Groundhog DAG
Representing Semantic Repetition in Literary Narratives

Greg Lessard*, Michael Levison**

*French Studies, ** School of Computing
Queen's University, Canada

June 2013

The challenge

How to represent semantic repetition in literary texts

Points to note:

- Repetition can be both *formal* (repeated words, phrases, etc.) but also *semantic* (repeated meanings, with or without repeated forms)
- Literary repetition isn't just about re-appearance of items in a string, but also about relations within textual structures
- Literary narratives aren't just stories... they can have multiple thematic levels

Two types of semantic repetition

Representational repetition

- A single event seen from the point of view of two different characters. For example, Livia sees Brutus stab Caesar, and so does Cassius.
- A single, perhaps complex, event, whose different facets are represented, perhaps in an interspersed fashion. Examples: Eisenstein's bridge scene in *October*, or the Odessa steps scene in *Battleship Potemkin*.

Class-based repetition

- Two sets of events in the same text represent instantiations of the same *topos*, or recurring narrative structure. Example: in Hebrew Bible, instances of parent favouring a younger sibling over an older one (ex. Abel/Cain, Jacob/Esau, etc.)
- Two different texts represent the same abstract plot. Example: *Pyramus and Thisbe* and *Romeo and Juliet*.

The *Groundhog Day* movie

Phil Connors, an egocentric weatherman, has been sent with his producer, Rita, and cameraman Larry, to cover the annual February 2 event at Punxsutawney, Pennsylvania, where a groundhog (Punxsutawney Phil), by seeing or not seeing his shadow, provides a prediction on the number of weeks remaining in winter. Displeased at such a lowly assignment, Connors behaves badly to all. However, on waking up the next day, he discovers that it is still February 2, and the day unfolds as it had previously. In the many subsequent iterations of the day, Connors discovers the possibilities inherent in there being no consequences to his acts, the advantages of being able to perfect the elements of a seduction by repeated trials, and finally, the value of altruism and love. At this point, after many iterations, the cycle is broken, and Phil and Rita, now in love, greet February 3.

The machinery 1: Semantic expressions

Semantic lexicon:

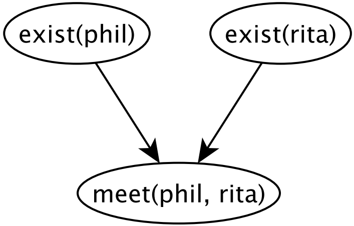
```
phil, rita :: entity
phil() = "Phil"; rita() = "Rita"

meet :: (entity, entity) -> completion
meet(x,y) = "[x] meets [y]"
```

Semantic expressions

```
meet(phil, rita)
```

Textual output: “Phil meets Rita”

The machinery 2: DAGs

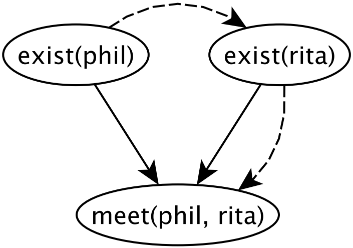


A simple DAG for Phil and Rita meeting. Nodes represent semantic expressions and arcs represent dependency relations (logical or temporal).

Characteristics: transitive, irreflexive, antisymmetric (no cycles)

Roughly equivalent to **sjuzhet** (Propp), **histoire** (Genette)

The machinery 3: Threading

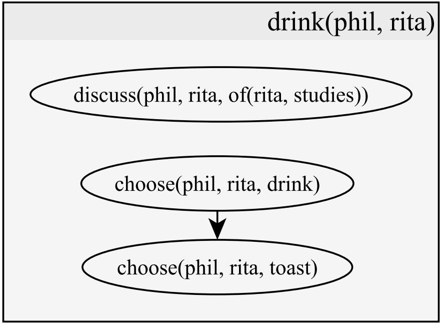


One threading for the simple DAG corresponding to “Phil exists and Rita exists and they meet.”

Characteristics: an ordered sequence of nodes in a graph. It need not follow the paths of the graph. There is no restriction on cyclicity.

