Compositional Quantum Natural Language Processing
Theoretical Foundations and Experimental Implementation

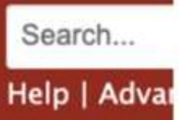
Muhammad Hamza Waseem
DPhil Physics, University of Oxford
Research Scientist at Quantinuum

Mathematics and Physics of Quantum Computing and Quantum Learning
Porquerolles, France

May 23-28, 2025

QUANTINUUM

UNIVERSITY OF OXFORD



Computer Science > Computation and Language

[Submitted on 6 Apr 2019 (v1), last revised 28 Feb 2020 (this version, v2)]

The Mathematics of Text Structure

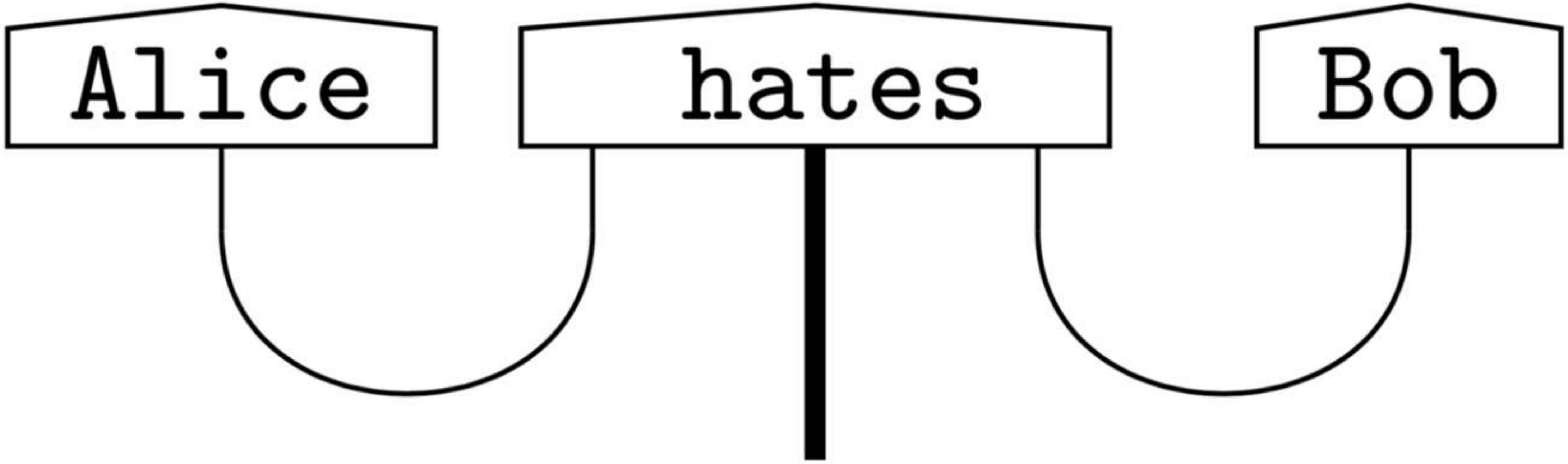
Bob Coecke

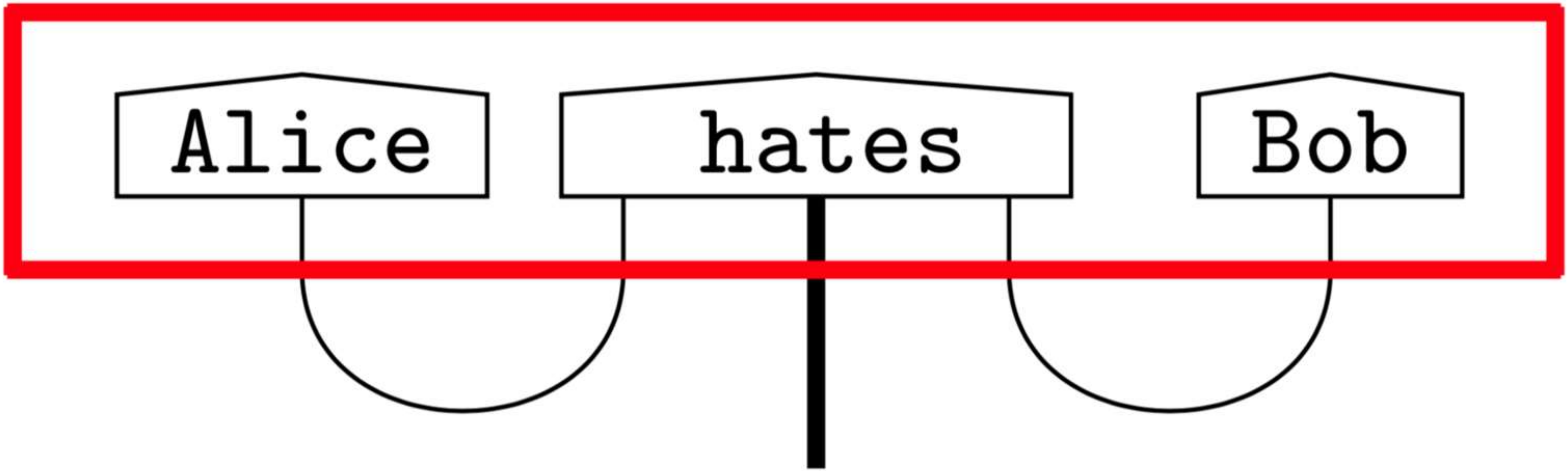
In previous work we gave a mathematical foundation, referred to as DisCoCat, for how words interact in a sentence in order to produce the meaning of that sentence. To do so, we exploited the perfect structural match of grammar and categories of meaning spaces. Here, we give a mathematical foundation, referred to as DisCoCirc, for how sentences interact in texts in order to produce the meaning of that text. First we revisit DisCoCat. While in DisCoCat all meanings are fixed as states (i.e. have no input), in DisCoCirc word meanings correspond to a type, or system, and the states of this system can evolve. Sentences are gates within a circuit which update the variable meanings of those words. Like in DisCoCat, word meanings can live in a variety of spaces e.g. propositional, vectorial, or cognitive. The compositional structure are string diagrams representing information flows, and an entire text yields a single string diagram in which word meanings lift to the meaning of an entire text. While the developments in this paper are independent of a physical embodiment (cf. classical vs. quantum computing), both the compositional formalism and suggested meaning model are highly quantum-inspired, and implementation on a quantum computer would come with a range of benefits. We also praise Jim Lambek for his role in mathematical linguistics in general, and the development of the DisCo program more specifically.

The main idea

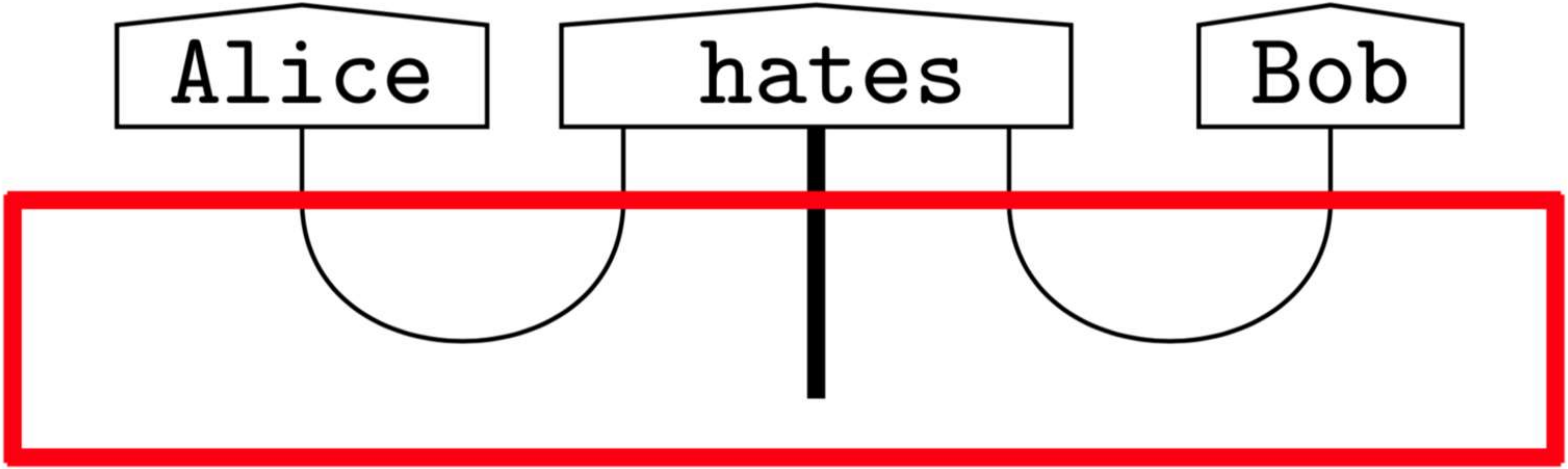
There are dictionaries for words.

There are dictionaries for words.
Why aren't there any dictionaries for sentences?


Alice
hates
Bob



Alice
hates
Bob


Alice
hates
Bob

Why aren't there any dictionaries for entire texts?

a. How sentence meanings compose in order to form the meaning of text.

a. How sentence meanings compose in order to form the meaning of text.
b. How word meanings evolve in text, when learning new things.

a. How sentence meanings compose in order to form the meaning of text.
b. How word meanings evolve in text, when learning new things.
c. What the type of the sentence meaning space is.

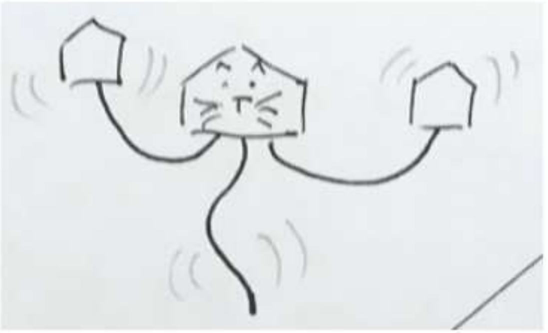
1. Word-meanings: states $\mapsto$ types

.

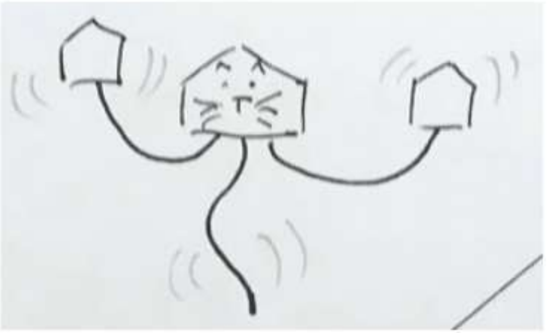
1. Word-meanings: states $\mapsto$ types
2. Sentence-meanings: states $\mapsto$ I/O-processes

1. Word-meanings: states $\mapsto$ types
2. Sentence-meanings: states $\mapsto$ I/O-processes
3. Text-meaning: $\varnothing$ $\mapsto$ circuit

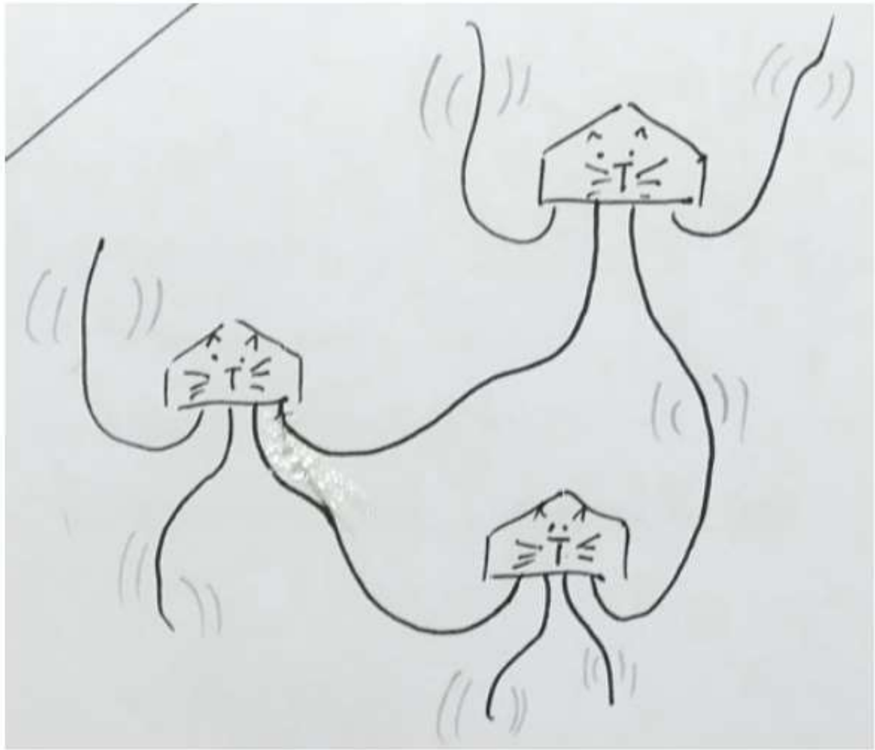
.



DisCoCat



DisCoCat

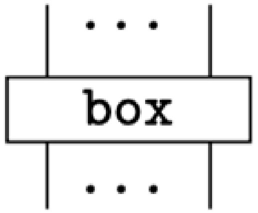


DisCoCirc

Diagrams

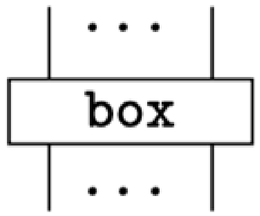
Diagrams

made up of boxes

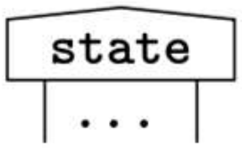


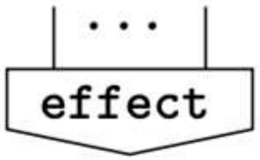
Diagrams

made up of boxes

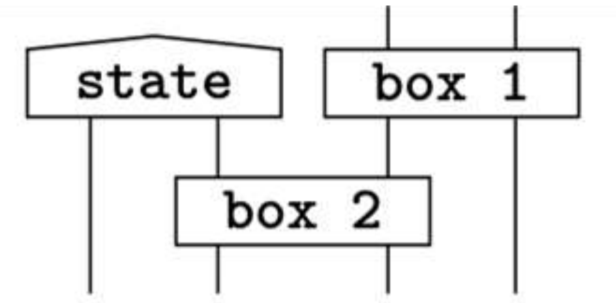


Special boxes

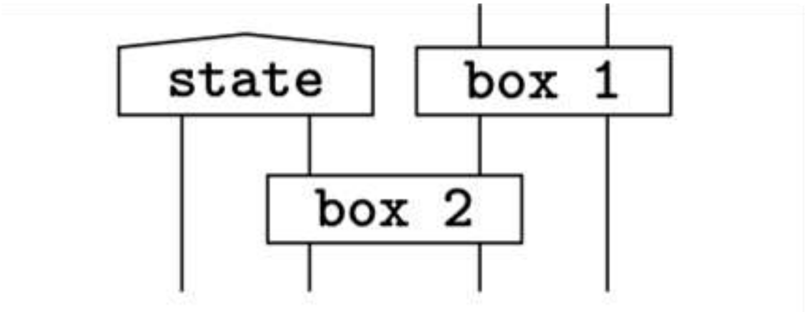




General Diagrams



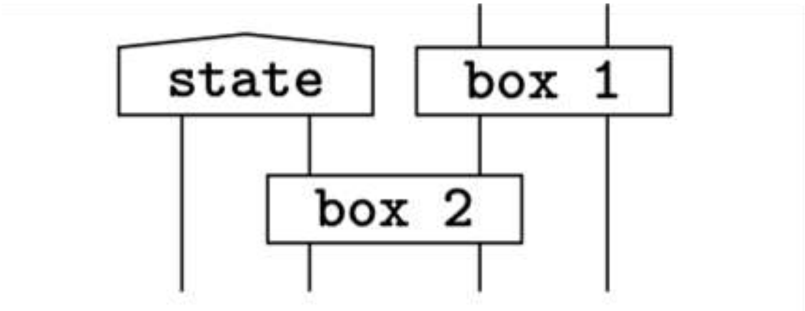
General Diagrams



determined by

- The connectedness of the wire-structure
- Labels on wires and boxes

General Diagrams



determined by

- The connectedness of the wire-structure
- Labels on wires and boxes

They live in monoidal categories where the wires correspond to objects and the boxes correspond to morphisms

Circuits

- are obtained by composing boxes in parallel and sequentially.

Circuits

- are obtained by composing boxes in parallel and sequentially.
- live in symmetric monoidal category

Circuits

- are obtained by composing boxes in parallel and sequentially.
- live in symmetric monoidal category
- admit a clear flow of time: outputs connected to inputs of future processes

String diagrams

allow for outputs to be connected to inputs or even to other outputs. Likewise, inputs can be connected to inputs.

String diagrams

allow for outputs to be connected to inputs or even to other outputs. Likewise, inputs can be connected to inputs.

Enabled by caps and cups

String diagrams

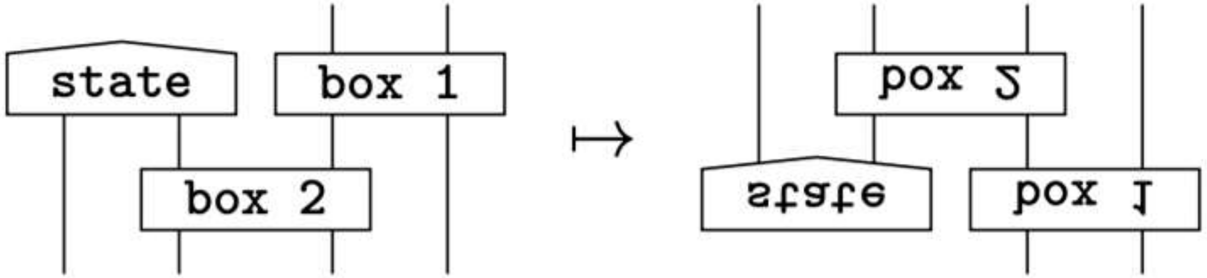
allow for outputs to be connected to inputs or even to other outputs.
Likewise, inputs can be connected to inputs.

Enabled by caps and cups

String diagrams live in a compact closed category.

String diagrams

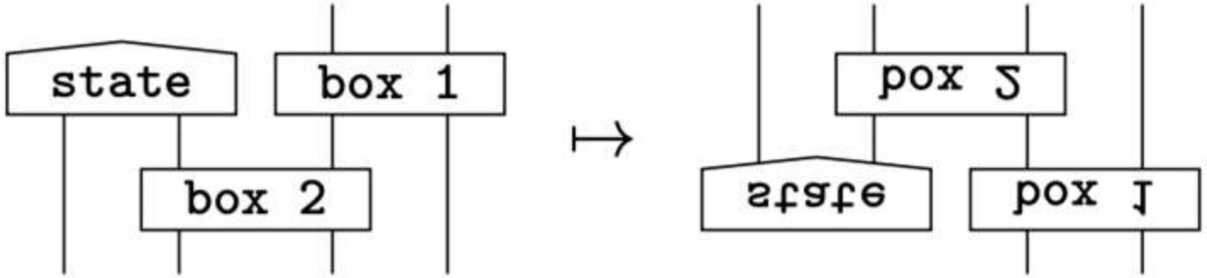
We usually assume string diagrams can be flipped vertically.



.

String diagrams

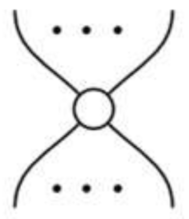
We usually assume string diagrams can be flipped vertically.



