linguistic environment (Harnad 1990).
 (Grounding might be one of the missing key ingredients in AI systems; Bender and Koller 2020.)
- Grounding/reference is a primitive phenomena in MTS.
 → probably why MTS has been more popular than proper proof-theoretic semantics in the XXth century
- Interestingly, MTS was first developed in the philosophy of mathematics and was not adopted by linguists before the 1960s.

- Advantage of question No. 2: it is less metaphysical.

For any given pair of sentences, does the first *entail* the second?

- Advantage of question No. 2: it is less metaphysical.

For any given pair of sentences, does the first *entail* the second?

- Strongly related to logic: the study of the formal validity of arguments.

- Advantage of question No. 2: it is less metaphysical.

For any given pair of sentences, does the first *entail* the second?

- Strongly related to logic: the study of the formal validity of arguments.
- The birth of logic is usually associated with Aristotle (385-322 BC).
- Has been interpreted as a reaction to the Sophists.

- Advantage of question No. 2: it is less metaphysical.

For any given pair of sentences, does the first *entail* the second?

- Strongly related to logic: the study of the formal validity of arguments.
- The birth of logic is usually associated with Aristotle (385-322 BC).
- Has been interpreted as a reaction to the Sophists.
- Aristotle's approach: What in the form of an argument makes it valid (independently of its rhetorical force)?

- Which of these arguments, if any, are valid?
 - (2)
 - a. Premise(s): All politicians are criminals, and some criminals are liars.
Conclusion: Some politicians are liars.
 - b. Premise(s): Some politicians are criminals, and all criminals are liars.
Conclusion: Some politicians are liars.

- Which of these arguments, if any, are valid?
 - (2) a. Premise(s): All politicians are criminals, and some criminals are liars.
Conclusion: Some politicians are liars.
 - b. Premise(s): Some politicians are criminals, and all criminals are liars.
Conclusion: Some politicians are liars.
- Only (2b).

- Any argument of the same form is valid.
- (3) Premise(s): Some As are Bs, and all Bs are Cs.
Conclusion: Some As are Cs.

- Any argument of the same form is valid.
 - (3) Premise(s): Some As are Bs, and all Bs are Cs.
Conclusion: Some As are Cs.
- This pattern expresses part of the meaning of *some* and *all*, in proof-theoretic terms.

- Any argument of the same form is valid.
- (3) Premise(s): Some As are Bs, and all Bs are Cs.
Conclusion: Some As are Cs.
- This pattern expresses part of the meaning of *some* and *all*, in proof-theoretic terms.
- It is possible to describe all possible entailment relations in natural language (NL) with such patterns?
- This looks (at least) very difficult.
- One of the reasons why such patterns are not ideal: NL is highly *ambiguous*.

- *Structural* ambiguity:

- (4) Fred saw Sabine with a pair of glasses.

- *Structural* ambiguity:

(4) Fred saw Sabine with a pair of glasses.

- *Lexical* ambiguity:

(5) Jamy had lunch near the bank.

- *Structural* ambiguity:
 - (4) Fred saw Sabine with a pair of glasses.
- *Lexical* ambiguity:
 - (5) Jamy had lunch near the bank.
- The methodology of formal semantics:
 - define a logical language with well-known inferential properties,
 - define a translation function from NL sentences to sets of logical formulas.
- The translation should preserve the entailment relations.

The language of first-order logic

- The development of FOL started in the work of Charles Sanders Peirce (1839-1914), Giuseppe Peano (1858-1932) and Gottlob Frege (1848-1925).
- Alfred Tarski (1901-1983) gave it its now standard (model-theoretic) semantics.

Alphabet of FOL

Alphabet of FOL

Five kinds of symbols:

Alphabet of FOL

Five kinds of symbols:

- a set \mathcal{C} of elements called 'constants';

Alphabet of FOL

Five kinds of symbols:

- a set \mathcal{C} of elements called 'constants';
- a set \mathcal{X} of elements called 'variables';

Alphabet of FOL

Five kinds of symbols:

- a set \mathcal{C} of elements called 'constants';
- a set \mathcal{X} of elements called 'variables';
- a set \mathcal{P} of elements called 'predicates', each of them associated with a natural number called its 'arity';

Alphabet of FOL

Five kinds of symbols:

- a set \mathcal{C} of elements called ‘constants’;
- a set \mathcal{X} of elements called ‘variables’;
- a set \mathcal{P} of elements called ‘predicates’, each of them associated with a natural number called its ‘arity’;
- a set of three elements called ‘logical operators’: negation \neg , conjunction \wedge and the existential quantifier \exists ;

Alphabet of FOL

Five kinds of symbols:

- a set \mathcal{C} of elements called ‘constants’;
- a set \mathcal{X} of elements called ‘variables’;
- a set \mathcal{P} of elements called ‘predicates’, each of them associated with a natural number called its ‘arity’;
- a set of three elements called ‘logical operators’: negation \neg , conjunction \wedge and the existential quantifier \exists ;
- the open (and close) parentheses.

Alphabet of FOL

Five kinds of symbols:

- a set \mathcal{C} of elements called ‘constants’;
- a set \mathcal{X} of elements called ‘variables’;
- a set \mathcal{P} of elements called ‘predicates’, each of them associated with a natural number called its ‘arity’;
- a set of three elements called ‘logical operators’: negation \neg , conjunction \wedge and the existential quantifier \exists ;
- the open (and close) parentheses.

(This is a slight simplification; full FOL uses *functions* of various arity instead of constants.)

Natural language in FOL

- Constants: *Sabine, Marcel*
- Unary predicates: *human, book, truck, sleep, old,*
- Binary predicates: *drive, read*

Natural language in FOL

- Constants: *Sabine, Marcel*
- Unary predicates: *human, book, truck, sleep, old,*
- Binary predicates: *drive, read*
- Formulas:
 - [Sabine is human] = *human(Sabine)*

Natural language in FOL

- Constants: *Sabine, Marcel*
- Unary predicates: *human, book, truck, sleep, old,*
- Binary predicates: *drive, read*
- Formulas:
 - [Sabine is human] = $human(Sabine)$
 - [There is a book] = $\exists x. book(x)$

Natural language in FOL

- Constants: *Sabine, Marcel*
- Unary predicates: *human, book, truck, sleep, old,*
- Binary predicates: *drive, read*
- Formulas:
 - [Sabine is human] = $human(Sabine)$
 - [There is a book] = $\exists x. book(x)$
 - [There is an old book] = $\exists x. book(x) \wedge old(x)$

Natural language in FOL

- Constants: *Sabine, Marcel*
- Unary predicates: *human, book, truck, sleep, old,*
- Binary predicates: *drive, read*
- Formulas:
 - [Sabine is human] = $human(Sabine)$
 - [There is a book] = $\exists x. book(x)$
 - [There is an old book] = $\exists x. book(x) \wedge old(x)$
 - [Marcel drives a truck] = $\exists x. drive(Marcel, x) \wedge truck(x)$

Natural language in FOL

- Constants: *Sabine, Marcel*
- Unary predicates: *human, book, truck, sleep, old,*
- Binary predicates: *drive, read*
- Formulas:
 - [Sabine is human] = $human(Sabine)$
 - [There is a book] = $\exists x. book(x)$
 - [There is an old book] = $\exists x. book(x) \wedge old(x)$
 - [Marcel drives a truck] = $\exists x. drive(Marcel, x) \wedge truck(x)$
 - [Sabine reads a book] = $\exists x. book(x) \wedge read(Sabine, x)$

Natural language in FOL

- Constants: *Sabine, Marcel*
- Unary predicates: *human, book, truck, sleep, old,*
- Binary predicates: *drive, read*
- Formulas:
 - [Sabine is human] = $human(Sabine)$
 - [There is a book] = $\exists x. book(x)$
 - [There is an old book] = $\exists x. book(x) \wedge old(x)$
 - [Marcel drives a truck] = $\exists x. drive(Marcel, x) \wedge truck(x)$
 - [Sabine reads a book] = $\exists x. book(x) \wedge read(Sabine, x)$
 - [a book is read by Sabine] =

Natural language in FOL

- Constants: *Sabine, Marcel*
- Unary predicates: *human, book, truck, sleep, old,*
- Binary predicates: *drive, read*
- Formulas:
 - [Sabine is human] = $human(Sabine)$
 - [There is a book] = $\exists x. book(x)$
 - [There is an old book] = $\exists x. book(x) \wedge old(x)$
 - [Marcel drives a truck] = $\exists x. drive(Marcel, x) \wedge truck(x)$
 - [Sabine reads a book] = $\exists x. book(x) \wedge read(Sabine, x)$
 - [a book is read by Sabine] = $\exists x. book(x) \wedge read(Sabine, x)$

Natural language in FOL

- Constants: *Sabine, Marcel*
- Unary predicates: *human, book, truck, sleep, old,*
- Binary predicates: *drive, read*
- Formulas:
 - [Sabine is human] = $human(Sabine)$
 - [There is a book] = $\exists x. book(x)$
 - [There is an old book] = $\exists x. book(x) \wedge old(x)$
 - [Marcel drives a truck] = $\exists x. drive(Marcel, x) \wedge truck(x)$
 - [Sabine reads a book] = $\exists x. book(x) \wedge read(Sabine, x)$
 - [a book is read by Sabine] = $\exists x. book(x) \wedge read(Sabine, x)$
 - [Sabine reads no book] = $\neg(\exists x. book(x) \wedge read(Sabine, x))$

Natural language in FOL

- Constants: *Sabine, Marcel*
- Unary predicates: *human, book, truck, sleep, old,*
- Binary predicates: *drive, read*
- Formulas:
 - [Sabine is human] = $human(Sabine)$
 - [There is a book] = $\exists x. book(x)$
 - [There is an old book] = $\exists x. book(x) \wedge old(x)$
 - [Marcel drives a truck] = $\exists x. drive(Marcel, x) \wedge truck(x)$
 - [Sabine reads a book] = $\exists x. book(x) \wedge read(Sabine, x)$
 - [a book is read by Sabine] = $\exists x. book(x) \wedge read(Sabine, x)$
 - [Sabine reads no book] = $\neg(\exists x. book(x) \wedge read(Sabine, x))$
= $\forall x. book(x) \rightarrow \neg read(Sabine, x)$

Natural language in FOL

- Constants: *Sabine, Marcel*
- Unary predicates: *human, book, truck, sleep, old,*
- Binary predicates: *drive, read*
- Formulas:
 - [Sabine is human] = $human(Sabine)$
 - [There is a book] = $\exists x. book(x)$
 - [There is an old book] = $\exists x. book(x) \wedge old(x)$
 - [Marcel drives a truck] = $\exists x. drive(Marcel, x) \wedge truck(x)$
 - [Sabine reads a book] = $\exists x. book(x) \wedge read(Sabine, x)$
 - [a book is read by Sabine] = $\exists x. book(x) \wedge read(Sabine, x)$
 - [Sabine reads no book] = $\neg(\exists x. book(x) \wedge read(Sabine, x))$
= $\forall x. book(x) \rightarrow \neg read(Sabine, x)$
 - [Sabine reads every book] = $\neg(\exists x. book(x) \wedge \neg read(Sabine, x))$

Natural language in FOL

- Constants: *Sabine, Marcel*
- Unary predicates: *human, book, truck, sleep, old,*
- Binary predicates: *drive, read*
- Formulas:
 - [Sabine is human] = $human(Sabine)$
 - [There is a book] = $\exists x. book(x)$
 - [There is an old book] = $\exists x. book(x) \wedge old(x)$
 - [Marcel drives a truck] = $\exists x. drive(Marcel, x) \wedge truck(x)$
 - [Sabine reads a book] = $\exists x. book(x) \wedge read(Sabine, x)$
 - [a book is read by Sabine] = $\exists x. book(x) \wedge read(Sabine, x)$
 - [Sabine reads no book] = $\neg(\exists x. book(x) \wedge read(Sabine, x))$
= $\forall x. book(x) \rightarrow \neg read(Sabine, x)$
 - [Sabine reads every book] = $\neg(\exists x. book(x) \wedge \neg read(Sabine, x))$
= $\forall x. book(x) \rightarrow read(Sabine, x)$

- Are these translations satisfying? (truth conditions? entailment relations?)
- These can be defined in terms of *models*.
- A model of FOL defines how to interpret the basic constituents of the language, except for the variables, via a *domain* (D) and an *interpretation function* (I):

- Are these translations satisfying? (truth conditions? entailment relations?)
- These can be defined in terms of *models*.
- A model of FOL defines how to interpret the basic constituents of the language, except for the variables, via a *domain* (D) and an *interpretation function* (I):
 - $I(\text{a constant}) = \text{an entity} \in D$

- Are these translations satisfying? (truth conditions? entailment relations?)
- These can be defined in terms of *models*.
- A model of FOL defines how to interpret the basic constituents of the language, except for the variables, via a *domain* (D) and an *interpretation function* (I):
 - $I(\text{a constant}) = \text{an entity} \in D$
 - $I(\text{a unary predicate}) = \text{a set of entities} \in D$

- Are these translations satisfying? (truth conditions? entailment relations?)
- These can be defined in terms of *models*.
- A model of FOL defines how to interpret the basic constituents of the language, except for the variables, via a *domain* (D) and an *interpretation function* (I):
 - $I(\text{a constant}) = \text{an entity } \in D$
 - $I(\text{a unary predicate}) = \text{a set of entities } \in D$
 - $I(\text{a binary predicate}) = \text{a set of pairs of entities } \in D$

- Are these translations satisfying? (truth conditions? entailment relations?)
- These can be defined in terms of *models*.
- A model of FOL defines how to interpret the basic constituents of the language, except for the variables, via a *domain* (D) and an *interpretation function* (I):
 - $I(\text{a constant}) = \text{an entity} \in D$
 - $I(\text{a unary predicate}) = \text{a set of entities} \in D$
 - $I(\text{a binary predicate}) = \text{a set of pairs of entities} \in D$
 - $I(\text{a ternary predicate}) = \text{a set triplets of entities} \in D$

- Are these translations satisfying? (truth conditions? entailment relations?)
- These can be defined in terms of *models*.
- A model of FOL defines how to interpret the basic constituents of the language, except for the variables, via a *domain* (D) and an *interpretation function* (I):
 - $I(\text{a constant}) = \text{an entity} \in D$
 - $I(\text{a unary predicate}) = \text{a set of entities} \in D$
 - $I(\text{a binary predicate}) = \text{a set of pairs of entities} \in D$
 - $I(\text{a ternary predicate}) = \text{a set triplets of entities} \in D$
 - etc.

