COSC451: Artificial Intelligence

Lecture 16: How infants learn syntax

Alistair Knott

Dept. of Computer Science, University of Otago

Recap

Last lecture, we looked at how infants learn single words.

- Cross-situational learning, and the role of the phonological loop
- The role of joint attention and intention recognition

Recall from Lecture 15: word-meaning mappings and syntactic processing appear to involve different brain areas.

- Word meanings: ‘Wernicke’s area’ and associated temporal areas
- Syntax: Broca’s area.

Today: a model of what’s happening in Broca’s area.

Outline of today's lecture

1. Learning syntax: early developmental stages
2. The nativist-empiricist debate
3. Empiricist models of syntax
4. A new model of syntactic processing

Outline of today's lecture

1. Learning syntax: early developmental stages
2. The nativist-empiricist debate
3. Empiricist models of syntax
4. A new model of syntactic processing

Learning syntax: early developmental stages

Early syntactic development has some fairly well agreed stages.

1. Single-word utterances (holophrases).

Utterances in service of specific *goals*.

- The goal can be ‘declarative’ (e.g. *car!*)
- The goal can be ‘imperative’ (e.g. *doggy! more!*)

It’s only when children have learned the mapping between meanings and words that such utterances become effective.

- But children still have to learn that (in some situations) ‘entering verbal mode’ can be a means to achieving their goals.

Learning syntax: early developmental stages

2. Simple two-word utterances.

Word combinations: unstructured collections of words.

- E.g. *my . . . cup! cup . . . my!*

Pivot schemas: two word units structured around a single word

- E.g. *my cup! my cake!* [*my X*]

Tomasello: pivot schemas support some generalisation, but are mainly based on surface word ordering conventions.

Learning syntax: early developmental stages

3. Item-based syntactic constructions

At 18 months, children begin to understand simple transitive sentences.

Around 24 months: the earliest 'syntactic constructions'.

- Children begin to produce transitive sentences.
- Children begin to use syntactic function words (e.g. *the*, *of*) and inflections (e.g. *likes*).

The interesting thing about early constructions is that they tend to be *tied to specific words*.

- *Open it with this.*
- *He hit me this.*

Learning syntax: early developmental stages

4. Progressively more complex syntactic constructions.

At this point, utterances are complex enough that you need a proper syntactic theory to chart development.

That's where things start to get contentious.

Outline of today's lecture

1. Learning syntax: early developmental stages
2. The nativist-empiricist debate
3. Empiricist models of syntax
4. A new model of syntactic processing

The nativist-empiricist debate

There's a huge debate between nativists and empiricists in developmental linguistics.

- Nativists believe that infants are born with 'knowledge' of the universal properties of language.
  All they have to learn from their environment are the parameter settings which define their particular language.
- Empiricists believe that infants use *general-purpose learning mechanisms* to acquire language.
  They learn language 'from scratch'.

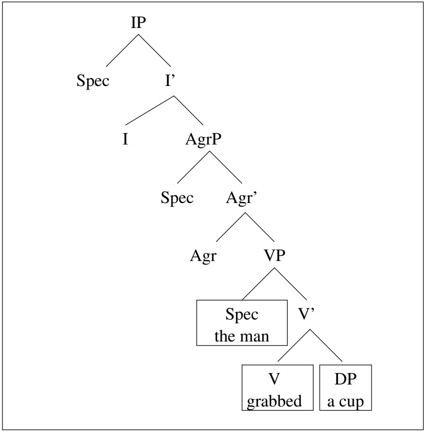
An example of a nativist model: Minimalism

Recall:

- The Minimalist model of ‘The man grabbed a cup’ holds that the same LF structure underlies this sentence in every language.
- This LF structure contains multiple positions for the agent, patient and inflected verb. (Because these items ‘move’ during derivation.)
- Children are born knowing how to derive the LF representation.
- What they have to learn is the *mapping from LF to PF*.
  I.e. whether to ‘read out’ items before or after movement.

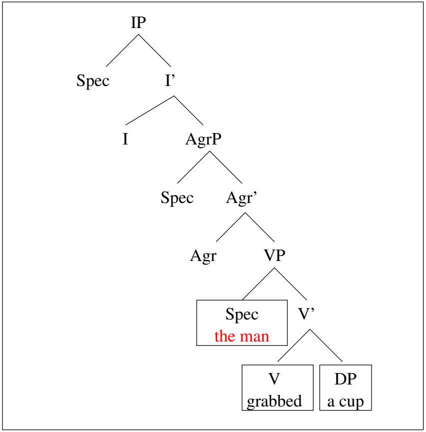
An example of a nativist model: Minimalism

Here's the LF derivation of our example sentence.



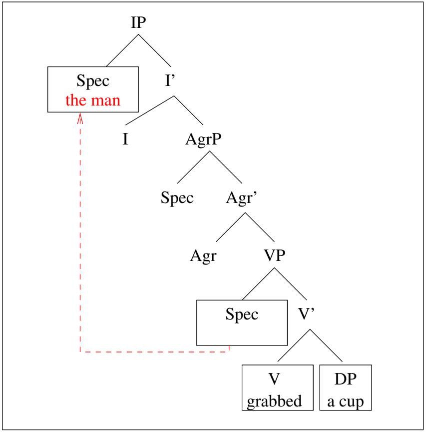
An example of a nativist model: Minimalism

Here's the LF derivation of our example sentence.



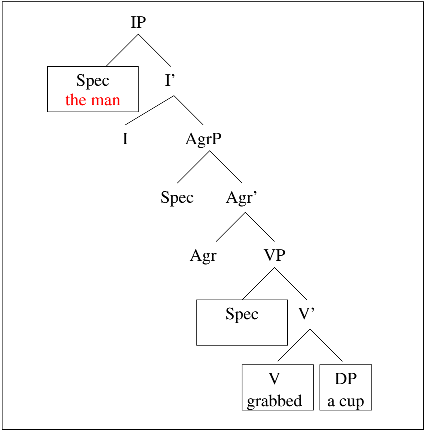
An example of a nativist model: Minimalism

Here's the LF derivation of our example sentence.



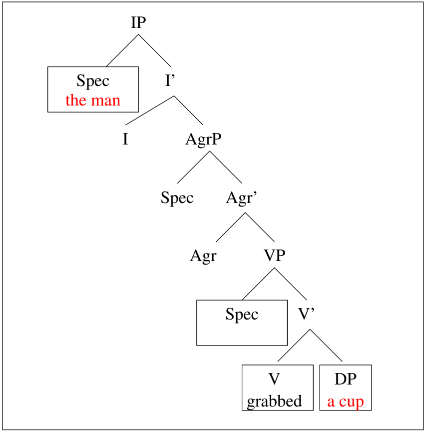
An example of a nativist model: Minimalism

Here's the LF derivation of our example sentence.



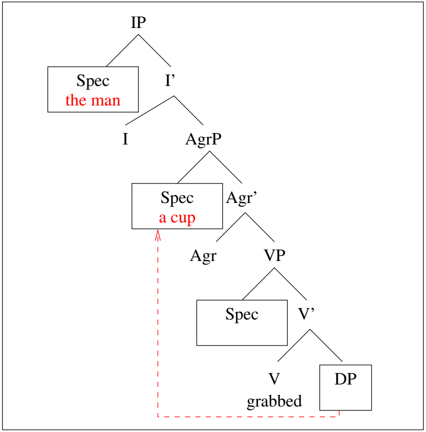
An example of a nativist model: Minimalism

Here's the LF derivation of our example sentence.



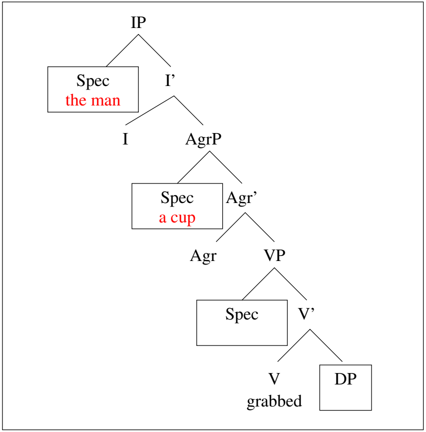
An example of a nativist model: Minimalism

Here's the LF derivation of our example sentence.



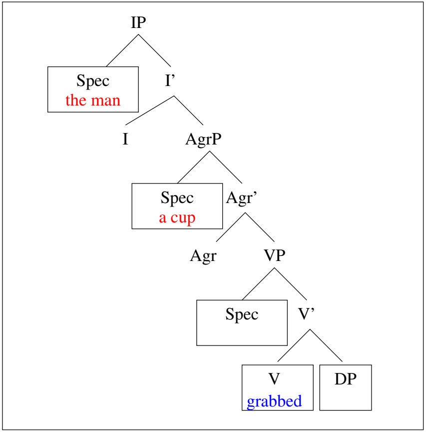
An example of a nativist model: Minimalism

Here's the LF derivation of our example sentence.



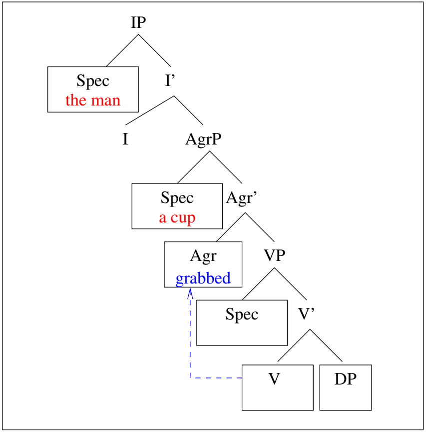
An example of a nativist model: Minimalism

Here's the LF derivation of our example sentence.



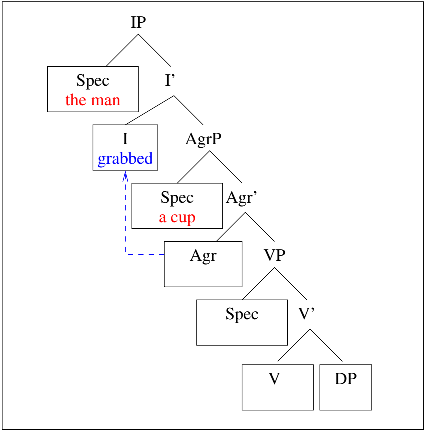
An example of a nativist model: Minimalism

Here's the LF derivation of our example sentence.



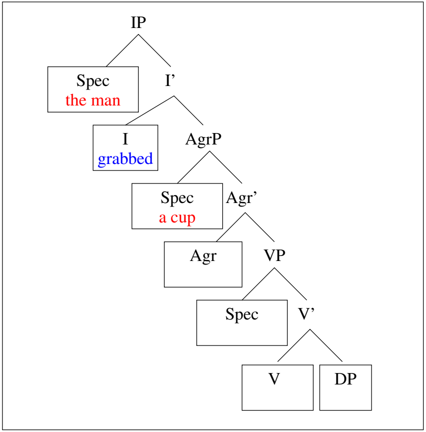
An example of a nativist model: Minimalism

Here's the LF derivation of our example sentence.



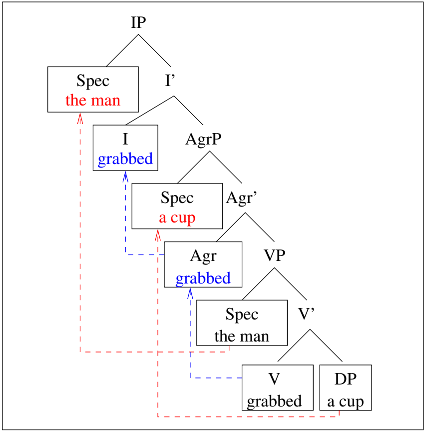
An example of a nativist model: Minimalism

Here's the LF derivation of our example sentence.



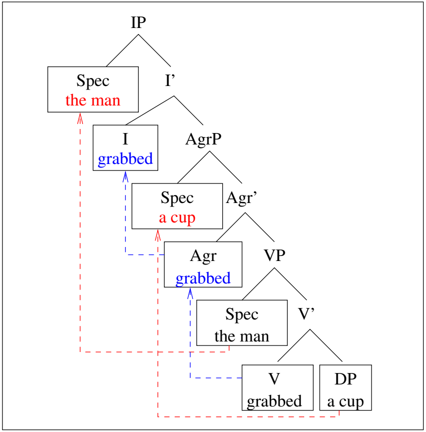
An example of a nativist model: Minimalism

Here's the LF derivation of our example sentence.



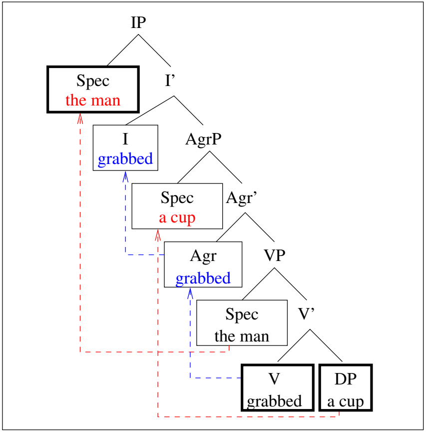
An example of a nativist model: Minimalism

Both agent and patient can be read out before or after movement.
The inflected verb can be read out in three positions.
All the child has to learn is ‘when to read out each item’.



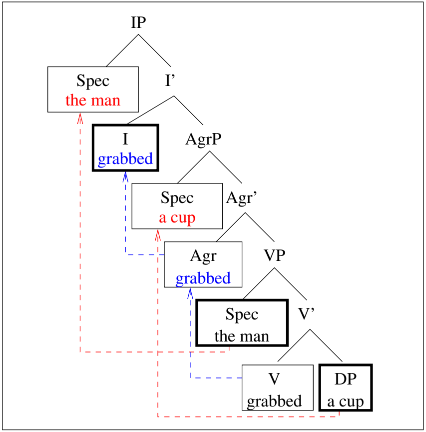
An example of a nativist model: Minimalism

In English, we read out as follows:



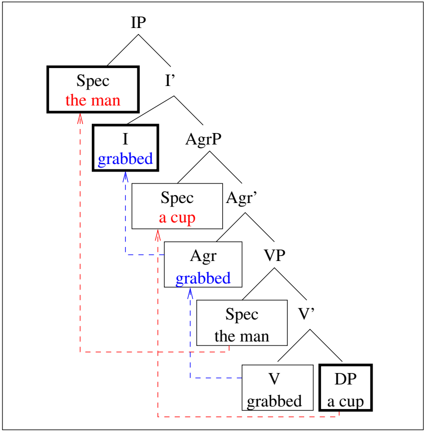
An example of a nativist model: Minimalism

In Maori, we read out as follows:



An example of a nativist model: Minimalism

In French/Italian, we read out as follows:



Some arguments for a nativist position

1. 'Poverty of the stimulus' arguments. (Chomsky, 1980)
   - 'There's not enough information in language exposure data to learn a language.'
   - 'Language is just too complex to be able to learn from data.'

2. Arguments from pidgins and creoles (Bickerton, 1981)
   - Pidgins are languages which are 'invented' when two language communities meet, and need to communicate.
     They are not true natural languages.
   - Children who grow up in communities speaking a pidgin language develop a creole.
     Creoles have all of the syntactic complexity of 'established' natural languages.

Outline of today's lecture

Empiricist models of syntax

Empiricist linguists argue that children have very powerful general-purpose learning mechanisms.

- These are sufficient to acquire a language without (much) innate language-specific machinery.

The training data: utterances occurring in communicative contexts.

- There are regularities *within utterances*.
- There are regularities *linking utterances and contexts*.

Children have pattern-finding mechanisms which pick up these regularities.

What pattern-finding mechanisms are involved?

What pattern-finding mechanisms are involved?

1. A mechanism which finds regularities in sequential data.
   - Consider the following sequence: *John went to the...*
     What word comes next?
   - We've already seen that infants can pick up regularities in a stream of phonemes (Saffran *et al.*).
     ga bi ro to ba di ga bi ro to ba di

What pattern-finding mechanisms are involved?

1. A mechanism which finds regularities in sequential data.
   - Consider the following sequence: *John went to the...* What word comes next?
   - We've already seen that infants can pick up regularities in a stream of phonemes (Saffran *et al.*). *ga bi ro to ba di ga bi ro to ba di*

2. A mechanism which finds mappings between pairs of *complex patterns*.
   - We've already hypothesised such a mechanism in our accounts of the mirror system.
   - It's also attested in our ability to perform analogical reasoning.

What sorts of pattern are found?

What sorts of pattern are found?

1. Patterns are *statistical tendencies*, rather than universal rules.

A traditional grammar divides sentences discretely into 'well-formed' and 'ill-formed'.
Empiricist language models often assign *probabilities* to sentences.

