Formal and informal meaning from documents through skeleton sentences

Complementing formal tag-set descriptions with intertextual semantics and vice-versa

Yves Marcoux

C. M. Sperberg-McQueen

Claus Huitfeldt

Balisage 2009, 2009-08-11

Background

- Two different approaches to semantics of markup:
  - Formal tag-set descriptions (FTSDs) [Sperberg-McQueen et al. 2000a, ...]
  - Intertextual semantics (IS) [Marcoux 2006, ...]
- They look very different
- They look very similar
- Opportunity for investigation

Overview

- Semantics
- Formal tag-set descriptions *vs* intertextual semantics
- Example
- Conclusion

[No prior familiarity assumed]

Semantics

Syntax *vs* semantics

- *Syntax* relates to the shape of documents

  Documents as bitstreams, character strings, trees, graphs, etc.

- *Semantics* relates to what documents mean

  More later...

Semantics and validation

- Syntactic validation ≈ verify well-formedness, acceptability
- Can semantics be validated?
- Semantic validation ≈ verify plausibility of contents according to predefined "business rules"

  Ex.: A person can be a buyer only if he / she is at least 18 years old

  Can detect forseeable inconsistencies in contents

- Very close to syntactic validation:
  - Can sometimes be expressed entirely by syntactic rules
  - *Schematron*
  - Easily performed on the document in its original form

More things to do with semantics (1/2)

- *Inferencing:*

  Deduce from a document facts that are not (necessarily) explicit in it:
  - Facts about the document (ex.: document not approved)
  - Facts about "the world" (ex.: person X is at least 18 years old)

  Can detect *unforeseen* inconsistencies in contents

  Another form of semantic checking

- *Reality-checking:*

  Verify that what the document says conforms to "the world"

  Yet another form of semantic checking

More things to do with semantics (2/2)

- Contrary to what we called semantic validation, inferencing and reality-checking will likely benefit from a representation of the meaning of the document in a form *different* from the document itself (sometimes *quite* different)
- That alternate representation is called the *semantics* of the document
- So the typical scenario is:

    1. “Compute” the semantics of the document
    2. Do inferencing | Do reality-checking

Who does inferencing and reality-checking?

- *Inferencing:* machine (likely)
- *Reality-checking:* human (likely)
- Notes:
  - Given appropriate representations, both could probably do both, at least in some settings (except maybe reality-checking by machine)
  - Data entry time is a natural time for reality checking *by a human*

What is the best representation of meaning?

- Two avenues: *formal* and *informal* representations

  Ex.: formal = **logic**; informal = **natural language** (NL, aka *prose*)

- Appropriate choice depends on what is to be done:
  - *Machine-performed inferences:* formal (likely)
  - *Human-performed reality-checking:* informal (likely)
- Notes:
  - Humans can work with formal representations (within limits)
  - Under certain hypotheses (AI, NLP, ...), machines can theoretically work with informal representations (but reliably?)

Formal tag-set descriptions and intertextual semantics

- Formal tag-set descriptions (FTSDs), in spite of the name, allow for both formal (logic) and informal (NL) representations of the meaning of a document

  In practice: only formal descriptions have been considered seriously

- Intertextual semantics (IS) is aimed *solely* at informal (NL) representations

  Main goal: providing semantic support to operations like human data-entry

Benefits of a well-defined semantics

- Allows inferencing and/or reality-checking (of course)
- Serves as documentation of the model itself and makes it better understandable (e.g., to programmers)
- If done at the same time as syntax, can reveal inconsistencies, flaws in the model

  Good, because early in the development process

Formal tag-set descriptions *vs* intertextual semantics

Common principles

- A *tag-set* is a set of tags occurring in either:
  - a DTD
  - a schema
  - the documents of some given context
- The comparable objects are *specifications*: an FTSD or an IS specification (ISS)
- Each of an FTSD and an ISS *for a certain tag-set* determines rules for computing a representation of the meaning of a document *conforming to that tag-set*
- These rules are used in the "compute the semantics of the document" step of the inferencing and reality-checking scenarios
- In theory, the approach is applicable to very diverse structures (databases, various markup formalisms, etc.)
- Here: XML documents only

What does the “semantics of a document” look like?

FTSDs:	**IS:**
Set (unordered) of sentences in some logical framework, usually first-order predicate logic	Sequence (ordered) of NL passages [In effect, a *single* NL passage]
The *universe of discourse* (predicates used in sentences, what they mean, types of individuals, how they map to individuals in the “real world”, etc.) is described elsewhere and taken for granted	Requirement: the NL passage must be self-contained, i.e., comprehensible without external knowledge to some given *target community* of persons
Toy example: `<doc><para>Elizabeth went to Sussex.</para></doc>`	

`is_document(d)`
`is_paragraph(p)`
`document_content(d, p)`
`paragraph_string(p,`
`  "Elizabeth went to Sussex.")`
	`This is a document:`
	`  This is a paragraph:`
	`  *Elizabeth went to Sussex.*`
	`  End of the paragraph.`
	`End of the document.`

What does a specification (FTSD or ISS) look like? (1/3)

- Key idea: *skeleton sentences*
- Zero or more skeleton sentences associated with each element type in the tag-set
  - **In IS:** Exactly one skeleton sentence per element type
- Each skeleton sentence contains one or more blanks to be filled with actual "element content" to produce a sentence or passage of the document semantics
  - **In IS:** Exactly one blank per skeleton sentence (which thus boils down to two "peritexts": "text-before" and "text-after")

