semanticizest Documentation

Release 0.2.1

NLeSC/UvA

January 30, 2015

Contents

semanticizest (Semanticizer, STandalone) is a package for entity linking, aka semantic linking, semanticizing, or wikification: You feed it text, and it outputs links to pertinent Wikipedia concepts. You can use these links as a "semantic representation" of the text for NLP or machine learning, or just to provide some links to background info on the Wikipedia.

CHAPTER 1

Quick usage

First we need to create a model for the semanticizer. The following command will download and read a wikipedia dump (in this case the Limburgish wiki, just because it's small) and subsequently create and store the corresponding model. If you want the English Wikipedia, replace `liwiki` by `enwiki` and be prepared to run `parse_wikidump` overnight on a server.

```
python -m semanticizest.parse_wikidump --download liwiki liwiki.model
```

Fire up a Python interpreter and import the required modules:

```
>>> import re
>>> from semanticizest import Semanticizer
```

Load the model from disk:

```
>>> sem = Semanticizer('liwiki.model')
```

Set up a piece of sample text and tokenize it:

```
>>> text = """'ne Donjon is 'ne zjwaore verdedigingstaore,
... meistal geboewd op 'ne hoage heuvel, de opperhaof,
... dae deil oetmaak van 'n motte."""
>>> tokens = re.findall('\w+', text)
```

Feed the tokens to the semanticizer to get the entity link candidates:

```
>>> for cand in sem.all_candidates(tokens):
...     print(cand)
(7, 8, u'Taore (boewwerk)', 1.0)
(13, 14, u'Heuvel', 1.0)
(15, 16, u'Opperhaof', 1.0)
(21, 22, u'Motte', 1.0)
```

As we can see, it finds four entity candidates in this short text. The first entity found is 'Taore (boewwerk)', corresponding to the seventh token: 'verdedigingstaore'.

CHAPTER 2

Contents

2.1 API Reference

The semanticizest API is not stable and may change without notice.

2.1.1 Semanticizest

class semanticizest.**Semanticizer**(*fname*)

Entity linker.

This is the main class for using Semanticizest. It's a handle on a statistical model that lives on disk.

Parameters **fname** : string

Filename of the stored model from which to load the Wikipedia statistics. Loading is lazy; the underlying file should not be modified while any Semanticizer is using it.

Methods

all_candidates(s)	Retrieve all candidate entities from a piece of text.

all_candidates(*s*)

Retrieve all candidate entities from a piece of text.

Parameters **s** : {string, iterable over string}

Tokens. If a string, it will be tokenized using a naive heuristic.

Returns **candidates** : iterable over (int, int, string, float)

Candidate entities are 4-tuples of the indices *start* and *end* (both in tokenized input, and both start at 1), *target entity* (title of the Wikipedia article) and *probability* (commonness.)

2.2 Algorithms & implementation details

2.2.1 Entity linking statistics

Many approaches to entity linking exist; this section gives a short overview of what is possible with semanticizest. It is intended as a reference guide; you don't need to memorize all this to use semanticizest.

All entity linking strategies use statistics gathered from Wikipedia. These statistics concern:

- *Wikilinks*, or "links" for short. These are the familiar blue and red hyperlinks between Wikipedia articles. External links (URLs) are not wikilinks, so they are not involved in entity linking.

  A link has a *target* and an *anchor text*, or anchor for short. The target is a page title, which may (blue link) or may not (red link) refer to an actual page in the Wikipedia. The anchor is the text of the link as it appears in the page where the link was found. Often, but not always, the target page's title and the anchor coincide. By picking up all the anchors for an entity (target page), semanticizest knows that "Napoleon Bonaparte" is common way of referring to what Wikipedia calls "Napoleon" (the title of the page about Bonaparte).

- *Entities*. An entity is any link target that lives/would live in the Wikipedia "main" namespace, the encyclopedic content, if the link is/were blue.

  Entities are represented by the titles or URLs of the links' targets. That means all articles in a Wikipedia are potential entities, but pages that are never linked to are not considered entities. However, all "red links" (links to non-existent pages) are considered entities, so there is no one-to-one mapping between articles and entities.

Note: Concepts that need explanation/clarification:

- sense

Formulas for determining link candidates and ranking candidates. These two steps correspond exactly to the baseline retrieval model of [Odijk2013].

- *Prior probability* determines how likely it is that anchor text a links to Wikipedia article w, also known as *commonness*, "the extent to which each sense is well-known" [Medelyan2008].

  $$P_{prior}(w|a) = \frac{|lnk_{a,w}|}{|lnk_a|}$$

  here w is a Wikipedia article, a is the anchor text, $|.|$ is (multi)set cardinality, $lnk_{a,w}$ is the multiset of links with anchor text a and target w, and lnk_a is the multiset of links with anchor text a.

- *Keyphraseness* "is defined as the number of Wikipedia articles that use it as an anchor, divided by the number of articles that mention it at all." [Milne2008] "[is an] estimate of the probability that a phrase is selected as a keyphrase for a document" [Mihalcea2007], "the probability of being a [link] candidate" [Medelyan2008]

  $$P_{keyphrase}(a) = \frac{DF(lnk_a)}{DF(a)}$$

  where a is the anchor text (an n-gram of one or more terms), $DF(.)$ denotes the document frequency in Wikipedia, $DF(lnk_a)$ is the number of Wikipedia articles where a is used as the anchor text of a link and $DF(a)$ the number of Wikipedia articles containing the text a at all.

- *Link probability* is the same as 'keyphraseness', except that link probability uses term frequencies instead of document frequencies ("we determine this probability based on all occurrences, also including multiple occurrences in an article" [Meij2012]):

  $$P_{link}(a) = \frac{|lnk_a|}{|a|}$$

- *Sense probability* "an estimate for the probability that an n-gram is used as an anchor linking to Wikipedia article w" [Odijk2013]:

$$P_{sense}(w|a) = \frac{|lnk_{a,w}|}{|a|}$$

2.2.2 Data structures

Note: Fill me in with enough details of the SQLite tables and count-min sketches to explain potential wtf's. We don't need to repeat the database schema here because it's an implementation detail.

CHAPTER 3

Indices and tables

- *genindex*
- *modindex*
- *search*

CHAPTER 4

Developed by

Figure 4.1: ILPS, University of Amsterdam

Figure 4.2: Netherlands eScience Center

Bibliography

[Mihalcea2007] Mihalcea, Rada, and Andras Csomai. “Wikify!: linking documents to encyclopedic knowledge.” Proceedings of the sixteenth ACM conference on Conference on information and knowledge management. ACM, 2007.

[Medelyan2008] Medelyan, Olena, Ian H. Witten, and David Milne. “Topic indexing with Wikipedia.” Proceedings of the AAAI WikiAI workshop. 2008.

[Milne2008] Milne, David, and Ian H. Witten. “Learning to link with wikipedia.” Proceedings of the 17th ACM conference on Information and knowledge management. ACM, 2008.

[Meij2012] Meij, Edgar, Wouter Weerkamp, and Maarten de Rijke. “Adding semantics to microblog posts.” ACM, 2012.

[Odijk2013] Odijk, Daan, Edgar Meij, and Maarten De Rijke. “Feeding the second screen: Semantic linking based on subtitles.” OAIR, 2013.

S

A

S