

1 Simulating Early Language Games

1.1 Representation

The model works with a set of exemplars. Exemplars contain an utterance and its meaning representation. The meaning representation consists of an operator and a series of predicates. The first predicate is the most salient, containing a topic and a comment, the rest consists of context, presuppositions and associated facts:

```
"utterance"  
operator: comment(topic) presup1(entity) presup2(entity) ...
```

For example:

```
"want some juice ?"  
ynquestion: want(juice) food(juice)
```

This could be translated to first-order logical form in the following way:

$$\exists x : [want(x) \wedge juice(x) \wedge food(x)]$$

but this translation does not preserve order, which is necessary because $want(juice) \neq juice(want)$, and the expressed **want** is more salient than the implicit categorization **food**. This shortcoming of logical form is recognized in construction grammar, eg. ‘the cat bites the dog’ versus ‘the dog is bitten by the cat’ would have the exact same logical form but are nevertheless as distinct as can be.

Both predicates and their arguments can be variable by writing them as a single uppercase letter. Variable means that the information is missing from or asked in the utterance.

This representation makes no hard-and-fast distinction between what is explicitly verbalized in the utterance, and that which is understood through context, because this distinction would amount to a fully context-free, introspectable understanding of each and every word in the utterance. Instead of precisely describing the semantic structure of the utterance, this style of representation views the utterance as an ellipsis glossing over parts which can reasonably be expected to be filled in by hearers. Since this filling in of contextual details is not necessarily a linguistic phenomenon, it is assumed to have completed successfully, and present in the initial corpus of exemplars.

1.2 The Model

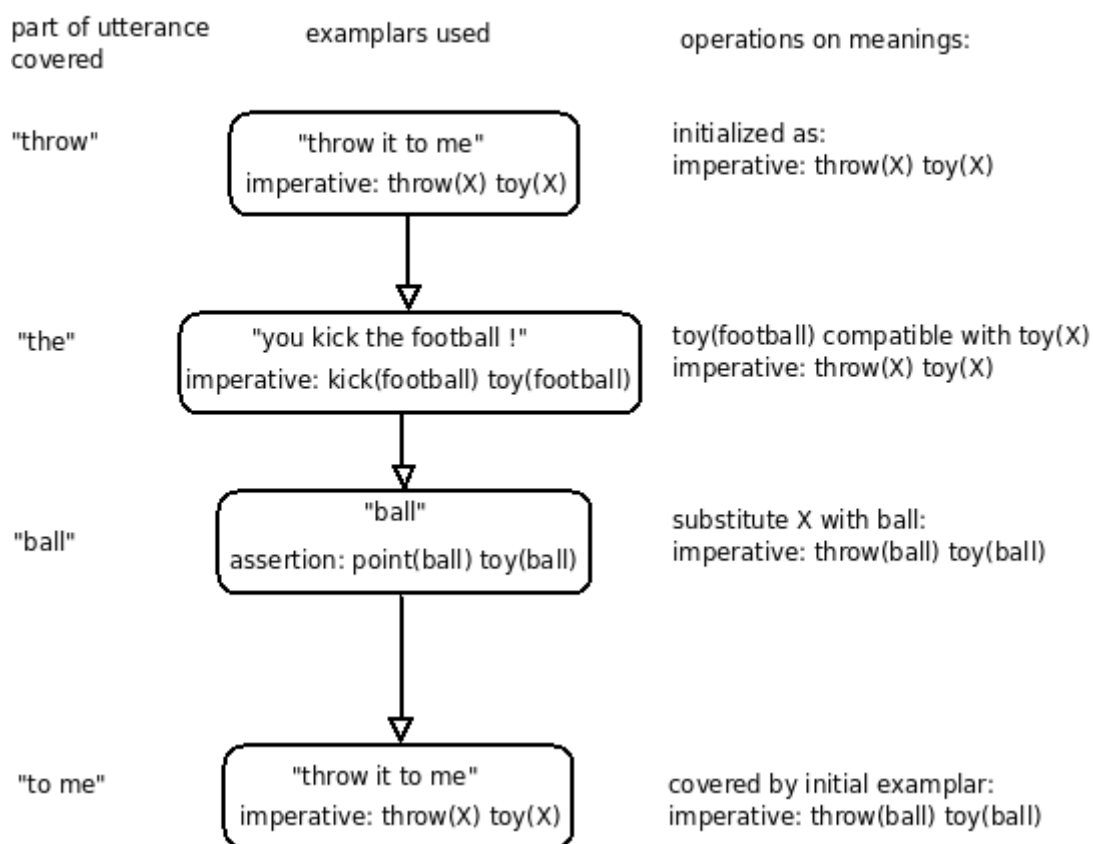
The first step in interpreting a novel utterance is finding the exemplar whose utterance that is most similar to it. This is implemented by computing the edit distance for every exemplar to the utterance to be interpreted, and picking the lowest value. The meaning of the exemplar that is found is then used as a template to which other exemplars must conform if they are to be used in

