Auckland University of Technology »

School of Computer & Information Sciences

(A New GOFAI Theory: How Language Works)

(Wai Yeap/albert)

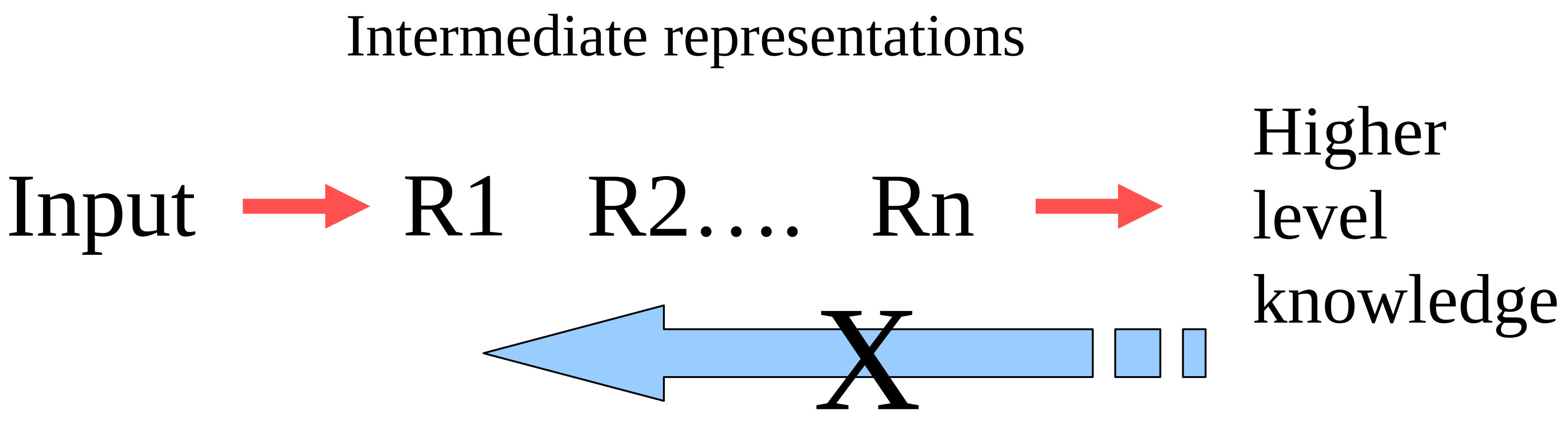
AUT

AUCKLAND UNIVERSITY OF TECHNOLOGY

TE WĀNANGA ARONUI O TAMAKI MAKAU RAU

What is GOFAI? – a gentle reminder

A GOFAI method – Marr’s approach



A GOFAI Theory of Human Language

John and Jane

Men bite dogs

We fed her chicken McNuggets.

They seem to enjoy boiling champagne.

Crain and Thornton 1998

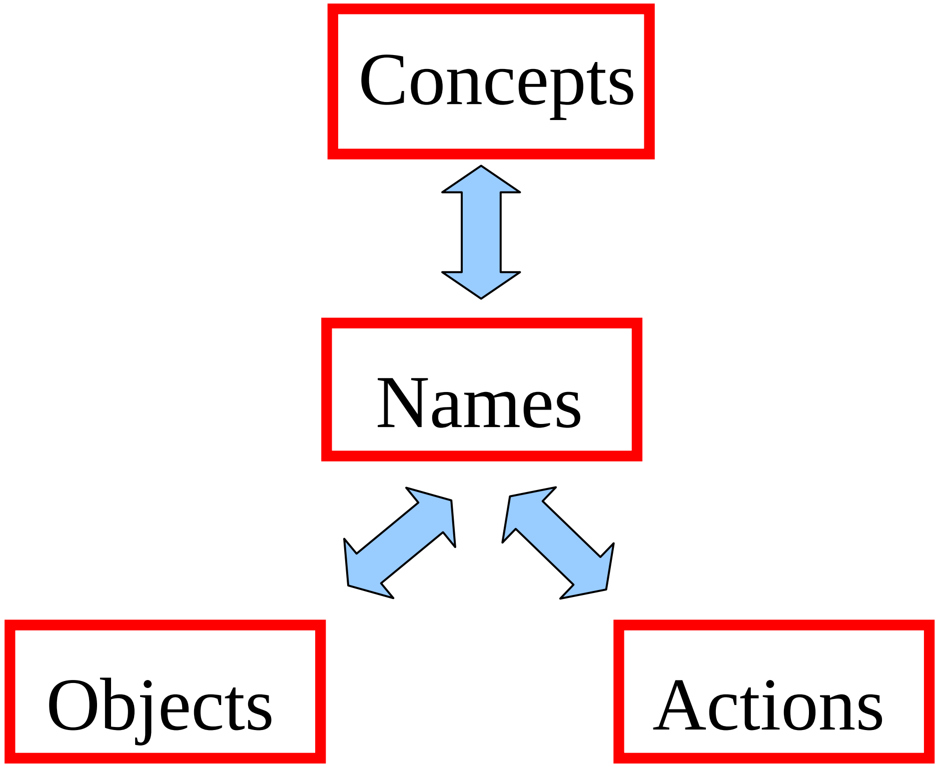
Psycholinguistic Theories of Language

No negative evidence - Baker’s Paradox

Chomsky’s universal grammar

Pinker’s semantic bootstrapping

What do children do when they learn their first language?