Dagger structure or adjoints

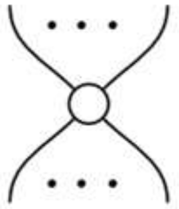
Spiders

a.k.a. dagger special commutative Frobenius algebras

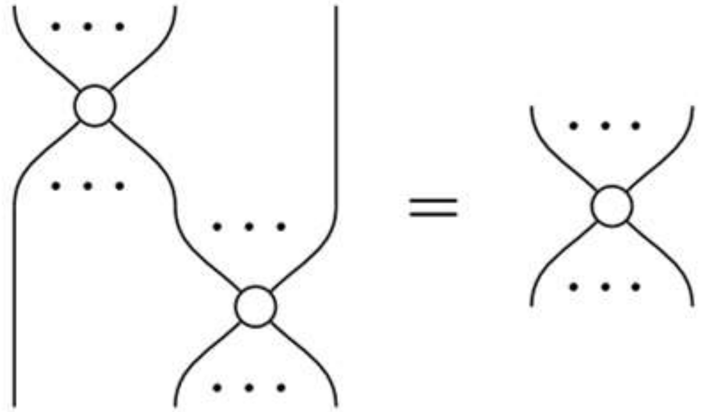


Spiders

a.k.a. dagger special commutative Frobenius algebras



Spider fusion



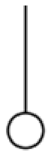
Spiders

Logical AND

Spiders

Logical AND

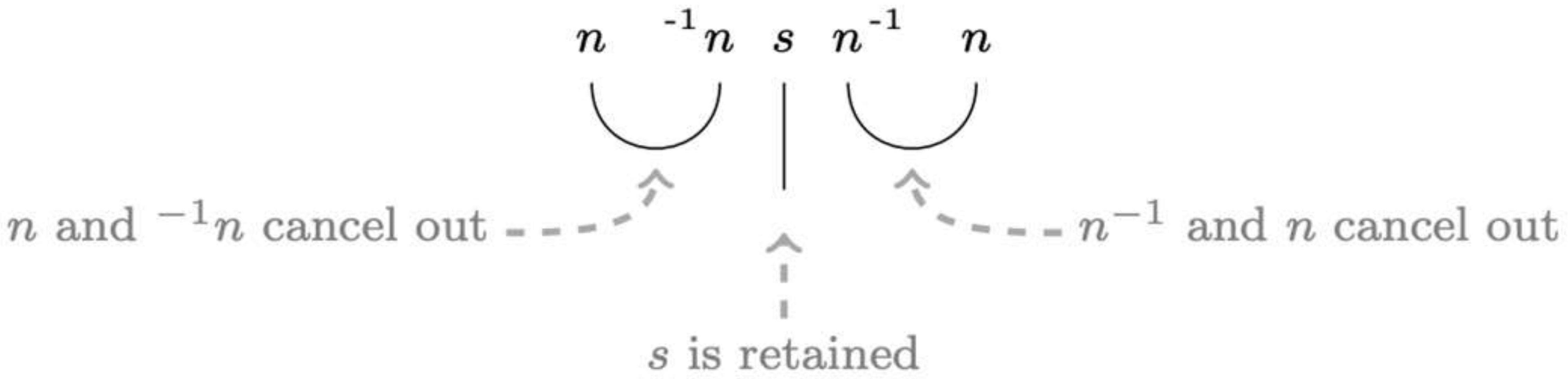
Discarding



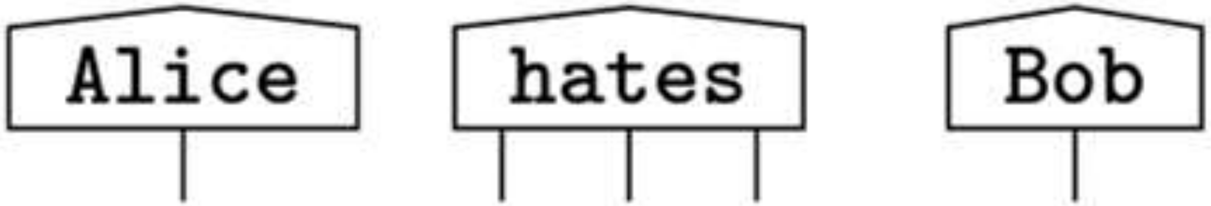
DisCoCat

$$\underbrace{\texttt{Alice}}_{n}\ \underbrace{\texttt{hates}}_{n^{-1}\cdot s\cdot n^{-1}}\ \underbrace{\texttt{Bob}}_{n}$$

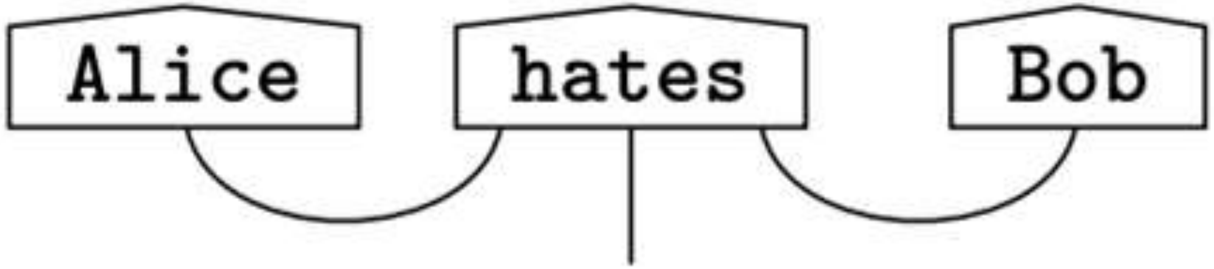
.



n ^{-1}n s n^{-1} n
n and ^{-1}n cancel out
n^{-1} and n cancel out
s is retained


Alice
hates
Bob

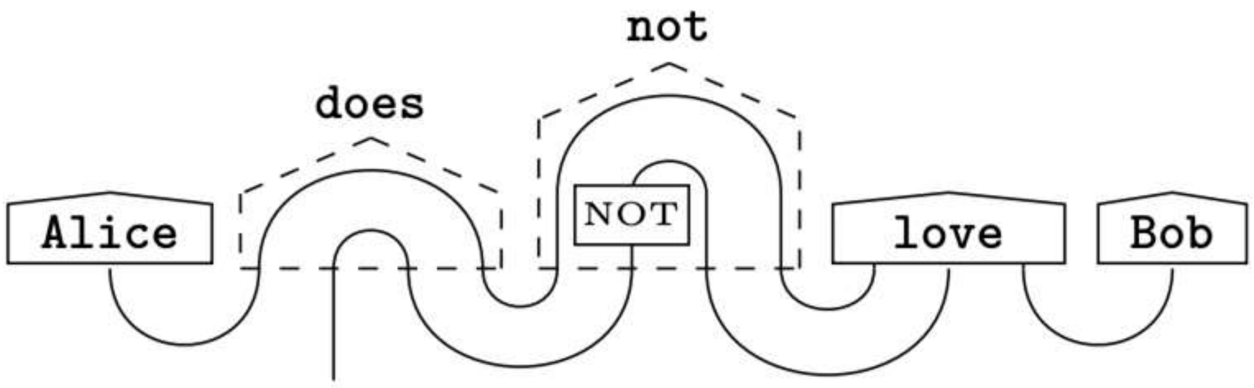
.


Alice
hates
Bob

.

Internal wirings of meanings

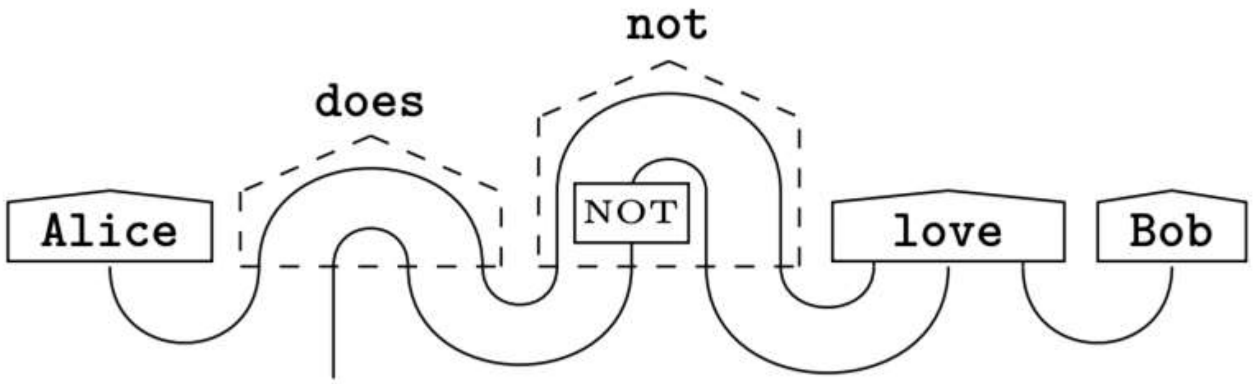
Functional words

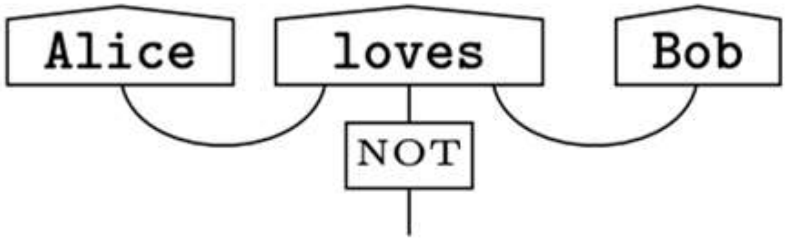


.

Internal wirings of meanings

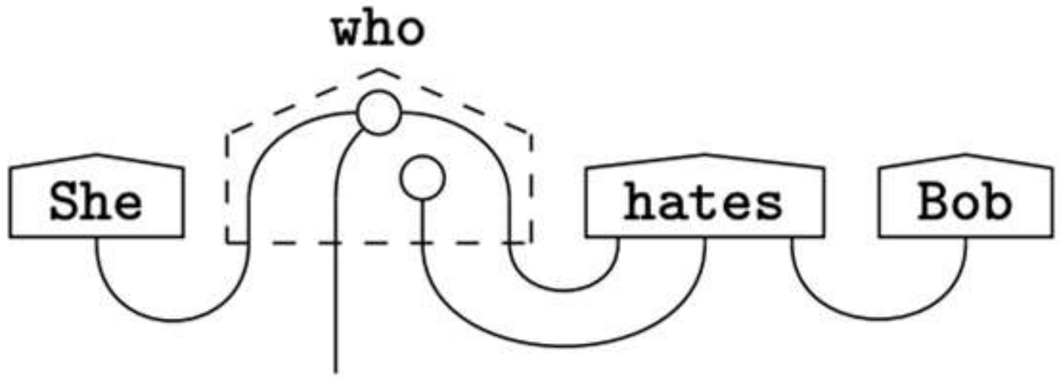
Functional words





Internal wirings of meanings

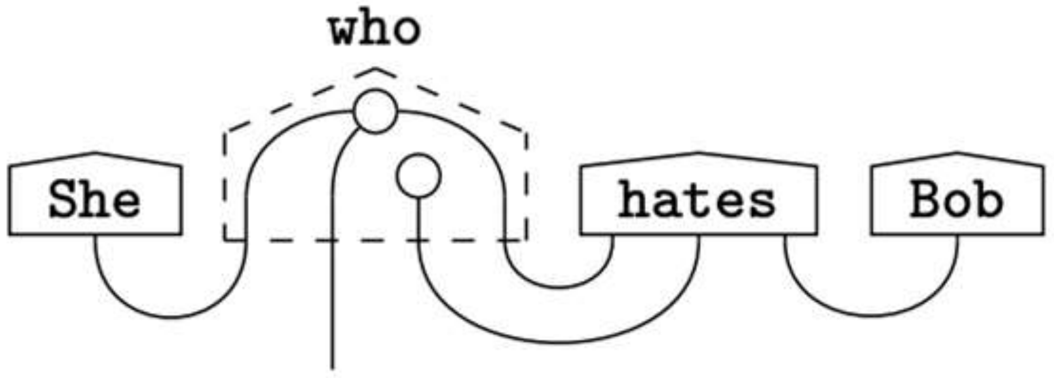
Functional words

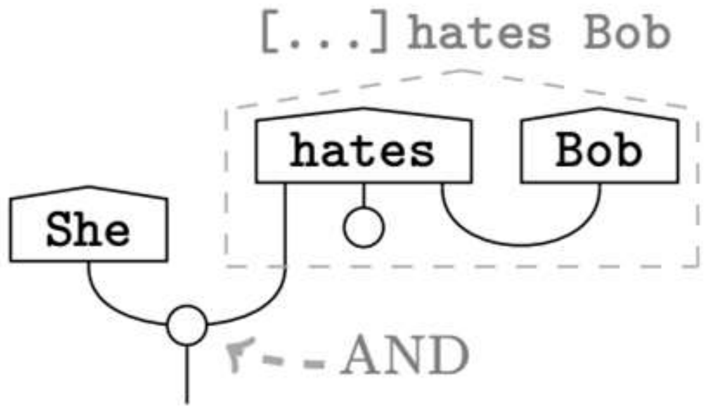


.

Internal wirings of meanings

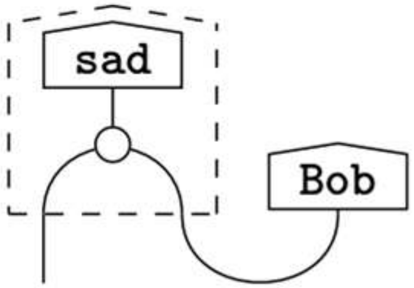
Functional words





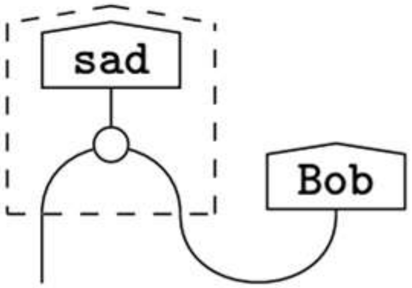
Internal wirings of meanings

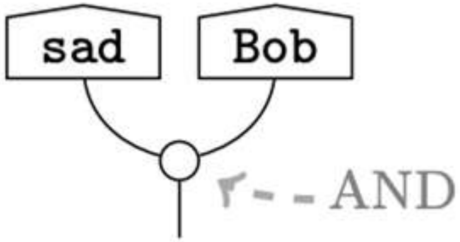
Adjectives



Internal wirings of meanings

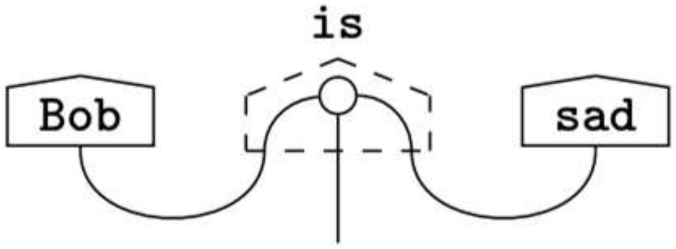
Adjectives





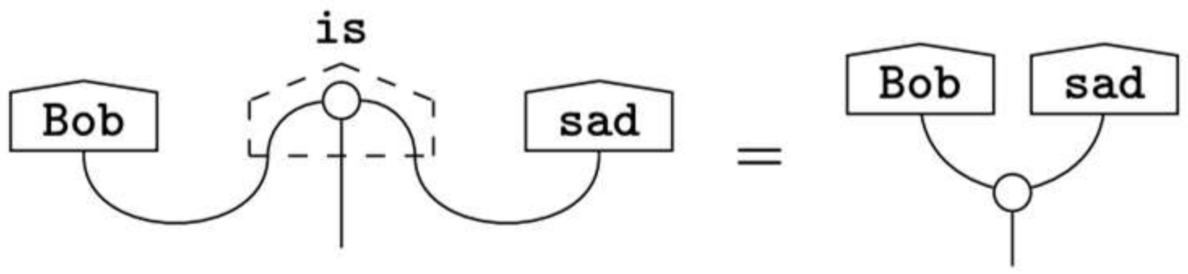
Internal wirings of meanings

`to be’



Internal wirings of meanings

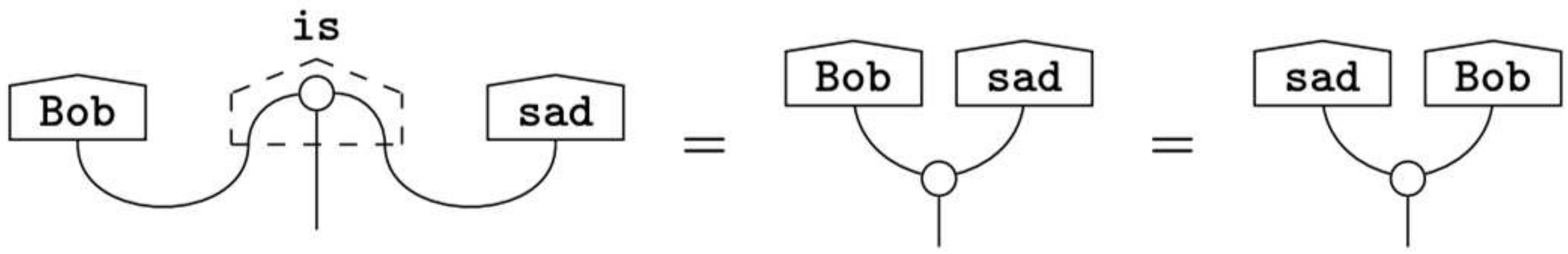
`to be’



.

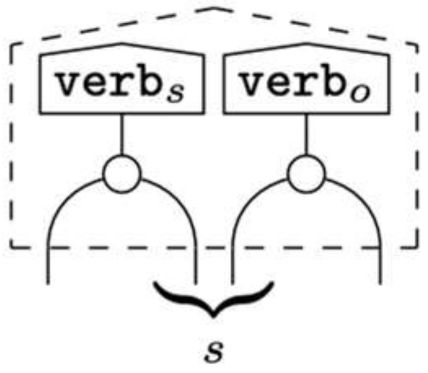
Internal wirings of meanings

`to be'



Internal wirings of meanings

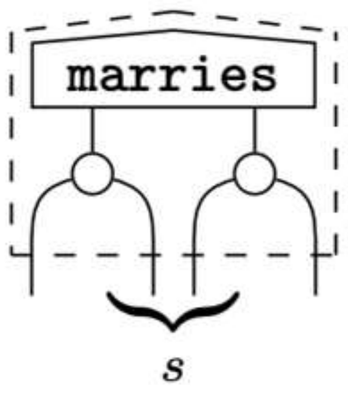
semi-Cartesian verbs



.

Internal wirings of meanings

Compact verbs



Models of meaning

Diagrams live in abstract (a.k.a. axiomatic) categories.

Models of meaning

Diagrams live in abstract (a.k.a. axiomatic) categories.

They allow for a wide range of concrete models.

Models of meaning

Diagrams live in abstract (a.k.a. axiomatic) categories.

They allow for a wide range of concrete models.

Pick a concrete category with cups and caps.

Models of meaning

Diagrams live in abstract (a.k.a. axiomatic) categories.

They allow for a wide range of concrete models.

Pick a concrete category with cups and caps.

Then wires become objects (a.k.a. spaces) and boxes become the morphisms (a.k.a. maps between these spaces)

Models of meaning

In NLP, the vector space model takes wires to be spaces of distributions and boxes to be linear maps.

Models of meaning

In NLP, the vector space model takes wires to be spaces of distributions and boxes to be linear maps.

Distributions are empirically established, by means of counting co-occurrences with a selected set of basis words.

Models of meaning

In NLP, the vector space model takes wires to be spaces of distributions and boxes to be linear maps.

Distributions are empirically established, by means of counting co-occurrences with a selected set of basis words.

