Information Structure

Overview

- What is Information Structure?

- Topic and Focus

- Representations

- Information structure marking constructions

- Variable property mapping

Information structure

- Information structure is the study of how speakers structure sentences to convey new information linked to the preceding context by old information.

- Many (all?) languages have particular linguistic resources (morphological, syntactic or prosodic) which facilitate the encoding of information structure.

- Lambrecht (1996) describes information structure as that which distinguishes between "allosentences": sentences with the same truth conditions that are nonetheless not felicitous in the same set of contexts.

  - Kim read a book.

  - It was Kim who read a book.

  - It was a book that Kim read.

Topic and Focus

- There is as yet little consensus in the literature about the range of information structural distinctions that must be drawn in an adequate model of language, nor on the definition of several key concepts.

- In the spirit of incremental development, we will begin by modeling each element in a sentence as topic-marked, focus-marked, or unmarked.

- Not worrying for now about:

    - Correlations between grammatical functions and info-str

    - Focus projection

Topic and Focus

- Topic and focus are properties of *linguistic expressions* that refer to referents, rather than of the referents themselves.

- Topic: A linguistic expression which refers to a known or inferable referent that the rest of the sentence provides more information about.

- Focus: The new information asserted by the speaker, against the background of that which is presupposed to be shared in the common ground (topic and tail).

  - Focus test: The part of the sentence answering an appropriate wh question is the focus: Who left? Ivan left.

- Tail: The rest of the sentence, that which is neither focus nor topic.

As-for topics

- In some languages/sentences, the topic element already plays a role in the clause (argument or adjunct).

- In other languages/sentences, the topic only plays the role of topic to the clause. It should be linked via a "_topic_p_rel".

  - In English, this is marked with "as for".

  - In other languages, as-for topics can take on ordinary topic marking.

  Amerika ha   supiido suketaa ga   hayai
  America TOP speed   skater   NOM fast

  'As for America, the speed skaters are fast.'

NB

- Every sentence is presumed to have a focus (possibly the entire sentence), but not every sentence has focus overtly marked.

- Not every sentence has a topic. Furthermore, in many languages at least, topics don't need to be overtly marked as such.

- Our "unmarked" means "not overtly marked for topic or focus", rather than "neither topic nor focus".

Representations

- We will be representing information structure via a new list of constraints called ICONS (individual constraints). Each element of the ICONS list will have a type indicating the information structure status it represents and the features CLAUSE and TARGET. CLAUSE should be identified with the INDEX (event) of the verb of the clause in question, TARGET with the INDEX (event or individual) of the word serving as the topic or focus.

- ICONS is a new feature of the type mrs, but unfortunately not yet integrated into the MRS-processing machinery (for display, generation or translation), so:

  - We have to work a little harder to see the effect of what we implemented.

  - We won't be able to have it reflected in generation or MT.

Representations: ICONS and --ICONS

```
mrs := mrs-min &
   [ HOOK hook,
     RELS diff-list,
     HCONS diff-list,
     ICONS diff-list ].
```

ICONS values are appended like HCONS values up the tree.

```
hook := avm &
   [ GTOP handle,
     LTOP handle,
     INDEX individual,
     XARG individual,
     --ICONS icons ].
```

--ICONS serves as a pointer to the particular icons added by a lexical entry (goes up the head path)

Representations: icons

```
icons := avm &
  [ CLAUSE individual,
    TARGET individual ].

info-str := icons.
topic := info-str.
contrast := info-str.
focus := info-str.
aboutness-topic := topic.
frame-setting-topic := topic.
contrast-topic := contrast & topic.
contrast-focus := contrast & focus.
semantic-focus := focus.
```

Information structure marking constructions

- Distinguished position in the sentence: Sentence initial, pre-sentence, pre-verbal, post-verbal, post-sentence

- Clitics or adpositions

- Cleft constructions (outside the scope)

- Focus prosody (can pseudo-model)

- NB: In some languages, information structure is more grammaticalized than in others.

Distinguished position

- Pre- or post-verbal: add a rule type which insists on a lexical ([LIGHT +]) verb and constrains the INFO-STR of the non-head daughter. Cross-classify this type with all appropriate head-nexus phrases.

- Sentence initial position: Modify existing rules for this position to cosntrain INFO-STR of left daughter.

- Pre-sentence position: Add a new phrase structure rule (possibly a head-filler rule, possibly an as-for topic rule) which licenses one constituent before the rest of the clause and constrains the INFO-STR on that constituent.

Clitics or adpositions

- Clitics (modifiers?) constrain the --ICONS of the head they modify.

- SLM "=jo" and Georgian "კი" are of this type.

- Adpositions constrain the --ICONS of their complement.

- In as-for topic constructions (see below), the adposition would serve as a modifier for the whole clause, introducing the "_topic_p_rel", while also constraining the --ICONS of their complement.

- In other topic constructions, the adposition functions like a case-marking one, passing up the semantic information of its complement.