Mommy cookie

Daddy go

The significance of the split of the names into objects and actions lies not just in knowing how to distinguish words but more importantly knowing how different kinds of words are combined.

Consider a simple phrase:

Mama Give

What could the algorithm be?

A straightforward method:

Mama $\rightarrow$ [mama*]

Give $\rightarrow$ [give* (:actor ?L)]

A more refined algorithm at a later stage:

Mama $\rightarrow$ [mama*]

Give $\rightarrow$ [give* (:actor ?L)
(:object-of-desire ?R)]

OK, a simple algorithm

Advantages – no use of rules that explicitly required the identification of categories. Categories (?L, ?R) are learned from meanings. No rigid formal rules.

Problem – can the algorithm be developed, powerful enough, to handle the full complexity of language use?

The man the police wanted took the money

How could the basic algorithm be extended to handle the complex variations in language and in ways which do not require information not made available as input to the (child's) process?

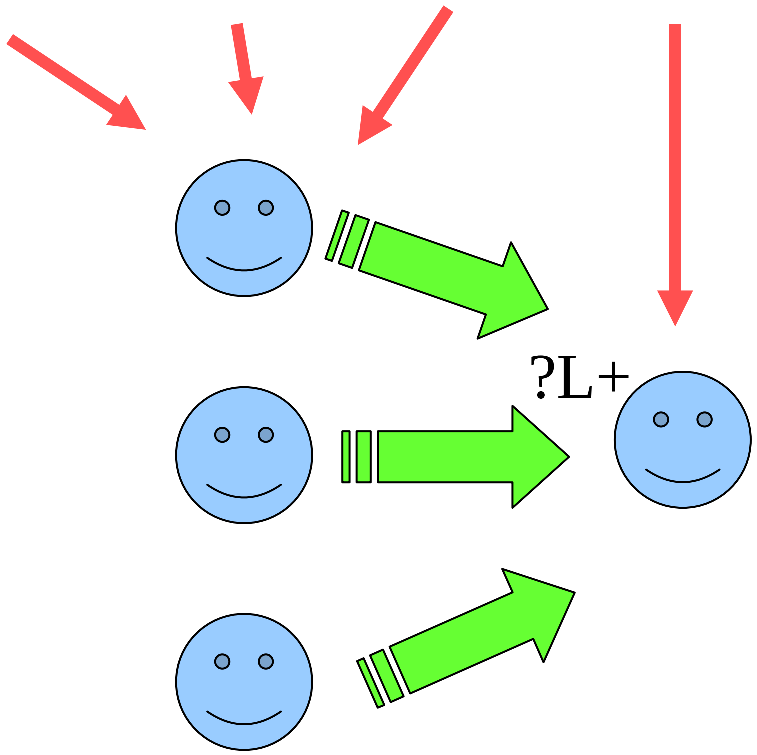
The solution lies in one’s ability to extend the initial ?L/?R labels for more sophisticated processing of semantics objects….or more appropriately, *Mental Sketches*.

The language process:

$$\begin{tikzcd}\text{Words} \arrow[r] & \text{Mental Sketches} \arrow[r] & \text{Mental Picture}\end{tikzcd}$$