Using this model in DisCoCat:

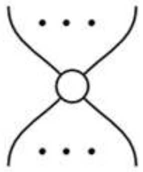
$$:= \sum_i |ii\rangle$$

$$:= \sum_i \langle ii|$$

Models of meaning

Spiders are in one-to-one correspondence with orthonormal bases

$\{|i\rangle\}_i$

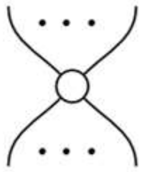


$$:= \sum_i |i \ldots i\rangle\langle i \ldots i|$$

Models of meaning

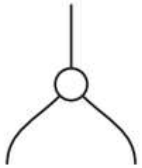
Spiders are in one-to-one correspondence with orthonormal bases

$$\{|i\rangle\}_i$$



$$:= \sum_i |i \dots i\rangle\langle i \dots i|$$

Copy and merge



$$:= \sum_i |ii\rangle\langle i|$$

$$:= \sum_i |i\rangle\langle ii|$$

Models of meaning

Other models:

Models of meaning

Other models:

1. Sets and relations

relations as Boolean-valued matrices
possibilistic model

Models of meaning

Other models:

1. Sets and relations

   relations as Boolean-valued matrices
   possibilistic model

1. Density matrices and superoperators

Models of meaning

Other models:

1. Sets and relations

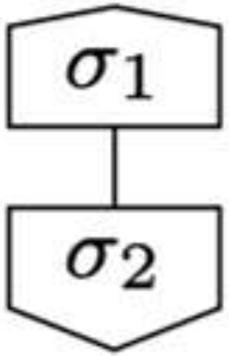
   relations as Boolean-valued matrices
   possibilistic model

1. Density matrices and superoperators

1. Conceptual spaces

   meanings corresponding to senses

Comparing meanings



.

Summary: features of DisCoCat

Meaning of sentences with different grammatical structures live in the same space, allowing comparison of meanings.

.

Summary: features of DisCoCat

Meaning of sentences with different grammatical structures live in the same space, allowing comparison of meanings.

Intuitive diagrammatic representation

Summary: features of DisCoCat

Meaning of sentences with different grammatical structures live in the same space, allowing comparison of meanings.

Intuitive diagrammatic representation

Wire structure for functional words and simplification

Summary: features of DisCoCat

Meaning of sentences with different grammatical structures live in the same space, allowing comparison of meanings.

Intuitive diagrammatic representation

Wire structure for functional words and simplification

Supports variety of grammars: Pregroup grammar, Lambek calculus, CCG, etc.

Summary: features of DisCoCat

Word meanings can live in different kind of spaces, provided these organise themselves in a monoidal category matching the grammatical structure.

Summary: features of DisCoCat

Word meanings can live in different kind of spaces, provided these organise themselves in a monoidal category matching the grammatical structure.

Integrates grammar and meaning in one whole

Summary: features of DisCoCat

Word meanings can live in different kind of spaces, provided these organise themselves in a monoidal category matching the grammatical structure.

Integrates grammar and meaning in one whole

Meaning spaces are typed, accounting for the the varying grammatical role of the words.

Flaws of DisCoCat

DisCoCat does not answer the question of how the meanings of sentences compose in order to provide the meaning of an entire text.

Flaws of DisCoCat

DisCoCat does not answer the question of how the meanings of sentences compose in order to provide the meaning of an entire text.

It assumes words to have a fixed meaning, while in text meanings will typically evolve.

Flaws of DisCoCat

DisCoCat does not answer the question of how the meanings of sentences compose in order to provide the meaning of an entire text.

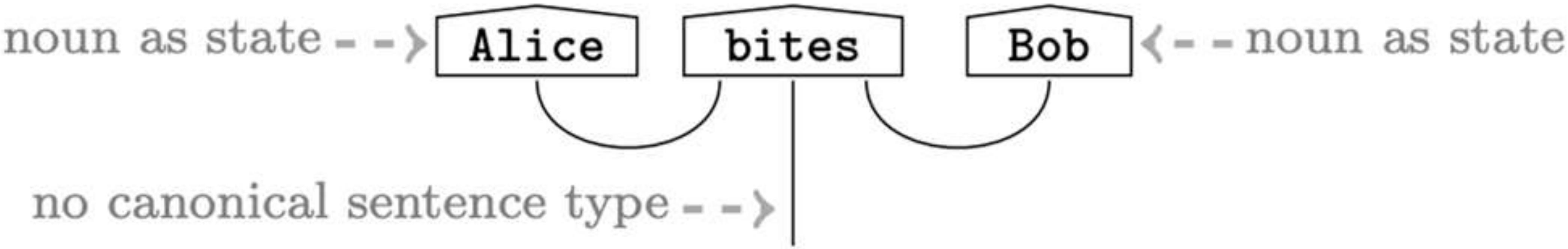
It assumes words to have a fixed meaning, while in text meanings will typically evolve.

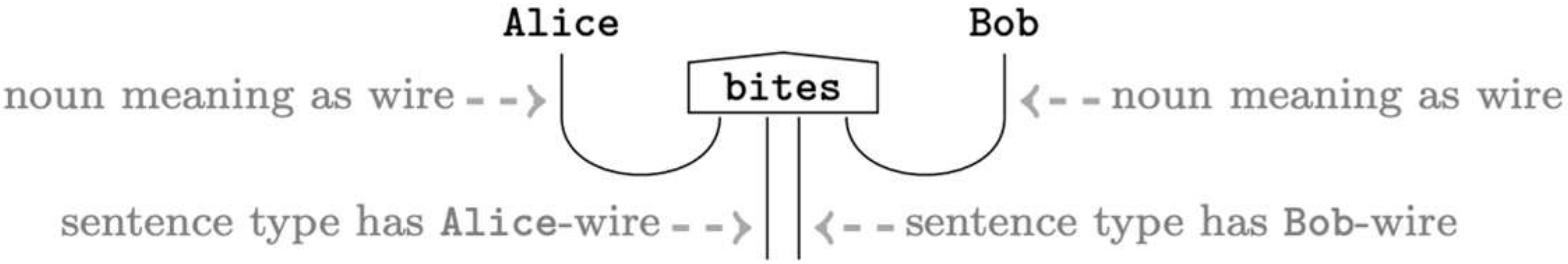
It does not determine the sentence type.

DisCoCirc

Alice is a dog.
Bob is a person.
Alice bites Bob.

.


noun as state
Alice
bites
Bob
noun as state
no canonical sentence type



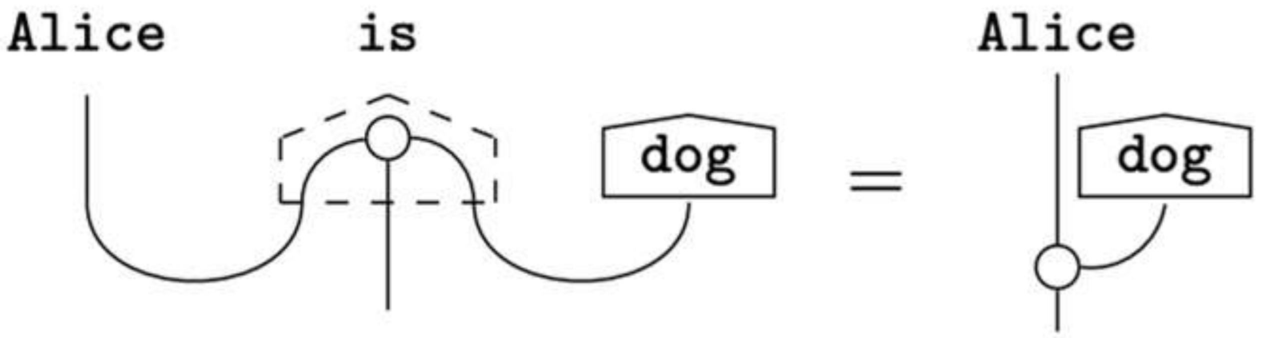
Alice
Bob
bites
noun meaning as wire
noun meaning as wire
sentence type has Alice-wire
sentence type has Bob-wire

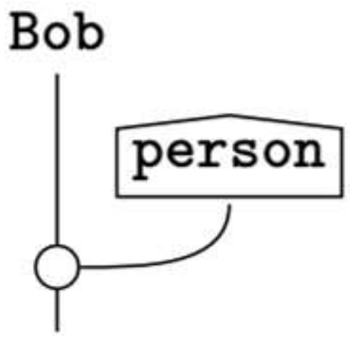
.

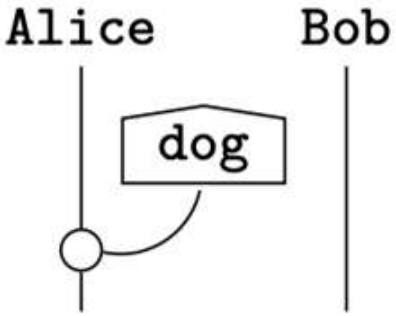

Alice bites Bob

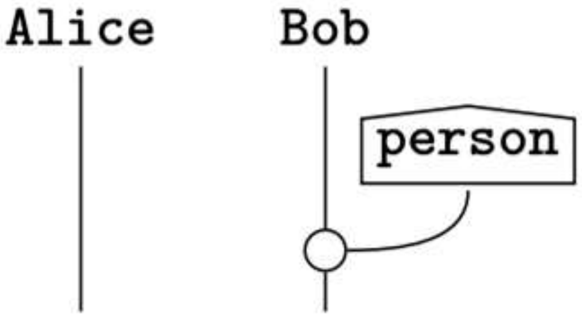
A sentence is not a state, but a process,

.

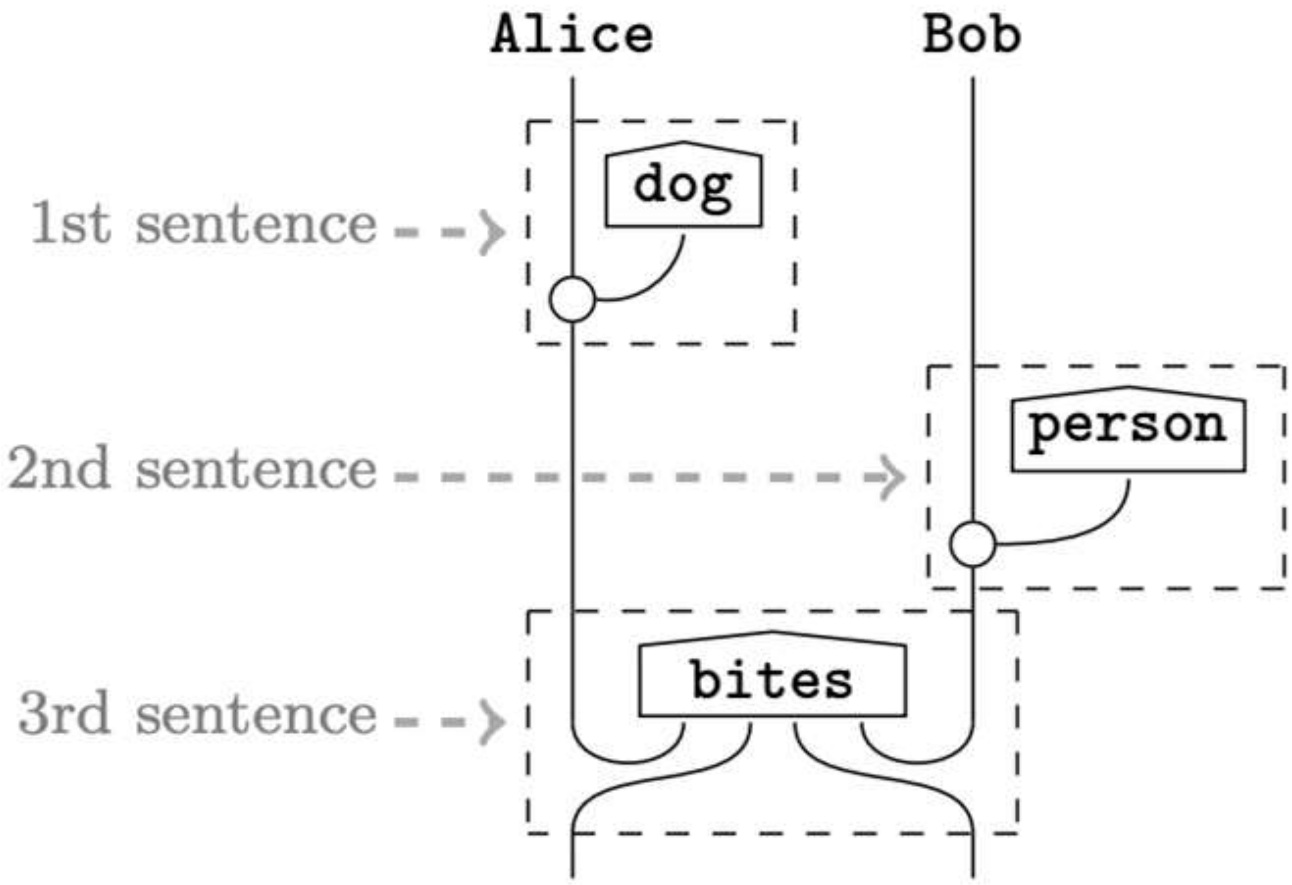

Alice
is
dog
=
Alice
dog


Bob
person

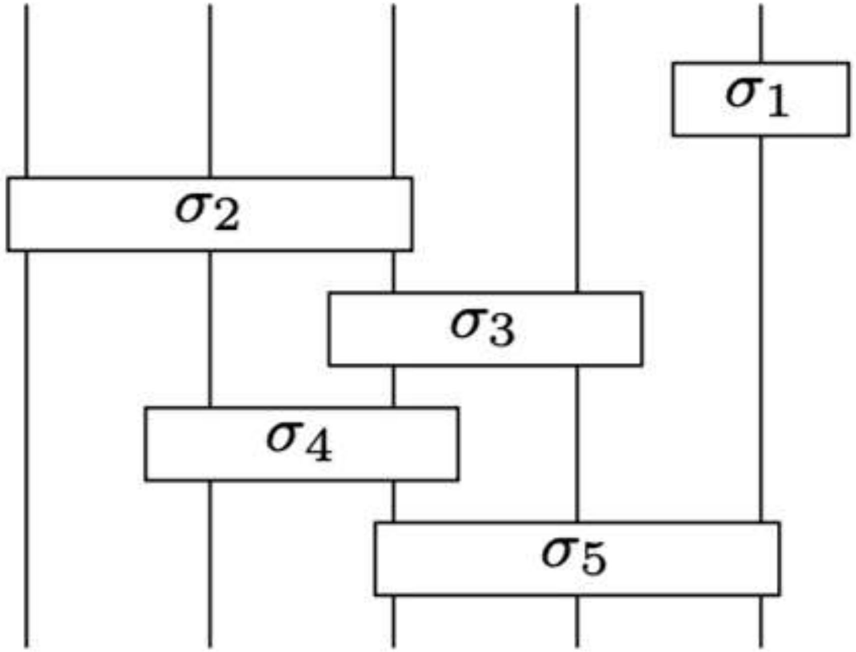

Alice
Bob
dog


Alice
Bob
person

.



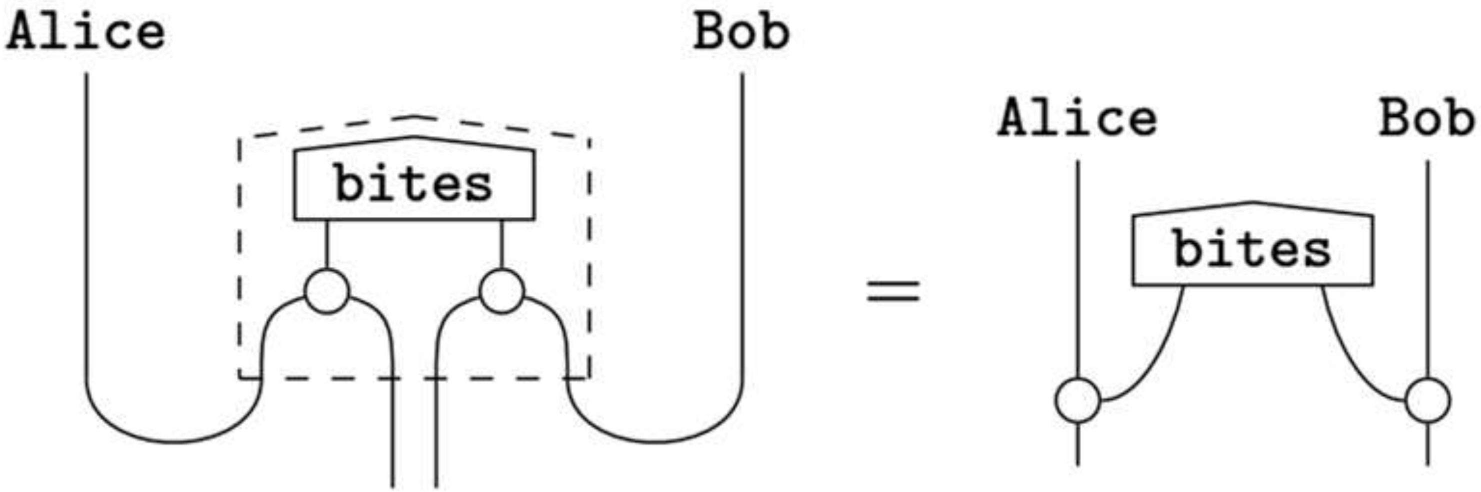
Alice
Bob
1st sentence
dog
2nd sentence
person
3rd sentence
bites



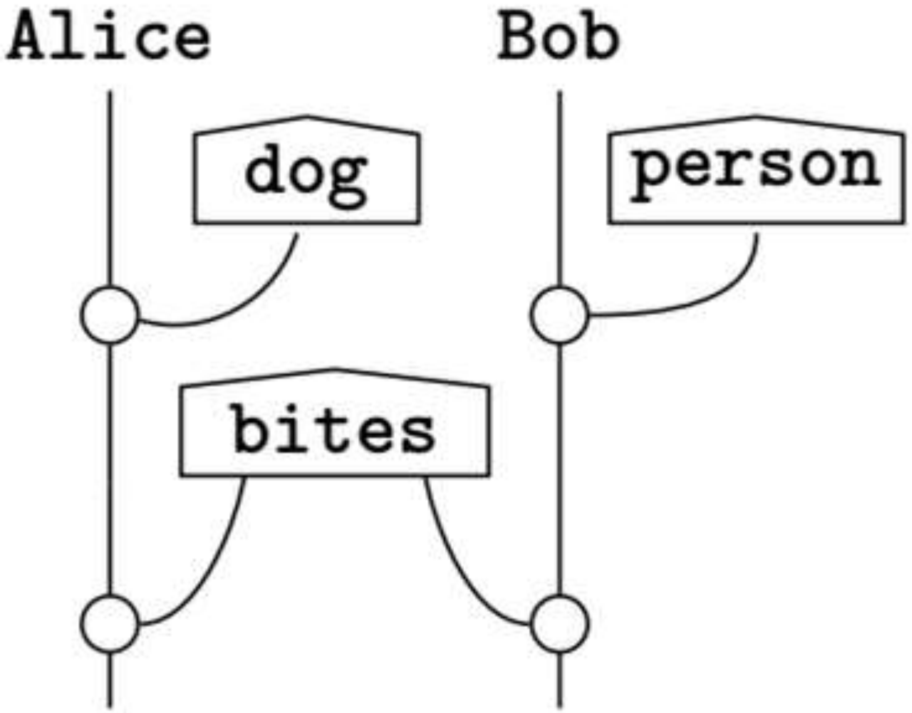
σ_1
σ_2
σ_3
σ_4
σ_5

Text is a process that alters meanings of words

.