Focus prosody

- In many languages, prosodic emphasis on the focussed element might be the only systematic mark of information structure.

- Such prosody is not ordinarily marked in orthography, but for the purposes of this lab, we will mark it with a suffix -FP.

- The goal is to translate Russian "Ivan li kupil sobaku" to "Did Ivan-FP buy a dog?" and "Sobaku-li Ivan kupil" to "Did Ivan buy a dog-FP?"

- Model via a add-only-no-ccont-lex-rule & inflecting-lex-rule, constrained to apply last in the pipeline, which adds the -FP affix and contributes the information that the --ICONS is focus.

As-for topics and pro-drop

- If your language has both as-for topics and pro-drop, there is a decision to be made about how much ambiguity to accept.

- Strategy one: Allow both as-for topics and regular (clause-connected) topics, getting two parses whenever the position the topic is linked to is not overtly filled.

- Strategy two: Treat all topics as as-for topics, and write extra transfer rules for the MT demo.

Ohno ha   hayai
Ohno TOP fast

'As for Ohno, he is fast.'
'Ohno is fast.'

Word order, information structure, sform and MKG

- When information structure is marked via specific positions in the sentence, then we need to add constraints to the phrase structure rules, and possibly add phrase structure rules.

- The feature MKG is a syntactic feature that can be used to mediate between morphosyntatic information and ICONS.

- sform is the root of a small hierarchy of types that constrain MKG and can relate it to ICONS. To be of use, these have to be inherited by instantiated phrase structure rules (e.g., subtypes of head-subj etc).

sform and MKG

```
;;; sentential forms (syntactic types which make differences in
;;; information structure)
sform := basic-binary-headed-phrase.

topicality := sform.
topic-comment := topicality &
   [ SYNSEM.LOCAL.CAT.MKG tp,
     HEAD-DTR.SYNSEM.LOCAL.CAT.MKG fc,
     NON-HEAD-DTR.SYNSEM.LOCAL.CAT.MKG tp ].
topicless := topicality.

focality := sform.
narrow-focus := focality.
wide-focus := focality.

topic-focus-bg := topic-comment & narrow-focus.
topic-bg-focus := topic-comment & narrow-focus.
topic-focus := topic-comment & wide-focus.
focus-bg := topicless & narrow-focus.
all-focus := topicless & wide-focus.
```

More on distinguished positions

- In some cases, the particular position is already licensed/filled by a specific set of phrase structure rules (e.g., pre-V position in V2). These rules can be modified to constrain --ICONS of the daughter in question.

- In other cases, the position isn't licensed yet at all: You may find yourself adding a topic-comment rule that takes a saturated sentence and add an as-for topic to the beginning, contributing both _topic_p_rel and the --ICONS info. Also under this heading come cases of filler-head constructions

  - Use filler-head only if there is evidence that it's actually a long-distance dependency.

- Or, you may have a position that is licensed by multiple different existing rules, which may also license different positions (e.g., pre-verbal in a V-final language). In this case, we need to elaborate the set of phrase structure rules available.

Variable property mapping

- In MT, transfer rules rewrite MRS substructures to MRS substructures, changing PRED values, the number of elementary predications, and/or the linking of variables and handles between predications (and handle constraints).

- Transfer rules cannot rewrite feature values on variables (png, tam, sf, cog-st, …)

- This is handled by "variable property mapping", or vpm.

- Each grammar is associated with a semi.vpm file which specifies the mapping between that grammar's variable properties and an interlingual set.

- Once the semi.vpm file is invoked, only properties specifically handled are passed through. Everything else is suppressed.

Variable property mapping

- If a variable property is underspecified, the LKB will find all strings that are consistent with it.

- Because we aren't handling information structure as a variable property (any more), it won't help with that.

- But: it can be very important for other variable properties and especially useful in cutting down realization ambiguity in generation.

Variable property mapping

- Most specific first

- <>: bidirectional

- >>: forward only

- <<: backward only

- *: whatever was matched

- [e]: event variable with no TENSE

- !: do not insert (forward)/match absence of property (backward)

```
E.TENSE : TENSE
    past        <> past
    present     <> pres
    future      <> fut
    real_tense  <> tensed
    untensed    <> untensed
    *           >> untensed
    untensed    << *
    untensed    << [e]
```

Using vpm to cut down on generation output

- Use case ex: You have lots of different aspect forms, but also forms that don't specify aspect. When you try to generate from one of those forms, you get all the forms out. (Also: in MT, you'll likely be getting underspecified aspect from the source language in at least some cases, so you'll want to have a default.)

- Define a type "no-aspect" which inherits from "aspect" but isn't otherwise mentioned in your grammar.

- Use the vpm to map to this "no-aspect" type just in case there is no aspect information provided:

E.ASPECT : ASPECT
imperfective <> imperfective
inceptive <> inceptive
no-aspect << *
no-aspect << [e]

Overview

- What is Information Structure?

- Topic and Focus

- Representations

- Information structure marking constructions

- Variable property mapping