interpreting the novel utterance. An exemplar conforms to the current interpretation if it has a family resemblance with it, ie. one of its terms has a predicate in common with the current interpretation.

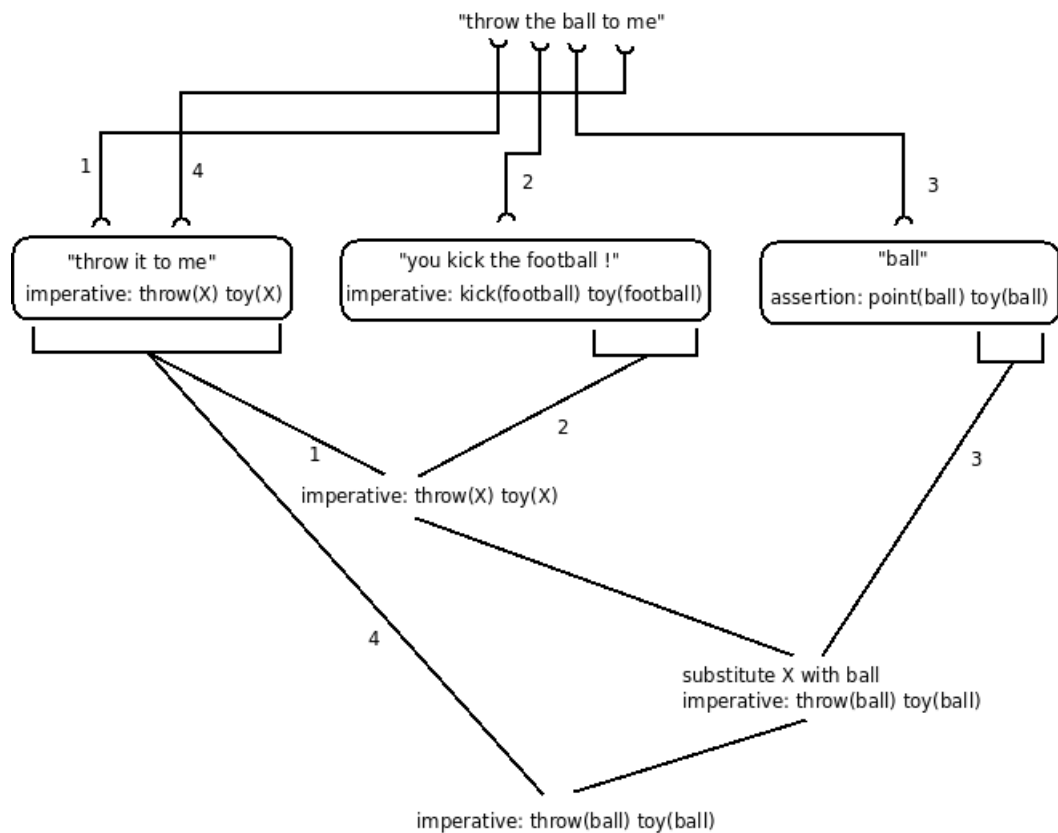
After finding the first exemplar further exemplars are sought in order to cover the remaining words in the utterance. The words are covered in a greedy fashion, the longest matching construction is used. This process currently does not perform backtracking, but this could be added for cases where first matching a shorter construction enables a longer construction to be used later.