Model (of a first-order language)

- $D = \{a, b, c, d, e\}$;
- $I(\textit{Sabine}) = a; I(\textit{Marcel}) = b$;
- $I(\textit{human}) = \{a, b\}; I(\textit{book}) = \{c, d\}; I(\textit{truck}) = \{e\}$;
- $I(\textit{read}) = \{(a, e)\};$ $(x, y) \in I(\textit{read})$ iff x reads y
- $I(\textit{drive}) = \{(b, c)\};$ $(x, y) \in I(\textit{drive})$ iff x drives y

- Variables get their interpretation from a *variable assignment* f :

- Variables get their interpretation from a *variable assignment* f :
 - $f(\text{a variable}) = \text{an entity} \in D$

- Variables get their interpretation from a *variable assignment* f :
 - $f(\text{a variable}) = \text{an entity} \in D$
- important detail: A variable assignment f can be ‘updated’: for $x \in \mathcal{X}$ and $d \in D$, $f[x := d]$ is a new variable assignment.

- Given a model $M = (D, I)$ and a variable assignment f , the *valuation* is the function defined inductively by:

- Given a model $M = (D, I)$ and a variable assignment f , the *valuation* is the function defined inductively by:

- $\forall n$ -ary $P \in \mathcal{P}, \forall t_1, t_2, \dots, t_n \in \mathcal{C} \cup \mathcal{X}$,

$$\llbracket P(t_1, t_2, \dots, t_n) \rrbracket_{M,f} = \begin{cases} T & \text{if } (\llbracket t_1 \rrbracket_{M,f}, \llbracket t_2 \rrbracket_{M,f}, \dots, \llbracket t_n \rrbracket_{M,f}) \in I(P); \\ F & \text{otherwise} \end{cases};$$

- Given a model $M = (D, I)$ and a variable assignment f , the *valuation* is the function defined inductively by:
 - $\forall n$ -ary $P \in \mathcal{P}, \forall t_1, t_2, \dots, t_n \in \mathcal{C} \cup \mathcal{X}$,

$$\llbracket P(t_1, t_2, \dots, t_n) \rrbracket_{M,f} = \begin{cases} T & \text{if } (\llbracket t_1 \rrbracket_{M,f}, \llbracket t_2 \rrbracket_{M,f}, \dots, \llbracket t_n \rrbracket_{M,f}) \in I(P); \\ F & \text{otherwise} \end{cases};$$
 - $\forall \phi \in \mathcal{F}, \llbracket \neg \phi \rrbracket_{M,f} = \begin{cases} T & \text{if } \llbracket \phi \rrbracket_{M,f} = F; \\ F & \text{otherwise} \end{cases};$

- Given a model $M = (D, I)$ and a variable assignment f , the *valuation* is the function defined inductively by:

- $\forall n$ -ary $P \in \mathcal{P}, \forall t_1, t_2, \dots, t_n \in \mathcal{C} \cup \mathcal{X}$,

$$\llbracket P(t_1, t_2, \dots, t_n) \rrbracket_{M,f} = \begin{cases} T & \text{if } (\llbracket t_1 \rrbracket_{M,f}, \llbracket t_2 \rrbracket_{M,f}, \dots, \llbracket t_n \rrbracket_{M,f}) \in I(P); \\ F & \text{otherwise} \end{cases};$$

- $\forall \phi \in \mathcal{F}, \llbracket \neg \phi \rrbracket_{M,f} = \begin{cases} T & \text{if } \llbracket \phi \rrbracket_{M,f} = F; \\ F & \text{otherwise} \end{cases};$

- $\forall \phi, \psi \in \mathcal{F}, \llbracket \phi \wedge \psi \rrbracket_{M,f} = \begin{cases} T & \text{if } \llbracket \phi \rrbracket_{M,f} = T \text{ and } \llbracket \psi \rrbracket_{M,f} = T; \\ F & \text{otherwise} \end{cases};$

- Given a model $M = (D, I)$ and a variable assignment f , the *valuation* is the function defined inductively by:

- $\forall n$ -ary $P \in \mathcal{P}, \forall t_1, t_2, \dots, t_n \in \mathcal{C} \cup \mathcal{X}$,

$$\llbracket P(t_1, t_2, \dots, t_n) \rrbracket_{M,f} = \begin{cases} T & \text{if } (\llbracket t_1 \rrbracket_{M,f}, \llbracket t_2 \rrbracket_{M,f}, \dots, \llbracket t_n \rrbracket_{M,f}) \in I(P); \\ F & \text{otherwise} \end{cases};$$

- $\forall \phi \in \mathcal{F}, \llbracket \neg \phi \rrbracket_{M,f} = \begin{cases} T & \text{if } \llbracket \phi \rrbracket_{M,f} = F; \\ F & \text{otherwise} \end{cases};$

- $\forall \phi, \psi \in \mathcal{F}, \llbracket \phi \wedge \psi \rrbracket_{M,f} = \begin{cases} T & \text{if } \llbracket \phi \rrbracket_{M,f} = T \text{ and } \llbracket \psi \rrbracket_{M,f} = T; \\ F & \text{otherwise} \end{cases};$

- $\forall \phi \in \mathcal{F}, \forall x \in \mathcal{X}$,

$$\llbracket \exists x. \phi \rrbracket_{M,f} = \begin{cases} T & \text{if there is some } d \in D \text{ s.t. } \llbracket \phi \rrbracket_{M,f[x:=d]} = T; \\ F & \text{otherwise} \end{cases}.$$

- Given a model $M = (D, I)$ and a variable assignment f , the *valuation* is the function defined inductively by:

- $\forall n$ -ary $P \in \mathcal{P}, \forall t_1, t_2, \dots, t_n \in \mathcal{C} \cup \mathcal{X}$,

$$\llbracket P(t_1, t_2, \dots, t_n) \rrbracket_{M,f} = \begin{cases} T & \text{if } (\llbracket t_1 \rrbracket_{M,f}, \llbracket t_2 \rrbracket_{M,f}, \dots, \llbracket t_n \rrbracket_{M,f}) \in I(P); \\ F & \text{otherwise} \end{cases};$$

- $\forall \phi \in \mathcal{F}, \llbracket \neg \phi \rrbracket_{M,f} = \begin{cases} T & \text{if } \llbracket \phi \rrbracket_{M,f} = F; \\ F & \text{otherwise} \end{cases};$

- $\forall \phi, \psi \in \mathcal{F}, \llbracket \phi \wedge \psi \rrbracket_{M,f} = \begin{cases} T & \text{if } \llbracket \phi \rrbracket_{M,f} = T \text{ and } \llbracket \psi \rrbracket_{M,f} = T; \\ F & \text{otherwise} \end{cases};$

- $\forall \phi \in \mathcal{F}, \forall x \in \mathcal{X}$,

$$\llbracket \exists x. \phi \rrbracket_{M,f} = \begin{cases} T & \text{if there is some } d \in D \text{ s.t. } \llbracket \phi \rrbracket_{M,f[x:=d]} = T; \\ F & \text{otherwise} \end{cases}.$$

- This definition covers (recursively) all possible FOL formulas.

Consequences

- [Sabine reads a book] = $\exists x. \text{book}(x) \wedge \text{read}(\text{Sabine}, x)$
True iff there is an entity $d \in I(\text{book})$ such that
 $(I(\text{Sabine}), d) \in I(\text{read})$

Consequences

- [Sabine reads a book] = $\exists x. \text{book}(x) \wedge \text{read}(\text{Sabine}, x)$
True iff there is an entity $d \in I(\text{book})$ such that $(I(\text{Sabine}), d) \in I(\text{read})$ i.e. iff there is an entity that is both a book and read by Sabine.

Consequences

- [Sabine reads a book] = $\exists x. \text{book}(x) \wedge \text{read}(\text{Sabine}, x)$
True iff there is an entity $d \in I(\text{book})$ such that $(I(\text{Sabine}), d) \in I(\text{read})$ i.e. iff there is an entity that is both a book and read by Sabine.
- [Sabine reads an old book] = $\exists x. \text{old}(x) \wedge \text{book}(x) \wedge \text{read}(\text{Sabine}, x)$
True iff there is an entity $d \in I(\text{old})$ such that $d \in I(\text{book})$ and $(I(\text{Sabine}), d) \in I(\text{read})$

Consequences

- [Sabine reads a book] = $\exists x. \text{book}(x) \wedge \text{read}(\text{Sabine}, x)$
True iff there is an entity $d \in I(\text{book})$ such that $(I(\text{Sabine}), d) \in I(\text{read})$ i.e. iff there is an entity that is both a book and read by Sabine.
- [Sabine reads an old book] = $\exists x. \text{old}(x) \wedge \text{book}(x) \wedge \text{read}(\text{Sabine}, x)$
True iff there is an entity $d \in I(\text{old})$ such that $d \in I(\text{book})$ and $(I(\text{Sabine}), d) \in I(\text{read})$ i.e. iff there is an entity that is old and a book and read by Sabine.

Consequences

- [Sabine reads no book] = $\neg(\exists x. \text{book}(x) \wedge \text{read}(\text{Sabine}, x))$
= $\forall x. \text{book}(x) \rightarrow \neg\text{read}(\text{Sabine}, x)$
True iff there is not entity that is both a book and read by Sabine.

Consequences

- [Sabine reads no book] = $\neg(\exists x. \text{book}(x) \wedge \text{read}(\text{Sabine}, x))$
= $\forall x. \text{book}(x) \rightarrow \neg \text{read}(\text{Sabine}, x)$
True iff there is not entity that is both a book and read by Sabine.
- [Sabine reads every book] = $\neg(\exists x. \text{book}(x) \wedge \neg \text{read}(\text{Sabine}, x))$
= $\forall x. \text{book}(x) \rightarrow \text{read}(\text{Sabine}, x)$
True iff all entities that are books are also read by Sabine.

- Lewis (1970):
 - The model-based valuation is important!
(or a logic for the formulas)

- Lewis (1970):
 - The model-based valuation is important!
(or a logic for the formulas)
 - Without truth-conditions nor logic, FOL formulas are meaningless.
 - Translating sentences to meaningless formulas is useless.

- Aristotle's syllogistic can be expressed in FOL.

premises:	Every human is mortal. Socrates is human.	$\forall x. human(x) \rightarrow mortal(x)$ $human(Socrates)$
conclusion:	Socrates is mortal.	$mortal(Socrates)$

- Validity of the syllogism: for any model in which both premises are true, the conclusion is also true.
(the premises taken together entail the conclusion)

- Many NL sentences can be translated into FOL.

- Many NL sentences can be translated into FOL. (even more with extensions)

- Many NL sentences can be translated into FOL. (even more with extensions)
- Various forms of knowledge too:

(6) a. $\forall x. \text{cat}(x) \rightarrow \text{mammal}(x)$

- Many NL sentences can be translated into FOL. (even more with extensions)
- Various forms of knowledge too:

(6) a. $\forall x. \text{cat}(x) \rightarrow \text{mammal}(x)$
(hyponymy)

- Many NL sentences can be translated into FOL. (even more with extensions)
- Various forms of knowledge too:

(6) a. $\forall x. \text{cat}(x) \rightarrow \text{mammal}(x)$

(hyponymy)

b. $\forall x. \text{bachelor}(x) \leftrightarrow \text{male}(x) \wedge \text{single}(x)$

- Many NL sentences can be translated into FOL. (even more with extensions)
- Various forms of knowledge too:

(6) a. $\forall x. \text{cat}(x) \rightarrow \text{mammal}(x)$

(hyponymy)

b. $\forall x. \text{bachelor}(x) \leftrightarrow \text{male}(x) \wedge \text{single}(x)$

(synonymy)

- Many NL sentences can be translated into FOL. (even more with extensions)
- Various forms of knowledge too:

(6) a. $\forall x. \text{cat}(x) \rightarrow \text{mammal}(x)$

(hyponymy)

b. $\forall x. \text{bachelor}(x) \leftrightarrow \text{male}(x) \wedge \text{single}(x)$

(synonymy)

c. $\forall x. \text{in}(x, \text{Paris}) \rightarrow \text{in}(x, \text{France})$

- Many NL sentences can be translated into FOL. (even more with extensions)
- Various forms of knowledge too:

(6) a. $\forall x. \text{cat}(x) \rightarrow \text{mammal}(x)$

(hyponymy)

b. $\forall x. \text{bachelor}(x) \leftrightarrow \text{male}(x) \wedge \text{single}(x)$

(synonymy)

c. $\forall x. \text{in}(x, \text{Paris}) \rightarrow \text{in}(x, \text{France})$

(geographical knowledge)

- Many NL sentences can be translated into FOL. (even more with extensions)
- Various forms of knowledge too:

(6) a. $\forall x. \text{cat}(x) \rightarrow \text{mammal}(x)$

(hyponymy)

b. $\forall x. \text{bachelor}(x) \leftrightarrow \text{male}(x) \wedge \text{single}(x)$

(synonymy)

c. $\forall x. \text{in}(x, \text{Paris}) \rightarrow \text{in}(x, \text{France})$

(geographical knowledge)

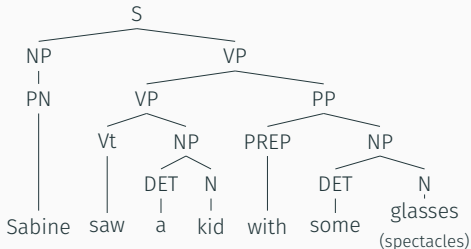
- However, everyday reasoning involves a lot of rules like *dogs usually have four legs* or *most adult birds can fly*, which cannot be faithfully expressed in standard FOL.

