

Modelling Dialogue: Preliminaries for Syntactic Change

Production, ellipsis, and re-use of actions from context

July 31, 2008

Getting towards an account of change

- DS mechanisms: an account of growth of information in context
- What is needed:
 - A formal specification of context
(storing processes for building content)
 - A specification of production and interaction between interlocutors
- Dialogue provides best possible window on speaker/hearer interaction
- Ellipsis provides window on context which speaker/hearer make use of
- Ellipsis and anaphora (in dialogue) equally make use of previously used *content, structure* and *actions*: context thus a complex of all three
- Routinisation of such mechanisms for re-use will provide basis for change
- Routinisation of actions for establishing context-dependence can only take place through pronouns (has to be lexicalised)
- Routinisation as a basis for syntactic change

Lecture 3: The Challenge of Modelling Dialogue

- Background: Challenge by psycholinguists Pickering & Garrod 2004
How well do models of language match core data of conversational dialogue?
This challenge is hard for grammar models which presume articulation of language capacity is independent of use
- Challenge of ellipsis: straddling syntax/semantics/pragmatics
- Shift into grammar formalism with dynamics of processing yields underspecification-plus-enrichment in syntax
 - Unifying ellipsis as context-dependent interpretation
 - Context-dependent incremental generation using parsing architecture
 - A Dynamic Syntax model of dialogue
- Reconstructing Latin conversational dialogue!

The Dialogue Challenge :hard for current grammars

- Cross-speaker parallelism effects:
 - (i) Ruth: What did Alex give Eliot?
Hugh: She gave him a music-box.
- Cross-speaker ellipsis - relying on context to interpret fragment
 - (ii) Ruth: Who does everyone admire?
Hugh: Their mother.
 - (iii) Alex: We're going to Edinburgh.
Hugh: because Mum asked us to.
- speaker/hearer exchange of roles across ALL syntactic dependencies:
 - (iv) Ruth: What did Alex design for
Hugh: Eliot? A music-box with flashing lights.
 - (v) Ruth: What did Alex
Hugh: design for herself? A self-loading washing-machine.
 - (vi) Ruth: Have you spoken to...
Hugh: any of the doctors? Not yet.
- Children can do it from earliest stages of language acquisition
 - (vii) Ruth: Old McDonald had a farm... And on that farm he had a
Eliot: cow.

Ellipsis - current heterogeneity

- Syntactic accounts postulate ambiguity and structure-specific stipulation
 - (1) John has checked his results but I haven't yet.
 - (2) I interviewed everyone that Bill did.
 - (3) I insisted Sue visit Bill in hospital, but not Harry.
 - (4) I insisted Sue visit Bill and Harry, Tom.
- Pragmatic explanation used only for some remainder
 - (5) At a cafe: Bacon and eggs, please
waiter to kitchen: Bacon and eggs.
waiter to client: bacon and eggs.
- Semantic accounts (as alternative to syntactic accounts) presume that the 'context' has to be massaged to provide appropriate input
 - (6) The man who arrested Bill failed to read him his rights,
and the man who arrested John did too.
- Neither account has any basis for characterising fragments in split utterances:
 - (7) Ruth: What shall I give..?
Hugh: Eliot? A teddy-bear.

Ellipsis - Filling out interpretation from Context

Can we recover the intuition that context directly provides a value?

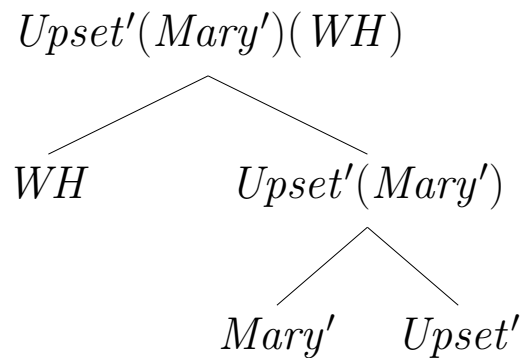
Context as record of structures and actions

Context-dependence: ellipsis (a) - re-use of terms from context

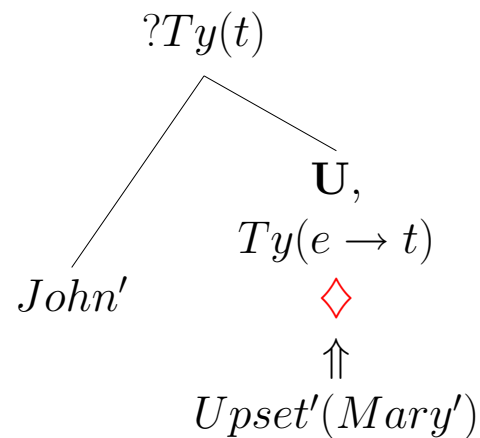
- Using **terms** from context:

(8) Q: Who upset Mary? Ans: John did.

CONTEXT AS TREE



TREE UNDER CONSTRUCTION:



(9) Parent to teenage son with surf-board standing in shallows:
 I wouldn't, if I were you. The flag's flying, so it'll be dangerous.

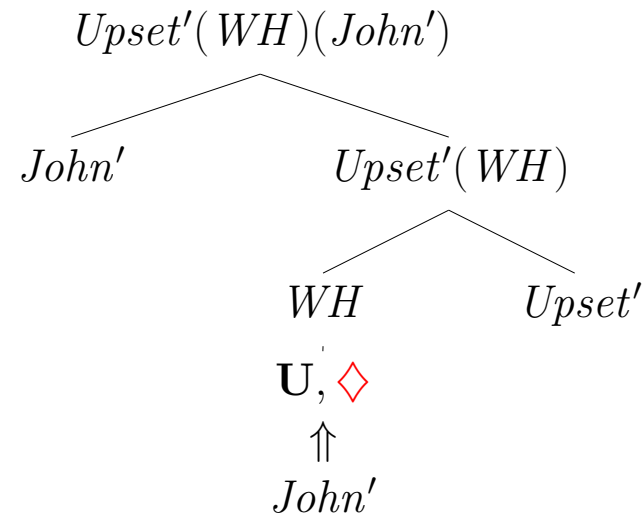
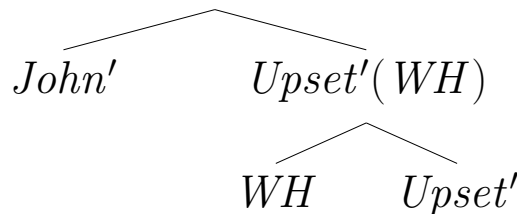
Context-dependence: ellipsis (b)

- re-use of **structure** from context : parser starts from partial tree

(10) Q: Who did John upset? Ans: Himself.

TREE AS CONTEXT: **becomes** TREE UNDER CONSTRUCTION:

Upset'(WH)(John')



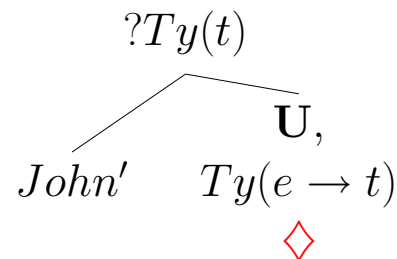
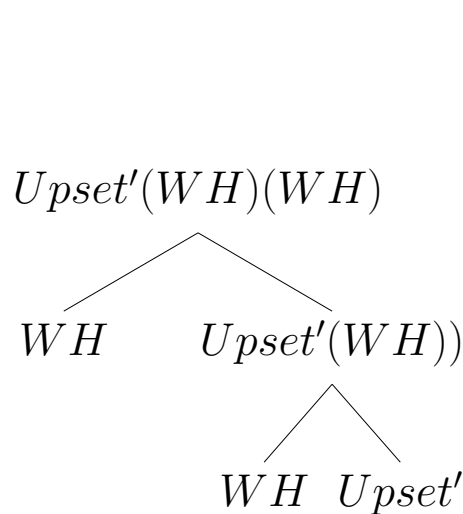
(11) Who did everyone ignore? Their husband.

(12) In cafe: (customer to waiter). Bacon-and-egg.
 (waiter to kitchen). Bacon-and-egg.

Context-dependence -ellipsis(c): re-use of actions from context

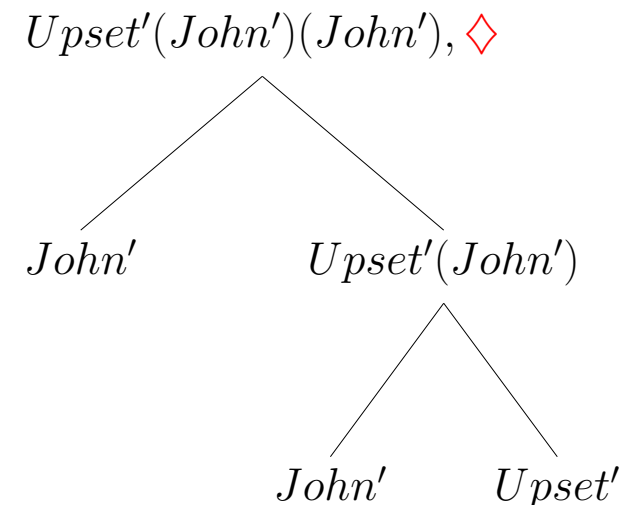
(13) Who upset himself? John did.

CONTEXT TREE TREE CONSTRUCTED



IF $?Ty(e \rightarrow t)$
 THEN make-go(\downarrow_1); put($Fo(Upset')$);
 go(\uparrow_1); make-go(\downarrow_0);
 put($?Ty(e)$); put($Fo(U, Ty(e))$);
 \langle IF $\uparrow_0 \uparrow_1^* \downarrow_0 \alpha,$
 THEN SUBSTITUTE(α), ELSE Abort \rangle

RESULT



(14) The man who arrested John failed to read him his rights.

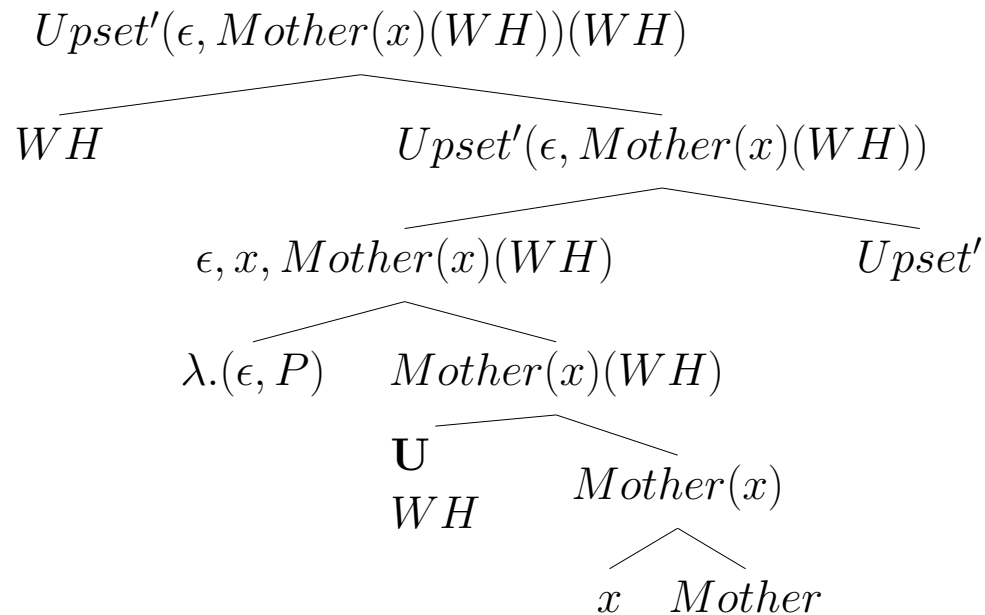
The man who arrested Tom did too.