Roughly equivalent to **fabula** (Propp), **récit** (Genette)

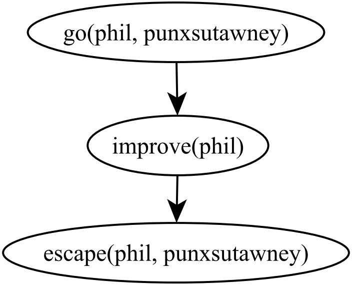
The machinery 4: DAGs and subDAGs


drink(phil, rita)
discuss(phil, rita, of(rita, studies))
choose(phil, rita, drink)
choose(phil, rita, toast)

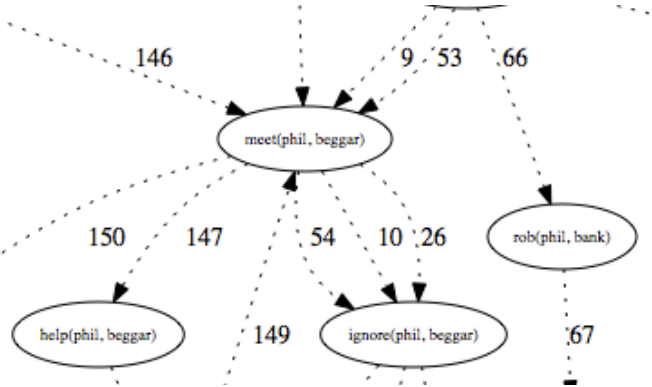
The higher-level DAG `drink(phil, rita)` can be thought of as a node which includes lower-level nodes. Some lower-level nodes may have dependencies among them, while others may not.

The highest-level DAG for *Groundhog Day*

Think of the movie as a video game, where Phil must improve in order to escape from the endless series of February 2nds.



Part of the low-level threading for *Groundhog Day*, with repetition



Parametrized subDAGs: two representations

Phil reports the appearance of the groundhog multiple times, in different ways: he is first sarcastic, then confused, then professional, then learned, then poetic, and finally profound. This may be represented ‘atomistically’:

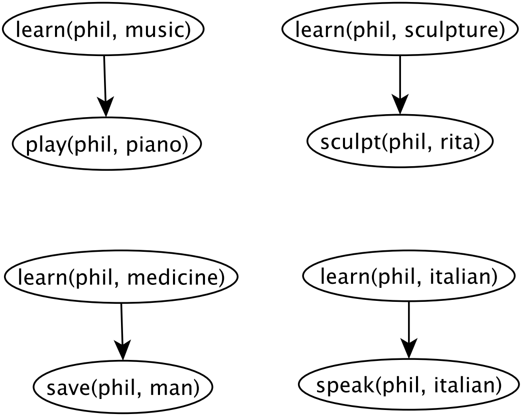
```
describe(phil, groundhog, ironic)
describe(phil, groundhog, confused)
...
```

or, better, more abstractly:

```
X = (sarcastic, confused, professional,
     learned, poetic, profound)

describe(phil, groundhog, X)
```

Phil's new skills and their applications

As Phil improves, he learns new skills (French poetry, piano, Italian, sculpture,...) and subsequently applies them.



learn(phil, music)

play(phil, piano)

learn(phil, sculpture)

sculpt(phil, rita)

learn(phil, medicine)

save(phil, man)

learn(phil, italian)

speak(phil, italian)

Capturing Phil's development

Over the course of the movie, Phil learns new things, and in various ways becomes a better person. For example, he learns music and then performs in public. We may capture this development by:

```
know(phil, music, 0)  {initial state}
...
know(phil, music, 1)  {end state}
```

This framework can be applied to other skills and to his moral development:

```
X = (music, medicine, sculpture, italian, altruism)
if (X < 1)
    (learn(phil, X), apply(phil, X))
```

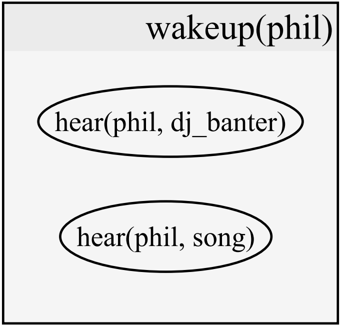
Parallel DAGs: Phil's seduction techniques

Phil tries to seduce both Nancy and Rita by learning about them and using this knowledge. This can be represented schematically as:

```
experiment(x,y) =
   slist(
   meet(x,y)
   learn(x, of(y, characteristics))
   ...
   )
```

Narrative shortcuts: variant threadings within a DAG

In threads 5, 22, 50, 103, 119, 122 and 135, Phil's waking up is followed by his hearing of a song but in thread 36, by DJ banter only. In threads 6, 23, 51 and 104, banter follows song, but in threads 120 and 123, the song is followed by suicide attempts. These are narrative shortcuts through the `wakeup` DAG.



Some preliminary conclusions

Semantic repetition...

- ... exists at the intersection of the text and the reader
- ... depends sometimes on abstract structures 'behind' the text
- ... may be captured by a combination of DAGs and threading
- ... may involve parametrized higher-level frameworks
- ... once activated, may be recalled by mention of only parts of a higher-level DAG.