Compositionality

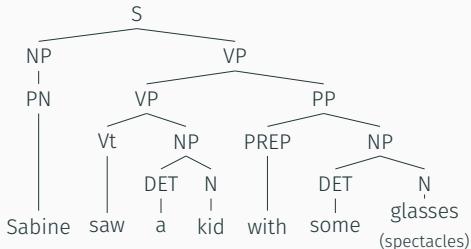
- Many NL sentences can be translated into FOL. How?

- Many NL sentences can be translated into FOL. How?
- Standard methodology in formal semantics:
 - start with a syntactic structure (or derivation, or ...), in which lexical items are disambiguated,
 - describe how this syntactic object is translated into a formula.

- Many NL sentences can be translated into FOL. How?
- Standard methodology in formal semantics:
 - start with a syntactic structure (or derivation, or ...), in which lexical items are disambiguated,
 - describe how this syntactic object is translated into a formula.



- Many NL sentences can be translated into FOL. How?
- Standard methodology in formal semantics:
 - start with a syntactic structure (or derivation, or ...), in which lexical items are disambiguated,
 - describe how this syntactic object is translated into a formula.



$\exists x. kid \wedge \exists y. glass.spectacles(y) \wedge see(Sabine, x) \wedge with(Sabine, y)$

- Until the 60s, formal linguists paid little attention to semantics.

- Until the 60s, formal linguists paid little attention to semantics.
- Formal logic had been used by philosophers of language

- Until the 60s, formal linguists paid little attention to semantics.
- Formal logic had been used by philosophers of language but more to study inference patterns between sentences (e.g. in relation to belief, using modal logic) than to understand the systematic relation that exists between a sentence and its meaning.

- Until the 60s, formal linguists paid little attention to semantics.
- Formal logic had been used by philosophers of language but more to study inference patterns between sentences (e.g. in relation to belief, using modal logic) than to understand the systematic relation that exists between a sentence and its meaning.
- Montague
 - was a logician who unified and applied some of the latest tools from symbolic and philosophical logic to natural language,
 - wanted to show how semantics could be computed *compositionally* from syntax.

- Until the 60s, formal linguists paid little attention to semantics.
- Formal logic had been used by philosophers of language but more to study inference patterns between sentences (e.g. in relation to belief, using modal logic) than to understand the systematic relation that exists between a sentence and its meaning.
- Montague
 - was a logician who unified and applied some of the latest tools from symbolic and philosophical logic to natural language,
 - wanted to show how semantics could be computed *compositionally* from syntax.
- See the work of Partee (2011) and Partee (2014) for more about the history of formal semantics.

- Bach (1989, p.8):

In Montague's papers on natural language, which were written in the late 1960s and early 1970s, Montague claimed that natural languages could be treated in just the same way as the formal artificial languages of the logician.

- Bach (1989, p.8):

In Montague's papers on natural language, which were written in the late 1960s and early 1970s, Montague claimed that natural languages could be treated in just the same way as the formal artificial languages of the logician. [...] Chomsky's thesis was that natural languages can be described as formal systems. Montague added to this the idea that natural languages can be described as interpreted formal systems.

- Bach (1989, p.8):
In Montague's papers on natural language, which were written in the late 1960s and early 1970s, Montague claimed that natural languages could be treated in just the same way as the formal artificial languages of the logician. [...] Chomsky's thesis was that natural languages can be described as formal systems. Montague added to this the idea that natural languages can be described as interpreted formal systems.
- Three seminal articles:

- Bach (1989, p.8):
In Montague's papers on natural language, which were written in the late 1960s and early 1970s, Montague claimed that natural languages could be treated in just the same way as the formal artificial languages of the logician. [...] Chomsky's thesis was that natural languages can be described as formal systems. Montague added to this the idea that natural languages can be described as interpreted formal systems.
- Three seminal articles:
 - *English as a formal language* (Montague 1970a);

- Bach (1989, p.8):
In Montague's papers on natural language, which were written in the late 1960s and early 1970s, Montague claimed that natural languages could be treated in just the same way as the formal artificial languages of the logician. [...] Chomsky's thesis was that natural languages can be described as formal systems. Montague added to this the idea that natural languages can be described as interpreted formal systems.
- Three seminal articles:
 - *English as a formal language* (Montague 1970a);
 - *Universal grammar* (Montague 1970b);

- Bach (1989, p.8):
In Montague's papers on natural language, which were written in the late 1960s and early 1970s, Montague claimed that natural languages could be treated in just the same way as the formal artificial languages of the logician. [...] Chomsky's thesis was that natural languages can be described as formal systems. Montague added to this the idea that natural languages can be described as interpreted formal systems.
- Three seminal articles:
 - *English as a formal language* (Montague 1970a);
 - *Universal grammar* (Montague 1970b);
 - *The proper treatment of quantification in ordinary English* (Montague 1973, 'PTQ').

- Principle of compositionality (Gamut 1991):
the meaning of a composite expression must be wholly determined by the meanings of its composite parts and of the syntactic rule by means of which it is formed

- Principle of compositionality (Gamut 1991):
the meaning of a composite expression must be wholly determined by the meanings of its composite parts and of the syntactic rule by means of which it is formed
- RMK: The ‘composite parts’ of a constituent are its *direct* subconstituents.

- Principle of compositionality (Gamut 1991):
the meaning of a composite expression must be wholly determined by the meanings of its composite parts and of the syntactic rule by means of which it is formed
- RMK: The ‘composite parts’ of a constituent are its *direct* subconstituents.
- Frege is often credited for this principle, but this is questionable (Pelletier 1994).

- Principle of compositionality (Gamut 1991):
the meaning of a composite expression must be wholly determined by the meanings of its composite parts and of the syntactic rule by means of which it is formed
- RMK: The ‘composite parts’ of a constituent are its *direct* subconstituents.
- Frege is often credited for this principle, but this is questionable (Pelletier 1994).
- According to compositionality, syntax describes how to combine the meaning of the units of meaning in a very systematic way.

- Some consequences:

- Some consequences:
 - replacing a component with a paraphrase (irrespectively of their structure) has no impact on the meaning of the whole,

- Some consequences:
 - replacing a component with a paraphrase (irrespectively of their structure) has no impact on the meaning of the whole,
 - semantics = one semantic composition rule per syntactic rule + lexical semantics,

- Some consequences:
 - replacing a component with a paraphrase (irrespectively of their structure) has no impact on the meaning of the whole,
 - semantics = one semantic composition rule per syntactic rule + lexical semantics,
 - the semantics of a constituent does not depend on its role (e.g. subject or object for an NP) in the sentence,

- Some consequences:
 - replacing a component with a paraphrase (irrespectively of their structure) has no impact on the meaning of the whole,
 - semantics = one semantic composition rule per syntactic rule + lexical semantics,
 - the semantics of a constituent does not depend on its role (e.g. subject or object for an NP) in the sentence,
 - the meaning of an expression is the same in all contexts,

- Some consequences:
 - replacing a component with a paraphrase (irrespectively of their structure) has no impact on the meaning of the whole,
 - semantics = one semantic composition rule per syntactic rule + lexical semantics,
 - the semantics of a constituent does not depend on its role (e.g. subject or object for an NP) in the sentence,
 - the meaning of an expression is the same in all contexts,
 - once one knows the meaning of a lexical item, they can use it in any structure/context.

- Some consequences:
 - replacing a component with a paraphrase (irrespectively of their structure) has no impact on the meaning of the whole,
 - semantics = one semantic composition rule per syntactic rule + lexical semantics,
 - the semantics of a constituent does not depend on its role (e.g. subject or object for an NP) in the sentence,
 - the meaning of an expression is the same in all contexts,
 - once one knows the meaning of a lexical item, they can use it in any structure/context.
- One would often like AI systems to process language more compositionally.

- Can be seen as a methodological principle. E.g.,
 - to draw the line between semantics and pragmatics;
 - to define multiword expressions (*to kick the bucket, ivory tower*).
- Stronger or less depending on the kind of composition rules and lexical semantic entries that one is ready to accept.

- Can be seen as a methodological principle. E.g.,
 - to draw the line between semantics and pragmatics;
 - to define multiword expressions (*to kick the bucket, ivory tower*).
- Stronger or less depending on the kind of composition rules and lexical semantic entries that one is ready to accept.
- Often seen as the source of the *productivity* of natural language, which ‘can (in Humboldt’s words) “make infinite use of finite means”’ (Chomsky 1965).

- Let us define a compositional grammar.

- Let us define a compositional grammar.
- This is an *introduction* to formal semantics; we're about to do something *much* simpler (both in terms of syntax and semantics) than what Montague did (and what has been done since).

- Let us define a compositional grammar.
- This is an *introduction* to formal semantics; we're about to do something *much* simpler (both in terms of syntax and semantics) than what Montague did (and what has been done since).
- Here: a context-free grammar (CFG) for syntax.

- Let us define a compositional grammar.
- This is an *introduction* to formal semantics; we're about to do something *much* simpler (both in terms of syntax and semantics) than what Montague did (and what has been done since).
- Here: a context-free grammar (CFG) for syntax.

Ex. CFG (excluding lexical rules):

- Let us define a compositional grammar.
- This is an *introduction* to formal semantics; we're about to do something *much* simpler (both in terms of syntax and semantics) than what Montague did (and what has been done since).
- Here: a context-free grammar (CFG) for syntax.

Ex. CFG (excluding lexical rules):

$$S \rightarrow NP \quad VP$$

- Let us define a compositional grammar.
- This is an *introduction* to formal semantics; we're about to do something *much* simpler (both in terms of syntax and semantics) than what Montague did (and what has been done since).
- Here: a context-free grammar (CFG) for syntax.

Ex. CFG (excluding lexical rules):

$S \rightarrow NP \quad VP$

$NP \rightarrow DET \quad N$

- Let us define a compositional grammar.
- This is an *introduction* to formal semantics; we're about to do something *much* simpler (both in terms of syntax and semantics) than what Montague did (and what has been done since).
- Here: a context-free grammar (CFG) for syntax.

Ex. CFG (excluding lexical rules):

$S \rightarrow NP \quad VP$

$NP \rightarrow DET \quad N$

$NP \rightarrow PN$

- Let us define a compositional grammar.
- This is an *introduction* to formal semantics; we're about to do something *much* simpler (both in terms of syntax and semantics) than what Montague did (and what has been done since).
- Here: a context-free grammar (CFG) for syntax.

Ex. CFG (excluding lexical rules):

$S \rightarrow NP \quad VP$

$NP \rightarrow DET \quad N$

$NP \rightarrow PN$

$VP \rightarrow Vi$

- Let us define a compositional grammar.
- This is an *introduction* to formal semantics; we're about to do something *much* simpler (both in terms of syntax and semantics) than what Montague did (and what has been done since).
- Here: a context-free grammar (CFG) for syntax.

Ex. CFG (excluding lexical rules):

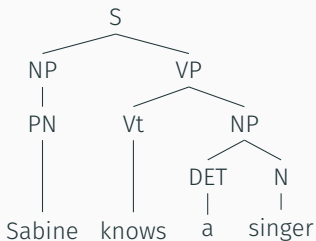
S	\rightarrow	NP	VP
NP	\rightarrow	DET	N
NP	\rightarrow	PN	
VP	\rightarrow	Vi	
VP	\rightarrow	Vt	NP

- Let us define a compositional grammar.
- This is an *introduction* to formal semantics; we're about to do something *much* simpler (both in terms of syntax and semantics) than what Montague did (and what has been done since).
- Here: a context-free grammar (CFG) for syntax.