What does a specification (FTSD or ISS) look like? (2/3)

- The “element content” used to fill each blank can of course be the string value of the current element, but also:
  - The string value of another element, identified by a *deictic expression* (≈ relative XPath expression) in the skeleton sentence
  - The result of an expression involving an element (current or other)
  - **In IS:** Essentially only the string value of the current element
- Restrictions in IS are *deliberate!*

  Goal: uncover and make explicit all complexities and subtleties of the model, however minute

- [Proper treatment of attributes]

What does a specification (FTSD or ISS) look like? (3/3)

For the tag-set of the toy example: `doc`, `para`

FTSD:	
For `doc` elements	`is_document( {generate-id()} )` `document_content( {generate-id()}, {generate-id(*)} )`
For `para` elements	`is_paragraph( {generate-id()} )` `paragraph_string( {generate-id()}, {string(.)} )`
ISS:	
For `doc` elements	`text-before=" This is a document: "` `text-after=" End of the document. "`
For `para` elements	`text-before=" This is a paragraph: "` `text-after=" End of the paragraph. "`

What does a specification (FTSD or ISS) look like? (4/3 :^)

The FTSD "universe of discourse"

is_document(x)	x is a document
document_content(x,y)	Document x contains y (a sequence of paragraphs — or in larger vocabularies, sections, heading, tables, and other paragraph-level objects)
is_paragraph(x)	x is a paragraph
paragraph_string(x, y)	The character-string value of the paragraph x is the string y (we will write strings enclosed in quotation marks in the conventional way)

Example

The document

```
<doc>
  <para>
    <person key="E.I.Regina">Elisabeth</person> went to
    <place key="getty:7008133">Sussex</place>.
    <person>Elizabeth</person>, on her part, went to
    <person>Sussex</person>, and told him the whole story.
  </para>
</doc>
```

FTSD “universe of discourse”

The toy example `doc` and `para` stuff +

is_personname(s)	s (typically a string of characters) is (here) a proper noun denoting a person
is_person(x)	x is a person
is_placename(s)	s (typically a string of characters) is (here) a proper noun denoting a place
is_place(x)	x is a place
denotes(s,x)	The string of character tokens s here denotes the object or individual x
person_dbkey(x, y)	The person x is denoted by the identifier y
place_dbkey(x, y)	The place x is denoted by the identifier y

FTSD

The toy example `doc` and `para` stuff +

For `person`	`is_personname( {string(.)})`
	`is_person( {concat('ref-',generate-id(.))} )`
	`denotes( {string(.)}, {concat('ref-',generate-id(.))})`

For `person/@key`	`person_dbkey( {concat('ref-',generate-id(.))}, {string(.)} )`

For `place`	`is_placename( {string(.)})`
	`is_place( {concat('ref-',generate-id(.))} )`
	`denotes( {string(.)}, {concat('ref-',generate-id(.))})`

For `place/@key`	`place_dbkey( {concat('ref-',generate-id(.))}, {string(.)} )`

IS specification

For `doc`	`text-before=" This is a document: "` `text-after=" End of the document. "`
For `para`	`text-before=" This is a paragraph: "` `text-after=" End of the paragraph. "`
For `person`	`text-before="THE PERSON NAMED "` `text-after=" @key[ (identified by the registry record {{http://my.person.registry/?@}})]"`
For `place`	`text-before="THE PLACE NAMED "` `text-after=" @key[ (identified by the registry record {{http://my.place.registry/?@}})]"`

Semantics of the document in FTSD

```
is_paragraph(id17806)
seq_pos_item(id19125-children, 1, id17806)
para_string(id17806, "Elisabeth went to Sussex.
    Elizabeth, on her part, went to Sussex,
    and told him the whole story.")
is_personname("Elisabeth")
is_person(ref-id17651)
denotes("Elisabeth", ref-id17651)
person_dbkey(ref-id17651, "E.I.Regina")
is_placename("Sussex")
is_place(ref-id19390)
denotes("Sussex", ref-id19390)
place_dbkey(ref-id19390, "getty:7008133")
is_personname("Elizabeth")
is_person(ref-id19224)
denotes("Elizabeth", ref-id19224)
is_personname("Sussex")
is_person(ref-id19558)
denotes("Sussex", ref-id19558)
```

Semantics of the document in IS

This is a document:

This is a paragraph:

THE PERSON NAMED **Elisabeth** *(identified by the registry record <http://my.person.registry/?E.I.Regina>)* **went to** *THE PLACE NAMED* **Sussex** *(identified by the registry record <http://my.place.registry/?getty:7008133>).* *THE PERSON NAMED* **Elizabeth** **, on her part, went to** *THE PERSON NAMED* **Sussex** **, and told him the whole story.**

End of the paragraph.

End of the document.

Conclusion

- Many common concepts and ideas
- Striking similarity in the type of intellectual effort that goes into writing a specification (FTSD or ISS)

  Defining target community *vs* universe of discourse

  Naming a predicate *vs* writing peritexts

- Complementary: one representation aimed at machines, the other at humans
- Conjecture: doing both at the same time for a given tag-set takes less effort than separately
- Conjecture: doing either/both at the same time as syntax results in more usable models
- Future work: experimenting with world-class tag-sets

Merci !

Thank you!

Takk !