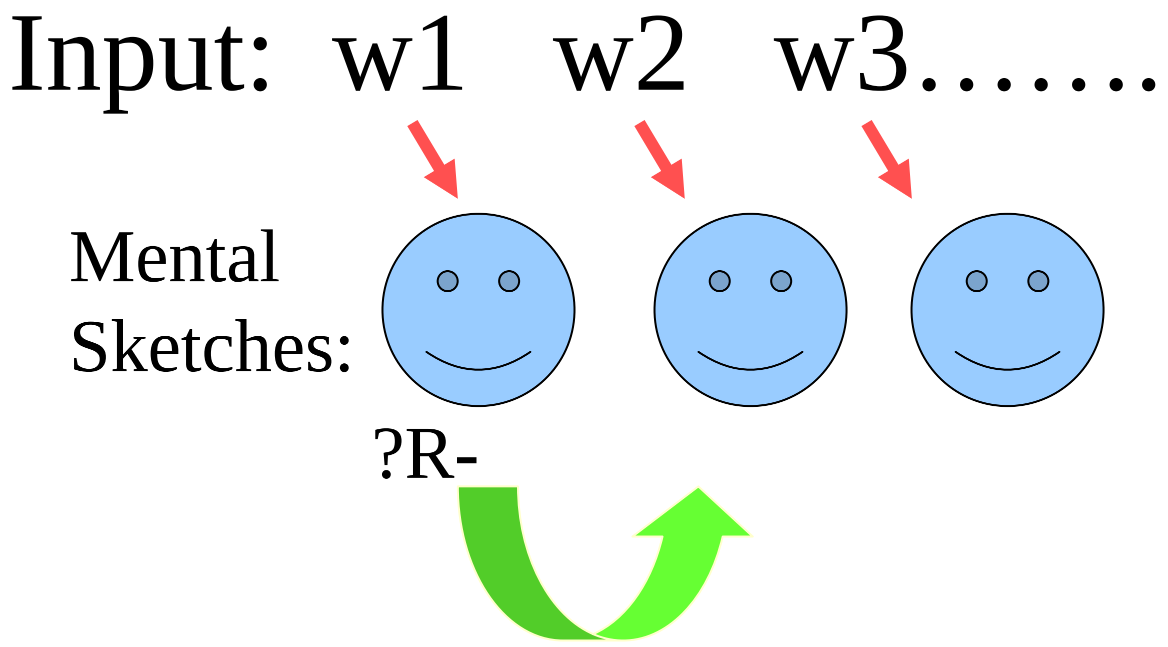
Input: w1 w2 w3 w4

Mental Sketches:



The first step: taking the mental sketch from the left (?L+) and from the right (?R+)

Eat $\longrightarrow$ [eat* (:actor ?L+) (:what ?R+)]



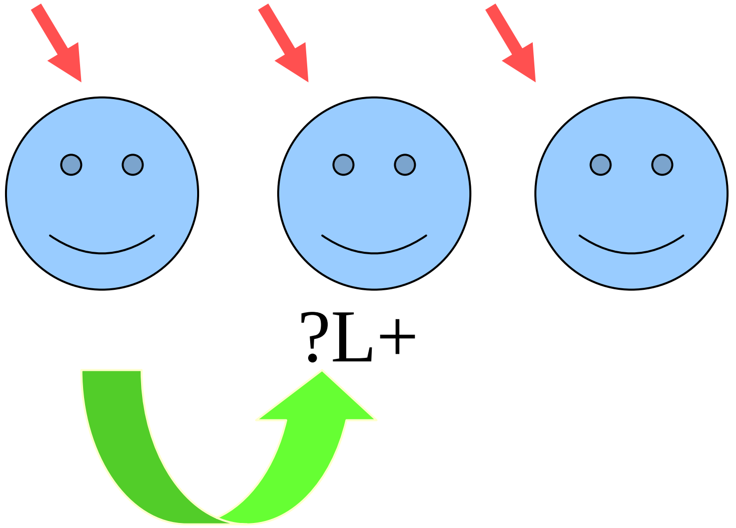
An extension: passing the information to the right

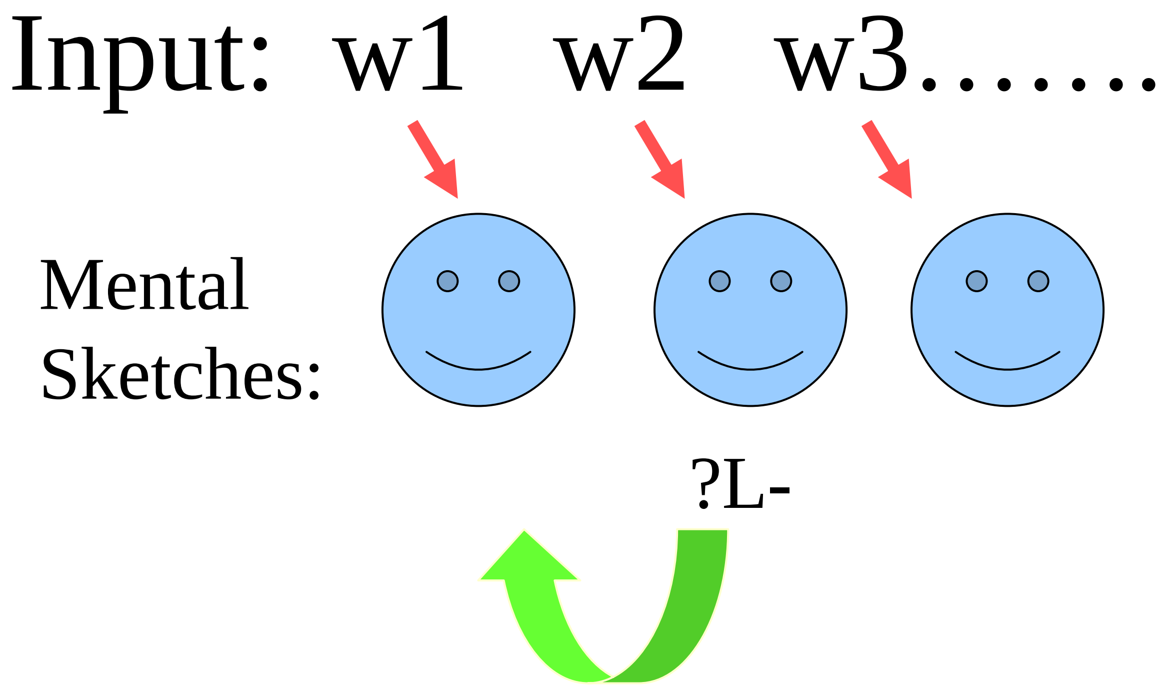
3 kinds of ?R-

Adjectives:	(?R- (:modifier big*))
Determiners:	(?R-* (:modifier the*))
Pre-determiners:	(?R-** (:modifier both*))

Input: w1 w2 w3……..