(15) John interviewed everyone that Bill did.

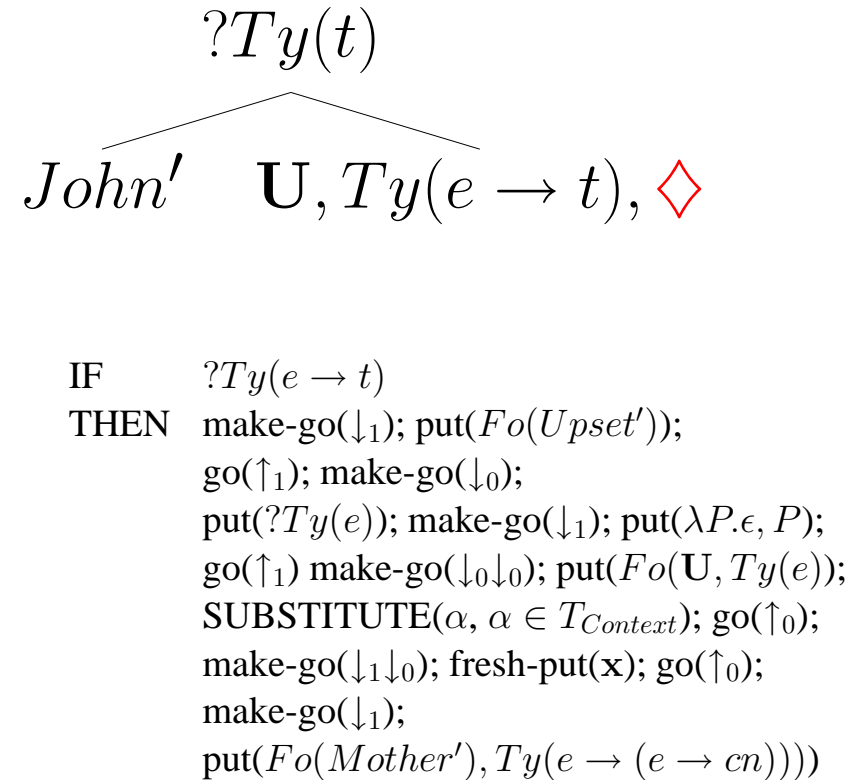
- Using **actions** from context – sloppy readings:

(16) Who upset his mother? John.

CONTEXT



TREE UNDER CONSTRUCTION

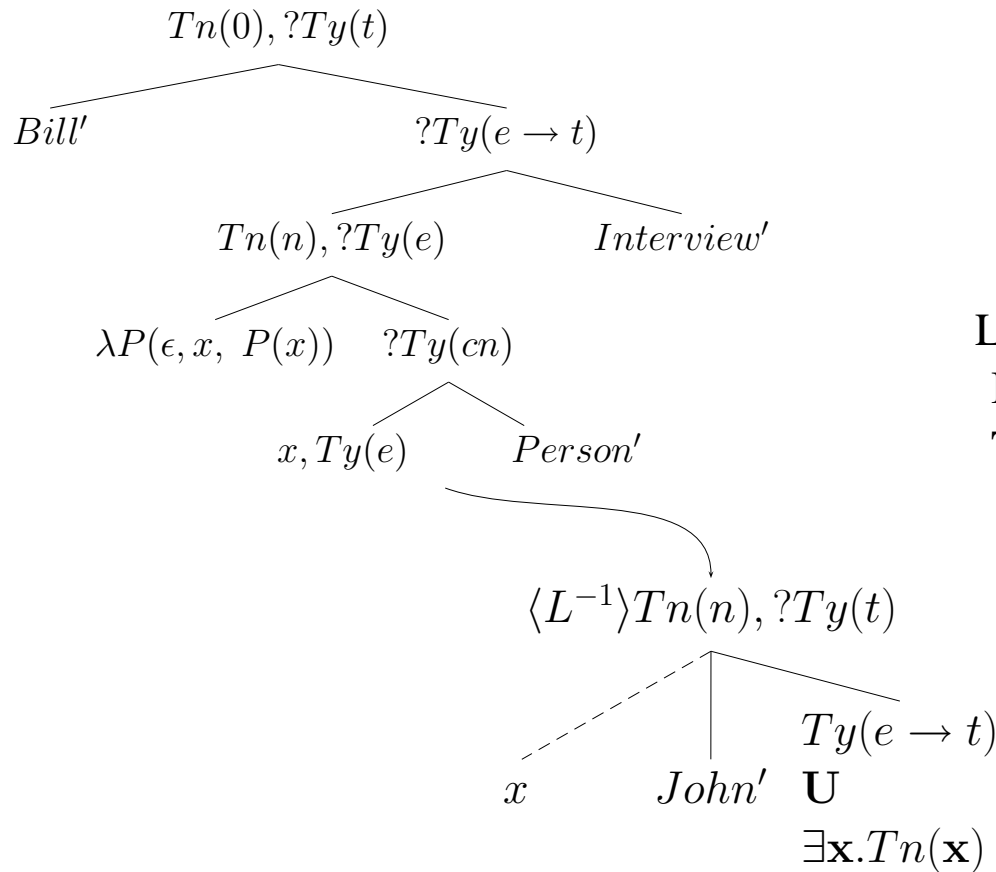


RESULT $Fo(\text{Upset}'(\epsilon x. \text{Mother}'(x)(\text{John}')))(\text{John}')$

Entire basis for parallelism is re-use of actions

Using actions from context cont...

- Bill interviewed someone that John did (antecedent contained ellipsis)



Lexical actions from partial tree as context:

```

IF      ?Ty(e → t)
THEN   make(⟨↓1⟩); go(⟨↓1⟩);
        put(Fo(Interview'), Ty(e → (e → t)))
        go(⟨↑1⟩); make(⟨↓0⟩)
        put(?Ty(e))
    
```

The Flow of Language Understanding – Parsing

- At any one transition in the construction process, there is a triple \mathfrak{T}_W :

$$\langle T, W, A \rangle$$

T a (possibly partial) propositional tree,

W a string of words so far parsed

A the set of actions (computational/lexical) used to construct T from W .

A PARSE STATE consists of a set of such triples

Final parse state: a non-empty set of triples of the form: $\langle T_\phi, \phi, A_\phi \rangle$

T_ϕ a complete propositional tree derived from ϕ by A_ϕ
with no outstanding requirements.

- Contexts for Parse States

A context \mathfrak{C} for some partial tree T_ϕ established in uttering a string ϕ is a sequence:

$$\mathfrak{C} = \mathfrak{C}_D \oplus \mathfrak{T}_\phi$$

\mathfrak{C}_D is a sequence of *inactive* triples (without a pointer)

.... $\langle T_W, W, A_W \rangle$..

– a ‘discourse context’,

\mathfrak{T}_ϕ is a current active triple of T_ϕ , the string ϕ , and actions A_ϕ

Getting towards dialogue modelling

Generation and Context

The Flow of Language Understanding – Generation

A GENERATOR STATE G is a pair

$$(X, T_G)$$

of a set X of pairs (S, P) , where

S is a candidate partial string

P is the associated PARSE STATE,

and

T_G a GOAL TREE,

which represents the content of the utterance to be produced.

Generation is thus characterised in **exactly** the same terms as parsing except that the the current parse state is constrained by the requirement that the current partial tree subsumes the goal tree.

Initial generator state G_0 will (usually) be the pair $(\{(\emptyset, P_0), T_G\})$:

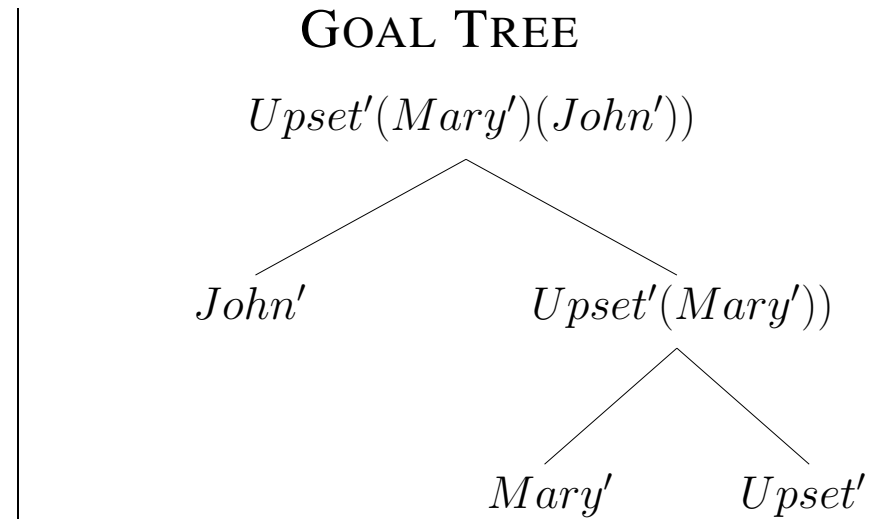
i.e. the null string and the initial parse state, plus the goal tree.

The Flow of Language Understanding – Generation

Generating ‘John upset Mary’:

PARSER TRIPLE:

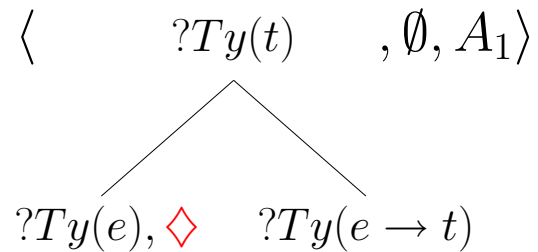
$\langle \{?Ty(t), \diamond\}, \emptyset, \emptyset \rangle$



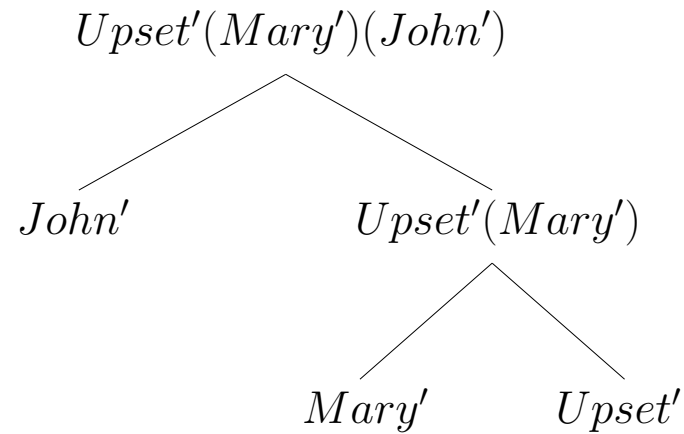
The Flow of Language Understanding – Generation

Generating ‘John upset Mary’:

PARSE TRIPLE



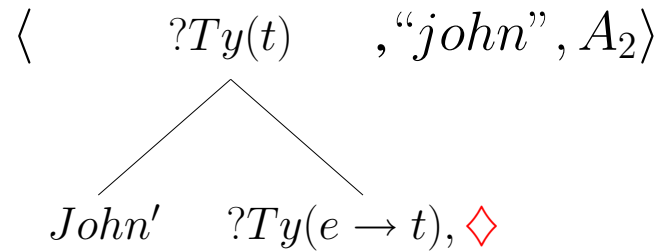
GOAL TREE



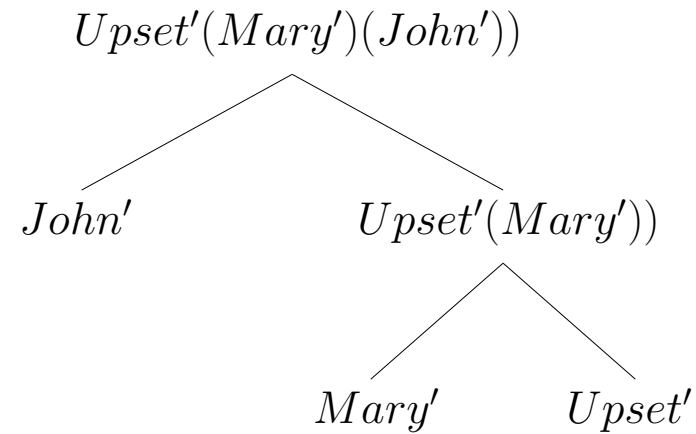
The Flow of Language Understanding – Generation