John went to the pub   *John went to the ?? cup*

Traditional grammar works with 'cleaned-up' sentences, with pauses, false starts, repetitions etc removed.

- Chomsky distinguished between syntactic competence and performance. He saw grammar as modelling competence.

Empiricist grammars tend to be trained on 'real' language data.

What sorts of pattern are found?

2. Patterns are often patterns in *surface language*.

In generative grammar, most of the rules are about deriving LF.

- There are no rules about the 'surface form' of a sentence.

However, in language, there appear to be lots of regularities which can *only* be expressed as surface regularities.

- The classic example is idioms.

Idioms

An idiom is an arbitrary sequence of words which collectely have an arbitrary semantic interpretation.

E.g. *by and large* (meaning 'typically').

- The meaning of this phrase doesn't come from the meanings of its individual words.
- It doesn't conform to any general syntactic rules.

Idioms often have 'slots', which can be filled by syntactically regular constituents.

- *Far be it from NP to VP*.

Idioms are often syntactically regular, even though their meaning is not compositional.

- *NP kicked the bucket*.

Idioms in the nativist-empiricist debate

Empiricist linguists argue that idioms are very common in language.

They argue that there's a *continuum* of idiomaticity.

- At one end, there are 'pure' idioms (e.g. *by and large*.
- In the middle there are idioms containing 'slots', and grammatically regular idioms.
- At the other end there are statistical tendencies. E.g. *went to the pub*, *give up*, *pull over*...

Empiricist models are well-suited for capturing idioms.

- Idioms are *statistical* regularities in *surface language*, mapped to arbitrary semantic/pragmatic patterns.

Minimalist models have real difficulties with idioms.

Idioms in a Minimalist model

If idioms are *continuous*, they can simply be treated as multi-word lexical items.

- E.g. *Winnie-the-Pooh*, *by-and-large*. . .

The difficulty is with *non-continuous idioms*, and with idioms which retain some degree of syntactic regularity.

- *Take NP to task* (= criticise NP)
- *NP let the cat out of the bag*
  The cat was let out of the bag by NP

There's nothing in Minimalism which can explain these constructions.

Empiricist models of language acquisition

Empiricist models of language acquisition have an easier time explaining the different stages of syntactic development.

- Infants begin by detecting simple statistical regularities in surface language, and map these to semantic representations.
- Then they identify progressively more complex regularities.

Minimalism is an account of 'mature' language competence; it's not clear how this emerges during development.

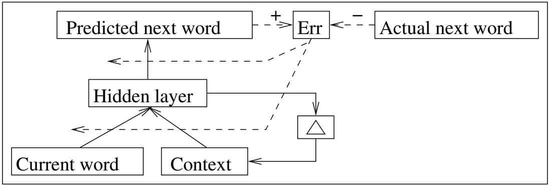
Simple recurrent networks

Obviously, empiricists need to propose models of the learning architectures which infants are using to learn patterns in language.

- One of the key models is the simple recurrent network (SRN; Elman, 1990).

A SRN takes as input a sequence of words (one word at a time). It is trained to *predict the next word in the sequence*.

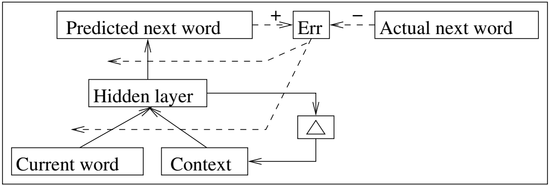
Simple recurrent networks



A SRN maintains a context representation, which is a copy of its hidden layer at the previous timestep.

- The context rep holds a history of recent inputs.
- After training, the context units can be interpreted as holding a representation of the most common sequences in the training data.

Simple recurrent networks



A trained SRN can't (normally) predict *exactly* which word will come next.

- It can distinguish between those words which are *likely* to come next, and those which are *unlikely*.
- It's basically a model of *syntax*.

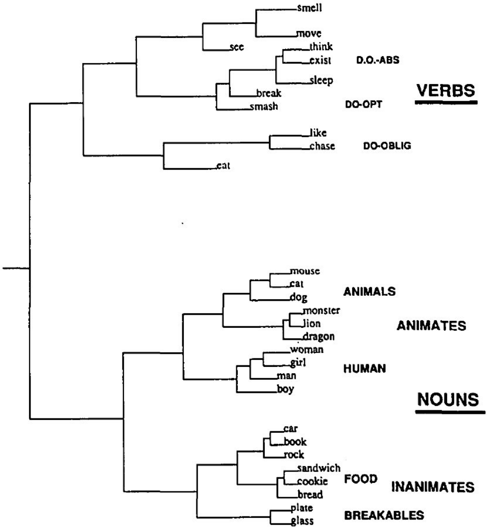
Simple recurrent networks

Interestingly, after training, words from the same syntactic (and even semantic) categories generate similar patterns of activation in the hidden layer of an SRN.

- This is because words from the same syntactic/semantic categories tend to occur in the same (surface) contexts.

Overleaf is a diagram showing how the activities hidden-unit word representations cluster after training.

Simple recurrent networks



Adding semantics to simple recurrent networks

As just described, SRNs can learn two things (from scratch):

- A (very surface-y) model of syntactic structure.
- A taxonomy of syntactic / semantic word categories.

That's very useful...
However, if we want to model the mapping between syntax and semantics, we need to extend the model.

Outline of today's lecture

A new model of syntactic processing

We already have an account of what semantic representations look like. (For concrete sentences.)

- When we 'entertain the meaning' of a concrete sentence, we internally rehearse a SM sequence stored in WM.

I need to give a model of how children learn to generate *surface word sequences* from these SM replay operations.
I.e. of how children learn the mapping from LF to PF.

- The model should ideally include an account of learned surface patterns in language.
- It should also support an account of the different stages in syntactic development.

The basic idea

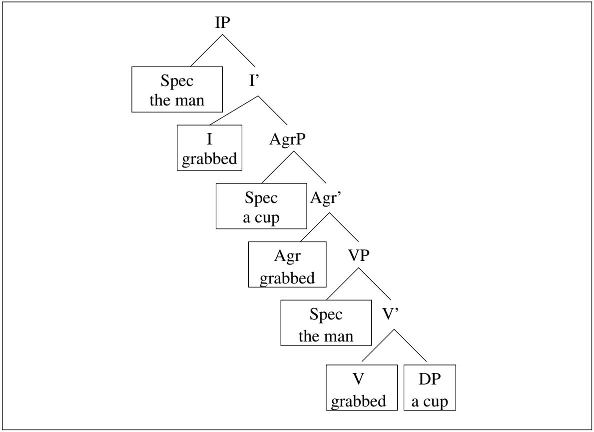
In our account, the semantics (LF) of a sentence takes the form of a rehearsed SM sequence.

- We need a network which learns to map a SM sequence onto a sequence of words.
- The network needs to learn which SM representations in the sequence need to be ‘pronounced’, and which need to be skipped.

I’ll introduce the network bit by bit.

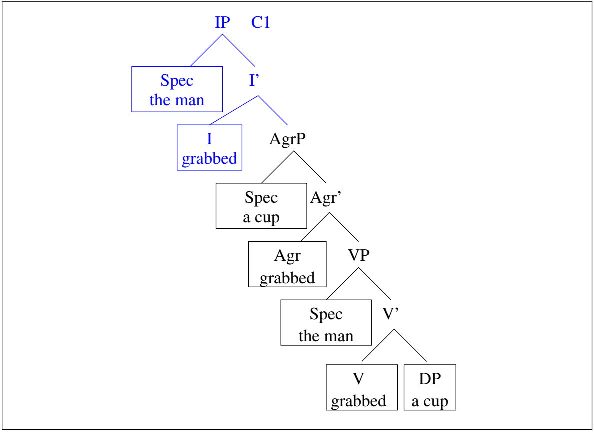
The basic idea

I'm thinking of the LF representation as a sequence of contexts, with two SM representations evoked in each context.



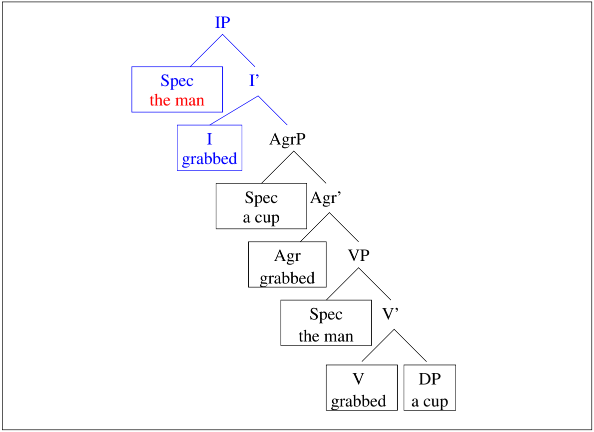
The basic idea

I'm thinking of the LF representation as a sequence of contexts, with two SM representations evoked in each context.



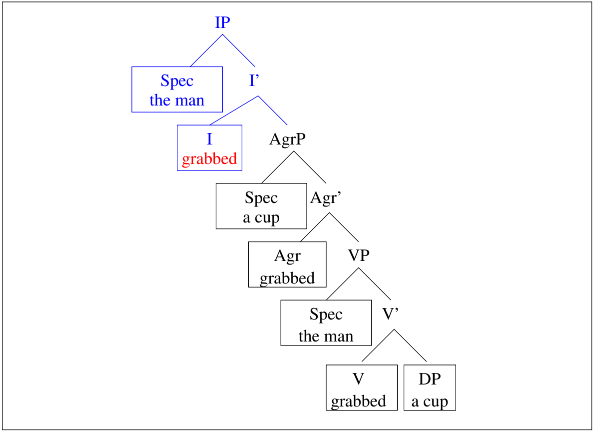
The basic idea

I'm thinking of the LF representation as a sequence of contexts, with two SM representations evoked in each context.



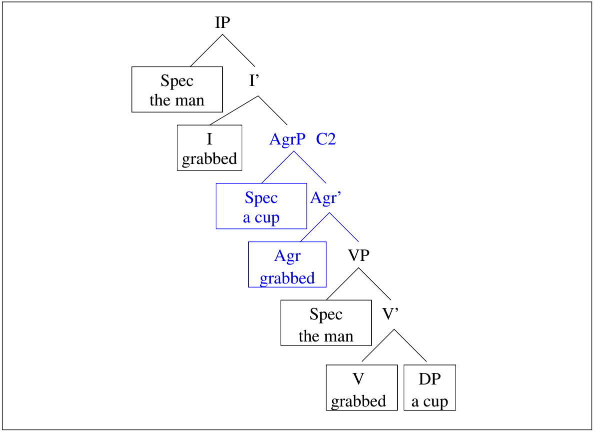
The basic idea

I'm thinking of the LF representation as a sequence of contexts, with two SM representations evoked in each context.



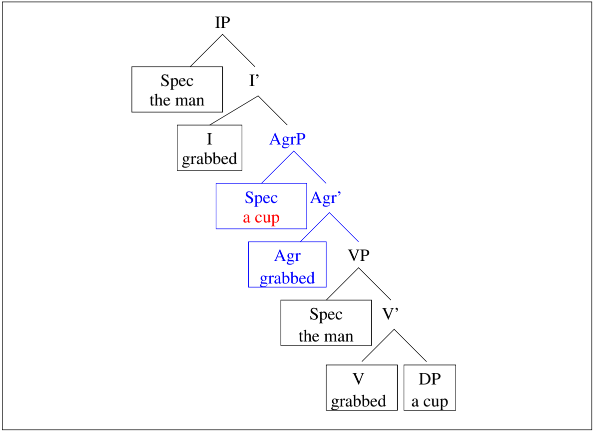
The basic idea

I'm thinking of the LF representation as a sequence of contexts, with two SM representations evoked in each context.



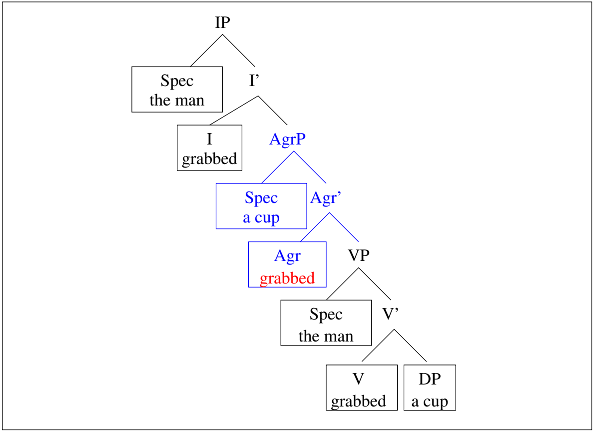
The basic idea

I'm thinking of the LF representation as a sequence of contexts, with two SM representations evoked in each context.



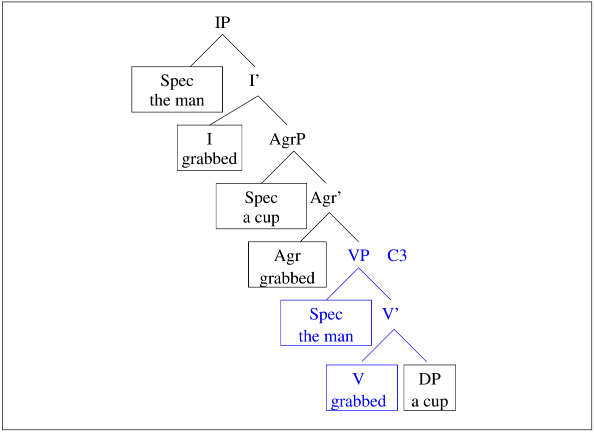
The basic idea

I'm thinking of the LF representation as a sequence of contexts, with two SM representations evoked in each context.



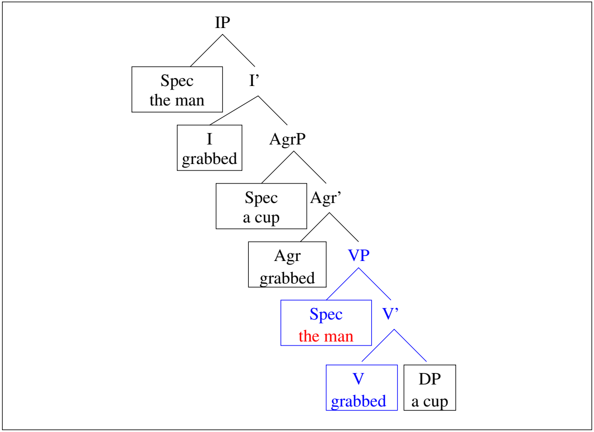
The basic idea

I'm thinking of the LF representation as a sequence of contexts, with two SM representations evoked in each context.



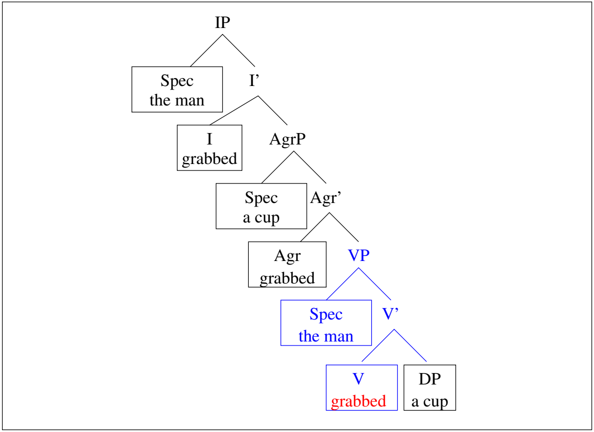
The basic idea

I'm thinking of the LF representation as a sequence of contexts, with two SM representations evoked in each context.



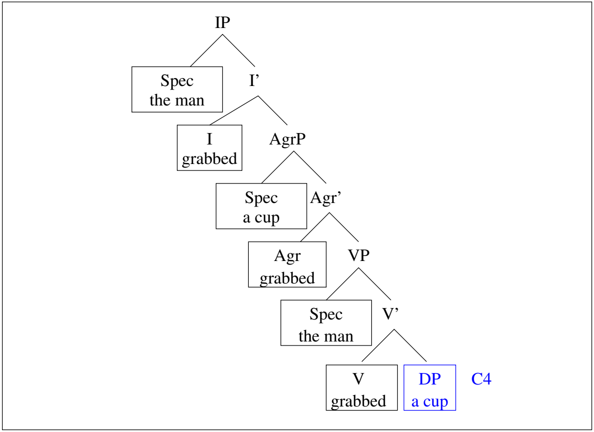
The basic idea

I'm thinking of the LF representation as a sequence of contexts, with two SM representations evoked in each context.