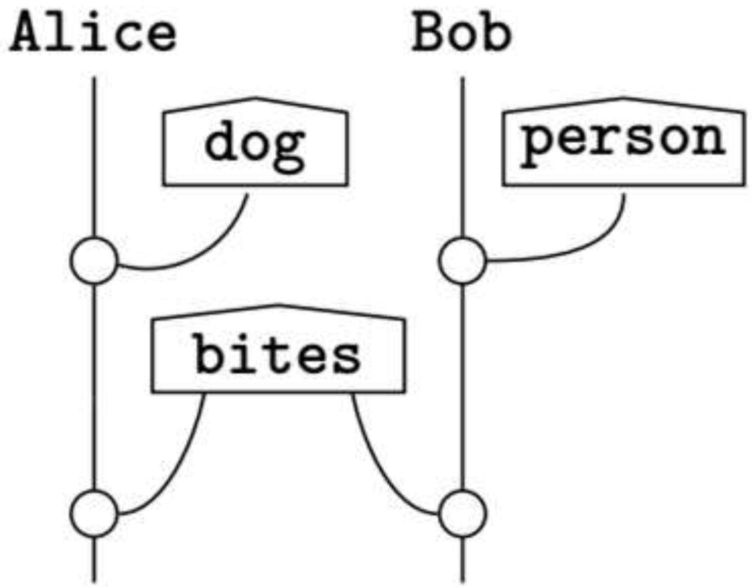

Alice
Bob
bites
=
Alice
Bob
bites

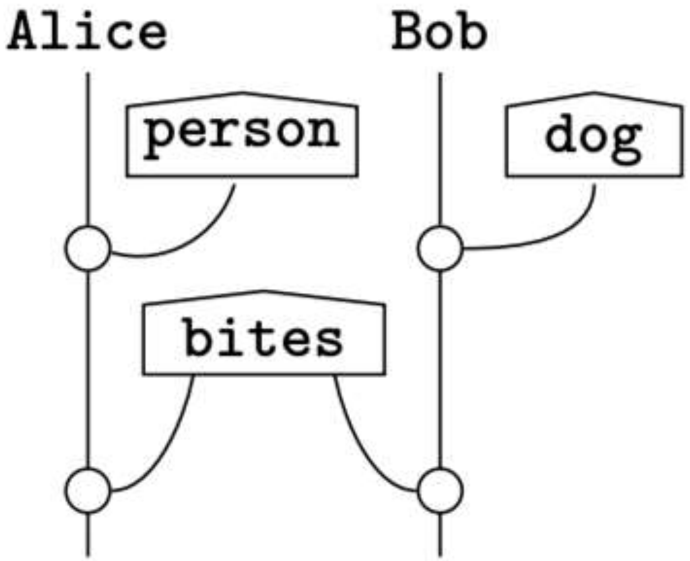
.


Alice
Bob
dog
person
bites

.

Different texts, different meanings



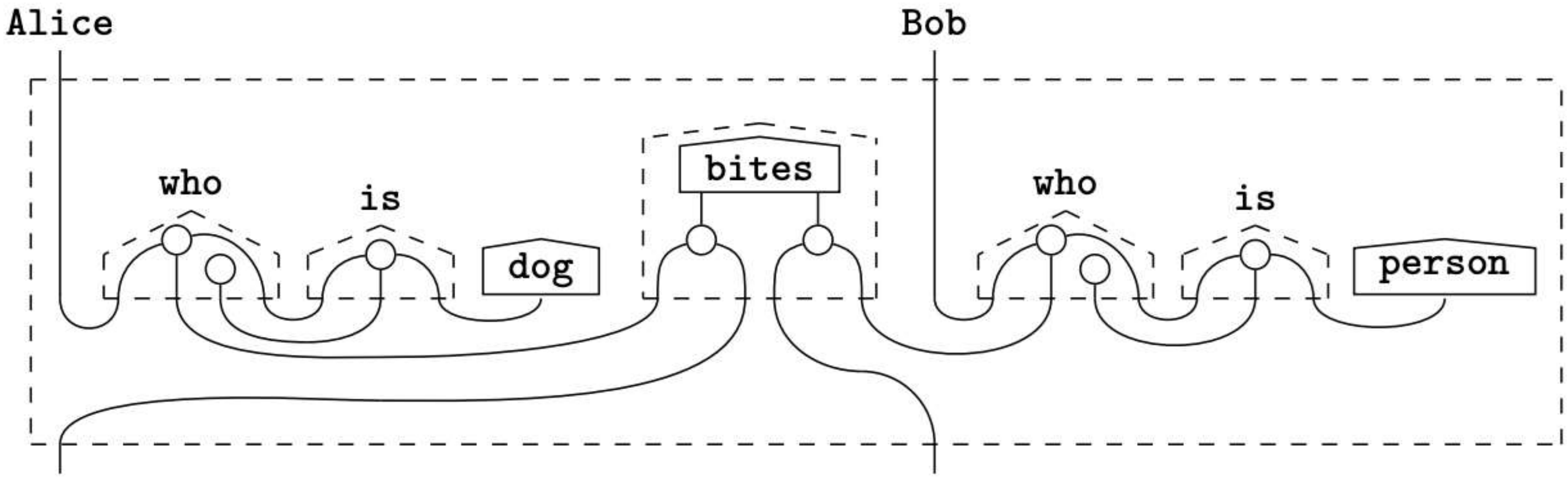


Different texts, same meaning

Alice who is a dog bites Bob who is a person.

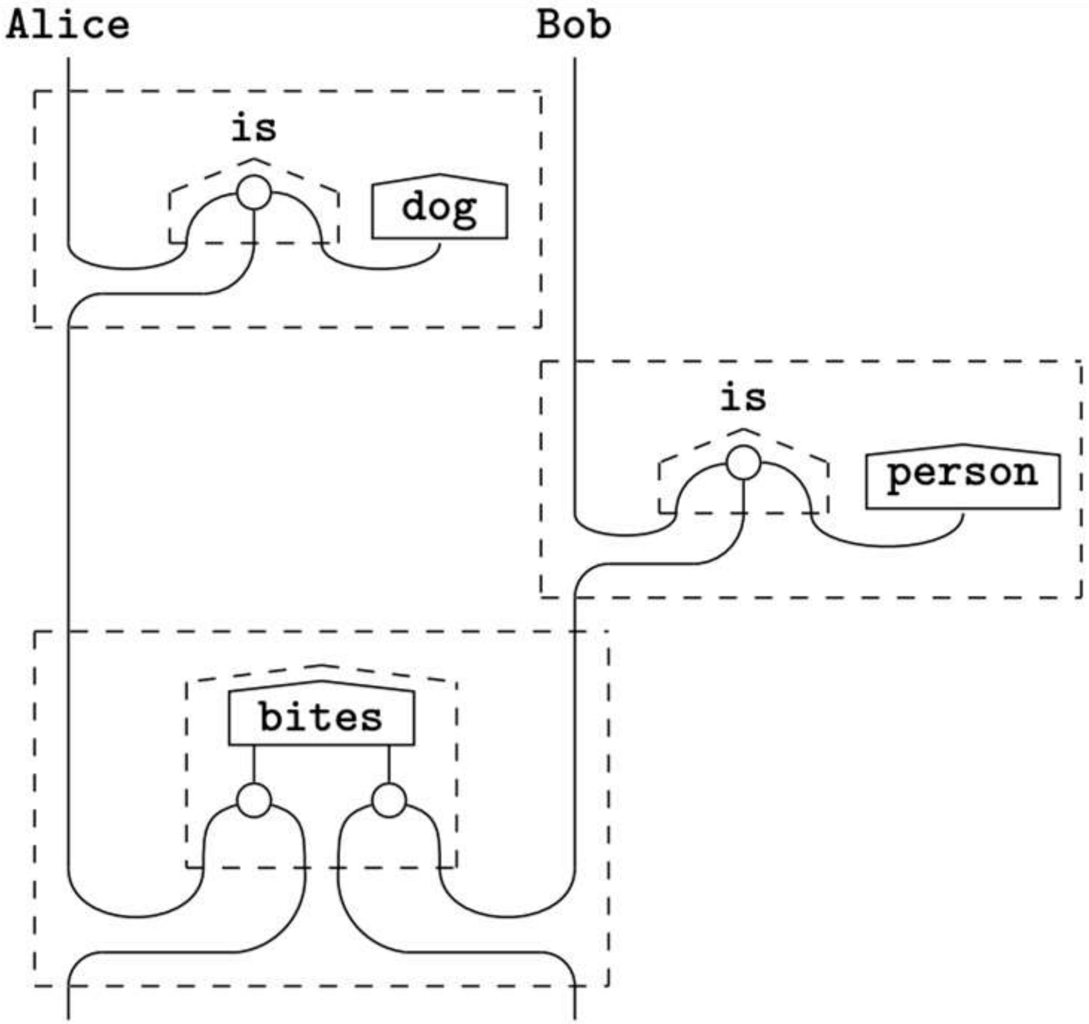
.

Different texts, same meaning

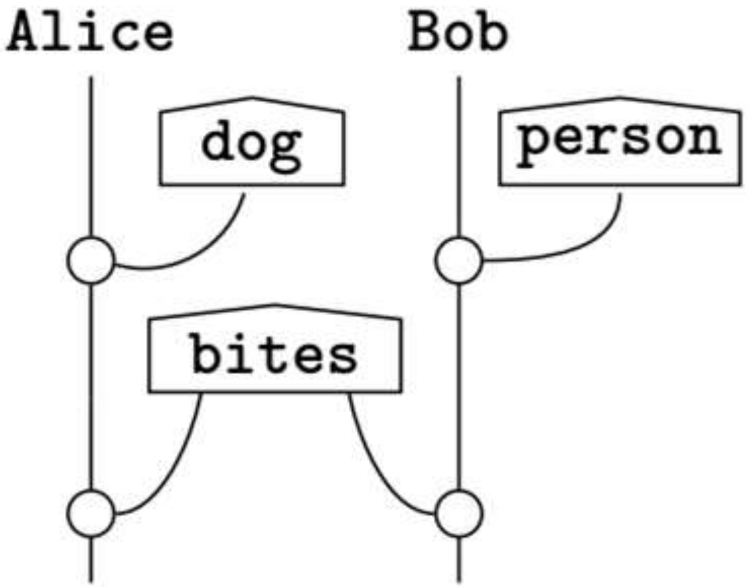


.

Different texts, same meaning



Different texts, same meaning



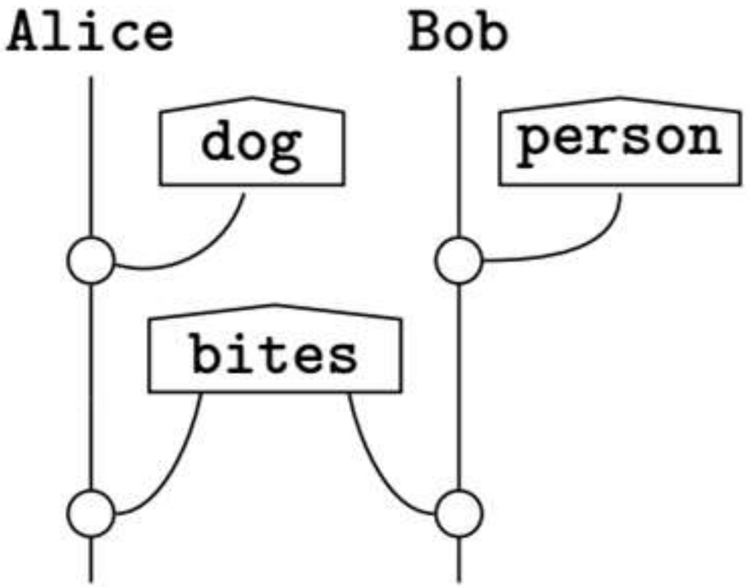
.

Different texts, same meaning

Alice is a dog.
Bob is a person.
Alice bites Bob.

.

Different texts, same meaning

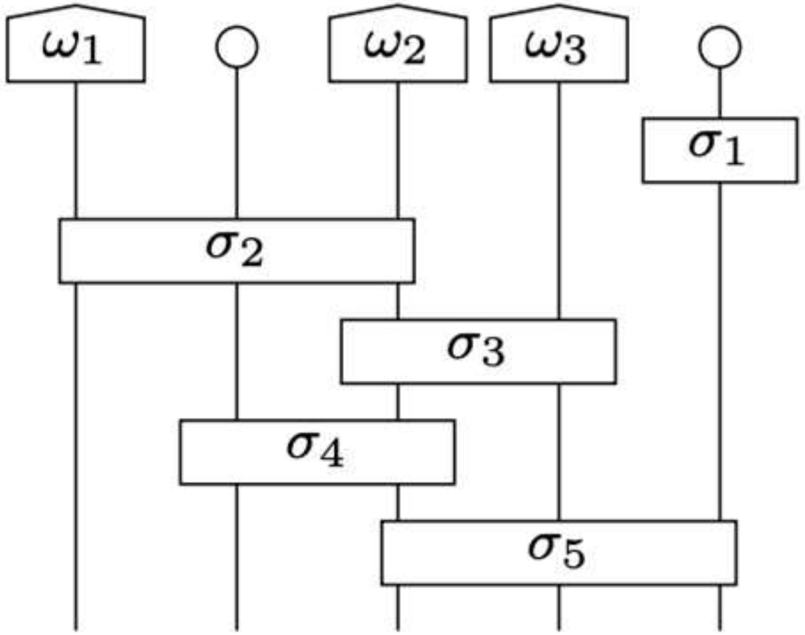


Noun initial states

Static nouns: the text does not alter our understanding of them

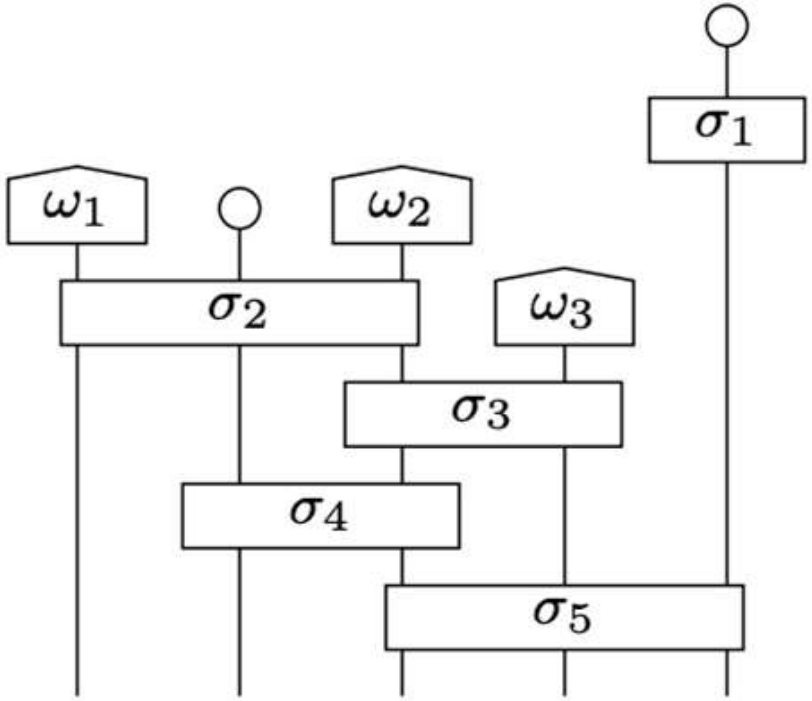
Dynamic nouns: the text alters our understanding of them

Noun initial states

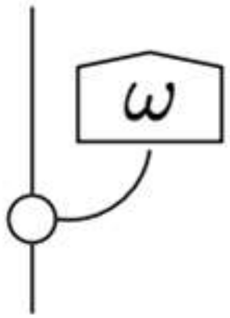


.