Generating ‘**John** upset Mary’:

PARSE TRIPLE



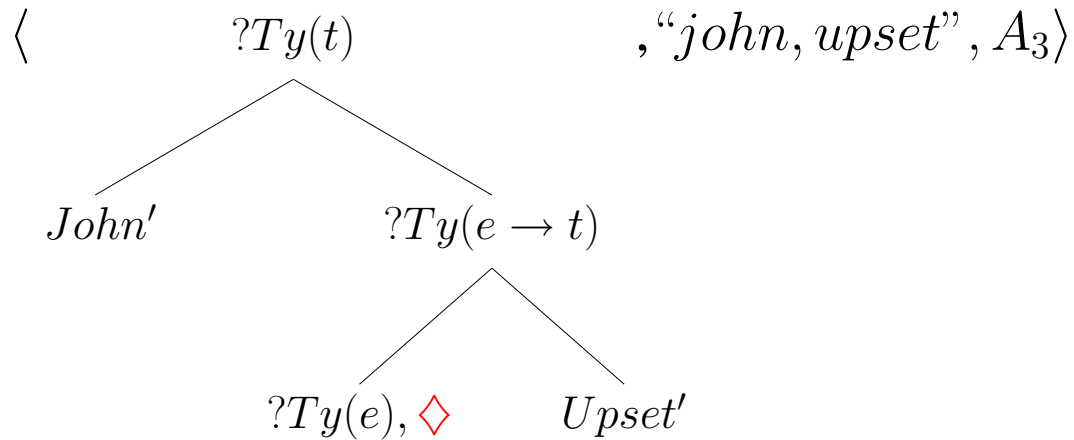
GOAL TREE



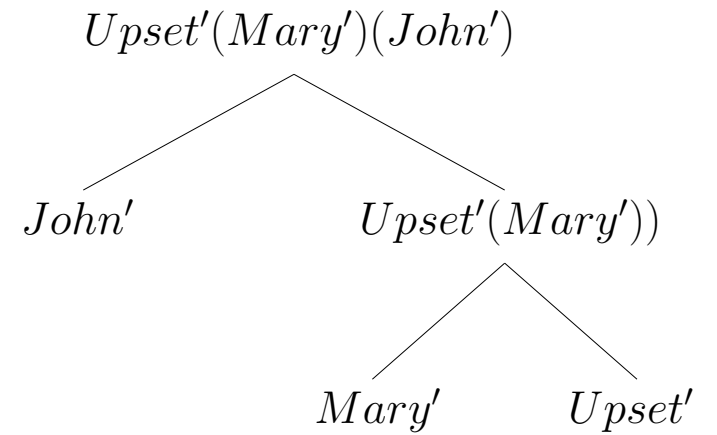
The Flow of Language Understanding – Generation

Generating ‘John upset Mary’:

PARSE TRIPLE:



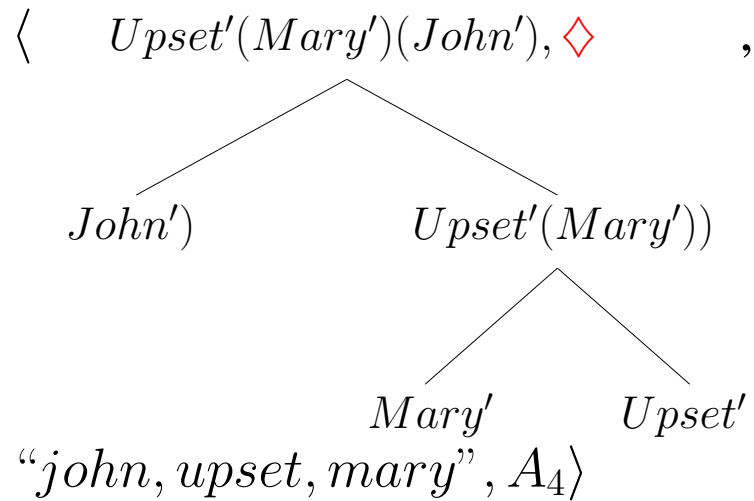
GOAL TREE



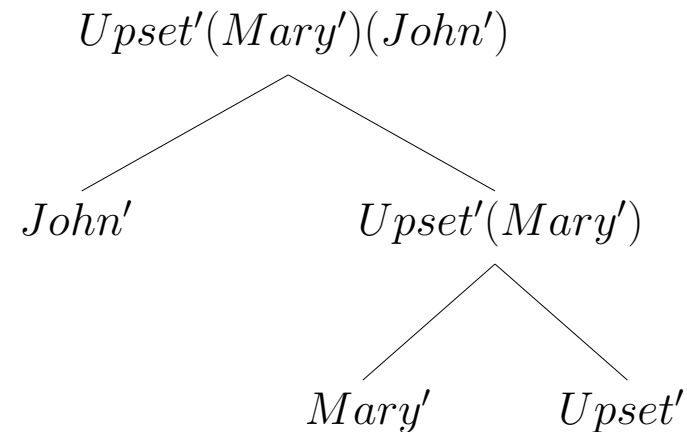
The Flow of Language Understanding – Generation

Generating ‘John upset Mary’:

PARSE TRIPLE



GOAL TREE

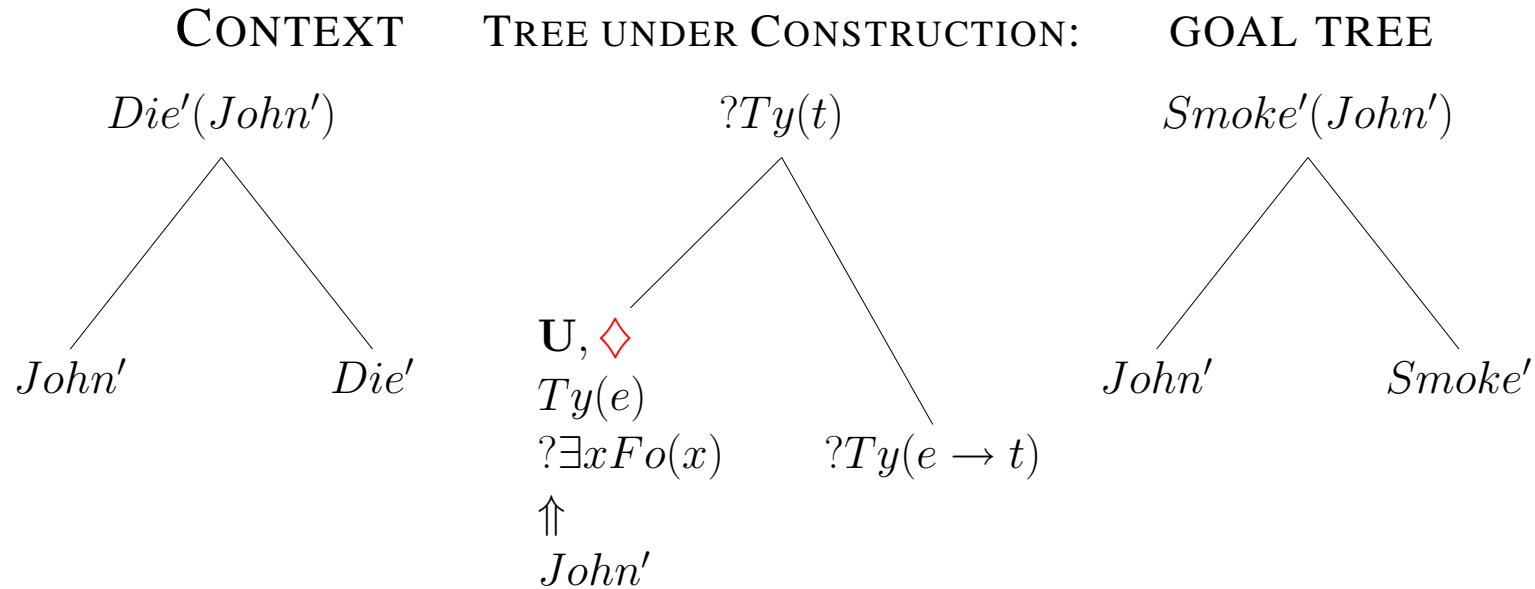


Generation relative to Context? – Avoiding full lexicon search

- As generation uses the same parsing actions, it uses context in the same way as parsing, eg for pronoun generation:

(17) John died. He smoked.

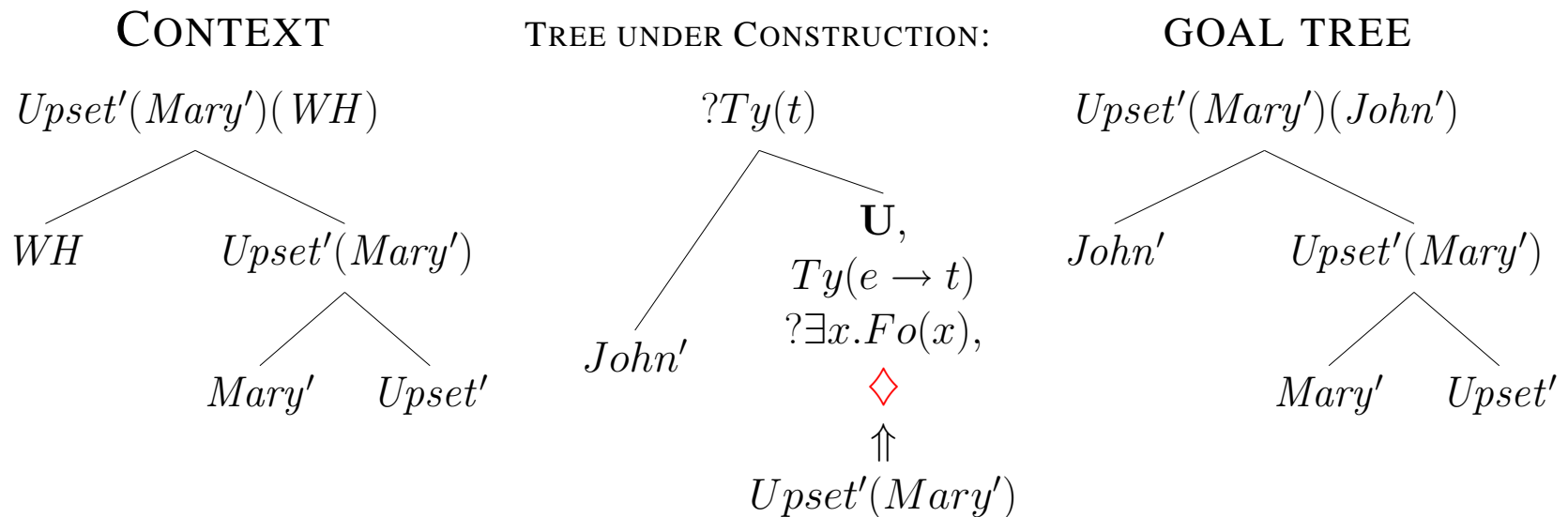
Parsing *he*:



Context-dependent generation : ellipsis (a)

- Ellipsis also requires use of **terms** from context, bypassing lexicon search:

(18) Q: Who upset Mary? Ans: John did.