Ex. CFG (excluding lexical rules):

S	\rightarrow	NP	VP
NP	\rightarrow	DET	N
NP	\rightarrow	PN	
VP	\rightarrow	V_i	
VP	\rightarrow	V_t	NP

Ex. tree:



- Let's pair each CFG rule with a semantic composition operation. \Rightarrow compositionality

- Let's pair each CFG rule with a semantic composition operation. \Rightarrow compositionality
- First try; something simple and intuitive:

- Let's pair each CFG rule with a semantic composition operation. \Rightarrow compositionality
- First try; something simple and intuitive:

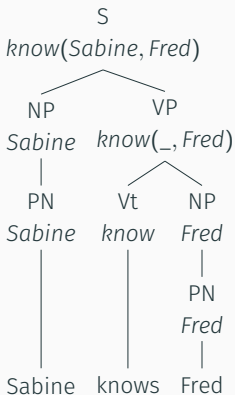
- Composition rules:

S	\rightarrow	NP	VP
$v(s, o)$	\leftarrow	s	$v(-, o)$
NP	\rightarrow	PN	
n	\leftarrow	n	
VP	\rightarrow	Vt	NP
$v(-, o)$	\leftarrow	v	o

- Let's pair each CFG rule with a semantic composition operation. \Rightarrow compositionality
- First try; something simple and intuitive:

- Composition rules:

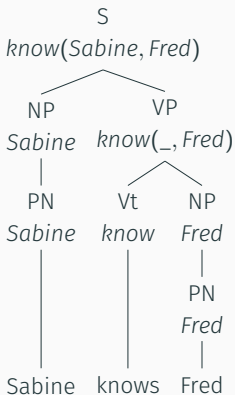
S	\rightarrow	NP	VP
$v(s, o)$	\leftarrow	s	$v(-, o)$
NP	\rightarrow	PN	
n	\leftarrow	n	
VP	\rightarrow	Vt	NP
$v(-, o)$	\leftarrow	v	o



- Let's pair each CFG rule with a semantic composition operation. \Rightarrow compositionality
- First try; something simple and intuitive:

- Composition rules:

S	\rightarrow	NP	VP
$v(s, o)$	\leftarrow	s	$v(-, o)$
NP	\rightarrow	PN	
n	\leftarrow	n	
VP	\rightarrow	Vt	NP
$v(-, o)$	\leftarrow	v	o



What is $know(-, Fred)$?

- Can we formalise $know(_, Fred)$?

- Can we formalise *know*(_, *Fred*)?
- What is the semantics of quantified NPs such as *every kid*?

- Can we formalise *know*(_, *Fred*)?
- What is the semantics of quantified NPs such as *every kid*?
- Some additional tool is needed.

λ -calculus

- λ -calculus is a formalisation of computation, developed by Alonzo Church in the 1930s.
- Reused by Montague and his followers in the context of NL to express semantic computation:
 - each lexical item is piece of program,
 - syntax indicates how to plug these pieces together, \rightarrow program
 - run the program. \rightarrow semantic representation

- λ -calculus is a formalisation of computation, developed by Alonzo Church in the 1930s.
- Reused by Montague and his followers in the context of NL to express semantic computation:
 - each lexical item is piece of program,
 - syntax indicates how to plug these pieces together, \rightarrow program
 - run the program. \rightarrow semantic representation
- '*know*(_, *Fred*)': $\lambda s. \textit{know}(s, \textit{Fred})$
 - This is a function.
 - Any other letter (i.e. *r*, or *t*, instead of *s*) is fine.

- β -reduction:
 - $(\lambda o. \lambda s. \textit{know}(s, o)) \textit{Fred}$

- β -reduction:
 - $(\lambda o. \lambda s. \textit{know}(s, o)) \textit{Fred} \rightarrow_{\beta}$

- β -reduction:

- $(\lambda o. \lambda s. \textit{know}(s, o)) \textit{Fred} \rightarrow_{\beta} \lambda s. \textit{know}(s, \textit{Fred})$

• β -reduction:

- $(\lambda o. \lambda s. \text{know}(s, o)) \text{Fred} \rightarrow_{\beta} \lambda s. \text{know}(s, \text{Fred})$
- $(\lambda s. \text{know}(s, \text{Fred})) \text{Sabine} \rightarrow_{\beta}$

- β -reduction:

- $(\lambda o. \lambda s. \text{know}(s, o)) \text{Fred} \rightarrow_{\beta} \lambda s. \text{know}(s, \text{Fred})$

- $(\lambda s. \text{know}(s, \text{Fred})) \text{Sabine} \rightarrow_{\beta} \text{know}(\text{Sabine}, \text{Fred})$

- β -reduction:

- $(\lambda o. \lambda s. \text{know}(s, o)) \text{Fred} \rightarrow_{\beta} \lambda s. \text{know}(s, \text{Fred})$

- $(\lambda s. \text{know}(s, \text{Fred})) \text{Sabine} \rightarrow_{\beta} \text{know}(\text{Sabine}, \text{Fred})$

(the actual definition is a bit technical)

- β -reduction:

- $(\lambda o. \lambda s. \textit{know}(s, o)) \textit{Fred} \rightarrow_{\beta} \lambda s. \textit{know}(s, \textit{Fred})$

- $(\lambda s. \textit{know}(s, \textit{Fred})) \textit{Sabine} \rightarrow_{\beta} \textit{know}(\textit{Sabine}, \textit{Fred})$

(the actual definition is a bit technical)

(the actual definition makes λ -calculus a Turing-complete programming language)

- A λ -term might expect anything as argument (not only constants).
Ex: $(\lambda P. P \textit{Marcel})$

- A λ -term might expect anything as argument (not only constants).
Ex: $(\lambda P. P \textit{Marcel}) (\lambda s. \textit{sleep}(s))$

- A λ -term might expect anything as argument (not only constants).
Ex: $(\lambda P. P \textit{ Marcel}) (\lambda s. \textit{ sleep}(s)) \rightarrow_{\beta} (\lambda s. \textit{ sleep}(s)) \textit{ Marcel}$

- A λ -term might expect anything as argument (not only constants).

Ex: $(\lambda P. P \textit{ Marcel}) (\lambda s. \textit{ sleep}(s)) \rightarrow_{\beta} (\lambda s. \textit{ sleep}(s)) \textit{ Marcel}$

$\rightarrow_{\beta} \textit{ sleep}(\textit{ Marcel})$

- A λ -term might expect anything as argument (not only constants).
Ex: $(\lambda P. P \textit{ Marcel}) (\lambda s. \textit{ sleep}(s)) \rightarrow_{\beta} (\lambda s. \textit{ sleep}(s)) \textit{ Marcel}$
 $\rightarrow_{\beta} \textit{ sleep}(\textit{ Marcel})$
- Existential quantifiers? (*some*, *a*)

- A λ -term might expect anything as argument (not only constants).
Ex: $(\lambda P. P \text{ Marcel}) (\lambda s. \text{sleep}(s)) \rightarrow_{\beta} (\lambda s. \text{sleep}(s)) \text{ Marcel}$
 $\rightarrow_{\beta} \text{sleep}(\text{Marcel})$
- Existential quantifiers? (*some*, *a*)
 - $[\text{some human is mortal}] = \exists x. \text{human}(x) \wedge \text{mortal}(x)$

- A λ -term might expect anything as argument (not only constants).

Ex: $(\lambda P. P \text{ Marcel}) (\lambda s. \text{sleep}(s)) \rightarrow_{\beta} (\lambda s. \text{sleep}(s)) \text{ Marcel}$

$\rightarrow_{\beta} \text{sleep}(\text{Marcel})$

- Existential quantifiers? (*some*, *a*)
 - $[\text{some human is mortal}] = \exists x. \text{human}(x) \wedge \text{mortal}(x)$
 - $[\text{some human}] =$

- A λ -term might expect anything as argument (not only constants).

Ex: $(\lambda P. P \text{ Marcel}) (\lambda s. \text{sleep}(s)) \rightarrow_{\beta} (\lambda s. \text{sleep}(s)) \text{ Marcel}$
 $\rightarrow_{\beta} \text{sleep}(\text{Marcel})$

- Existential quantifiers? (*some*, *a*)
 - [some human is mortal] = $\exists x. \text{human}(x) \wedge \text{mortal}(x)$
 - [some human] = $\lambda Q. \exists x. \text{human}(x) \wedge Qx$

- A λ -term might expect anything as argument (not only constants).

Ex: $(\lambda P. P \text{ Marcel}) (\lambda s. \text{sleep}(s)) \rightarrow_{\beta} (\lambda s. \text{sleep}(s)) \text{ Marcel}$

$\rightarrow_{\beta} \text{sleep}(\text{Marcel})$

- Existential quantifiers? (*some*, *a*)

- $[\text{some human is mortal}] = \exists x. \text{human}(x) \wedge \text{mortal}(x)$

- $[\text{some human}] = \lambda Q. \exists x. \text{human}(x) \wedge Qx$

- $[\text{some}] =$

- A λ -term might expect anything as argument (not only constants).

Ex: $(\lambda P. P \text{ Marcel}) (\lambda s. \text{sleep}(s)) \rightarrow_{\beta} (\lambda s. \text{sleep}(s)) \text{ Marcel}$
 $\rightarrow_{\beta} \text{sleep}(\text{Marcel})$

- Existential quantifiers? (*some*, *a*)
 - $[\text{some human is mortal}] = \exists x. \text{human}(x) \wedge \text{mortal}(x)$
 - $[\text{some human}] = \lambda Q. \exists x. \text{human}(x) \wedge Qx$
 - $[\text{some}] = \lambda P. \lambda Q. \exists x. Px \wedge Qx$

- A λ -term might expect anything as argument (not only constants).

Ex: $(\lambda P. P \text{ Marcel}) (\lambda s. \text{sleep}(s)) \rightarrow_{\beta} (\lambda s. \text{sleep}(s)) \text{ Marcel}$
 $\rightarrow_{\beta} \text{sleep}(\text{Marcel})$

- Existential quantifiers? (*some, a*)
 - $[\text{some human is mortal}] = \exists x. \text{human}(x) \wedge \text{mortal}(x)$
 - $[\text{some human}] = \lambda Q. \exists x. \text{human}(x) \wedge Q x$
 - $[\text{some}] = \lambda P. \lambda Q. \exists x. P x \wedge Q x$
- Universal quantifiers? (*every, all*)

- A λ -term might expect anything as argument (not only constants).

Ex: $(\lambda P. P \text{ Marcel}) (\lambda s. \text{sleep}(s)) \rightarrow_{\beta} (\lambda s. \text{sleep}(s)) \text{ Marcel}$
 $\rightarrow_{\beta} \text{sleep}(\text{Marcel})$

- Existential quantifiers? (*some, a*)

- [some human is mortal] = $\exists x. \text{human}(x) \wedge \text{mortal}(x)$
- [some human] = $\lambda Q. \exists x. \text{human}(x) \wedge Q x$
- [some] = $\lambda P. \lambda Q. \exists x. P x \wedge Q x$

- Universal quantifiers? (*every, all*)

- [every human is mortal] =

- A λ -term might expect anything as argument (not only constants).

Ex: $(\lambda P. P \text{ Marcel}) (\lambda s. \text{sleep}(s)) \rightarrow_{\beta} (\lambda s. \text{sleep}(s)) \text{ Marcel}$
 $\rightarrow_{\beta} \text{sleep}(\text{Marcel})$

- Existential quantifiers? (*some, a*)
 - [some human is mortal] = $\exists x. \text{human}(x) \wedge \text{mortal}(x)$
 - [some human] = $\lambda Q. \exists x. \text{human}(x) \wedge Q x$
 - [some] = $\lambda P. \lambda Q. \exists x. P x \wedge Q x$
- Universal quantifiers? (*every, all*)
 - [every human is mortal] = $\forall x. \text{human}(x) \rightarrow \text{mortal}(x)$

- A λ -term might expect anything as argument (not only constants).

Ex: $(\lambda P. P \text{ Marcel}) (\lambda s. \text{sleep}(s)) \rightarrow_{\beta} (\lambda s. \text{sleep}(s)) \text{ Marcel}$
 $\rightarrow_{\beta} \text{sleep}(\text{Marcel})$

- Existential quantifiers? (*some, a*)
 - [some human is mortal] = $\exists x. \text{human}(x) \wedge \text{mortal}(x)$
 - [some human] = $\lambda Q. \exists x. \text{human}(x) \wedge Qx$
 - [some] = $\lambda P. \lambda Q. \exists x. Px \wedge Qx$
- Universal quantifiers? (*every, all*)
 - [every human is mortal] = $\forall x. \text{human}(x) \rightarrow \text{mortal}(x)$
 - [every human] =

- A λ -term might expect anything as argument (not only constants).

Ex: $(\lambda P. P \text{ Marcel}) (\lambda s. \text{sleep}(s)) \rightarrow_{\beta} (\lambda s. \text{sleep}(s)) \text{ Marcel}$
 $\rightarrow_{\beta} \text{sleep}(\text{Marcel})$

- Existential quantifiers? (*some, a*)
 - [some human is mortal] = $\exists x. \text{human}(x) \wedge \text{mortal}(x)$
 - [some human] = $\lambda Q. \exists x. \text{human}(x) \wedge Qx$
 - [some] = $\lambda P. \lambda Q. \exists x. Px \wedge Qx$
- Universal quantifiers? (*every, all*)
 - [every human is mortal] = $\forall x. \text{human}(x) \rightarrow \text{mortal}(x)$
 - [every human] = $\lambda Q. \forall x. \text{human}(x) \rightarrow Qx$

- A λ -term might expect anything as argument (not only constants).

Ex: $(\lambda P. P \text{ Marcel}) (\lambda s. \text{sleep}(s)) \rightarrow_{\beta} (\lambda s. \text{sleep}(s)) \text{ Marcel}$
 $\rightarrow_{\beta} \text{sleep}(\text{Marcel})$

- Existential quantifiers? (*some, a*)

- [some human is mortal] = $\exists x. \text{human}(x) \wedge \text{mortal}(x)$
- [some human] = $\lambda Q. \exists x. \text{human}(x) \wedge Qx$
- [some] = $\lambda P. \lambda Q. \exists x. Px \wedge Qx$

- Universal quantifiers? (*every, all*)

- [every human is mortal] = $\forall x. \text{human}(x) \rightarrow \text{mortal}(x)$
- [every human] = $\lambda Q. \forall x. \text{human}(x) \rightarrow Qx$
- [every] =

- A λ -term might expect anything as argument (not only constants).