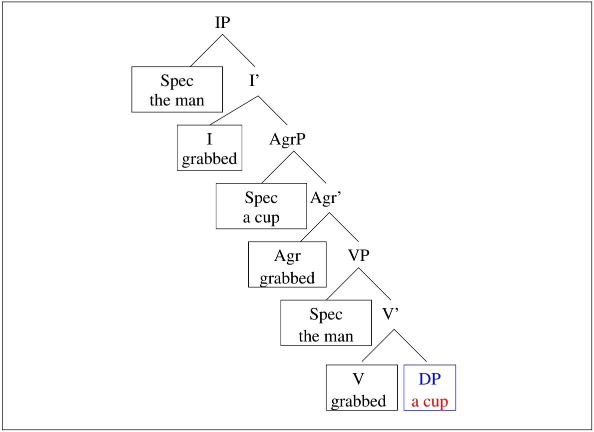
The basic idea

I'm thinking of the LF representation as a sequence of contexts, with two SM representations evoked in each context.



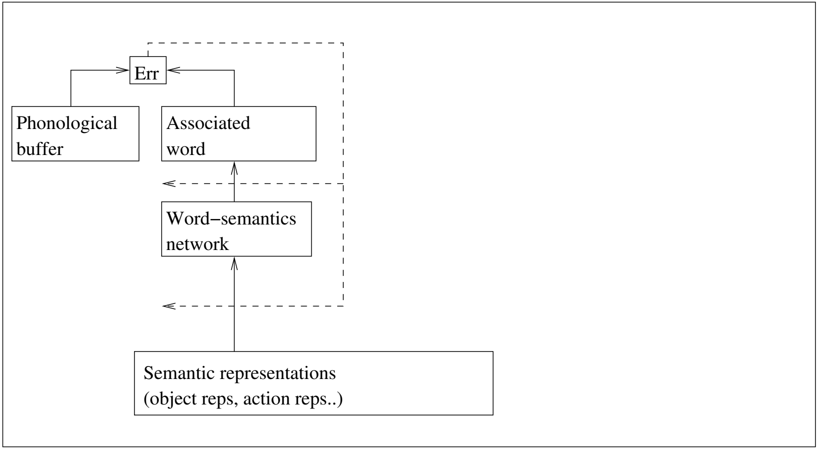
The basic idea

I'm thinking of the LF representation as a sequence of contexts, with two SM representations evoked in each context.



A network for syntactic processing

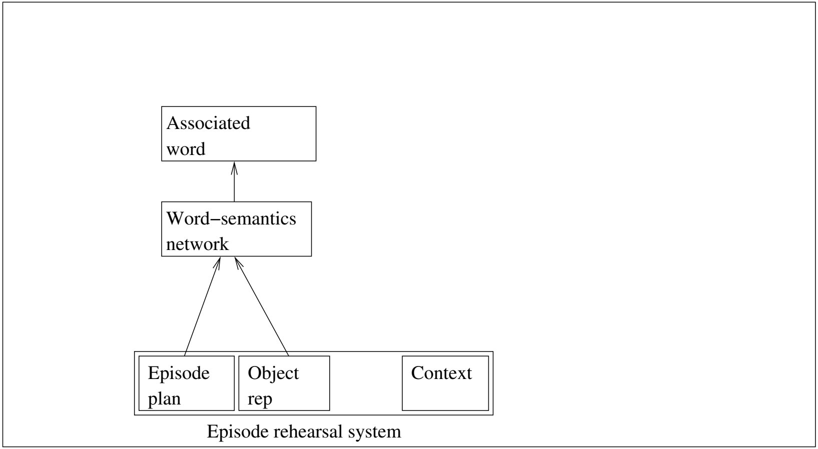
Here's the network for learning word meanings from last lecture.



A network for syntactic processing

We can be more precise about where the semantic reps come from:

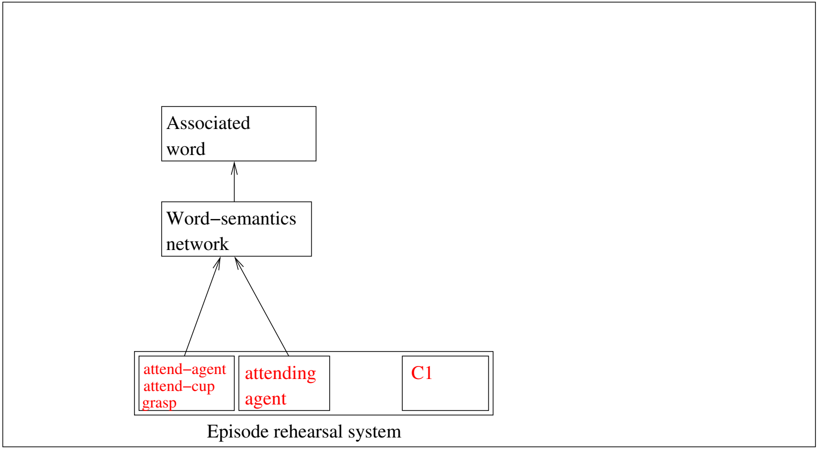
- They're generated when WM episodes are *rehearsed*.



A network for syntactic processing

We can be more precise about where the semantic reps come from:

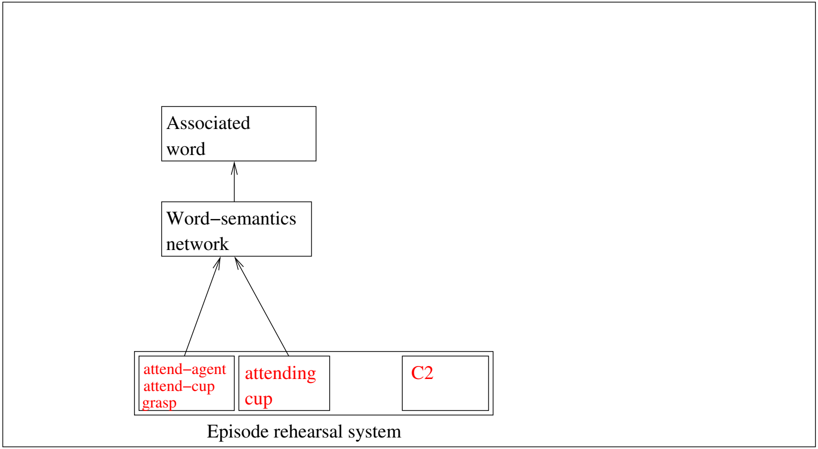
- Rehearsal evokes a sustained episode plan, and transient object reps. (With an updating SM context.)



A network for syntactic processing

We can be more precise about where the semantic reps come from:

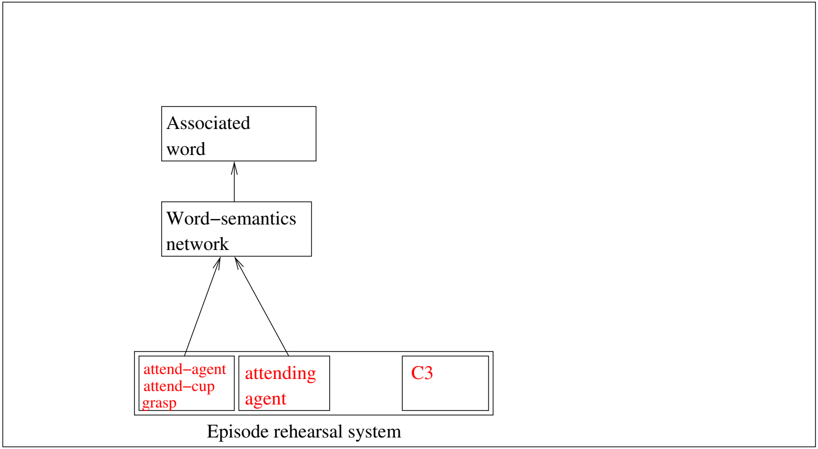
- Rehearsal evokes a sustained episode plan, and transient object reps. (With an updating SM context.)



A network for syntactic processing

We can be more precise about where the semantic reps come from:

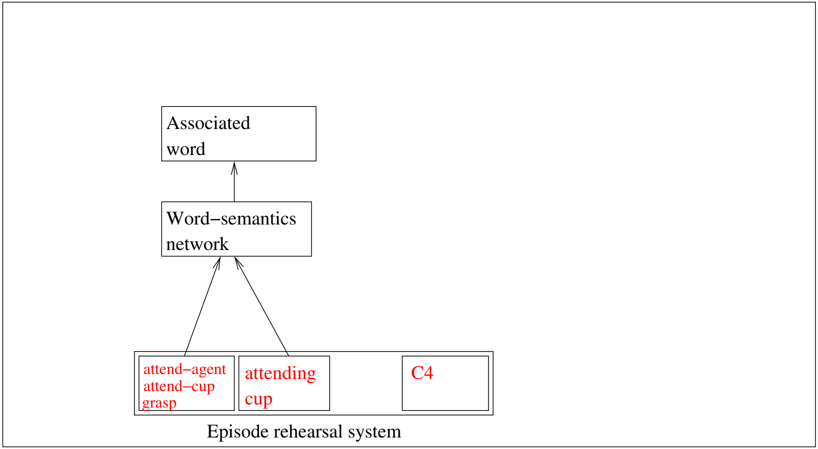
- Rehearsal evokes a sustained episode plan, and transient object reps. (With an updating SM context.)



A network for syntactic processing

We can be more precise about where the semantic reps come from:

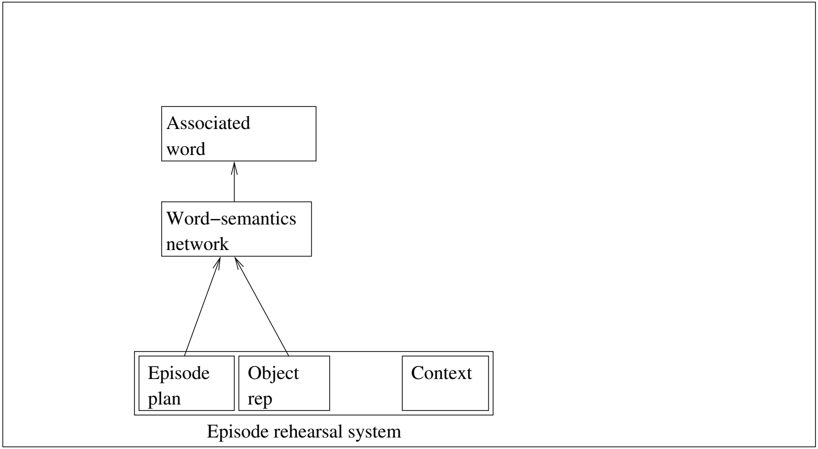
- Rehearsal evokes a sustained episode plan, and transient object reps. (With an updating SM context.)



A network for syntactic processing

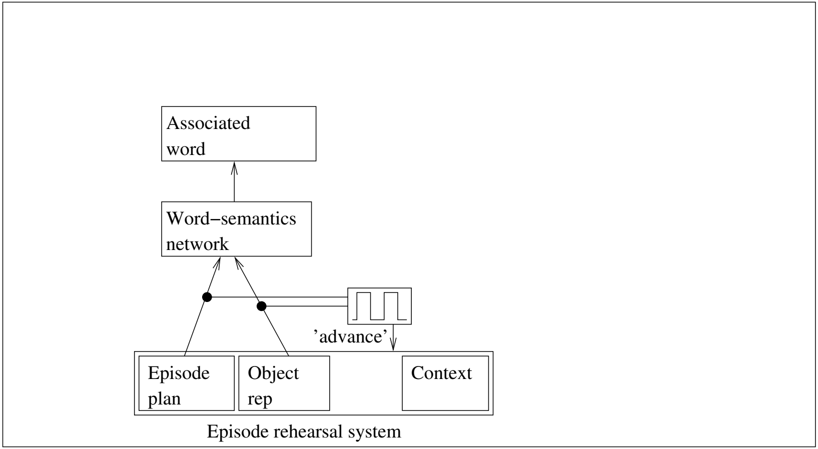
Currently the word-semantics network generates *pairs* of words.

- It really needs to deliver words one at a time.
  Idea: a pattern generator can alternate between semantic signals.



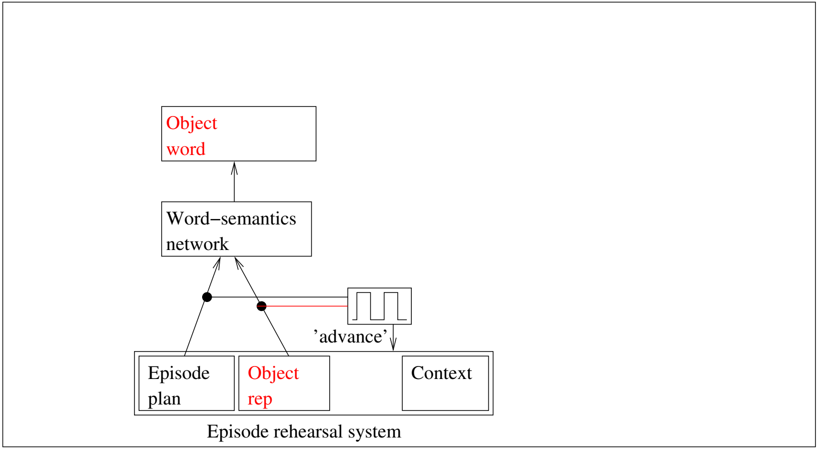
A network for syntactic processing

The pattern generator alternately sends the object representation and the episode plan to the word-sem network.



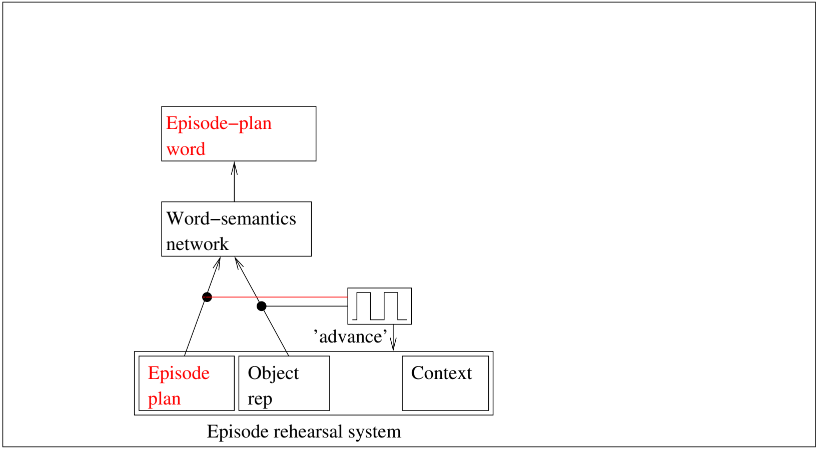
A network for syntactic processing

The pattern generator alternately sends the object representation and the episode plan to the word-sem network.



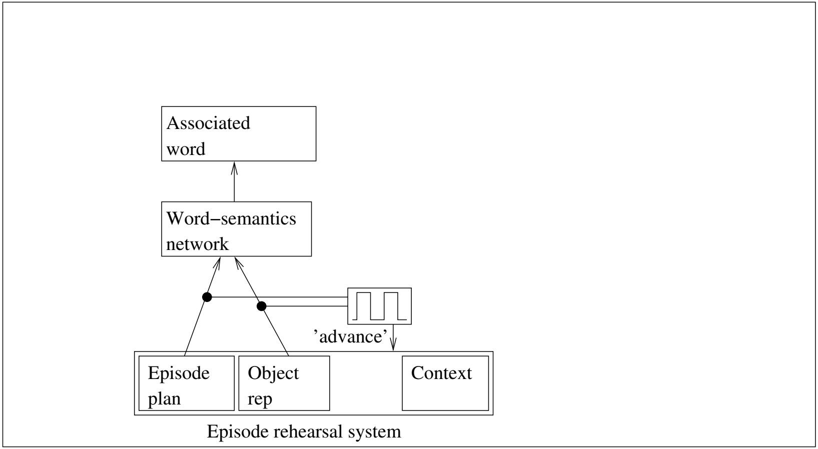
A network for syntactic processing

The pattern generator alternately sends the object representation and the episode plan to the word-sem network.