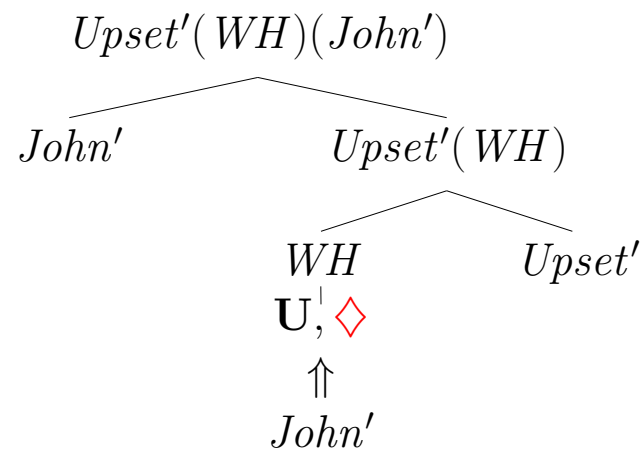
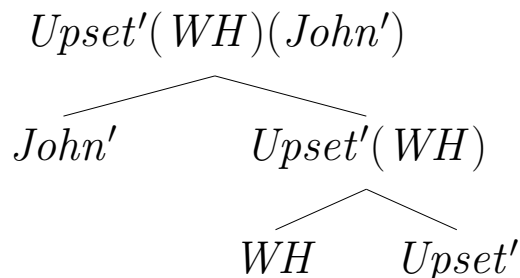
(19) A: The man from next door who owns a Cadillac crashed it into our wall.
 B: Did he?

Context-dependent generation : ellipsis (b)

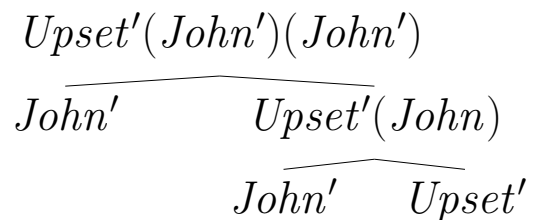
- Using **structure** from context - generator starts from partial tree:

(20) Q: Who did John upset? Ans: Himself.

PARSED TREE AS CONTEXT: **becomes** TREE UNDER CONSTRUCTION:



GOAL TREE



(21) Who did everyone ignore? Their husband.

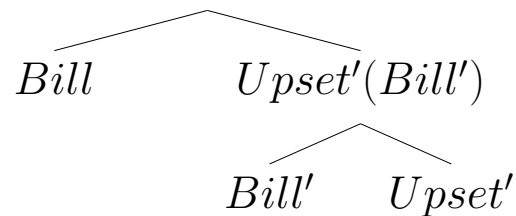
Context-dependent generation : ellipsis (c)

- Using (parsed) **actions** from context ensures viability of production task

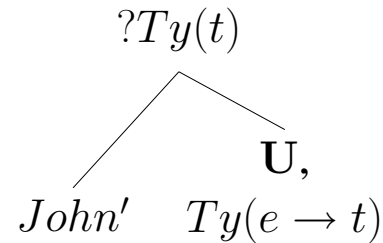
Bill upset himself. And John did.

CONTEXT

$Upset'(Bill')(Bill')$

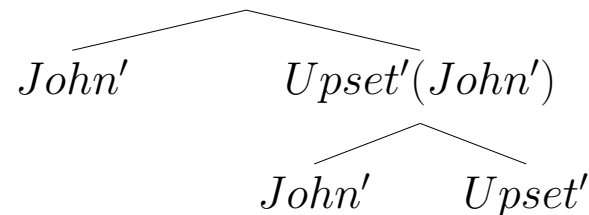


TREE UNDER CONSTRUCTION



GOALTREE

$Upset'(John')(John')$

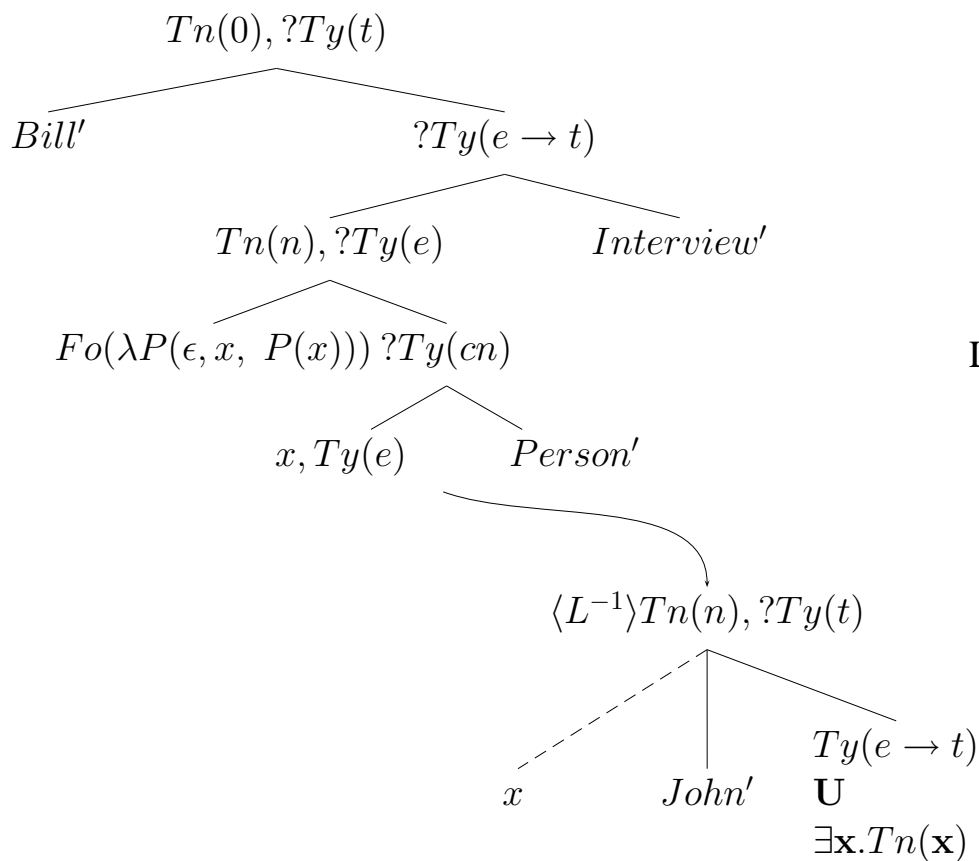


IF $?Ty(e \rightarrow t)$
 THEN make-go(\downarrow_1); put($Fo(Upset')$);
 go(\uparrow_1); make-go(\downarrow_0);
 put($?Ty(e)$);
 < IF $\uparrow_0 \uparrow_1^* \downarrow_0 \alpha$,
 THEN put(α), ELSE Abort >

Other eg's *John interviewed everyone Bill did ,*

Using actions from context cont...

- Bill interviewed someone that John did (antecedent contained ellipsis)



Lexical actions from partial tree as context:

IF $?Ty(e \rightarrow t)$
 THEN $make(\langle \downarrow_1 \rangle); go(\langle \downarrow_1 \rangle);$
 $put(Fo(Interview'), Ty(e \rightarrow (e \rightarrow t)))$
 $go(\langle \uparrow_1 \rangle); make(\langle \downarrow_0 \rangle)$
 $put(?Ty(e))$

Goal:

$Interview'(\epsilon, x, person'(x) \wedge Interview'(x)(John))(Bill)$

Dialogue: Parsing and Generation in Context

- (i) Pronouns and ellipsis
- (ii) Alignment
- (iii) Shared Utterances

Ellipsis and Pronouns

- Re-use of procedures
 - whether parsed or produced
 - whether current task parsing or production
- Use of actions bypasses lexicon; also shortcuts choices, since the sequence of actions in any one selection will itself constitute a sequence of choices.
- Good for hearer, essential for producer
- Coordination of parsing and production activity ensures presence of actions in context.

Alignment as Action Re-Use:Lexical Search Minimisation

- Lexical/syntactic/semantic alignment (priming) all as re-use of actions in context – instead of looking in lexicon, use lexical action from context
- eg use of syntactic pattern established (Branigan et al., 2000):

(22) A:John gave Mary a teddy-bear.
B: And Tom gave Sue a music-box.

(23) A: John gave a teddy-bear to Mary.
B: And Tom gave a music-box to Sue.

(24) A: How are your new neighbours?
B: John, I like. Sue, I don't much care for.

- Parallelism of structure and content in combination

(25) . The man who arrested John failed to read him his rights.
So did the man who arrested Bill.

(26) What did Mary give you?
She gave each of us a hug. Then a present.

- Priming no longer posited as a primitive:

Alignment = re-use of selected action-sequence from context

Shared Utterances

- Processes *necessarily* share same intermediate representations
 - parsing can begin from intermediate generator state
 - generator can begin from intermediate parser state
- Parsing/generation can both start from **any** parser state
 - replace initial ?**Ty**(t) requirement with current partial tree set
- No restriction from constituency or head position
- Cross-speaker effects seem unproblematic:
 - underspecification from long-distance dependency

(27) A: *What did Alex give ...* B: *... Eliot? A teddy-bear.*

– actions for indexicals involve current speaker/hearer reference

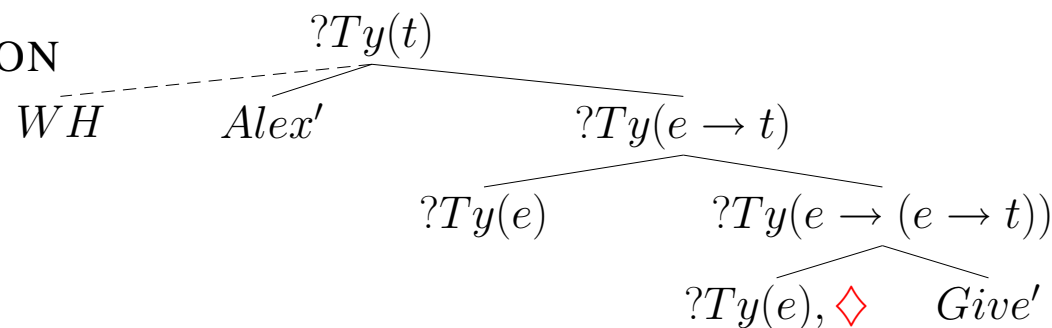
(28) A: *Have you ...* B: *... read your latest chapter?*

Generation of Shared Utterances

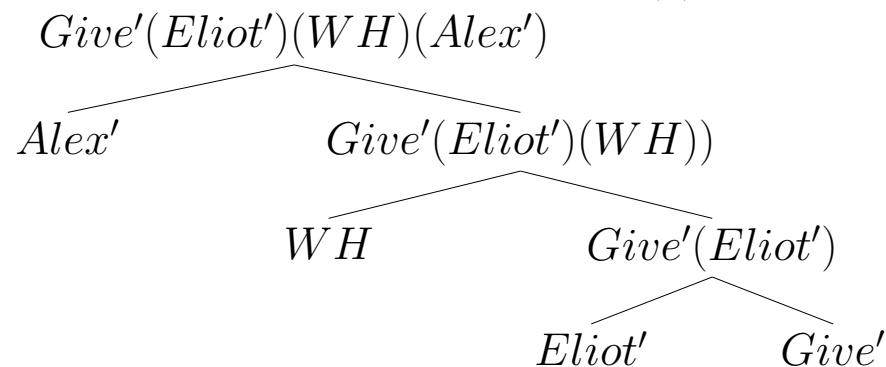
(29) *Ruth*: What did Alex give ...
Hugh: ...Eliot? A music-box.