Here is the process in a step-wise fashion:

reconstructing the meaning of "throw the ball to me"



Alternatively, it can be viewed as a resolution process:



It is possible to use demonstratives in sentences, as long as the referent is supplied on input:

```
utterance: what is this [=ball] ?
meaning: whquestion: point(X) Y(X)
interpretation: whquestion: point(ball) toy(ball)
```

Interpreting this exemplar will cause the meaning of “ball” to be inserted, in this case by substituting the variable predicate Y with the meaning of “ball” (without a variable predicate it will be concatenated).

1.3 The current corpus of annotated exemplars

- “uh” * ‘,’
- “eh” * ‘,’
- “ah” * ‘acknowledgement’
- “ok” * ‘confirmation’
- “yes” * ‘confirmation’
- “yeah” * ‘confirmation’
- “no” * ‘denial’

“ball” * assertion: point(ball) toy(ball)
 “what is this [=ball] ?” whquestion: point(ball) toy(ball)
 “throw it to me” imperative: throw(X) toy(X)
 “you kick the football !” imperative: kick(football) toy(football)
 “what does a bunny do ?” whquestion: do(X) animal(bunny)
 “hop” assertion: do(hop) animal(bunny)
 “what animal does woof woof ?” * whquestion: animal(X) do(woof)
 “what’s a kitty say ?” ’whquestion: do(X) animal(cat)’
 “meouw” ’assertion: do(meouw) animal(cat)’
 “bunny” * ’assertion: animal(bunny) do(hop)’
 “dog” * ’assertion: animal(dog) do(woof)’
 “woof woof” ’assertion: do(woof) animal(dog)’
 “want some juice ?” ynquestion: want(juice) food(juice)
 “chocolate” * ’assertion: food(chocolate)’

* denotes an utterance not taken from a corpus; the rest comes from Childes, New England corpus.

1.4 Sample output

The following output was produced by entering the parent utterances, the rest is produced by the model.

```

Parent: what does a dog do ?
        'what does a' in 'what does a bunny do ?'
meaning initialized as: whquestion: do(X) animal(bunny)
substituted (bunny) with (dog)
        'dog' in 'dog'
        and 'assertion: animal(dog) do(woof)' matches
'whquestion: do(X) animal(bunny)'
        'do ?' in 'what does a bunny do ?'
        and 'whquestion: do(X) animal(bunny)' matches
        'whquestion: do(X) animal(dog)'
        interpretation: whquestion: do(X) animal(dog)
substituted (X) with (woof)
        reaction: assertion: do(woof) animal(dog)
Child: woof woof
  
```

Asking the opposite question is also possible, since the order represents salience:

Parent: what animal does woof woof ?
 'what animal does woof woof ??' in 'what animal does woof woof ??'
 meaning initialized as: whquestion: animal(X) do(woof)
 interpretation: whquestion: animal(X) do(woof)
 substituted (X) with (dog)
 reaction: assertion: animal(dog) do(woof)
 Child: dog

Combining words in a non-sensical manner (colorless green ideas...) fails to elicit a response (no production without comprehension):

Parent: what does chocolate do ?
 'what does' in 'what does a bunny do ??'
 meaning initialized as: whquestion: do(X) animal(bunny)
 interpretation:
 reaction:
 Child: eh

This behavior seems to be supported by the observation that children's utterances are always meaningful combinations of words, and a reaction to their most meaningful interpretation of the parent's utterance.

The use of demonstratives:

Parent: what does this [=bunny] animal do ?
 'what does' in 'what does a bunny do ??'
 meaning initialized as: whquestion: do(X) animal(bunny)
 'this' in 'what is this ?'
 and 'whquestion: point(X) Y(X)' matches 'whquestion: do(X) animal(bunny)'
 substituted (bunny) with (bunny)
 demonstrative dereferenced: whquestion: do(X) animal(bunny)
 'animal do' in 'what animal does woof woof ?'
 and 'whquestion: animal(X) do(woof)' matches 'whquestion: do(X) animal(bunny)'
 '??' in 'what is this ?'
 and 'whquestion: point(X) Y(X)' matches 'whquestion: do(X) animal(bunny)'
 interpretation: whquestion: do(X) animal(bunny)
 substituted (X) with (hop)
 reaction: assertion: do(hop) animal(bunny)
 Child: hop