Mental Sketches:





Another extension: passing the information to the left

?L+ → ?L- (?L#)

I answered the question foolishly

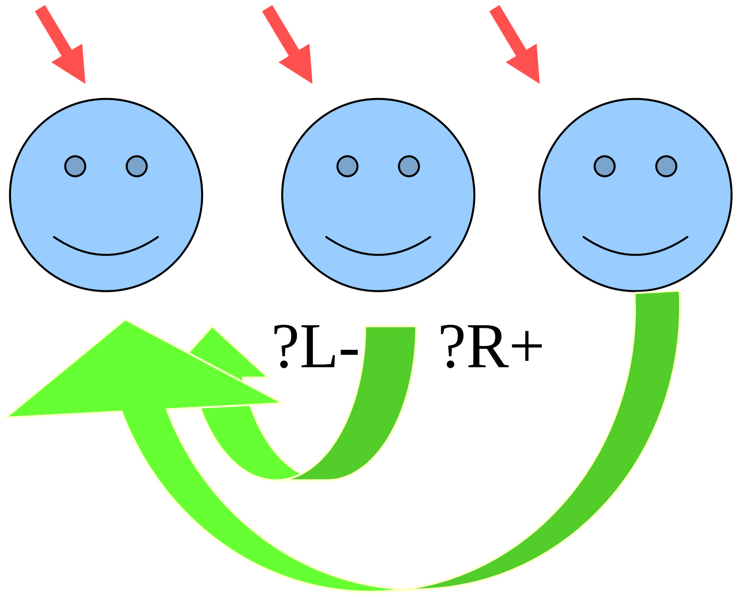
```
[answered* (:actor (I* (:noun))
           (:what (questions* (:noun)
                             (:modifier (the*))))
           (:manner (foolishly*))]
```

I foolishly answered the question

```
[answered* (:actor (I* (:noun)
                      (:manner (foolishly*))))
           (:what (questions* (:noun)
                             (:modifier (the*))))]
```

Input: w1 w2 w3…….

Mental Sketches:



?L+/?R+ $\rightarrow$?L-/?R+

I saw the car of John

```
[SAW* (:ACTOR (I* (:NOUN)))
      (:WHAT (CAR* (:NOUN)
                   (:MODIFIER (THE*))
                   (:OF* (JOHN*
                           (:PERSON
                              (:NAME (JOHN*)))))))]
```

?L+/?R+ $\rightarrow$?L-/?R+

Connectives: (?L- (and* ?R+))

Prepositions: (?L- (of* ?R+))

Wh-words: (?L- (who* ?R+))

Be-verbs: (?L- (am* ?R+))

Connectives:	(?L- (and* ?R+))
Prepositions:	(?L- (of* ?R+))
Wh-words:	(?L- (who* ?R+))
Be-verbs:	(?L- (am* ?R+))

How do we distinguish between them?

An example: I saw John in the car park

Dictionary entries:

```
(defword I (I1) (I* (:role (speaker*)) (:word (I))))
(defword saw (saw1) (saw* (:actor ?L+) (:what ?R+)))
(defword park (park1 park2) (park** (:noun))
                                   (park* (:actor ?L+) (:what ?R+)))
(defword in (in1 in2) (?L- (:in ?R+))
                              (?L# (:manner (in*))))
```

An example: **I** saw John in the car park

```
[I* (:ROLE (:SPEAKER))
    (:WORD (I))]
```

An example: **I saw** John in the car park

```
[I* (:ROLE (:SPEAKER))
    (:WORD (I))]


[SAW* (:ACTOR (I* (:ROLE (:SPEAKER)) (:WORD (I))))
      (:WHAT ?R+)]
```

An example: **I saw John in the car park**

```
[SAW* (:ACTOR (I* (:ROLE (:SPEAKER)) (:WORD (I))))
        (:WHAT ?R+)]
```

$$\downarrow$$

```
[SAW* (:ACTOR (I* (:ROLE (:SPEAKER)) (:WORD (I))))
        (:WHAT (JOHN* (:PERSON (:NAME (JOHN)))))]
```

An example: **I saw John in the car park**

```
[SAW* (:ACTOR (I* (:ROLE (:SPEAKER)) (:WORD (I))))
          (:WHAT ?R+)]
```

$\downarrow$

```
[JOHN* (:PERSON (:NAME (JOHN)))]
    [SAW* (:ACTOR (I* (:ROLE (:SPEAKER)) (:WORD (I))))
              (:WHAT ?R+)]
```