Ex: $(\lambda P. P \text{ Marcel}) (\lambda s. \text{sleep}(s)) \rightarrow_{\beta} (\lambda s. \text{sleep}(s)) \text{ Marcel}$
 $\rightarrow_{\beta} \text{sleep}(\text{Marcel})$

- Existential quantifiers? (*some, a*)

- [some human is mortal] = $\exists x. \text{human}(x) \wedge \text{mortal}(x)$
- [some human] = $\lambda Q. \exists x. \text{human}(x) \wedge Q x$
- [some] = $\lambda P. \lambda Q. \exists x. P x \wedge Q x$

- Universal quantifiers? (*every, all*)

- [every human is mortal] = $\forall x. \text{human}(x) \rightarrow \text{mortal}(x)$
- [every human] = $\lambda Q. \forall x. \text{human}(x) \rightarrow Q x$
- [every] = $\lambda P. \lambda Q. \forall x. P x \rightarrow Q x$

- β -reductions that are going to be useful:
 - $(\lambda P. \lambda Q. \exists x. P x \wedge Q x) (\lambda x. dog(x))$

- β -reductions that are going to be useful:

- $(\lambda P. \lambda Q. \exists x. P x \wedge Q x) (\lambda x. \text{dog}(x)) \rightarrow_{\beta} \lambda Q. \exists x. ((\lambda x. \text{dog}(x)) x) \wedge Q x$

- β -reductions that are going to be useful:

- $(\lambda P. \lambda Q. \exists x. P x \wedge Q x) (\lambda x. \text{dog}(x)) \rightarrow_{\beta} \lambda Q. \exists x. ((\lambda x. \text{dog}(x)) x) \wedge Q x$
 $\rightarrow_{\beta} \lambda Q. \exists x. \text{dog}(x) \wedge Q x$

- β -reductions that are going to be useful:

- $(\lambda P. \lambda Q. \exists x. P x \wedge Q x) (\lambda x. dog(x)) \rightarrow_{\beta} \lambda Q. \exists x. ((\lambda x. dog(x)) x) \wedge Q x$
 $\rightarrow_{\beta} \lambda Q. \exists x. dog(x) \wedge Q x$
- $\lambda s. ((\lambda Q. \exists x. dog(x) \wedge Q x) (\lambda o. like(s, o)))$

- β -reductions that are going to be useful:

- $(\lambda P. \lambda Q. \exists x. P x \wedge Q x) (\lambda x. \text{dog}(x)) \rightarrow_{\beta} \lambda Q. \exists x. ((\lambda x. \text{dog}(x)) x) \wedge Q x$
 $\rightarrow_{\beta} \lambda Q. \exists x. \text{dog}(x) \wedge Q x$
- $\lambda s. ((\lambda Q. \exists x. \text{dog}(x) \wedge Q x) (\lambda o. \text{like}(s, o)))$
 $\rightarrow_{\beta} \lambda s. \exists x. \text{dog}(x) \wedge ((\lambda o. \text{like}(s, o)) x)$

- β -reductions that are going to be useful:

- $(\lambda P. \lambda Q. \exists x. P x \wedge Q x) (\lambda x. dog(x)) \rightarrow_{\beta} \lambda Q. \exists x. ((\lambda x. dog(x)) x) \wedge Q x$
 $\rightarrow_{\beta} \lambda Q. \exists x. dog(x) \wedge Q x$

- $\lambda s. ((\lambda Q. \exists x. dog(x) \wedge Q x) (\lambda o. like(s, o)))$

- $\rightarrow_{\beta} \lambda s. \exists x. dog(x) \wedge ((\lambda o. like(s, o)) x) \rightarrow_{\beta} \lambda s. \exists x. dog(x) \wedge like(s, x)$

- β -reductions that are going to be useful:

- $(\lambda P. \lambda Q. \exists x. P x \wedge Q x) (\lambda x. \text{dog}(x)) \rightarrow_{\beta} \lambda Q. \exists x. ((\lambda x. \text{dog}(x)) x) \wedge Q x$
 $\rightarrow_{\beta} \lambda Q. \exists x. \text{dog}(x) \wedge Q x$
- $\lambda s. ((\lambda Q. \exists x. \text{dog}(x) \wedge Q x) (\lambda o. \text{like}(s, o)))$
 $\rightarrow_{\beta} \lambda s. \exists x. \text{dog}(x) \wedge ((\lambda o. \text{like}(s, o)) x) \rightarrow_{\beta} \lambda s. \exists x. \text{dog}(x) \wedge \text{like}(s, x)$
- $(\lambda Q. \exists x. \text{cat}(x) \wedge Q x) (\lambda s. \exists y. \text{dog}(y) \wedge \text{like}(s, y))$

- β -reductions that are going to be useful:

- $(\lambda P. \lambda Q. \exists x. P x \wedge Q x) (\lambda x. \text{dog}(x)) \rightarrow_{\beta} \lambda Q. \exists x. ((\lambda x. \text{dog}(x)) x) \wedge Q x$
 $\rightarrow_{\beta} \lambda Q. \exists x. \text{dog}(x) \wedge Q x$

- $\lambda s. ((\lambda Q. \exists x. \text{dog}(x) \wedge Q x) (\lambda o. \text{like}(s, o)))$
 $\rightarrow_{\beta} \lambda s. \exists x. \text{dog}(x) \wedge ((\lambda o. \text{like}(s, o)) x) \rightarrow_{\beta} \lambda s. \exists x. \text{dog}(x) \wedge \text{like}(s, x)$

- $(\lambda Q. \exists x. \text{cat}(x) \wedge Q x) (\lambda s. \exists y. \text{dog}(y) \wedge \text{like}(s, y))$
 $\rightarrow_{\beta} \exists x. \text{cat}(x) \wedge ((\lambda s. \exists y. \text{dog}(y) \wedge \text{like}(s, y)) x)$

- β -reductions that are going to be useful:

- $(\lambda P. \lambda Q. \exists x. P x \wedge Q x) (\lambda x. \text{dog}(x)) \rightarrow_{\beta} \lambda Q. \exists x. ((\lambda x. \text{dog}(x)) x) \wedge Q x$
 $\rightarrow_{\beta} \lambda Q. \exists x. \text{dog}(x) \wedge Q x$

- $\lambda s. ((\lambda Q. \exists x. \text{dog}(x) \wedge Q x) (\lambda o. \text{like}(s, o)))$
 $\rightarrow_{\beta} \lambda s. \exists x. \text{dog}(x) \wedge ((\lambda o. \text{like}(s, o)) x) \rightarrow_{\beta} \lambda s. \exists x. \text{dog}(x) \wedge \text{like}(s, x)$

- $(\lambda Q. \exists x. \text{cat}(x) \wedge Q x) (\lambda s. \exists y. \text{dog}(y) \wedge \text{like}(s, y))$
 $\rightarrow_{\beta} \exists x. \text{cat}(x) \wedge ((\lambda s. \exists y. \text{dog}(y) \wedge \text{like}(s, y)) x)$
 $\rightarrow_{\beta} \exists x. \text{cat}(x) \wedge \exists y. \text{dog}(y) \wedge \text{like}(x, y)$

α -conversion

- *Bound* variables can be renamed:
 - $\lambda x. Pxy \equiv \lambda z. Pzy$,
 - $\exists x. Pxy \equiv \exists z. Pzy$,
- During some β -reductions, they must be renamed, to avoid 'clashes':

$(\lambda Q. \exists x. \text{cat}(x) \wedge Qx) (\lambda s. \exists x. \text{dog}(x) \wedge \text{like}(s, x))$

$\not\rightarrow_{\beta} \exists x. \text{cat}(x) \wedge \exists x. \text{dog}(x) \wedge \text{like}(x, x)$

$\rightarrow_{\beta} \exists x. \text{cat}(x) \wedge \exists y. \text{dog}(y) \wedge \text{like}(x, y)$

(Defining when/how to do this is what makes defining β -reduction precisely technical.)

Putting everything together

• Lexicon:

$\llbracket \text{every}_{DET} \rrbracket = \lambda P. \lambda Q. \forall x. Px \rightarrow Qx$

$\llbracket \text{a}_{DET} \rrbracket = \lambda P. \lambda Q. \exists x. Px \wedge Qx$

$\llbracket \text{cat}_N \rrbracket = \lambda x. \text{cat}(x)$

$\llbracket \text{dog}_N \rrbracket = \lambda x. \text{dog}(x)$

$\llbracket \text{Jamy}_{PN} \rrbracket = \text{Jamy}$

$\llbracket \text{Fred}_{PN} \rrbracket = \text{Fred}$

$\llbracket \text{sleeps}_{V_i} \rrbracket = \lambda s. \text{sleep}(s)$

$\llbracket \text{likes}_{V_t} \rrbracket = \lambda s. \lambda o. \text{likes}(s, o)$

• Composition rules:

S	\rightarrow	NP	VP
sv	\leftarrow	s	v
NP	\rightarrow	DET	N
dn	\leftarrow	d	n
NP	\rightarrow	PN	
$\lambda P. Pn$	\leftarrow	n	
VP	\rightarrow	V_i	
v	\leftarrow	v	
VP	\rightarrow	V_t	NP
$\lambda s. (O(\lambda o. vs o))$	\leftarrow	v	O

• Lexicon:

$\llbracket \text{every}_{DET} \rrbracket = \lambda P. \lambda Q. \forall x. P x \rightarrow Q x$

$\llbracket a_{DET} \rrbracket = \lambda P. \lambda Q. \exists x. P x \wedge Q x$

$\llbracket \text{cat}_N \rrbracket = \lambda x. \text{cat}(x)$

$\llbracket \text{dog}_N \rrbracket = \lambda x. \text{dog}(x)$

$\llbracket \text{Jamy}_{PN} \rrbracket = \text{Jamy}$

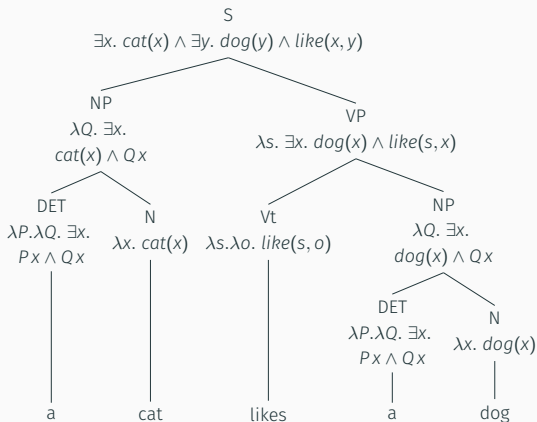
$\llbracket \text{Fred}_{PN} \rrbracket = \text{Fred}$

$\llbracket \text{sleeps}_{V_i} \rrbracket = \lambda s. \text{sleep}(s)$

$\llbracket \text{likes}_{V_t} \rrbracket = \lambda s. \lambda o. \text{likes}(s, o)$

• Composition rules:

S	\rightarrow	NP	VP
sv	\leftarrow	s	v
NP	\rightarrow	DET	N
dn	\leftarrow	d	n
NP	\rightarrow	PN	
$\lambda P. P n$	\leftarrow	n	
VP	\rightarrow	V_i	
v	\leftarrow	v	
VP	\rightarrow	V_t	NP
$\lambda s. (O(\lambda o. v s o))$	\leftarrow	v	O



• Lexicon:

$\llbracket \text{every}_{DET} \rrbracket = \lambda P. \lambda Q. \forall x. Px \rightarrow Qx$

$\llbracket a_{DET} \rrbracket = \lambda P. \lambda Q. \exists x. Px \wedge Qx$

$\llbracket \text{cat}_N \rrbracket = \lambda x. \text{cat}(x)$

$\llbracket \text{dog}_N \rrbracket = \lambda x. \text{dog}(x)$

$\llbracket \text{Jamy}_{PN} \rrbracket = \text{Jamy}$

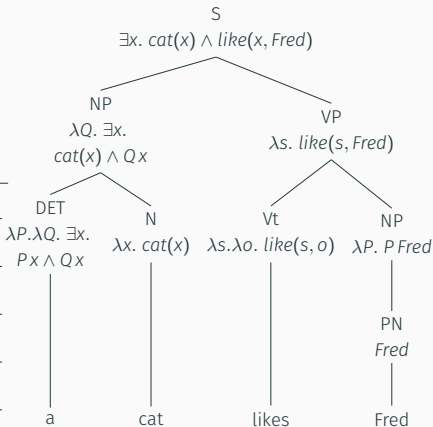
$\llbracket \text{Fred}_{PN} \rrbracket = \text{Fred}$

$\llbracket \text{sleeps}_{V_i} \rrbracket = \lambda s. \text{sleep}(s)$

$\llbracket \text{likes}_{V_t} \rrbracket = \lambda s. \lambda o. \text{likes}(s, o)$

• Composition rules:

S	→	NP	VP
sv	←	s	v
NP	→	DET	N
dn	←	d	n
NP	→	PN	
$\lambda P. Pn$	←	n	
VP	→	V _i	
v	←	v	
VP	→	V _t	NP
$\lambda s. (O(\lambda o. vs o))$	←	v	O



• Lexicon:

$\llbracket \text{every}_{DET} \rrbracket = \lambda P. \lambda Q. \forall x. Px \rightarrow Qx$