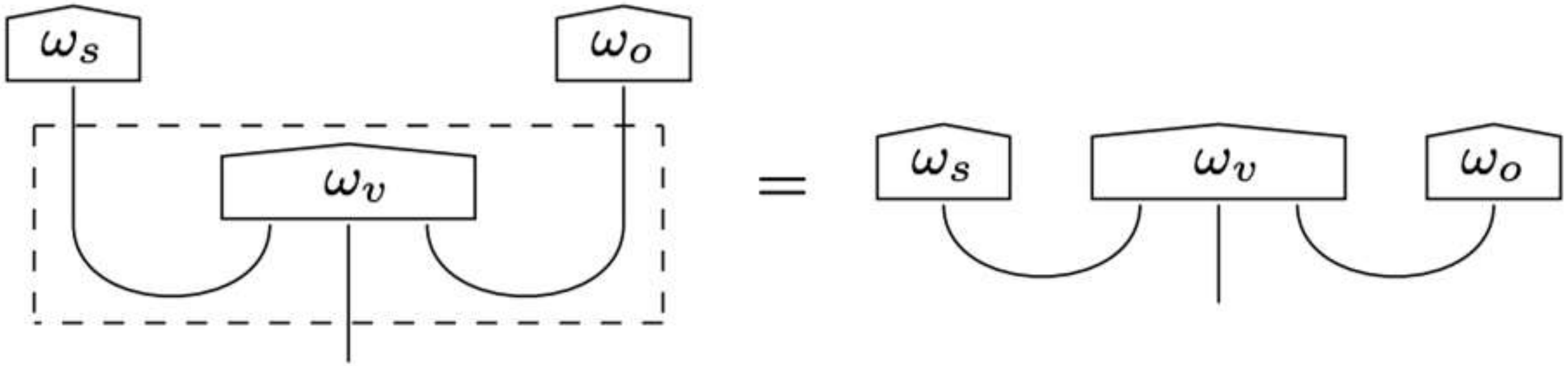
Noun initial states



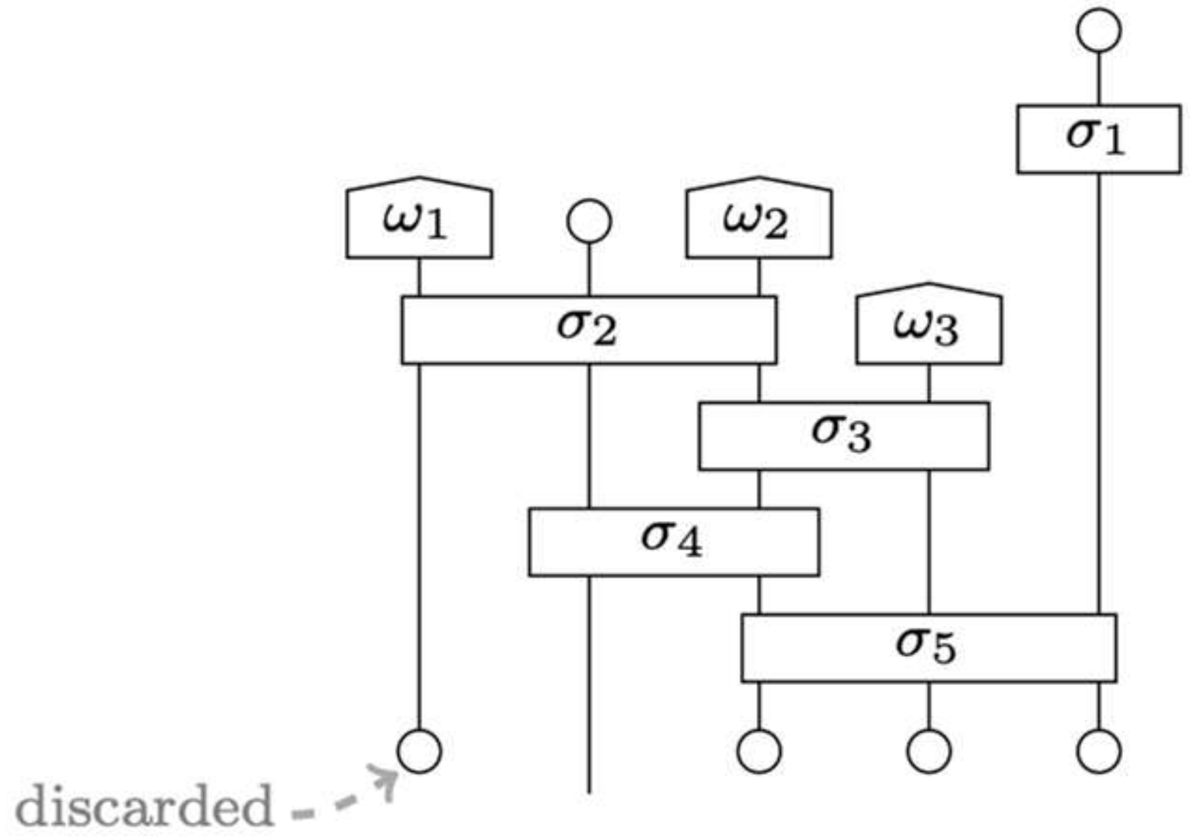
Initial processes



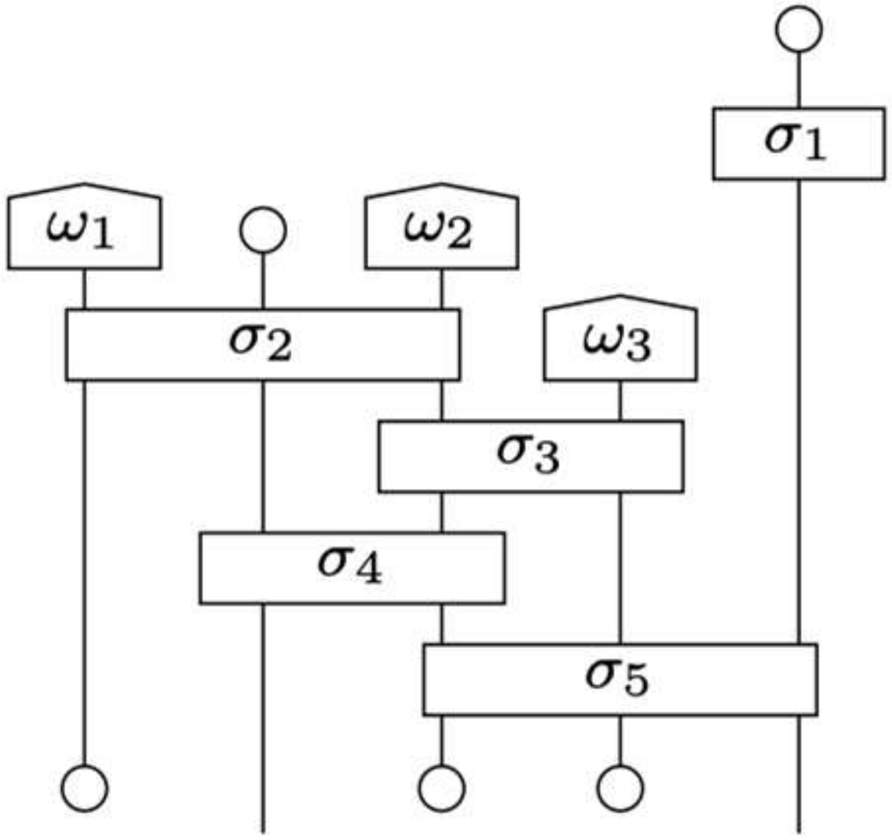
DisCoCat from DisCoCirc



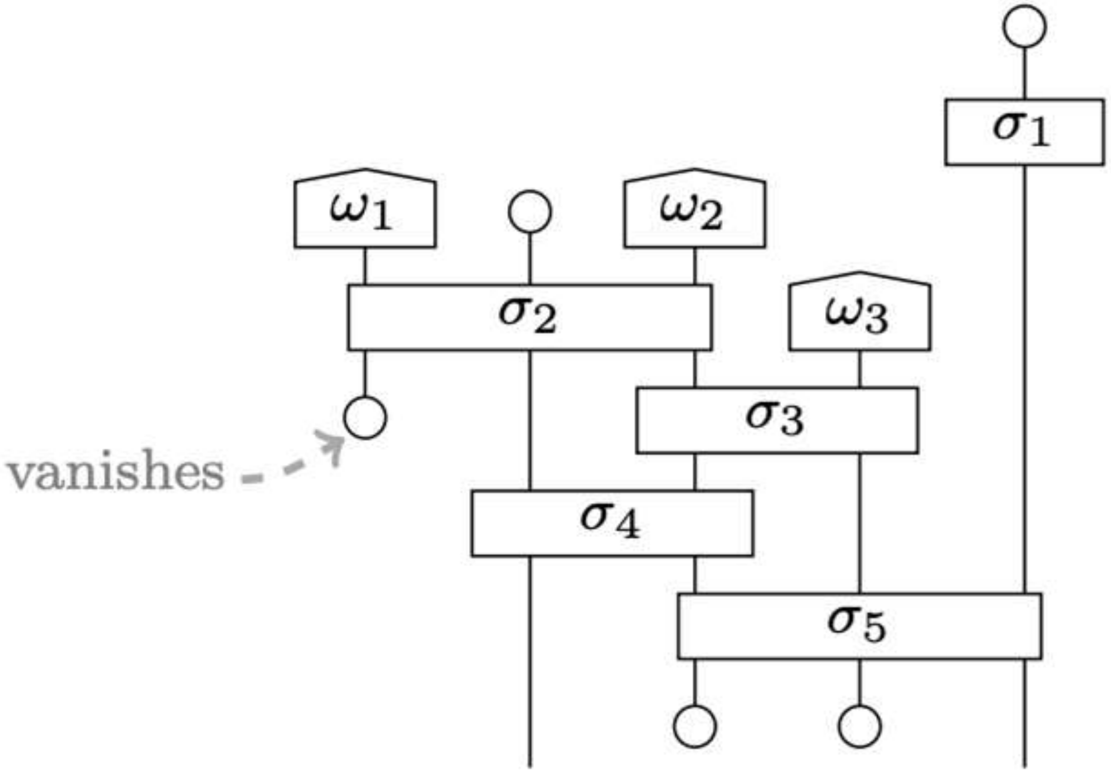
Individual meaning



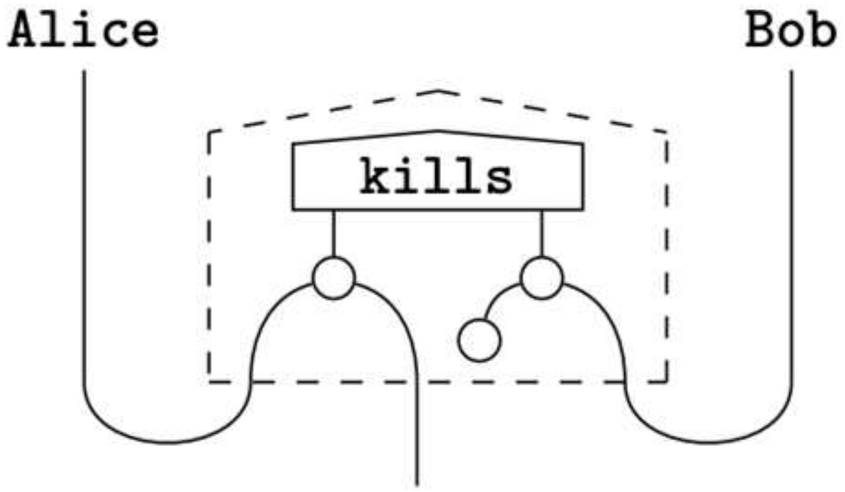
Subgroup meaning



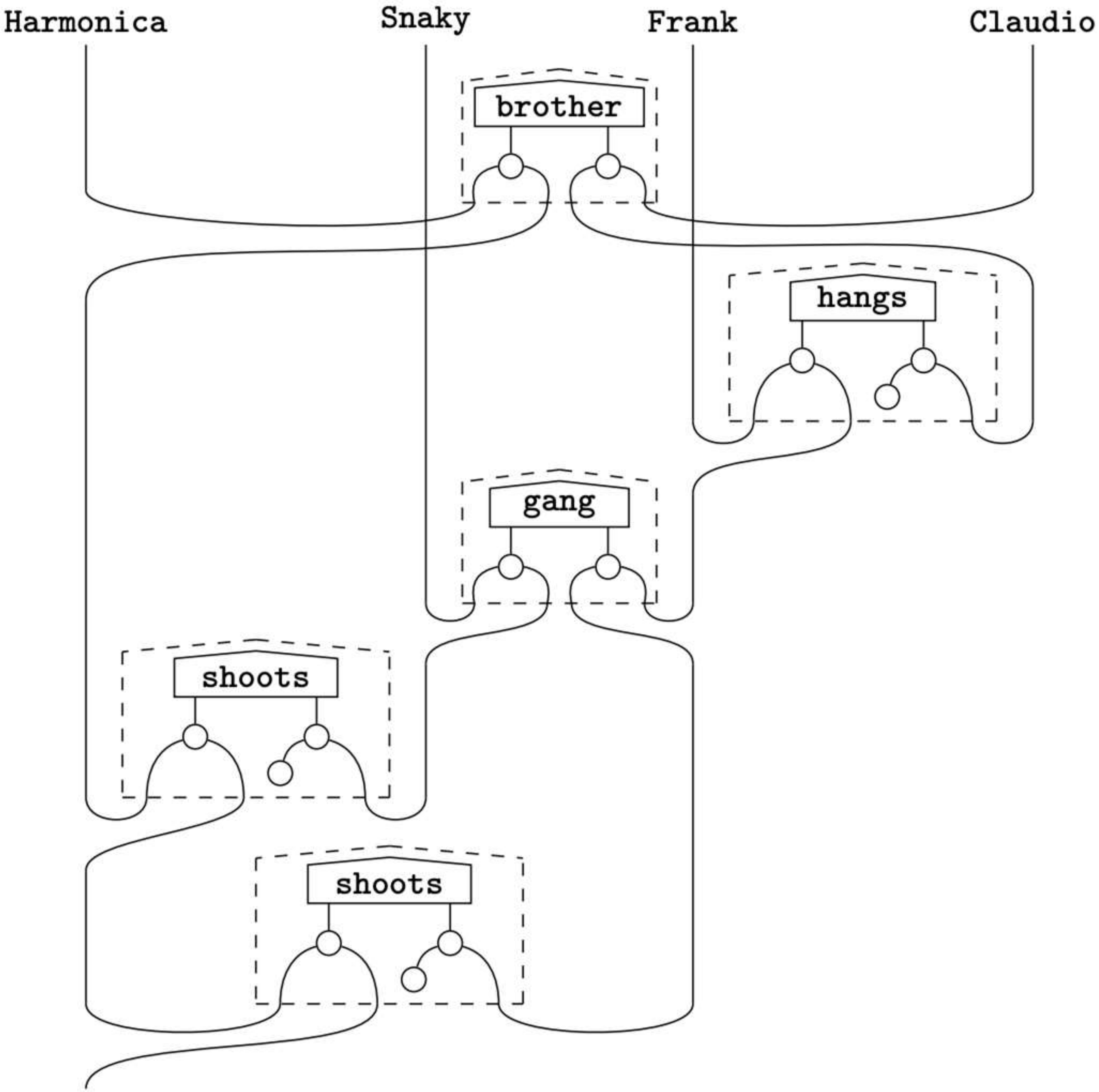
Vanishing of agents

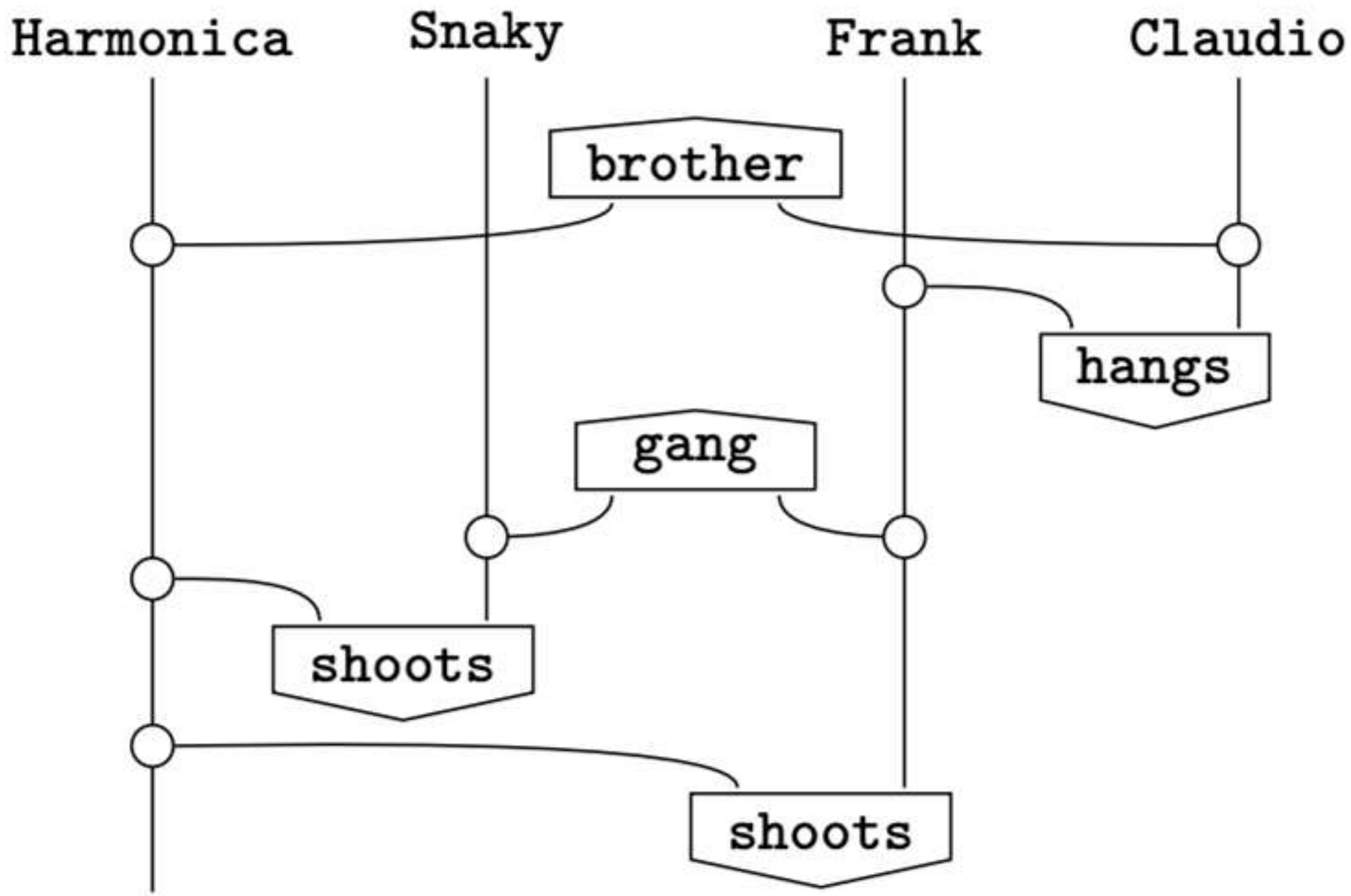


Vanishing of agents



Harmonica (is the brother of) Claudio.
Frank hangs Claudio.
Snaky (is in the gang of) Frank.
Harmonica shoots Snaky.
Harmonica shoots Frank.