PRIOR TO SPEAKER/HEARER TRANSITION:

TREE UNDER
CONSTRUCTION



GOAL TREE

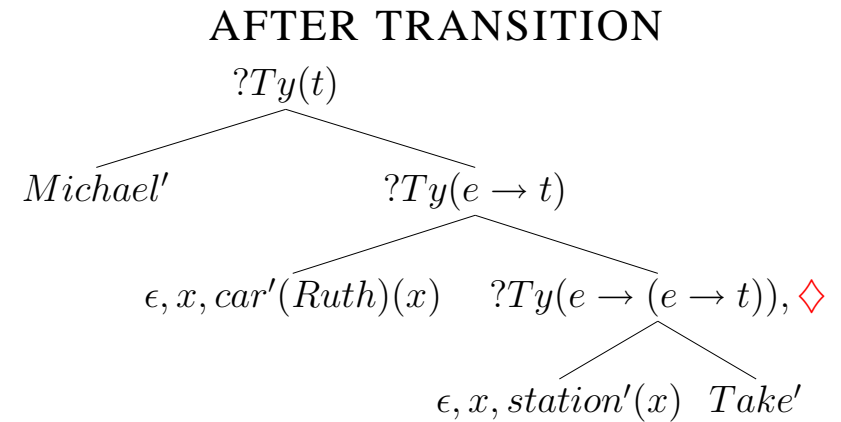
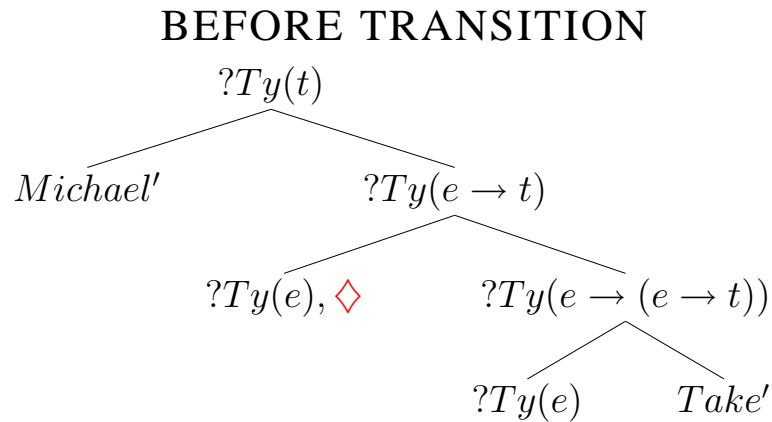


Speaker/Hearer coordination is an emergent property.
 Incremental enrichment for hearer licenses shift of roles.

Switch of indexicals

(30) Ruth: You took ...

Michael: your car to the station? No I didn't.



Towards Modelling Latin Dialogues!

- (i) Generation for Latin
- (ii) Ellipsis Generation
- (iii) Coordination in Speaker/Hearer Exchange
- (iv) Linearisation Decisions in free-order languages

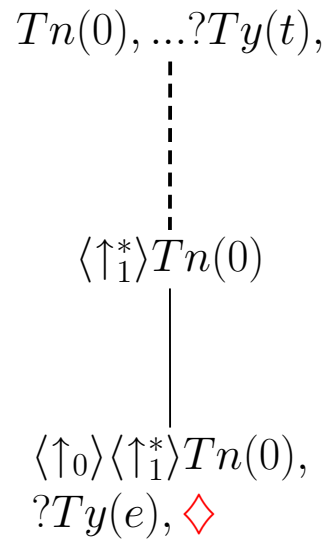
Ellipsis Data in Latin

- (31) *Xerxes iussit milites castrum captare. Praemium proposuit.*
 Xerxes ordered the soldiers to capture the camp. He offered a reward.
- (32) *Sed si poterit solam brassicam esse, edit*
 But if he is able only to eat cabbage, he could eat it. (Cato)
- (33) *X: Quis praemium proposuit? Y: Xerxes*
 X: Who offered a reward? Y: Xerxes
- (34) *Quis eos postulavit? Appius. Quis produxit? Appius.*
 Who interrogated them ? Appius. Who led them? Appius. (Cicero)
- (35) *X: Redis tu tandem? Y: Redeo.*
 Are you back already? Yes, I am. (Plautus)
- (36) *X: Clodius insidias fecit Miloni? Y: Fecit.*
 X: Did Clodius make plots against Miloni? Yes, he did.
- (37) *Mihi valde placent; mallem tibi.*
 I am very pleased with them; I would prefer that you were.
- (38) *Si sine febre erit, dato vini..... Si febris erit, aquam.*
 If there is no fever, give wine [dark, watered down]. If there are fevers, give water. (Cato)
- (39) *Te, dea, te fugiunt venti, te nubila coeli.*
 You, goddess; you, winds flee from; you, clouds of heaven flee too.

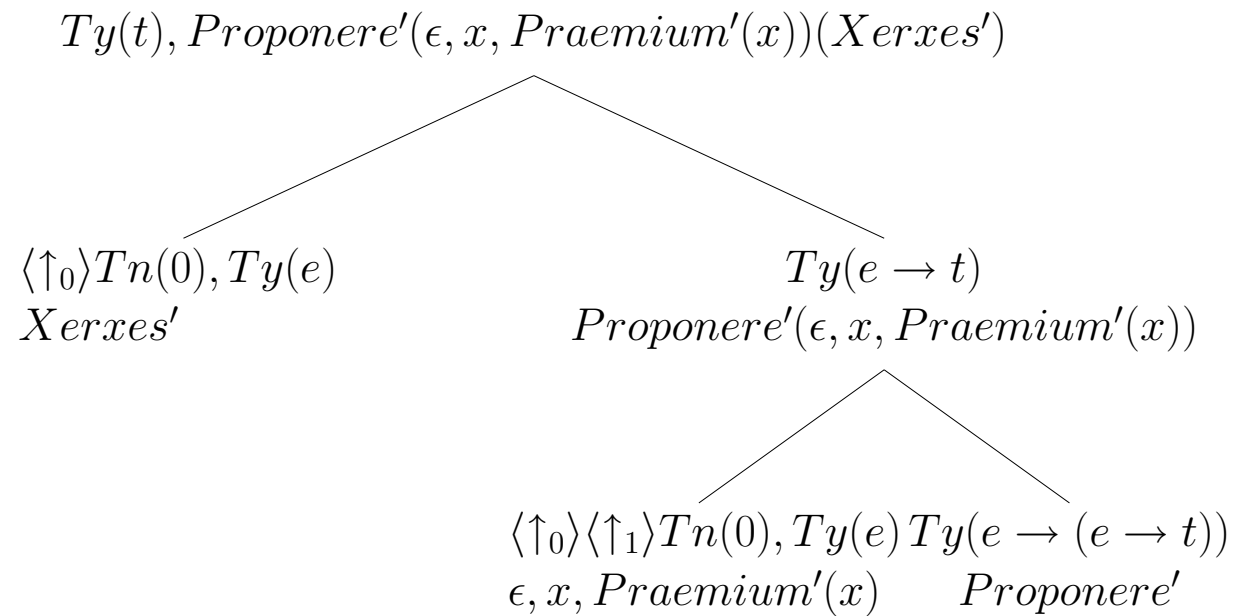
Incremental generation for scrambling (1)

Generating *Praemium Xerxes proposuit* using a “goal” tree as check on licensed construction-moves

Construction Tree



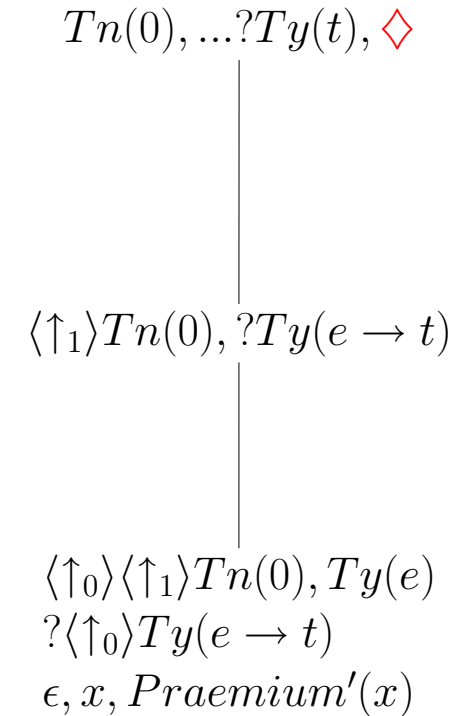
Goal Tree



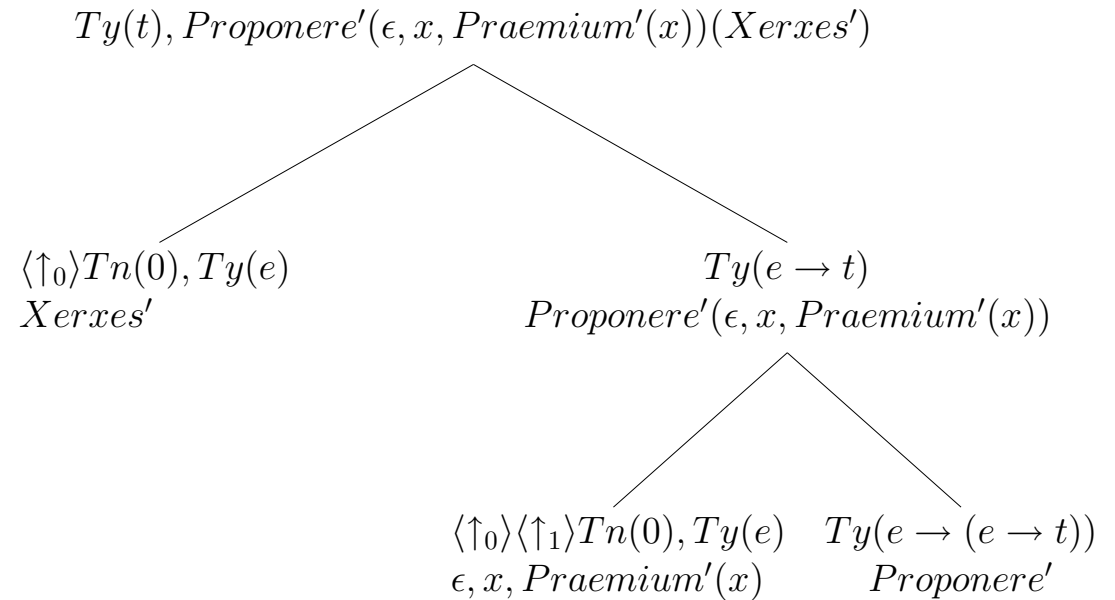
Incremental generation for scrambling (2)