An example: **I saw John** in the car park

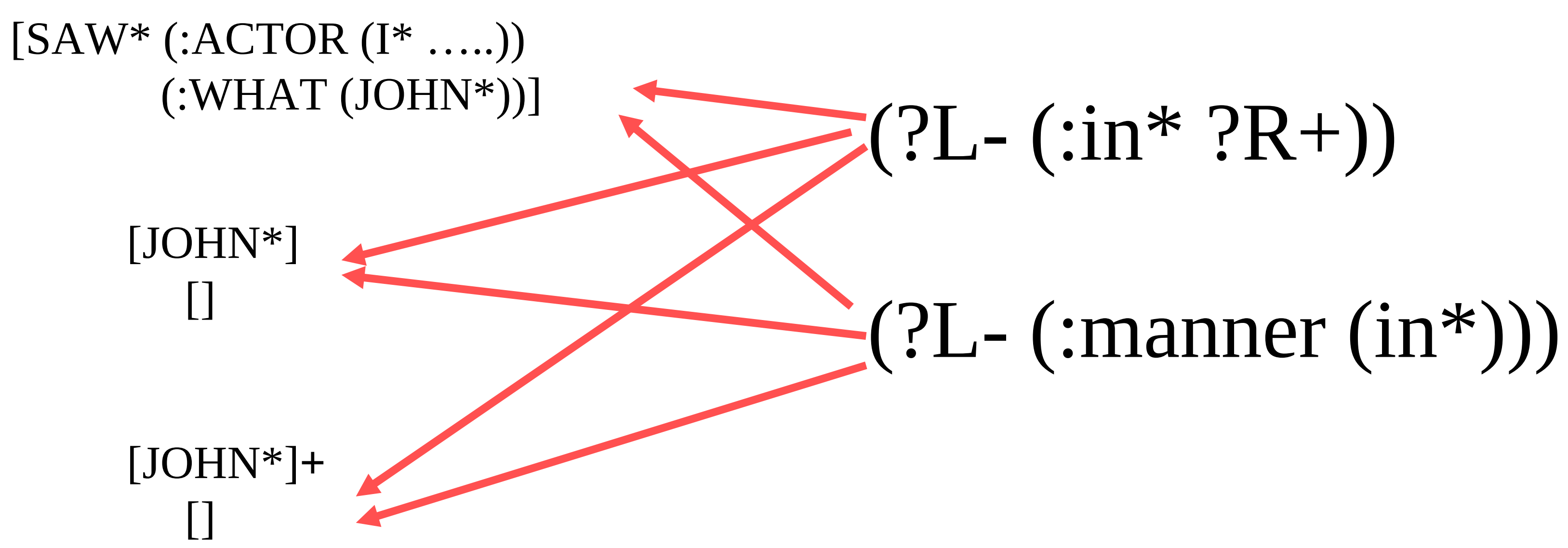
```
[SAW* (:ACTOR (I* (:ROLE (:SPEAKER)) (:WORD (I))))
          (:WHAT ?R+)]
```

$$\downarrow$$

```
[JOHN* (:PERSON (:NAME (JOHN)))]+
    [SAW* (:ACTOR (I* (:ROLE (:SPEAKER)) (:WORD (I))))
              (:WHAT ?R+)]
```

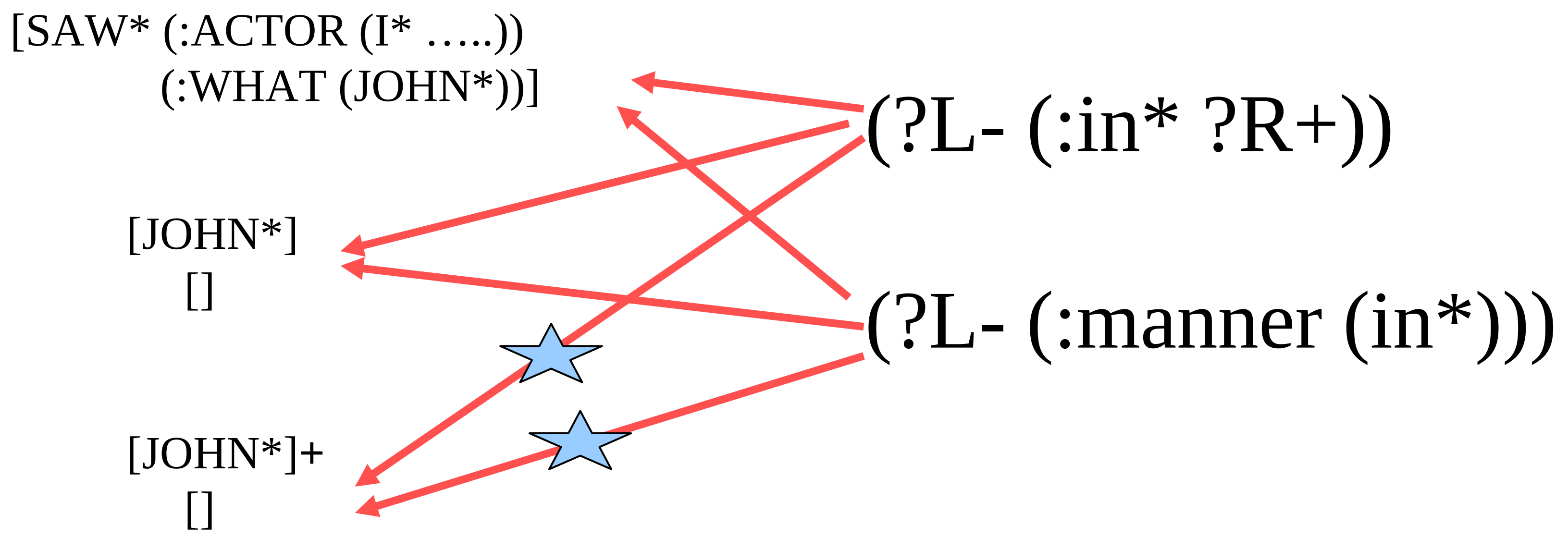
An example: **I saw John in** the car park