Harmonica
Snaky
Frank
Claudio
brother
hangs
gang
shoots
shoots

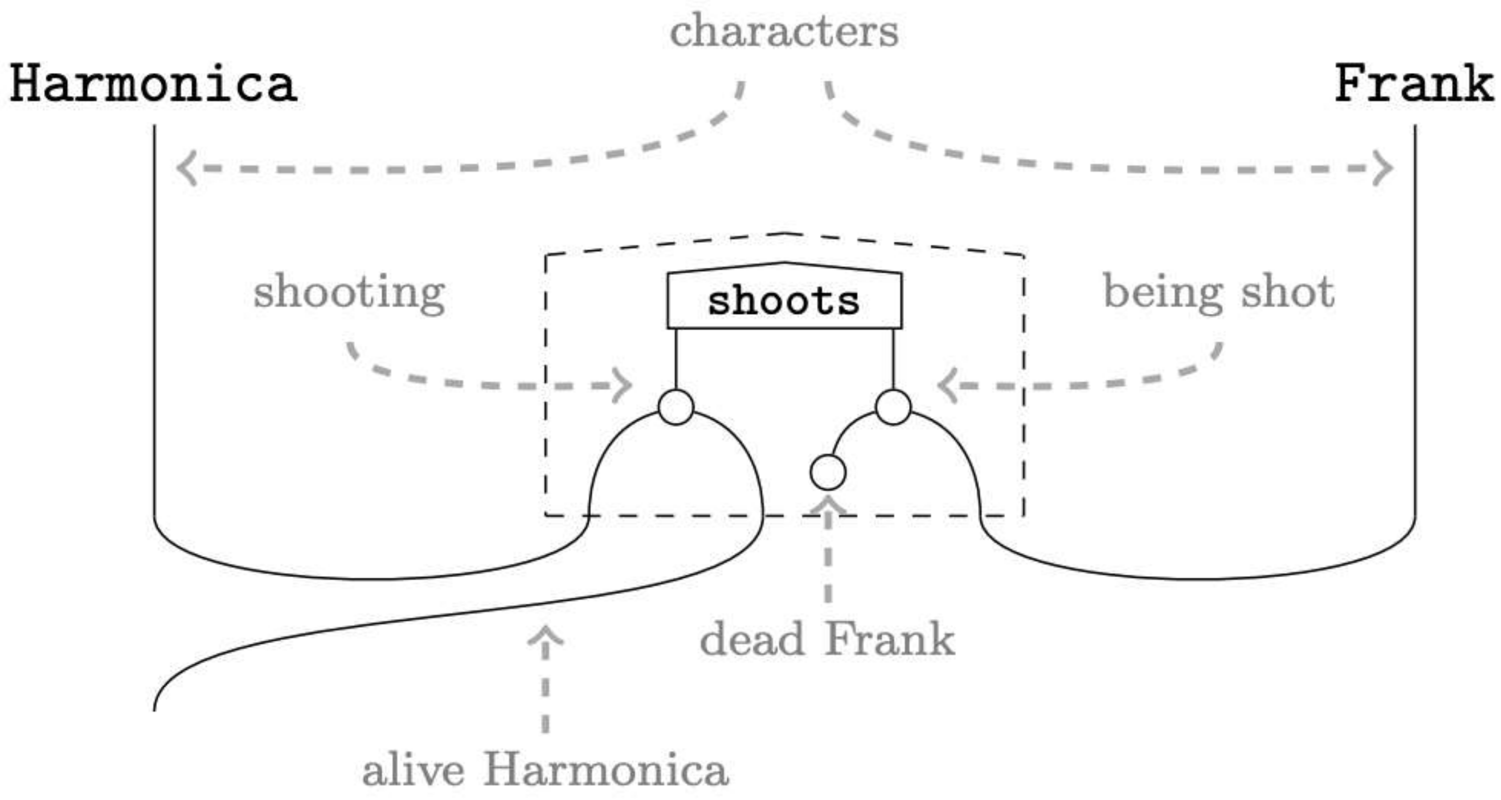


Harmonica
Snaky
Frank
Claudio
brother
hangs
gang
shoots
shoots

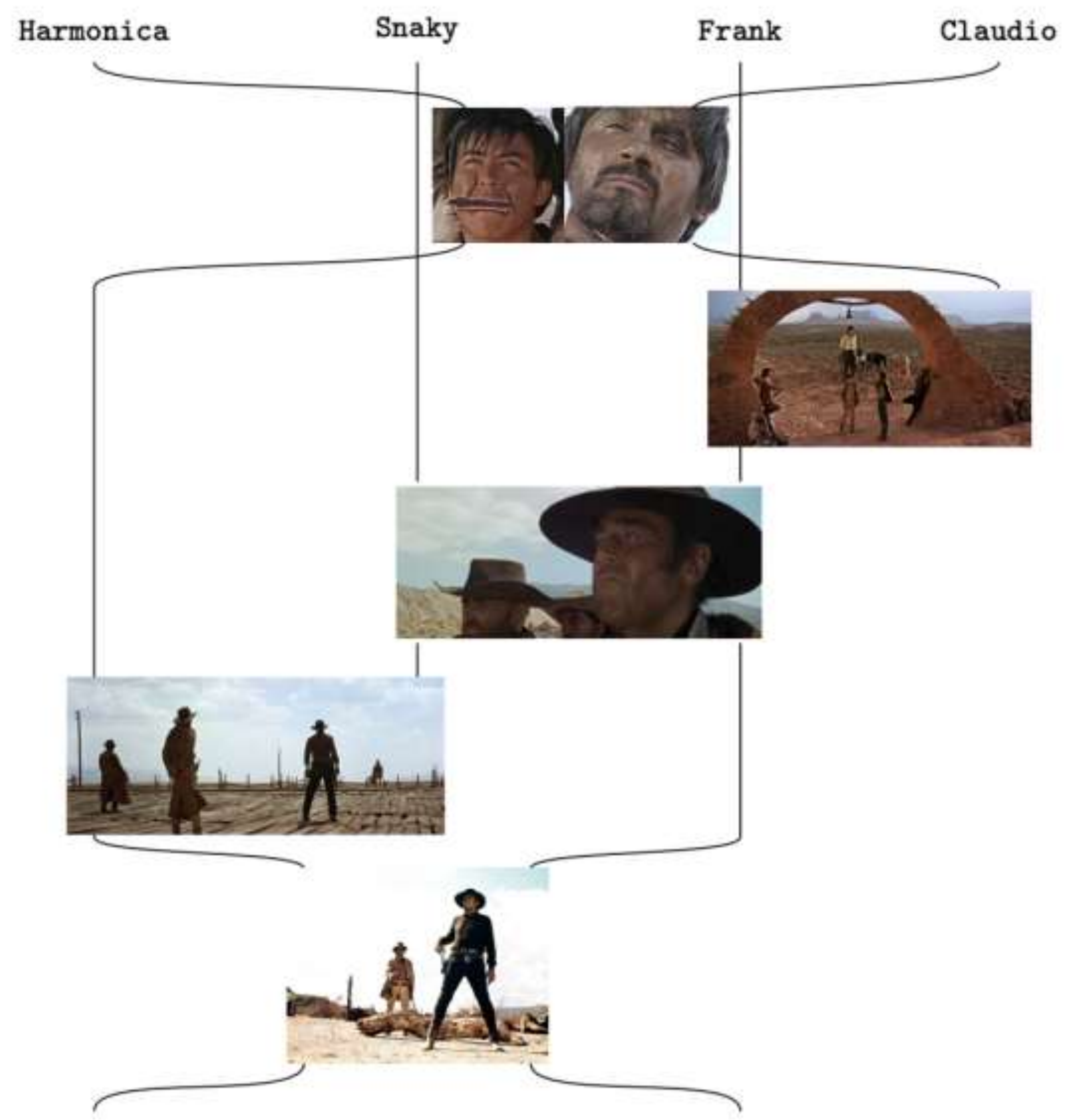
Other cognitive modes

Harmonica shoots Frank.

Other cognitive modes



Other cognitive modes



•

arXiv > cs > arXiv:2301.10595

Search

Help

Computer Science > Computation and Language

[Submitted on 25 Jan 2023]

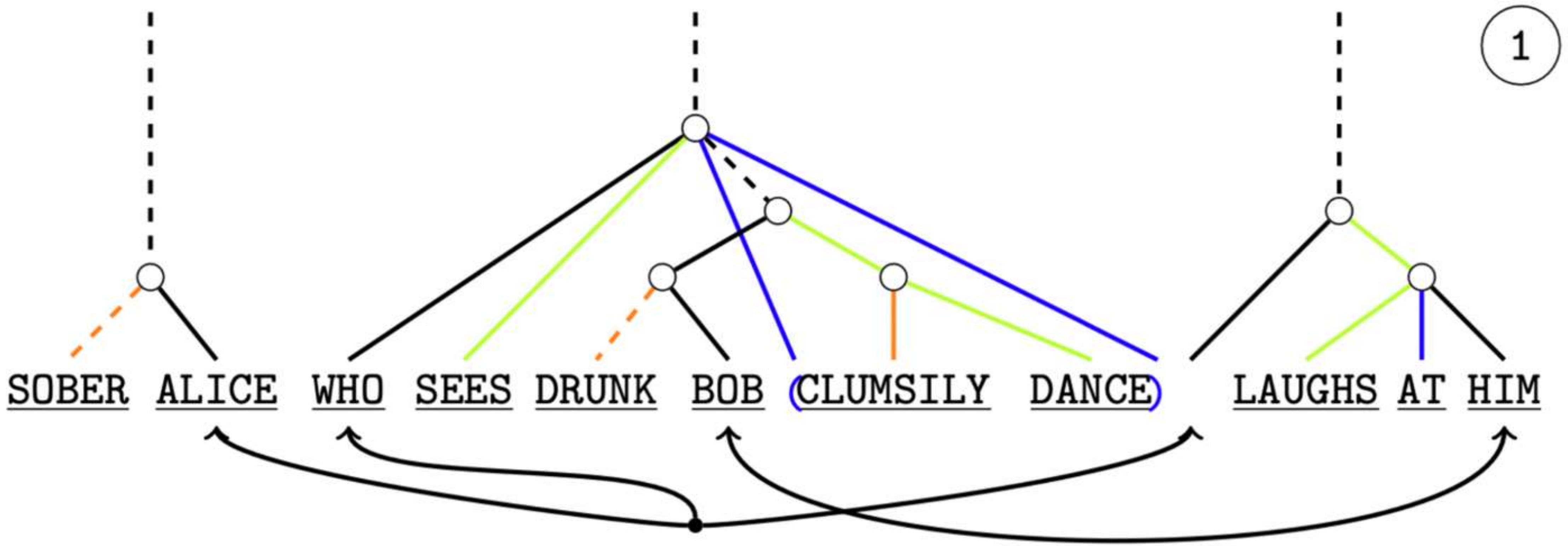
Distilling Text into Circuits

Vincent Wang-Mascianica, Jonathon Liu, Bob Coecke

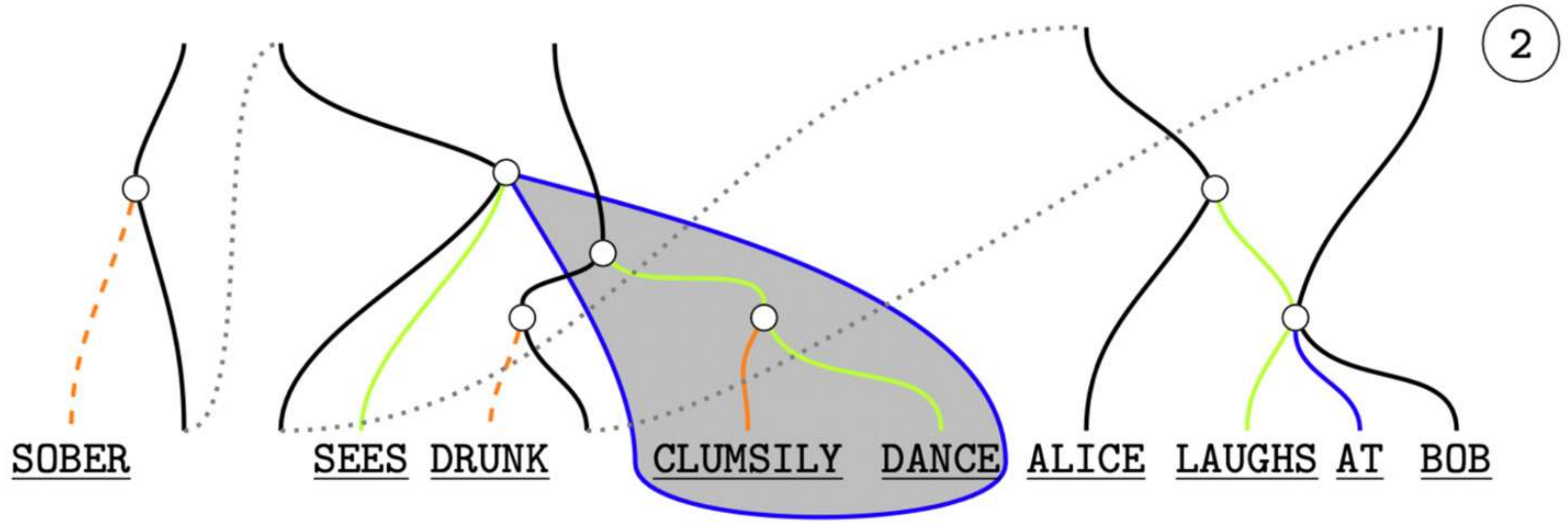
This paper concerns the structure of meanings within natural language. Earlier, a framework named DisCoCirc was sketched that (1) is compositional and distributional (a.k.a. vectorial); (2) applies to general text; (3) captures linguistic `connections' between meanings (cf. grammar) (4) updates word meanings as text progresses; (5) structures sentence types; (6) accommodates ambiguity. Here, we realise DisCoCirc for a substantial fragment of English.

When passing to DisCoCirc's text circuits, some `grammatical bureaucracy' is eliminated, that is, DisCoCirc displays a significant degree of (7) inter- and intra-language independence. That is, e.g., independence from word-order conventions that differ across languages, and independence from choices like many short sentences vs. few long sentences. This inter-language independence means our text circuits should carry over to other languages, unlike the language-specific typings of categorial grammars. Hence, text circuits are a lean structure for the `actual substance of text', that is, the inner-workings of meanings within text across several layers of expressiveness (cf. words, sentences, text), and may capture that what is truly universal beneath grammar. The elimination of grammatical bureaucracy also explains why DisCoCirc: (8) applies beyond language, e.g. to spatial, visual and other cognitive modes. While humans could not verbally communicate in terms of text circuits, machines can.

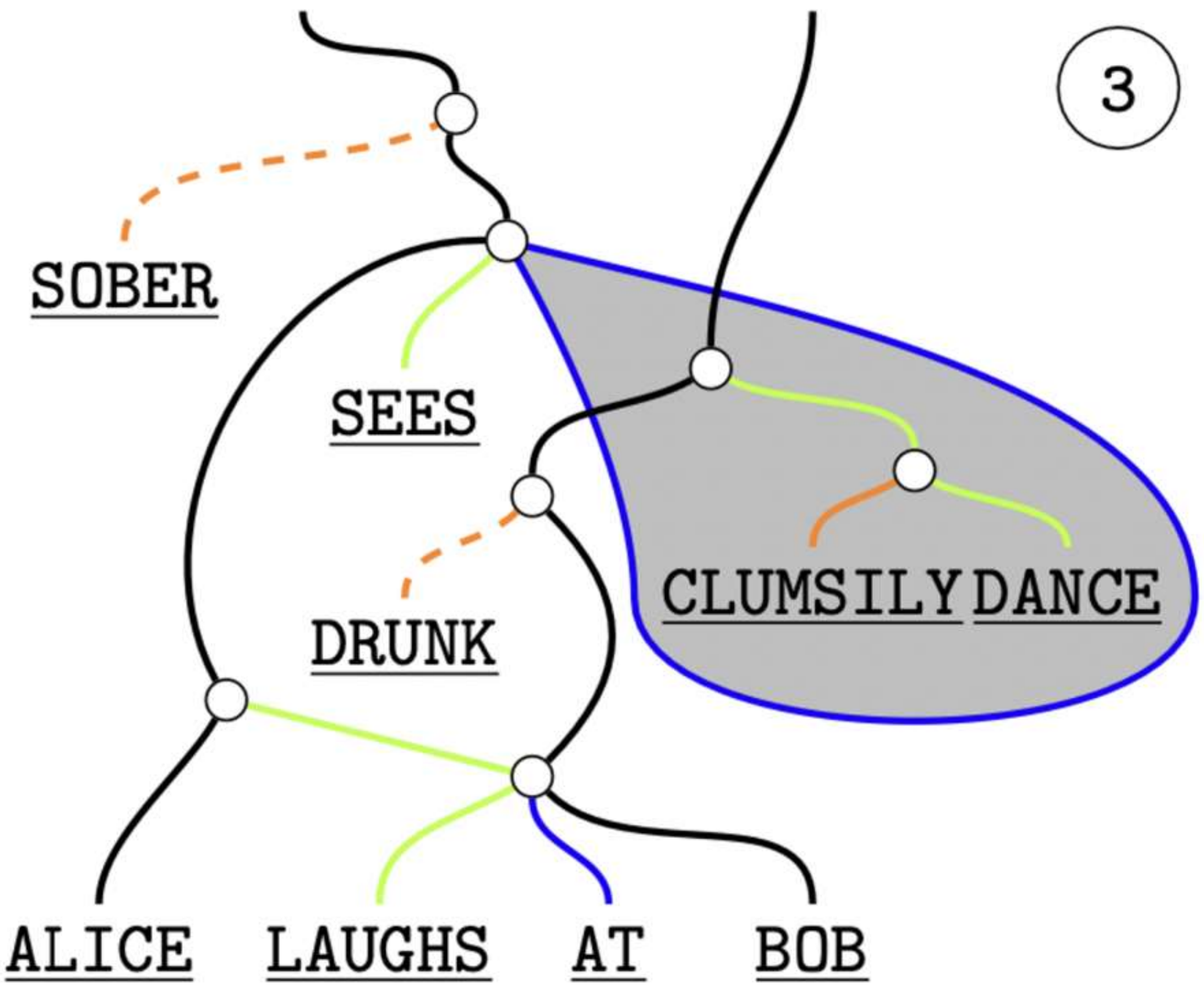
We first define a `hybrid grammar' for a fragment of English, i.e. a purpose-built, minimal grammatical formalism needed to obtain text circuits. We then detail a translation process such that all text generated by this grammar yields a text circuit. Conversely, for any text circuit obtained by freely composing the generators, there exists a text (with hybrid grammar) that gives rise to it. Hence: (9) text circuits are generative for text.