Generating *Praemium Xerxes proposuit*

Construction Tree



Goal Tree

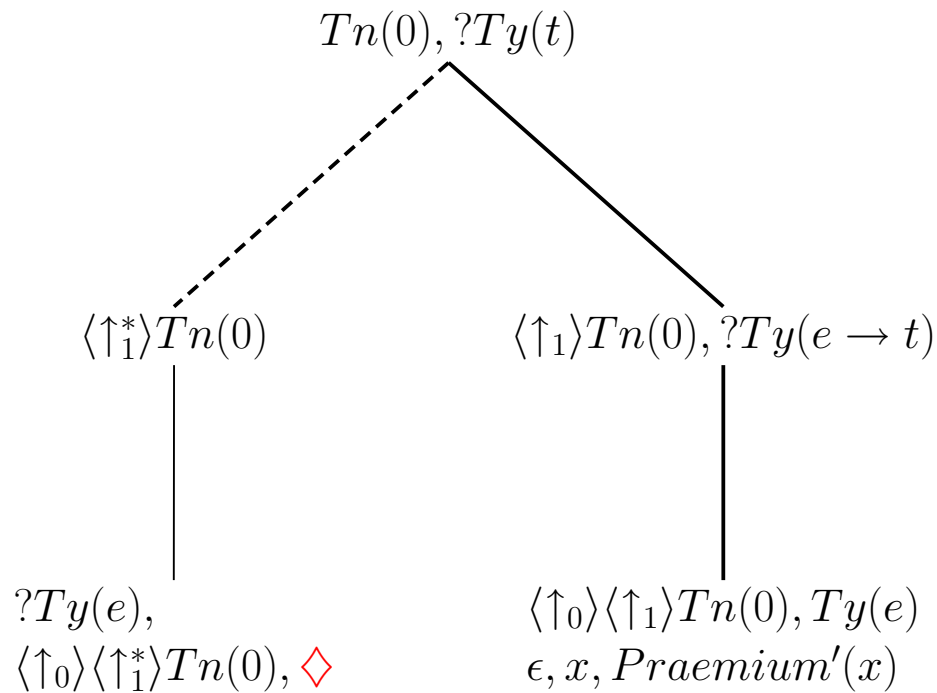


Praemium

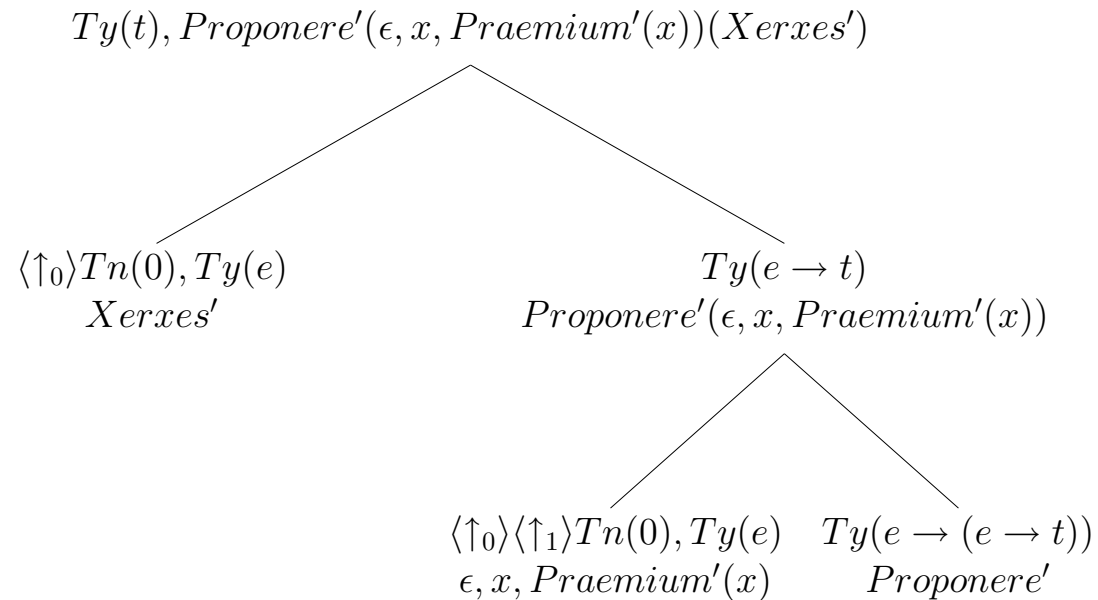
Incremental generation for scrambling (3)

Generating *Praemium Xerxes proposuit*

Construction Tree



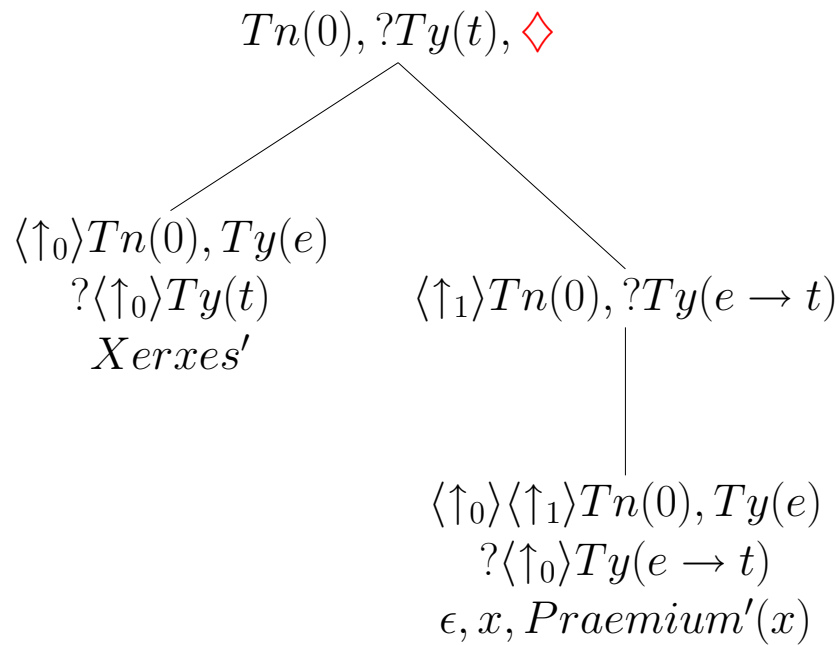
Goal Tree



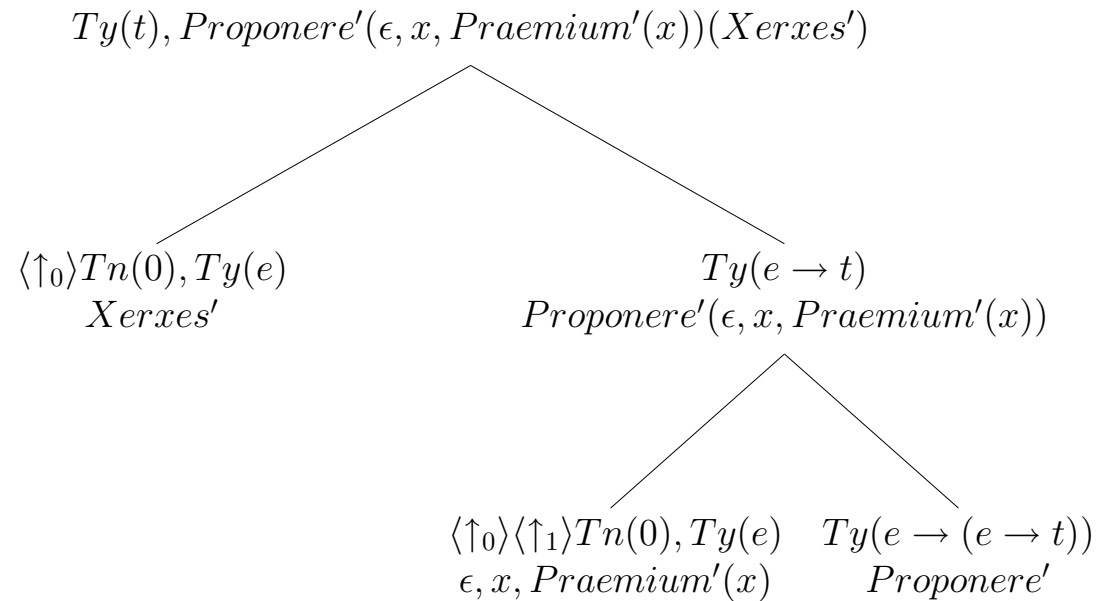
Incremental generation for scrambling (4)

Generating *Praemium Xerxes proposuit*

Construction Tree



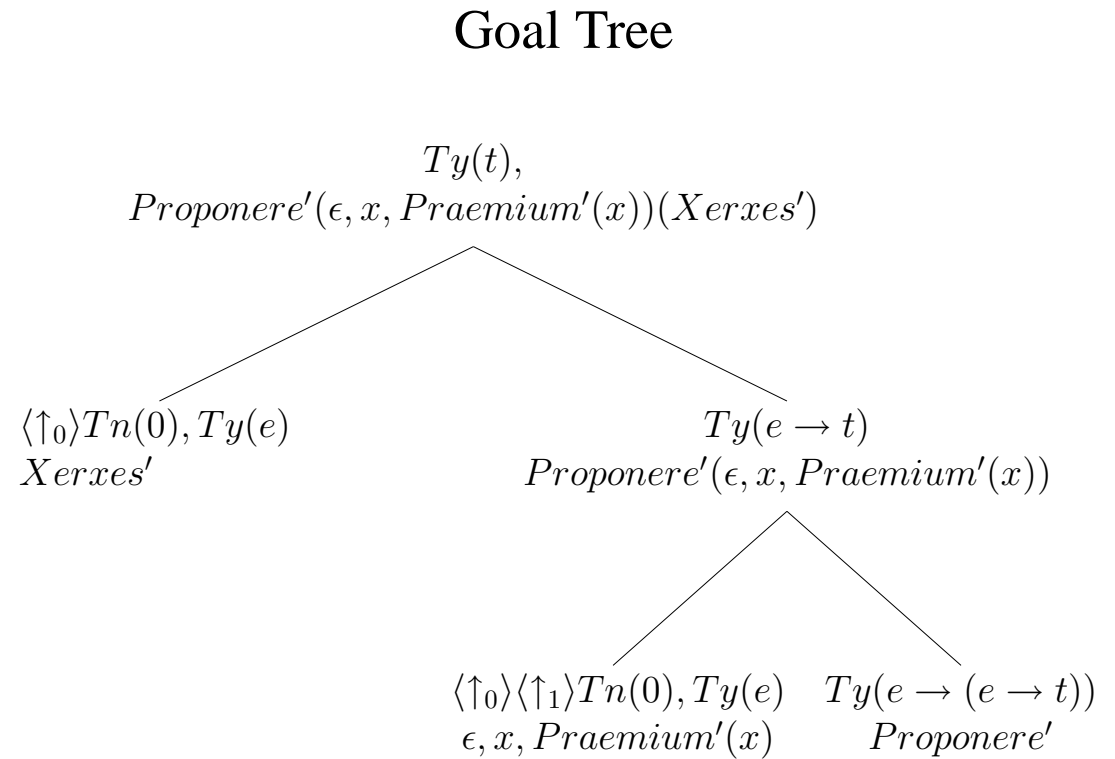
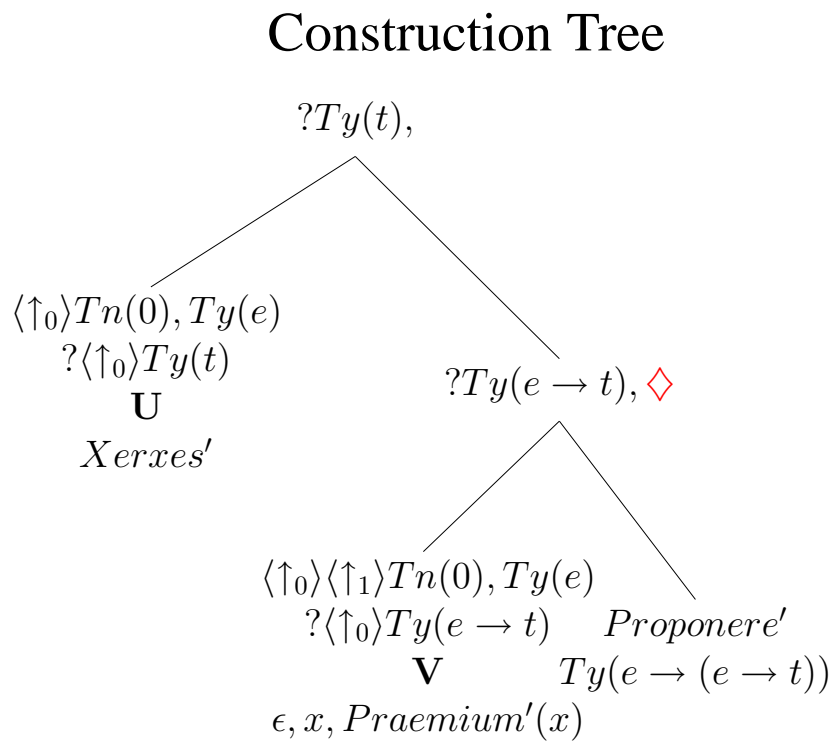
Goal Tree