Current Mental Sketches | Next inputs



An example: **I saw John in** the car park

Current Mental Sketches | Next inputs



An example: **I saw John in** the car park

Current Mental Sketches

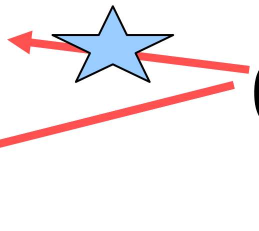
[SAW* (:ACTOR (I* …..))
(:WHAT (JOHN*))]

[JOHN*]
[]

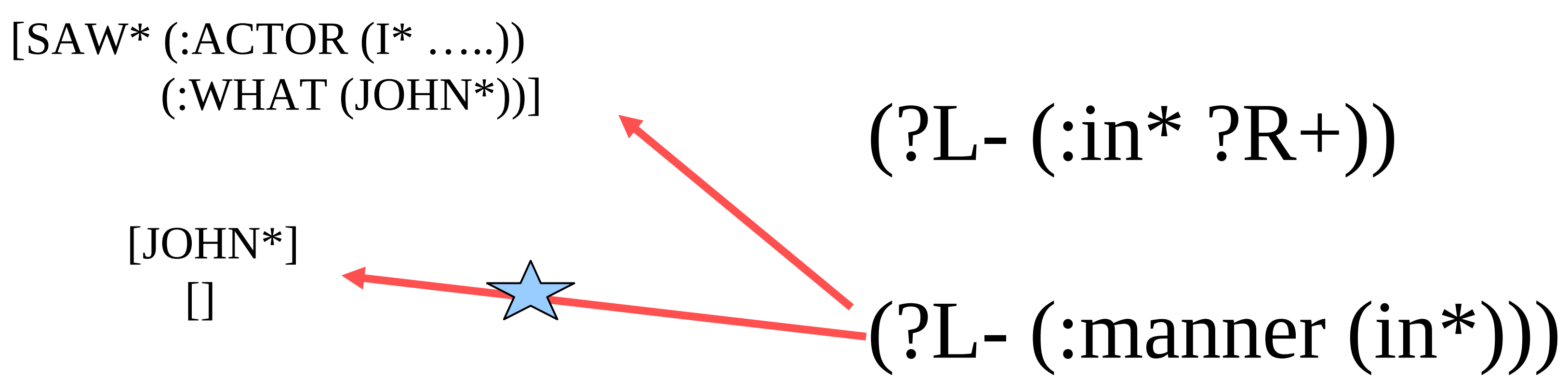
Next inputs

(?L- (:in* ?R+))

(?L- (:manner (in*)))



An example: **I saw John in** the car park



Current Mental Sketches

```
[SAW* (:ACTOR (I* …..))
      (:WHAT (JOHN*))]

[JOHN*]
[]
```

Next inputs

```
(?L- (:in* ?R+))

(?L- (:manner (in*)))
```

An example: **I saw John in** the car park

```
[JOHN* (:PERSON (:NAME (JOHN)))
             (:IN* ?R+)]
    [SAW* (:ACTOR (I* (:ROLE (:SPEAKER)) (:WORD (I))))
               (:WHAT ?R+)]

[SAW* (:ACTOR (I* (:ROLE (:SPEAKER)) (:WORD (I))))
           (:WHAT (JOHN* (:PERSON (:NAME (JOHN)))))
           (:MANNER (IN*))]
```

An example: **I saw John in the** car park

```
[JOHN* (:PERSON (:NAME (JOHN)))
            (:IN* ?R+)]
    [SAW* (:ACTOR (I* (:ROLE (:SPEAKER)) (:WORD (I))))
              (:WHAT ?R+)]

[SAW* (:ACTOR (I* (:ROLE (:SPEAKER)) (:WORD (I))))
          (:WHAT (JOHN* (:PERSON (:NAME (JOHN)))))
          (:MANNER (IN*))]
```