A network for syntactic processing

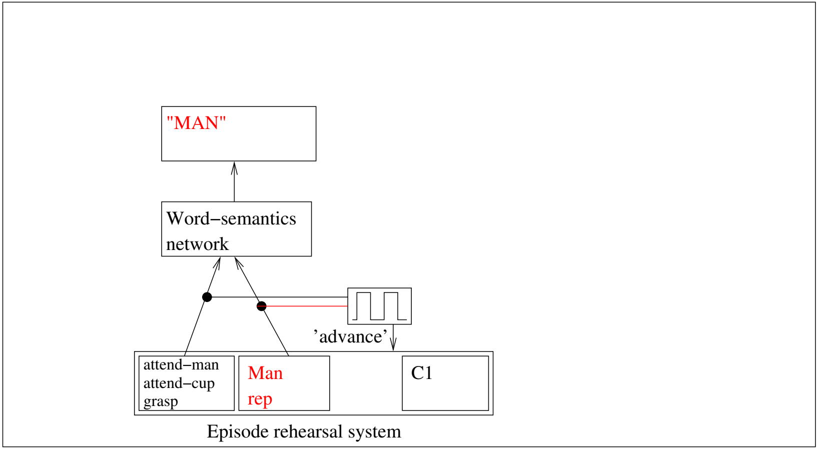
Assume that the pattern generator also ‘advances’ a rehearsed SM sequence (at the end of every cycle).



A network for syntactic processing

Assume that the pattern generator also ‘advances’ a rehearsed SM sequence (at the end of every cycle).

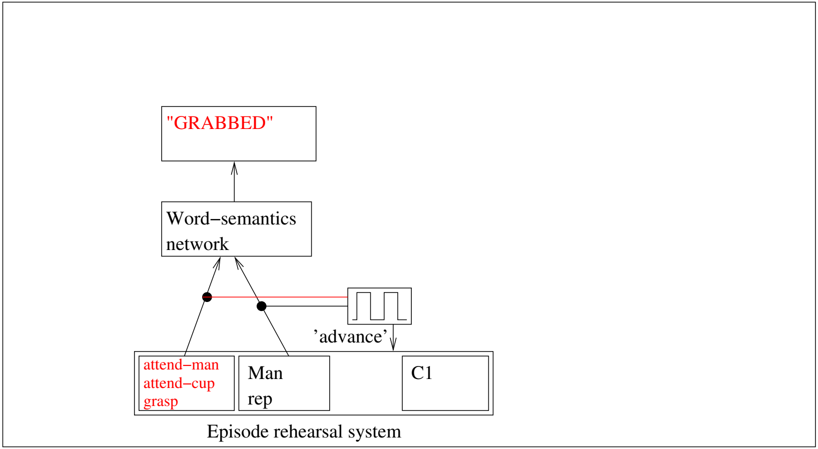
- Now we get a unique word at each time step.



A network for syntactic processing

Assume that the pattern generator also ‘advances’ a rehearsed SM sequence (at the end of every cycle).

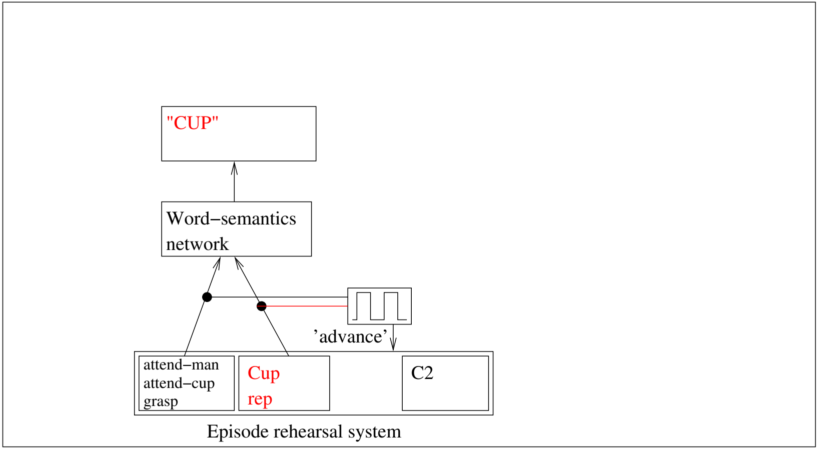
- Now we get a unique word at each time step.



A network for syntactic processing

Assume that the pattern generator also 'advances' a rehearsed SM sequence (at the end of every cycle).

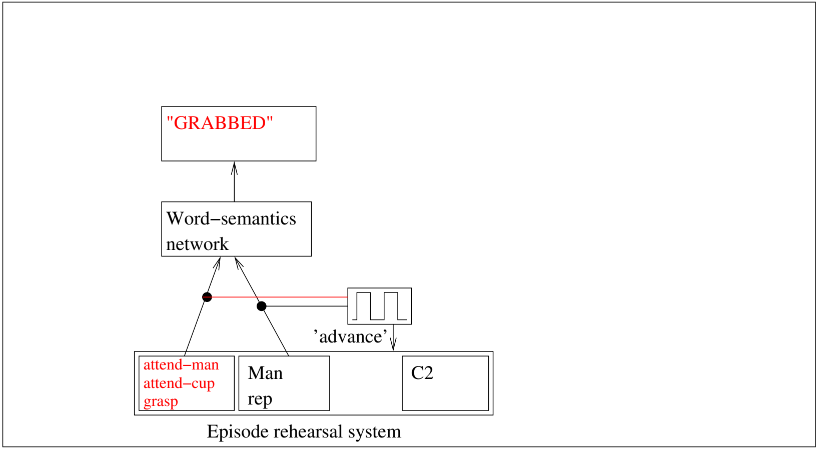
- Now we get a unique word at each time step.



A network for syntactic processing

Assume that the pattern generator also ‘advances’ a rehearsed SM sequence (at the end of every cycle).

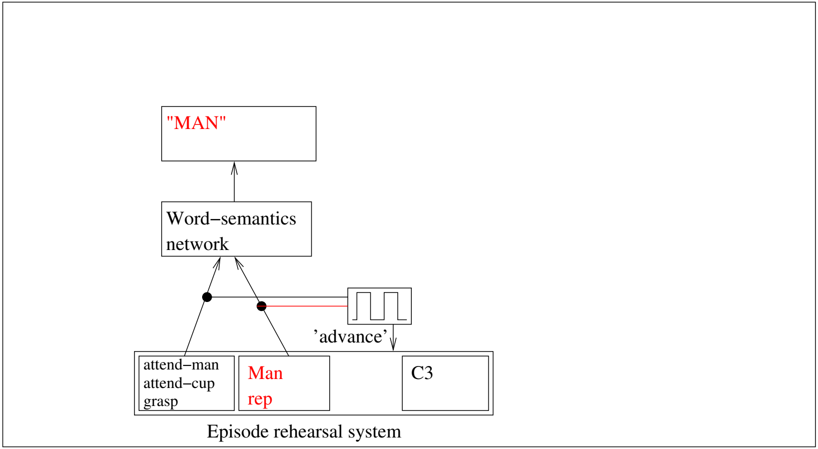
- Now we get a unique word at each time step.



A network for syntactic processing

Assume that the pattern generator also ‘advances’ a rehearsed SM sequence (at the end of every cycle).

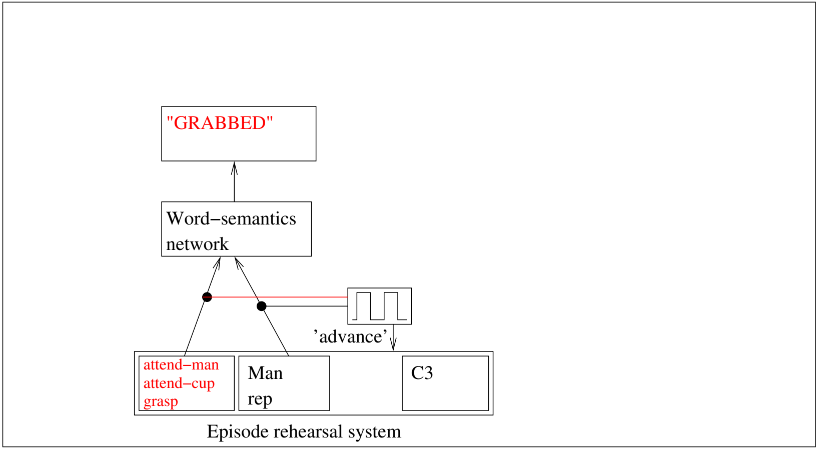
- Now we get a unique word at each time step.



A network for syntactic processing

Assume that the pattern generator also ‘advances’ a rehearsed SM sequence (at the end of every cycle).

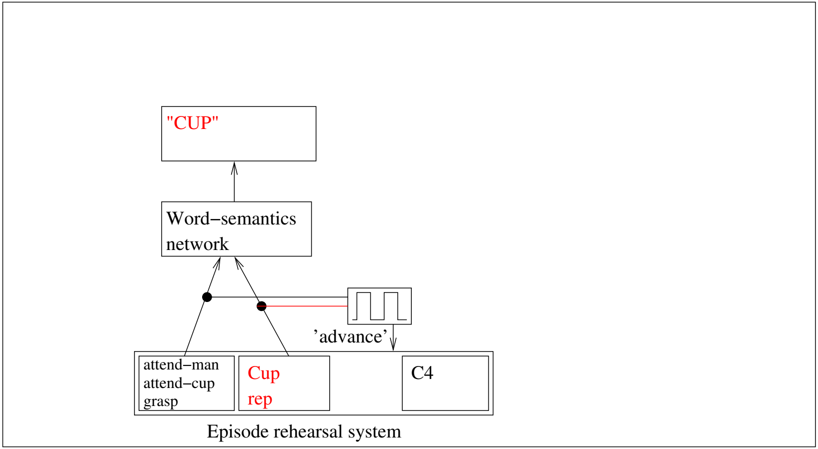
- Now we get a unique word at each time step.



A network for syntactic processing

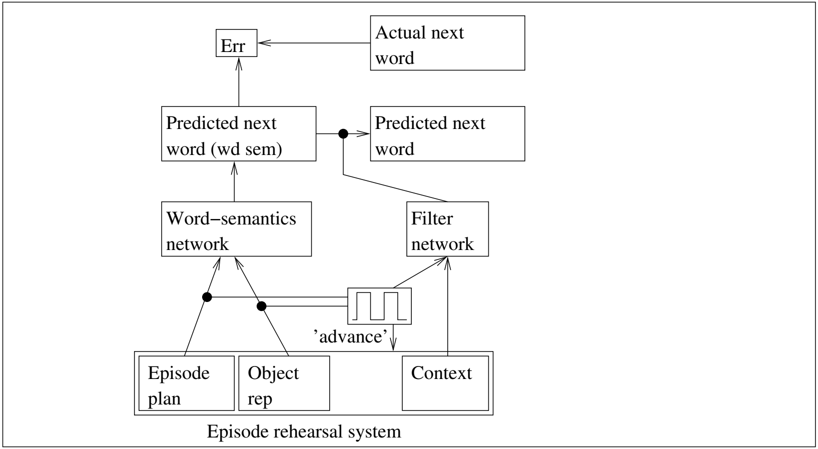
Assume that the pattern generator also ‘advances’ a rehearsed SM sequence (at the end of every cycle).

- Now we get a unique word at each time step.



A network for syntactic processing

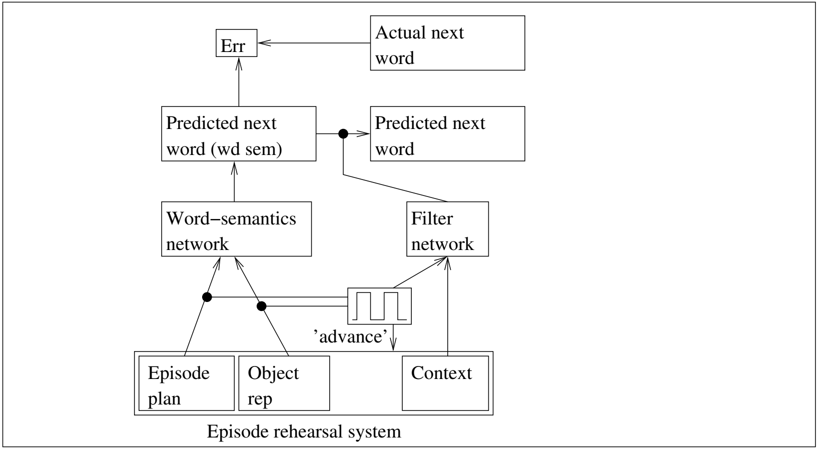
Now we can envisage a filter network which learns when to 'produce' words generated by the word-semantics network.

- 

A network for syntactic processing

Now we can envisage a filter network which learns when to ‘produce’ words generated by the word-semantics network.

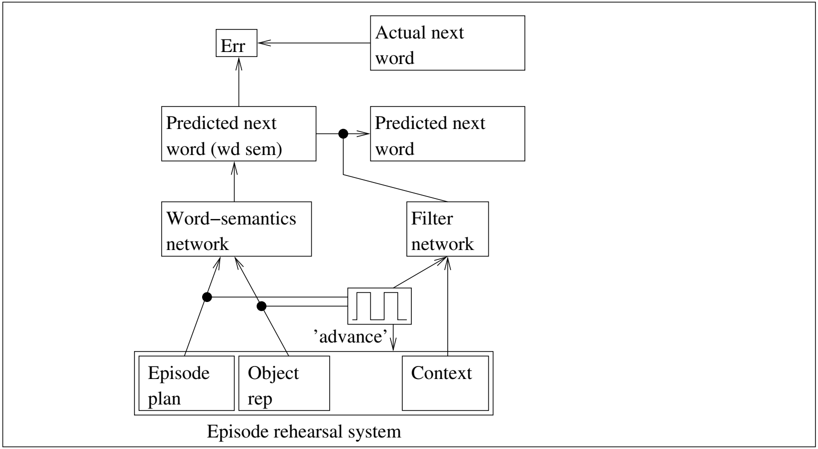
- The network is trained in joint attention situations.



A network for syntactic processing

Now we can envisage a filter network which learns when to ‘produce’ words generated by the word-semantics network.

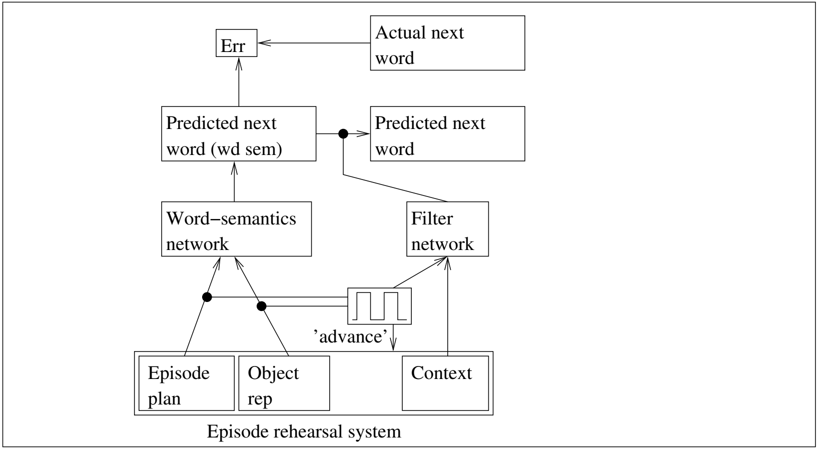
- The agent observes an episode, and hears an utterance.



A network for syntactic processing

Now we can envisage a filter network which learns when to 'produce' words generated by the word-semantics network.

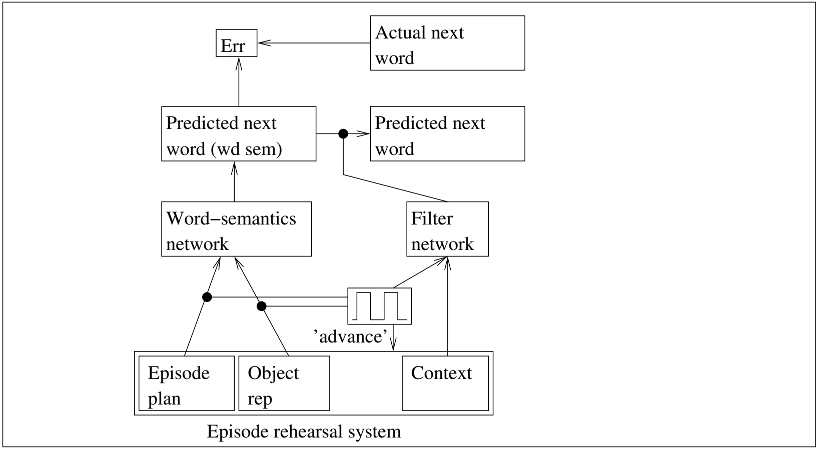
- The agent *replays* the episode and the utterance in synch.



A network for syntactic processing

Now we can envisage a filter network which learns when to 'produce' words generated by the word-semantics network.