$\llbracket a_{DET} \rrbracket = \lambda P. \lambda Q. \exists x. Px \wedge Qx$

$\llbracket \text{cat}_N \rrbracket = \lambda x. \text{cat}(x)$

$\llbracket \text{dog}_N \rrbracket = \lambda x. \text{dog}(x)$

$\llbracket \text{Jamy}_{PN} \rrbracket = \text{Jamy}$

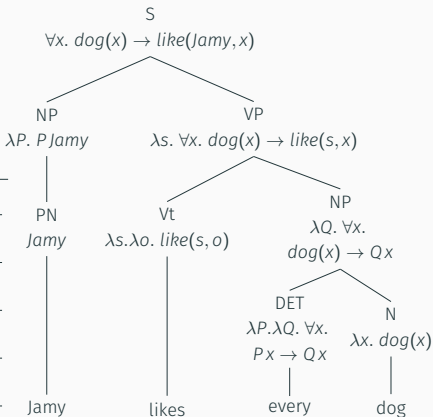
$\llbracket \text{Fred}_{PN} \rrbracket = \text{Fred}$

$\llbracket \text{sleeps}_{V_i} \rrbracket = \lambda s. \text{sleep}(s)$

$\llbracket \text{likes}_{V_t} \rrbracket = \lambda s. \lambda o. \text{likes}(s, o)$

• Composition rules:

S	\rightarrow	NP	VP
sv	\leftarrow	s	v
NP	\rightarrow	DET	N
dn	\leftarrow	d	n
NP	\rightarrow	PN	
$\lambda P. Pn$	\leftarrow	n	
VP	\rightarrow	V_i	
v	\leftarrow	v	
VP	\rightarrow	V_t	NP
$\lambda s. (O(\lambda o. vs o))$	\leftarrow	v	O



• Lexicon:

$\llbracket \text{every}_{DET} \rrbracket = \lambda P. \lambda Q. \forall x. P x \rightarrow Q x$

$\llbracket a_{DET} \rrbracket = \lambda P. \lambda Q. \exists x. P x \wedge Q x$

$\llbracket \text{cat}_N \rrbracket = \lambda x. \text{cat}(x)$

$\llbracket \text{dog}_N \rrbracket = \lambda x. \text{dog}(x)$

$\llbracket \text{Jamy}_{PN} \rrbracket = \text{Jamy}$

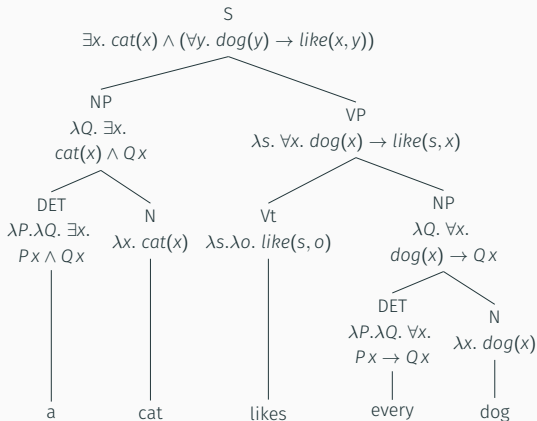
$\llbracket \text{Fred}_{PN} \rrbracket = \text{Fred}$

$\llbracket \text{sleeps}_{V_i} \rrbracket = \lambda s. \text{sleep}(s)$

$\llbracket \text{likes}_{V_t} \rrbracket = \lambda s. \lambda o. \text{likes}(s, o)$

• Composition rules:

S	\rightarrow	NP	VP
sv	\leftarrow	s	v
NP	\rightarrow	DET	N
dn	\leftarrow	d	n
NP	\rightarrow	PN	
$\lambda P. P n$	\leftarrow	n	
VP	\rightarrow	V_i	
v	\leftarrow	v	
VP	\rightarrow	V_t	NP
$\lambda s. (O(\lambda o. v s o))$	\leftarrow	v	O



• Lexicon:

$\llbracket \text{every}_{DET} \rrbracket = \lambda P. \lambda Q. \forall x. Px \rightarrow Qx$

$\llbracket a_{DET} \rrbracket = \lambda P. \lambda Q. \exists x. Px \wedge Qx$

$\llbracket \text{cat}_N \rrbracket = \lambda x. \text{cat}(x)$

$\llbracket \text{dog}_N \rrbracket = \lambda x. \text{dog}(x)$

$\llbracket \text{Jamy}_{PN} \rrbracket = \text{Jamy}$

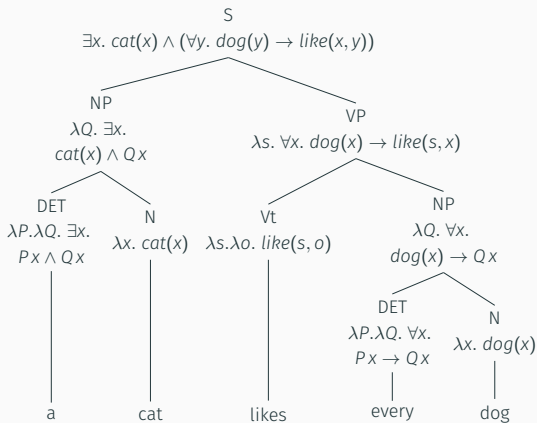
$\llbracket \text{Fred}_{PN} \rrbracket = \text{Fred}$

$\llbracket \text{sleeps}_{V_i} \rrbracket = \lambda s. \text{sleep}(s)$

$\llbracket \text{likes}_{V_t} \rrbracket = \lambda s. \lambda o. \text{likes}(s, o)$

• Composition rules:

S	\rightarrow	NP	VP
sv	\leftarrow	s	v
NP	\rightarrow	DET	N
dn	\leftarrow	d	n
NP	\rightarrow	PN	
$\lambda P. Pn$	\leftarrow	n	
VP	\rightarrow	V_i	
v	\leftarrow	v	
VP	\rightarrow	V_t	NP
$\lambda s. (O(\lambda o. v s o))$	\leftarrow	v	O



Is this translation satisfying?

- According to Partee (2014), the problem of quantifier scope ambiguity played a key role in the history of semantics and: *[it is] a difficult problem for every theory of the syntax-semantics interface. The basic problem is that if one accepts the principle of compositionality, then an ambiguous sentence like [Every student read one book.] must have two different syntactic structures, even though there may be no independent syntactic evidence of ambiguity.*

- According to Partee (2014), the problem of quantifier scope ambiguity played a key role in the history of semantics and: *[it is] a difficult problem for every theory of the syntax-semantics interface. The basic problem is that if one accepts the principle of compositionality, then an ambiguous sentence like [Every student read one book.] must have two different syntactic structures, even though there may be no independent syntactic evidence of ambiguity.*
- Many solutions have been proposed; a.o.:
 - Montague translates syntactic *derivations* (not trees),
 - quantifier raising (May 1977),
 - using a logical language which allows under-specified quantifier scope (e.g. Minimal Recursion Semantics; Copestake et al. 2005).

Natural language inference

- There are natural language processing (NLP) systems that produce semantic representations (logic-based or otherwise).

- There are natural language processing (NLP) systems that produce semantic representations (logic-based or otherwise).
- How to test them (at scale)?

- There are natural language processing (NLP) systems that produce semantic representations (logic-based or otherwise).
- How to test them (at scale)?
- 'Reasoning' datasets:
 - FraCas (Cooper et al. [1996](#)).
 - Recognizing textual entailment (RTE; Dagan, Glickman and Magnini [2006](#)).
 - Natural language inference (NLI; Bowman et al. [2015](#)).

- There are natural language processing (NLP) systems that produce semantic representations (logic-based or otherwise).
- How to test them (at scale)?
- 'Reasoning' datasets:
 - FraCas (Cooper et al. [1996](#)).
 - Recognizing textual entailment (RTE; Dagan, Glickman and Magnini [2006](#)).
 - Natural language inference (NLI; Bowman et al. [2015](#)).
- Logic vs common-sense reasoning.

FraCas 3.66

- Every resident of the North American continent can travel freely within Europe. Every Canadian resident is a resident of the North American continent.
- Can every Canadian resident freely within Europe?

FraCas 3.66

- Every resident of the North American continent can travel freely within Europe. Every Canadian resident is a resident of the North American continent.
- Can every Canadian resident freely within Europe?

FraCas 3.38

- No delegate finished the report.
- Did any delegate finish the report on time?

PASCAL RTE 1550

- T: In 1998, the General Assembly of the Nippon Sei Ko Kai (Anglican Church in Japan) voted to accept female priests.
- H: The Anglican church in Japan approved the ordination of women.

PASCAL RTE 1550

- T: In 1998, the General Assembly of the Nippon Sei Ko Kai (Anglican Church in Japan) voted to accept female priests.
- H: The Anglican church in Japan approved the ordination of women.

PASCAL RTE 731

- T: The city Tenochtitlan grew rapidly and was the center of the Aztec's great empire.
- H: Tenochtitlan quickly spread over the island, marshes, and swamps.

NLI

- A soccer game with multiple males playing.
- Some men are playing a sport.

NLI

- A soccer game with multiple males playing.
- Some men are playing a sport.

NLI

- A smiling costumed woman is holding an umbrella.
- A happy woman in a fairy costume holds an umbrella.

NLI

- A soccer game with multiple males playing.
- Some men are playing a sport.

NLI

- A smiling costumed woman is holding an umbrella.
- A happy woman in a fairy costume holds an umbrella.

NLI

- A man inspects the uniform of a figure in some East Asian country.
- The man is sleeping.

- Basis of the formal approach (e.g. Bos and Markert 2005; Haruta, Mineshima and Bekki 2022):
 - syntactic parsing (usually with CCG; Steedman and Baldridge 2011),
 - conversion to logical formulas,
 - gathering axioms about lexical- and world-knowledge,
 - theorem proving.

- Basis of the formal approach (e.g. Bos and Markert 2005; Haruta, Mineshima and Bekki 2022):
 - syntactic parsing (usually with CCG; Steedman and Baldridge 2011),
 - conversion to logical formulas,
 - gathering axioms about lexical- and world-knowledge,
 - theorem proving.
- The formal approach works well on most logic-oriented datasets.
- Not so much on common-sense reasoning ones.

- Basis of the formal approach (e.g. Bos and Markert 2005; Haruta, Mineshima and Bekki 2022):
 - syntactic parsing (usually with CCG; Steedman and Baldridge 2011),
 - conversion to logical formulas,
 - gathering axioms about lexical- and world-knowledge,
 - theorem proving.
- The formal approach works well on most logic-oriented datasets.
- Not so much on common-sense reasoning ones.
- Why?

- Syntactic parsing is hard (and compositional semantics is heavily dependent on syntax).

- Syntactic parsing is hard (and compositional semantics is heavily dependent on syntax).
 - Formal semantics is hard.
- (7)
- a. The kids saw three cows {each/in total}.
 - b. Cédric is a good mathematician but a bad politician.
 - c. The rabbit in the hat has long ears.
 - d. I saw Marcel not stop at the traffic light.
 - e. What she said is more true than what he said.

- Syntactic parsing is hard (and compositional semantics is heavily dependent on syntax).
 - Formal semantics is hard.
- (7)
- a. The kids saw three cows {each/in total}.
 - b. Cédric is a good mathematician but a bad politician.
 - c. The rabbit in the hat has long ears.
 - d. I saw Marcel not stop at the traffic light.
 - e. What she said is more true than what he said.
- Listing enough lexical- and world-knowledge axioms is hard.
 - Common sense reasoning and pragmatics is hard.
 - Most of our concepts (e.g. being in holiday, what a country is) are fuzzy. Most of our expectations (e.g. birds can fly, dogs have four legs) are cancelable.

- Scaling symbolic approaches (within a language but also across languages) is very hard.

- Scaling symbolic approaches (within a language but also across languages) is very hard.
- Contemporary neural approaches seem to fair better on common-sense reasoning. (but not the same kind of errors)

- Scaling symbolic approaches (within a language but also across languages) is very hard.
- Contemporary neural approaches seem to fair better on common-sense reasoning. (but not the same kind of errors)
- Basis of the neural approach:
 - formatting the problem so that it can be sent to a text encoder (e.g. BERT; Devlin et al. [2019](#)),
 - feeding the vector encoding of the text to a classifier (entailment/neutral/contradiction, yes/no/idk).

- Scaling symbolic approaches (within a language but also across languages) is very hard.
- Contemporary neural approaches seem to fair better on common-sense reasoning. (but not the same kind of errors)
- Basis of the neural approach:
 - formatting the problem so that it can be sent to a text encoder (e.g. BERT; Devlin et al. [2019](#)),
 - feeding the vector encoding of the text to a classifier (entailment/neutral/contradiction, yes/no/idk).
- No symbolic representations, only vectors.
- What are these?

Distributional semantics

- Harris (1954):
each language can be described in terms of a distributional structure, i.e. in terms of the occurrence of parts (ultimately sounds) relative to other parts

- Harris (1954):

each language can be described in terms of a distributional structure, i.e. in terms of the occurrence of parts (ultimately sounds) relative to other parts [...] The distribution of an element will be understood as the sum of all its environments. An environment of an element A is an existing array of its co-occurents, i.e. the other elements, each in a particular position, with which A occurs to yield an utterance.

- Harris (1954):
each language can be described in terms of a distributional structure, i.e. in terms of the occurrence of parts (ultimately sounds) relative to other parts [...] The distribution of an element will be understood as the sum of all its environments. An environment of an element A is an existing array of its co-occurents, i.e. the other elements, each in a particular position, with which A occurs to yield an utterance.
- A language is structured (different types of items can be defined, and these items are variously related to each other).
- This structure can be uncovered by studying items' distribution.

- Relation with structuralism (de Saussure 1916; Ducrot 1973)?

- Relation with structuralism (de Saussure 1916; Ducrot 1973)?
 - Related questions.
 - Related methods.

- Relation with structuralism (de Saussure 1916; Ducrot 1973)?
 - Related questions.
 - Related methods.
 - More emphasis on distributions, less on meaning.