An example: **I saw John in the** car park

```
[JOHN* (:PERSON (:NAME (JOHN)))
           (:IN* ?R+)]
    [SAW* (:ACTOR (I* (:ROLE (:SPEAKER)) (:WORD (I))))
             (:WHAT ?R+)]

[SAW* (:ACTOR (I* (:ROLE (:SPEAKER)) (:WORD (I))))
         (:WHAT (JOHN* (:PERSON (:NAME (JOHN)))))
         (:MANNER (IN*))]
```

An example: **I saw John in the** car park

```
[?R-* (:MODIFIER (THE*))]
   [JOHN* (:PERSON (:NAME (JOHN)))
                  (:IN* ?R+)]
        [SAW* (:ACTOR (I* (:ROLE (:SPEAKER)) (:WORD (I))))
                     (:WHAT ?R+)]
```

An example: **I saw John in the car** park

```
[CAR* (:NOUN)]+
   [?R-* (:MODIFIER (THE*))]
      [JOHN* (:PERSON (:NAME (JOHN)))
                     (:IN* ?R+)]
           [SAW* (:ACTOR (I* (:ROLE (:SPEAKER)) (:WORD (I))))
                        (:WHAT ?R+)]


[CAR* (:NOUN) (:MODIFIER (THE*))]
   [JOHN* (:PERSON (:NAME (JOHN)))
                  (:IN* ?R+)]
        [SAW* (:ACTOR (I* (:ROLE (:SPEAKER)) (:WORD (I))))
                     (:WHAT ?R+)]
```

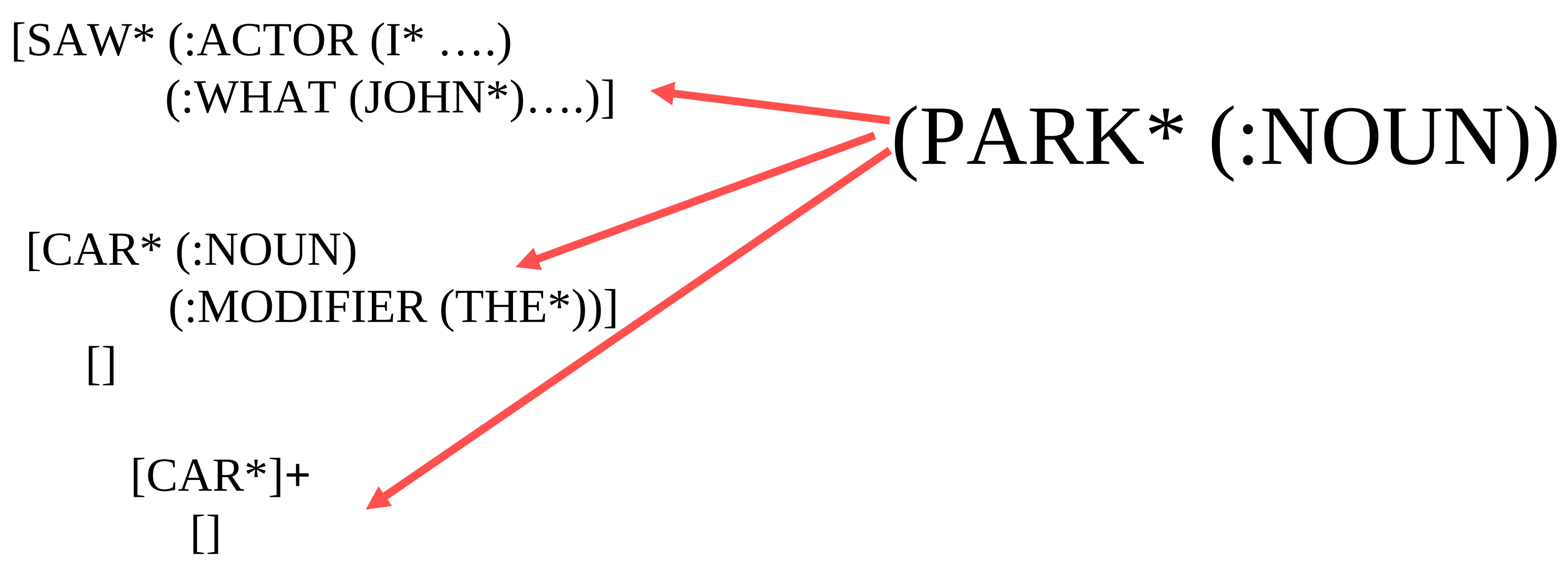
An example: **I saw John in the car** park

```
[SAW* (:ACTOR (I* (:ROLE (:SPEAKER)) (:WORD (I))))
      (:WHAT (JOHN* (:PERSON (:NAME (JOHN)))
                    (:IN* (CAR* (:NOUN)
                                (:MODIFIER (THE*)))))]
```