Praemium Xerxes

Incremental generation for scrambling (5)

Generating *Praemium Xerxes proposuit*



Praemium Xerxes proposuit

Context-dependent generation: Use of terms from context

Challenge of incremental word search - full lexicon search is expensive

Context used to bypass lexical search mechanisms

Anaphora resolution: substitution of terms from context

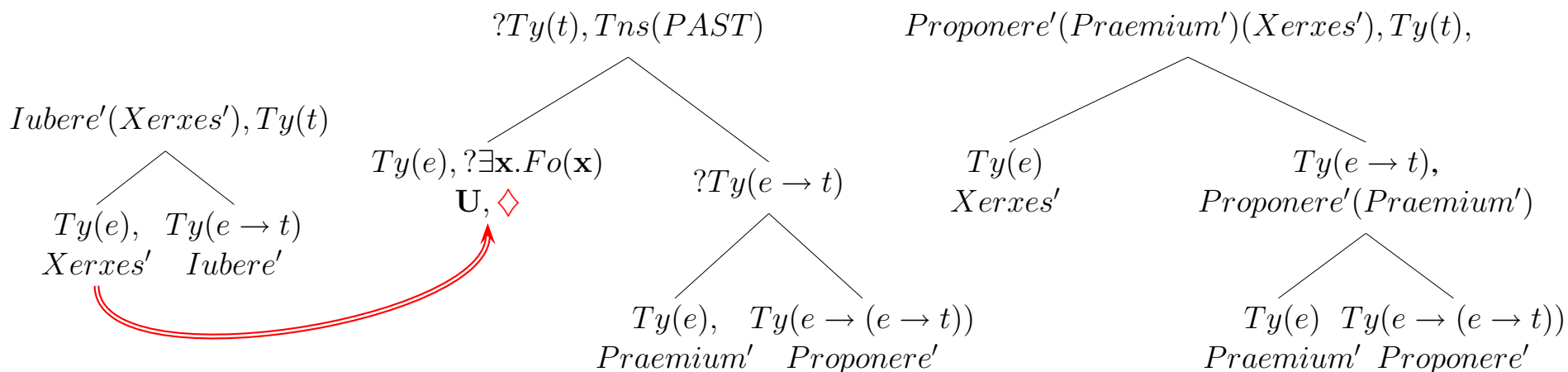
Xerxes iussit milites castrum captare. Praemium proposuit.

Xerxes ordered the soldiers to capture the camp. He offered a reward.

CONTEXT:

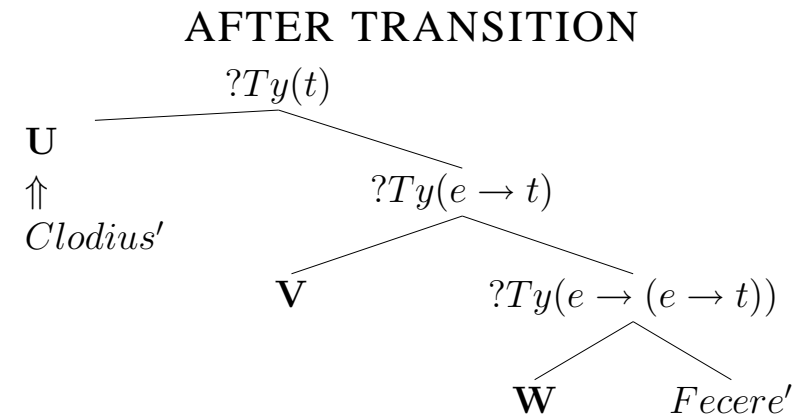
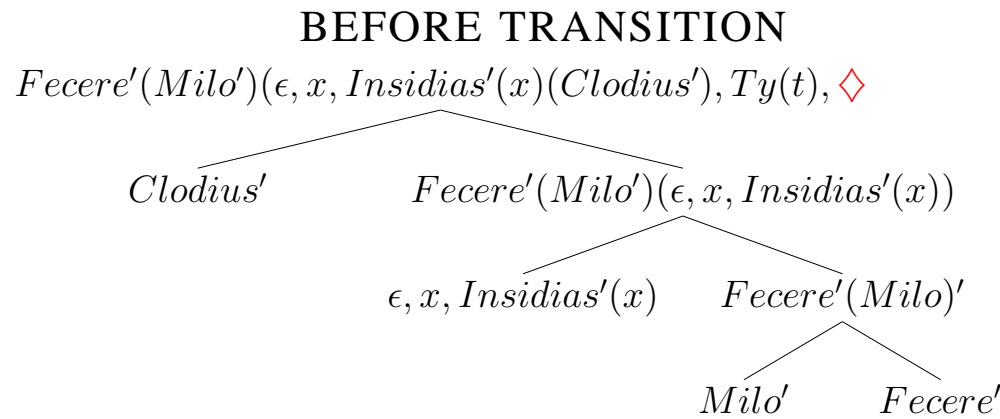
TREE UNDER CONSTRUCTION

GOAL TREE:



Multiple argument provision from context

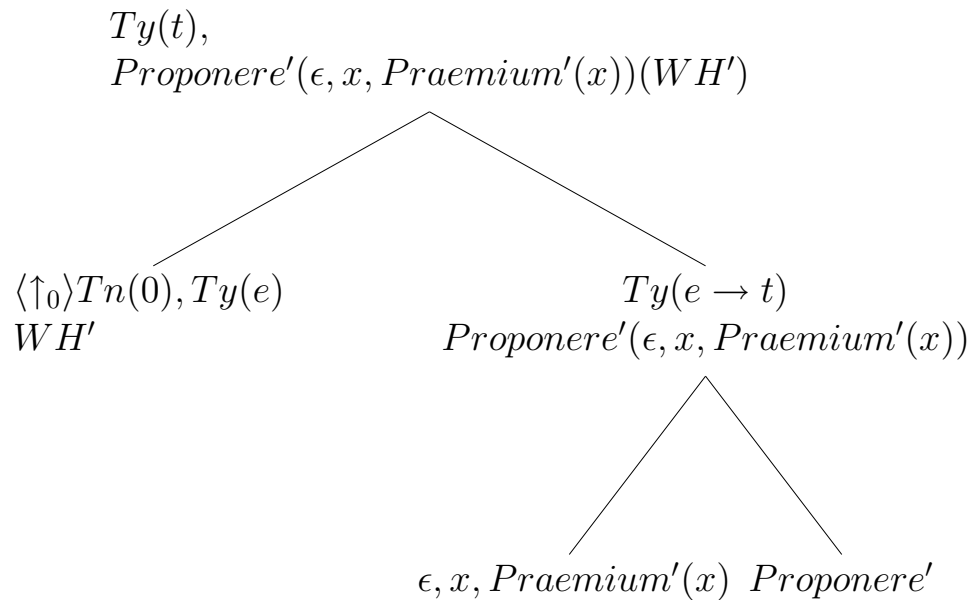
- (40) *Clodius insidias fecit Miloni? Fecit.*
 Clodius_{NOM} plots made Miloni_{DAT} Made
 Did Clodius plot against Milo? Yes, he did.



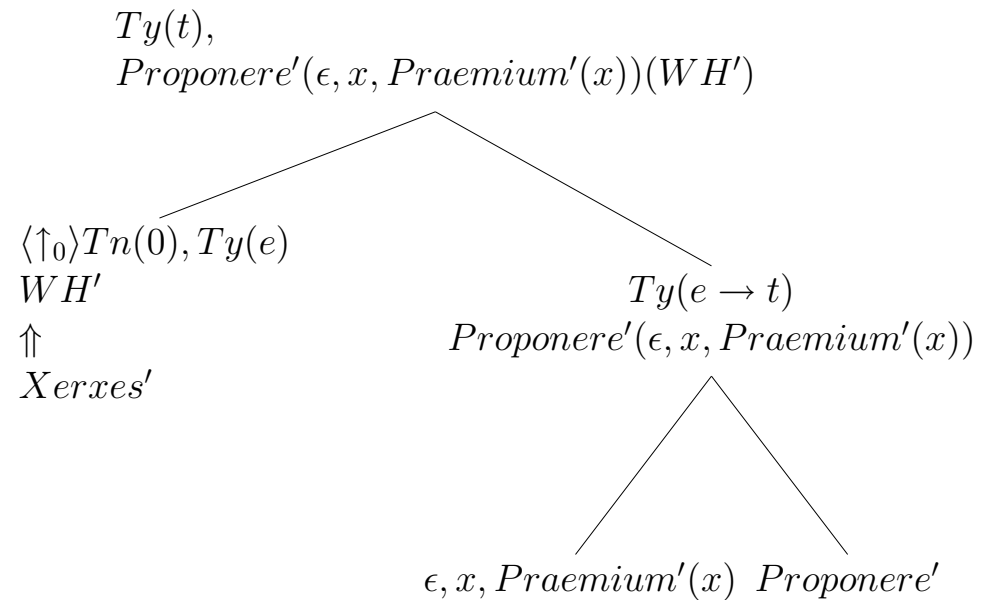
Use of Structure from Context

- Question-answer sequences
 - X: *Quis praemium proposuit?* Y: Xerxes
 - X: Who_{NOM} reward offered Y: Xerxes
- X: Who offered a reward? Y: Xerxes