- Distributional hypothesis (in semantics): the meaning of an item is strongly correlated with its distribution.
- Harris (1954): 'certain aspects of meaning [are] functions of measurable distributional relations'

- Distributional hypothesis (in semantics): the meaning of an item is strongly correlated with its distribution.
- Harris (1954): 'certain aspects of meaning [are] functions of measurable distributional relations'
- Strong version: you can extract the meaning of a word from its distribution in a large enough corpus.

- Semantic space (Osgood, Suci and Tannenbaum [1957](#)): words are represented as vector, each dimension corresponds to some opposition (e.g. happy/sad, soft/hard, slow/fast).
- Not built distributionally.

- In information retrieval (IR): development of vector representations of documents based on document-term cooccurrence matrices (Salton, Wong and Yang [1975](#); Dumais et al. [1988](#)).

- In information retrieval (IR): development of vector representations of documents based on document-term cooccurrence matrices (Salton, Wong and Yang 1975; Dumais et al. 1988).
- Birth of distributional semantics: convergence of distributionalism, semantic spaces and IR methods?

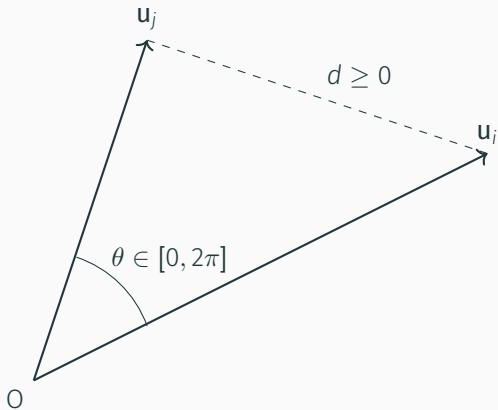
- Distribution: set of contexts
→ impractical definition; need for a simplification/approximation
- Basic implementation (Schütze 1993; Lund and Burgess 1996):
 - vocabulary of forms $V = \{w_1, w_2, \dots, w_N\}$,
 - corpus (sequence of forms) $T = t_1 t_2 \dots t_n$,
 - *window size* $s \geq 1$,

- Distribution: set of contexts
→ impractical definition; need for a simplification/approximation
- Basic implementation (Schütze 1993; Lund and Burgess 1996):
 - vocabulary of forms $V = \{w_1, w_2, \dots, w_N\}$,
 - corpus (sequence of forms) $T = t_1 t_2 \dots t_n$,
 - *window size* $s \geq 1$,
 - $\forall i, j \leq N, A_{i,j} =$ number of times w_i occurs at distance $\leq s$ from w_j in T ,

- Distribution: set of contexts
→ impractical definition; need for a simplification/approximation
- Basic implementation (Schütze 1993; Lund and Burgess 1996):
 - vocabulary of forms $V = \{w_1, w_2, \dots, w_N\}$,
 - corpus (sequence of forms) $T = t_1 t_2 \dots t_n$,
 - window size $s \geq 1$,
 - $\forall i, j \leq N, A_{i,j} =$ number of times w_i occurs at distance $\leq s$ from w_j in T ,
 - $u_i = [A_{i,1}, A_{i,2}, \dots, A_{i,N}]$ is a representation of the distribution of w_i ,
→ vector space of dimension $N = |V|$

- Distribution: set of contexts
→ impractical definition; need for a simplification/approximation
- Basic implementation (Schütze 1993; Lund and Burgess 1996):
 - vocabulary of forms $V = \{w_1, w_2, \dots, w_N\}$,
 - corpus (sequence of forms) $T = t_1 t_2 \dots t_n$,
 - *window size* $s \geq 1$,
 - $\forall i, j \leq N, A_{i,j} =$ number of times w_i occurs at distance $\leq s$ from w_j in T ,
 - $u_i = [A_{i,1}, A_{i,2}, \dots, A_{i,N}]$ is a representation of the distribution of w_i ,
→ vector space of dimension $N = |V|$
 - *dimensionality reduction*
→ vector space of arbitrary dimension N'

- Distribution: set of contexts
→ impractical definition; need for a simplification/approximation
- Basic implementation (Schütze 1993; Lund and Burgess 1996):
 - vocabulary of forms $V = \{w_1, w_2, \dots, w_N\}$,
 - corpus (sequence of forms) $T = t_1 t_2 \dots t_n$,
 - *window size* $s \geq 1$,
 - $\forall i, j \leq N, A_{i,j} =$ number of times w_i occurs at distance $\leq s$ from w_j in T ,
 - $u_i = [A_{i,1}, A_{i,2}, \dots, A_{i,N}]$ is a representation of the distribution of w_i ,
→ vector space of dimension $N = |V|$
 - *dimensionality reduction*
→ vector space of arbitrary dimension N'
- semantic space?



- Euclidean distance = $d = |u_i - u_j|$
- Cosine similarity = $\cos(\theta) = \cos((u_i, u_j))$

- Human similarity judgements on words correlate with similarity in the vector space.
Ex: *queen* is more similar to *prince* than to *bread*.

- Human similarity judgements on words correlate with similarity in the vector space.
Ex: *queen* is more similar to *prince* than to *bread*.
- Given an ambiguous word in context (*glass*) in *I'd like a ___ of water*, the sum/average of the u_i of the context helps a lot in the disambiguation.

- Human similarity judgements on words correlate with similarity in the vector space.
Ex: *queen* is more similar to *prince* than to *bread*.
- Given an ambiguous word in context (*glass*) in *I'd like a ___ of water*, the sum/average of the u_i of the context helps a lot in the disambiguation.
- Forms are also (more or less) clustered by part-of-speech.

- Dimensionality reduction?
 - too large vectors are computationally impractical
 - lot of redundancy in the original dimensions
 - reduction finds generates a smaller and less redundant space
 - intuition: some dimensions are combined, others are dropped
 - the original space can be (approximately) reconstructed
 - similar topology in the original and new spaces

- Dimensionality reduction?
 - too large vectors are computationally impractical
 - lot of redundancy in the original dimensions
 - reduction finds generates a smaller and less redundant space
 - intuition: some dimensions are combined, others are dropped
 - the original space can be (approximately) reconstructed
 - similar topology in the original and new spaces
- Various spaces:
 - cooccurrence criterion,
 - reduction operation.

- Machine learning systems infer *parameters* from *data*.

- Machine learning systems infer *parameters* from *data*.
- Maybe word representations can be inferred?

- Machine learning systems infer *parameters* from *data*.
- Maybe word representations can be inferred?
- *Language models* (LM) have always been useful for natural language generation (in e.g. machine translation, speech recognition).
- Bengio, Ducharme and Vincent (2001) suggested to replace traditional *n-gram models* with neural models.

n-gram LM

$$P(t_i | t_{i-n+1}, t_{i-n+2}, \dots, t_{i-1}) \approx \frac{\#t_{i-n+1}t_{i-n+2}\dots t_{i-1}t_i}{\#t_{i-n+1}t_{i-n+2}\dots t_{i-1}}$$

Neural LM

classifier using a concatenation of *word embeddings* as input

$$P(t_i | t_{i-n+1}, t_{i-n+2}, \dots, t_{i-1}) = f([u_{t_{i-n+1}} \cdot u_{t_{i-n+2}} \cdot \dots \cdot u_{t_{i-1}}])$$

n-gram LM

$$P(t_i | t_{i-n+1}, t_{i-n+2}, \dots, t_{i-1}) \approx \frac{\#t_{i-n+1}t_{i-n+2}\dots t_{i-1}t_i}{\#t_{i-n+1}t_{i-n+2}\dots t_{i-1}}$$

Neural LM

classifier using a concatenation of *word embeddings* as input

$$P(t_i | t_{i-n+1}, t_{i-n+2}, \dots, t_{i-1}) = f([u_{t_{i-n+1}} \cdot u_{t_{i-n+2}} \cdot \dots \cdot u_{t_{i-1}}])$$

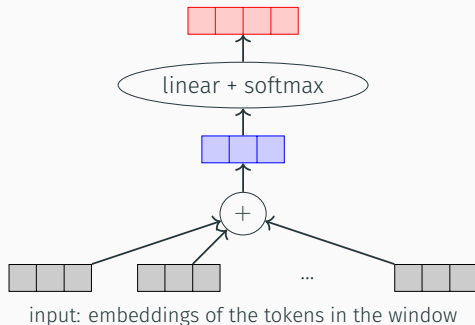
→ these u_i are distributional

- Mikolov et al. (2013) go ‘full distributional’:
 - left and right context,
 - no interest in language modelling, but in representations.

- Mikolov et al. (2013) go ‘full distributional’:
 - left and right context,
 - no interest in language modelling, but in representations.
- Word2Vec (CBOW) uses a window around the form to predict:

- Mikolov et al. (2013) go ‘full distributional’:
 - left and right context,
 - no interest in language modelling, but in representations.
- Word2Vec (CBOW) uses a window around the form to predict:

output: probability distribution over the vocabulary



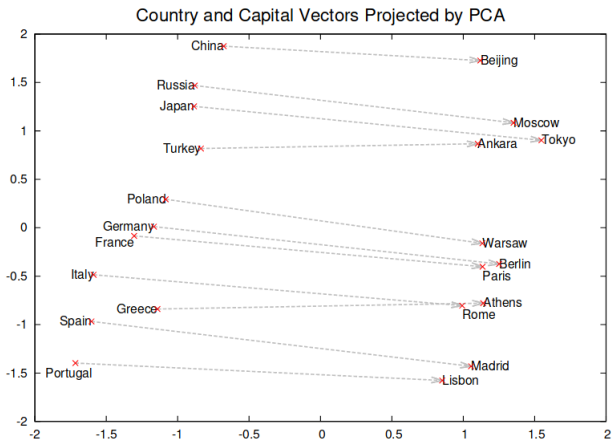


Figure 2: Two-dimensional PCA projection of the 1000-dimensional Skip-gram vectors of countries and their capital cities. The figure illustrates ability of the model to automatically organize concepts and learn implicitly the relationships between them, as during the training we did not provide any supervised information about what a capital city means.

(from Mikolov et al. 2013)

Analogyes

- $Berlin \approx (Paris - France) + Germany$
- $queen \approx (king - man) + female$
- $walking \approx (swimming - swam) + walked$

- Embedding spaces can be visualised with methods such as t-SNE (van der Maaten and Hinton [2008](#)).

- Embedding spaces can be visualised with methods such as t-SNE (van der Maaten and Hinton [2008](#)).
- Want to get intuitions about the evolution of a word?
Build different spaces, one per period of interest (Tainturier [2020](#), for the French *islamo-gauchisme* in the written medias).

- Embedding spaces can be visualised with methods such as t-SNE (van der Maaten and Hinton 2008).
- Want to get intuitions about the evolution of a word?
Build different spaces, one per period of interest (Tainturier 2020, for the French *islamo-gauchisme* in the written medias).
(other idea, that I have not tried: build a single space, but add the corresponding period at the end of the word; e.g. *unicorn_1990s* for occurrences of *unicorn* in the 1990s)

- Count-based and neural methods may look very different but still correspond to related distributional computations; (Levy and Goldberg 2014; Pennington, Socher and Manning 2014).
- For most NLP tasks (e.g. syntactic parsing), relatively little annotated data is available compared to the complexity of the task.
- Consequence: overall performance is *highly* dependant on the word representations used (i.a. Pennington, Socher and Manning 2014; Peters et al. 2018; Devlin et al. 2019).
- Distributional methods provide very useful representations.

- Word embeddings often contain morpho-syntactic and semantic information (i.a. Köhn [2015](#); Gupta et al. [2015](#); Gaddy, Stern and Klein [2018](#)).
- Contextual embeddings:
 - e.g. ELMo (Peters et al. [2018](#)) and BERT (Devlin et al. [2019](#)),
 - take the context into account,
 - contain information related to syntax and sentence semantics (Tenney et al. [2019](#)).

- Word embeddings often contain morpho-syntactic and semantic information (i.a. Köhn [2015](#); Gupta et al. [2015](#); Gaddy, Stern and Klein [2018](#)).
- Contextual embeddings:
 - e.g. ELMo (Peters et al. [2018](#)) and BERT (Devlin et al. [2019](#)),
 - take the context into account,
 - contain information related to syntax and sentence semantics (Tenney et al. [2019](#)).

(Count-based contextual distributional embeddings: Schütze [1992](#))

- Word embeddings often contain morpho-syntactic and semantic information (i.a. Köhn [2015](#); Gupta et al. [2015](#); Gaddy, Stern and Klein [2018](#)).
- Contextual embeddings:
 - e.g. ELMo (Peters et al. [2018](#)) and BERT (Devlin et al. [2019](#)),
 - take the context into account,
 - contain information related to syntax and sentence semantics (Tenney et al. [2019](#)).

(Count-based contextual distributional embeddings: Schütze [1992](#))



- How to disentangle the syntactic and semantic information? (Chen et al. [2019](#); J. Y. Huang, K.-H. Huang and Chang [2021](#))



- Distributional embeddings tend to encode harmful biases (i.a. Caliskan, Bryson and Narayanan 2017).
- Related problem: reporting bias
 - Gordon and Van Durme (2013): ‘researchers need to be aware that the frequency of occurrence of particular types of events or relations in text can represent significant distortions of real-world frequencies and that much of our general knowledge is never alluded to in natural discourse’
 - Daumé III (2016): ‘if you were to try to guess what color most sheep were by looking and language data, it would be very difficult for you to conclude that they weren’t almost all black. [...] In English, “black sheep” outnumbers “white sheep” about 25:1 (many “black sheep”s are movie references); in French it’s 3:1; in German it’s 12:1.’