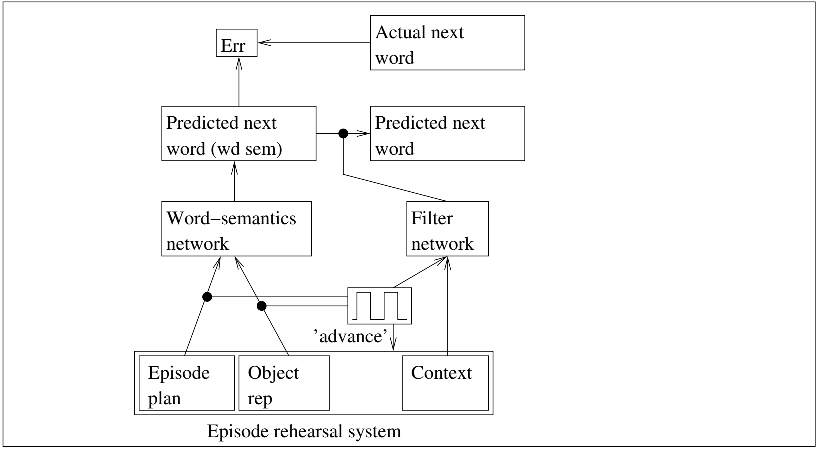
- The network is trained to predict the next word in the utterance.



A network for syntactic processing

Now we can envisage a filter network which learns when to 'produce' words generated by the word-semantics network.

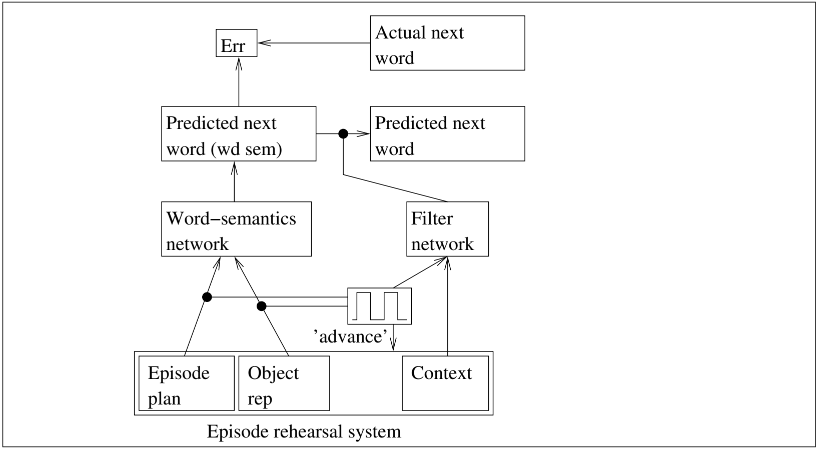
- Its input is the context (C1...C4) and the oscillator.



A network for syntactic processing

Now we can envisage a filter network which learns when to ‘produce’ words generated by the word-semantics network.

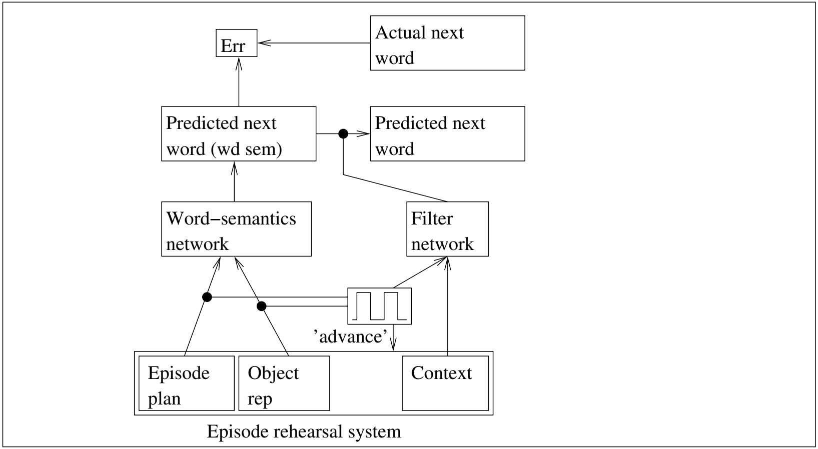
- If word-sem network output matches the next word, it’s passed on.



A network for syntactic processing

Now we can envisage a filter network which learns when to ‘produce’ words generated by the word-semantics network.

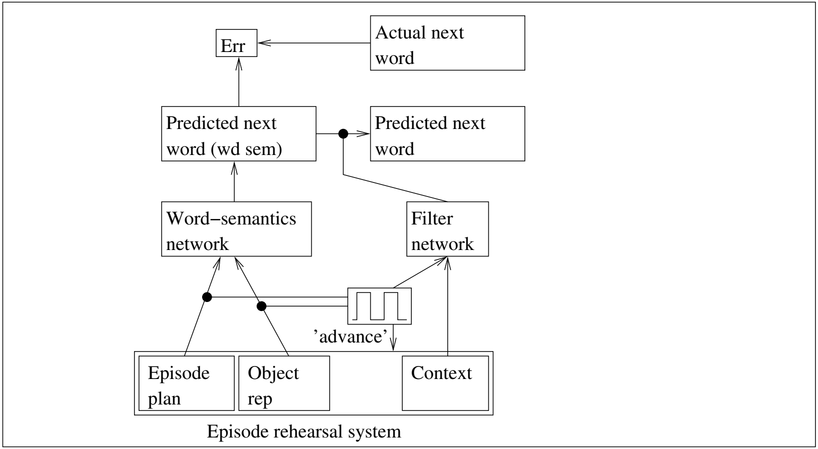
- If not, it’s blocked.



A network for syntactic processing

Now we can envisage a filter network which learns when to 'produce' words generated by the word-semantics network.

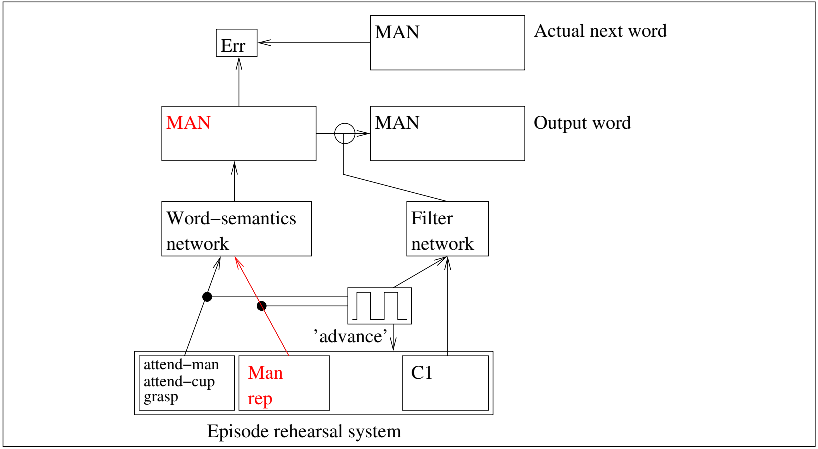
- Example: training in English (*man, grabbed, cup*).



A network for syntactic processing

Now we can envisage a filter network which learns when to ‘produce’ words generated by the word-semantics network.

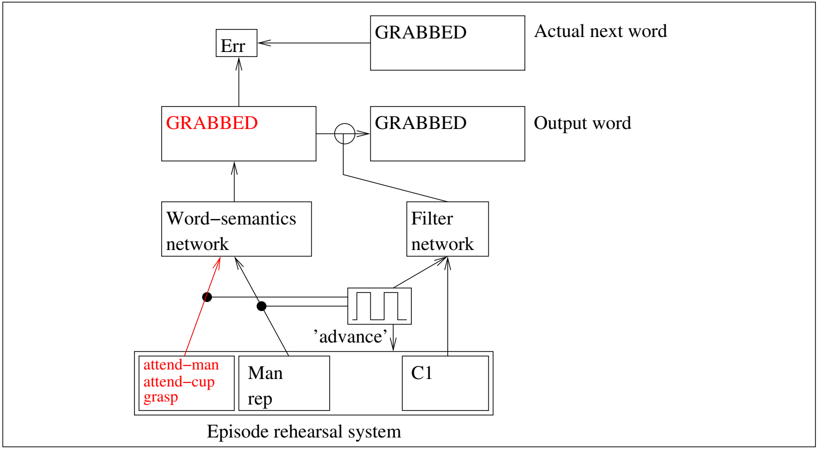
- Output matches: train C1/1 → PASS.



A network for syntactic processing

Now we can envisage a filter network which learns when to ‘produce’ words generated by the word-semantics network.

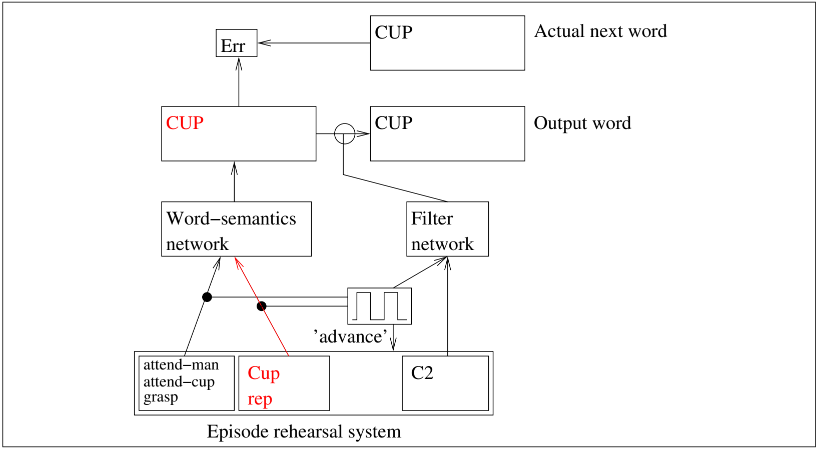
- Output matches: train C1/2 → PASS.



A network for syntactic processing

Now we can envisage a filter network which learns when to ‘produce’ words generated by the word-semantics network.

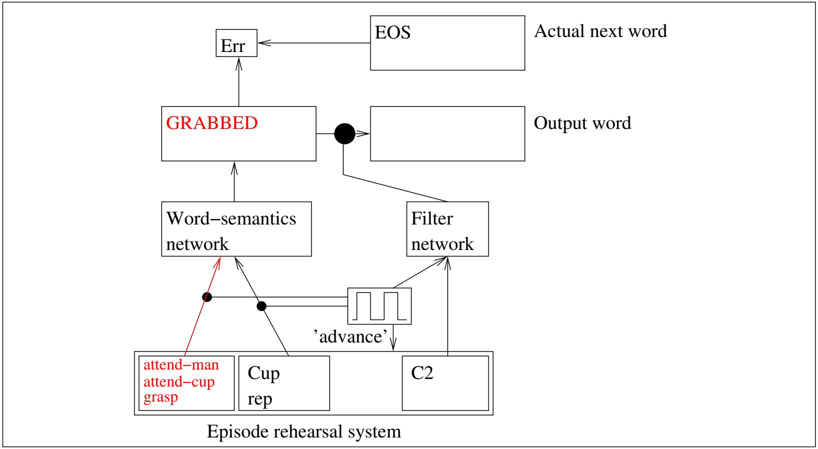
- Output matches: train C2/1 → PASS.



A network for syntactic processing

Now we can envisage a filter network which learns when to ‘produce’ words generated by the word-semantics network.

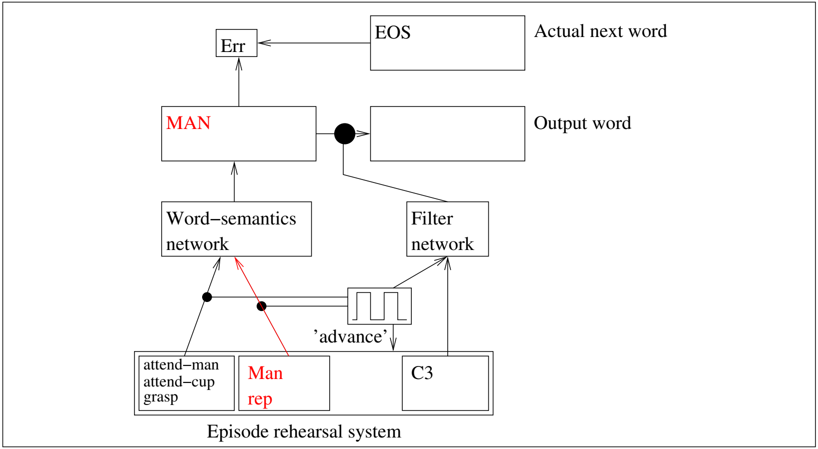
- Output doesn’t match: train C2/2 → BLOCK.



A network for syntactic processing

Now we can envisage a filter network which learns when to ‘produce’ words generated by the word-semantics network.

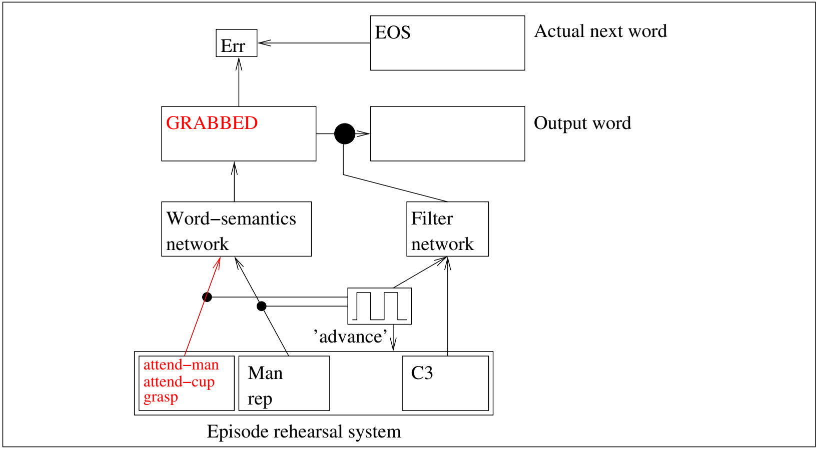
- Output doesn’t match: train C3/1 → BLOCK.



A network for syntactic processing

Now we can envisage a filter network which learns when to 'produce' words generated by the word-semantics network.

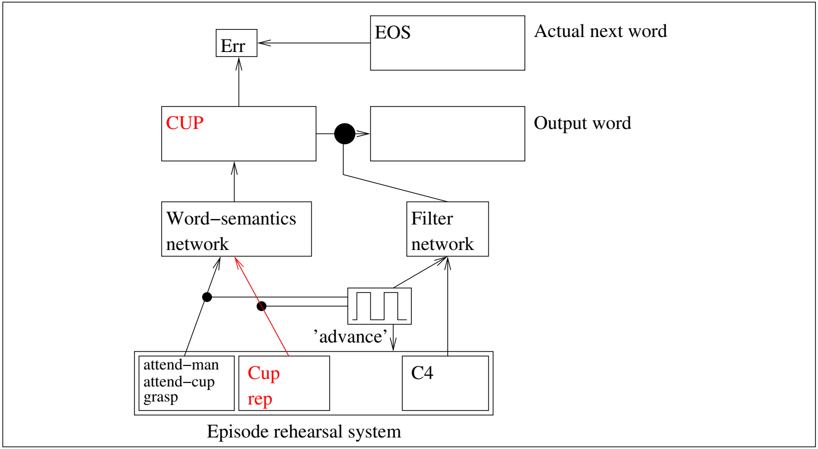
- Output doesn't match: train C3/2 → BLOCK.



A network for syntactic processing

Now we can envisage a filter network which learns when to ‘produce’ words generated by the word-semantics network.

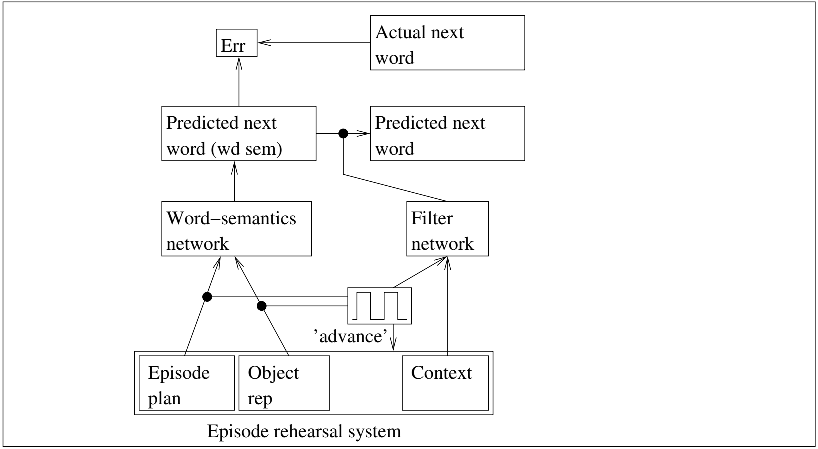
- Output doesn’t match: train C4/1 → BLOCK.