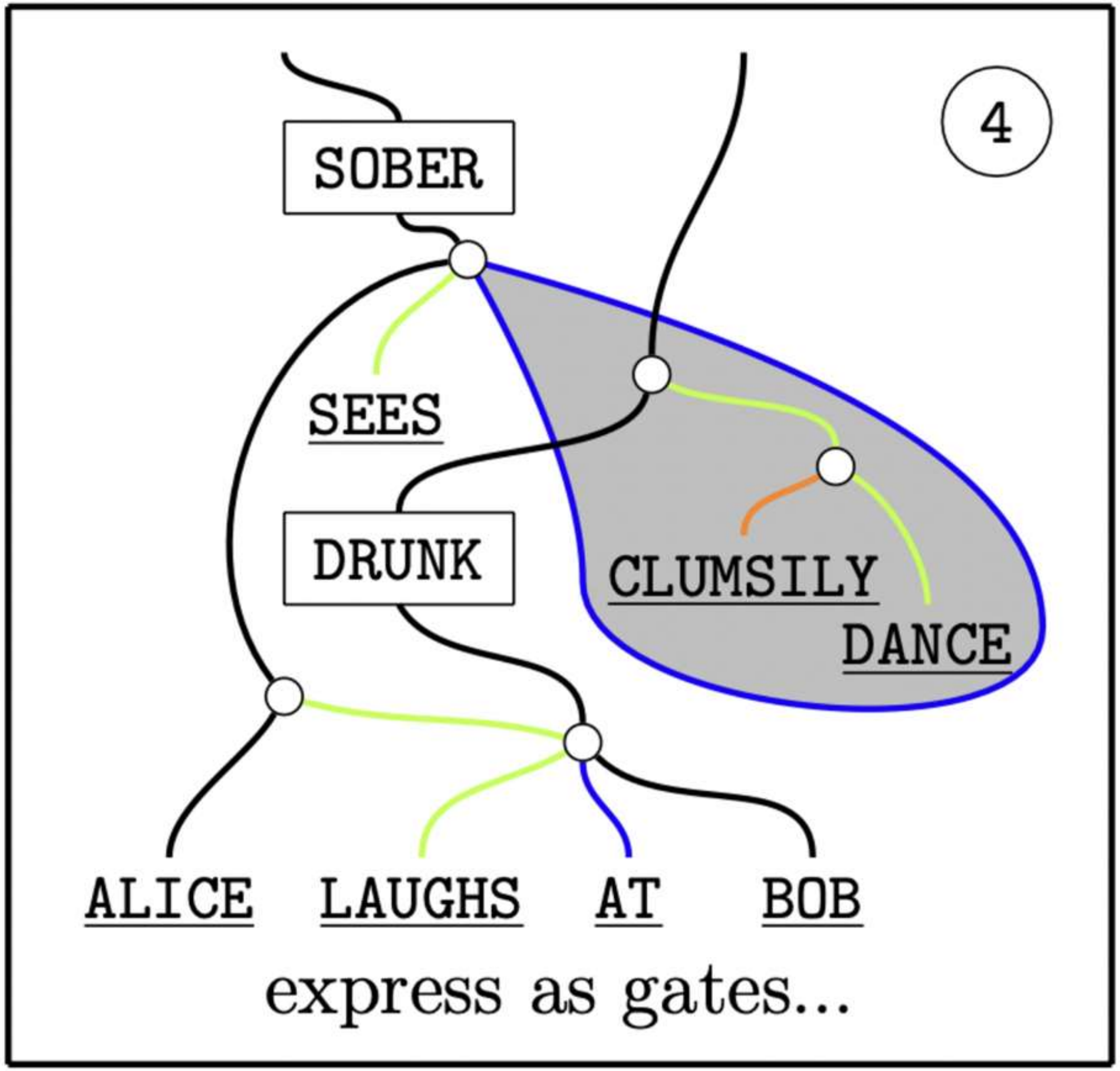
Text with hybrid grammar

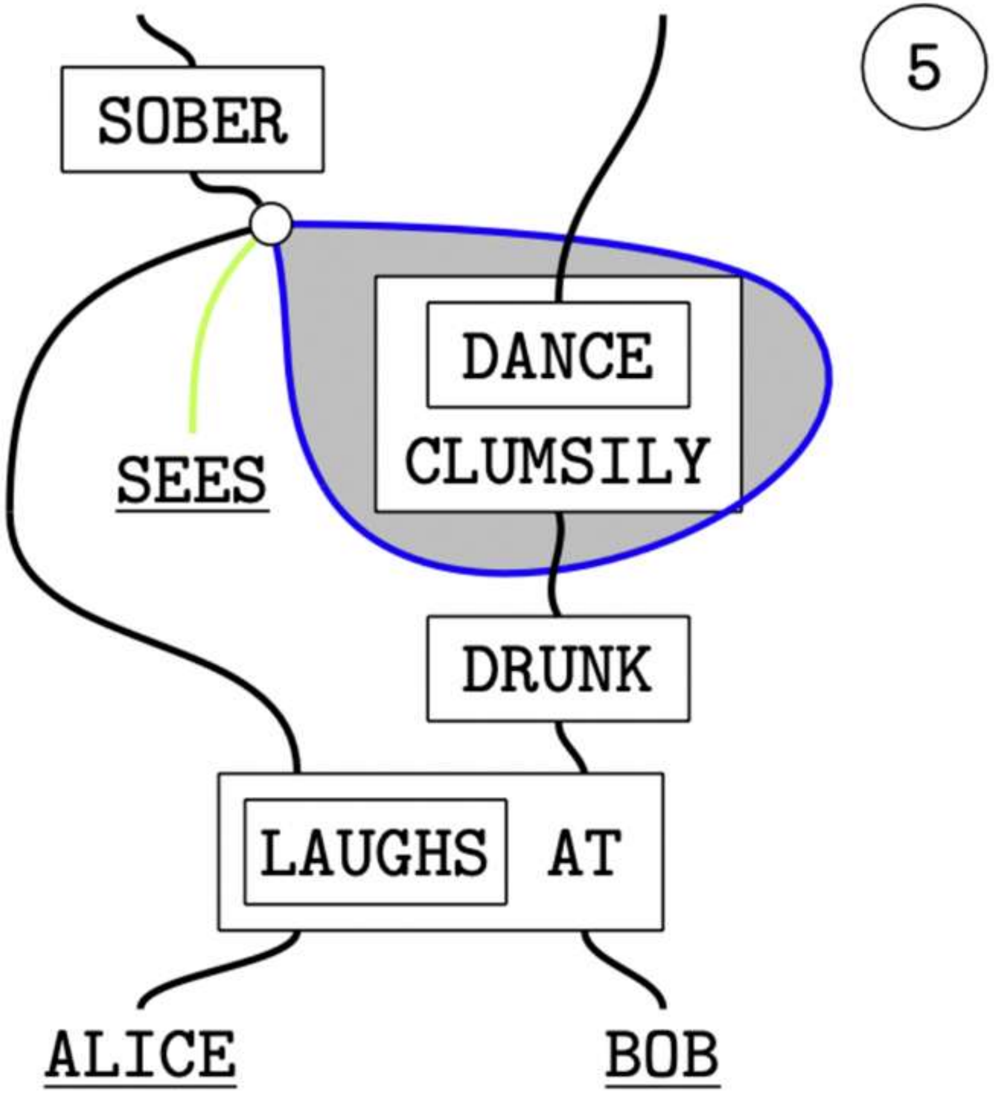


The same data, as a text diagram

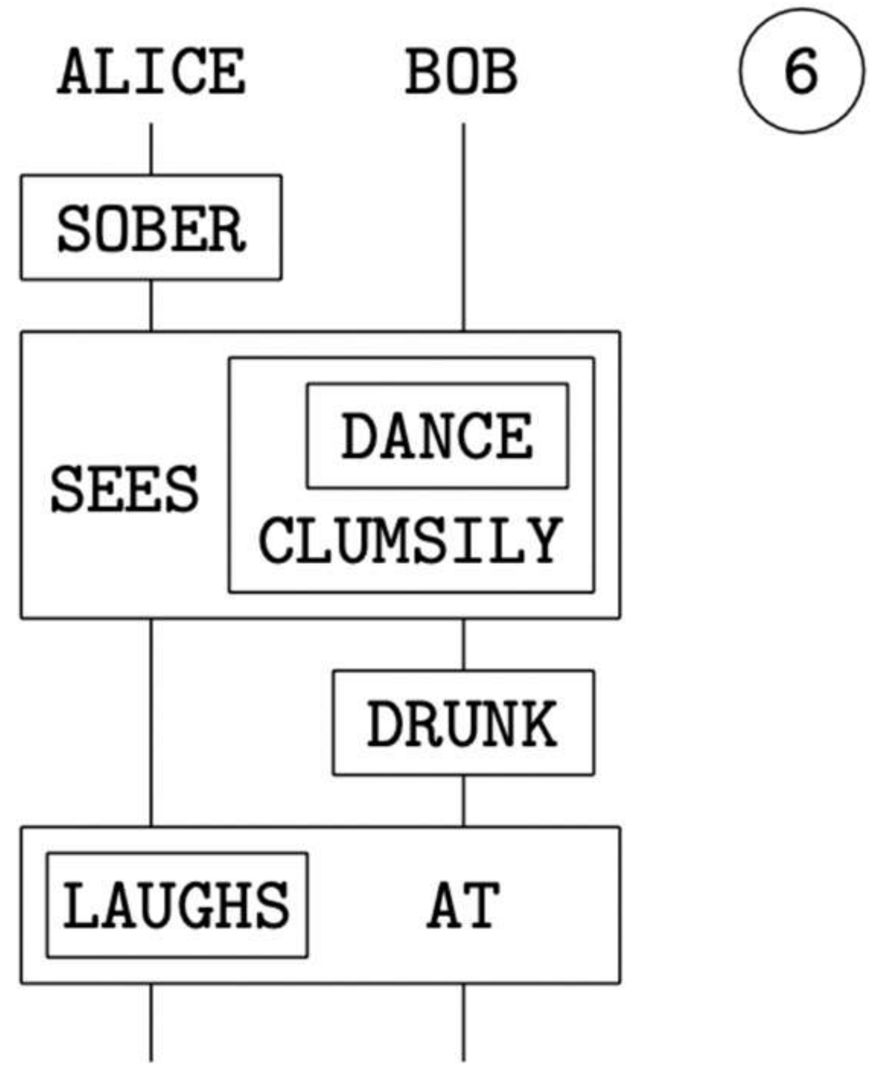


Rewrite links into composition


4
SOBER
SEES
CLUMSILY
DANCE
DRUNK
ALICE
LAUGHS
AT
BOB
express as gates...

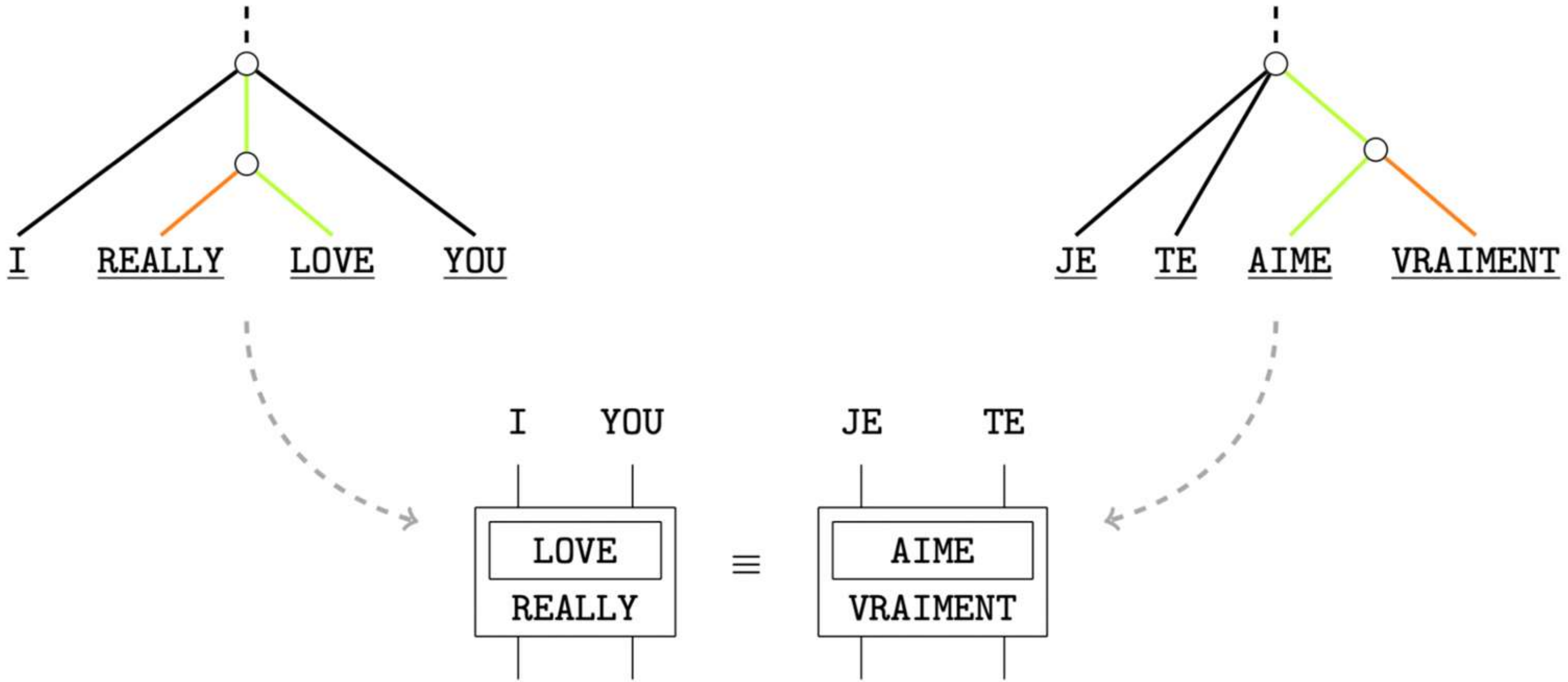


...and boxes...

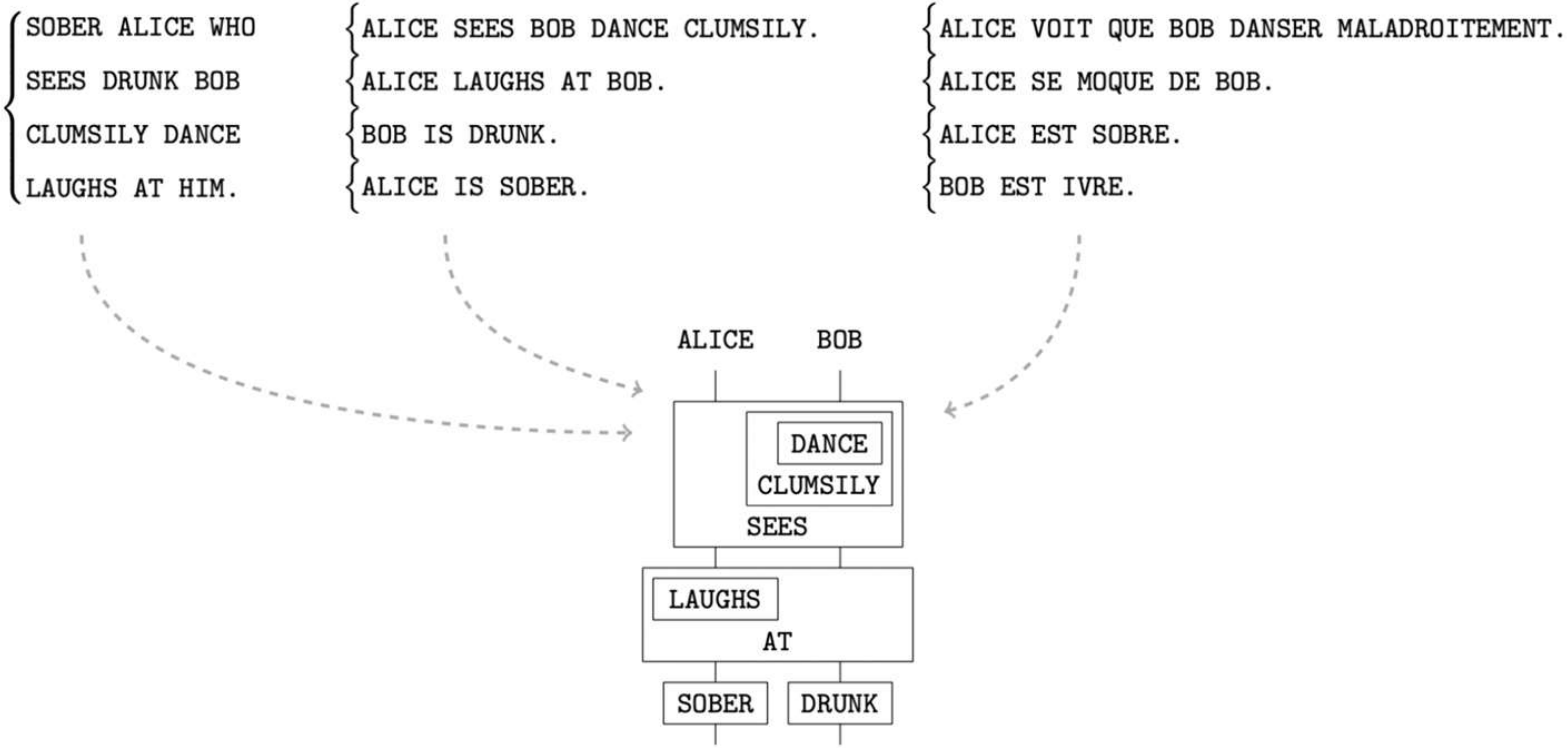


...to get a circuit

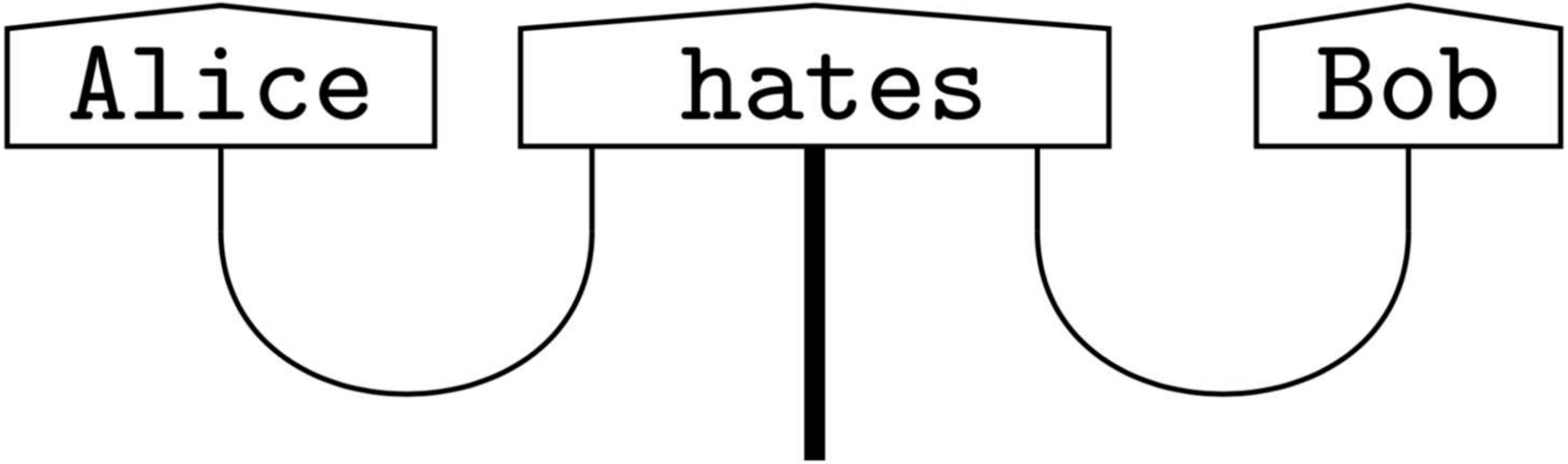
Different languages become the same!

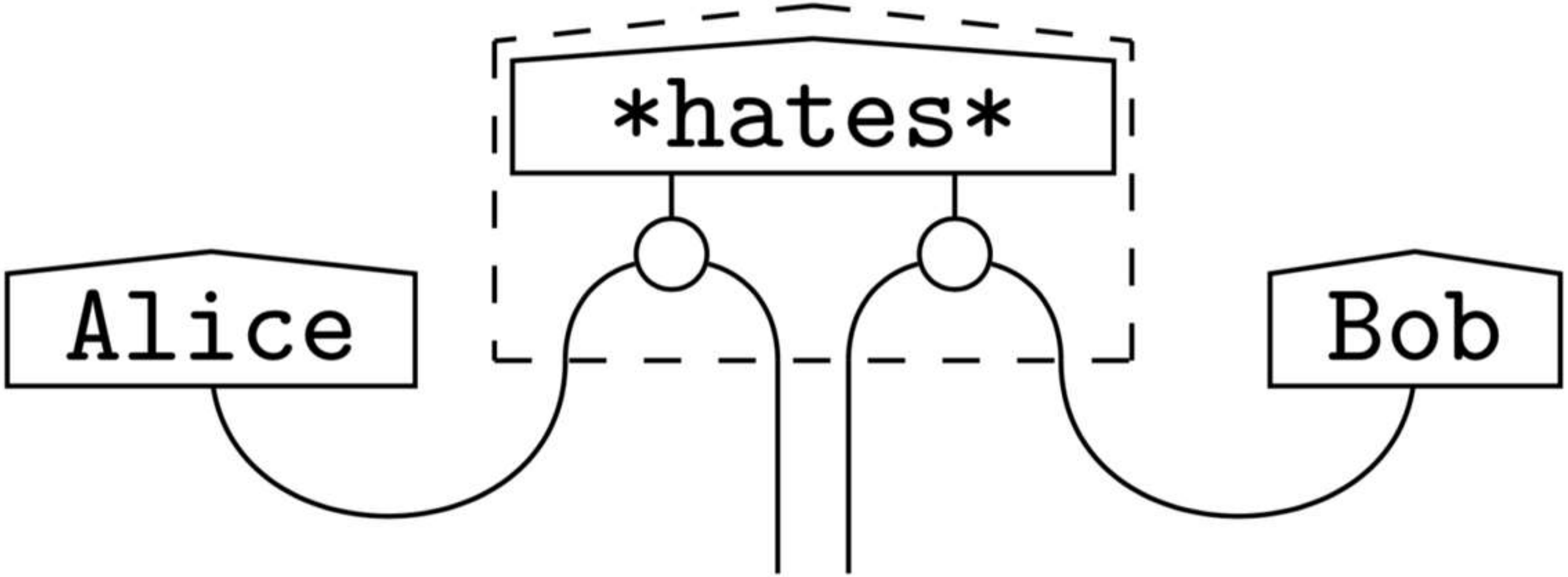


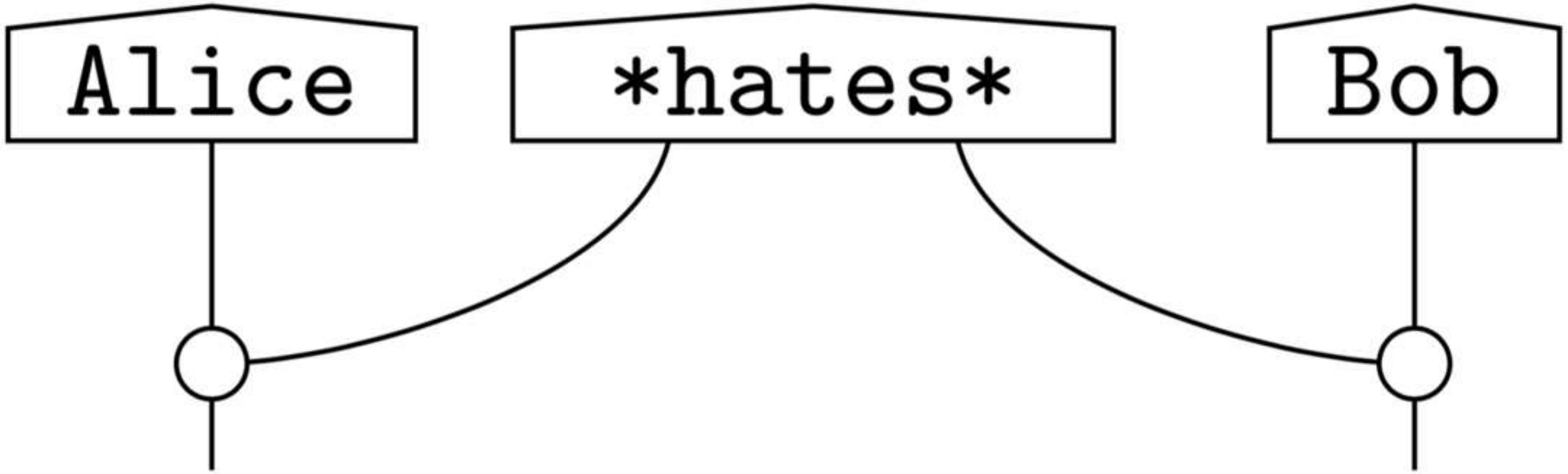
Different styles become the same!