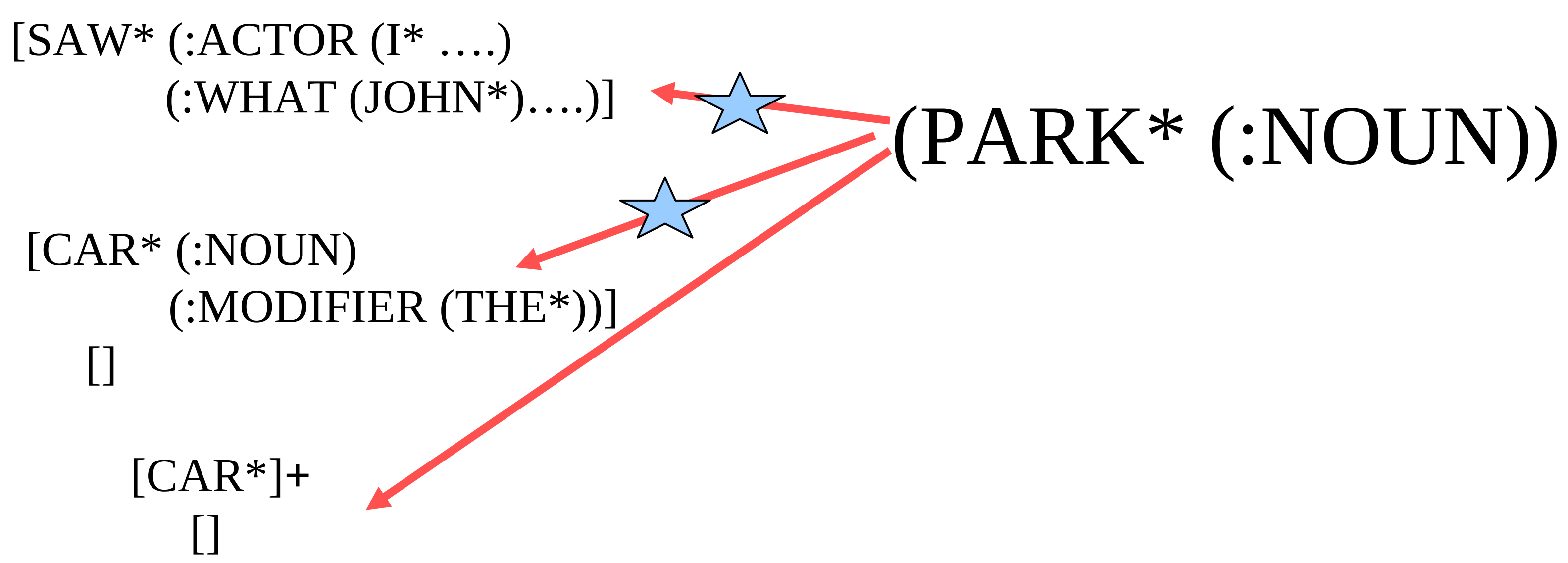
An example: **I saw John in the car park**

Current Mental Sketches

Next inputs



An example: **I saw John in the car park**



Current Mental Sketches

Next inputs

```
[SAW* (:ACTOR (I* ….)
        (:WHAT (JOHN*)….)]

[CAR* (:NOUN)
        (:MODIFIER (THE*))]
    []

      [CAR*]+
        []
```

(PARK* (:NOUN))

The set of labels created in my system:

$$\begin{tikzcd}
\text{?L+} \arrow[dr] & & \text{?R+} \arrow[dl] \arrow[dr] & & \text{?L-} \arrow[dl] & \text{?R-} \arrow[d] \\
& \text{?L+/?R+} \arrow[d] & & \text{?L-/?R+} & & \begin{array}{l}\text{?R-*} \\ \text{?R-**}\end{array} \\
& \text{?L+/?R+/?R++} \arrow[d] & & & & \\
& \text{Higher order: MS: ?R+} & & & &
\end{tikzcd}$$

Summary

We offer a new theory of language. It has 3 components:

A set of ?L/?R labels (as opposed to formal categories)

A stack

A procedure for manipulating each set of labels (as opposed to formal rules). Each procedure has 2 distinct phases – an elimination phase and a construction phase

Discussion: So, how does language work?

It begins by realizing that sounds/symbols have meanings.

When meanings of phrases are learned, one pays attention to positional information. The latter tells us how words meanings are moved between words. Knowing the meanings of each phrase then helps one to develop a set of routines to re-construct meanings of phrases.

I propose a labeling scheme and demonstrate that it is powerful enough to capture the grammar of the (English) language

Categorial Grammar

John	likes	Jane
n	(n\s)/n	n

Chris	gave	a fish	to Tigger
np	((s\np)/pp)/np	np	pp

Future Work

Can this method be extended as a basis for describing all languages – is this a new universal grammar?

Can this method explain many of the interesting observations about language use?

Will this approach be a more powerful method for practical applications?

Thank you for accepting this paper and of course for listening