A network for syntactic processing

Now we can envisage a filter network which learns when to ‘produce’ words generated by the word-semantics network.

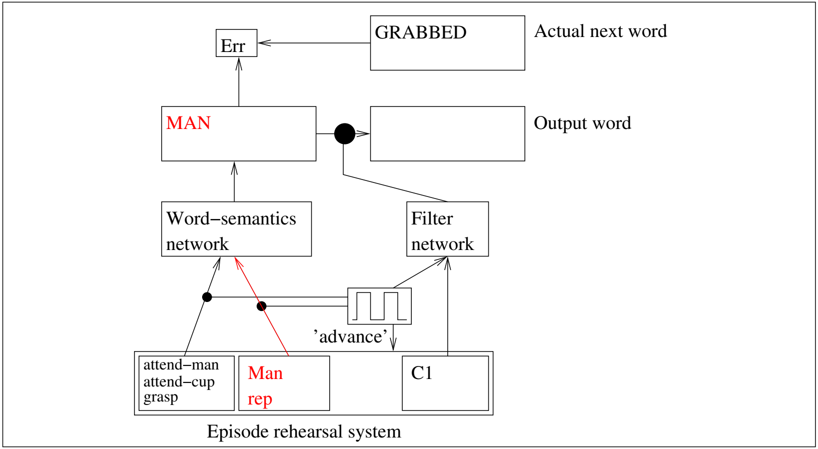
- Example: training in a VSO language (*grabbed, man, cup*).



A network for syntactic processing

Now we can envisage a filter network which learns when to 'produce' words generated by the word-semantics network.

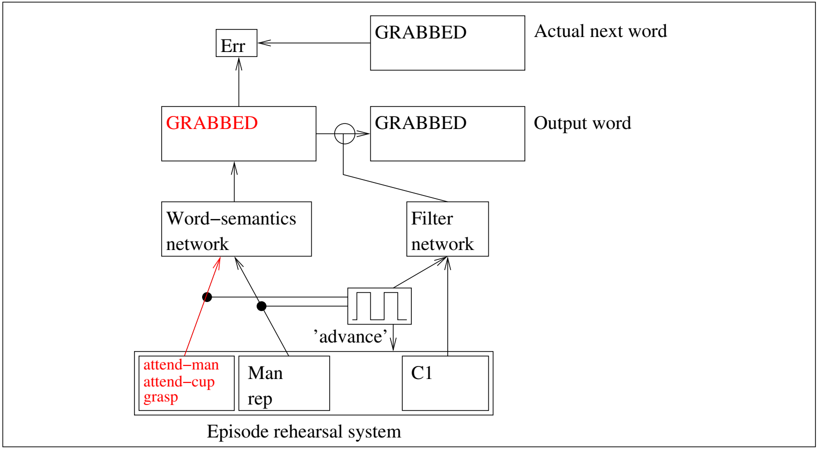
- Output doesn't match: train C1/1 → BLOCK.



A network for syntactic processing

Now we can envisage a filter network which learns when to ‘produce’ words generated by the word-semantics network.

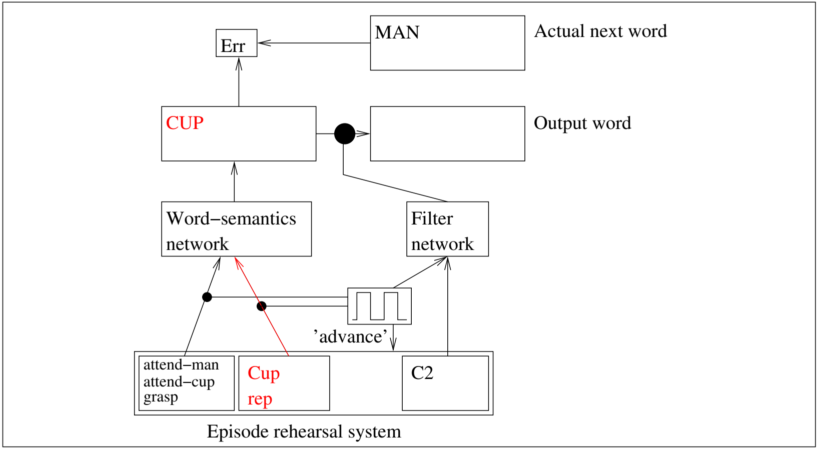
- Output matches: train C1/2 → PASS.



A network for syntactic processing

Now we can envisage a filter network which learns when to 'produce' words generated by the word-semantics network.

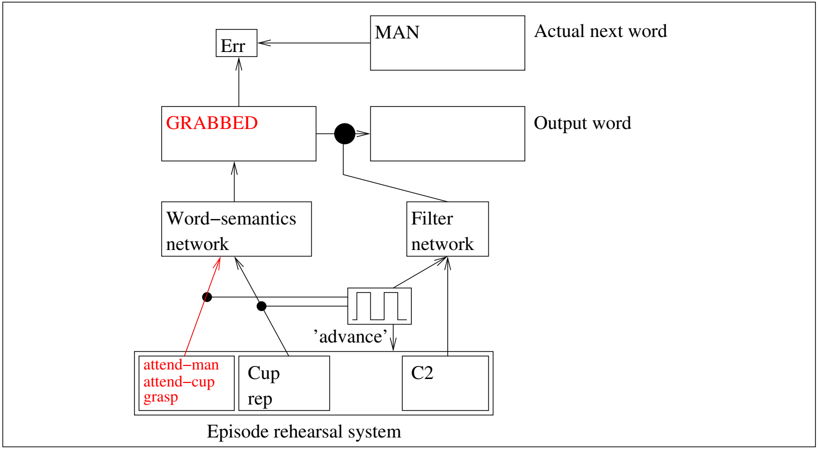
- Output doesn't match: train C2/1 → BLOCK.



A network for syntactic processing

Now we can envisage a filter network which learns when to 'produce' words generated by the word-semantics network.

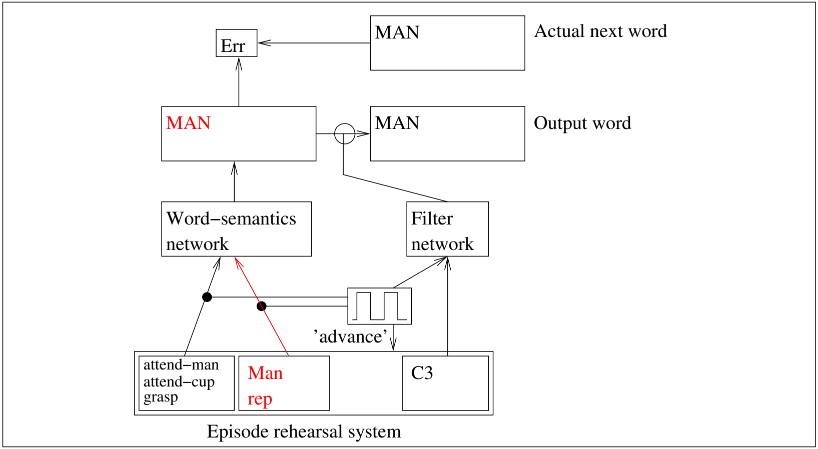
- Output doesn't match: train C2/2 → BLOCK.



A network for syntactic processing

Now we can envisage a filter network which learns when to 'produce' words generated by the word-semantics network.

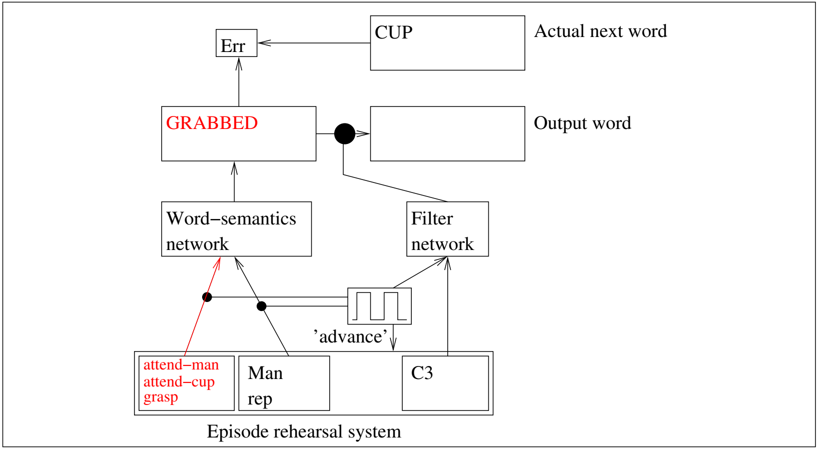
- Output matches: train C3/1 → PASS.



A network for syntactic processing

Now we can envisage a filter network which learns when to ‘produce’ words generated by the word-semantics network.

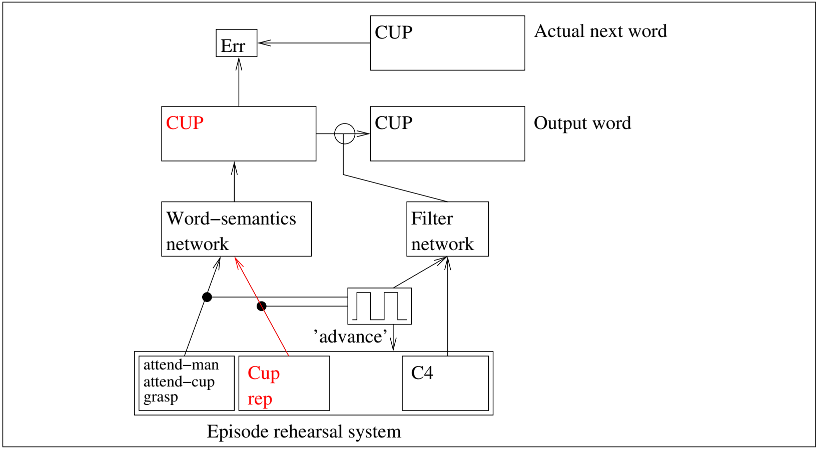
- Output doesn’t match: train C3/2 → BLOCK.



A network for syntactic processing

Now we can envisage a filter network which learns when to ‘produce’ words generated by the word-semantics network.

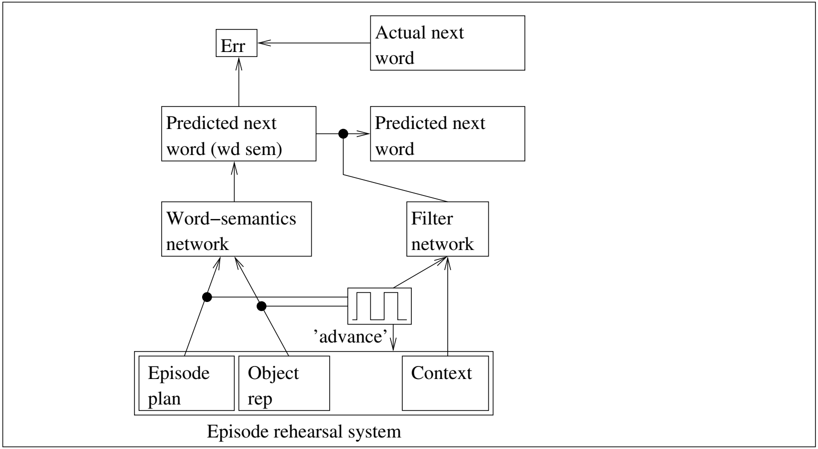
- Output matches: train C4/1 → PASS.



A network for syntactic processing

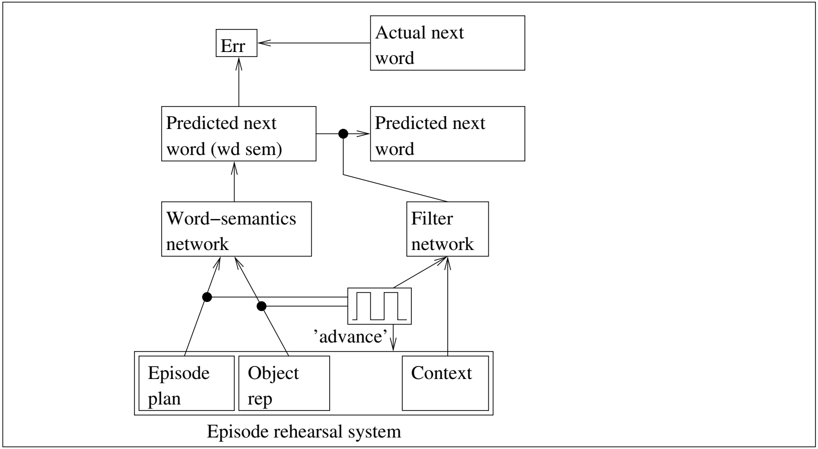
This network generates sentences 'compositionally':

- it uses mappings from semantic reps to words, and a mechanism for rehearsing episode plans.



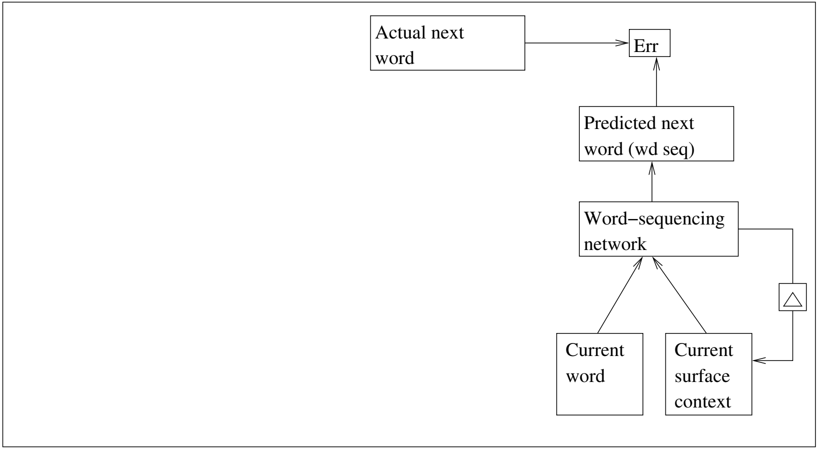
A network for syntactic processing

But remember that there's probably another mechanism for generating word sequences, which doesn't rely on word semantics.



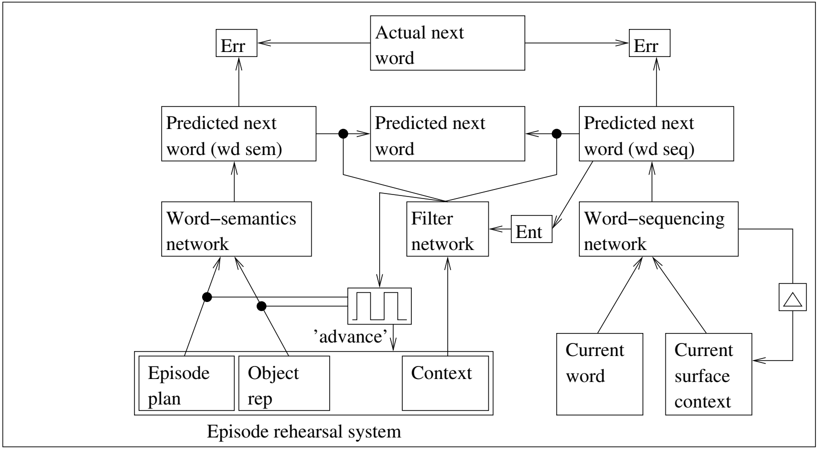
A network for syntactic processing

Proposal: an Elman-style word-sequencing network operates in parallel with the word-semantics/filter network.



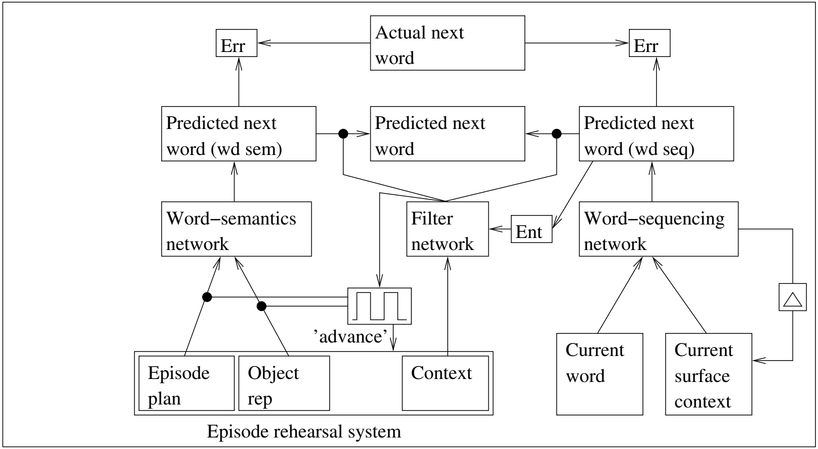
A network for syntactic processing

Both networks generate predictions about the next word in an utterance.