(see also Shwartz and Choi 2020)





- TODO talk about:
 - grounding and multi-modality
 - the fact that formal semantics and distributional semantics seem complementary
 - directions for combining the two approaches

References




-  Bach, Emmon (1989). *Informal Lectures on Formal Semantics*. SUNY Series in Linguistics. State University of New York Press. URL: <https://sunypress.edu/Books/I/Informal-Lectures-on-Formal-Semantics2>.
-  Bender, Emily M. and Alexander Koller (2020). 'Climbing towards NLU: On Meaning, Form, and Understanding in the Age of Data'. In: *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*.





-  Bengio, Yoshua, Réjean Ducharme and Pascal Vincent (2001). **'A Neural Probabilistic Language Model'**. In: *Advances in Neural Information Processing Systems 13*. Ed. by T. K. Leen, T. G. Dietterich and V. Tresp. MIT Press, pp. 932–938. URL: <http://papers.nips.cc/paper/1839-a-neural-probabilistic-language-model.pdf>.
-  Bos, Johan and Katja Markert (2005). **'Recognising Textual Entailment with Logical Inference'**. In: *Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing*. HLT '05. Stroudsburg, PA, USA: Association for Computational Linguistics, pp. 628–635. DOI: [10.3115/1220575.1220654](https://doi.org/10.3115/1220575.1220654).

-  Bowman, Samuel R. et al. (2015). **'A large annotated corpus for learning natural language inference'**. In: *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, pp. 632–642. DOI: [10.18653/v1/D15-1075](https://doi.org/10.18653/v1/D15-1075).
-  Caliskan, Aylin, Joanna J. Bryson and Arvind Narayanan (2017). **'Semantics derived automatically from language corpora contain human-like biases'**. In: *Science* 356.6334, pp. 183–186. DOI: [10.1126/science.aal4230](https://doi.org/10.1126/science.aal4230).
-  Chen, Mingda et al. (2019). **'A Multi-Task Approach for Disentangling Syntax and Semantics in Sentence Representations'**. In: *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. Minneapolis, Minnesota: Association for Computational Linguistics, pp. 2453–2464. DOI: [10.18653/v1/N19-1254](https://doi.org/10.18653/v1/N19-1254).





-  Chomsky, Noam (1965). *Aspects of the Theory of Syntax*. Massachusetts Institute of Technology. Research Laboratory of Electronics. Special technical report 11. The MIT Press. URL: <https://www.jstor.org/stable/j.ctt17kk81z>.
-  Cooper, Robin et al. (1996). *Using the Framework*. Tech. rep. Deliverable D16. URL: <https://gu-clasp.github.io/multifracas/D16.pdf>.
-  Copestake, Ann et al. (2005). 'Minimal Recursion Semantics: An Introduction'. In: *Research on Language and Computation* 3.2-3, pp. 281–332. DOI: [10.1007/s11168-006-6327-9](https://doi.org/10.1007/s11168-006-6327-9).
-  Dagan, Ido, Oren Glickman and Bernardo Magnini (2006). 'The PASCAL Recognising Textual Entailment Challenge'. In: *Machine Learning Challenges. Evaluating Predictive Uncertainty, Visual Object Classification, and Recognising Tectual Entailment*. Ed. by Joaquin Quiñonero-Candela et al. Lecture Notes in Computer Science 3944. Springer Berlin Heidelberg, pp. 177–190. DOI: [10.1007/11736790_9](https://doi.org/10.1007/11736790_9).




-  Daumé III, Hal (2016). *Language bias and black sheep*. URL: <https://nlpers.blogspot.com/2016/06/language-bias-and-black-sheep.html>.
-  de Saussure, Ferdinand (1916). *Cours de linguistique générale*. Ed. by Charles Bally and Albert Sechehaye. Payot. URL: https://fr.wikisource.org/wiki/Cours_de_linguistique_g%C3%A9n%C3%A9rale.
-  Devlin, Jacob et al. (2019). ‘BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding’. In: *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. Minneapolis, Minnesota: Association for Computational Linguistics, pp. 4171–4186. DOI: [10.18653/v1/N19-1423](https://doi.org/10.18653/v1/N19-1423).
-  Ducrot, Oswald (1973). *Le Structuralisme en linguistique*. Éditions du Seuil.




-  Dumais, Susan T. et al. (1988). **'Using latent semantic analysis to improve access to textual information'**. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. CHI '88. Washington, D.C., USA: Association for Computing Machinery, pp. 281–285. DOI: [10.1145/57167.57214](https://doi.org/10.1145/57167.57214).
-  Gaddy, David, Mitchell Stern and Dan Klein (2018). **'What's Going On in Neural Constituency Parsers? An Analysis'**. In: *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*. New Orleans, Louisiana: Association for Computational Linguistics, pp. 999–1010. DOI: [10.18653/v1/N18-1091](https://doi.org/10.18653/v1/N18-1091).
-  Gamut, L. T. F. (1991). **Logic, Language, and Meaning**. Vol. 2: Intensional logic and logical grammar. University of Chicago Press. URL: <http://www.press.uchicago.edu/ucp/books/book/chicago/L/bo3628700.html>.



-  Gordon, Jonathan and Benjamin Van Durme (2013). **'Reporting bias and knowledge acquisition'**. In: *Proceedings of the 2013 workshop on Automated knowledge base construction*. AKBC '13. New York, NY, USA: Association for Computing Machinery, pp. 25–30. DOI: [10.1145/2509558.2509563](https://doi.org/10.1145/2509558.2509563).
-  Gupta, Abhijeet et al. (2015). **'Distributional vectors encode referential attributes'**. In: *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*. Lisbon, Portugal: Association for Computational Linguistics, pp. 12–21. DOI: [10.18653/v1/D15-1002](https://doi.org/10.18653/v1/D15-1002).
-  Harnad, Stevan (1990). **'The symbol grounding problem'**. In: *Physica D: Nonlinear Phenomena* 42.1. 04880, pp. 335–346. DOI: [10.1016/0167-2789\(90\)90087-6](https://doi.org/10.1016/0167-2789(90)90087-6).
-  Harris, Zellig S. (1954). **'Distributional Structure'**. In: *Word* 10.2-3, pp. 146–162. DOI: [10.1080/00437956.1954.11659520](https://doi.org/10.1080/00437956.1954.11659520).




-  Haruta, Izumi, Koji Mineshima and Daisuke Bekki (2022). **'Implementing Natural Language Inference for comparatives'**. In: *Journal of Language Modelling* 10.1. Number: 1, pp. 139–191. DOI: [10.15398/jlm.v10i1.294](https://doi.org/10.15398/jlm.v10i1.294).
-  Huang, James Y., Kuan-Hao Huang and Kai-Wei Chang (2021). **'Disentangling Semantics and Syntax in Sentence Embeddings with Pre-trained Language Models'**. In: *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. Online: Association for Computational Linguistics, pp. 1372–1379. DOI: [10.18653/v1/2021.naacl-main.108](https://doi.org/10.18653/v1/2021.naacl-main.108).
-  Köhn, Arne (2015). **'What's in an Embedding? Analyzing Word Embeddings through Multilingual Evaluation'**. In: *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*. Lisbon, Portugal: Association for Computational Linguistics, pp. 2067–2073. DOI: [10.18653/v1/D15-1246](https://doi.org/10.18653/v1/D15-1246).




-  Levy, Omer and Yoav Goldberg (2014). **'Neural Word Embedding as Implicit Matrix Factorization'**. In: *Advances in Neural Information Processing Systems 27*. Ed. by Z. Ghahramani et al. Curran Associates, Inc., pp. 2177–2185. URL: <http://papers.nips.cc/paper/5477-neural-word-embedding-as-implicit-matrix-factorization.pdf>.
-  Lewis, David (1970). **'General Semantics'**. In: *Synthese* 22, pp. 18–67. URL: <http://www.jstor.org/stable/20114749>.
-  Lund, Kevin and Curt Burgess (1996). **'Producing high-dimensional semantic spaces from lexical co-occurrence'**. In: *Behavior Research Methods, Instruments, & Computers* 28.2, pp. 203–208. DOI: [10.3758/BF03204766](https://doi.org/10.3758/BF03204766).
-  May, Robert Carlen (1977). **'The grammar of quantification'**. Ph.D. Thesis. Massachusetts Institute of Technology. URL: <https://dspace.mit.edu/handle/1721.1/16287>.



-  Mikolov, Tomas et al. (2013). **'Distributed Representations of Words and Phrases and their Compositionality'**. In: *Advances in Neural Information Processing Systems 26*. Ed. by C. J. C. Burges et al. Curran Associates, Inc., pp. 3111–3119. URL: <http://papers.nips.cc/paper/5021-distributed-representations-of-words-and-phrases-and-their-compositionality.pdf>.
-  Montague, Richard (1970a). **'English as a Formal Language'**. In: *Linguaggi nella Società e nella Tecnica*. Ed. by Bruno Visentini. Milan, Italy: Edizioni di Comunità, pp. 189–224.
-  Montague, Richard (1970b). **'Universal grammar'**. In: *Theoria* 36.3, pp. 373–398. DOI: [10.1111/j.1755-2567.1970.tb00434.x](https://doi.org/10.1111/j.1755-2567.1970.tb00434.x).

-  Montague, Richard (1973). **'The Proper Treatment of Quantification in Ordinary English'**. In: *Approaches to Natural Language: Proceedings of the 1970 Stanford Workshop on Grammar and Semantics*. Ed. by Jaakko Hintikka, Julius Moravcsik and Patrick Suppes. Synthese Library 49, pp. 221–242. DOI: [10.1007/978-94-010-2506-5_10](https://doi.org/10.1007/978-94-010-2506-5_10).
-  Osgood, Charles E., George J. Suci and Percy H. Tannenbaum (1957). ***The measurement of meaning***. Urban, Illinois, USA: University of Illinois Press.
-  Partee, Barbara Hall (2011). **'Formal Semantics: Origins, Issues, Early Impact'**. In: *Baltic International Yearbook of Cognition, Logic and Communication* 6 - Formal Semantics and Pragmatics: Discourse, Context, and Models, pp. 1–52. DOI: [10.4148/biyclc.v6i0.1580](https://doi.org/10.4148/biyclc.v6i0.1580).

-  Partee, Barbara Hall (2014). 'A brief history of the syntax-semantics interface in Western formal linguistics'. In: *Semantics-Syntax Interface* 1.1, Spring 2014, pp. 1–21. URL: <https://www.academia.edu/download/40460288/SSI11.2014.pdf>.
-  Pelletier, Francis Jeffrey (1994). 'The Principle of Semantic Compositionality'. In: *Topoi* 13.1, pp. 11–24. DOI: [10.1007/BF00763644](https://doi.org/10.1007/BF00763644).
-  Pennington, Jeffrey, Richard Socher and Christopher D. Manning (2014). 'GloVe: Global Vectors for Word Representation'. In: *Empirical Methods in Natural Language Processing (EMNLP)*. 11283, pp. 1532–1543.
-  Peters, Matthew E. et al. (2018). 'Deep Contextualized Word Representations'. In: *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*. Association for Computational Linguistics, pp. 2227–2237. DOI: [10.18653/v1/N18-1202](https://doi.org/10.18653/v1/N18-1202).

-  Salton, G., A. Wong and C. S. Yang (1975). **'A vector space model for automatic indexing'**. In: *Communications of the ACM* 18.11, pp. 613–620. DOI: [10.1145/361219.361220](https://doi.org/10.1145/361219.361220).
-  Schütze, Hinrich (1992). **'Word Space'**. In: *Advances in Neural Information Processing Systems*. Vol. 5. Morgan-Kaufmann. URL: https://proceedings.neurips.cc/paper_files/paper/1992/hash/d86ea612dec96096c5e0fcc8dd42ab6d-Abstract.html.
-  Schütze, Hinrich (1993). **'Distributed syntactic representations with an application to part-of-speech tagging'**. In: *Proceedings of the IEEE International Conference on Neural Networks*, pp. 1504–1509. DOI: [10.1109/ICNN.1993.298779](https://doi.org/10.1109/ICNN.1993.298779).

-  Shwartz, Vered and Yejin Choi (2020). ‘Do Neural Language Models Overcome Reporting Bias?’ In: *Proceedings of the 28th International Conference on Computational Linguistics*. Ed. by Donia Scott, Nuria Bel and Chengqing Zong. Barcelona, Spain (Online): International Committee on Computational Linguistics, pp. 6863–6870. DOI: [10.18653/v1/2020.coling-main.605](https://doi.org/10.18653/v1/2020.coling-main.605).
-  Steedman, Mark and Jason Baldrige (2011). ‘Combinatory Categorical Grammar’. In: *Non-Transformational Syntax*. Ed. by Robert D. Borsley and Kersti Börjars. Wiley-Blackwell, pp. 181–224. DOI: [10.1002/9781444395037.ch5](https://doi.org/10.1002/9781444395037.ch5).
-  Tainturier, Benjamin (2020). *Islamogauchisme : carrière médiatique d’une notion polémique*. URL: <https://larevuedesmedias.ina.fr/islamo-gauchisme-carriere-mediatique>.

-  Tenney, Ian et al. (2019). **‘What do you learn from context? Probing for sentence structure in contextualized word representations’**. In: *International Conference on Learning Representations*. URL: <https://openreview.net/forum?id=SJzSgnRcKX>.
-  van der Maaten, Laurens and Geoffrey Hinton (2008). **‘Visualizing data using t-SNE’**. In: *Journal of Machine Learning Research* 9, Nov, pp. 2579–2605. URL: <http://www.jmlr.org/papers/v9/vandermaaten08a.html>.