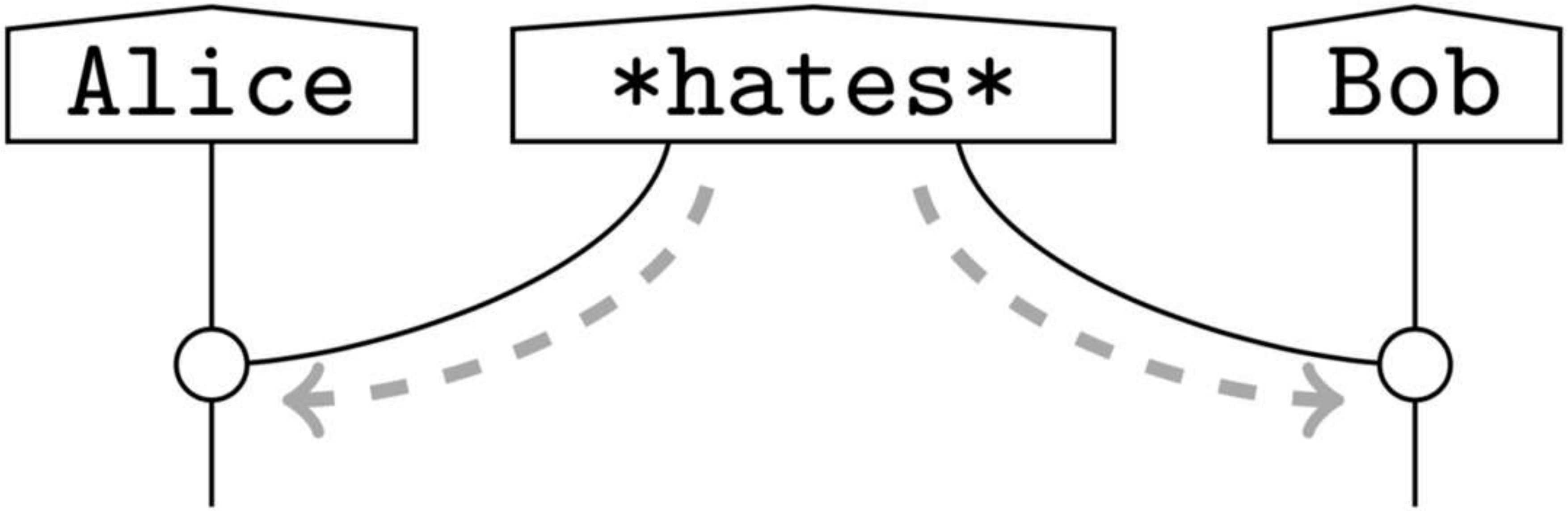


Summary

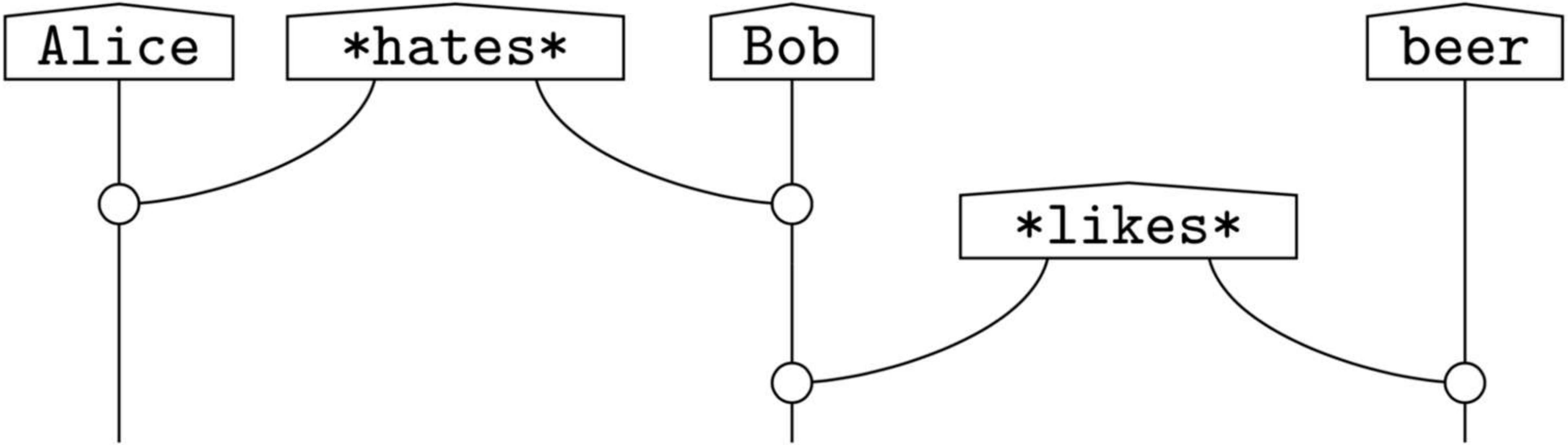

Alice
hates
Bob


hates
Alice
Bob


Alice
hates
Bob


Alice
hates
Bob

Compose!



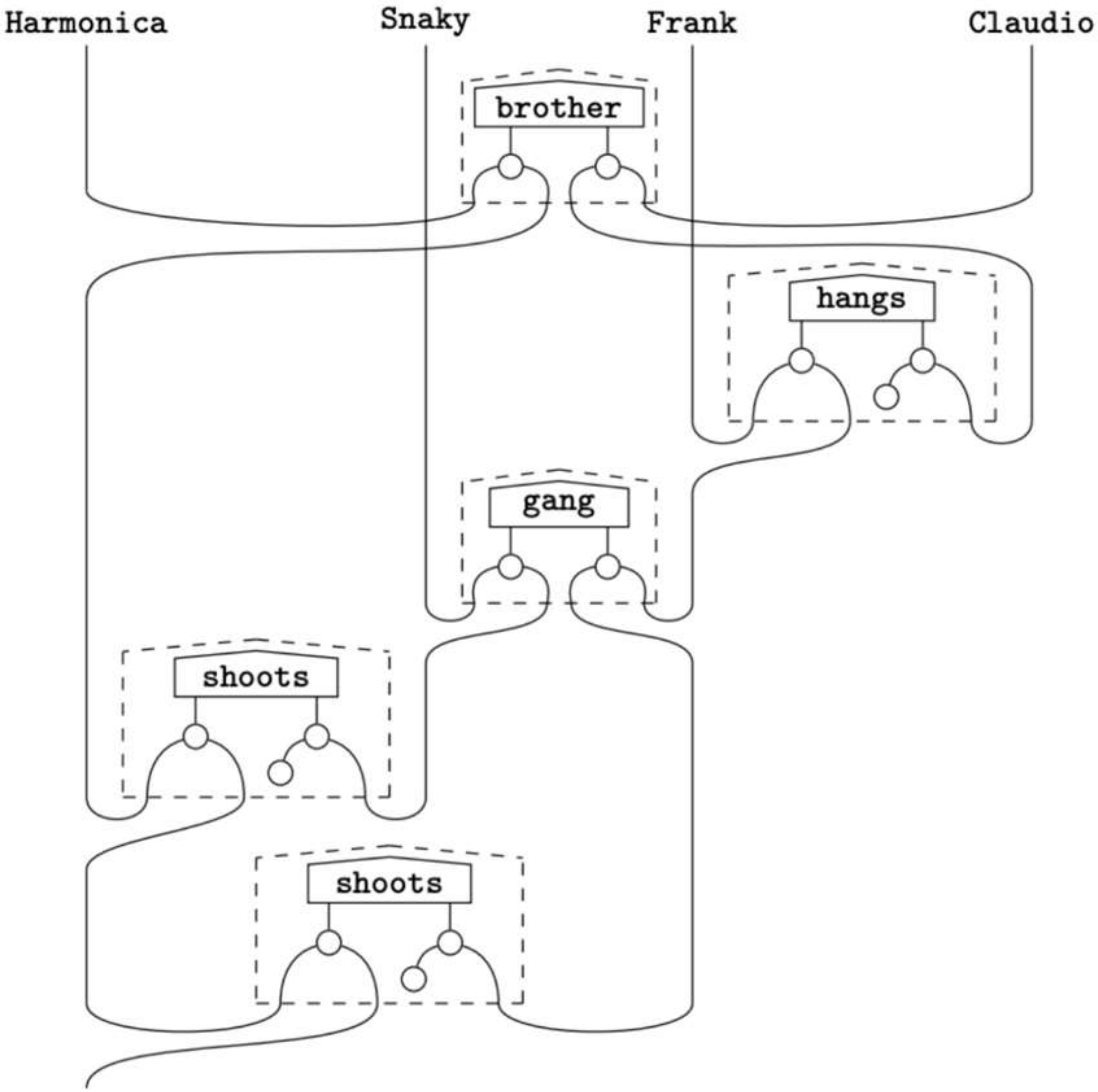
Advantages of language circuits:

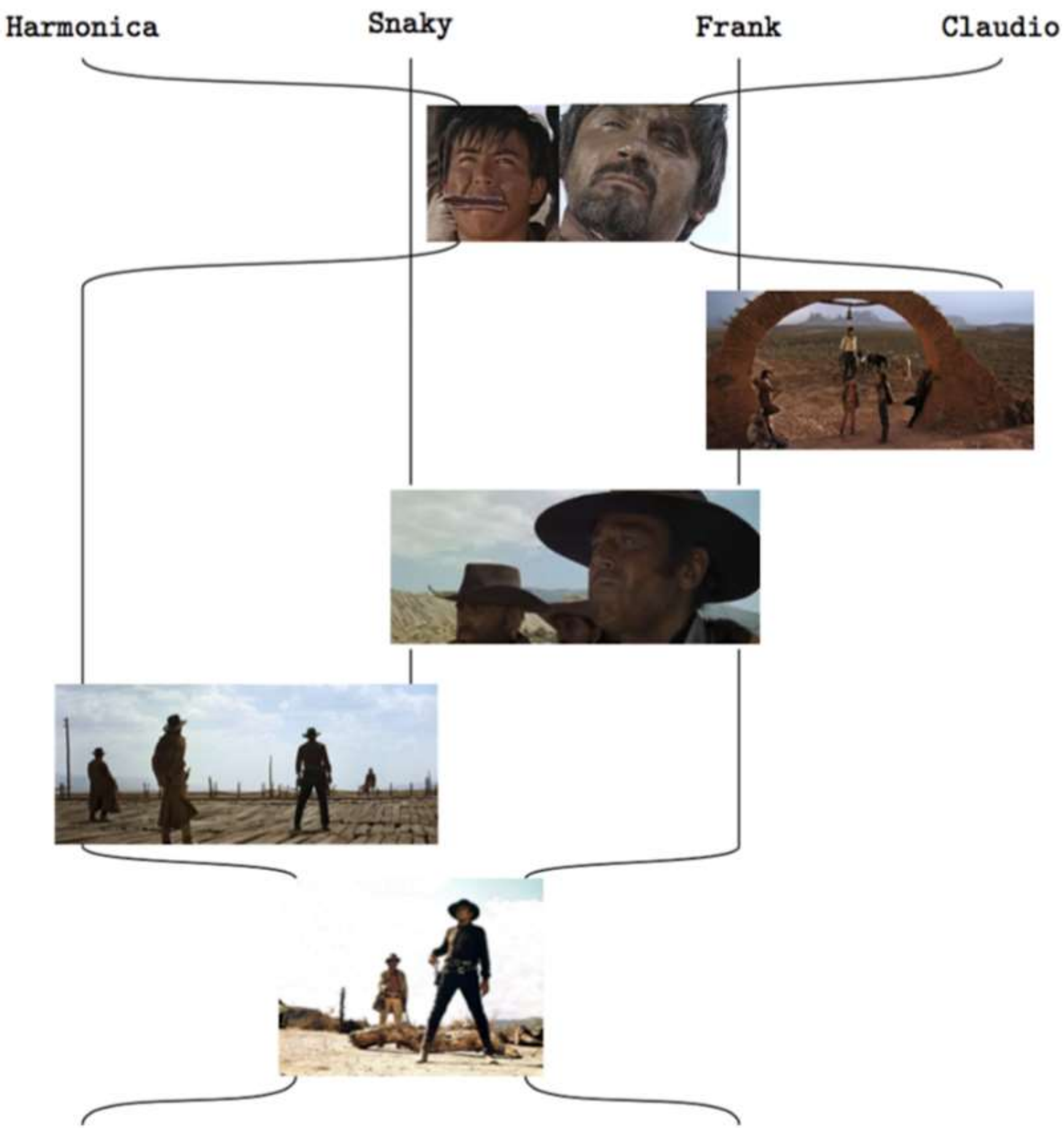
Advantages of language circuits:

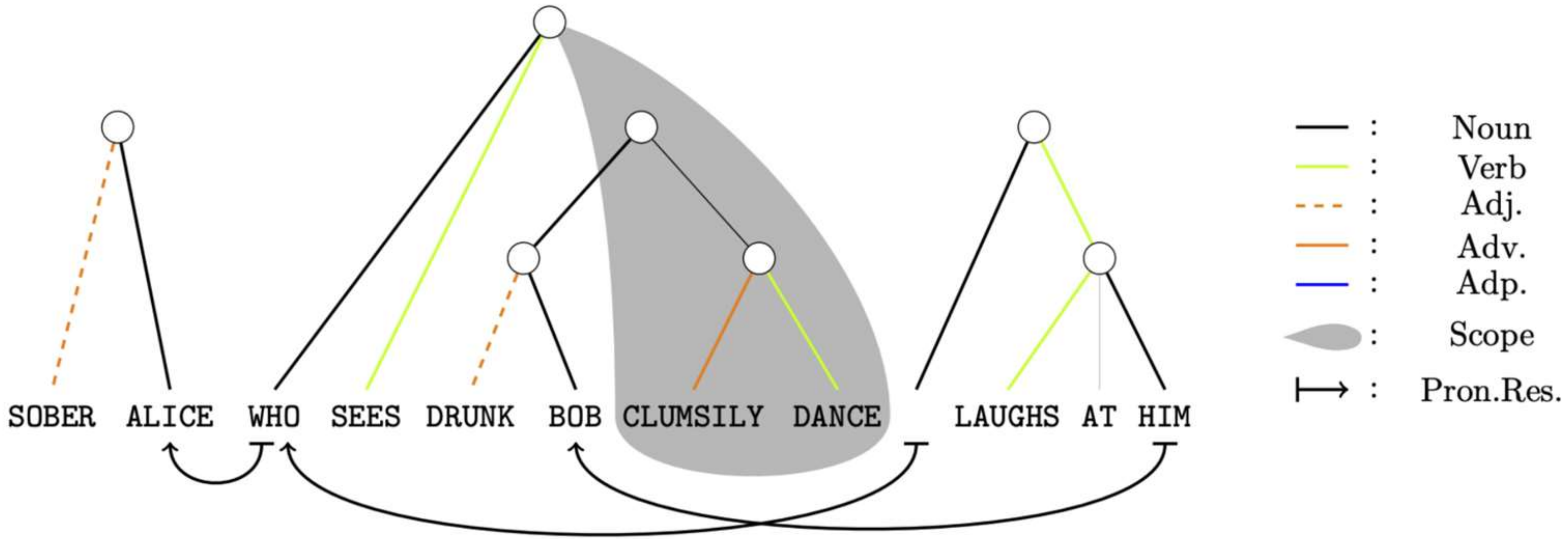
- evolving meanings

Advantages of language circuits:

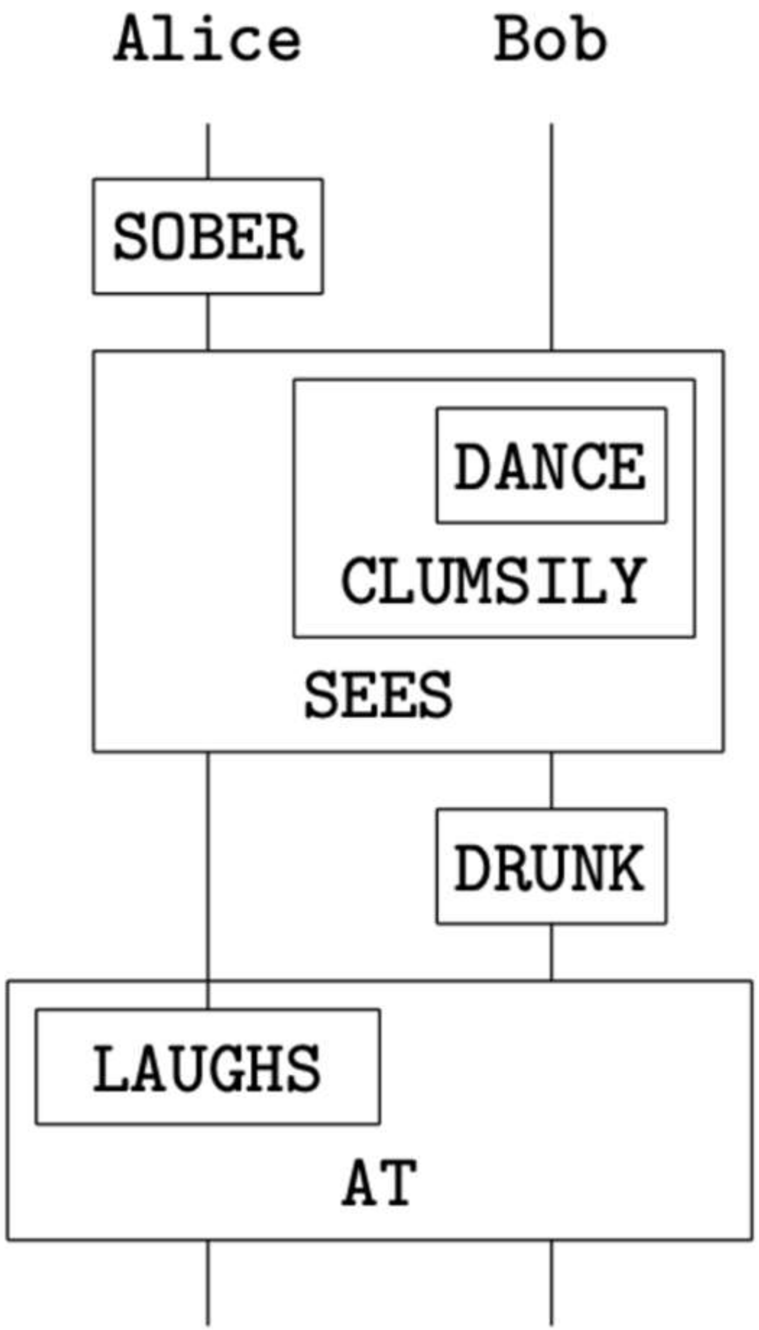
- evolving meanings
- strips off language-dependent overheads


Harmonica
Snaky
Frank
Claudio
brother
hangs
gang
shoots
shoots


Harmonica
Snaky
Frank
Claudio



SOBER
ALICE
WHO
SEES
DRUNK
BOB
CLUMSILY
DANCE
LAUGHS
AT
HIM
Noun
Verb
Adj.
Adv.
Adp.
Scope
Pron.Res.



Alice
Bob
SOBER
DANCE
CLUMSILY
SEES
DRUNK
LAUGHS
AT

Thank you!

Tomorrow: DisCoCat and DisCoCirc Experiments