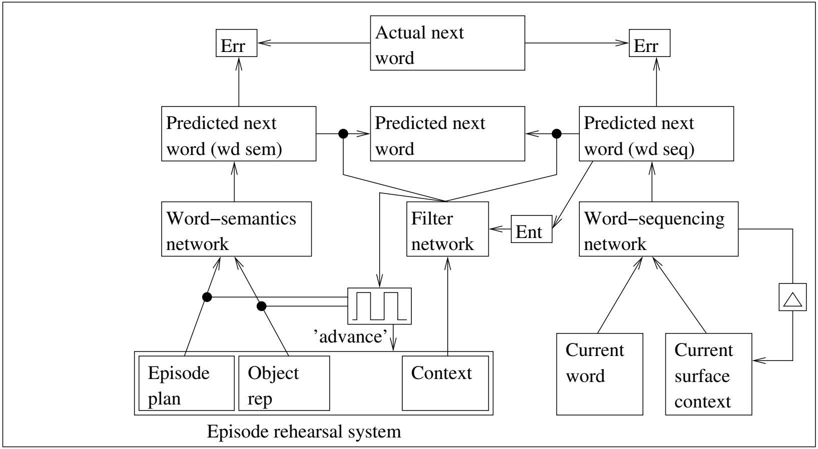
A network for syntactic processing

For ‘compositional’ language, the episode-rehearsal/word-meaning network’s predictions are more accurate.



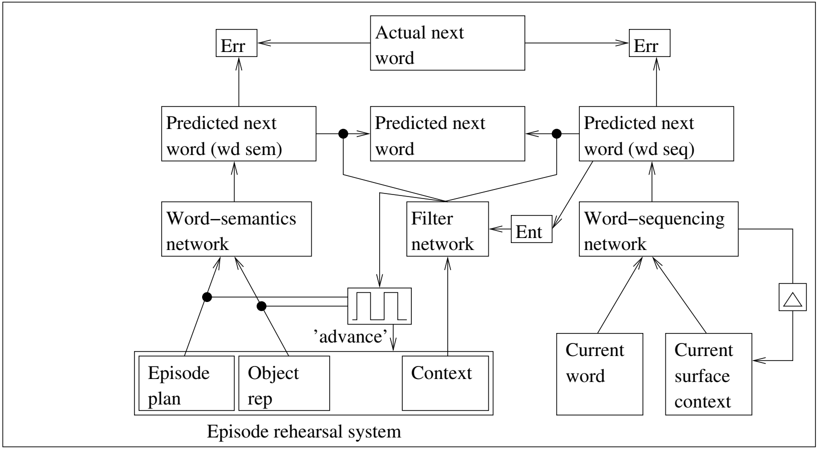
A network for syntactic processing

But for idiomatic language, the word-sequencing network's predictions are more accurate.



A network for syntactic processing

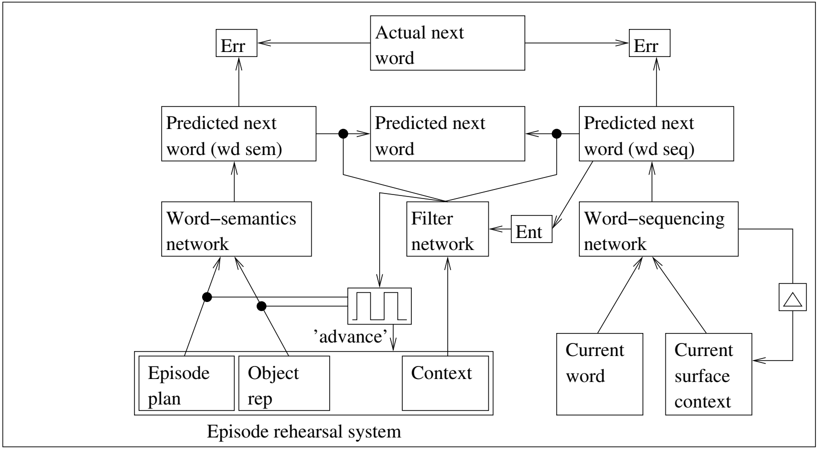
The filter network has to learn when to rely on which predictor.



A network for syntactic processing

It can use the entropy of the word-sequencing network to decide this.

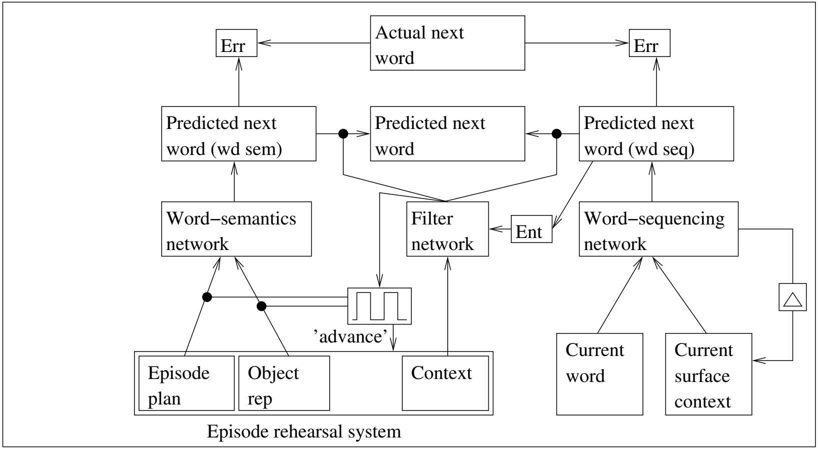
- Entropy is a measure of the confidence of the word-sequencing network in its prediction.



A network for syntactic processing

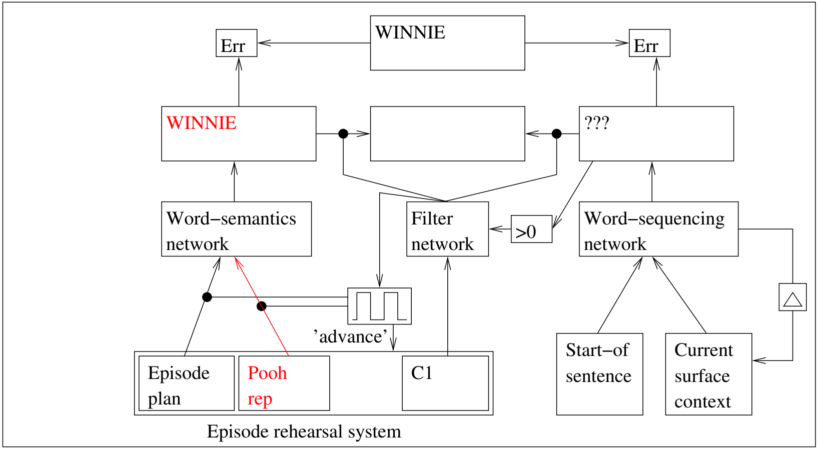
Here's an example of a sentence containing an idiom:

- *Winnie the pooh* *grabbed cup*



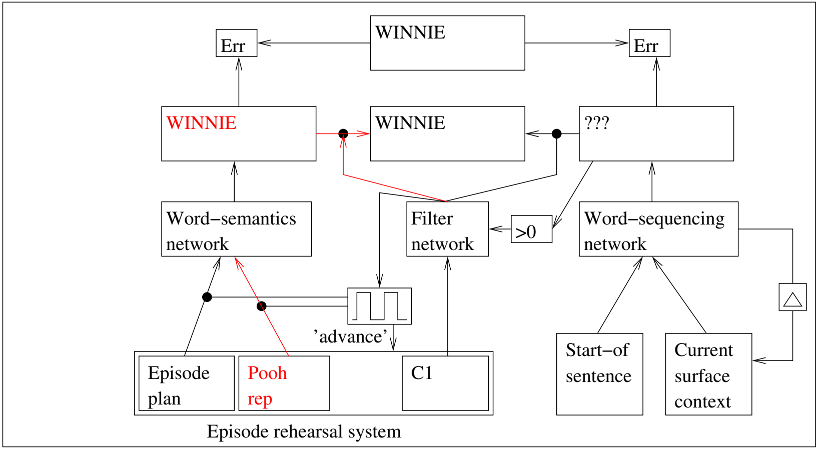
A network for syntactic processing

For the first word, the compositional network is more accurate.



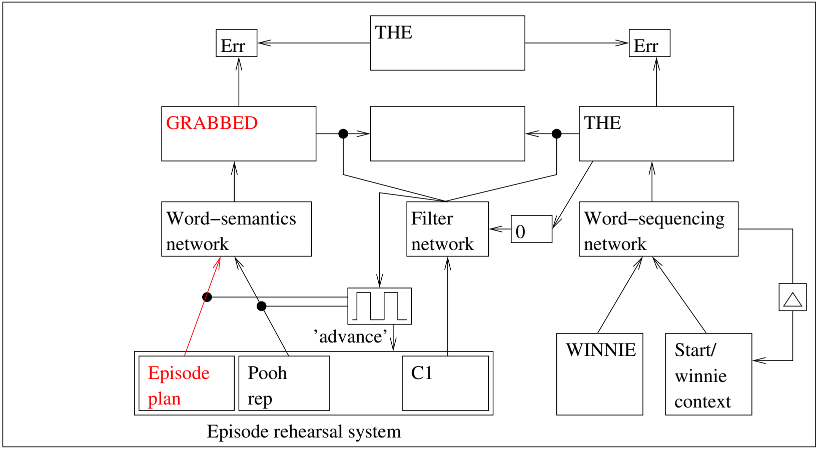
A network for syntactic processing

For the first word, the compositional network is more accurate.



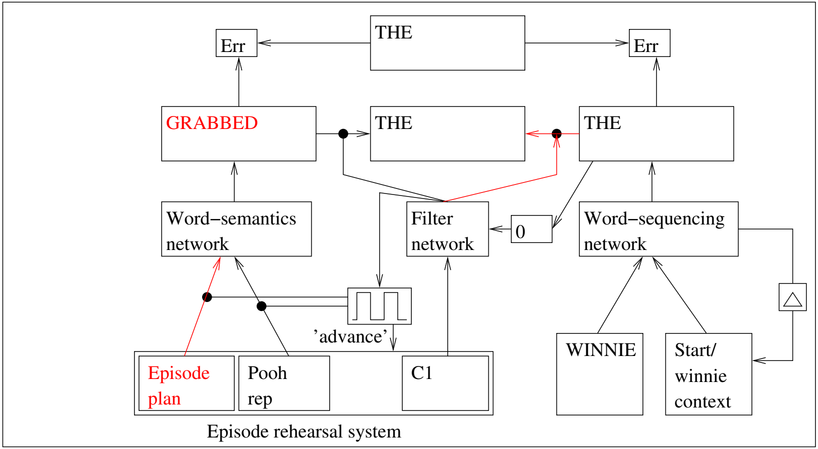
A network for syntactic processing

For the second word (part of an idiom), the word-sequencing network is more accurate. So its output is chosen.



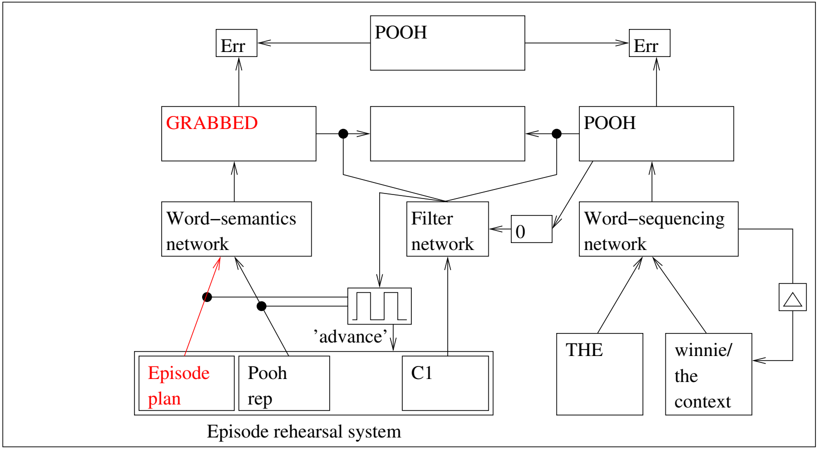
A network for syntactic processing

For the second word (part of an idiom), the word-sequencing network is more accurate. So its output is chosen.



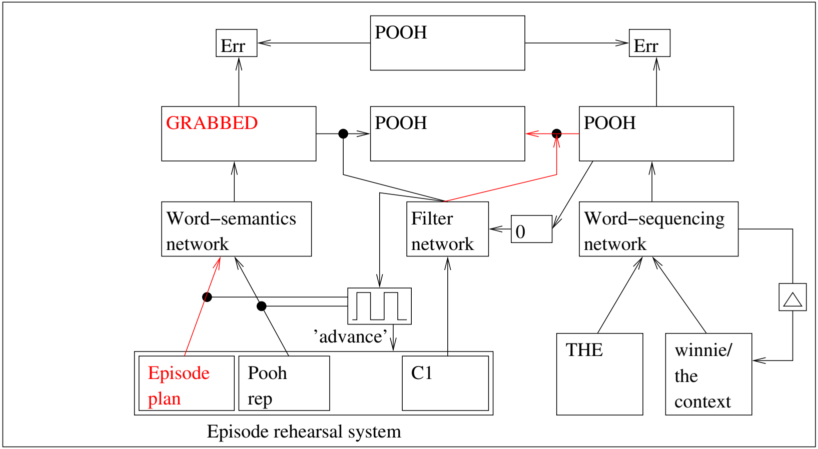
A network for syntactic processing

Likewise for the third word (still part of the idiom).



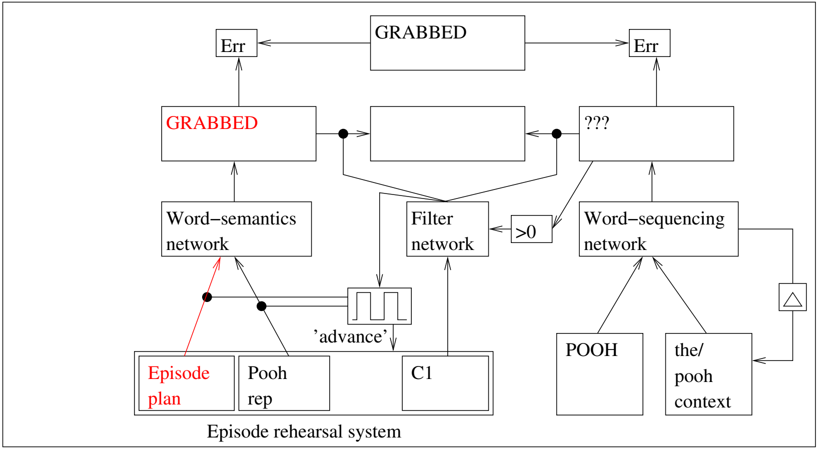
A network for syntactic processing

Likewise for the third word (still part of the idiom).



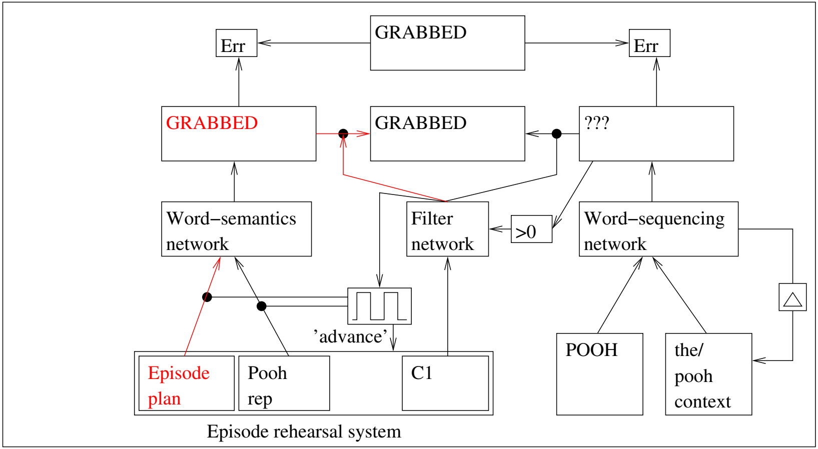
A network for syntactic processing

For the fourth word (no longer part of the idiom), the compositional network is again more accurate.



