Distributional semantics practical 2

Aurélie Herbelot

1 Semantic anomaly: the problem

Semantic anomaly is a graded concept. As opposed to syntactic errors, which are (usually) clearly identifiable, they may not result in good inter-annotator agreement. See the following:

- *The cats plays in the garden.
- ?Parliamentary tomatoes are flying.

The first example is clearly an agreement error. The second is semantically odd, but some people will find an interpretation for this sentence (e.g. there was a fight in parliament and the representatives started throwing tomatoes at each other).

Still, some phrases are more transparent than others. Can distributional semantics quantify this?

anomalous	acceptable
biological coup	biological longevity
cultural starch	ethical nationalism
dry expiration	huge limit
exact autumn	institutional minimum
industrial horseback	optional licensing
innovative temper	printed anecdote
naval overdose	reasonable tension
optional coma	royal salad
printed fear	spectacular sauce
vulnerable quotation	vulnerable dinosaur

Exercises:

- Look at the above list of anomalous and acceptable adjective-noun phrases (ANs). Check your human intuition: what makes those ANs acceptable or not?
- Think about the relative position of the ANs elements in the semantic space. Try to get a feel for what kind of vectors might be produced via additive and multiplicative composition.
- Develop a hypothesis for what might distinguish the two classes in the semantic space.

2 Verifying your hypothesis

In order to verify your hypothesis, you will need to write a script that utilises some of the main components of the DISSECT system: the similarity measure and the composition operations. You can check the use of the similarity function in *utils/similarity.py*.

The additive and multiplicative composition operations require the following modules to be imported at the top of your program:

```
from composes.composition.weighted_additive import WeightedAdditive
from composes.composition.multiplicative import Multiplicative
```

Then, the functions can be defined as:

```
add = WeightedAdditive(alpha = 1, beta = 1)
mult = Multiplicative()
```

You then use them over actual elements of your semantic space by writing e.g.:

```
composed_space = add.compose([(word1, word2, "_composed_")],
 my_space)
```

where *_composed_* is a new space containing your composed vectors.

More information can be found at `http://clic.cimec.unitn.it/composes/toolkit/composing.html`.

3 A solution

Update your git repository using *git pull* from the command line. You will find a new program in the utils/ folder, called *anomaly.py*. This script computes some of the measures used by Vecchi et al in their 2011 paper: *(Linear) Maps of the Impossible: Capturing semantic anomalies in distributional space* (`http://clic.cimec.unitn.it/marco/publications/vbz-impossible-and-disco11.pdf`). It can be run as follows:

```
python anomaly.py ../spaces/wikipedia.pkl
../data/semantic-anomaly.examples
```

The data used by Vecchi et al can be found here: `http://www.vecchi.com/eva/resources/vbz2011_deviant_AN_testset.txt` and `http://www.vecchi.com/eva/resources/vbz2011_acceptable_AN_testset.txt`.