Context Tree



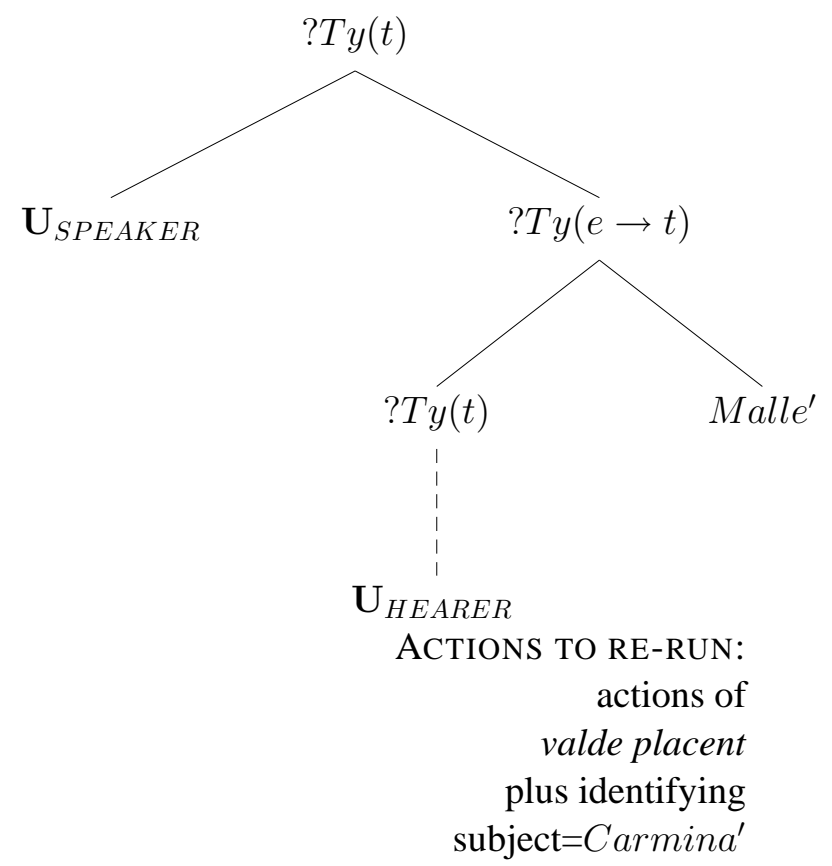
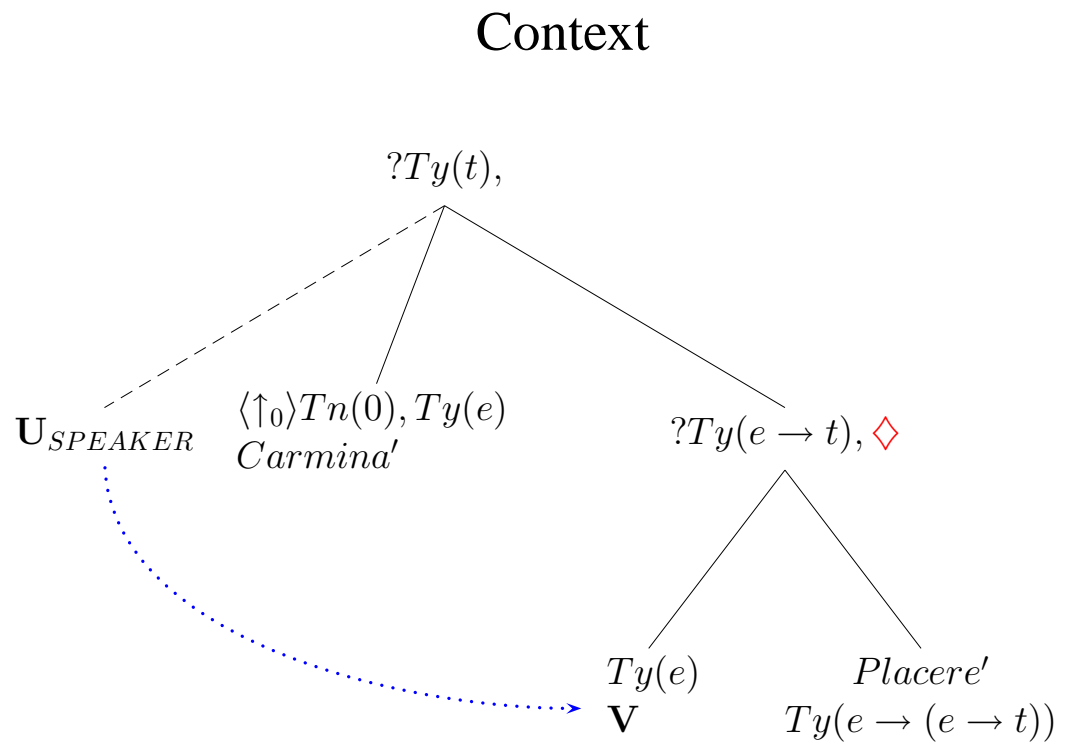
becomes Content Tree



Re-Use of Actions from context

(41) *Mihi valde placent malle* *tibi*
 me_{DAT} strongly please, prefer_{1st.sg.Imp.Subjunct} you_{DAT}
 They've pleased me a lot, I would prefer that they pleased you.

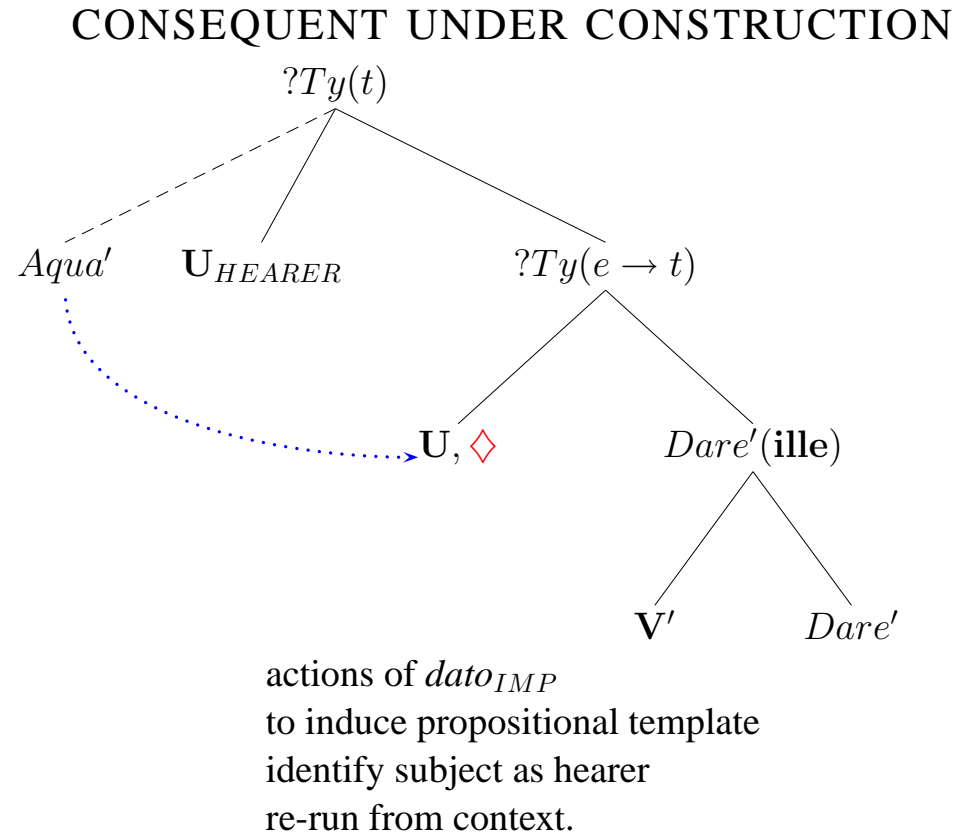
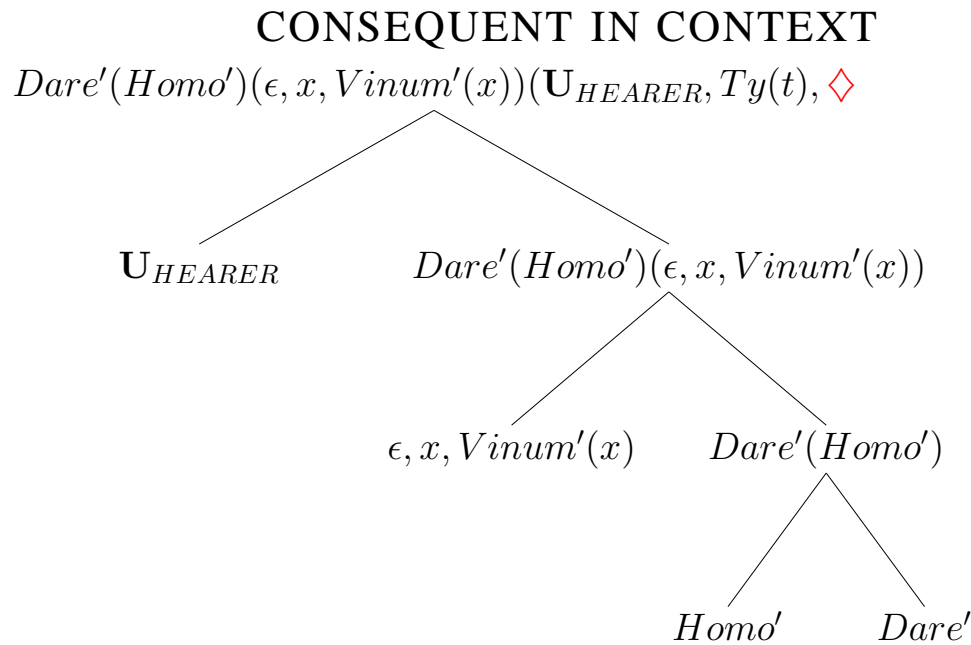
Construction Tree



Re-Use of Actions from context

Si sine febre erit, dato vini..... Si febris erit, aquam.

If there is no fever, give wine, If there are fevers, give water.



Switch of Indexicals

Actions built onto structure from context (as confirmation)

(12a) X. *Redis tu ?* Y. *Redeo*
 Return_{2.ps} you ? Return_{1.ps}

X. Are you back ? Y. Yes I am

Y'S CONTEXT FOR GENERATING *redeo* [TREE/ACTIONS]

Y'S GOAL TREE

Redire'(Y), Ty(t)

IF ?*Ty(t)*
 THEN make($\langle \downarrow_1 \rangle$); go($\langle \downarrow_1 \rangle$); put(*Fo(Redire')*, *Ty(e → t)*); go(\uparrow_1);
 make($\langle \downarrow_0 \rangle$); go($\langle \downarrow_0 \rangle$); put(*Fo(U)*), SubstituteU = Y go($\langle \uparrow_0 \rangle$);

Repetition of verb as acknowledgement is widespread cross-linguistic pattern

Summary

- The context is not just representations, but stored actions.
- Dialogue patterns as a consequence, with production, following parsing, making critical use of context.
- Dynamic Syntax assumes:
 - The centrality of underspecification plus update
 - Syntax as progressive and incremental building of conceptual representations
 - Closely twinned production and parsing, with parsing basic.
- Coordination in dialogue a consequence of language-as-parsing-mechanism
- Interweaving of grammar-internal and relevance-constrained processes

Ellipsis References

- Branigan, H. et al 2000. Syntactic co-ordination in dialogue. *Cognition* 75. B13-25.
- Cann, R. R. Kempson, and M. Purver. 2007. 'Context wellformedness: the dynamics of ellipsis.' *Research on Language and Computation*, 5.
- Dalrymple, M, Shieber, S. and Pereira, F. 1991. 'Ellipsis and higher-order unification'. *Linguistics and Philosophy* 14, 399–452.
- Fernandez, R. 2006 'Nonsentential utterances in dialogue: classification, resolution and use'. Ph.D. London University.
- Fiengo, R. and May, R. 1994. *Indices and Identity*. Cambridge MA: MIT Press.
- Garrod, S. and Doherty, G. 1994. 'Conversation, co-ordination and convention: an empirical investigation of how groups establish linguistic conventions'. *Cognition* 53, 181-215.
- Ginzburg, J. and Cooper, R. 2004. 'Clarification, ellipsis, and the nature of contextual updates'. *Linguistics and Philosophy* 27, 297–366.
- Hardt, D. fthcg 'VP ellipsis and constraints on interpretation' in Johnson, K. *Topics in Ellipsis*. CUP
- Lappin, S. 1999. 'An HPSG account of antecedent-contained ellipsis' in Lappin, S. and Benmamoun, E. (eds.), *Fragments: Studies in Ellipsis and Gapping*, 68-97. OUP.
- Ludlow, P. 2004. 'A note on alleged cases of nonsentential assertion' in Elugardo, R and Stainton, R. (eds.), *Ellipsis and Nonsentential Speech*, 95–108. Dordrecht: Kluwer.
- Merchant, J. 2004. 'Fragments and ellipsis'. *Linguistics and Philosophy* 27, 661–738.
- Pickering, M. and Garrod, S. 2004 'Toward a mechanistic psychology of dialogue' *BBS* 27, 169–226
- Purver, M. 2004. 'The theory and use of clarification requests in dialogue'. Ph.D. London.
- Purver, M. Cann, R. and Kempson, R. 2006. 'Grammars as parsers: Meeting the dialogue challenge'. *Research on Language and Computation*, 4 :289–326.
- Stainton, R. and Elugardo, R. 2004. 'Shorthand, syntactic ellipsis, and the pragmatic determinants of what is said'. *Mind and Language* 19, 442–71.