A network for syntactic processing

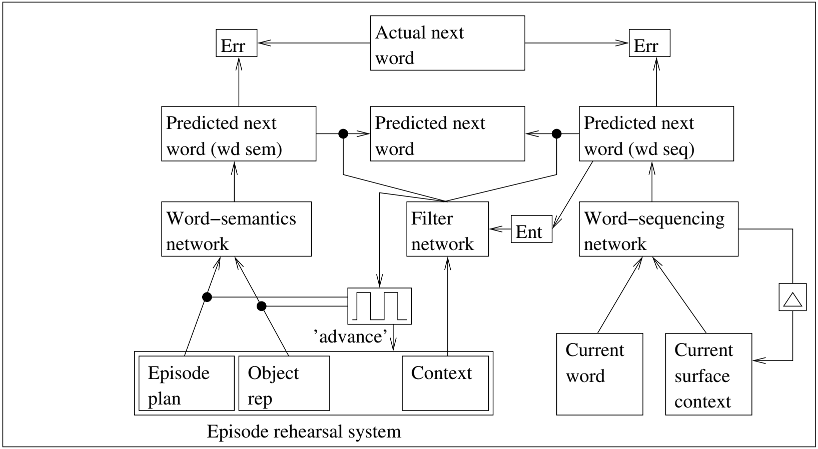
For the fourth word (no longer part of the idiom), the compositional network is again more accurate.



A network for syntactic processing

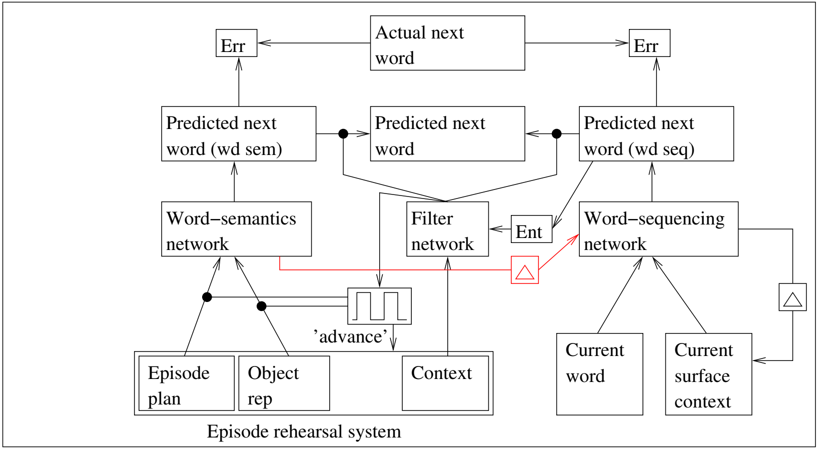
Discontinuous idioms are a bit more tricky.

- E.g. *John takes Bill to task.*



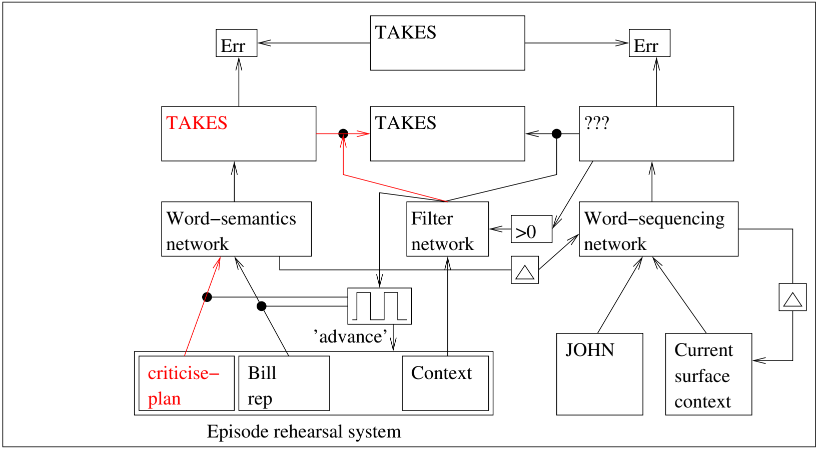
A network for syntactic processing

One way of dealing with these is to allow word semantics to contribute to the context representation of the word-sequencing network.



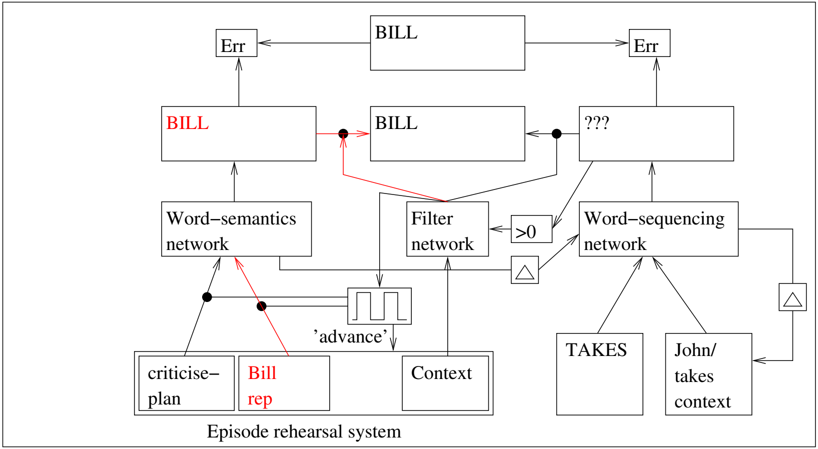
A network for syntactic processing

The first word of the idiom is produced compositionally.



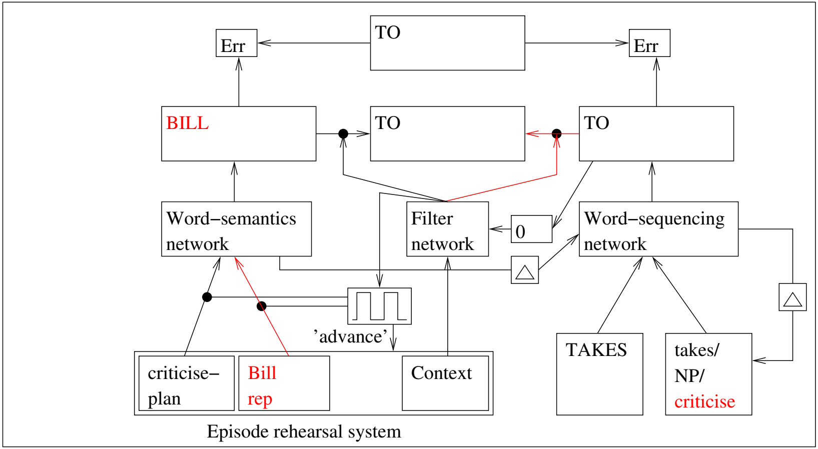
A network for syntactic processing

Then another compositional expression is produced.
And the word-sequencing network's context is updated.



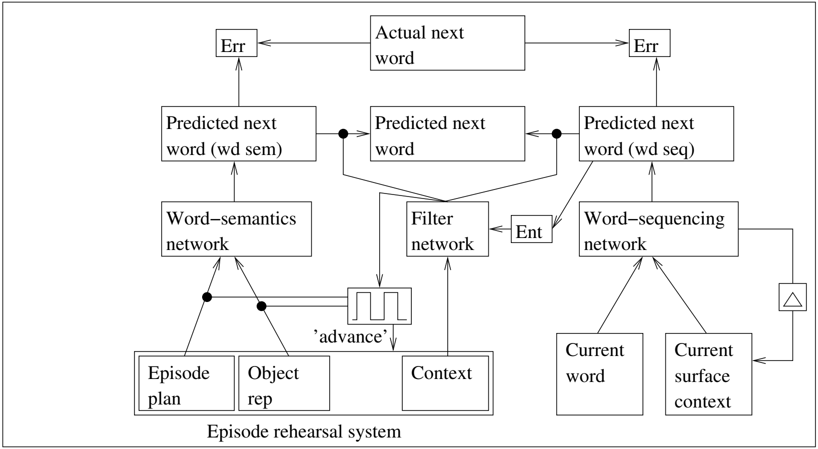
A network for syntactic processing

For the second part of the idiom, the word-sequencing network is again confident.



A network for syntactic processing

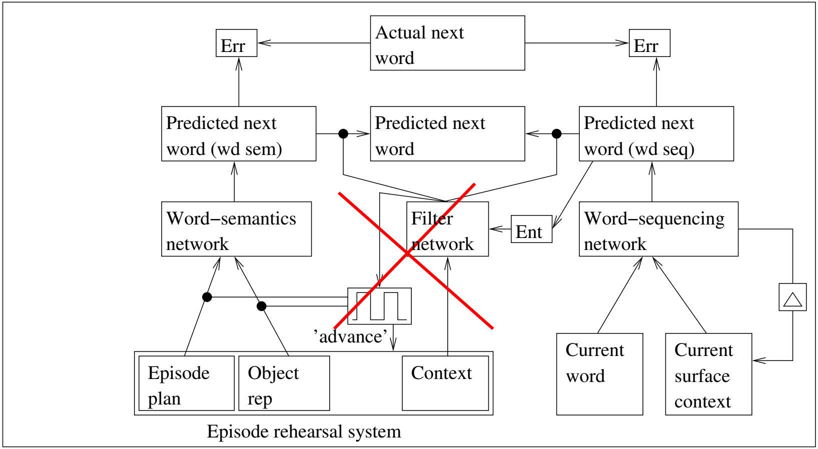
How does this network map onto the brain?



A network for syntactic processing

The filter network is part of Broca's area.

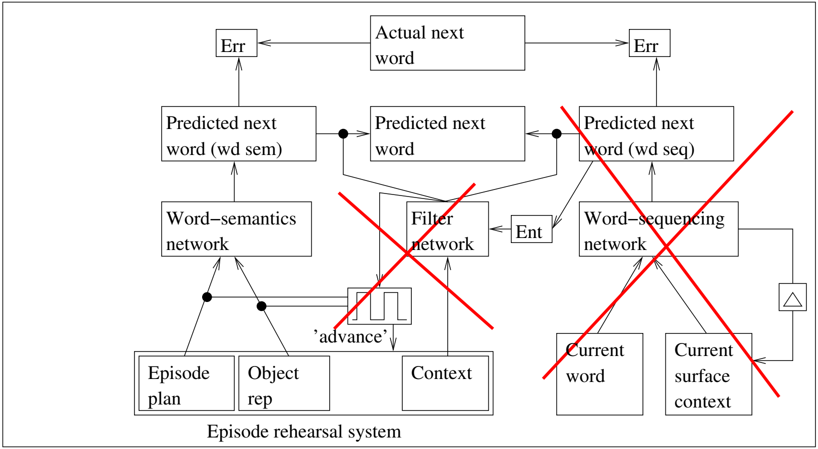
- Language is about learning control/gating strategies. (C.f. Earl Miller's model of PFC.)



A network for syntactic processing

The word-sequencing network is also part of Broca's area.

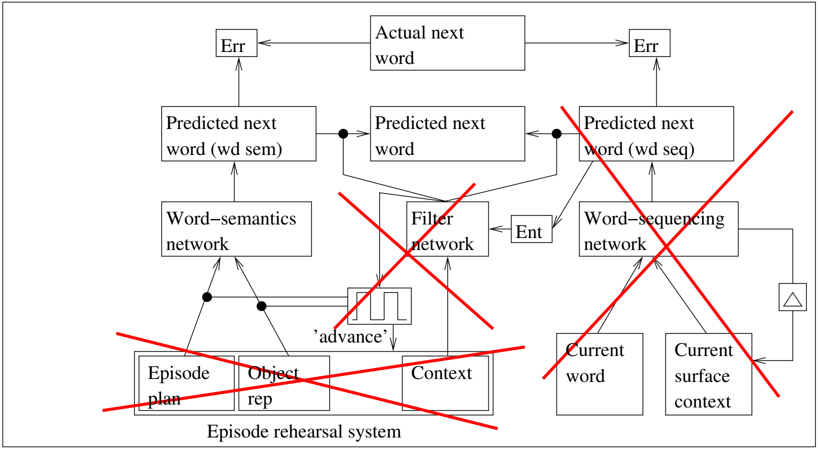
- Language is about sequencing abilities.
  (C.f. Elman and many others).



A network for syntactic processing

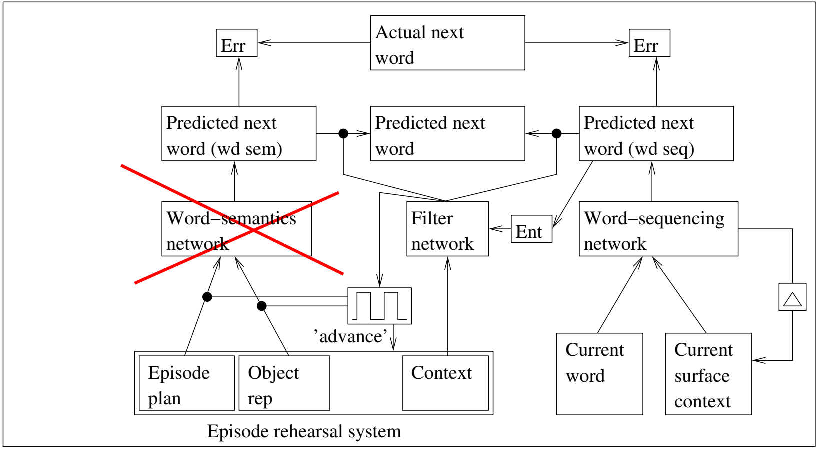
The episode-rehearsal system is also part of Broca's area.

- Language is about working memory episode representations. (C.f. Baddeley and many others).



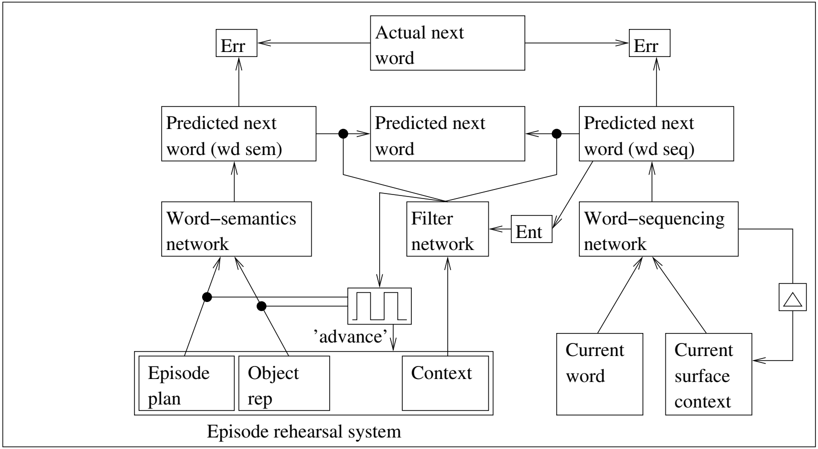
A network for syntactic processing

The word-semantics network is part of 'Wernicke's area' (and associated temporal regions).



A network for syntactic processing

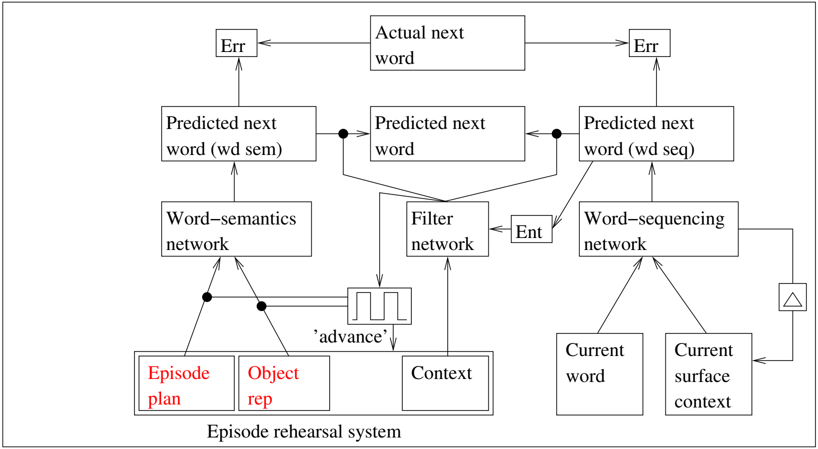
What can the network say about the developmental stages of syntax in childhood?



A network for syntactic processing

The single-word / holophrase stage:

- Children represent their *own intentions* in the 'episode plan' medium. And they attend to what they want…



A network for syntactic processing

The pivot-schemas / item-based constructions stage:

- Perhaps there can be direct connections from semantic representations to the word-sequencing network.